Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

ОТЧЕТ

по лабораторной работе №4
«Интерполяция и аппроксимация»

по дисциплине «**Вычислительная математика**»

Автор: Пряничников Кирилл Сергеевич

Факультет: ПИиКТ

Группа: P32202

Преподаватель: Перл Ольга Вячеславовна

Санкт-Петербург, 2023

Описание метода интерполяции кубическими сплайнами

При решении задачи интерполяции кубическими сплайнами для $n+1$ точек с координатами $(x_i, y_i), 0 \leq i \leq n$ используется n сплайнов S_i, для которых выполняются следующие условия:

$$\begin{cases} S_i(x_i) = y_i \\ S_i(x_{i+1}) = S_{i+1}(x_{i+1}) \\ S_i'(x_{i+1}) = S_{i+1}'(x_{i+1}) \\ S_i''(x_{i+1}) = S_{i+1}''(x_{i+1}) \end{cases}$$

Каждый сплайн представляет собой кубический многочлен вида

$$a(x-x_i)^3 + b(x-x_i)^2 + c(x-x_i) + d$$

где a, b, c и d – коэффициенты многочлена

Кроме того, для поиска коэффициентов сплайнов используются краевые условия, в которых на краях интерполируемого интервала задаётся значение второй производной сплайна.

Поиск коэффициентов сплайнов

Начнём с коэффициентов d_i

По условию $S_i(x_i) = y_i$, отсюда $a_i(x_i - x_i)^3 + b_i(x_i - x_i)^2 + c_i(x_i - x_i) + d_i = y_i \Rightarrow d_i = y_i$

Для поиска остальных коэффициентов положим, что $M_i = 2b_i, h_i = x_{i+1} - x_i$

Тогда $f_i''(x_{i+1}) = 6a_i h + M_i,\ f_{i+1}''(x_{i+1}) = M_{i+1}$

Отсюда $a_i = {}^{M_{i+1} - M_i}/_{6h_i}$

$$f_{i+1}(x_{i+i}) = d_{i+1} \Leftrightarrow d_{i+1} = a_i h^3 + b_i h^2 + c_i h + d_i \Leftrightarrow c_i = {}^{d_{i+1} - d_i}/_{h} - (a_i h^2 + b_i h)$$
$$= {}^{y_{i+1} - y_i}/_{h} - \left({}^{M_{i+1} + 2M_i}/_{6}\right)$$

Для того, чтобы найти M, нужно решить следующую систему уравнений:

$$\begin{bmatrix} 4 & 1 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & 1 & 4 & 1 & \dots & 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & & 1 & 4 \end{bmatrix} \begin{bmatrix} M_2 \\ M_3 \\ M_4 \\ \dots \\ \dots \\ M_{n-3} \\ M_{n-2} \\ M_{n-1} \end{bmatrix} = \frac{6}{h^2} \begin{bmatrix} y_0 - 2y_1 + y_2 \\ y_1 - 2y_2 + y_3 \\ y_2 - 2y_3 + y_4 \\ .. \\ \dots \\ y_{n-4} - 2y_{n-3} + y_{n-3} \\ y_{n-3} - 2y_{n-2} + y_{n-1} \\ y_{n-2} - 2y_{n-1} + y_n \end{bmatrix}$$

Блок-схема метода

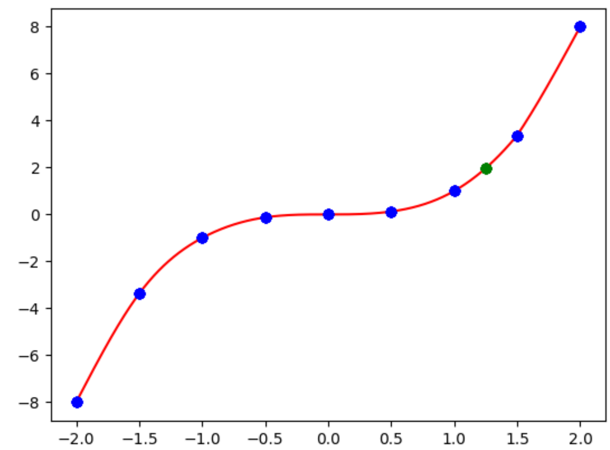
Функция, реализующая метод интерполяции кубическими сплайнами на Python

```
def interpolate_by_spline(X, Y, x):
    const_size = len(X) - 1
    A, B, C, D = [], [], [], []
    M = get_m_coefficients(X, Y)

    for i in range(const_size):
        h = X[i + 1] - X[i]
        A.append((M[i + 1] - M[i]) / (6 * h))
        B.append(M[i] / 2)
        C.append((Y[i + 1] - Y[i]) / h - (M[i + 1] + 2 * M[i]) * h / 6)
        D.append(Y[i])

    for i in range(const_size):
        if X[i] <= x <= X[i + 1]:
            x -= X[i]
            return A[i] * pow(x, 3) + B[i] * pow(x, 2) + C[i] * x + D[i]
```

Примеры работы программы

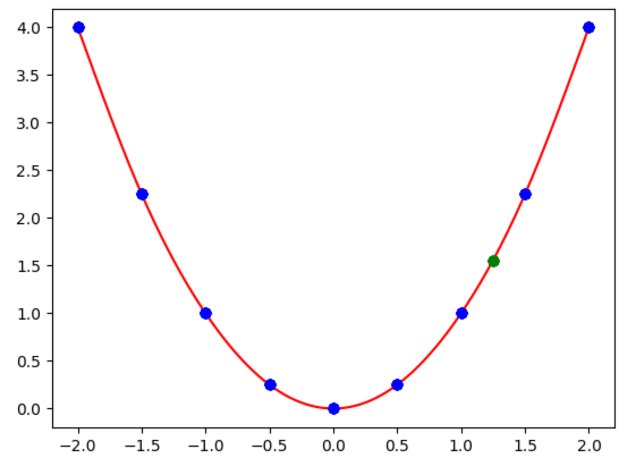
Пример 1:

Функция x^3
Входные данные для x и y:
[-2 -1.5 -1 -0.5 0 0.5 1 1.5 2]
[-8 -3.375 -1 -0.125 0 0.125 1 3.375 8]
Координата x неизвестной точки: 1.25
Результат: 1.9714625138070692
График функции, полученный в результате интерполяции:



Пример 2:

Функция x^2
Входные данные для x и y:
[-2 -1.5 -1 -0.5 0 0.5 1 1.5 2]
[-8 -3.375 -1 -0.125 0 0.125 1 3.375 8]
Координата x неизвестной точки: 1.25
Результат: 1.5563788659793816
График функции, полученный в результате интерполяции:



Пример 3:

Функция $\sin(x)$

Входные данные для x и y:

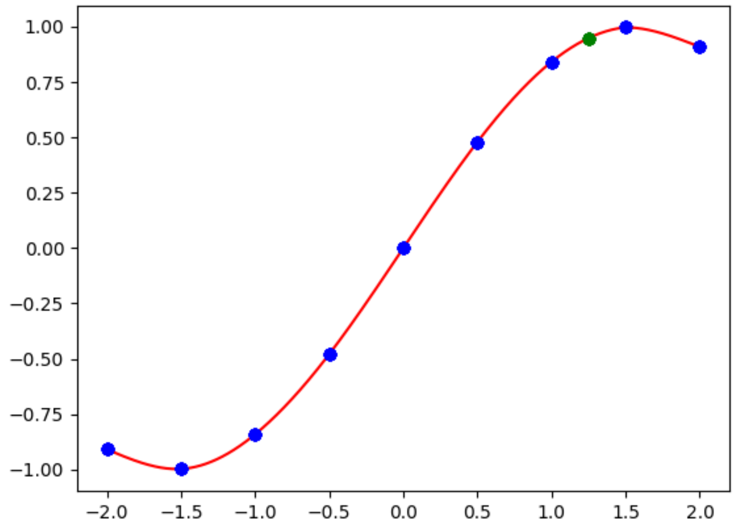
[-2 -1.5 -1 -0.5 0 0.5 1 1.5 2]

[-0.9092974268256817 -0.9974949866040544 -0.8414709848078965 -0.479425538604203 0 0.479425538604203 0.8414709848078965 0.9974949866040544 0.9092974268256817]

Координата x неизвестной точки: 1.25

Результат: 0.9488598406231616

График функции, полученный в результате интерполяции:



Вывод

Метод интерполяции кубическими сплайнами является достаточно эффективным, работает лучше с большими объёмами данных и лучше обрабатывает шум по сравнению с другими методами интерполяции. Однако этим методом сложно аппроксимировать простые функции. Поэтому метод сплайнов лучше использовать для обработки больших данных с вероятностью шума.