|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** \_***ИУК «Информатика и управление»*\_\_**\_\_\_\_\_\_\_\_\_\_\_\_

**КАФЕДРА** \_\_***ИУК4 «Программное обеспечение ЭВМ, информационные технологии»***

**Лабораторная работа №2**

**«Сплайн-интерполяция табличных функций»**

**ДИСЦИПЛИНА: «Методы обработки информации»**

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИУК4-42Б | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_Петроченков И. А.\_\_)  (Подпись) (Ф.И.О.) |
| Проверил(а): | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_Никитенко У. В.\_\_\_)  (Подпись) (Ф.И.О.) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |
| Калуга, 2023 г. | | |

# Цели работы:

Изучение методов сплайн-интерполяции табличных функций.

**Задачи:**

1. Для заданной функции построить линейный и кубический сплайны;
2. Визуализировать результаты;
3. Оценить погрешность приближения;

**Вариант №8**

Для заданной функции y = f (x) , x∈[a, b] , построить линейный и кубический сплайн. Оценить погрешность полученных приближений на каждом элементарном отрезке [ xi −1, xi ], i =1,…, n, где a=x0≤ x1≤…≤ xn= b. Визуализировать результаты.

**Результаты выполнения работы:**

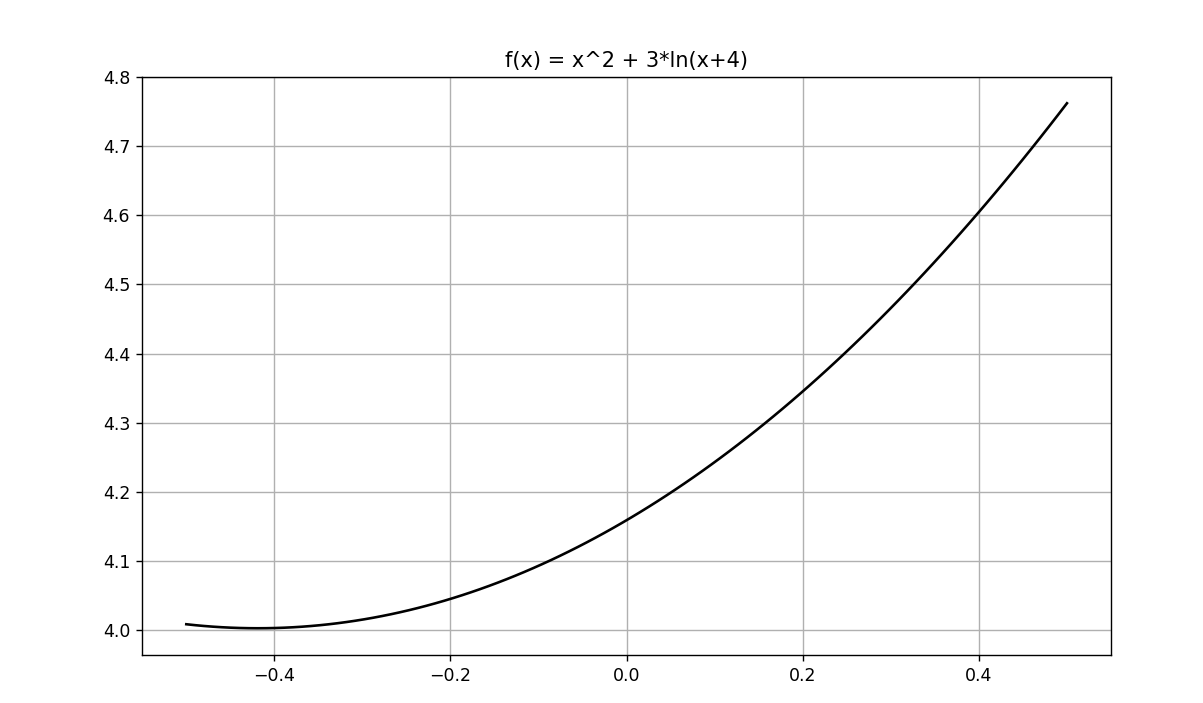
****

Рис. 1 Исходная функция

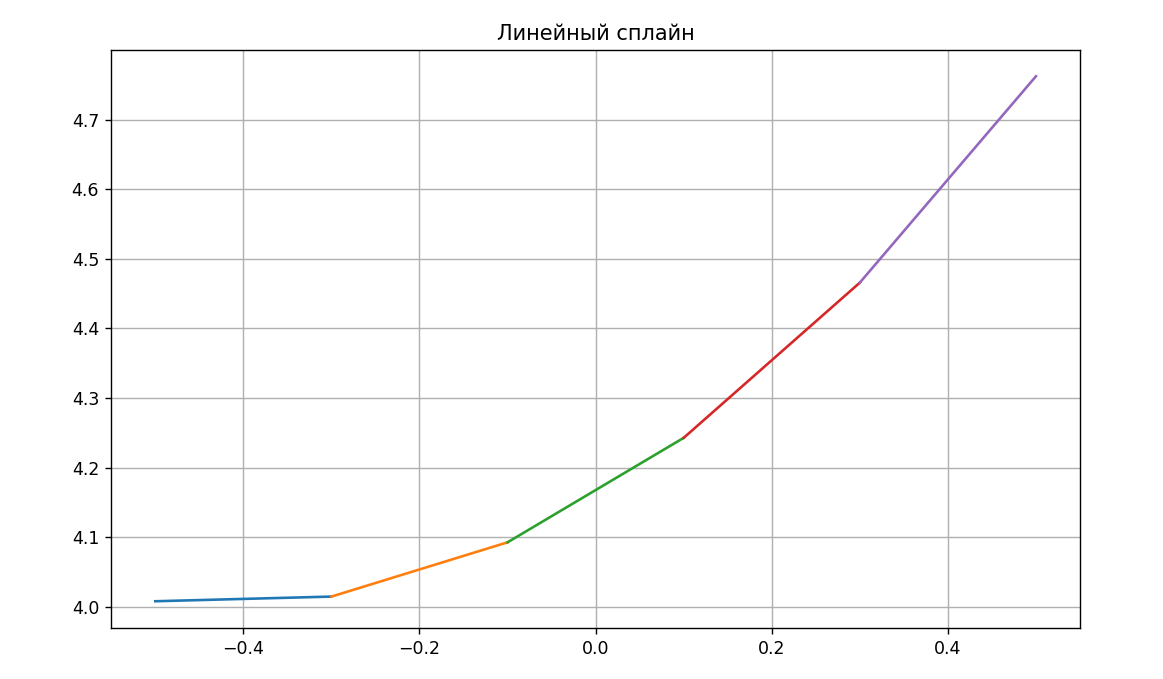
****

Рис. 2 Линейный сплайн

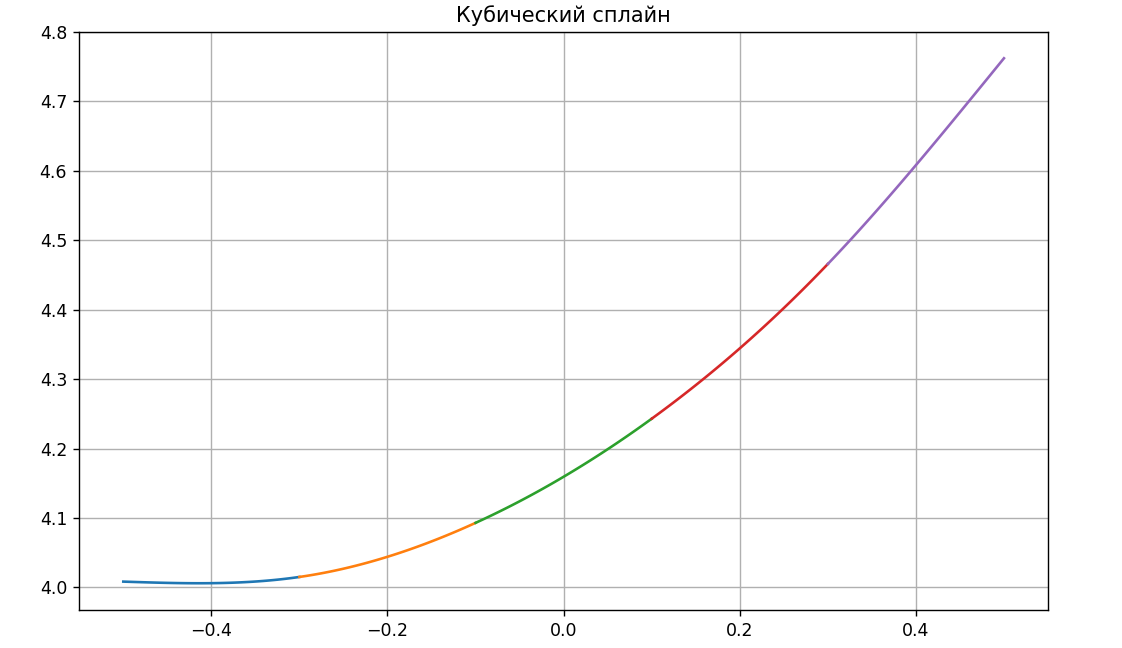
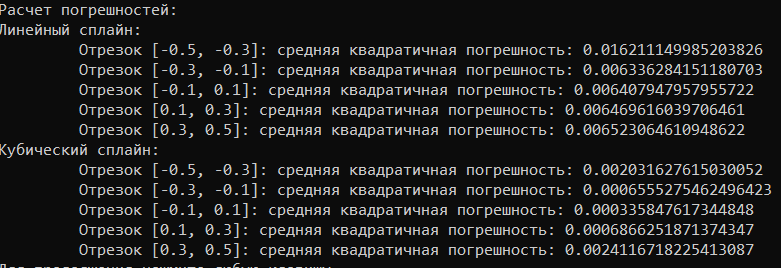
****

Рис. 3 Кубический сплайн

****Рис. 4 Оценка погрешностей построения

**Листинг кода**

import numpy as np

import matplotlib.pyplot as plt

from scipy.interpolate import interp1d

left = -0.3;

right = 0.3;

step = 5;

def f(x):

return x\*\*2 + 3\*np.log(x+4);

#return np.tan(np.sin(x));

#return np.sin(np.cos(x))#np.sqrt(np.tan(x)\*\*2 + 7\*x\*\*2)

def plot\_function(func, x\_range, label, \_type = '-'):

x = np.linspace(x\_range[0], x\_range[1], 100)

y = func(x)

plt.plot(x, y, \_type, label=label)

#plt.title(title)

plt.grid(True)

def linear\_func(a: float, b: float):

def f(x):

return a\*x + b;

return f;

def quadratic\_func(a: float, b: float, c: float):

def f(x):

return a\*x\*\*2 + b\*x + c;

return f;

def cubic\_func(a: float, b: float, c: float, d:float, xi: float):

def f(x):

return a\*(x-xi)\*\*3 + b\*(x-xi)\*\*2 + c\*(x-xi) + d;

return f;

#x = np.linspace(left, right, step)

#y = f(x)

#linear\_spline = interp1d(x, y)

#cubic\_spline = interp1d(x, y, kind='cubic')

# print(f"h = {(right - left)/(step-1)}")

# print("Расчет погрешностей: ")

# for i in range(1, len(x)):

# x\_mid = (x[i] + x[i-1]) / 2

# y\_true\_mid = f(x\_mid)

# y\_linear\_mid = linear\_spline(x\_mid)

# y\_cubic\_mid = cubic\_spline(x\_mid)

# error\_linear = abs(y\_linear\_mid - y\_true\_mid)

# error\_cubic = abs(y\_cubic\_mid - y\_true\_mid)

# print(f"Отрезок [{x[i-1]:.4f} {x[i]:.4f}]\n\tлинейный сплайн: {error\_linear:.4f}\n\tкубический сплайн: {error\_cubic:.4f}")

#x\_dense = np.linspace(-0.3, 0.3, 1000)

##y\_true = f(x\_dense)

#y\_linear = linear\_spline(x\_dense)

#y\_cubic = cubic\_spline(x\_dense)

\_x = [-0.5, -0.3, -0.1, 0.1, 0.3, 0.5];

\_y = [f(x) for x in \_x];

def plot\_func():

xs = np.linspace(\_x[0], \_x[-1], 100);

ys = f(xs);

plt.plot(xs, ys, color='black')

def plot\_linear\_spline():

#plt.plot(x\_dense, linear\_spline(x\_dense), color='blue')

print("Линейный сплайн: ")

spline = [];

for i in range(len(\_x) - 1):

a = f(\_x[i]);

b = (f(\_x[i+1]) - f(\_x[i]))/(\_x[i+1] - \_x[i]);

a -= \_x[i]\*b;

spline.append(linear\_func(b, a));

for i in range(len(spline)):

x = np.linspace(\_x[i], \_x[i+1])

\_y = spline[i](x);

plt.plot(x, \_y);

err = 0;

cur\_x = \_x[i];

ctr = 0;

while cur\_x < \_x[i+1]:

cur\_x += 0.1;

err += (spline[i](cur\_x) - f(cur\_x))\*\*2;

ctr += 1;

err /= ctr;

err \*\*= 1/2;

print(f'\t Отрезок [{\_x[i]}, {\_x[i+1]}]: средняя квадратичная погрешность: {err}')

plt.show();

def half\_diff(f, x1, x2):

return (f(x1)-f(x2))/(x1-x2);

def plot\_cubic\_spline():

print('Кубический сплайн: ')

matrix = [[0 for \_ in range(len(\_x))] for \_ in range(len(\_x))];

for i in range(len(\_x)):

if i == 0:

matrix[i][0] = 1;

elif i == len(\_x) - 1:

matrix[i][-1] = 1

else:

matrix[i][i-1] = \_x[i] - \_x[i-1];

matrix[i][i] = 2\*(\_x[i] - \_x[i-1] + \_x[i+1] - \_x[i]);

matrix[i][i+1] = \_x[i+1] - \_x[i];

fns = [0 for \_ in range(len(\_x))];

for i in range(len(\_x)):

if i == 0 or i == len(\_x) - 1: fns[i] = 0;

else: fns[i] = 3\*(half\_diff(f, \_x[i+1], \_x[i]) - half\_diff(f, \_x[i], \_x[i-1]))

#cs = [1 for \_ in range(len(\_x))];

cs = np.linalg.solve(matrix, fns);

cs[0] = 0;

cs[-1] = 0;

bs = [0 for \_ in range(len(\_x))];

for i in range(len(\_x)-1):

bs[i] = (f(\_x[i+1]) - f(\_x[i])) / (\_x[i+1] - \_x[i]);

bs[i] -= (\_x[i+1] - \_x[i]) \* (2\*cs[i] + cs[i+1])/3;

ds = [0 for \_ in range(len(\_x))];

for i in range(len(\_x) - 1):

ds[i] = (cs[i+1] - cs[i])/(3\*(\_x[i+1] - \_x[i]));

\_as = [f(x) for x in \_x];

spline = [0 for \_ in range(len(\_x) - 1)];

for i in range(len(\_x) - 1):

spline[i] = cubic\_func(ds[i], cs[i], bs[i], \_as[i], \_x[i] );

for i in range(len(spline)):

x = np.linspace(\_x[i], \_x[i+1])

\_y = spline[i](x);

plt.plot(x, \_y);

err = 0;

cur\_x = \_x[i];

ctr = 0;

while cur\_x < \_x[i+1]:

cur\_x += 0.1;

err += (spline[i](cur\_x) - f(cur\_x))\*\*2;

ctr += 1;

err /= ctr;

err \*\*= 1/2;

print(f'\t Отрезок [{\_x[i]}, {\_x[i+1]}]: средняя квадратичная погрешность: {err}')

plt.show();

#plt.plot(x\_dense, y\_cubic, color='green')

graphs = [(plot\_func, 'f(x) = '), (plot\_linear\_spline, 'Линейный сплайн'), (plot\_cubic\_spline, 'Кубический сплайн')];

if \_\_name\_\_ == "\_\_main\_\_":

print("Расчет погрешностей: ")

for graph, title in graphs:

plt.figure(figsize=(10, 6))

# plt.scatter(x, y, color='red', label='Точки интерполяции')

plt.grid(True)

plt.title(title);

graph();

plt.show();

**Вывод:**

В ходе выполнения лабораторной работы были изучены методы построения линейного и кубического сплайнов, измерена погрешность построения.