

ФАКУЛЬТЕТ «Информатика и системы управления»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Лабораторная работа № 4**

|  |  |
| --- | --- |
| **Тема:** Построение и программная реализация алгоритма наилучшего среднеквадратичного приближения.  **Студент:** Елгин И. Ю.  **Группа:** ИУ7-44Б  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель:** Градов В. М. |  |

Москва.

2021 г.

**Цель работы**: Получение навыков построения алгоритма метода наименьших квадратов с использованием полинома заданной степени при аппроксимации табличных функций с весами.

**1 Исходные данные.**

1. Таблица функции с весами ρ i с количеством узлов N.

1. Степень аппроксимирующего полинома - n

Программа считывает таблицу из файла “*input.txt*”, находящегося в директории программы.

Значение n вводятся с клавиатуры

**2 Код на Python.**

Main.py

**import** matplotlib.pyplot **as** plt

**import** numpy **as** np

#Вычислить значение функции

**def** **f**(x, coeff):

res = np.zeros(len(x))

**for** i **in** range(len(coeff)):

res += coeff[i] \* (x\*\*i)

**return** res

# Ввести данные из файла

**def** **read\_table**(fname):

f = open(fname, "r")

x, y, p = [], [], []

**for** strs **in** f:

strs = strs.split(" ")

x.append(float(strs[0]))

y.append(float(strs[1]))

p.append(float(strs[2]))

**return** x, y, p

### Вычислить значение соэффициентов

**def** **calculate\_coef**(x, y, p, n):

sum\_x\_n = [sum([x[i]\*\*j\*p[i] **for** i **in** range(len(x))]) **for** j **in** range(n\*2 -1)]

sum\_y\_x\_n = [sum([x[i]\*\*j\*p[i]\*y[i] **for** i **in** range(len(x))]) **for** j **in** range(n)]

matr = [sum\_x\_n[i:i+n] **for** i **in** range(n)]

**for** i **in** range(n):

matr[i].append(sum\_y\_x\_n[i])

print(matr)

**return** Gauss(matr)

#Вычисление СЛАУ

**def** **Gauss**(matr):

n = len(matr)

**for** k **in** range(n):

**for** i **in** range(k+1,n):

coeff = -(matr[i][k]/matr[k][k])

**for** j **in** range(k,n+1):

matr[i][j] += coeff\*matr[k][j]

a = [0 **for** i **in** range(n)]

**for** i **in** range(n-1, -1, -1):

**for** j **in** range(n-1, i, -1):

matr[i][n] -= a[j]\*matr[i][j]

a[i] = matr[i][n]/matr[i][i]

**return** a

# Построить граффик

**def** **makeGraph**(coef, x, y, p):

x\_arr = np.arange(-1.0, 5.0, 0.02)

plt.figure(1)

plt.ylabel("Y")

plt.xlabel("X")

plt.plot(x\_arr, f(x\_arr, coef), 'k')

#вывод точек

**for** i **in** range(len(x)):

plt.plot(x[i], y[i], 'ro', markersize=p[i]+2)

plt.show()

x, y, p = read\_table("d:/input.txt")

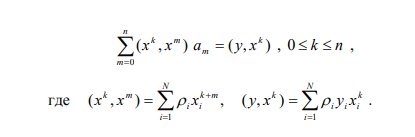
n = int(input("Input n: "))

coef = calculate\_coef(x, y, p, n+1)

print("\na:", coef)

makeGraph(coef, x, y, p)

Для выбранной степени полинома программа строит матрицу СЛАУ



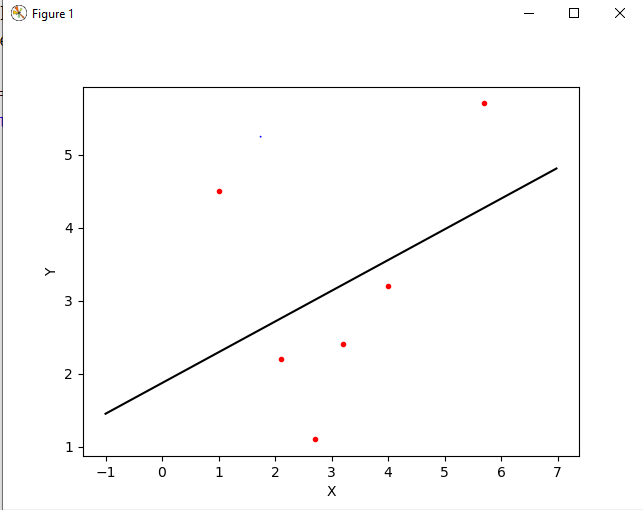
и по методу Гауса вычисляет значения коэффициентов для уравнения после чего строит по нему граффик.

**3 Результаты работы.**

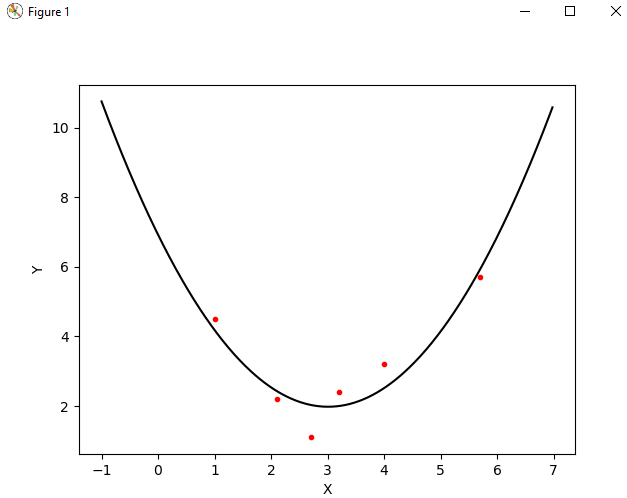
**1.**

|  |  |  |  |
| --- | --- | --- | --- |
| № | X | Y | P |
| 1 | 1 | 4.5 | 1 |
| 2 | 2.1 | 2.2 | 1 |
| 3 | 2.7 | 1.1 | 1 |
| 4 | 3.2 | 2.4 | 1 |
| 5 | 4 | 3.2 | 1 |
| 6 | 5.7 | 5.7 | 1 |

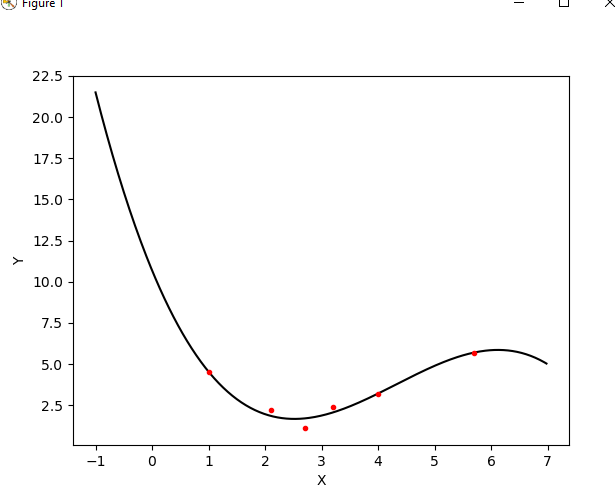
**N=1**

****

**N=2**

****

**N=3**

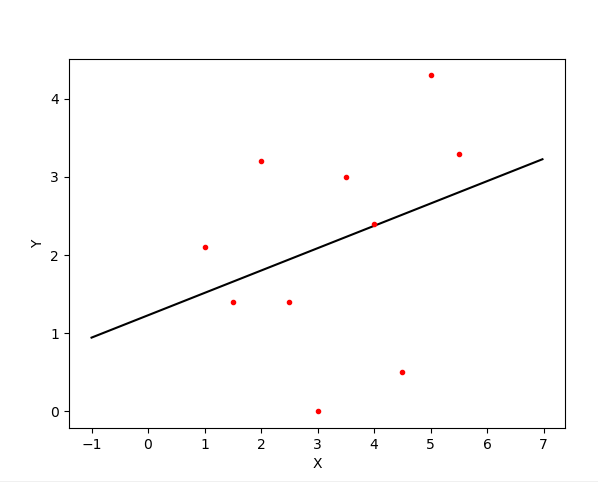
****

**2.**

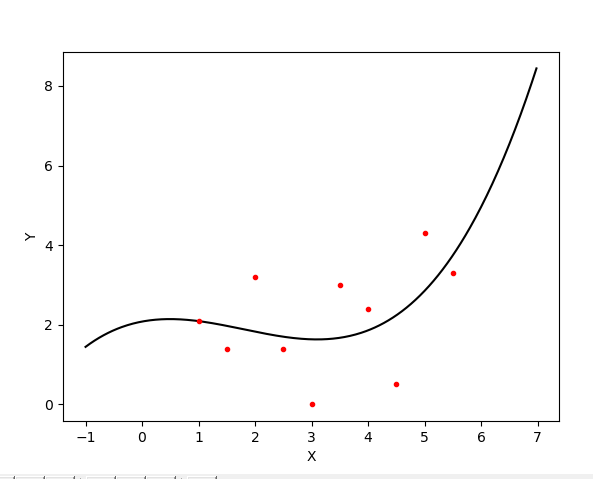
**Таблица данных**

|  |  |  |
| --- | --- | --- |
| **x** | **y** | **p** |
| **1** | **2.1** | **1** |
| **1.5** | **1.4** | **1** |
| **2** | **2.1** | **1** |
| **2.5** | **1.4** | **1** |
| **3** | **0** | **1** |
| **3.5** | **3** | **1** |
| **4** | **2.4** | **1** |
| **4.5** | **0.5** | **1** |
| **5** | **4.3** | **1** |
| **5.5** | **3.3** | **1** |

**N=1**

****

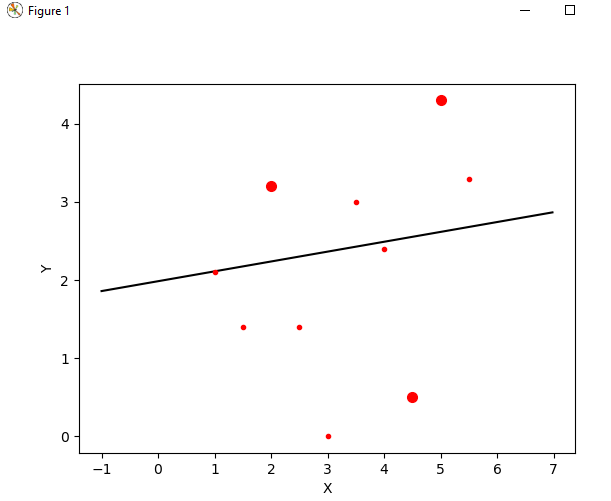
**N = 3**

****

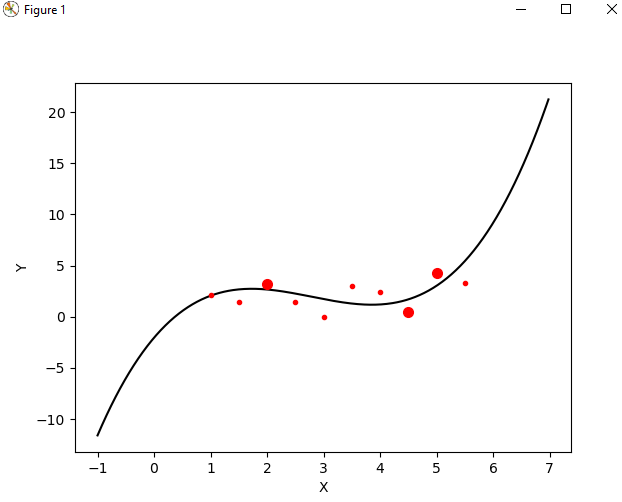
**Таблица с изменёнными весами**

|  |  |  |
| --- | --- | --- |
| **x** | **y** | **p** |
| **1** | **2.1** | **1** |
| **1.5** | **1.4** | **1** |
| **2** | **2.1** | **5** |
| **2.5** | **1.4** | **1** |
| **3** | **0** | **1** |
| **3.5** | **3** | **1** |
| **4** | **2.4** | **1** |
| **4.5** | **0.5** | **5** |
| **5** | **4.3** | **5** |
| **5.5** | **3.3** | **1** |

**N=1**

****

**N=3**

****

**4 Вопросы при защите лабораторной работы.**

1. Что произойдет при задании степени полинома n=N-1

*Кривая пройдет по всем точкам независимо от весов*

2. Будет ли работать Ваша программа при n ≥ N ? Что именно в алгоритме требует отдельного анализа данного случая и может привести к аварийной остановке?

*Невозможно**построить кривую n-й или большей степени по n точкам. Определитель СЛАУ будет равен нулю. Программа работает и считает.*

1. Получить формулу для коэффициента полинома a0 при степени полинома n=0. Какой смысл имеет величина, которую представляет данный коэффициент?

График горизонтальная прямая проходящая между точками , смысл математическое ожидание (среднее)

1. Записать и вычислить определитель матрицы СЛАУ для нахождения коэффициентов полинома для случая, когда n=N=2. Принять все ρ i =1.

СЛАУ:

Матрица:

|  |  |  |
| --- | --- | --- |
| 2 |  |  |
|  |  |  |
|  |  |  |

Определитель = =0

1. Построить СЛАУ при выборочном задании степеней аргумента полинома m n x a a x a x 0 1 2 ϕ( ) = + + , причем степени n и m в этой формуле известны.

СЛАУ:

1. Предложить схему алгоритма решения задачи из вопроса 5, если степени n и m подлежат определению наравне с коэффициентами ak , т.е. количество неизвестных равно 5

*Необходимо перебрать n от 1 до 5 и m от 1 до n и вычислить для них коэффициенты и выбрать ту пару (n,m) которая даёт минимальное значение*