
非线性方程的数值解第二次作业

一、问题（方程组）：

设映射 $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ 定义为

$$F_1(x) = x_1 - e^{\cos(\frac{1}{n+1}(x_1+x_2))}$$

$$F_i(x) = x_i - e^{\cos(\frac{1}{n+1}(x_{i-1}+x_i+x_{i+1}))}$$

$$F_n(x) = x_n - e^{\cos(\frac{1}{n+1}(x_{n-1}+x_n))}$$

二、数值实验

实验：用牛顿迭代法求解上述非线性方程组的解

牛顿迭代法：

$$\begin{cases} x^{k+1} = x^k + \Delta x^k \\ F'(x^k)\Delta x^k + F(x^k) = 0 \end{cases}$$

用 Matlab2018b 进行牛顿迭代法的编程，方程的维数为

100，初始值 $x^0=[1,\dots,1]$ ，精度 $\varepsilon=10^{-6}$ 。

停止准则为以下两者

$$(1) \|x^{k+1} - x^k\| < \varepsilon * \|x^k\|$$

$$(2) \|F(x^k)\| < \varepsilon$$

三、实验程序(fun.m) (dfun.m) (newton1.m)

File:fun1.m(函数F)

```
function f = fun1(x,n)
f=[];
format rat
I=eye(n);
P=diag(ones(1,n))+diag(ones(1,n-1),1)+diag(ones(1,n-1),-1);
B=P/6;

X=[];
syms x
for k=1:n
    eval(sprintf('syms x%d',k));
    x(k)=sprintf('x%d',k);
    X=[X,x(k)];
end %定义未知数x向量

C=B*X';
D=cos(C);
E=exp(D);
A=I*X'+E; %既我的非线性方程组
f= A'; %既我的非线性方程组，按行输出，便于等会计算
```

File:dfun1.m(函数的 jacobin 矩阵)

```
function df=dfun1(x,n)
f=fun1(x,n);
df=[];
for h=1:n
    dfh=diff(f,sprintf('x%d',h));
    df=[df;dfh];
    y=0;
end
df; %我的jacobin 矩阵（不带点的）
end
```

File:newton.m(牛顿迭代法)

```
clear;clc
format short e;
n=100; %迭代次数
costs=[]; %记录每次迭代后的误差
xlast=[ ];

for i=1:n
    Karray={};
```

```
B(i) = {sprintf('x%d', i)};
Karray=[Karray, B];
end %函数所要的参数1
```

```
for i=1:n
    xlast=[xlast,1]; % 迭代初始值
end
eps = 0.000001; % 定位精度要求
```

```
disp('wait');
for i = 1:100
    C=[];
    for l=1:n
        P=xlast(l);
        C=[C,P];
    end %函数所要的参数2

    f = double(subs(fun1(xlast,n),Karray,{C}));
    df = double(subs(dfun1(xlast,n),Karray,{C})); % 得到雅克比矩阵
    format short e;
    delk=f/df; %delta k
    x=xlast- delk; % 牛顿法

    cost=(norm(-delk, 2));
    fprintf('第%d次误差\n',i);
    Cost %输出第i次误差

    costs = [costs, cost];
    if (cost) < norm(eps*xlast, 2) %迭代停止条件
        break;
    end
    xlast = x; % 更新迭代结果
end
disp('方程: ');
f=fun1(x,n) '

disp('定位坐标: ');
x
disp('迭代次数: ');
i
disp('误差总览表: ');
format short e
```

四、数值实验结果

	停止条件 1: $\ x^{k+1} - x^k\ < \varepsilon * \ x^k\ $		停止条件 2: $\ F(x^k)\ < \varepsilon$	
初始值	迭代次数	$\ x^{k+1} - x^k\ $	迭代次数	$\ F(x^k)\ $
[0, ..., 0]	1	27.18281828	1	27.18281828
	2	9.207523076	2	14.74672184
	3	0.431678826	3	0.743106864
	4	3.04E-05	4	2.77E-05
	5	3.78E-13	5	3.77E-13
[1, ..., 1]
	3	30.33457253	3	27.92264291
	4	11.93883392	4	18.38908032
	5	0.837697626	5	1.446653964
	6	0.000274334	6	0.000473777
[20, ..., 20]	7	2.47E-11	7	4.26E-11

	9	17.49526454	9	14.531269289013
	10	9.599137236	10	11.24358681
	11	1.609389575	11	2.464247536
	12	0.037460134	12	0.05483113
	13	1.33E-06	13	2.03E-06
			14	2.89E-15

五、问题

第一次用 matlab 编程，没有使用向量带点，用了过多的循环，导致程序过慢，以后用向量化带点。

附录：方程的值

x1	-2.20253	x21	-1.83581	x41	-1.83581	x61	-1.83581	x81	-1.83581
x2	-1.76124	x22	-1.83581	x42	-1.83581	x62	-1.83581	x82	-1.83581
x3	-1.851	x23	-1.83581	x43	-1.83581	x63	-1.83581	x83	-1.83581
x4	-1.83272	x24	-1.83581	x44	-1.83581	x64	-1.83581	x84	-1.83581
x5	-1.83644	x25	-1.83581	x45	-1.83581	x65	-1.83581	x85	-1.83581
x6	-1.83568	x26	-1.83581	x46	-1.83581	x66	-1.83581	x86	-1.83581
x7	-1.83584	x27	-1.83581	x47	-1.83581	x67	-1.83581	x87	-1.83581
x8	-1.8358	x28	-1.83581	x48	-1.83581	x68	-1.83581	x88	-1.83581
x9	-1.83581	x29	-1.83581	x49	-1.83581	x69	-1.83581	x89	-1.83581
x10	-1.83581	x30	-1.83581	x50	-1.83581	x70	-1.83581	x90	-1.83581
x11	-1.83581	x31	-1.83581	x51	-1.83581	x71	-1.83581	x91	-1.83581
x12	-1.83581	x32	-1.83581	x52	-1.83581	x72	-1.83581	x92	-1.83581
x13	-1.83581	x33	-1.83581	x53	-1.83581	x73	-1.83581	x93	-1.8358
x14	-1.83581	x34	-1.83581	x54	-1.83581	x74	-1.83581	x94	-1.83584
x15	-1.83581	x35	-1.83581	x55	-1.83581	x75	-1.83581	x95	-1.83568
x16	-1.83581	x36	-1.83581	x56	-1.83581	x76	-1.83581	x96	-1.83644
x17	-1.83581	x37	-1.83581	x57	-1.83581	x77	-1.83581	x97	-1.83272
x18	-1.83581	x38	-1.83581	x58	-1.83581	x78	-1.83581	x98	-1.851
x19	-1.83581	x39	-1.83581	x59	-1.83581	x79	-1.83581	x99	-1.76124
x20	-1.83581	x40	-1.83581	x60	-1.83581	x80	-1.83581	x100	-2.20253