1.同一坐标下绘制在内的曲线图。

代码：

t = linspace(0,2\*pi);

plot(t,t.^2);

hold on;

plot(t,-t.^2);

hold on;

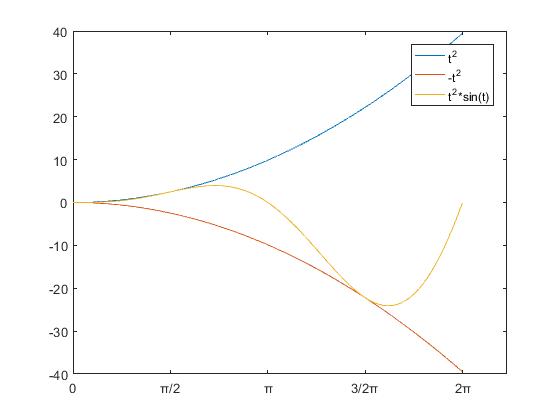
plot(t,t.^2.\*sin(t));

set(gca,'XTick',[0:0.5\*pi:2\*pi]);

set(gca,'xtickLabel',{'0','π/2','π','3/2π','2π'});

legend('t^2','-t^2','t^2\*sin(t)');

结果图：



2. 用LU分解法求解方程组



代码：

function[L, U] = LU(a)

[n, m]=size(a);

L=eye(n); U=zeros(n);

for k=1:n %对于U是第k行，对于L是第k列

%计算U矩阵元素U(k,j) 第k行

for j=k:n

z=0; %用来存储累加项

for p=1:k-1 %由于U矩阵是上三角矩阵，行下标k不大于列下标j，所以累加是以行下标减1(k-1)为标准

z=z+L(k,p)\*U(p,j);

end

U(k,j)=a(k,j)-z;

end

if abs(U(k, k))<1e-6

return;

end

%计算L矩阵元素L(i,k) 第k列

for i=k+1:n

z=0;

for p=1:k-1

z=z+L(i,p)\*U(p,k);

end

L(i,k)=(a(i,k)-z)/U(k,k);

end

end

求解得出的L矩阵及U矩阵：

[L,U]=LU([4,2,-2;2,2,-2;-2,-3,13]);

L =

1.0000 0 0

0.5000 1.0000 0

-0.5000 -2.0000 1.0000

U =

4 2 -2

0 1 -1

0 0 10

求出Y向量：

Y=L^-1\*[8;4;5];

得出要求解的X向量：

X=U^-1\*Y;

X =

2.0000

0.9000

0.9000

3. 已知，用插值法估计的近视值。

拉格朗日插值法代码：

function v = lagrange\_1(x, y, u)

% x 插值点

% y 插值

% u 计算序列点(就是需要插值的x点)

% v 计算序列点值(对应插值x点的y值)

% 整个矩阵计算，一步到位

n = length (x);

v = zeros(size(u));

for k =1:n

w = ones (size(u));

for j = [1:k-1 k+1:n] %这个遍历跳过了k

w = (u-x(j))./(x(k)-x(j)).\*w;

end

v = v+w\*y(k);

end

结算结果：

v=lagrange\_1([pi/6,pi/4,pi/3],[0.5,0.7071,0.8660],2\*pi/9)

v =

0.6434

牛顿插值法代码：

function s = Newton(x, y, nn,x0)

% Newton插值， x与y为已知的插值点及其函数值

% x0为要求的插值点的x坐标值，nn为Newton插值多项式的阶数

nx = length(x);

ny = length(y);

if nx ~= ny

warning( '矢量x与y的长度应该相同');

return;

end

if nn > ny-1

warning( '阶数过高');

return;

end

%创建插值公式

yy = zeros(nn); %创建矩阵保存各级差商

for i = 1:nn %按列生成各级差商

for j = 1:nn %每列按行生成

%分母处规律均为(x(j-(i-1))-x(j+1))

%一阶差商

if i==1

yy(j,i)=(y(j)-y(j+1))/(x(j-(i-1))-x(j+1));

%二阶差商及以上

elseif j>=i

yy(j,i)=(yy(j-1,i-1)-yy(j,i-1))/(x(j-(i-1))-x(j+1));

end

end

end

%用得到的牛顿插值公式计算

m = length(x0);

q=diag(yy);%仅需用对角元素计算插值

t=y(1);

%按照公式，对要求的插值点x0的每个元素进行计算

for i = 1:m

for j = 1:length(q)

r=q(j);

for p = 1:j

r=r\*(x0(i)-x(p));

end

t=t+r;

end

s(i)=t;

end

计算结果：

s = Newton([30/180\*pi,45/180\*pi,60/180\*pi],[0.5,0.7071,0.8660],2,40/180\*pi)

s =

0.6434