

ИНТЕРПОЛЯЦИЯ ТАБЛИЧНО ЗАДАНЫХ ФУНКЦИЙ

Пусть на отрезке $[x_{\min}, x_{\max}]$ задана система точек (узлы интерполяции) x_i и значения неизвестной функции в этих узлах f_i , $i = 1, \dots, N$. Необходимо построить функцию $F(x)$, принимающую в узлах интерполяции x_i , заданные значения f_i :

$$F(x_i) = f_i, \quad i = 1, \dots, N \quad (1)$$

(условия интерполяции).

Локальная интерполяция. На каждом интервале $[x_i, x_{i+1}]$, $1 \leq i \leq N-1$ строится своя функция $F_i(x)$.

При кусочно-линейной интерполяции

$$F_i(x) = a_i + b_i x.$$

Неизвестные коэффициенты a_i и b_i определяются из условий (1) на границах интервала:

$$F_i(x_i) = f_i, \quad F_i(x_{i+1}) = f_{i+1}.$$

При сплайн-интерполяции $F_i(x)$ представляется полиномом третьей степени (кубическим сплайном)

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3, \quad i = 1, 2, \dots, N-1. \quad (2)$$

Из условий интерполяции (1), непрерывности $S(x)$, ее первых (отсутствие изломов) и вторых (отсутствие скачков кривизны) производных в узлах можно получить $4N-6$ уравнений для $4N-4$ неизвестных:

$$\begin{aligned} a_i &= f_i, \quad i = 1, 2, \dots, N-1, \\ f_i + b_i h_i + c_i h_i^2 + d_i h_i^3 &= f_{i+1}, \quad i = 1, 2, \dots, N-1, \\ b_i + 2c_i h_i + 3d_i h_i^2 &= b_{i+1}, \quad i = 1, 2, \dots, N-2, \\ 2c_i + 6d_i h_i &= 2c_{i+1}, \quad i = 1, 2, \dots, N-2, \end{aligned} \quad (3)$$

где $h_i = x_{i+1} - x_i$.

Для замыкания системы (3) задаются значения вторых производных (кривизны) на концах отрезка интерполяции:

$$S_1^{(2)}(x_1) = s_1'', \quad S_N^{(2)}(x_N) = s_N''$$

или

$$c_1 = s_1''/2, \quad c_N = s_N''/2. \quad (4)$$

Из (3) получаются уравнения для определения коэффициентов сплайна:

$$d_i = \frac{c_{i+1} - c_i}{3h_i}, \quad i = 1, 2, \dots, N-1, \quad (5)$$

$$b_i = \frac{f_{i+1} - f_i}{h_i} - \frac{h_i}{3}(c_{i+1} + 2c_i), \quad i = 1, 2, \dots, N-1 \quad (6)$$

$$h_{i-1}c_{i-1} + 2(h_{i-1} + h_i)c_i + h_ic_{i+1} = 3 \left[\frac{f_{i+1} - f_i}{h_i} - \frac{f_i - f_{i-1}}{h_{i-1}} \right], \quad i = 2, 3, \dots, N-1. \quad (7)$$

Трехдиагональная система уравнений (4), (7) решается методом прогонки. После того как найдены все c_i , из (5), (6) определяются остальные неизвестные.

```
def SpSet(x,f,n,s1,sN,a,b,c,d):
    """
    Коэффициенты кубического сплайна
    """
    q = [0]*(n+1)
    a[1],c[1],b[1],d[1] = 0.,1.,0.,s1/2
    c[n],d[n] = 1.,sN/2
    for i in range(2,n):
        h1 = x[i]-x[i-1]
        h2 = x[i+1]-x[i]
        a[i] = h1
        c[i] = 2.*(h1+h2)
        b[i] = h2
        d[i] = 3.*((f[i+1]-f[i])/h2-(f[i]-f[i-1])/h1)
    Progon(a,c,b,d,q,n)
    for i in range(1,n):
        h2=x[i+1]-x[i]
        a[i]=f[i]
        c[i]=q[i]
        d[i]=(q[i+1]-q[i])/(3.*h2)
        b[i]=(f[i+1]-f[i])/h2-h2/3.*(q[i+1]+2.*q[i])
```

Здесь «N» – количество узлов таблицы, «X» и «F» – массивы табличных данных x_i и f_i , «s1» и «sN» – граничные значения s_1'' и s_N'' , «A», «B», «C» и «D» – массивы коэффициентов сплайна a_i , b_i , c_i и d_i . Используется процедура метода прогонки «PROGON».

Зная наборы коэффициентов (по четыре коэффициента для каждого участка), можно определить значение сплайн-функции по формуле (2) в любой точке.

```
def SpVal(x,a,b,c,d,n,z):
    """
    Значение кубического сплайна S(z) в точке z
    """
    for i in range(1,n+1):
        ! самостоятельно!!!
    return s
```

Глобальная интерполяция. Строится одна функция для всего интервала $[x_{\min}, x_{\max}]$. Примером является интерполяционный полином Лагранжа:

$$L(x) = \sum_{i=1}^N f_i \prod_{\substack{j=1 \\ i \neq j}}^N \frac{x - x_j}{x_i - x_j}. \quad (8)$$

Для повышения точности интерполяции вместо полного набора узлов часто используют несколько узлов, близких к данной точке x . В этом случае в формуле (8) индексы i и j должны изменяться от i_H (левой границы «окна») до i_K (правой границы «окна»), причем ширина «окна» $l = i_K - i_H$ обычно берется равной 4-5.

```
def Lagrok(x,f,n,lok,z):
    """
    Значение полинома Лагранжа P(z) (оконный вариант) в точке z,
    lok - ширина окна
    """
    i2 = lok//2 # ищем границы "окна" ina,iko
    if i2 == 0: i2 = 1
    i1 = i2
    for j in range(1,n+1):
        if z>x[i1] and i1<n: i1 += 1
    ina = i1-i2
    if ina<1: ina = 1
    iko = ina+lok
```

```

if iko>n:
    iko = n
    ina = iko-lok
p=0.0
for i in range(ina,iko+1):
    r = 1.0
    for j in range(ina,iko+1):

```

```

if i==j: continue
q = x[i]-x[j]
s = z-x[j]
r *= s/q
p += r*f[i]
return p

```

Здесь «lok» – ширина «окна».

Задание. Написать программу для расчета значения $F(z)$ таблично заданной функции $(x_i, f_i)_{i=1, \dots, N}$ ($x_1 = x_{\min}$, $x_N = x_{\max}$) в произвольной точке $z \in [x_{\min}, x_{\max}]$ с помощью:

- 1) кусочно-линейной интерполяции;
- 2) сплайн-интерполяции;
- 3) полинома Лагранжа в «оконном» варианте;
- 4) модуля `scipy.interpolate`.

Исследовать влияние на максимальную абсолютную погрешность $\Delta = \max|f - F|$ применяемых методов интерполяции:

- числа узлов N при $N = 11, \dots, 81$ (для $s_1'' = s_N'' = 0$, $l = 5$);
- граничных значений s_1'' и s_N'' 2-й производной сплайна при $s_1'' = s_N'' = -1 \dots 1$ (для $N = 11$);
- ширины «окна» l полинома Лагранжа при $l = 3 \dots 7$ (для $N = 11$).

Варианты заданий приведены в таблице:

Вариант	$f(x), [x_{\min}, x_{\max}]$	Вариант	$f(x), [x_{\min}, x_{\max}]$
1	$x^2 \cos(2x) + x^3/3, [-1, 1]$	2	$\sin^3(5x) + \exp(x/3), [0, 1]$
3	$x^2 - (1+x)/\operatorname{tg}^2(x/4), [1, 2]$	4	$\sqrt[3]{x+1} - 1/x, [2, 3]$
5	$6/x + 2^x \exp(-x^2), [1, 3]$	6	$x^3 \cos^2(x/3) + \lg(x/3), [1, 2]$
7	$\sin(3x^2) - 2x/(x+4), [-2, 0]$	8	$2x^2 - \sin^3(5x), [0, 1]$
9	$\exp(x) + x/(1+x^2), [-1, 1]$	10	$x^2 - \cos^2(\pi x), [1, 3]$
11	$0.1e^x - \sin^2 x + 0.5, [0, 1]$	12	$x \cos(x^2 - 2x) + x^2, [-1, 1]$
13	$\lg x - (x-1)/x, [1, 4]$	14	$x \sin(x/2) - x^4/4, [-2, 1]$
15	$\cos^2(3x) - \exp(x/5), [0, 1]$	16	$x^3 - (2+x)/\operatorname{ctg}(x/5), [2, 3]$
17	$2/x^2 - 3^{\sqrt{x}} \exp(-x), [2, 3]$	18	$x^2 \sin^3(x/5) + \log_2(x/2), [3, 4]$
19	$\sqrt{x} \sin(x^3 + 3x^2) - x, [1, 3]$	20	$\cos(2x^3) + 3x/(x-1), [2, 4]$
21	$\cos(2x^3) + x^3/(x^2+5), [-2, 1]$	22	$12 \sin(x) - 2^{\sqrt{3}x} \operatorname{sh}(-x), [2, 3]$
23	$\lg x - (x-1)/x, [1, 3]$		

Содержание отчета:

1. Постановка задачи.
2. Краткое описание используемых методов.

3. Текст расчетной программы.
4. Листинг результатов.
5. Графики зависимости погрешностей Δ от N , от s_1'' и от l . Определение практического порядка точности методов (показателя степени α в формуле $\Delta \sim N^{-\alpha}$).
6. Выводы.

Указания. Воспользоваться приведенными выше процедурами методов.

Пример главной программы:

```
from math import pi,sin
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import interp1d

def list_format(x,decimals=3):
    return (' '.join(f'{val:.{decimals}f}' for val in x))

def Progon(c,d,e,f,x,n):
    ...
def SpSet(x,f,n,s1,sN,a,b,c,d):
    ...
def SpVal(x,a,b,c,d,n,z):
    ...
def Lagrok(x,f,n,lok,z):
    ...
def fe(x): return 10.*sin(2*pi*x)

N,Nz = 11,31
xmin,xmax = -1.,1.
dx = (xmax-xmin)/(N-1)
x,f,a,b,c,d = ([0]*(N+1) for k in range(6))
z,s,p,ye = ([0]*(Nz+1) for k in range(4))
x[1:] = [xmin+(i-1)*dx for i in range(1,N+1)]
print('x=',list_format(x[1:]))
f[1:] = [fe(xi) for xi in x[1:]]
print('f=',list_format(f[1:]))
s1,sN = 0.0,0.0 # ГY S"(x1)=s1, S"(xN)=sN
SpSet(x,f,N,s1,sN,a,b,c,d)
print('a=',list_format(a[1:]))
print('b=',list_format(b[1:]))
print('c=',list_format(c[1:]))
print('d=',list_format(d[1:]))

dz = (x[N]-x[1])/(Nz-1)
dsmax,dpmax = 0.0,0.0
print((' '*3,'x',(' '*7),'f',(' '*9),'S',(' '*9),'P',(' '*7),'|f-S|',(' '*5),'|f-P|')
for i in range(1,Nz+1): # сплайн и полином
    Лагранжа в промежут. точках
    z[i] = x[1]+(i-1)*dz
    s[i] = SpVal(x,a,b,c,d,N,z[i])
    p[i] = Lagrok(x,f,N,5,z[i])
    ye[i] = fe(z[i])
    ds,dp = abs(s[i]-ye[i]),abs(p[i]-ye[i])
    if ds>dsmax: dsmax = ds
    if dp>dpmax: dpmax = dp
    print(f'{z[i]:7.3f} {ye[i]:11.3e} {s[i]:11.3e} {p[i]:11.3e} {ds:11.3e} {dp:11.3e}')
print(f' max|f-S|={dsmax:.3e}')
print(f' max|f-P|={dpmax:.3e}')

scipy_interp =
interp1d(np.array(x[1:]),np.array(f[1:]),kind='cubic')
scipy_results = scipy_interp(np.array(z[1:]))

plt.figure(dpi=300)
plt.plot(x[1:],f[1:], 'o',label='f(x)')
plt.plot(z[1:],s[1:],label='S(x)')
plt.plot(z[1:],p[1:],label='L(x)')
plt.plot(z[1:],scipy_results,label='Scipy')
plt.legend(loc=4)
plt.xlabel('x')
plt.ylabel('y')
plt.grid(True)
plt.show()
```

Программа «Interp» рассчитывает в промежуточных точках, не совпадающих с узлами, и выводит на экран значения табличной и интерполирующих функций и соответствующие абсолютные погрешности. Используются функции расчета точного значения табличной функции «fe», а также вышеприведенные «SpSet», «SpVal» и «Lagrok».

Результаты:

```
x= -1.000, -0.800, -0.600, -0.400, -0.200, 0.000, 0.200, 0.400, 0.600, 0.800, 1.000
f= 0.000, 9.511, 5.878, -5.878, -9.511, 0.000, 9.511, 5.878, -5.878, -9.511, -0.000
a= 0.000, 9.511, 5.878, -5.878, -9.511, 0.000, 9.511, 5.878, -5.878, -9.511, 0.000
b= 61.783, 19.092, -49.984, -49.984, 19.092, 61.783, 19.092, -49.984, -49.984, 19.092, 0.000
c= 0.000, -213.456, -131.923, 131.923, 213.456, -0.000, -213.456, -131.923, 131.923, 213.456, 1.000
d= -355.760, 135.888, 439.743, 135.888, -355.760, -355.760, 135.888, 439.743, 135.888, -355.760, 0.000
```

x	f	S	P	f-S	f-P
-1.000	2.449e-15	2.449e-15	2.449e-15	0.000e+00	0.000e+00
-0.933	4.067e+00	4.013e+00	3.801e+00	5.390e-02	2.664e-01
-0.867	7.431e+00	7.394e+00	7.294e+00	3.697e-02	1.373e-01
-0.800	9.511e+00	9.511e+00	9.511e+00	0.000e+00	0.000e+00
-0.733	9.945e+00	9.875e+00	9.989e+00	7.028e-02	4.425e-02
-0.667	8.660e+00	8.584e+00	8.686e+00	7.674e-02	2.554e-02
-0.600	5.878e+00	5.878e+00	5.878e+00	8.882e-16	0.000e+00
-0.533	2.079e+00	2.090e+00	1.905e+00	1.046e-02	1.744e-01
-0.467	-2.079e+00	-2.090e+00	-2.224e+00	1.046e-02	1.448e-01
-0.400	-5.878e+00	-5.878e+00	-5.878e+00	2.665e-15	3.553e-15
-0.333	-8.660e+00	-8.584e+00	-8.812e+00	7.674e-02	1.520e-01
-0.267	-9.945e+00	-9.875e+00	-1.006e+01	7.028e-02	1.150e-01
-0.200	-9.511e+00	-9.511e+00	-9.511e+00	0.000e+00	0.000e+00
-0.133	-7.431e+00	-7.394e+00	-7.351e+00	3.697e-02	8.043e-02
-0.067	-4.067e+00	-4.013e+00	-3.994e+00	5.390e-02	7.372e-02
0.000	0.000e+00	-1.332e-15	0.000e+00	1.332e-15	0.000e+00
0.067	4.067e+00	4.013e+00	4.269e+00	5.390e-02	2.017e-01
0.133	7.431e+00	7.394e+00	7.592e+00	3.697e-02	1.606e-01
0.200	9.511e+00	9.511e+00	9.511e+00	1.776e-15	1.776e-15
0.267	9.945e+00	9.875e+00	9.989e+00	7.028e-02	4.425e-02
0.333	8.660e+00	8.584e+00	8.686e+00	7.674e-02	2.554e-02
0.400	5.878e+00	5.878e+00	5.878e+00	8.882e-16	8.882e-16
0.467	2.079e+00	2.090e+00	2.075e+00	1.046e-02	4.150e-03
0.533	-2.079e+00	-2.090e+00	-2.075e+00	1.046e-02	4.150e-03
0.600	-5.878e+00	-5.878e+00	-5.878e+00	8.882e-16	0.000e+00
0.667	-8.660e+00	-8.584e+00	-8.686e+00	7.674e-02	2.554e-02
0.733	-9.945e+00	-9.875e+00	-9.989e+00	7.028e-02	4.425e-02
0.800	-9.511e+00	-9.511e+00	-9.511e+00	1.776e-15	0.000e+00
0.867	-7.431e+00	-7.394e+00	-7.294e+00	3.697e-02	1.373e-01
0.933	-4.067e+00	-4.013e+00	-3.801e+00	5.390e-02	2.664e-01
1.000	-2.449e-15	-1.776e-15	-2.449e-15	6.729e-16	0.000e+00

$\max|f-S|=7.674e-02$
 $\max|f-P|=2.664e-01$

