计算方法

第一章 绪论

误差来源及分类

舍入误差：无限存储为有限产生的误差

截断误差（方法误差）：将无穷过程截断为有限过程

由于舍入误差的存在，任何近似方法都不应该取非常小的值来近似，太小的数可能会超出计算机的表示范围。

有效数字

设 x 的近似值有如下标准形式

$$x^* = \pm 10^m \cdot 0.x_1x_2\dots x_nx_{n+1}\dots x_p$$
$$|e^*| = |x^* - x| \le \frac{1}{2}\cdot 10^{m-n}$$

则称 x^* 为 x 的具有 n 位有效数字的近似数。

当 x^* 精确到末位，即 $n = p$，则称 x^* 为有效数。

误差的度量及传播

绝对误差：$e(x^*) = x^* - x$

相对误差：$e_r(x^*) = \frac{x^* - x}{x}^* = \frac{e(x^*)}{x}^*$

绝对误差限（ϵ^*）：有效数末尾数字所在单位的一半

相对误差限：$\epsilon_r^* = \frac{\epsilon^*}{|x^*|}$（无量纲）

若有函数 $y = f(x_1, x_2, \dots, x_n)$，用含有误差的近似值计算出的 y^*，其:

绝对误差

$$e(y^*) = \sum_{i=1}^{n} f_i' e(x_i^*)$$

相对误差

$$e_r(x^*) = \frac{1}{y^*}\sum_{i=1}^{n} f_i' x_i^* e_r(x_i^*)$$

二元函数算术运算误差传播规律

绝对误差限

$$
\begin{aligned}
\epsilon(x_1+x_2) &\approx \epsilon(x_1)+\epsilon(x_2) \\
\epsilon(x_1x_2) &\approx |x_2|\epsilon(x_1)+|x_1|\epsilon(x_2) \\
\epsilon(\frac{x_1}{x_2}) &\approx \frac{|x_2|\epsilon(x_1)+|x_1|\epsilon(x_2)}{|x_2^2|}
\end{aligned}
$$

相对误差限

$$
\begin{aligned}
\epsilon_r(x_1+x_2) &\approx max\{\epsilon_r(x_1),\epsilon_r(x_2)\} \\
\epsilon_r(x_1x_2) &\approx \epsilon_r(x_1)+\epsilon_r(x_2) \\
\epsilon_r(\frac{x_1}{x_2}) &\approx \epsilon_r(x_1)+\epsilon_r(x_2)
\end{aligned}
$$

算法设计原则

1. 避免相近数相减
2. 避免大数吃小数
3. 避免绝对值过大的数除以绝对值过小的数

第二章 非线性方程数值解法

二分法

二分法的优点是方法及相应的程序均简单，且对f(x)性质要求不高，只要连续即可。但二分法不能用于求复数根和偶数重根，且收敛速度比较慢。因此，一般常用该方法求根的初始近似值，然后再用其它的求根方法精确化。

```c
int search(int* nums, int numsSize, int target){
    int left = 0;
    int right = numsSize-1;
    int mid = 0;
    //若left小于等于right，说明区间中元素不为0
    while(left<=right) {
        //更新查找下标mid的值
        middle = (left+right)/2;
        //此时target可能会在[left,mid-1]区间中
        if(nums[mid] > target) {
            right = mid-1;
        }
        //此时target可能会在[mid+1,right]区间中
        else if(nums[mid] < target) {
            left = mid+1;
        }
        //当前下标元素等于target值时，返回mid
        else if(nums[middle] == target){
            return mid;
        }
    }
    //若未找到target元素，返回-1
    return -1;
```

```
}
```

误差估计：

$$|x_k - x^*| \le \frac{1}{2^k}(b-a) = |x_k - x_{k-1}| \le \epsilon$$

$$\text{对分次数}k \ge \frac{ln(b-a) - ln\epsilon}{ln2}$$

$$k\text{为当前次数，}k-1\text{为迭代次数，}k\text{为对分次数}$$

简单迭代法

$$x_{k+1} = \Phi(x_k)$$

全局收敛定理：

1. 迭代函数在区间上可导
2. 当 $x \in [a,b]$时，$\Phi(x) \in [a,b]$
3. 存在常数0<L<1，使得 $\forall x \in (a,b), |\Phi'(x)| \le L$

性质：

1. $x = \Phi(x)$ 在该区间上有唯一实根
2. 对于任意的 $x \in [a,b]$，迭代公式均收敛
3. $\lim_{k \to +\infty} \frac{x_{k+1} - x^*}{x_k - x^*} = \phi'(x^*)$

由全局收敛定理知，若$L \approx 1$，则必然收敛较慢；若$L << 1$，则收敛速度快

误差估计：

$$\text{后验误差估计}|x_k - x^*| \le \frac{L}{1-L}|x_k - x_{k-1}| \le \epsilon$$

$$\text{先验误差估计}|x_k - x^*| \le \frac{L^k}{1-L}|x_1 - x_0| \le \epsilon$$

$$k \ge \frac{ln\frac{\epsilon(1-L)}{|x_1 - x_0|}}{lnL}$$

迭代法的收敛阶

记迭代误差 $e_k = x_k - x^*$，若存在常数 $c > 0, p \ge 1$，使得

$$\lim_{k \to +\infty} \frac{|e_{k+1}|}{|e_k|^p} = c$$

则称p 阶收敛，且 p=1 时为线性收敛，且此时必然有0<c<1。

p>1 的时候为超线性收敛，p = 2称为平方收敛（二次收敛），p 越大收敛速度越快，若两个迭代方法相同的时候，c 越小收敛越快

Aitken 加速法

$$若 \lim_{k\to+\infty}\frac{|x_{k+1}-x_k|}{x_k-x^*}=c<1$$

$$则\ y_k=x_k-\frac{(x_{k+1}-x_k)^2}{x_{k+2}-2x_{k+1}+x_k}$$

Steffensen 迭代法

Steffensen 迭代在一定条件下可以达到二阶收敛。

$$\begin{cases} s=\phi(x_k) \\ t=\phi(s) \\ x_{k+1}=x_l-\dfrac{(s-x_k)^2}{t-2s+x_k} \end{cases}$$

Newton 迭代法

特点：较为依赖 x 点的选取（二阶收敛）

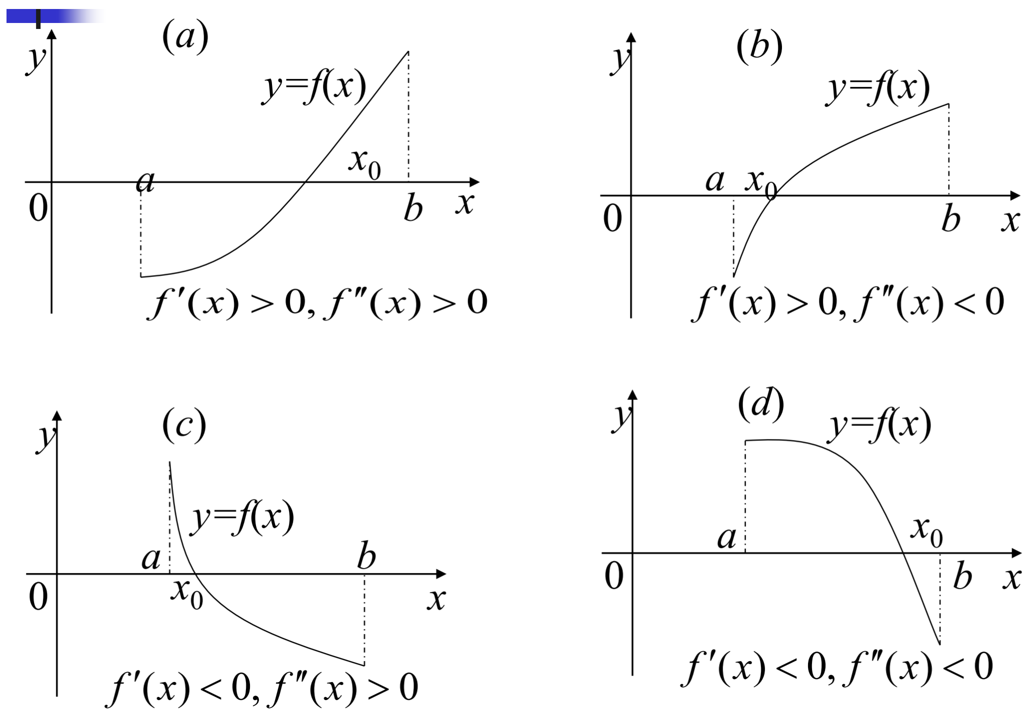
$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)}$$

全局收敛性：

①对于任意的 $x\in[a,b]$，均有 $f'(x),f''(x)$连续且不变号（确保单调性和凹凸性不变）

②对于初值的要求：$f(x_0)f''(x_0)>0$

$$\lim_{k\to+\infty}\left|\frac{x_{k+1}-x^*}{(x_k-x^*)^2}\right|=\left|\frac{f''(x^*)}{2f'(x^*)}\right|$$



~~Newton下山法~~

~~计算重根的牛顿迭代法~~

第三章 线性代数方程组的解法

向量和矩阵的范数

$$
列范数||A||_1 = \max_{1\leq j\leq n} \sum_{i=1}^{n} |a_{ij}|
$$

$$
行范数||A||_\infty = \max_{1\leq i\leq n} \sum_{j=1}^{n} |a_{ij}|
$$

$$
谱范数||A||_2 = \sqrt{\lambda_{max}(A^TA)}
$$

$$
F范数||A||_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}^2}
$$

收敛的充要条件：迭代矩阵的谱半径 $\rho(A) < 1$或 $\lim_{k\to+\infty} A^k = 0$

充分不必要： ||A||<1

Gauss 消去法

列主元消去法

$$
B=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i,1} & a_{i,2} & \cdots & a_{i,n} & b_i \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{bmatrix} = \left[A^{(1)} \vdots \vec{b}^{(1)}\right]
$$

在A中选取绝对值最大的元素作为主元素，如 $\left|a_{i_1,j_1}\right| = \max\limits_{\substack{1\leq i\leq n \\ 1\leq j\leq n}} \left|a_{ij}\right|$，然后交换$B$ 的第一行与第 i_1行，第一列与第 j_1 列，这时的 $a_{11}^{(1)}$就是元素的a_{i_1,j_1}，然后进行消元法的第一步，即

Doolittle 分解

A有唯一 Doolittle 分解的充要条件为：A的前n−1阶顺序主子式不为零。

$$
A = LU
$$

$$
\begin{cases} L^{-1}A = U \\ L^{-1}\vec{b} = \vec{y} \end{cases}
$$

设$A=LU$ 即

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ \vdots & & & \ddots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{bmatrix}$$

$$u_{kj} = a_{kj} - \sum_{m=1}^{k-1} l_{km} u_{mj}$$

$$l_{ik} = \frac{a_{ik} - \sum_{m=1}^{k-1} l_{im} u_{mk}}{u_{kk}}$$

Cholesky 分解

$$A = LL^T$$

与 Doolittle 分解相似，不作过多说明。

Thomas 方法

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ l_2 & 1 & & & \\ & l_3 & 1 & & \\ & & \ddots & \ddots & \\ & & & l_n & 1 \end{bmatrix} \begin{bmatrix} u_1 & c_1 & & & \\ & u_2 & c_2 & & \\ & & \ddots & \ddots & \\ & & & u_{n-1} & c_{n-1} \\ & & & & u_n \end{bmatrix}$$

$$\begin{cases} l_i = \dfrac{a_i}{u_{i-1}} \\ u_1 = b_i - l_i c_{i-1} \end{cases}$$

Jacobi 迭代法

$$A\vec{x} = \vec{b}$$

$$\text{记} A = D + L + U$$

$$\vec{x}^{k+1} = -D^{-1}(L+U)\vec{x}^k + D^{-1}\vec{b}$$

收敛的充分不必要条件：原矩阵 A 按行按列严格对角占优

Jacobi-Gauss-Seidel 迭代法

$$\vec{x}^{k+1} = -(D+L)^{-1}U\vec{x}^k + (D+L)^{-1}\vec{b}$$

Successive Over Relaxation

$$x_i^{k+1} = (1-w)x_i^k + \frac{w}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=k+1}^{n} a_{ij}x_k^k)$$

当 0< w <1 时，称为低松弛方法，当 w > 1时，称为超松弛方法。适当选取松弛因子 w 的值，可以得到比 G-S 方法更快的收敛格式。

SOR 方法收敛的必要条件是0 < ω <2。

若系数矩阵 A 对称正定，且0 < ω <2，则 SOR 方法收敛

若系数矩阵 A 严格对角占优，且 $0 < \omega \leq 1$，则 SOR 方法收敛

最佳松弛因子：

$$\omega_{opt} = \frac{2}{1+\sqrt{1-(\rho(J))^2}}$$

第四章 函数插值

Lagrange 插值

$$L_n(x) = \sum_{k=0}^{n} l_k(x) f(x_k)$$

$$l_k(x) = \frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$$

$$\text{插值余项：} R_n(x) = f(x) - L_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

$$\text{其中} M_{n+1} = max|f^{(n+1)}(x)|, w_{n+1}(x) = \prod_{i=0}^{n}(x-x_i)$$

当$n=2$时，有

$$L_2(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}y_0 + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}y_1 + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}y_2$$

Newton 插值

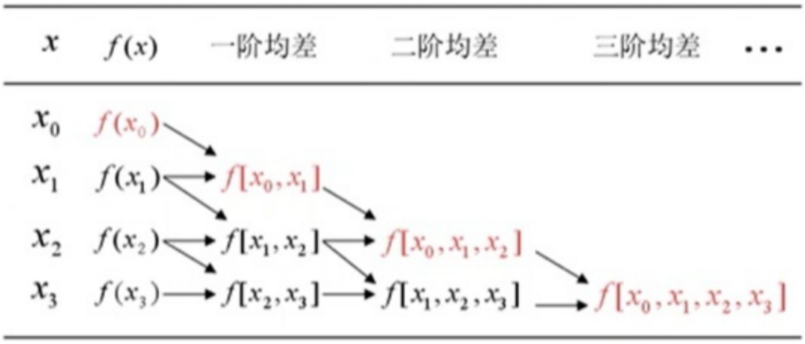
一阶差商

$$f[x_i, x_j] = \frac{f(x_i) - f(x_j)}{x_i - x_j}$$

二阶差商

$$f[x_i, x_j, x_k] = \frac{f[x_i, x_j] - f[x_j - x_k]}{x_i - x_k}$$

即 f(X) *k*-1 阶差商的差商称为*k*阶差商，若为重节点，则此处差商为其导数



导数和差商的关系：差商与插值节点的排列次序无关

$$f[x_0,x_1,\cdots,x_k]=\frac{f^{(k)}(\xi)}{k!}$$

差分与差商的关系

$$f[x_0,x_1,\cdots,x_n]=\frac{\Delta^n y_0}{n!h^n}$$

Newton 型插值公式

$$L_2(x)=f(x_0)+f[x_0,x_1]w_1(x)+f[x_0,x_1,x_2]w_2(x)$$
$$L_n(x)=L_{n-1}(x)+f[x_0,x_1,\cdots,x_n]w_n(x)$$

Newton 向前插值公式（主对角线）

$$N_n(a+th)=f(x_0)+\sum_{k=1}^{n}\frac{\Delta^k f(x_0)}{k!}t(t-1)\cdots(t-k+1)$$

Newton 向后插值公式（最下面一行）

$$N_n(a-th)=f(x_n)+\sum_{k=1}^{n}\frac{\nabla^k f(x_0)}{k!}t(t+1)\cdots(t+k-1)$$

分段插值

两点插值：在每两个点构成的小区间上使用拉格朗日插值

$$|R_n(x)|\le\frac{h^2}{8}M_2$$

三点插值（二次插值）

$$|R_n(x)|\le\frac{\sqrt{3}h^3}{27}M_3$$

三次样条插值

显然我也不想看，我是懒狗

第五章 曲线拟合的最小二乘法

矛盾方程组的求解过程：始终记住左乘 A^T 即可

$$Ax=b\to A^TAx=A^Tb\to Cx=d$$

第六章 数值微分与数值积分

Taylor展开法

$$f'(x_k)=\frac{f(x_k+h)-f(x_k-h)}{2h}-\frac{h^2}{6}f'''(\xi_3)$$

$$f''(x_k)=\frac{f(x_k+h)-2f(x_k)+f(x_k+h)}{h^2}-\frac{h^2}{12}f^{(4)}(\xi_4)$$

一阶三点公式

$$f'(x_0)=\frac{1}{2h}(-3f(x_0)+4f(x_1)-f(x_2))+\frac{h^2}{3}f'''(\xi_1)$$

$$f'(x_1)=\frac{1}{2h}(-f(x_0)+f(x_2))-\frac{h^2}{6}f'''(\xi_2)$$

$$f'(x_2)=\frac{1}{2h}(f(x_0)-4f(x_1)+3f(x_2))+\frac{h^2}{3}f'''(\xi_3)$$

Richardson 外推法

$$S-S^*(h) \tag{1}$$

$$S-S^*(\frac{h}{2}) \tag{2}$$

将上述两式进行加权平均，消去误差级数中的第一项，可得到精度更高的数值计算公式

Newton-Cotes 公式

$$\int_a^b f(x)dx\approx(b-a)\sum_{k=0}^{n}c_k^{(n)}f(a+kh)$$

$$\text{截断误差}E_n[f]=\frac{1}{(n+1)!}\int_a^b f^{(n+1)}(\xi)w_{n+1}(x)dx$$

中点求积公式

$$M(f)=\int_a^b f(x)dx\approx(b-a)f(\frac{a+b}{2})$$

$$E_M(f)=\frac{(b-a)^3}{24}f''(\xi)$$

梯形求积公式

$$T(f)=\int_a^b f(x)dx\approx\frac{b-a}{2}[f(a)+f(b)]$$

$$E_T(f)=-\frac{(b-a)^3}{12}f''(\xi)$$

Simpson 求积公式

$$S(f)=\int_a^b f(x)dx\approx\frac{b-a}{6}[f(a)+4f(\frac{b-a}{2})+f(b)]$$

$$E_S(f)=-\frac{(b-a)^5}{2880}f''''(\xi)$$

Cotes 求积公式

$$C(f)=\int_a^b f(x)dx\approx\frac{b-a}{90}[7f(x_1)+32f(x_2)+12f(x_3)+32f(x_4)+7f(x_5)]$$

$$E_C(f)=-\frac{(b-a)^7}{1935360}f^{(6)}(\xi)$$

Cotes 系数每一行和都为1且对称，当 $n\geq 8$ 时系数存在负值

复化梯形公式

$$T_n=\frac{h}{2}[f(a)+2\sum_{k=1}^{n-1}f(x_k)+f(b)]$$

$$1、2、2、2、2、2、2、2、1$$

$$E_{T_n}=-\frac{b-a}{12}h^2f''(\xi)$$

复化 Simpson 公式

$$S_n=\sum_{k=0}^{n-1}\frac{h}{6}[f(x_k)+4f(x_{k+\frac{1}{2}})+f(x_{k+1})]$$

$$1、4、2、4、2、4、2、4、1$$

$$E_{S_n}=-\frac{b-a}{2880}h^4f^{(4)}(\xi)$$

代数精度

具有n+1个节点的求积公式，其代数精确度最高为 2n+1

梯形求积代数精度为 1，Simpson 公式求积代数精度为 3，Cotes 求积公式代数精度为 5。（复化不变）

Romberg 求积法

$$T_1=\frac{b-a}{2}[f(a)+f(b)]$$

$$T_{2n}=\frac{1}{2}T_n+\frac{b-a}{2n}\sum_{j=1}^{n}f(a+(2j-1)\frac{b-a}{2n}) \tag{1}$$

$$S_n=\frac{4}{4-1}T_{2n}-\frac{1}{4-1}T_n \tag{2}$$

$$C_n=\frac{4^2}{4^2-1}S_{2n}-\frac{1}{4^2-1}S_n \tag{3}$$

$$R_n = \frac{4^3}{4^3-1}C_{2n} - \frac{1}{4^3-1}C_n \tag{4}$$

区间等分数 $n=2^k$	T_{2^k}	$S_{2^{k-1}}$	$C_{2^{k-2}}$	$R_{2^{k-3}}$
1	T_1			
2	T_2	S_1		
4	T_4	S_2	C_1	
8	T_8	S_4	C_2	R_1
16	T_{16}	S_8	C_4	R_2
32	T_{32}	S_{16}	C_8	R_4
⋮	⋮	⋮	⋮	⋮

~~Gauss 型求积公式~~

经过以下特定节点的求积公式具有最高的代数精确度：（考试不作要求）

$$\int_{-1}^{1} f(x)dx \approx 2f(0)$$

$$\int_{-1}^{1} f(x)dx \approx f(-\frac{\sqrt{3}}{3}) + f(\frac{\sqrt{3}}{3})$$

$$\int_{-1}^{1} f(x)dx \approx \frac{5}{9}f(-\sqrt{0.6}) + \frac{8}{9}f(0) + \frac{5}{9}f(\sqrt{0.6})$$

称以上公式为 Gauss-Legendre 求积公式，其代数精度为1、3、5。

第七章 常微分方程初值问题的数值解法

Lipschitz 条件

若存在常数 L，对任意的 y_1、y_2

$$|f(x,y_1) - f(x,y_2)| = |\frac{\partial f(x,\xi)}{\partial y}(y_1 - y_2)| \le L|y_1 - y_2|$$

则上面的初值问题存在唯一解，且解是连续可微的。

显式 Euler 公式

$$y_{n+1} = y_n + hf(x_n, y_n)$$
$$R_{n+1} = \frac{h^2}{2}y''(x_n) + o(h^3)$$

隐式 Euler 公式

$$y_{n+1} = y_n + hf(x_n, y_{n+1})$$

$$R_{n+1} = -\frac{h^2}{2} y''(x_n) + o(h^3)$$

$$hL < 1$$

梯形公式

$$y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}，\ y_{n+1})]$$

$$R_{n+1} = -\frac{h^3}{12} y'''(x_n) + o(h^4)$$

$$\frac{hL}{2} < 1$$

Euler-梯形预估校正公式

$$\begin{cases} y_{n+1}^{[0]} = y_n + hf(x_n, y_n) & \text{预估算式} \\ y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}，\ y_{n+1}^{[0]})] & \text{校正算式} \end{cases}$$

Taylor 级数展开法

二阶 Taylor 方法:

$$y_{n+1} = y_n + hy_n' + \frac{h^2}{2} y_n'' + o(h^3)$$

Runge—Kutta 方法

$$\begin{cases} y_{n+1} = y_n + h\phi(x_n, y_n, h) \\ \phi(x_n, y_n, h) = \sum_{i=1}^{r} c_i K_i \\ k_1 = f(x_n, y_n) \\ k_i = f\left(x_n + \alpha_i h, y_n + h\sum_{j=1}^{i-1} \beta_{ij} K_j\right), i = 2, 3, \cdots, r \end{cases}$$

二阶 Heun（休恩）方法

$$y_{n+1} = y_n + \frac{h}{4}(K_1 + 3K_2)$$

$$K_1 = f(x_n, y_n)$$

$$K_2 = f(x_n + \frac{2}{3}h, y_n + \frac{2}{3}hK_1)$$

三阶 Heun（休恩）公式

$$y_{n+1} = y_n + \frac{h}{4}(K_1 + 3K_3)$$
$$K_1 = f(x_n, y_n)$$
$$K_2 = f(x_n + \frac{1}{3}h, y_n + \frac{1}{3}hK_1)$$
$$K_3 = f(x_n + \frac{2}{3}h, y_n + \frac{2}{3}hK_2)$$

三阶 Kutta 公式

$$y_{n+1} = y_n + \frac{h}{6}(K_1 + 4K_2 + K_3)$$
$$K_1 = f(x_n, y_n)$$
$$K_2 = f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hK_1)$$
$$K_3 = f(x_n + h, y_n - hK_1 + 2hK_2)$$

标准四阶 Runge-Kutta 公式

$$y_{n+1} = y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4)$$
$$K_1 = f(x_n, y_n)$$
$$K_2 = f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hK_1)$$
$$K_3 = f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hK_2)$$
$$K_4 = f(x_n + h, y_n + hK_3)$$

基尔（Gill）公式

$$y_{n+1} = y_n + \frac{h}{6}[K_1 + (2-\sqrt{2})K_2 + (2+\sqrt{2})K_3 + K_4)$$
$$K_1 = f(x_n, y_n)$$
$$K_2 = f(x_n + \frac{h}{2}, y_n + \frac{1}{2}hK_1)$$
$$K_3 = f(x_n + \frac{h}{2}, y_n + \frac{\sqrt{2}-1}{2}hK_1 + \frac{2-\sqrt{2}}{2}hK_2)$$
$$K_4 = f(x_n + h, y_n - \frac{\sqrt{2}}{2}hK_2 + (1+\frac{\sqrt{2}}{2})hK_3)$$

43

稳定性

Euler 公式满足 $|1+\lambda h| \leq 1$ 时，方法稳定；

梯形公式对于任意步长的 h 都稳定；

Euler-梯形预估校正公式满足 $|1+\lambda h+\frac{1}{2}(\lambda h^2)| \leq 1$ 时，方法稳定。

四阶经典R - K方法的绝对稳定区域为

$$\left|1+h\lambda+\frac{1}{2}(h\lambda)^2+\frac{1}{6}(h\lambda)^3+\frac{1}{24}(h\lambda)^4\right|<1$$

线性四步 Adams 显式公式

$$y_{n+1}=y_n+\frac{h}{24}[55f(x_n,y_n)-59f(x_{n-1},y_{n-1})+37f(x_{n-2},y_{n-2})-9f(x_{n-3},y_{n-3})]$$

$$R_n=\frac{251}{720}h^5y^{(5)}(\xi)$$

三步隐式方法 Adams 内插公式

$$y_{n+1}=y_n+\frac{h}{24}[9f(x_{n+1},y_{n+1})+19f(x_n,y_n)-5f(x_{n-1},y_{n-1})+f(x_{n-2},y_{n-2})]$$

$$R_n=-\frac{19}{720}h^5y^{(5)}(\xi)$$

Taylor 展开法构造线性多步公式

$$y_{n+1}=y_{n-1}+\frac{h}{3}(f_{n+1}+4f_n+f_{n-1})$$

$$R_n=-\frac{1}{90}h^5y^{(5)}(x_n)+o(h^6)$$

第八章 矩阵特征值和特征向量的计算

乘幂法

$$V^{(k)}=AV^{(k-1)}$$

$$\lambda_i=\frac{V_i^{(k+1)}}{V_i^{(k)}}$$

为避免溢出，可每五次进行一次规范化。

$$\tilde{V}=\frac{V^{(k)}}{\max V^{(k)}}$$

原点平移法

$$B=A-pE$$

反幂法

$$V^{(k+1)} = A^{-1}V^{(k)}$$

$$AV^{(k+1)} = V^{(k)}$$

$$A = LU$$

$$\begin{cases} Ly = V^{(k)} \\ UV^{(k+1)} = y \end{cases}$$

~~Jacobi 方法~~

哈哈哈哈哈哈哈哈哈哈 不考！