

Національний технічний університет України
«Київський політехнічний інститут ім. І. Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Лабораторна робота №3
Тема: «Проведення трьохфакторного експерименту з використанням
Лінійного рівняння регресії.»

Виконав:
студент групи ІО-92
Уткін Владислав Антонович
Номер в списку групи: 19
Перевірив:
ас. Регіда П. Г.

Київ 2021

Лабораторна робота № 3

«Проведення трьохфакторного експерименту з використанням Лінійного рівняння регресії.»

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

Завдання на лабораторну роботу:

1. Скласти матрицю планування для дробового трьохфакторного експерименту. Провести експеримент в усіх точках факторного простору, повторивши N експериментів, де N – кількість експериментів (рядків матриці планування) в усіх точках факторного простору – знайти значення функції відгуку Y. Значення функції відгуку знайти у відповідності з варіантом діапазону, зазначеного далі (випадковим чином).
2. Знайти коефіцієнти лінійного рівняння регресії. Записати лінійне рівняння регресії.
3. Провести 3 статистичні перевірки.
4. Написати комп'ютерну програму, яка усе це виконує.

№ варіанта	X1		X2		X3	
	min	max	min	max	min	max
219	15	45	-70	-10	15	30

```
import numpy as np
import random
from numpy.linalg import solve
from scipy.stats import f, t
from functools import partial

x1_min, x1_max = 15, 45
x2_min, x2_max = -70, -10
x3_min, x3_max = 15, 30

aver_xmin = (15 - 70 + 15) / 3
aver_xmax = (45 - 10 + 30) / 3

y_max = 200 + int(aver_xmax)
y_min = 200 + int(aver_xmin)
```

```

y_min_max = [y_min, y_max]

def experiment(n, m):
    y = np.random.randint(*y_min_max, size=(n, m))

    x_plan = np.array([[1, -1, -1, -1],
                        [1, -1, 1, 1],
                        [1, 1, -1, 1],
                        [1, 1, 1, -1]])

    xn = np.array([[1, 15, 30, 15],
                   [1, 15, 80, 45],
                   [1, 45, 30, 45],
                   [1, 45, 80, 15]])

    print("\nМатриця планування")
    labels_table = ["x0", "x1", "x2", "x3"] + ["y{ }".format(i + 1) for i in range(4)]
    rows_table = [list(xn[i]) + list(y[i]) for i in range(3)]
    print((" " * 4).join(labels_table))
    print("\n".join([" " * 4 + join(map(lambda j: "{:<5}".format(j), rows_table[i])) for i in range(len(rows_table))]))
    return xn, y, x_plan

def Regression(x, b):
    y = sum([x[i]*b[i] for i in range(len(x))])
    return y

def coefficient(x, y_avarg, n):
    mx1 = sum(x[:, 1]) / n
    mx2 = sum(x[:, 2]) / n
    mx3 = sum(x[:, 3]) / n
    my = sum(y_avarg) / n

    a12 = sum([x[i][1] * x[i][2] for i in range(len(x))]) / n
    a13 = sum([x[i][1] * x[i][3] for i in range(len(x))]) / n
    a23 = sum([x[i][2] * x[i][3] for i in range(len(x))]) / n
    a11 = sum([i ** 2 for i in x[:, 1]]) / n
    a22 = sum([i ** 2 for i in x[:, 2]]) / n
    a33 = sum([i ** 2 for i in x[:, 3]]) / n
    a1 = sum([y_avarg[i] * x[i][1] for i in range(len(x))]) / n
    a2 = sum([y_avarg[i] * x[i][2] for i in range(len(x))]) / n
    a3 = sum([y_avarg[i] * x[i][3] for i in range(len(x))]) / n

    X = [[1, mx1, mx2, mx3], [mx1, a11, a12, a13], [mx2, a12, a22, a23], [mx3, a13, a23, a33]]
    Y = [my, a1, a2, a3]
    B = [round(i, 2) for i in solve(X, Y)]

    print("\nРівняння регресії")
    print(f'{B[0]} + {B[1]}*x1 + {B[2]}*x2 + {B[3]}*x3')

    return B

def count_dispersion(y, y_avarg, n, m):
    y_var = np.var(y, axis=1)

```

```

return y_var

def check_criter_Cochran(y, y_avarg, n, m):
    S_kv = count_dispersion(y, y_avarg, n, m)
    Gp = max(S_kv) / sum(S_kv)
    print("\nПеревірка за критерієм Кохрена")
    return Gp

# оцінки коефіцієнтів
def bs(x, y, y_avarg, n):
    res = [sum(1 * y for y in y_avarg) / n]
    for i in range(3): # 4 - ксть факторів
        b = sum(j[0] * j[1] for j in zip(x[:, i], y_avarg)) / n
        res.append(b)
    return res

def significance_student(x, y, y_avarg, n, m):

    S_kv = count_dispersion(y, y_avarg, n, m)
    kv_aver = sum(S_kv) / n

    s_Bs = (kv_aver / n / m) ** 0.5
    Bs = bs(x, y, y_avarg, n)
    ts = [abs(B) / s_Bs for B in Bs]
    return ts

def adequacy_Fishera(y, y_avarg, y_new, n, m, d):
    S_ad = m / (n - d) * sum([(y_new[i] - y_avarg[i])**2 for i in range(len(y))])
    S_kv = count_dispersion(y, y_avarg, n, m)
    S_kv_aver = sum(S_kv) / n

    return S_ad / S_kv_aver

def Cochran(f1, f2, q=0.05):
    q1 = q / f1
    fisher_value = f.ppf(q=1 - q1, dfn=f2, dfd=(f1 - 1) * f2)
    return fisher_value / (fisher_value + f1 - 1)

def main(n, m):
    f1 = m - 1
    f2 = n
    f3 = f1 * f2
    q = 0.05

    student = partial(t.ppf, q=1-0.025)
    t_student = student(df=f3)

```

```

G_kr = Cochran(f1, f2)

x, y, x_plan = experiment(n, m)
y_avarg = [sum(y[i][j] for j in range(4)) / 4 for i in range(4)]

B = coefficient(x, y_avarg, n)

Gp = check_criter_Cochran(y, y_avarg, n, m)
print(f'Gp = {Gp}')
if Gp < G_kr:
    print(f'Дисперсії однорідні {1-q}')
else:
    print("Збільшити кількість дослідів")
    m += 1
    main(n, m)

ts = significance_student(x_plan[:, 1:], y, y_avarg, n, m)
print(f'\nКритерій Стьюдента:\n', ts)
res = [t for t in ts if t > t_student]
final_k = [B[ts.index(i)] for i in ts if i in res]
print('Статистично незначущі {}, тому виключаємо їх з рівняння.'.format([i for i in B if i not in final_k]))

y_regr = []
for j in range(n):
    y_regr.append(Regression([x[j][ts.index(i)] for i in ts if i in res], final_k))

print(f'\n"y" => {final_k}')
print(y_regr)

d = len(res)
f4 = n - d
F_p = adequacy_Fishera(y, y_avarg, y_regr, n, m, d)

fisher = partial(f.ppf, q=1 - 0.05)
f_t = fisher(dfn=f4, dfd=f3)

print(f'\nПеревірка адекватності за критерієм Фішера')
print('Fp =', F_p)
print('F_t =', f_t)
if F_p < f_t:
    print('Математична модель є адекватною')
else:
    print('Математична модель не є адекватною')

if __name__ == '__main__':
    main(4, 4)

```

```
C:\Anaconda3\envs\labs\python.exe C:/Users/Влад/PycharmProjects/labs/laba3.py
```

Матриця планування

x0	x1	x2	x3	y1	y2	y3	y4
1	15	30	15	214	201	211	219
1	15	80	45	218	219	193	210
1	45	30	45	218	215	210	206

Рівняння регресії

$212.7 + -0.03 \cdot x_1 + -0.07 \cdot x_2 + 0.07 \cdot x_3$

Перевірка за критерієм Кохрена

$G_p = 0.4375762505083367$

Дисперсії однорідні 0.95

Критерій Стюдента:

[95.88119597375845, 0.22815275663000228, 0.7414964590475074, 0.45630551326000457]

Статистично незначущі [-0.03, -0.07, 0.07], тому виключаємо їх з рівняння.

"y" => [212.7]

[212.7, 212.7, 212.7, 212.7]

Перевірка адекватності за критерієм Фішера

$F_p = 0.7302263792869689$

$F_t = 3.490294819497605$

Математична модель адекватна

Process finished with exit code 0

Контрольні запитання:

1. Що називається дробовим факторним експериментом?

Дробовий факторний експеримент – це частина ПФЕ, який мінімізує число дослідів, за рахунок тієї інформації, яка не дуже істотна для побудови лінійної моделі.

2. Для чого потрібно розрахункове значення Кохрена?

Критерій Кохрена — використовують для порівняння трьох і більше виборок однакового обсягу n.

Якщо вибіркві дисперсії отримані за вибірками однакових обсягів, для їх порівняння використовують більш зручний і точний критерій Кохрена.

Кохрена досліджував розподіл максимальної вибіркової дисперсії до суми всіх дисперсій.

3. Для чого перевіряється критерій Стюдента?

Критерій Стюдента - загальна назва для статистичних тестів, в яких статистика критерію має розподіл Стюдента. Найбільш часто t-критерії застосовуються для перевірки рівності середніх значень у двох вибірках. Тому перед застосуванням критерію Стюдента рекомендується виконати перевірку нормальності.

4. Чим визначається критерій Фішера і як його застосовувати?

Критерій Фішера застосовується для перевірки рівності дисперсій двох вибірок. Його відносять до критеріїв розсіювання. Критерій Фішера заснований на додаткових припущеннях про незалежність і нормальності вибірок даних. Перед його застосуванням рекомендується виконати перевірку нормальності.

Висновок:

В процесі лабораторної роботи було проведено дробовий трьохфакторний експеримент. Після чого склали матрицю планування, знайшли коефіцієнти рівняння регресії та провели перевірку однорідності дисперсій за критерієм Кохрена, нуль-гіпотезу за критерієм Стюдента та адекватність моделі за критерієм Фішера.