计 算 方 法

第二章　方程求根

计算方法课程组

华中科技大学数学与统计学院

§2 方程求根

- §2.0 引言
- §2.1 二分法
- §2.2 迭代法
- §2.3 牛顿 (Newton) 法
- §2.4 迭代过程的加速方法

§2　方程求根—引言

方程是在科学研究中不可缺少的工具，$f(x)=0$

方程求解是科学计算中一个重要的研究对象．

几百年前就已经找到了代数方程中二次至四次方程的求解公式；

但是，对于更高次数的代数方程目前仍无有效的精确解法；

对于无规律的非代数方程的求解也无精确解法．

因此，研究非线性方程的数值解法成为必然．

本节主要研究单根区间上方程求根的各种近似算法．

$x^3+ax^2+bx+c=0$ 求根公式

令 $A=\dfrac{a^2-3b}{9}$ and $B=\dfrac{2a^3-9ab+27c}{54}$

if $B^2-A^3>0$ obtain $x_1=-\mathrm{sgn}(B)(\beta+\dfrac{A}{\beta})-\dfrac{a}{3}$

where $\beta=(\sqrt{B^2-A^3}+|B|)^{1/3}$;

else

$x_1=2\sqrt{A}\cos(\dfrac{\gamma}{3})-\dfrac{a}{3}$,

$x_2=2\sqrt{A}\cos(\dfrac{\gamma+2\pi}{3})-\dfrac{a}{3}$,

$x_3=2\sqrt{A}\cos(\dfrac{\gamma+4\pi}{3})-\dfrac{a}{3}$,

where $\gamma=\arccos(-B/A^{3/2})$;

本章讨论非线性方程 $f(x)=0$ 的求根问题，

1. 其中一类特殊的问题就是多项式方程的求根。

$$f(x)=a_n x^n+a_{n-1}x^{n-1}+\cdots+a_1 x+a_0 \quad (a_n \neq 0)$$

2. 另一类就是超越方程的求根。

$$cos(x)cosh(x)+1=0$$

基本概念 $f(x)=0$

方程$f(x)=0$ 的根x^* 又称为$f(x)$ 的零点，它使$(x^*)=0$

若 $g(x^*)\neq 0$ ，$f(x)$ 可表示为$f(x)=(x-x^*)^m g(x)$ ，

其中 m 为正整数，且 $f(x^*)=0$ 。

当$m=1$ 时，称x^* 为单根，若$m>1$ 称x^* 为$f(x)$ 的m

重根，或 $f(x)$的 m 重零点。若 x^* 是 $f(x)$的 m 重零点

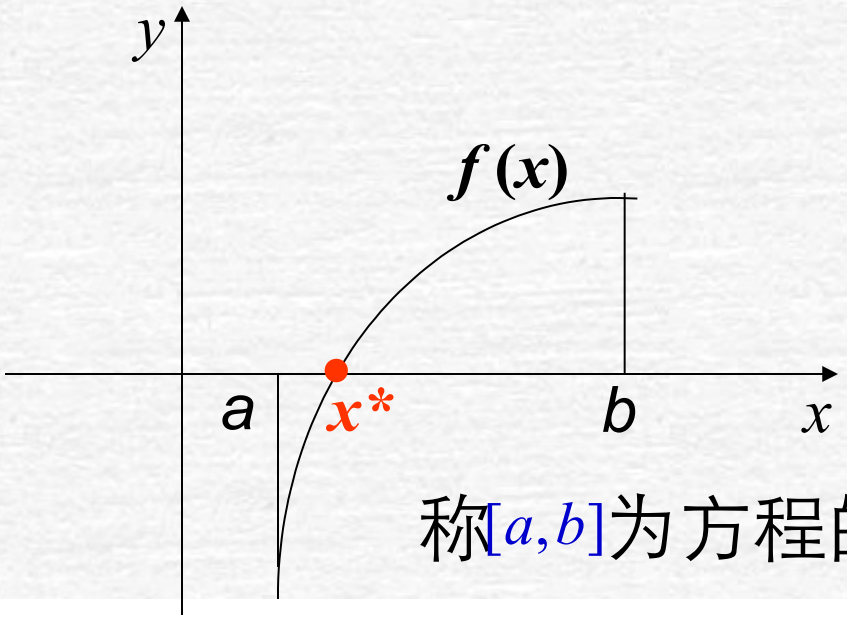
且 $g(x)$ 充分光滑，则

$$f(x^*)=f'(x^*)=\cdots=f^{(m-1)}(x^*)=0, f^{(m)}(x^*)\neq 0$$

§2.1 二分法

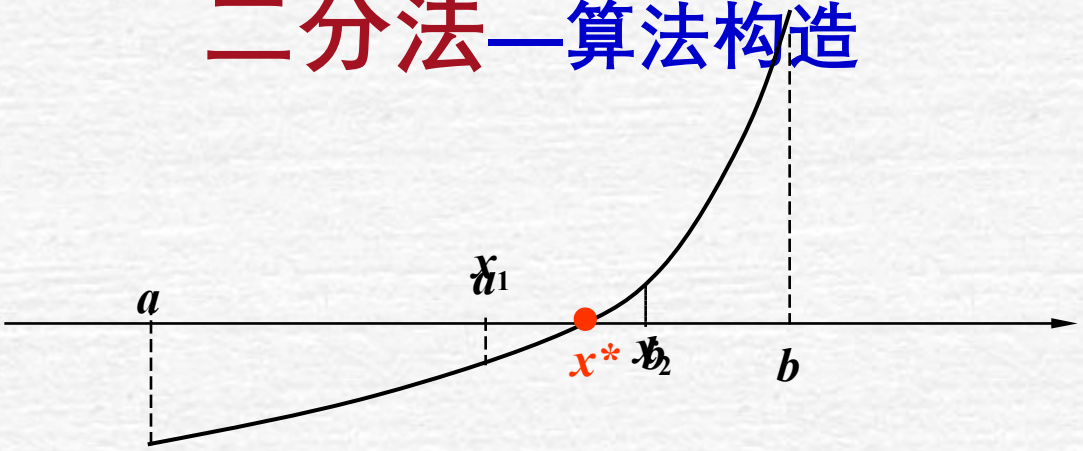
求 $f(x)=0$ 的根

原理：若 $f \in C[a,b]$，且 $f(a)\cdot f(b)<0$，则 f 在 (a,b) 上必有一根。



称$[a,b]$为方程的有根区间。

§2.1 二分法—算法构造



给定有根区间 $[a, b]$ $(f(a) \cdot f(b) < 0)$ 和 精度 ε 或 δ

1. 令 $x = (a+b)/2$

2. 如果 $b - a < \varepsilon$ 或 $f(x) < \delta$，停机，输出 x

3. 如果 $f(a) f(x) < 0$，则令 $b = x$，否则令 $a = x$，返回第 1 步

用二分法求根，通常先给出 $f(x)$ 草图以确定根的大概位置。

§2.1 二分法—误差分析

记 $a_1=a,\ b_1=b,$ 第 k 步的有根区间为 $[a_k,b_k]$

$$|x_k-x^*|=\left|\frac{b_k+a_k}{2}-x^*\right|\le\frac{b_k-a_k}{2}=\frac{b_{k-1}-a_{k-1}}{4}=\cdots=\frac{b_1-a_1}{2^k}$$

对于给定的精度 ε，可估计二分法所需的步数 k：

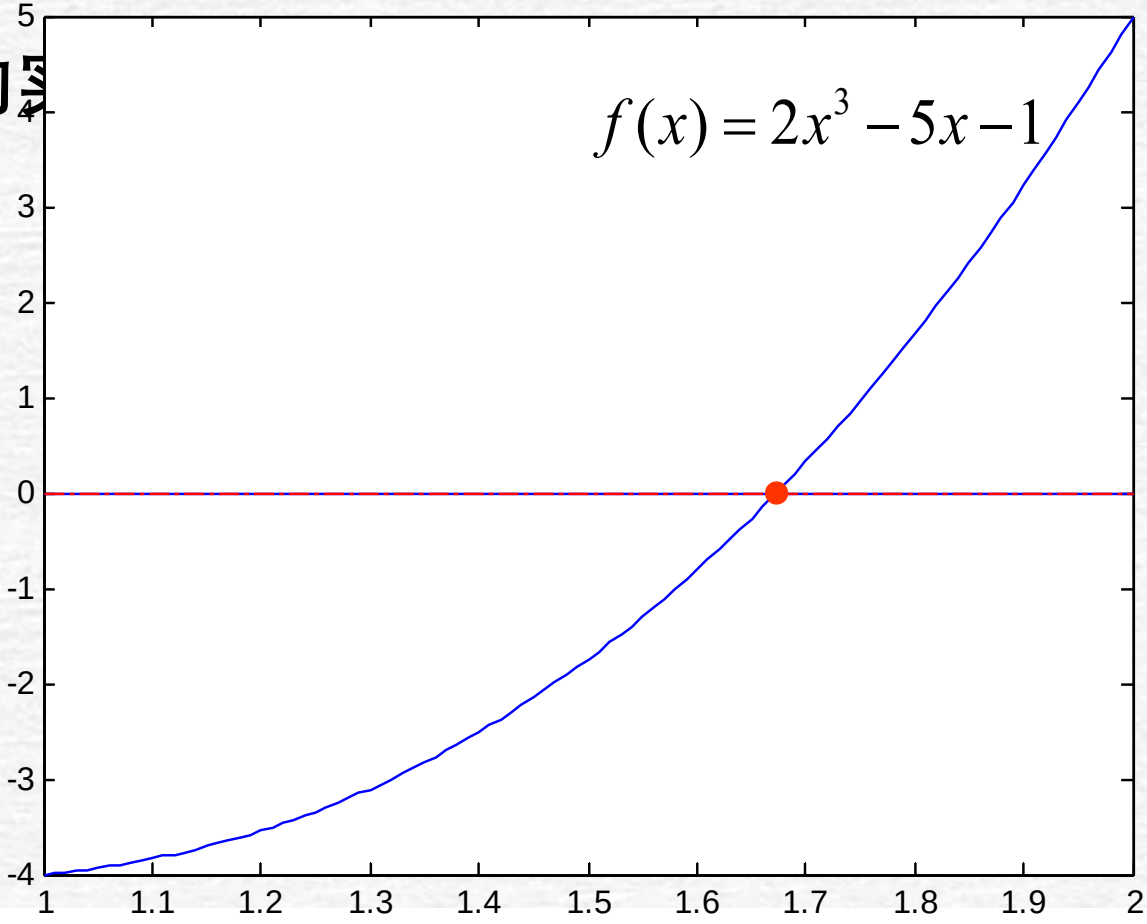
$$\frac{b-a}{2^k}<\varepsilon \quad \Leftarrow \quad k>\log_2\frac{b-a}{\varepsilon},\qquad \text{取}\ k=\left\lceil\log_2\frac{b-a}{\varepsilon}\right\rceil$$

- ✓ 简单易用
- ✓ 对 $f(x)$ 要求不高，只要连续即可收敛

- ✓ 收敛速度慢
- ✓ 无法求复根及偶重根

§2.1 二分法—例题分析

例 1： 用二分法求方程 $2x^3 - 5x - 1 = 0$ 在区间 (1,2) 内的近似根，要求 $\varepsilon \le 10^{-2}$



§2.1 二分法—例题分析

例 1: 用二分法求方程 $2x^3 - 5x - 1 = 0$ 在区间 (1,2) 内 $\varepsilon \le 10^{-2}$

解：令 $f(x) = 2x^3 - 5x - 1$，要求误差限为 。

$f(1)<0, f(2)>0$ 记 $I_0=[1,2]$, $x_0=(1+2)/2=1.5$

因为 $f(x_0)f(1)>0$ 得 $I_1=[1.5, 2]$, $x_1=(1.5+2)/2=1.75$

$f(x_1)f(1.5)<0$ 得 $I_2=[1.5, 1.75]$, $x_2=(1.5+1.75)/2=1.625$

.....

$I_6=[1.681875, 1.6875]$,

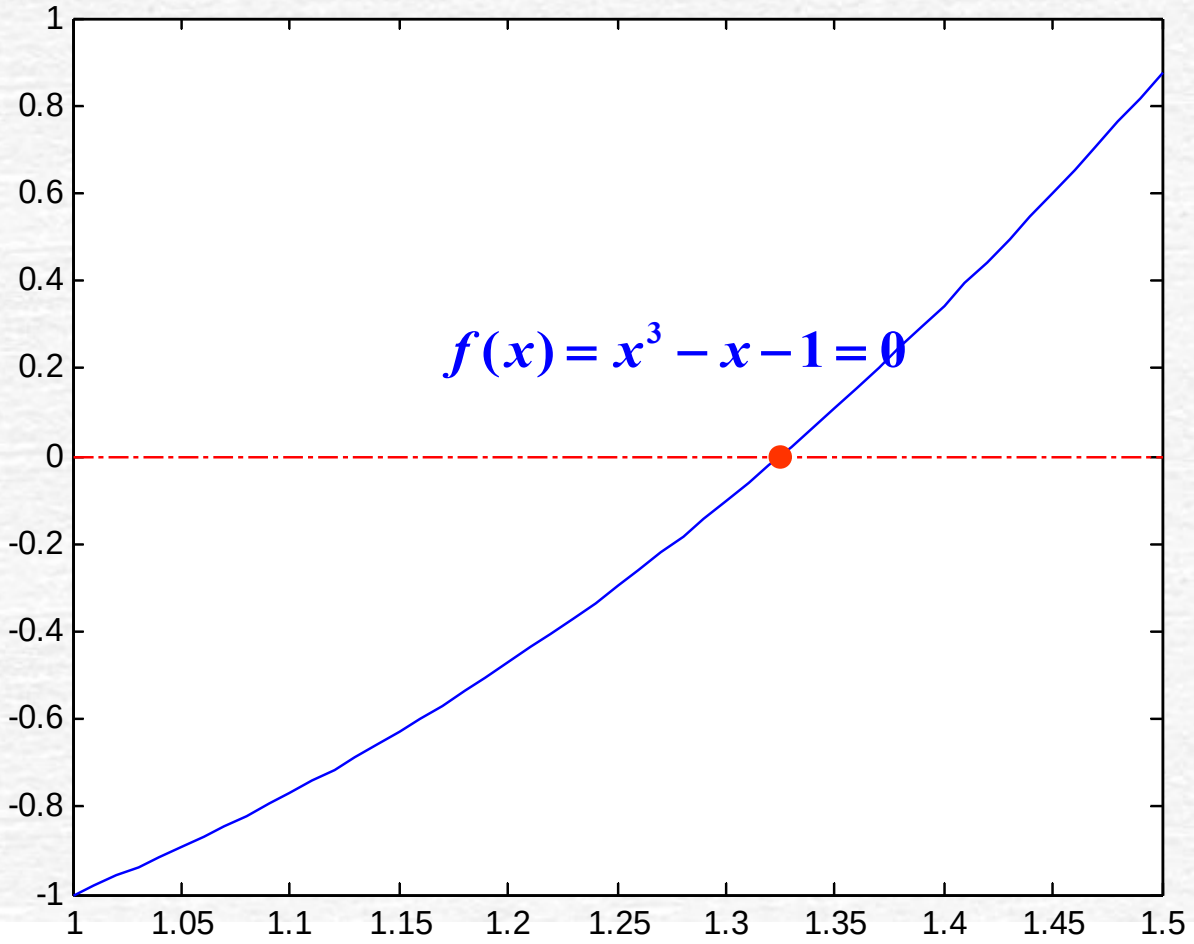
$I_7=[1.671875, 1.679688]$

$b_7 - a_7=0.7813\times10^{-2} < 10^{-2}$

$x^* \approx 1.6758$

§2.1 二分法—算法步骤

例 2 ： 求 $f(x)=x^3-x-1=0$ 在 $(1, 1.5)$ 的实根，要求误差不超过 0.005 。



§2.1 二分法—算法步骤

例 2 ： 求 $f(x)=x^3-x-1=0$ 在 (1, 1.5) 的实根，要求误差不超过 0.005 。

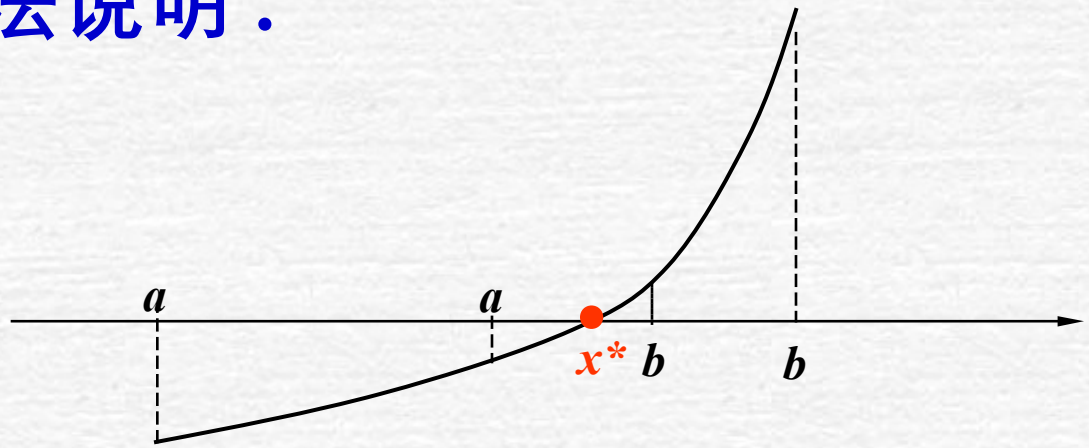
STEP 0	输入 *a*, *b*, *eps*, *delta* , *fa*=*f* (*a*) , *fb*=*f* (*b*)
STEP 1	*x*=(*a*+*b*)/2 , *fx*=*f* (*x*)
STEP 2	判断： |*b*-*a*|<*eps* or | *f x* |< *delta* 若是， goto step 4 ; 否则，执行下一步
STEP 3	若 *fb***fx*<0, 则 *a*=*x* 否则 *b*=*x*. goto step 1
STEP 4	输出 *x*, *fx*, 停机 .

```
function [xvect,xdif,fx,nit]=bisect(a,b,toll,nmax,fun)
err=toll+1;
nit=0;
xvect=[];
fx=[];
xdif=[];
while (nit < nmax & err > toll)
   nit=nit+1;
   c=(a+b)/2;
   x=c;
   fc=feval(fun,x);
   xvect=[xvect;x];
   fx=[fx;fc];
   x=a;
   if (fc*feval(fun,x) > 0)
     a=c;
   else
     b=c;
   end;
   err=abs(b-a);
   xdif=[xdif;err];
end
return
```

```
>> fun=inline('2*x.^3-5*x-1');
>> [xvect,xdif,fx,nit]=bisect(1,2,0.01,50,fun)
```

x 值	区间长	函数值
1.2500	0.2500	-0.2969
1.3750	0.1250	0.2246
1.3125	0.0625	-0.0515
1.3438	0.0313	0.0826
1.3281	0.0156	0.0146
1.3203	0.0078	-0.0187
1.3242	0.0039	-0.0021

程序算法说明：



$|x_{k+1}-x_k|<\varepsilon$ 或 $|f(x)|<\delta$

不能保证 x 的精度