

基本数值计算方法

第6周习题

更新时间： 2020.04.10

作业要求

1. 请在下周五(2020.04.17)之前在Canvas平台上交作业。 小组作业只需交**一份**。
2. 作业报告应包括
 - a) 对算法的简述（可以用思维导图，概念图，流程图）
 - b) 源码
 - c) 结果展示（图表）
 - d) 结果分析
 - e) 附件内容（如参考文献等）
3. 脚本文件的要求：
 - a) 首行加入注释：用途，作者，日期，输入变化和输出变量的简要说明
 - b) 程序主体的首行加入： `clear all; close all; clc`
 - c) 在程序中加入对变量及算符的注释
 - d) 等号两端加入空格。
4. 电子版文件名(若包含多个源文件，请放置在一个目录下后打包成一个文件)格式：第n周作业_姓名_学号.xxx, 或第n周小组作业_小组k.xxx.

小组作业

1. 请查阅Ridders方法^[1]，并应用此算法求解以下方程, 请将你的结果和Matlab内置函数fzero的结果比较：

$$T(x) = T_s + (T_i - T_s)\operatorname{erf}\left(\frac{x}{2\sqrt{\alpha t_m}}\right) = 0$$

其中 T 是随位置 x (用米计算)变化的温度函数，参数 $\alpha = 0.138 \times 10^{-6} \text{ m}^2/\text{s}$. 设起始温度 $T_i = 200^\circ$ ，极限温度 $T_s = -150^\circ$ 。时间间隔 $t_m = 60$ 天。误差函数 erf 的定义为

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

2. 一个引人注意的想法是结合二分法和牛顿法 - 先应用二分法，当达到一个设定的小区间后改用牛顿法来加快收敛速度。

请编写函数脚本 `bisection_Newton.m` 实现这样的算法：设置初始试探区间 $[a, b]$ ，当使用二分法达到的试探区间是原来的一部分 s (即区间长度为 $s(b - a)$) 后采用牛顿法。考虑到牛顿法有可能发散，所以当牛顿法的结果使得试探区间在二分法结果之外时，应该重新使用二分法。 s 将作为函数的输入变量，它的缺省值为 0.1。

尝试用你的函数在初始区间 $[-10, 15]$ 求解 $\tan(x) = 0$ 。

提示：下列说明函数缺省值的用法

```
x = 1:5;
y2 = demo_default(x);
y3 = demo_default(x,3);
function res = demo_default(x, exponent)
    if nargin < 2 % nargin: 函数输入变量的个数
        exponent = 2;
    end
    res = x.^exponent;
end
```

3. 运行以下程序，描述该程序解决的问题，并在每行加入注释。

```
g1 = @(v) -((24.43+1.8*v(2,1).^2)/3.2).^(1/3);
g2 = @(v) ((5.92+2*v(1,1).^2)/3).^(1/3);
tol = 1e-3; kmax = 10;
v(:,1) = [-1;-2];
for k = 1:kmax,
    v(:,k+1) = [g1(v(:,k)); g2(v(:,k))];
    if norm(v(:,k+1)-v(:,k)) < tol,
        break
    end
end
```

4. 应用定点迭代法求解非线性方程组

$$\begin{cases} 2e^x + y &= 0 \\ 3x^2 + 4y^2 &= 8 \end{cases}$$

提示：1) 先绘制函数曲线，确定根的大致位置。2) 选择适当的辅助函数。3) 请将结果与Matlab 内置函数 `fsolve` 的求解结果比较。

提示：下列说明 `fsolve` 的用法，求解的方程为

$$\begin{cases} e^{-e^{-(x_1+x_2)}} = x_2(1+x_1^2) \\ x_1 \cos(x_2) + x_2 \sin(x_1) = \frac{1}{2} \end{cases}$$

```
options = optimoptions('fsolve','Display','iter');
fun = @root2d;
x0 = [0,0];
x = fsolve(fun,x0,options)

function F = root2d(x)
    F(1) = exp(-exp(-(x(1)+x(2)))) - x(2)*(1+x(1)^2);
    F(2) = x(1)*cos(x(2)) + x(2)*sin(x(1)) - 0.5;
end
```

1. 请下载并阅读Ridders_method_Wiki.pdf [↩](#)