|  |  |
| --- | --- |
|  | **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

**Лабораторная работа № 3**

«Приближение функции кубическими сплайнами»

по курсу «Численные методы»

Выполнила:

студент группы ИУ9-61Б

Яровикова Анастасия

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 (узлов интерполяции) и вычисление значения сплайна третьего порядка в серединах частичных отрезков между узлами интерполяции.

1. **Постановка задачи**

**Дано:** функция задана конечным набором точек

на отрезке

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Найти:** интерполяционную функцию (т.е. функцию, совпадающую со значениями в узлах интерполяции ):

1. Для заданных узлов построить кубический сплайн (распечатать массивы a, b, c, d).

2. Вычислить значения в серединах частичных отрезков между узлами интерполяции, т.е. в точках .

**Индивидуальный вариант:** задана конечным набором точек:

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

1. **Основные теоретические сведения**

Интерполяционной функцией называется функция , проходящая через заданные точки, называемые узлами интерполяции:

. При этом в промежуточных точках равенство выполняется с некоторой погрешностью . Задача интерполяции заключается в поиске такой функции

Приближение функции кубическим сплайном — пример задачи интерполяции.

Сплайн k-го порядка — функция, проходящая через все узлы , являющаяся многочленом -ой степени на каждом частичном отрезке разбиения и имеющая первые непрерывных на производных.

– дефект сплайна. Чем выше дефект, тем грубее сплайн.

Наиболее употребительны сплайны третьего порядка с дефектом (кубические сплайны).

На каждом частичном отрезке разбиения кубический сплайн описывается

На частные многочлены накладываются условия:

* 1. Сплайн проходит через все узлы
  2. Условие гладкости на краях

; 0

* 1. Непрерывность сплайна и его первых двух производных в промежуточных узлах

Эти условия позволяют выразить коэффициенты и приводят к трехдиагональной СЛАУ относительно коэффициента

СЛАУ с трехдиагональной матрицей относительно коэффициента :

,

где

1. **Реализация**

Листинг 1. Сплайн-интерполяция

package main

import (

"fmt"

"math"

)

const SIZE = 8

func f(x float64) float64 {

return math.Exp(x)

}

func direct(b, a, c, d []float64, size int) (alpha, beta []float64) {

alpha = append(alpha, -c[0] / b[0])

beta = append(beta, d[0] / b[0])

var y float64

for i := 1; i < size - 1; i++ {

y = a[i - 1] \* alpha[i - 1] + b[i]

alpha = append(alpha, -c[i] / y)

beta = append(beta, (d[i] - a[i - 1] \* beta[i - 1]) / y)

}

y = a[size - 2] \* alpha[size - 2] + b[size - 1]

beta = append(beta, (d[size - 1] - a[size - 2] \* beta[size - 2]) / y)

return alpha, beta

}

func reverse(alpha, beta []float64, size int) (x []float64) {

x = make([]float64, size)

x[size - 1] = beta[size - 1]

for i := size - 2; i >= 0; i-- {

x[i] = alpha[i] \* x[i + 1] + beta[i]

}

return x

}

func main() {

var l, r, h float64

l , r = 1.0, 5.0

h = (r - l) / SIZE

xs := []float64{1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0}

ys := []float64{3.33, 2.30, 1.60, 1.27, 1.18, 0.99, 1.41, 0.80, 1.12}

fmt.Println("table for f(x):")

for i := 0; i <= SIZE; i++ {

fmt.Printf("%.1f; %.16f\n", xs[i], ys[i])

}

d := []float64{}

for i := 1; i < SIZE; i++ {

d = append(d, 3 \* (ys[i + 1] - 2 \* ys[i] + ys[i - 1]) / (h \* h) )

}

b := []float64{}

for i := 1; i < SIZE; i++ {

b = append(b, 4)

}

a := []float64{}

for i := 1; i < SIZE - 1; i++ {

a = append(a, 1)

}

c := []float64{}

for i := 1; i < SIZE - 1; i++ {

c = append(c, 1)

}

alpha, beta := direct(b, a, c, d, SIZE - 1)

coefC := reverse(alpha, beta, SIZE - 1)

coefC = append([]float64{0}, coefC...)

coefC = append(coefC, 0)

coefA := make([]float64, 0, SIZE)

for i := 1; i <= SIZE; i++ {

coefA = append(coefA, ys[i - 1])

}

coefB := make([]float64, 0, SIZE + 1)

for i := 1; i <= SIZE; i++ {

coefB = append(coefB, (ys[i] - ys[i - 1]) / h - (h / 3) \* (coefC[i] + 2 \* coefC[i - 1]))

}

coefD := make([]float64, 0, SIZE)

for i := 1; i <= SIZE; i++ {

coefD = append(coefD, (coefC[i] - coefC[i - 1]) / ( 3 \* h))

}

for i := 0; i < SIZE; i++ {

varX := l + float64(i) \* h

varY := ys[i]

s := coefA[i] + coefB[i] \* (varX - xs[i]) + coefC[i] \* math.Pow(varX - xs[i], 2) + coefD[i] \* math.Pow(varX - xs[i], 3)

fmt.Printf("x: %.1f, y: %.16f, y\*: %.16f, |y-y\*|: %.16f\n", varX, varY, s, math.Abs(varY - s))

}

for i := 0; i < SIZE; i++ {

varX := l + (float64(i + 1) - 0.5) \* h

s := coefA[i] + coefB[i] \* (varX - xs[i]) + coefC[i] \* math.Pow(varX - xs[i], 2) + coefD[i] \* math.Pow(varX - xs[i], 3)

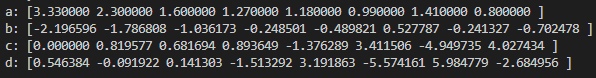
fmt.Printf("x: %.2f, y\*: %.16f\n", varX, s)

}

}

1. **Результаты**

Для заданных узлов интерполяции построен кубический сплайн с коэффициентами:



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

Таблица 1 – Результаты программы

|  |  |
| --- | --- |
|  |  |
| 1 | 3.3300000000000001 |
| 1.25 | 2.7893882317746685 |
| 1.5 | 2.2999999999999998 |
| 1.75 | 1.9030853046759939 |
| 2 | 1.6000000000000001 |
| 2.25 | 1.3857705495213550 |
| 2.5 | 1.2700000000000000 |
| 2.75 | 1.2400824972385862 |
| 3 | 1.1799999999999999 |
| 3.25 | 1.0213994615243005 |
| 3.5 | 0.9900000000000000 |
| 3.75 | 1.2480696566642120 |
| 4 | 1.4099999999999999 |
| 4.25 | 1.1338219118188511 |
| 4.5 | 0.8000000000000000 |
| 4.75 | 0.8341426960603830 |
| 5 | 1.1200000000000001 |

1. **Вывод**

В ходе выполнения лабораторной работы был изучен метод приближения функции с помощью интерполяции кубическим сплайном, был построен сплайн третьего порядка на основе заданных узлов интерполяции и найдены значения функции в серединах частичных отрезков между узлами интерполяции. В результате тестирования сделан вывод о точности данного метода приближения функции: значения функции и значения кубического сплайна совпадают в узлах интерполяции, а в серединах между узлами интерполяции присутствует погрешность результат вследствие увеличения погрешности при реализации арифметических операций на ЦВМ. Однако катастрофического роста погрешности не наблюдается.