%利用正交多项式进行曲线拟合

%要求体现多项式做拟合的优越性

%函数代码如下

%以多项式为例进行的拟合分析

%Legendre

syms x n

a = input('a');

b = input('b');

y = input('y');

m = input('m');

for i = 1:m

phing = legendreP(i-1,x)/sqrt(2/(2\*i-1));

B(i) = int(str2sym(y)\*phing,-1,1);

end

number = 0;

hit = eval(y);

c=inv(eye(m))\*B';

Q = 0;

for i = 1:m

Q = Q + c(i)\*legendreP(i-1,x)/sqrt(2/(2\*i-1));

end

x = linspace(a,b,m+1);

t = x;

for i = 1:m+1

x = t(i);

number = number + (hit(i)-eval(Q))^2;

end

number=t;

disp(number);

x = linspace(a,b,200);

y1 = eval(Q);

plot(x,y1);

hold on

end

%yibanhanshu

syms x n

P = x^n;

a = input('a');

b = input('b');

y = input('y');

m = input('m');

A = zeros(m,m);

for i = 1:m

for j = 1:m

phing = subs(P,'n',i-1);

phiing = subs(P,'n',j-1);

A(i,j) = int(phing\*phiing,a,b);%AËæ×ÅnµÄÔö´ó»áÇ÷ÓÚ²¡Ì¬£¡

end

end

B = zeros(1,m);

for i = 1:m

phing = subs(P,'n',i-1);

B(i) = int(str2sym(y)\*phing,a,b);

end

c = inv(A)\*B';

Q = 0;

for i = 1:m

Q = Q + c(i)\*subs(P,'n',i-1);

end

number = 0;

x = linspace(a,b,m+1);

ht = eval(y);

t = x;

for i = 1:m+1

x = t(i);

number = number + (ht(i)-eval(Q))^2;

end

number = toc;

disp(number);

x = linspace(a,b,200);

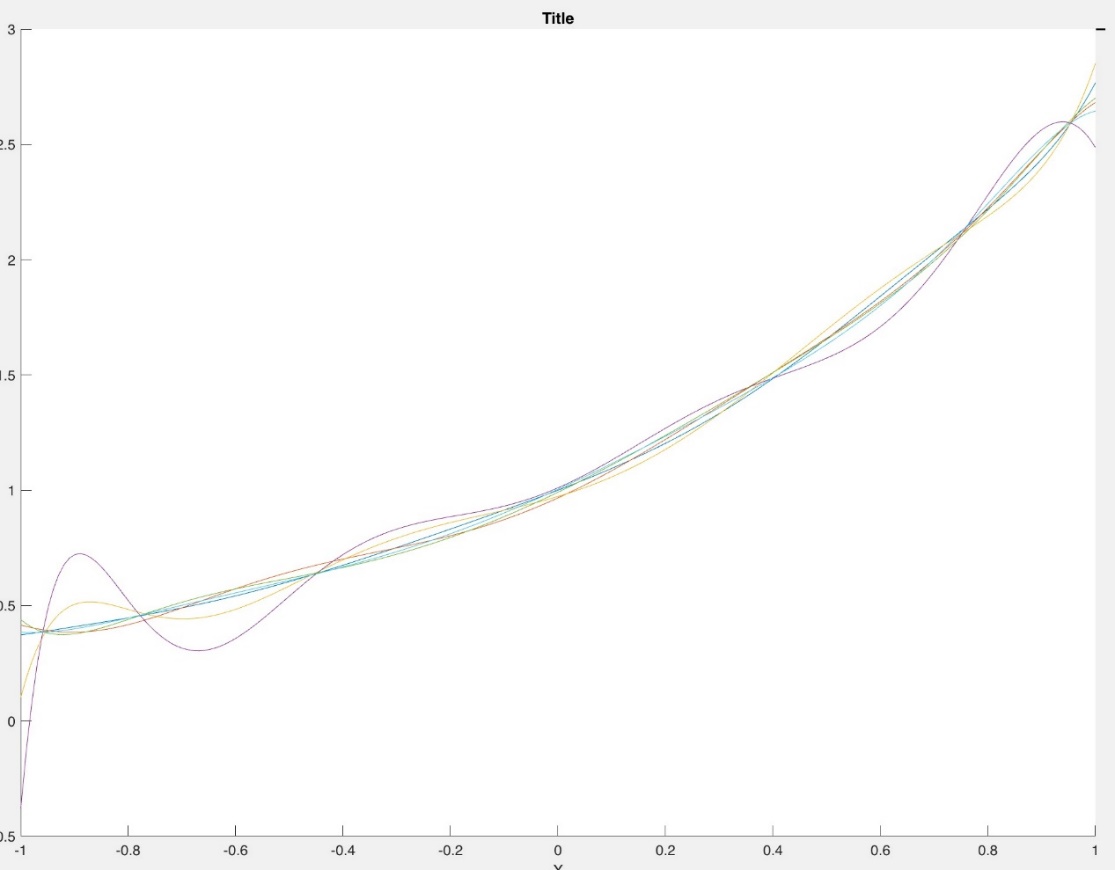
y1 = eval(Q);

plot(x,y1);

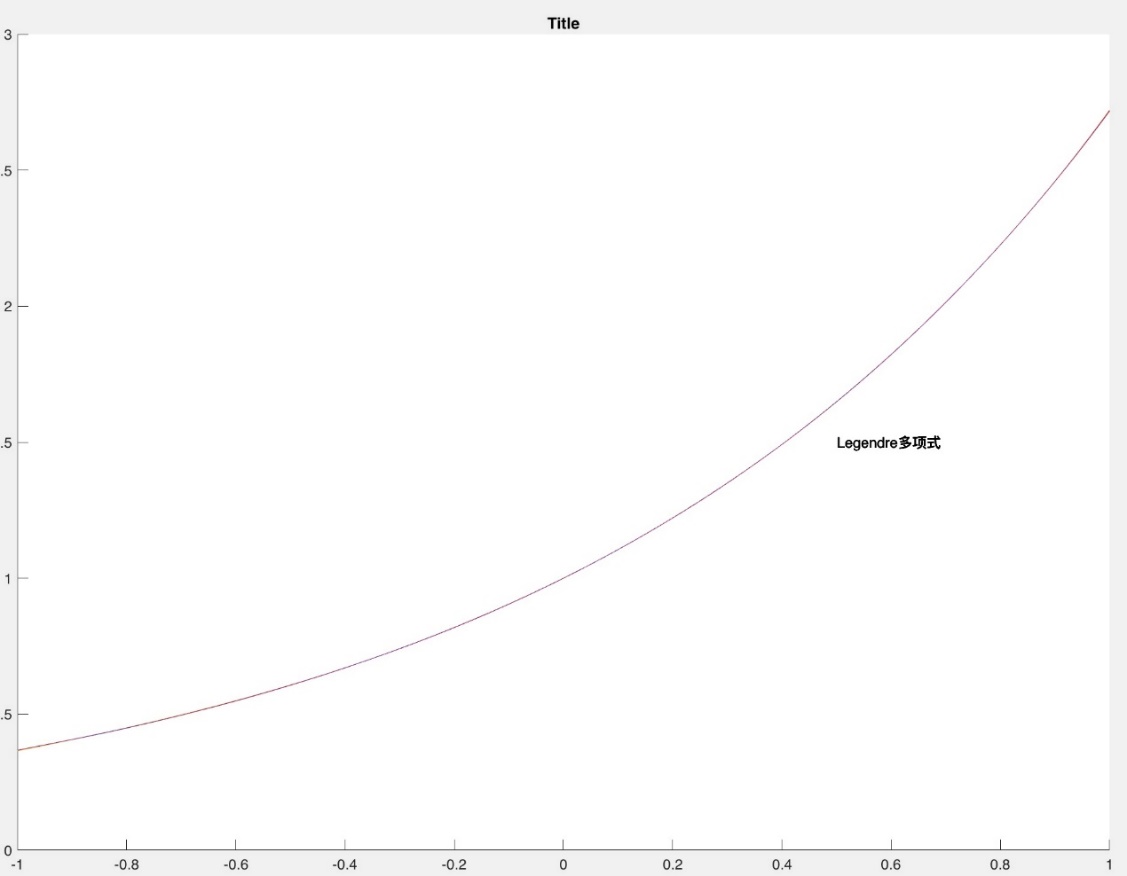
hold on

结论如下：对于一般多项式，依照平方逼近算法绘图；而对于正交多项式Legendre多项式正交，在规范化之后，系数矩阵是单位阵，造成的体现结果会有所不同。对于一般多项式，我们分别对A和b给予微小的扰动，得到的拟合曲线误差变化情况：从图中可以明显看出，一般多项式受到微小的扰动后，误差变化很大。相反，正交规范多项式却几乎没有任何变化。个人认知：正交函数的系数矩阵是一个单位阵，因此稳定性特别好，而一般多项式函数系张成的系数矩阵随n的增大而不稳定，边缘极易出现龙格现象

图像如下：



一般多项式收到扰动后



规范正交受到扰动后