Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатики и систем управления

КАФЕДРА Теоретической информатики и компьютерных технологий

Отчет по лабораторной работе № 3
«Приближение функции кубическими сплайнами»
по курсу «Численные методы»

Выполнил:

студент группы ИУ9-61Б

Баев Д. А.

Проверила:

Домрачева А. Б.

Москва, 2023

1. Цель

Целью данной работы является изучение приближения заданной функции путем сплайн-интерполяции, построение сплайна третьего порядка на основе заданных точек и вычисление значения сплайна третьего порядка в серединах частичных отрезков между узлами интерполяции.

2. Постановка задачи

Дано: функция $y = f(x)$ задана конечным набором точек $y_i = f(x_i)$, $i = \overline{0, n}$ на отрезке $[a, b]$, $a = x_0$, $b = x_n$, $x_i = a + ih$, $h = \frac{(b-a)}{n}$

x_i	x_0	x_1	...	x_{n-1}	x_n
y_i	y_0	y_1	...	y_{n-1}	y_n

Найти: интерполяционную функцию $y = g(x)$: $g(x_i) = f(x_i)$, $i = \overline{0, n}$

1. Для заданных узлов (x_i, y_i) построить кубический сплайн.

2. Вычислить значения $g(x)$ в серединах частичных отрезков между узлами интерполяции (в точках $x_i^* = a + \left(i - \frac{1}{2}\right)h$, $h = \frac{(b-a)}{n}$).

Индивидуальный вариант: $y = f(x)$ задана конечным набором точек:

x_i	1	1.5	2	2.5	3	3.5	4	4.5	5
y_i	3.33	2.30	1.60	1.27	1.18	0.99	1.41	0.80	1.12

3. Основные теоретические сведения

Интерполяционной функцией называется функция $y = g(x)$, проходящая через заданные точки, называемые узлами интерполяции: $g(x_i) = f(x_i)$, $i = \overline{0, n}$. При этом в промежуточных точках равенство выполняется с некоторой погрешностью $g(x_i^*) \approx f(x_i^*)$.

Сплайн k-го порядка — функция, проходящая через все узлы (x_i, y_i), $i = \overline{0, n}$, которая является многочленом k-ой степени на каждом частичном отрезке разбиения $[x_i, x_{i+1}]$ и имеющая первые p непрерывных на $[a, b]$ производных.

$d = k - p$ – дефект сплайна. Наиболее употребительны сплайны третьего порядка с дефектом $d = 1$ (кубические сплайны).

Для каждого отрезка разбиения отыскиваем кубический сплайн в виде:

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$$

$$x\in[x_i, x_{i+1}],\ \ i = \overline{0, n - 1}$$

На частные многочлены накладываются условия:

1. Сплайн проходит через все узлы

$$S_i(x_i) = y_i\ ,\ \ i = \overline{0, n - 1};\ \ \ S_{n-1}(x_n) = y_n$$

2. Условие гладкости на краях

$$S_0''(x_0) = 0;\ S_{n-1}''(x_i) = 0$$

3. Непрерывность сплайна и его первых двух производных в промежуточных узлах

$$S_{i-1}'(x_i) = S_i'(x_i);$$

$$S_{i-1}''(x_i) = S_i''(x_i);$$

$$i = \overline{0, n - 1}$$

Эти условия позволяют выразить коэффициенты a_i, b_i, d_i и приводят к трехдиагональной СЛАУ относительно коэффициента c_i:

$$a_i = y_i,\ \ i = \overline{0, n - 1};$$

$$b_i = \frac{y_{i+1} - y_i}{h} - \frac{h}{3}(c_{i+1} + 2c_i),\ \ \ \ i = \overline{0, n - 2};$$

$$b_{n-1} = \frac{y_n - y_{n-1}}{h} - \frac{2h}{3}c_{n-1};$$

$$d_i = \frac{c_{i+1} - c_i}{3h},\ \ i = \overline{0, n - 2};$$

$$d_{n-1} = -\frac{c_{n-1}}{3h}$$

СЛАУ с трехдиагональной матрицей относительно коэффициента c_i:

$$c_{i-1} + 4c_i + c_{i+1} = \frac{3(y_{i+1} - 2y_i + y_{i-1})}{h^2}, \quad i = \overline{1, n-1};$$

$$c_0 = c_n = 0,$$

где $h = x_{i+1} - x_i$, $i = \overline{0, n-1}$.

4. Реализация

Листинг 1. Сплайн-интерполяция

```
import pandas as pd


#def f(x):
#    return (x ** 4) + 2 * (x ** 3) + 3 * (x ** 2) + 4 * x

#def f(x):
#     return math.exp(x)

def run(n, h, y):
    top = [1] * n
    top[n-1] = 0
    mid = [4] * n
    bot = [1] * n
    bot[0] = 0
    answer = [0] * n
    for i in range(n):
        answer[i] = 3 * (y[i + 2] - 2 * y[i + 1] + y[i]) / (h ** 2)

    v = [0] * n
    u = [0] * n
    x = [0] * n

    v[0] = - top[0] / mid[0]
    u[0] = answer[0] / mid[0]

    for i in range(1, n):
        v[i] = - top[i] / (bot[i] * v[i - 1] + mid[i])
        u[i] = (answer[i] - bot[i] * u[i - 1]) / (bot[i] * v[i - 1] + mid[i])

    x[n - 1] = u[n - 1]
    for i in range(n - 1, 0, -1):
        x[i - 1] = v[i - 1] * x[i] + u[i - 1]

    return [0] + x + [0]


def calculate_coefficients(c, y, n, h):
```

```python
    a = [0] * n
    b = [0] * n
    d = [0] * n
    for i in range(n):
        a[i] = y[i]
        b[i] = (y[i + 1] - y[i]) / h - h * (c[i + 1] + 2 * c[i]) / 3
        d[i] = (c[i + 1] - c[i]) / (3 * h)
    return a, b, d


if __name__ == '__main__':
    x_0 = 1
    x_n = 5
    n = 8
    h = (x_n - x_0) / n

    #x = [0] * (n + 1)
    #y = [0] * (n + 1)

    #for i in range(n + 1):
    #     x[i] = x_0 + i * h
    #     y[i] = f(x[i])
    x = [1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5]
    y = [3.33, 2.30, 1.60, 1.27, 1.18, 0.99, 1.41, 0.80, 1.12]
    c = run(n - 1, h, y)
    a, b, d = calculate_coefficients(c, y, n, h)
    print(a)
    print(b)
    print(c)
    print(d)

    func_arr = []
    spline_arr = []
    delta = []
    x_arr = [1, 1.25, 1.5, 1.75, 2, 2.25, 2.5, 2.75, 3, 3.25, 3.5, 3.75, 4, 4.25,
4.5, 4.75, 5]
    for i in range(2 * n + 1):
        x_i = x_0 + 0.5 * i * h
        j = i // 2
        if j == n:
            j -= 1
         spline_i = a[j] + b[j] * (x_i - x[j]) + c[j] * (x_i - x[j]) ** 2 + d[j] *
(x_i - x[j]) ** 3
    #     y_i = f(x_i)

    #     func_arr.append(y_i)
        spline_arr.append(spline_i)
```

```
#     delta.append(spline_i - y_i)
#     x_arr.append(x_i)

answer = pd.DataFrame({'x': x_arr, 'interpolation': spline_arr})
print(answer)
```

5. Результаты

Для заданных узлов интерполяции был построен кубический сплайн с коэффициентами:

a: [3.33, 2.3, 1.6, 1.27, 1.18, 0.99, 1.41, 0.8]

b: [-2.196596097201768, -1.7868078055964653, -1.0361726804123708, -0.24850147275405038, -0.48982142857142846, 0.5277871870397643, -0.24132731958762899, -0.7024779086892488]

c: [0, 0.8195765832106058, 0.6816936671575832, 0.8936487481590576, -1.376288659793814, 3.4115058910161995, -4.9497349042709855, 4.02743372606774712.86131, 1.23472, 0.43977, -4.31383, 8.05557, -5.46845, 1.45825, -0.36456]

d: [0.5463843888070705, -0.09192194403534841, 0.14130338733431627, -1.5132916053019143, 3.1918630338733425, -5.574160530191456, 5.9847790868924875, -2.6849558173784978]

Значения функции в узлах интерполяции и в серединах частичных отрезков между узлами интерполяции представлены в таблице 1.

Таблица 1 – Результаты работы программы

x_i	y_i
1	3.33
1.25	2.789388
1.5	2.30
1.75	1.903085
2	1.60
2.25	1.385771
2.5	1.27

2.75	1.240082
3	1.18
3.25	1.021399
3.5	0.99
3.75	1.248070
4	1.41
4.25	1.133822
4.5	0.80
4.75	0.834143
5	1.12

6. Вывод

В процессе выполнения лабораторной работы был исследован и успешно реализован метод интерполяции кубическими сплайнами. Был построен сплайн третьего порядка для заданной функции, и также были вычислены значения в точках интерполяции, которые не совпадают с узлами.

При анализе полученных результатов можно заметить, что значения кубического сплайна в узлах интерполяции совпадают с значениями исходной функции, что соответствует цели задачи. Это достигается благодаря методу вычисления коэффициентов a.