

插值逼近——Newton 插值

湘潭大学, 数学与计算科学学院, 21 级王艺博

Lagrange 插值

- 优点：容易得到，公式紧凑
- 不足：当增删节点时，计算需重新进行

解决方案：采用新的基函数，逐次生成多项式。下面介绍 Newton 插值

一、Newton 插值

已知 $n+1$ 个不同的样本点 $(x_i, y_i), i = 0, 1, 2, \dots, n$

n 次 Newton 插值多项式

$$N_n(x) = C_0\phi_0(x) + C_1\phi_1(x) + \cdots + C_n\phi_n(x)$$

其使用的基函数如下：

$$\phi_0(x) = 1$$

$$\phi_1(x) = (x - x_0)$$

$$\phi_2(x) = (x - x_0)(x - x_1)$$

$$\vdots$$

$$\phi_n(x) = (x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

基函数应满足

$$\begin{pmatrix} \phi_0(x_0) & \phi_1(x_0) & \cdots & \phi_n(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_n(x_1) \\ \vdots & \vdots & & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \cdots & \phi_n(x_n) \end{pmatrix} \begin{pmatrix} C_0 \\ C_1 \\ \vdots \\ C_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

即

$$\begin{pmatrix} \phi_0(x_0) & 0 & \cdots & 0 \\ \phi_0(x_1) & \phi_1(x_1) & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \cdots & \phi_n(x_n) \end{pmatrix} \begin{pmatrix} C_0 \\ C_1 \\ \vdots \\ C_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

解这个方程组，可以得到 C_0, C_1, \dots, C_n

结合差商的定义 $C_0 = f(x_0), C_1 = f[x_0, x_1], C_2 = f[x_0, x_1, x_2], \dots, C_n = f[x_0, x_1, x_2, \dots, x_n]$

得

$$N_n(x) = f(x_0) + f[x_0, x_1](x - x_0) + \cdots + f[x_0, x_1, \cdots, x_n](x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

其中，差商可由差商表进行计算，

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$	\cdots
x_0	$f[x_0]$				
x_1	$f[x_1]$	$f[x_0, x_1]$			
x_2	$f[x_2]$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$		
x_3	$f[x_3]$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$	$f[x_0, x_1, x_2, x_3]$	
\cdots					

二、算法的简单介绍

♡ Newton 插值 (这里只强调差商表对计算的节省)

输入:

- $n + 1$ 个样本点的横纵坐标分别构成的向量 x_0, y_0
- 目标近似点 x

输出:

- 近似点的值 y

实现步骤

- 步骤 1: 计算差商表, 用矩阵 D 表示, 有 n 行, $n + 1$ 列

$$\begin{array}{ccccccc}
 x_0 & f[x_0] & & & & & \\
 x_1 & f[x_1] & f[x_0, x_1] & & & & \\
 x_2 & f[x_2] & f[x_1, x_2] & f[x_0, x_1, x_2] & & & \\
 x_3 & f[x_3] & f[x_2, x_3] & f[x_1, x_2, x_3] & f[x_0, x_1, x_2, x_3] & & \\
 \cdots & & & & & &
 \end{array}$$

- 步骤 2: 计算基函数 $\phi_i(x)$
- 步骤 3:

$$N_n(x) = f(x_0) + f[x_0, x_1](x - x_0) + \cdots + f[x_0, x_1, \cdots, x_n](x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

三、北太天元源程序

计算差商表

```
function [D] = divided_differences(x0,y0,c)
% Newton插值中用到的差商计算
% Input: 互异样本点构成的向量 (横) x0, (纵) y0, c:控制列循环的次数
% Output: 差商表 D
% Version: 1.0
% last modified: 04/01/2024
n = length(x0);
D = zeros(n,n+1);
D(:,1) = x0 ;D(:,2) = y0;
if nargin < 3
    c = n+1;
end
% 列循环
for k = 3:1:c
    % 行循环
    for i =k-1:1:n
        D(i,k) = D(i,k-1) - D(i-1,k-1);
        D(i,k) = D(i,k)/(x0(i)-x0(i-(k-2)));
        %D % 取消注释 D 可以观察每次的变化
    end
end
end
end
```

将上述代码保存为 divided_differences.m 文件。

Newton 插值

```
function [N,D] = Newton_interp(x0,y0,x)
% Newton 插值法 插值逼近
% Input: 样本点构成的向量x0 (升序), (纵) y0
%       x0,y0 是行向量
%       x
% Output: N 插值结果 , D 差商表
% Version: 1.0
% last modified: 04/01/2024
n = length(x0);
D = divided_differences(x0,y0);
D_need = diag(D(:,2:n+1)); % 列
% 表示基函数
n1 = length(x);
for k = 1:1:n1
    xjx = [1,x(k)-x0(1:n-1)];
    for i = 2:length(xjx)
        xjx(i) = xjx(i)*xjx(i-1);
    end
end
```

```

    N(k) = xjx * D_need;
end
end

```

将上述代码保存为 `Newton_interp.m` 文件。

四、数值算例

例 1 利用 $f(x) = \ln x$ 的如下数据：

x	0.4	0.5	0.6	0.7	0.8
$\ln x$	-0.916291	-0.693147	-0.510826	-0.357765	-0.223143

进行 *Newton* 插值：

简单实现一下

```

%% 测试差商计算
clc;clear;
x0 = [0.4 0.5 0.6 0.7 0.8];
y0 = [-0.916291 -0.693147 -0.510826 -0.357765 -0.223143];
D =divided_differences(x0,y0,)

%% Newton插值 test 1
clc;clear;
x0 = [0.4 0.5 0.6 0.7 0.8];
y0 = [-0.916291 -0.693147 -0.510826 -0.357765 -0.223143];
x = linspace(0.4,0.8);
y = Newton_interp(x0,y0,x)
figure(1)
    plot(x,y,'b')
    hold on
        plot(x,log(x),'r')
    hold off
    legend('N(x)','ln(x)')

%%
clc;clear all;
x0 = linspace(0,10,5);
y0 = sin(x0);
x= linspace(0,10,100);
y = sin(x);
[s,M] = Newton_interp(x0,y0,x);
plot(x,y,'r');
hold on
    plot(x,s,'b');

```

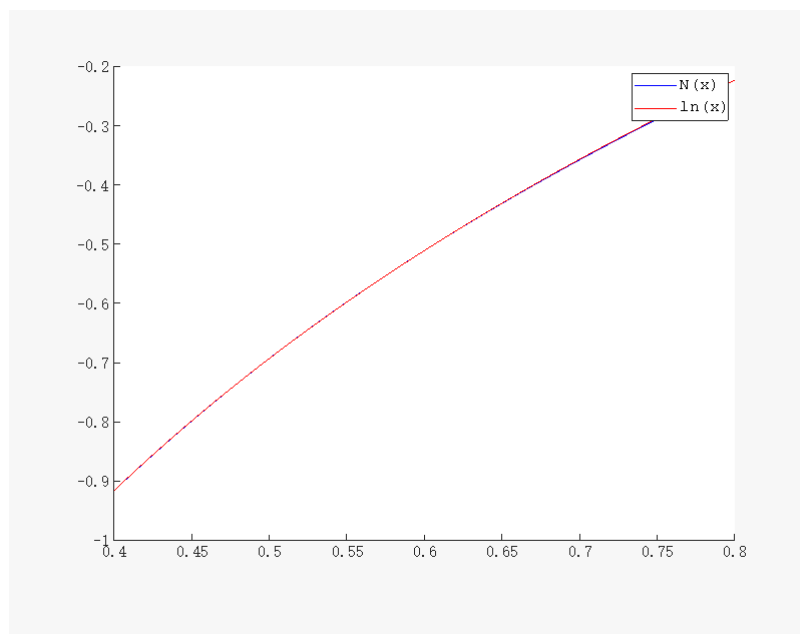


图1 $N(x)$ 与 $\ln x$

```
hold off
```

将上述代码保存为 `test_1.m` 文件。

这次写的 Newton 插值代码只强调了差商计算表对计算的一些简便之处，对于一组相同的样本点 x_0, y_0 ，不同的近似点 x ，有一张共用的差商计算表，从而节省一定的计算量。

如果是在原有样本点的基础上添加新的样本点，需要注意将其的横坐标按升序重新排列，再结合旧的差商表，节省计算量的同时计算出新的差商表，这里不再进行探讨，如果有兴趣的话，可以自己实现一下。