**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информатика»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4

по дисциплине **«Математическое моделирование сложных систем»**

на тему: **«Планирование эксперимента»**

Выполнил: студент гр. ИП-31

Кузнецова Е. А.

Принял: доцент

Трохова Т. А.

Дата сдачи отчета: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата допуска к защите: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Гомель 2024

**Цель работы:** получить навыки разработки планов полного факторного эксперимента в СКМ, научиться выполнять регрессионный и корреляционный анализ результатов эксперимента.

**Часть1**

1. Ввести исходные данные для построения ПФЭ:

- число факторов - 3;

- область планирования;

- функцию отклика.

Индивидуальные задания приведены в приложении А.

1. Составить матрицу ПФЭ.
2. Найти среднее значение функции отклика для пяти повторов опытов.
3. Построить регрессионную модель без учета взаимодействия факторов и определить значения коэффициентов регрессии.
4. Получить расчетные значения функции отклика по регрессионной модели
5. Вычислить абсолютную и относительную ошибки моделирования
6. Построить регрессионную модель с учетом взаимодействия факторов и определить значения коэффициентов регрессии.
7. Получить расчетные значения функции отклика по регрессионной модели
8. Вычислить абсолютную и относительную ошибки моделирования

10) Сделать вывод об адекватности модели

Индивидуальные варианты исходных данных

Вариант 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3,651 | 3,605 | 3,653 | 3,592 | 3,627 |
| 6,547 | 6,514 | 6,535 | 6,562 | 6,581 |
| 4,761 | 4,793 | 4,816 | 4,792 | 4,801 |
| 9,515 | 9,566 | 9,534 | 9,552 | 9,528 |
| 5,828 | 5,847 | 5,842 | 5,905 | 5,886 |
| 13,041 | 13,081 | 13,051 | 13,089 | 13,063 |
| 8,364 | 8,371 | 8,338 | 8,365 | 8,366 |
| 25,575 | 25,563 | 25,611 | 25,578 | 25,534 |

**Код программы:**

import numpy as np

from numpy.linalg import inv

# Число факторов

num\_factors = 3

levels\_per\_factor = 2

# Матрица планирования

num\_experiments = levels\_per\_factor \*\* num\_factors

planning\_matrix = np.zeros((num\_experiments, num\_factors), dtype=int)

for i in range(num\_experiments):

for j in range(num\_factors):

planning\_matrix[i][j] = ((i // (levels\_per\_factor \*\* j)) % levels\_per\_factor) \* 2 - 1

# Функция отклика

response\_function = np.array([

3.651, 3.605, 3.653, 3.592, 3.627,

6.547, 6.514, 6.535, 6.562, 6.581,

4.761, 4.793, 4.816, 4.792, 4.801,

9.515, 9.566, 9.534, 9.552, 9.528,

5.828, 5.847, 5.842, 5.905, 5.886,

13.041, 13.081, 13.051, 13.089, 13.063,

8.364, 8.371, 8.338, 8.365, 8.366,

25.575, 25.563, 25.611, 25.578, 25.534

])

# Среднее значение функции отклика для пяти повторов опытов

average\_response = np.mean(response\_function.reshape(-1, 5), axis=1)

# Матрица планирования без учета взаимодействия факторов

X\_matrix = np.column\_stack((np.ones(len(planning\_matrix)), planning\_matrix))

# Вычисление коэффициентов регрессии

coefficients = inv(X\_matrix.T @ X\_matrix) @ X\_matrix.T @ average\_response

# Значения факторов для расчета вектора Q

# np.ones создает массив из единиц длиной, равной количеству строк в planning\_matrix. Этот массив представляет собой столбец, состоящий из единиц.

X\_values = np.column\_stack((np.ones(len(planning\_matrix)), planning\_matrix)) #np.column\_stack(), которая объединяет массив из единиц (первый столбец) с planning\_matrix (остальные столбцы).

# Расчет вектора Q по регрессионной модели

Q\_vector = X\_values @ coefficients

# Расчет абсолютной ошибки модели

W = np.max(np.abs(average\_response - Q\_vector))

# Расчет относительной ошибки модели в процентах

w = (W / np.max(np.abs(average\_response))) \* 100

print("Матрица ПФЭ:\n", planning\_matrix)

print("\nСреднее значение функции отклика для пяти повторов опытов:")

print(average\_response)

print("\nКоэффициенты регрессии:")

print("b0 =", coefficients[0])

print("b1 =", coefficients[1])

print("b2 =", coefficients[2])

print("b3 =", coefficients[3])

print("\nВектор Q по регрессионной модели:", Q\_vector)

print("\nАбсолютная ошибка модели (W):", W)

print("Относительная ошибка модели в процентах (w):", w)

# Добавление столбцов для взаимодействия всех трех факторов

interaction\_columns\_extended = np.column\_stack((

planning\_matrix[:, 0] \* planning\_matrix[:, 1],

planning\_matrix[:, 0] \* planning\_matrix[:, 2],

planning\_matrix[:, 1] \* planning\_matrix[:, 2],

planning\_matrix[:, 0] \* planning\_matrix[:, 1] \* planning\_matrix[:, 2]

))

# Расширенная матрица планирования с учетом взаимодействия всех трех факторов

extended\_X\_matrix\_extended = np.column\_stack((X\_matrix, interaction\_columns\_extended))

# Вычисление коэффициентов расширенной регрессионной модели

extended\_coefficients\_extended = inv(extended\_X\_matrix\_extended.T @ extended\_X\_matrix\_extended) @ extended\_X\_matrix\_extended.T @ average\_response

# Вывод коэффициентов расширенной регрессионной модели с учетом взаимодействия всех трех факторов

print("\nКоэффициенты расширенной регрессионной модели (с учетом взаимодействия всех трех факторов):")

print("b0 =", extended\_coefficients\_extended[0])

print("b1 =", extended\_coefficients\_extended[1])

print("b2 =", extended\_coefficients\_extended[2])

print("b3 =", extended\_coefficients\_extended[3])

print("b4 =", extended\_coefficients\_extended[4])

print("b5 =", extended\_coefficients\_extended[5])

print("b6 =", extended\_coefficients\_extended[6])

print("b7 =", extended\_coefficients\_extended[7])

# Значения факторов для расчета расширенного вектора Q с учетом взаимодействия всех трех факторов

extended\_X\_values\_extended = np.column\_stack((np.ones(len(planning\_matrix)), planning\_matrix, interaction\_columns\_extended))

# Расчет расширенного вектора Q по расширенной регрессионной модели с учетом взаимодействия всех трех факторов

extended\_Q\_vector\_extended = extended\_X\_values\_extended @ extended\_coefficients\_extended

print("\nСреднее значение функции отклика для пяти повторов опытов:\n", average\_response)

# Вывод расчетных значений функции отклика по расширенной регрессионной модели с учетом взаимодействия всех трех факторов

print("\nРасчетные значения функции отклика по расширенной регрессионной модели (с учетом взаимодействия всех трех факторов)\n:", extended\_Q\_vector\_extended)

# Расчет абсолютной ошибки модели с учетом взаимодействия всех трех факторов

extended\_W\_extended = np.max(np.abs(average\_response - extended\_Q\_vector\_extended))

# Расчет относительной ошибки модели в процентах с учетом взаимодействия всех трех факторов

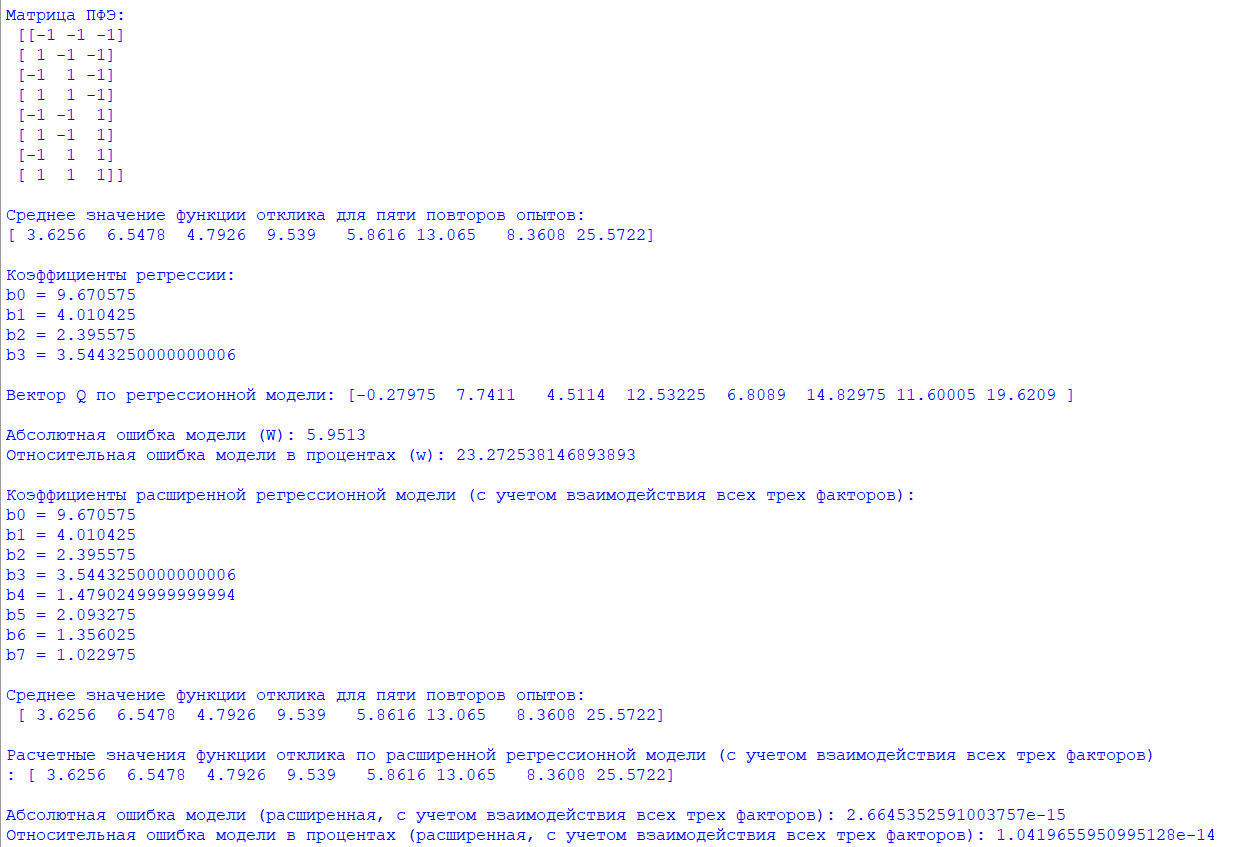
extended\_w\_extended = (extended\_W\_extended / np.max(np.abs(average\_response))) \* 100

# Вывод результатов с учетом взаимодействия всех трех факторов

print("\nАбсолютная ошибка модели (расширенная, с учетом взаимодействия всех трех факторов):", extended\_W\_extended)

print("Относительная ошибка модели в процентах (расширенная, с учетом взаимодействия всех трех факторов):", extended\_w\_extended)

**Результат выполнения:**



**Часть 2**

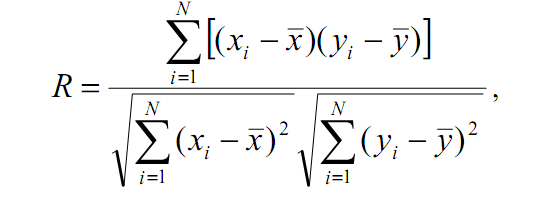
***Задание 2***

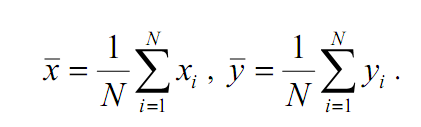
1. Создать с использованием программы «Блокнот» файлы значений двух наборов данных X и Y, приведенный в таблице 2 (данные лучше представить в виде строк).
2. Считать файлы в вектора X и Y .
3. Построить диаграмму рассеивания (простой двумерный график), не соединяя точки отрезками прямых.
4. Найти коэффициент парной корреляции по формуле.
5. Сделать вывод о наличии корреляционной связи.

**Описание задачи**

Пусть в результате эксперимента получены два набора данных х и y, характеризующие параметры технического объекта. Нужно определить, являются ли эти параметры взаимосвязанными. Простейшей, но информативной характеристикой связи двух величин X и Y является коэффициент корреляции.

Коэффициент корреляции вычисляется по формуле:





В зависимости от полученного значения R можно сделать вывод о взаимосвязи заданных параметров:

R=0,3 и менее – связь слабая;

R=0,3 – 0,5 – связь умеренная;

R=0,5 – 0,7 – связь существенная;

R=0,7 – 0,9 – связь сильная;

R=0,9 – 1 – связь очень сильная.

Таблица 2

|  |  |  |
| --- | --- | --- |
|  |  | Значения параметров |
| 2 | X | 9.69 14.87 9.15 14.88 11.98 14.70 11.44 3.03 8.50 10.34 15.03 3.37 9.83 13.99 2.95 2.83 13.85 8.53 |
| Y | 16.43 2.60 13.96 3.25 13.89 2.64 12.44 4.75 11.59 12.27 2.09 14.18 8.78 7.24 16.13 9.34 8.01 14.39 |

**Код программы:**

import numpy as np

import matplotlib.pyplot as plt

from sklearn.linear\_model import LinearRegression

# Часть 2

# Данные, предоставленные в задании

X = np.array([18.99, 1.92, 15.60, 19.01, 6.44, 14.181, 18.23, 19.18, 20.07, 9.68, 7.78, 9.40, 0.01, 17.01, 1.97, 15.04, 3.38, 1.06, 0.40])

Y = np.array([9.71, 2.61, 4.12, 12.58, 5.56, 3.48, 3.474, 14.56, 0.34, 2.27, 0.13, 9.65, 6.21, 1.36, 10.48, 9.86, 8.03, 0.80, 3.89])

# Вычисляем среднее значение X и

mean\_X = np.mean(X)

mean\_Y = np.mean(Y)

# Вычисляем отклонения от среднего значения

dev\_X = X - mean\_X

dev\_Y = Y - mean\_Y

# Вычисляем коэффициент корреляции

R = np.sum(dev\_X \* dev\_Y) / np.sqrt(np.sum(dev\_X\*\*2) \* np.sum(dev\_Y\*\*2))

print(f"Коэффициент корреляции R равен {R}")

# Интерпретируем корреляцию

if R <= 0.3:

print("Корреляция слабая.")

elif R <= 0.5:

print("Корреляция умеренная.")

elif R <= 0.7:

print("Корреляция существенная.")

elif R <= 0.9:

print("Корреляция сильная.")

else:

print("Корреляция очень сильная.")

# Диаграмма рассеивания

plt.scatter(X, Y)

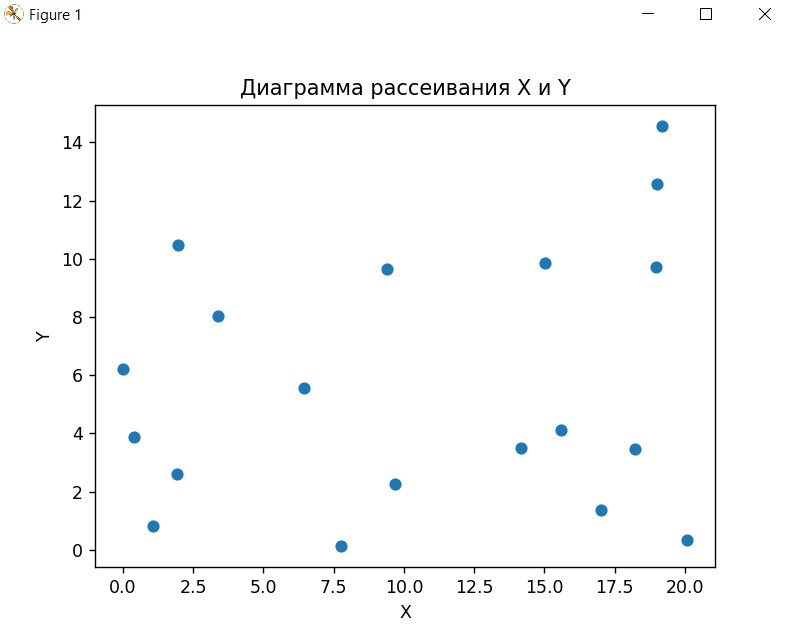
plt.title('Диаграмма рассеивания X и Y')

plt.xlabel('X')

plt.ylabel('Y')

plt.show()

**Результат выполнения:**



**Вывод:** в результате выполнения лабораторной работы мноюполучены навыки разработки планов полного факторного эксперимента в СКМ, научиться выполнять регрессионный и корреляционный анализ результатов эксперимента.