МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Кафедра «Информатика»

**Отчет по заданию №3**

**по дисциплине**

**«Численные методы»**

Выполнил: студент гр. БЭИ2202

Кулешов А. С.

Вариант 16.

Проверил: доц. каф. «Информатика»

Мацкевич А. Г.

Москва, 2023 г.

* 1. **Задание для решения задачи аппроксимации**

Для решения задачи аппроксимации методом наименьших квадратов выберем функцию y(x), заданную следующей таблицей:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1.2 | 1.4 | 1.6 | 1.8 | 2.0 |
|  | 1,12 | 1,506 | 0.526 | -0.82 | -1.66 |

* 1. **Линейная аппроксимация:**

Вычислить и записать в табл. 1-3 элементы матрицы Грамма и столбец свободных членов:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 0 | 1.2 | 1.12 | 1.344 | 1.44 |
| 1 | 1.4 | 1.506 | 2.1084 | 1.96 |
| 2 | 1.6 | 0.526 | 0.8416 | 2.56 |
| 3 | 1.8 | -0.82 | -1.476 | 3.24 |
| 4 | 2.0 | -1.66 | -3.32 | 4 |
|  | 8 | 0.672 | -0.502 | 13.2 |

составить системы нормальных уравнений:

для линейной функции **P1(x) = А0+А1\*x** система нормальных уравнений примет вид (линейная аппроксимация):

**5\*А0+8\*А1 = 0.672**

**8\*А0+13.2\*А1 = -0.502**

решить систему уравнений:

получим коэффициенты **А0 = 6.4432** и **А1 = -3.943**, тогда полином первой степени будет таким:

**P1(x) = 6.4432** **- 3.943\*x**

* 1. **Линейная аппроксимация:**

Решу задачу аппроксимации при помощи ЯП Python.

import numpy as np

import matplotlib.pyplot as plt

# Задаем исходные данные

x = np.array([1.2, 1.4, 1.6, 1.8, 2.0])

y = np.array([1.12, 1.506, 0.526, -0.82, -1.66])

# Создаем список степеней многочлена

degrees = [1, 2, 3, 4]

# Выполняем аппроксимацию наименьших квадратов для каждой степени многочлена

for degree in degrees:

    coefficients = np.polyfit(x, y, degree)

    poly = np.poly1d(coefficients)

    print(coefficients)

    # Генерируем значения для построения графика

    x\_vals = np.linspace(x.min(), x.max(), 100)

    y\_vals = poly(x\_vals)

    # Вычисляем среднеквадратичное отклонение

    y\_pred = poly(x)

    rmse = np.sqrt(np.mean((y - y\_pred) \*\* 2))

    # Строим график и выводим RMSE

    plt.plot(x\_vals, y\_vals, label=f'Степень: {degree}, СКО: {rmse:.2f}')

plt.scatter(x, y, label='Исходные данные')

plt.xlabel('x')

plt.ylabel('y')

plt.legend()

plt.show()

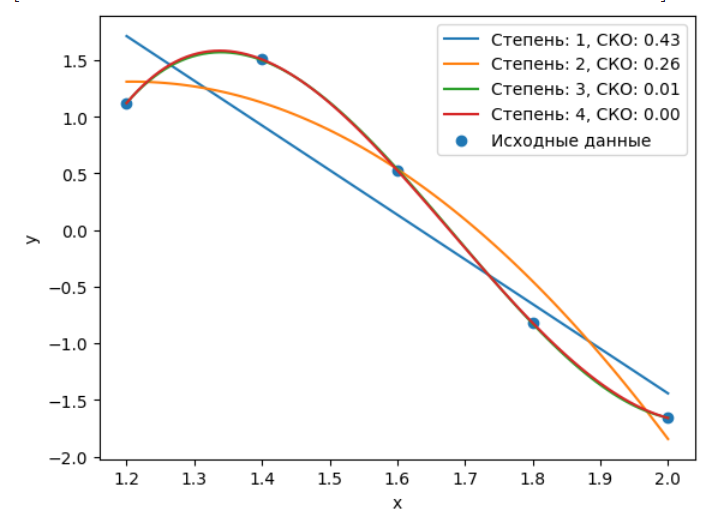


Рисунок 1 – полученные полиномы и их СКО

Выпишу также полученные многочлены и их СКО