MATLAB插 值 法

作者：凯鲁嘎吉 - 博客园
http://www.cnblogs.com/kailugaji/

一、实验目的

在已知$f(x), x\in[a,b]$的表达式，但函数值不便计算，或不知$f(x), x\in[a,b]$而又需要给出其在$[a,b]$上的值时，按插值原则$f(x_i)=y_i\ (i=0,1,\cdots,n)$求出简单函数$P(x)$(常是多项式)，使其在插值基点$x_i$处成立$P(x_i)=y_i(i=0,1,\cdots,n)$，而在$[a,b]$上的其它点处成立$f(x)\approx P(x)$。

二、实验原理

1. Lagrange多项式插值

插值基函数： $l_j(x)=\frac{(x-x_0)\cdots(x-x_{j-1})(x-x_{j+1})\cdots(x-x_n)}{(x_j-x_0)\cdots(x_j-x_{j-1})(x_j-x_{j+1})(x_j-x_n)}$，$(j=0,1,2,\cdots,n)$

插值多项式： $L_n(x)=l_0(x)y_0+l_1(x)y_1+\cdots+l_n(x)y_n=\sum_{j=0}^{n}l_j(x)y_j$.

余项： $R_n(x)=\frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n),\xi\in(a,b)$.

2. Newton多项式插值

$$N_n(x)=f(x_0)+f[x_0,x_1](x-x_0)+f[x_0,x_1,x_2](x-x_0)(x-x_1)$$
$$+\cdots+f[x_0,x_1,\cdots,x_n](x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

余项 $R_n(x)=f[x,x_0,x_1,\cdots,x_n](x-x_0)(x-x_1)\cdots(x-x_n)$.

3. Hermite多项式插值

插值基函数： $h_j(x)=[1-2l_j'(x_j)(x-x_j)]l_j^2(x),(j=0,1,2,\cdots,n)$, $l_j'(x_j)=\sum_{\substack{k=0\\k\neq j}}^{n}\frac{1}{x_j-x_k}$.

$\bar{h}_j(x)=(x-x_j)l_j^2(x)$, $(j=0,1,2,\cdots,n)$.

插值多项式： $H_{2n+1}(x)=\sum_{j=0}^{n}h_j(x)y_j+\sum_{j=0}^{n}\bar{h}_j(x)y_j'$.

余项： $R_{2n+1}(x)=\frac{f^{(2n+2)}(\xi)}{(2n+2)!}[(x-x_0)(x-x_1)\cdots(x-x_n)]^2,\xi\in(a,b)$.

4. 三次样条插值 −−参 $P37$ 式(8.17),(8.19),(8.20)及倒数第2行式子

按三弯矩方程 $S''(x_j)=M_j$ 推导出的三变矩 M_j 的方程组

$$\mu_jM_{j-1}+2M_j+\lambda_jM_{j+1}=d_j=6f[x_{j-1},x_j,x_{j+1}],j=1,2,\cdots,n-1.$$

(1)端点条件：$S(x_0)=f_0',S(x_n)=f_n'\Rightarrow$ 端点方程

$$2M_0+M_1=\frac{6}{h_0}(f[x_0,x_1]-f_0'),M_{n-1}+2M_n=\frac{6}{h_{n-1}}(f_n'-f[x_{n-1},x_n])$$

与方程组构成 $n+1$ 个方程 $n+1$ 个未知量 M_j 的方程组，用"追赶法"求 $M_j(j=0,1,\cdots,n)$.

(2)端点条件：$M_0=f_0'',M_n=f_n''$，代入方程组可直接用"追赶法"再解 $M_1,M_2,\cdots,M_n$.

三、实验程序

前三个插值程序均易编程.样条插值问题求解,遇到三对角线方程组求解,可按提示编程:

追:可以推得递推式 $x_k = p_k - q_k x_{k+1}$, $k = 0,1,2,\cdots,n-1$ (*)

其中: $p_0 = \frac{d_0}{b_0}$, $q_0 = \frac{c_0}{b_0}$; $p_k = \frac{d_k - a_k p_{k-1}}{b_k - a_k q_{k-1}}$, $q_k = \frac{c_k}{b_k - a_k q_{k-1}}$, $k = 2,3,\cdots,n-1$

赶: 将最后一个式子 $x_{n-1} = p_n - q_n x_n$ 代入第 $n+1$ 个方程 $a_n x_{n-1} + b_n x_n = d_n$ 得

$$x_n = \frac{d_n - a_n p_{n-1}}{b_n - a_n q_{n-1}},$$

依次由式(*): $x_n \to x_{n-1} \Rightarrow \cdots \Rightarrow x_1 \Rightarrow x_0$.

以上即解三对角线方程组的"追赶法".

四、实验内容

四、实验内容

求 $f(x) = x^4$ 在 $[0,2]$ 上按 5 个等距节点确定的 Lagrange, Newton 插值多项式.

五、解答

1. 程序

(1) Lagrange插值多项式

```
function [C, L,L1,l]=lagran1(X,Y)
%输出C为插值多项式的系数，L为插值多项式，L1为l的系数，l为小l
%输入数据表X=[];Y=[];行向量
m=length(X); L=ones(m,m);
for k=1: m
    V=1;
    for i=1:m
     if k~=i
        V=conv(V,poly(X(i)))/(X(k)-X(i));
     end
    end
L1(k,:)=V; l(k,:)=poly2sym (V);
end
C=Y*L1;L=Y*l;
```

（2）Newton插值多项式

```
function [A,C,L,wcgs,Cw]= newploy(X,Y)
n=length(X); A=zeros(n,n); A(:,1)=Y';
q=1.0; c1=1.0;
for  j=2:n
   for i=j:n
       A(i,j)=(A(i,j-1)- A(i-1,j-1))/(X(i)-X(i-j+1));
   end
   b=poly(X(j-1));q1=conv(q,b); c1=c1*j;  q=q1;
end
C=A(n,n); b=poly(X(n)); q1=conv(q1,b);
for k=(n-1):-1:1
  C=conv(C,poly(X(k))); d=length(C); C(d)=C(d)+A(k,k);
end
L(k,:)=poly2sym(C); Q=poly2sym(q1);
syms M
wcgs=M*Q/c1; Cw=q1/c1;
```

2. 运算结果

```
（1）
>> X=[0:0.4:2];
>> Y=X.^4;
>> [C, L,L1,l]=lagran1(X,Y)

C =

    0.0000    1.0000         0   -0.0000         0         0


L =

x^4


L1 =

   -0.8138    4.8828  -11.0677   11.7188   -5.7083    1.0000
    4.0690  -22.7865   46.2240  -40.1042   12.5000         0
   -8.1380   42.3177  -76.8229   55.7292  -12.5000         0
    8.1380  -39.0625   63.8021  -40.6250    8.3333         0
   -4.0690   17.9036  -26.6927   15.8854   -3.1250         0
    0.8138   -3.2552    4.5573   -2.6042    0.5000         0
```

```
l =

 - (625*x^5)/768 + (625*x^4)/128 - (2125*x^3)/192 + (375*x^2)/32 - (137*x)/24 + 1
       (3125*x^5)/768 - (4375*x^4)/192 + (8875*x^3)/192 - (1925*x^2)/48 + (25*x)/2
     - (3125*x^5)/384 + (8125*x^4)/192 - (7375*x^3)/96 + (2675*x^2)/48 - (25*x)/2
            (3125*x^5)/384 - (625*x^4)/16 + (6125*x^3)/96 - (325*x^2)/8 + (25*x)/3
    - (3125*x^5)/768 + (6875*x^4)/384 - (5125*x^3)/192 + (1525*x^2)/96 - (25*x)/8
                (625*x^5)/768 - (625*x^4)/192 + (875*x^3)/192 - (125*x^2)/48 + x/2
```

(2)

```
>> X=[0:0.4:2];
>> Y=X.^4;
>> [A,C,L,wcgs,Cw]= newploy(X,Y)

A =

         0         0         0         0         0         0
    0.0256    0.0640         0         0         0         0
    0.4096    0.9600    1.1200         0         0         0
    2.0736    4.1600    4.0000    2.4000         0         0
    6.5536   11.2000    8.8000    4.0000    1.0000         0
   16.0000   23.6160   15.5200    5.6000    1.0000    0.0000


C =

    0.0000    1.0000    0.0000   -0.0000    0.0000         0


L =

(57*x^5)/18014398509481984 + x^4 + (209*x^3)/9007199254740992 - (525*x^2)/36028797018963968 + (213*x)/72057594037927936


wcgs =

-(M*(- x^6 + 6*x^5 - (68*x^4)/5 + (72*x^3)/5 - (4384*x^2)/625 + (768*x)/625))/720


Cw =

    0.0014   -0.0083    0.0189   -0.0200    0.0097   -0.0017         0
```

3. 拓展

3.(拓展（方法改进、体会等））

已知 $\sin 30^\circ=0.5$ ， $\sin 45^\circ=0.707\,1$， $\sin 60^\circ=0.866\,0$ ，用拉格朗日插值及其误差估计的MATLAB主程序求 $\sin 40^\circ$ 的近似值，并估计其误差.

解　源程序：

```
function [y,R]=lagran2(X,Y,x,M)
%输入X=[];Y=[];行向量，x预测点，可以一个，也可以为矩阵x=[];M为x的个数，
n=length(X); m=length(x);
for i=1:m
   z=x(i);s=0.0;
   for k=1:n
       p=1.0; q1=1.0; c1=1.0;
     for j=1:n
         if j~=k
            p=p*(z-X(j))/(X(k)-X(j));
         end
         q1=abs(q1*(z-X(j)));c1=c1*j;
     end
     s=p*Y(k)+s;
   end
   y(i)=s;
end
R=M*q1/c1;
```

在MATLAB工作窗口输入程序

\>\> x=2*pi/9; M=1; X=[pi/6 ,pi/4, pi/3];

Y=[0.5,0.7071,0.8660]; [y,R]=lagran2(X,Y,x,M)

运行后输出插值*y*及其误差限*R*为

y =

0.6434

R =

8.8610e-04