

数值分析

南京邮电大学数理学院杨振华制作
njuptshumo2006@126.com

目 录

- 第一章 绪论
- 第二章 插值法
- 第三章 函数逼近与计算
- 第四章 数值积分与数值微分
- 第五章 常微分方程数值解法
- 第六章 方程求根
- 第七章 解线性方程组的直接方法
- 第八章 解线性方程组的迭代法
- 第九章 矩阵的特征值与特征向量计算

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

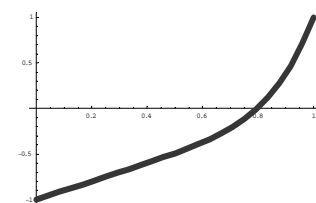
2

第一章 绪 论

简单示例

例A 求方程 $x^7 + x - 1 = 0$
在区间 $[0,1]$ 中的根.

分析: 由于函数
 $f(x) = x^7 + x - 1$ 是连续函数,
且 $f(0) < 0, f(1) > 0$,
根据介值定理, 它在
 $(0,1)$ 内必然有根.



困难: 五次以上方程没有求根公式, 无法求出精确解.

只能用数值方法求近似解!

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

4

简单示例

例B 已知施加于弹簧上的外力与其的伸长的长度成正比, 即有 $F = kx$, 其中 k 称为弹簧的弹性系数. 现已测得伸长长度和外力的若干数据如下:

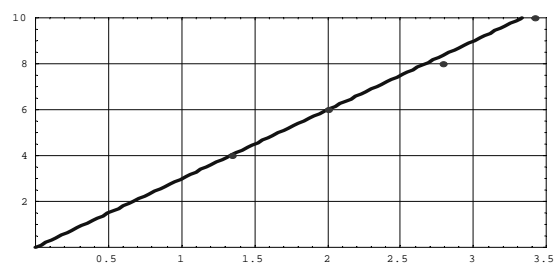
$(1.35, 4.00), (2.00, 6.00), (2.69, 8.00), (3.42, 10.00)$

请根据这组数据求出弹性系数 k .

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

5

分析: 我们先将数据画在一张图上. 可以看出, 四个点近似的在一条通过原点的直线上. 但是如果我们作出直线 $y = 3x$, 它通过第二个点, 却不通过其它的点. “最好”的系数应该是多少?



6

数值分析

上面的例B必须通过数值分析的方法给出一定的理论基础.

事实上,数值分析就是研究用计算机解决实际问题的数值方法和理论.

数值分析一般以需计算的数学问题为研究对象,也有自身严谨的理论系统.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

7

误差来源

模型误差:建立数学模型的近似带来的误差;

观测误差:测量物理量所包含的误差;

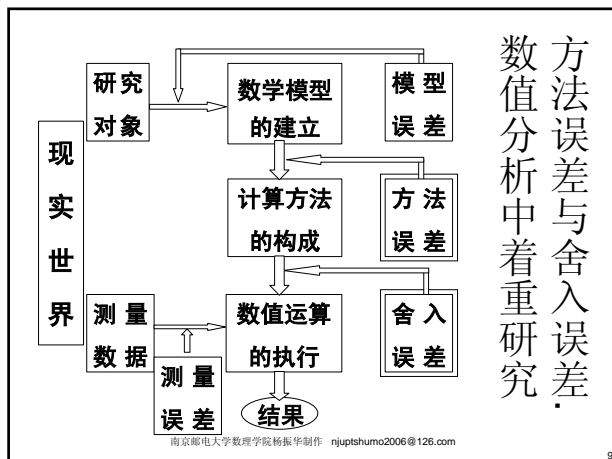
截断误差:也称为方法误差用数值方法求近似解时,近似解与精确解之间的误差;

舍入误差:计算机字长有限,在表示无限小数时产生的误差.

我们可以用下面的图来显示各种误差.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

8



9

误差分析

例1 计算 $I_n = e^{-1} \int_0^1 x^n e^x dx (n=0,1,\dots)$ 并估计误差

由分部积分公式可以得到 I_n 的递推公式

$$I_n = 1 - nI_{n-1} (n=1,2,\dots), I_0 = 1 - e^{-1};$$

下面采用4位小数计算.由于 $e^{-1} \approx 0.3679$ (根据 Taylor 展式,误差小于 0.25×10^{-4}), $I_0 \approx 0.6321$.

我们用此初值以及递推公式计算若干步得到表中的结果.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

10

误差分析

n	\bar{I}_n	n	\bar{I}_n
0	0.6321	5	0.1480
1	0.3679	6	0.1120
2	0.2642	7	0.2160
3	0.2074	8	-0.7280
4	0.1704	9	7.5520

由于被积函数的值在0,1之间,积分值也应该在0,1之间. \bar{I}_7 和 \bar{I}_8 的值是不可靠的.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

11

例1的误差分析

初值 I_0 有误差 $E_0 = I_0 - \bar{I}_0$, 引起以后各步的误差.

$$\begin{aligned} E_n &= I_n - \bar{I}_n \\ &= (1 - nI_{n-1}) - (1 - n\bar{I}_{n-1}) \\ &= -n(I_{n-1} - \bar{I}_{n-1}) \\ &= -nE_{n-1} \end{aligned}$$

因此有 $E_n = (-1)^n n! E_0$.

这说明第 n 步的误差是初始误差的 $n!$ 倍.

下面的表给出了实际的误差.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

12

例1的误差分析

n	\bar{I}_n	E_n	n	\bar{I}_n	E_n
0	0.6321	$2.056 \cdot 10^{-5}$	5	0.1480	$-2.467 \cdot 10^{-3}$
1	0.3679	$-2.056 \cdot 10^{-5}$	6	0.1120	$1.480 \cdot 10^{-2}$
2	0.2642	$4.112 \cdot 10^{-5}$	7	0.2160	$-1.036 \cdot 10^{-1}$
3	0.2074	$-1.234 \cdot 10^{-4}$	8	-0.7280	0.8289
4	0.1704	$4.934 \cdot 10^{-4}$	9	7.5520	-7.460

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

13

稳定性的算法

通过误差分析,我们可以发现上面的算法中,随着计算的深入,误差是逐渐扩大的,我们称之为不稳定的算法.

反之,误差不扩大的算法称为稳定的算法.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

14

稳定性的算法

由上面的递推公式可以得到: $I_{n-1}=(1-I_n)/n$.
我们已知 I_0 在0和1之间,可以取为0.5(误差小于0.5).用上面的递推公式计算得 $I_0=0.6321$,
计算结果是相当准确的.
类似于前面的方法可以得到 $|E_0|=|E_n|/n!$,
该算法是稳定的.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

15

误差分析的重要性

由上面的算法的例子可以发现,在数值计算中如果不注意误差分析,许多在理论上好像是正确的算法,在实际应用时会出现“差之毫厘,失之千里”的错误结果.

在计算时,应尽量采用稳定性好的算法.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

16

误差的基本概念

绝对误差: $e^* = x^* - x$

e^* :误差 x^* :近似值 x :准确值

注:在实际应用中,也可以将 $x - x^*$ 作为误差,没有实质性的区别.

误差限 ε^* : (绝对)误差绝对值的上限
用毫米刻度的米尺测量时,其误差限为0.5毫米

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

17

误差的基本概念

相对误差 $e_r^* = e^*/x = (x^* - x)/x$

e_r^* :相对误差, e^* :绝对误差, x^* :近似值, x :准确值
注:常用 $(x^* - x)/x^*$ 来作为 x^* 的相对误差

相对误差限 ε_r^* :相对误差的绝对值上界
用毫米刻度的米尺测量得一物体的长度为128毫米,则其相对误差限可计算为 $0.5/128 = 0.39\%$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

18

有效数字

$$x=\pi=3.14159265\dots$$

按照四舍五入的原则可以得到 x 的前几位.

取3位 $x_3^*=3.14$, $\varepsilon_3^*\leq 0.002$,

取5位 $x_5^*=3.1416$, $\varepsilon_5^*\leq 0.000008$,

一般地,误差小于末位数字的半个单位

$$|\pi-3.14|\leq 0.5\times 10^{-2}, |\pi-3.1416|\leq 0.5\times 10^{-4}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

19

有效数字

若近似值 x^* 的误差限是某一位的半个单位,该位到 x^* 的第一位非零数字共有 n 位,就称 x^* 有 n 位有效数字.

x^* 有 n 位有效数字可写成标准形式

$$x^*=\pm 10^m\times (a_1+a_2\times 10^{-1}+\dots+a_n\times 10^{-(n-1)}),$$

其误差限为 $0.5\times 10^{m-n-1}$,即

$$|x-x^*|\leq 0.5\times 10^{m-n-1}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

20

数值运算的误差估计(四则运算)

两个近似数 x^* 与 y^* ,其误差限分别为 $\varepsilon(x^*)$ 与 $\varepsilon(y^*)$,它们进行加,减,乘,除运算得到的误差限分别为

$$\varepsilon(x^*\pm y^*)=\varepsilon(x^*)+\varepsilon(y^*) \text{ (书中有错!)}$$

$$\varepsilon(x^*\times y^*)=\varepsilon(y^*)|x^*|+\varepsilon(x^*)|y^*|$$

$$\varepsilon(x^*/y^*)=(\varepsilon(y^*)|x^*|+\varepsilon(x^*)|y^*|)/|y^*|^2$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

21

数值运算的误差估计(函数计算)

对于多元函数 $A=f(x_1,x_2,\dots,x_n)$,如果 x_1,x_2,\dots,x_n 的近似值分别为 x_1^*,x_2^*,\dots,x_n^* ,误差限分别为 $\varepsilon(x_k^*)$ ($k=1,2,\dots,n$),则 A 的近似值为 $A^*=f(x_1^*,x_2^*,\dots,x_n^*)$,其误差限可以用下式计算

$$\varepsilon(A^*)\approx \sum_{k=1}^n \left| \left(\frac{\partial f}{\partial x_k} \right)^* \right| \varepsilon(x_k^*)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

22

例5 已测得某场地长 l 的近似值为 $l^*=110\text{m}$,宽 d 的近似值为 $d^*=80\text{m}$,已知 $|l-l^*|\leq 0.2\text{m}$, $|d-d^*|\leq 0.1\text{m}$.试求面积 $s=ld$ 的绝对误差限与相对误差限.

解: $s=ld$, $\partial s/\partial l=d$, $\partial s/\partial d=l$,因此绝对误差限约为

$$\varepsilon(s^*)\approx |(\partial s/\partial l)^*|\varepsilon(l^*)+|(\partial s/\partial d)^*|\varepsilon(d^*)$$

$$=|d^*|\varepsilon(l^*)+|l^*|\varepsilon(d^*)$$

$$=80\times 0.2+110\times 0.1=27(\text{m}^2)$$

$$\text{相对误差限 } \varepsilon_r(s^*)=\varepsilon(s^*)/|s^*|\approx 27/8800=0.31\%.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

23

数值运算的若干原则

避免“大数”除以“小数”

避免相近数相减

避免“大数”吃“小数”

减少运算次数

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

24

计算 $1.23+0.0567+0.0957+0.0246+0.0753$
(用三位有效数字)

直接计算 $=1.29+0.0957+0.0246+0.0753$
 $=1.39+0.0246+0.0753$
 $=1.42+0.0753=1.50$

先计算后四项

$=1.23+(0.0567+0.0957+0.0246+0.0753)$
 $=1.23+0.252=1.48$

后者更精确(用5位数字得结果为1.4823)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

25

数值运算的若干原则

计算 $\sqrt{3}-\sqrt{2}$ (用两位有效数字)

直接计算: $\sqrt{3}-\sqrt{2} \approx 1.7-1.4=0.3$

转换以后计算:

$$\sqrt{3}-\sqrt{2} = \frac{1}{\sqrt{3}+\sqrt{2}} \approx \frac{1}{1.7+1.4} = \frac{1}{3.1} \approx 0.32$$

后一种方法更为精确

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

26

第二章 插值法

引 言

实际问题中碰到的函数是各种各样的.有的表达很复杂,有的甚至给不出数学式子,而只是给出了一些离散数据——譬如某些点的函数值和导数值.面对这种情况,一个很自然的想法就是构造某个简单的函数作为要考察的函数的近似.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

28

引 言

如果要求近似函数取给定的离散数据,则称之为插值函数.实用上,我们常取结构相对比较简单代数多项式作为插值函数,这就是所谓的代数插值.

本章先讨论代数插值,然后在此基础上进一步研究所谓的样条插值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

29

引例

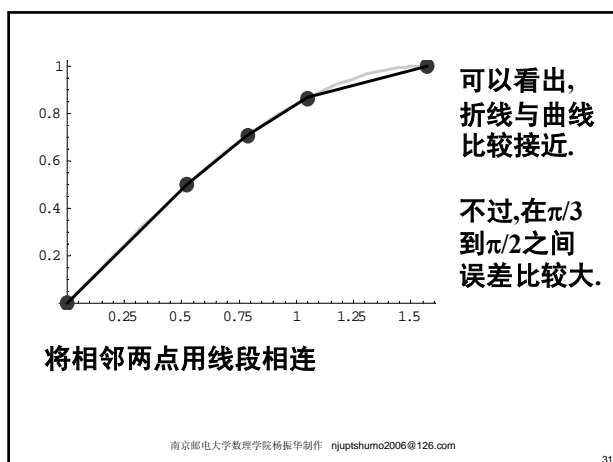
三角函数表的构造

如何用初等方法给出 $\sin x$ 在一系列点处的函数值?

已知 $\sin x$ 在 $x=0, \pi/6, \pi/4, \pi/3, \pi/2$ 等处的函数值.将这些点处的函数值在图形上标出.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

30



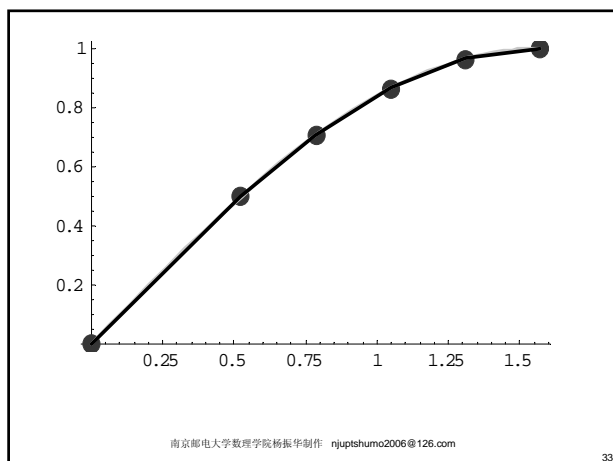
再增加一个点 $x=(\pi/3+\pi/2)/2$,这一点的函数值可以通过初等方法求出.

此时,在 $\pi/3$ 到 $\pi/2$ 之间的误差变小(见下页图).

通过这一方法可以用一条折线替代函数 $y=\sin x$,而且随着点的增加,误差越来越小.

这个例子说明,我们可以用简单的函数去替代复杂的函数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com



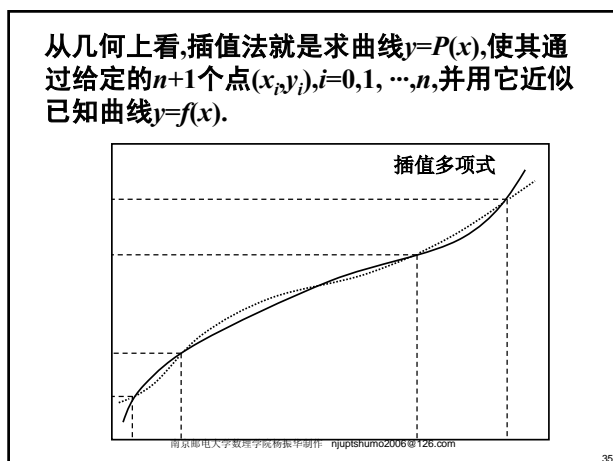
引言

设函数 $f(x)$ 在区间 $[a,b]$ 上有定义,且已知在点 $a \leq x_0 < x_1 < \dots < x_n \leq b$ 上的函数值 $f(x_i)=y_i (i=0,1,\dots,n)$

如果存在简单函数 $P(x)$,使 $P(x_i)=y_i (i=0,1,\dots,n)$ 成立,就称 $P(x)$ 为 $f(x)$ 的插值函数,点 x_0, x_1, \dots, x_n (它们互不相同)称为插值节点,包含插值节点的区间 $[a,b]$ 称为插值区间.

如果 $P(x)$ 是不超过 n 次的多项式,就称 $P(x)$ 为插值多项式.若 $P(x)$ 为分段的函数,就是分段插值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com



插值多项式的存在唯一性

用 H_n 代表所有次数不超过 n 的多项式的集合.所谓插值多项式存在唯一,是指集合 H_n 中有且只有一个 $P(x)$ 满足插值条件.若设

$$P(x) = a_0 + a_1x + \dots + a_nx^n$$

根据插值条件 $P(x_i)=y_i, (i=0,1,\dots,n)$,有

$$\begin{cases} a_0 + a_1x_0 + \dots + a_nx_0^n = y_0 \\ a_0 + a_1x_1 + \dots + a_nx_1^n = y_1 \\ \dots \\ a_0 + a_1x_n + \dots + a_nx_n^n = y_n \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

插值多项式的存在唯一性

上面的方程组是关于 a_0, a_1, \dots, a_n 的线性方程组, 其系数行列式为

$$V_n(x_0, x_1, \dots, x_n) = \begin{vmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{vmatrix} = \prod_{i=1}^n \prod_{j=0}^{i-1} (x_i - x_j) \neq 0$$

所以方程组存在唯一的解.
从而插值多项式存在唯一.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

37

唯一性的另一种证明

设 $f(x), g(x)$ 都是不超过 n 次的多项式, 而且满足 $f(x_i) = g(x_i) = y_i, (i=0, 1, \dots, n)$,

若 $h(x) = f(x) - g(x)$ 不是零多项式, 则其必然是不超过 n 次的多项式. 由于 $h(x_i) = 0 (i=0, 1, \dots, n)$, 因此 $(x - x_i)$ 是 $h(x)$ 的因式,

从而 $h(x)$ 有因式 $(x - x_0)(x - x_1) \cdots (x - x_n)$, 其次数必然大于等于 $n+1$, 矛盾.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

38

线性插值

问题 求作一次式 $L_1(x)$, 满足条件

$$L_1(x_k) = y_k, L_1(x_{k+1}) = y_{k+1},$$

从几何图形上看, $y = L_1(x)$ 表示过两点 $(x_k, y_k), (x_{k+1}, y_{k+1})$ 的直线, 因此可表为如下对称形式:

$$L_1(x) = y_k l_k(x) + y_{k+1} l_{k+1}(x)$$

$$\text{其中 } l_k(x) = \frac{x - x_{k+1}}{x_k - x_{k+1}}, l_{k+1}(x) = \frac{x - x_k}{x_{k+1} - x_k}$$

$l_k(x)$ 与 $l_{k+1}(x)$ 分别满足条件

$$l_k(x_k) = 1, l_k(x_{k+1}) = 0 \text{ 以及 } l_{k+1}(x_k) = 0, l_{k+1}(x_{k+1}) = 1.$$

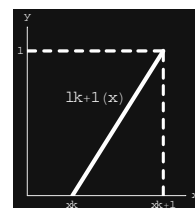
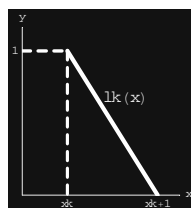
南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

39

线性插值

可见, 插值问题的解 $L_1(x)$ 可以通过线性插值基函数 $l_k(x)$ 和 $l_{k+1}(x)$ 的组合得出, 且组合系数分别是所给数据 y_k, y_{k+1} .

下面是两个基函数的图形.



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

40

抛物插值

问题 求作二次函数 $L_2(x)$, 使满足条件

$$L_2(x_{k-1}) = y_{k-1}, L_2(x_k) = y_k, L_2(x_{k+1}) = y_{k+1}$$

二次插值的几何解释是用通过三个点

$$(x_{k-1}, y_{k-1}), (x_k, y_k), (x_{k+1}, y_{k+1})$$

的抛物线 $y = L_2(x)$ 来近似曲线.

类似于线性插值, 令 $L_2(x) = y_{k-1} l_{k-1}(x) + y_k l_k(x) + y_{k+1} l_{k+1}(x)$

我们希望 $l_{k-1}(x_{k-1}) = 1, l_{k-1}(x_k) = l_{k-1}(x_{k+1}) = 0$

$$l_k(x_k) = 1, l_k(x_{k-1}) = l_k(x_{k+1}) = 0$$

$$l_{k+1}(x_{k+1}) = 1, l_{k+1}(x_{k-1}) = l_{k+1}(x_k) = 0$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

41

抛物插值

$$l_{k-1}(x_{k-1}) = 1, l_{k-1}(x_k) = l_{k-1}(x_{k+1}) = 0$$

$$l_k(x_k) = 1, l_k(x_{k-1}) = l_k(x_{k+1}) = 0$$

$$l_{k+1}(x_{k+1}) = 1, l_{k+1}(x_{k-1}) = l_{k+1}(x_k) = 0$$

此时, 将 x_{k-1} 带入到 $P_2(x)$ 的表达式中, 显然有

$$P_2(x_{k-1}) = y_{k-1} l_{k-1}(x_{k-1}) + y_k l_k(x_{k-1}) + y_{k+1} l_{k+1}(x_{k-1}) = y_{k-1}.$$

类似地有 $P_2(x_k) = y_k, P_2(x_{k+1}) = y_{k+1}$, 即 $P_2(x)$ 满足插值条件.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

42

抛物插值

$$l_{k-1}(x_{k-1})=1, l_{k-1}(x_k)=l_{k-1}(x_{k+1})=0$$

$$l_k(x_k)=1, l_k(x_{k-1})=l_k(x_{k+1})=0$$

$$l_{k+1}(x_{k+1})=1, l_{k+1}(x_{k-1})=l_{k+1}(x_k)=0$$

根据 $l_{k-1}(x_k)=l_{k-1}(x_{k+1})=0$ 可知 x_k 和 x_{k+1} 是 $l_{k-1}(x)$ 的两个根,可以设 $l_{k-1}(x)=A(x-x_k)(x-x_{k+1})$,再根据 $l_{k-1}(x_{k-1})=1$,有 $1=A(x_{k-1}-x_k)(x_{k-1}-x_{k+1})$,因此

$$A = \frac{1}{(x_{k-1}-x_k)(x_{k-1}-x_{k+1})} \quad l_{k-1}(x) = \frac{(x-x_k)(x-x_{k+1})}{(x_{k-1}-x_k)(x_{k-1}-x_{k+1})}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

43

抛物插值

类似的
$$l_k(x) = \frac{(x-x_{k-1})(x-x_{k+1})}{(x_k-x_{k-1})(x_k-x_{k+1})}$$

$$l_{k+1}(x) = \frac{(x-x_{k-1})(x-x_k)}{(x_{k+1}-x_{k-1})(x_{k+1}-x_k)}$$

从而插值函数为

$$L_2(x) = y_{k-1} \frac{(x-x_k)(x-x_{k+1})}{(x_{k-1}-x_k)(x_{k-1}-x_{k+1})} + y_k \frac{(x-x_{k-1})(x-x_{k+1})}{(x_k-x_{k-1})(x_k-x_{k+1})} + y_{k+1} \frac{(x-x_{k-1})(x-x_k)}{(x_{k+1}-x_{k-1})(x_{k+1}-x_k)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

44

Lagrange插值多项式

仿照前述作法,对于求作次数不超过 n 的多项式 $L_n(x)$,使满足条件 $L_n(x_j)=y_j(j=0,1,\dots,n)$ 的问题,我们可构造插值基函数 $l_j(x), j=1,2,\dots,n$

它们都是次数小于 n 的多项式,且满足条件

$$l_j(x_k) = \delta_{kj} = \begin{cases} 0, & k \neq j \\ 1, & k = j \end{cases}$$

则有如下Lagrange插值公式:

$$L_n(x) = \sum_{k=0}^n y_k l_k(x) = \sum_{k=0}^n \prod_{\substack{j=0 \\ j \neq k}}^n \frac{x-x_j}{x_k-x_j} y_k$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

45

基函数的性质

性质1
$$l_j(x_k) = \delta_{kj} = \begin{cases} 0, & k \neq j \\ 1, & k = j \end{cases}$$

性质2
$$\sum_{j=0}^n l_j(x) \equiv 1$$

这是由于通过 $(x_0,1), (x_1,1), \dots, (x_n,1)$ 的不超过 n 次的多项式是唯一的。

而多项式 $P(x)=1$ 以及 $\sum_{j=0}^n l_j(x)$ 都是通过上述的点的不超过 n 次的多项式,从而性质2成立

性质3
$$\sum_{j=0}^n x_j^k l_j(x) = x^k \quad (k=0,1,\dots,n)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

46

Lagrange余项定理

若在 $[a,b]$ 上用 $L_n(x)$ 近似 $f(x)$,则其截断误差为 $R_n(x)=f(x)-L_n(x)$,也称为插值余项。

定理 Lagrange余项定理设 $f^{(n)}(x)$ 在 $[a,b]$ 连续, $f^{(n+1)}(x)$ 在 (a,b) 内存在,节点 $a \leq x_0 < x_1 < \dots < x_n \leq b$, $L_n(x)$ 是满足插值条件的插值多项式,则对 $[a,b]$ 中任意的 x ,存在 (a,b) 内的 ξ ,使得

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

其中
$$\omega_{n+1}(x) = (x-x_0)(x-x_1)\cdots(x-x_n)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

47

证明

当 x 是某个节点时,结论显然成立,若 x 不是节点

$$\text{设 } \varphi(t) = f(t) - L_n(t) - (f(x) - L_n(x)) \frac{(t-x_0)(t-x_1)\cdots(t-x_n)}{(x-x_0)(x-x_1)\cdots(x-x_n)}$$

显然 $\varphi(t)$ 有 $n+2$ 个零点: x_0, x_1, \dots, x_n, x .根据Rolle定理, $\varphi(t)$ 的任意两个零点之间有 $\varphi'(t)$ 的一个零点.因此, $\varphi'(t)$ 至少有 $n+1$ 个零点.

再根据Rolle定理, $\varphi'(t)$ 的任意两个零点之间有 $\varphi''(t)$ 的一个零点,因此, $\varphi''(t)$ 至少有 n 个零点.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

48

证 明

依此类推, $\phi^{(n+1)}(t)$ 在 (a, b) 内至少有一个零点, 记为 $\xi \in (a, b)$, 则有

$$0 = \phi^{(n+1)}(\xi) = f^{(n+1)}(\xi) - (f(x) - L_n(x)) \frac{(n+1)!}{(x-x_0)(x-x_1)\cdots(x-x_n)}$$

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

逐次线性插值

设 $f(x)$ 在 x_0, x_1 处的插值函数为 $I_{01}(x)$, 在 x_1, x_2 处的插值函数为 $I_{12}(x)$, 在 x_0, x_1, x_2 处的插值函数为 $I_{012}(x)$.

问题: 由 $I_{01}(x), I_{12}(x)$ 求 $I_{012}(x)$?

考虑函数 $\frac{x-x_2}{x_0-x_2} f(x_0) + \frac{x-x_0}{x_2-x_0} f(x_2)$

该函数在 x_0 处取值为 $f(x_0)$, 在 x_2 处取值为 $f(x_2)$

现将该函数中的 $f(x_0)$ 换为 $I_{01}(x)$, $f(x_2)$ 换为 $I_{12}(x)$

显然变化后的函数在 x_0 处取值为 $f(x_0)$, 在 x_2 处取值为 $f(x_2)$.

逐次线性插值

变化后的函数为 $\frac{x-x_2}{x_0-x_2} I_{01}(x) + \frac{x-x_0}{x_2-x_0} I_{12}(x)$

此函数在 x_1 处的函数值为

$$\begin{aligned} & \frac{x-x_2}{x_0-x_2} I_{01}(x_1) + \frac{x-x_0}{x_2-x_0} I_{12}(x_1) \\ &= \frac{x-x_2}{x_0-x_2} f(x_1) + \frac{x-x_0}{x_2-x_0} f(x_1) = f(x_1) \end{aligned}$$

因此函数 $\frac{x-x_2}{x_0-x_2} I_{01}(x) + \frac{x-x_0}{x_2-x_0} I_{12}(x)$

在 x_0, x_1, x_2 处的函数值为 $f(x_0), f(x_1), f(x_2)$

因此, $I_{012}(x) = \frac{x-x_2}{x_0-x_2} I_{01}(x) + \frac{x-x_0}{x_2-x_0} I_{12}(x)$

逐次线性插值

$$I_{012}(x) = I_{01}(x) + \frac{I_{12}(x) - I_{01}(x)}{x_2 - x_0} (x - x_0)$$

这相当于在 x_0 处的函数值为 $I_{01}(x)$, 在 x_2 处的函数值为 $I_{12}(x)$, 然后进行线性插值, 就得到插值函数 $I_{012}(x)$.

逐次线性插值

如果已知插值函数 $I_{0,1,\dots,k}(x)$ 以及 $I_{0,1,\dots,k-1,l}(x)$,

考虑函数 $P(x) = \frac{x-x_l}{x_k-x_l} I_{0,1,\dots,k}(x) + \frac{x-x_k}{x_l-x_k} I_{0,1,\dots,k-1,l}(x)$

容易证明 $P(x_k) = f(x_k), P(x_l) = f(x_l)$

若 $i=0, 1, \dots, k-1, P(x_i) = f(x_i)$

因此 $P(x) = I_{0,1,\dots,k,l}(x)$

Aitken逐次线性插值公式

$$\begin{aligned} I_{0,1,\dots,k,l}(x) &= \frac{x-x_l}{x_k-x_l} I_{0,1,\dots,k}(x) + \frac{x-x_k}{x_l-x_k} I_{0,1,\dots,k-1,l}(x) \\ &= I_{0,1,\dots,k}(x) + \frac{I_{0,1,\dots,k-1,l}(x) - I_{0,1,\dots,k}(x)}{x_l - x_k} (x - x_k) \end{aligned}$$

上述公式称为Aitken逐次线性插值公式.

Aitken插值公式算例

已知 $f(x)=\sin x$ 的函数值,用Aitken插值方法求 $f(0.23)$.

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134		0.03
0.30	0.30452		-0.07
0.50	0.52110		-0.27
0.60	0.63665		-0.37

南京邮电大学数理学院杨新华制作 njuptshumo2006@126.com

55

$$\begin{aligned}
 I_{0,1} &= I_0 + \frac{I_1 - I_0}{x_1 - x_0}(x - x_0) \\
 &= 0.00000 + \frac{0.20134 - 0.00000}{0.20 - 0.00} 0.23 \\
 &= 0.231541
 \end{aligned}$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452		-0.07
0.50	0.52110		-0.27
0.60	0.63665		-0.37

南京邮电大学数理学院杨新华制作 njuptshumo2006@126.com

56

$$\begin{aligned}
 I_{0,2} &= I_0 + \frac{I_2 - I_0}{x_2 - x_0}(x - x_0) \\
 &= 0.00000 + \frac{0.30452 - 0.00000}{0.30 - 0.00} 0.23 \\
 &= 0.233465
 \end{aligned}$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.233465	-0.07
0.50	0.52110		-0.27
0.60	0.63665		-0.37

南京邮电大学数理学院杨新华制作 njuptshumo2006@126.com

57

$$\begin{aligned}
 I_{0,3} &= I_0 + \frac{I_3 - I_0}{x_3 - x_0}(x - x_0) \\
 &= 0.00000 + \frac{0.52110 - 0.00000}{0.50 - 0.00} 0.23 \\
 &= 0.239706
 \end{aligned}$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.233465	-0.07
0.50	0.52110	0.239706	-0.27
0.60	0.63665		-0.37

南京邮电大学数理学院杨新华制作 njuptshumo2006@126.com

58

$$\begin{aligned}
 I_{0,4} &= I_0 + \frac{I_4 - I_0}{x_4 - x_0}(x - x_0) \\
 &= 0.00000 + \frac{0.63665 - 0.00000}{0.60 - 0.00} 0.23 \\
 &= 0.244049
 \end{aligned}$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.233465	-0.07
0.50	0.52110	0.239706	-0.27
0.60	0.63665	0.244049	-0.37

南京邮电大学数理学院杨新华制作 njuptshumo2006@126.com

59

$$\begin{aligned}
 I_{0,1,2} &= I_{0,1} + \frac{I_{0,2} - I_{0,1}}{x_2 - x_1}(x - x_1) \\
 &= 0.231541 + \frac{0.233465 - 0.231541}{0.30 - 0.20} 0.03 \\
 &= 0.232118
 \end{aligned}$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.233465	-0.07
0.50	0.52110	0.239706	-0.27
0.60	0.63665	0.244049	-0.37

南京邮电大学数理学院杨新华制作 njuptshumo2006@126.com

60

$$I_{0,1,3} = I_{0,1} + \frac{I_{0,3} - I_{0,1}}{x_3 - x_1}(x - x_1)$$

$$= 0.231541 + \frac{0.239706 - 0.231541}{0.50 - 0.20}0.03$$

$$= 0.232358$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.2334650.232118	-0.07
0.50	0.52110	0.2397060.232358	-0.27
0.60	0.63665	0.244049	-0.37

61

$$I_{0,1,4} = I_{0,1} + \frac{I_{0,4} - I_{0,1}}{x_4 - x_1}(x - x_1)$$

$$= 0.231541 + \frac{0.244049 - 0.231541}{0.60 - 0.20}0.03$$

$$= 0.232479$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.2334650.232118	-0.07
0.50	0.52110	0.2397060.232358	-0.27
0.60	0.63665	0.2440490.232479	-0.37

62

$$I_{0,1,2,3} = I_{0,1,2} + \frac{I_{0,1,3} - I_{0,1,2}}{x_3 - x_2}(x - x_2)$$

$$= 0.232118 + \frac{0.232358 - 0.232118}{0.50 - 0.30}(-0.07)$$

$$= 0.232034$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.2334650.232118	-0.07
0.50	0.52110	0.2397060.232350.232034	-0.27
0.60	0.63665	0.2440490.232479	-0.37

63

$$I_{0,1,2,4} = I_{0,1,2} + \frac{I_{0,1,4} - I_{0,1,2}}{x_4 - x_2}(x - x_2)$$

$$= 0.232118 + \frac{0.232479 - 0.232118}{0.60 - 0.30}(-0.07)$$

$$= 0.232034$$

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.2334650.232118	-0.07
0.50	0.52110	0.2397060.232350.232034	-0.27
0.60	0.63665	0.2440490.2324790.232034	-0.37

64

Aitken插值公式算例

由于两个三次插值的结果相同,无需作四次插值,求得 $f(0.23)=0.232034$.

x_i	$f(x_i)$		$x-x_i$
0.00	0.00000		0.23
0.20	0.20134	0.231541	0.03
0.30	0.30452	0.2334650.232118	-0.07
0.50	0.52110	0.2397060.232350.232034	-0.27
0.60	0.63665	0.2440490.2324790.232034	-0.37

65

均差及其性质

Lagrange插值插值公式可以看作直线方程两点式的推广.我们也可以从直线方程的点斜式

$$P_1(x) = f_0 + \frac{f_1 - f_0}{x_1 - x_0}(x - x_0)$$

推广到具有 $n+1$ 个插值点 $(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)$ 的情况.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

66

均差及其性质

假设过 $(x_0, f_0), (x_1, f_1), \dots, (x_{n-1}, f_{n-1})$ 的插值多项式为 $P_{n-1}(x)$, 过 $(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)$ 的插值多项式为 $P_n(x)$, 显然 $P_n(x) - P_{n-1}(x)$ 在 x_0, x_1, \dots, x_{n-1} 的值为零, 因此可以写为

$$P_n(x) - P_{n-1}(x) = a_n(x-x_0)(x-x_1)\cdots(x-x_{n-1}).$$

于是, 我们可以将 $P_n(x)$ 写为

$$P_n(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_1) + \cdots + a_n(x-x_0)(x-x_1)\cdots(x-x_{n-1}).$$

均差及其性质

$$\text{在 } P_n(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_1) + \cdots + a_n(x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

中, 令 $x=x_0$, 可以得到 $P(x_0)=a_0=f_0$;

再令 $x=x_1$ 得 $f_1=f_0+a_1(x_1-x_0)$, 因此

$$a_1 = (f_1 - f_0)/(x_1 - x_0).$$

类似地可以推出

$$a_2 = \frac{\frac{f_2 - f_0}{x_2 - x_0} - \frac{f_1 - f_0}{x_1 - x_0}}{x_2 - x_1}$$

均差及其性质

定义 称 $f[x_0, x_k] = \frac{f(x_k) - f(x_0)}{x_k - x_0}$ 为 $f(x)$ 关于点 x_0, x_k 的一阶均差,

$$f[x_0, x_1, x_k] = \frac{f[x_0, x_k] - f[x_0, x_1]}{x_k - x_1} \text{ 为 } f(x) \text{ 的二阶均差.}$$

一般地, 称

$$f[x_0, x_1, \dots, x_k] = \frac{f[x_1, x_2, \dots, x_k] - f[x_0, x_1, \dots, x_{k-1}]}{x_k - x_0}$$

为 $f(x)$ 的 k 阶均差.

均差及其性质

均差由函数值可以表示为

$$f[x_0, x_1, \dots, x_k] = \sum_{j=0}^k \frac{f(x_j)}{(x_j - x_0) \cdots (x_j - x_{j-1})(x_j - x_{j+1}) \cdots (x_j - x_k)}$$

由此可知, 改变式中的节点次序, 差商值保持不变. 这种性质称为差商的对称性.

均差和导数的关系:

$$f[x_0, x_1, \dots, x_n] = \frac{f^{(n)}(\xi)}{n!}, \xi \in [a, b]$$

Newton插值公式

根据均差的定义, 把 x 看成 $[a, b]$ 上的一点, 可得

$$f(x) = f(x_0) + f[x, x_0](x-x_0)$$

$$f[x, x_0] = f[x_0, x_1] + f[x, x_0, x_1](x-x_1)$$

将后一个式子代入前一个式子, 得到

$$f(x) = f(x_0) + f[x_0, x_1](x-x_0) + f[x, x_0, x_1](x-x_0)(x-x_1)$$

再由 $f[x, x_0, x_1] = f[x_0, x_1, x_2] + f[x, x_0, x_1, x_2](x-x_2)$ 得到

$$f(x) = f(x_0) + f[x_0, x_1](x-x_0) + f[x_0, x_1, x_2](x-x_0)(x-x_1) + f[x, x_0, x_1, x_2](x-x_0)(x-x_1)(x-x_2)$$

Newton插值公式

将这一过程进行下去可以得到

$$\begin{aligned} f(x) = & f(x_0) + f[x_0, x_1](x-x_0) + f[x_0, x_1, x_2](x-x_0)(x-x_1) \\ & + \cdots + f[x_0, x_1, \dots, x_n](x-x_0)(x-x_1)\cdots(x-x_{n-1}) \\ & + f[x, x_0, x_1, \dots, x_n](x-x_0)(x-x_1)\cdots(x-x_{n-1})(x-x_n) \\ = & N_n(x) + R(x) \end{aligned}$$

称 $N_n(x)$ 为Newton插值多项式.

$N_n(x)$ 满足插值条件?

思路一(数学归纳法,利用Lagrange插值)

假设过 $(x_0, f_0), \dots, (x_{n-1}, f_{n-1})$ 的插值多项式 $L_{n-1}(x)$ 与 $N_{n-1}(x) = f(x_0) + f[x_0, x_1](x-x_0) + f[x_0, x_1, x_2](x-x_0)(x-x_1) + \dots + f[x_0, x_1, \dots, x_{n-1}](x-x_0)(x-x_1) \dots (x-x_{n-2})$ 恒等过 $(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)$ 的插值多项式为

$$L_n(x) = \sum_{k=0}^n \prod_{\substack{j=0 \\ j \neq k}}^n \frac{x-x_j}{x_k-x_j} f_k$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

73

$N_n(x)$ 满足插值条件?

$L_n(x) - N_{n-1}(x)$ 在 x_0, x_1, \dots, x_{n-1} 的值为零,因此

$$L_n(x) - N_{n-1}(x) = a_n(x-x_0)(x-x_1) \dots (x-x_{n-1})$$

比较两端 x^n 的系数得到

$$a_n = \sum_{j=0}^n \frac{f(x_j)}{(x_j-x_0) \dots (x_j-x_{j-1})(x_j-x_{j+1}) \dots (x_j-x_n)} = f[x_0, x_1, \dots, x_n]$$

$$\text{因此 } L_n(x) = N_{n-1}(x) + f[x_0, x_1, \dots, x_n](x-x_0) \dots (x-x_{n-1}) = N_n(x)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

74

$N_n(x)$ 满足插值条件?

思路二(数学归纳法,利用Neville逐次线性插值)作为作业.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

75

算例(Lagrange插值)

已知 $\sin 0.32 = 0.314567, \sin 0.34 = 0.333487, \sin 0.36 = 0.352274$, 用Lagrange插值的方法计算 $\sin 0.3367$, 并估计误差.

解: $x_0 = 0.32, x_1 = 0.34, x_2 = 0.36, y_0 = 0.314567, y_1 = 0.333487, y_2 = 0.352274$

采用线性插值有两种方法

$$\begin{aligned} (1) \quad \sin 0.3367 &\approx L_{01}(x) = y_0 \frac{x-x_1}{x_0-x_1} + y_1 \frac{x-x_0}{x_1-x_0} \\ &= 0.314567 \frac{0.3367-0.34}{0.32-0.34} + 0.333487 \frac{0.3367-0.32}{0.34-0.32} \\ &= 0.330365 \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

76

算例(Lagrange插值)

误差估计 $R_{01}(x) \leq \frac{M_2}{2} |(x-x_0)(x-x_1)|$

其中 $M_2 = \max_{x_0 \leq x \leq x_1} |f''(x)| = \max_{x_0 \leq x \leq x_1} \sin x = \sin x_1 \leq 0.3335$

因此 $|R_{01}(x)| \leq \frac{1}{2} \cdot 0.3335 \cdot 0.0167 \cdot 0.0033 \leq 0.92 \cdot 10^{-5}$

$$\begin{aligned} (2) \quad \sin 0.3367 &\approx L_{12}(x) = y_1 \frac{x-x_2}{x_1-x_2} + y_2 \frac{x-x_1}{x_2-x_1} \\ &= 0.333487 \frac{0.3367-0.36}{0.34-0.36} + 0.352274 \frac{0.3367-0.34}{0.32-0.34} \\ &= 0.330387 \end{aligned}$$

类似地可以计算误差

$$|R_{12}(x)| \leq \frac{1}{2} \cdot 0.3523 \cdot 0.0033 \cdot 0.0233 \leq 1.36 \cdot 10^{-5}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

77

算例(Lagrange插值)

抛物插值 $\sin 0.3367 \approx L_{012}(x)$

$$\begin{aligned} &= y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\ &= 0.314567 \frac{0.7689 \cdot 10^{-4}}{0.0008} + 0.333487 \frac{3.89 \cdot 10^{-4}}{0.0004} + 0.352274 \frac{-0.5511 \cdot 10^{-4}}{0.0008} \\ &= 0.330374 \end{aligned}$$

误差估计 $R_{012}(x) \leq \frac{M_3}{6} |(x-x_0)(x-x_1)(x-x_2)|$

$$M_3 = \max_{x_0 \leq x \leq x_2} |f'''(x)| = \max_{x_0 \leq x \leq x_2} \cos x = \cos x_0 \leq 0.828$$

$$|R_{012}(x)| \leq \frac{1}{6} \cdot 0.828 \cdot 0.0167 \cdot 0.0033 \cdot 0.0233 \leq 0.178 \cdot 10^{-6}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

78

算例(Newton插值)

给出 $f(x)$ 的函数值表,求4次插值多项式,并由此计算 $f(0.596)$ 的近似值.

解:首先构造均差表

1.11600=(0.57815-0.41075)/(0.55-0.40)						
0.40	0.41075					
0.55	0.57815	1.11600				
0.65	0.69675	1.18600	0.28000			
0.80	0.88811	1.27573	0.35893	0.19733		
0.90	1.02652	1.38410	0.43348	0.21300	0.03134	
1.05	1.25382	1.51533	0.52493	0.22363	0.03126	-0.00012
0.21300=(0.43348-0.35893)/(0.90-0.55)						

0.40	0.41075					
0.55	0.57815	1.11600				
0.65	0.69675	1.18600	0.28000			
0.80	0.88811	1.27573	0.35893	0.19733		
0.90	1.02652	1.38410	0.43348	0.21300	0.03134	
1.05	1.25382	1.51533	0.52493	0.22863	0.03126	-0.00012

根据均差表,可以求出4次插值函数

$$N_4(x)=0.41075+1.116(x-0.4)+0.28(x-0.4)(x-0.55)+0.17933(x-0.4)(x-0.55)(x-0.65)+0.03134(x-0.4)(x-0.55)(x-0.65)(x-0.8)$$

$$f(0.596)\approx N_4(0.596)=0.63195$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

80

截断误差约为

$$|R_4(x)|\approx|f[x_0,x_1,\dots,x_5](x-x_0)(x-x_1)\dots(x-x_5)|\leq 3.63\times 10^{-9}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

81

差分与等距节点插值公式

差分及其性质

等距节点: $x_0, x_1=x_0+h, \dots, x_k=x_0+kh, \dots, x_n=x_0+nh,$

函数值: $f_k=f(x_k)$

$f(x)$ 在 x_k 处的向前差分 $\Delta f_k = f_{k+1} - f_k$

向后差分 $\nabla f_k = f_k - f_{k-1}$

中心差分 $\delta f_k = f_{k+\frac{1}{2}} - f_{k-\frac{1}{2}}$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

82

高阶差分

一阶向前差分 $\Delta f_k = f_{k+1} - f_k$

二阶向前差分

$$\Delta^2 f_k = \Delta f_{k+1} - \Delta f_k = (f_{k+2} - f_{k+1}) - (f_{k+1} - f_k) = f_{k+2} - 2f_{k+1} + f_k$$

m 阶向前差分 $\Delta^m f_k = \Delta^{m-1} f_{k+1} - \Delta^{m-1} f_k$

m 阶向后差分 $\nabla^m f_k = \nabla^{m-1} f_{k+1} - \nabla^{m-1} f_k$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

83

差分的算子表示

引入不变算子 I 和移位算子 E $If_k=f_k, Ef_k=f_{k+1}$

则 $\Delta f_k=f_{k+1}-f_k=Ef_k-If_k=(E-I)f_k$, 于是 $\Delta=E-I$

类似的 $\nabla = I - E^{-1}, \delta = E^{\frac{1}{2}} - E^{-\frac{1}{2}}$

因此 $\Delta^n f_k = (E-I)^n f_k = \sum_{j=0}^n (-1)^j \binom{n}{j} E^{n-j} f_k$

$$= \sum_{j=0}^n (-1)^j \binom{n}{j} f_{n+k-j}$$

这是用函数值计算 n 阶向前差分的计算公式

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

84

差分的算子表示

类似得到函数值计算 n 阶向后差分的计算公式

$$\begin{aligned}\nabla^n f_k &= (I - E^{-1})^n f_k = \sum_{j=0}^n (-1)^{n-j} \binom{n}{j} E^{j-n} f_k \\ &= \sum_{j=0}^n (-1)^j \binom{n}{j} f_{k+j-n}\end{aligned}$$

差分表示函数值

$$f_{n+k} = E^n f_k = (I + \Delta)^n f_k = \left[\sum_{j=0}^n \binom{n}{j} \Delta^j \right] f_k = \sum_{j=0}^n \binom{n}{j} \Delta^j f_k$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

85

均差和差分的关系

$$f[x_k, x_{k+1}] = \frac{f_{k+1} - f_k}{x_{k+1} - x_k} = \frac{\Delta f_k}{h}$$

$$f[x_k, x_{k+1}, x_{k+2}] = \frac{f[x_{k+1}, x_{k+2}] - f[x_k, x_{k+1}]}{x_{k+2} - x_k} = \frac{1}{2h^2} \Delta^2 f_k$$

$$\text{一般的, } f[x_k, x_{k+1}, \dots, x_{k+m}] = \frac{1}{m!} \frac{1}{h^m} \Delta^m f_k \quad (m = 1, 2, \dots, n)$$

$$\text{类似, } f[x_k, x_{k-1}, \dots, x_{k-m}] = \frac{1}{m!} \frac{1}{h^m} \nabla^m f_k \quad (m = 1, 2, \dots, n)$$

根据导数和均差的关系,可以得到 $\Delta^n f_k = h^n f^{(n)}(\xi)$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

86

等距节点插值公式

Newton均差插值多项式

$$N_n(x) = f(x_0) + f[x_0, x_1](x-x_0) + f[x_0, x_1, x_2](x-x_0)(x-x_1) + \dots + f[x_0, x_1, \dots, x_n](x-x_0)(x-x_1) \dots (x-x_{n-1})$$

令 $x = x_0 + th$, 则 $x - x_0 = th, x - x_1 = (t-1)h, x - x_k = (t-k)h$,

因此

$$\begin{aligned}N_n(x_0 + th) &= f_0 + t \Delta f_0 + \frac{t(t-1)}{2} \Delta^2 f_0 \\ &\quad + \dots + \frac{t(t-1) \dots (t-n+1)}{n!} \Delta^n f_0\end{aligned}$$

这称为Newton向前差分插值公式(一般取 $0 \leq t \leq 1$).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

87

等距节点插值公式

Newton向前差分插值公式的余项

$$R_n(x_0 + th) = \frac{t(t-1) \dots (t-n)}{(n+1)!} h^{n+1} f^{(n+1)}(\xi), \xi \in (x_0, x_n)$$

Newton向后差分插值公式(一般 $t < 0$):

$$\begin{aligned}N_n(x_n + th) &= f_n + t \nabla f_n + \frac{t(t+1)}{2} \nabla^2 f_n \\ &\quad + \dots + \frac{t(t+1) \dots (t+n-1)}{n!} \nabla^n f_n\end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

88

算例

k	B_k	$f(B_k)$	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$
0	4000	1.38	0.10		
1	4500	1.48	0.10	0	
2	5000	1.58	0.11	0.01	0.01
3	5500	1.69	0.12	0.01	0
4	6000	1.81	0.13	0.01	0
5	6500	1.94	0.16	0.03	0.02
6	7000	2.10	0.18	0.02	-0.01
7	7500	2.28	0.22	0.04	0.02
8	8000	2.50	0.26	0.04	0
9	8500	2.76	0.30	0.04	0
10	9000	3.06	0.35	0.05	0.01
11	9500	3.41	0.42	0.07	0.02
12	10000	3.83	0.50	0.08	0.01
13	10500	4.33	0.60	0.10	0.02
14	11000	4.93			

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

89

左边的表给出了一个函数在 $[4000, 10500]$ 上等距节点的函数值(步长500), 用插值方法求 $f(5200)$.

解: 分别给出一,二,三阶差分. 发现三阶差分近似于零. 因此计算时只需用到二阶差分.

算例

k	B_k	$f(B_k)$	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$
0	4000	1.38	0.10		
1	4500	1.48	0.10	0	
2	5000	1.58	0.11	0.01	0.01
3	5500	1.69	0.12	0.01	0

计算 $f(5200)$ 时, 取 $B_0 = 5000, f_0 = 1.58, \Delta f_0 = 0.11, \Delta^2 f_0 = 0.01, h = 500, B = 5200, t = 0.4$,

根据Newton向前差分插值公式

$$f(5200) \approx 1.58 + 0.11 \cdot 0.4 + 0.01 \cdot 0.4(0.4-1)/2 = 1.62$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

90

Hermite插值

在某些问题中,为了保证插值函数能更好反映原来的函数,不但要求“过点”,即两者在节点上有相同的函数值,而且要求“相切”。

即在节点上还有相同的导数值,这类插值称作Hermite插值。

我们主要考虑在所有节点上的函数值与导数值都已知的情况。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

91

Hermite插值

设在节点 $a \leq x_0 < x_1 < \dots < x_n \leq b$ 上 $f(x_j) = y_j, m_j = f'(x_j)$ ($j=0, 1, \dots, n$), 求多项式 $H(x)$, 满足条件

$$H(x_j) = y_j, H'(x_j) = m_j (j=0, 1, \dots, n)$$

一共给出了 $2n+2$ 个已知条件, 可以唯一确定一个次数不超过 $2n+1$ 的多项式。

$$H_{2n+1}(x) = a_0 + a_1x + \dots + a_{2n+1}x^{2n+1}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

92

已知 x_0, \dots, x_n 处有 y_0, \dots, y_n 和 m_0, \dots, m_n , 求 $H_{2n+1}(x)$ 满足 $H_{2n+1}(x_i) = y_i, H'_{2n+1}(x_i) = m_i$ 。

解: 设 $H_{2n+1}(x) = \sum_{j=0}^n [y_j \alpha_j(x) + m_j \beta_j(x)]$

$$\alpha_j(x_k) = \delta_{jk} = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases}, \alpha'_j(x_k) = 0$$

$$\beta_j(x_k) = 0, \beta'_j(x_k) = \delta_{jk} = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

93

$\beta_j(x)$ 的确定

$$\beta_j(x_k) = 0, \beta'_j(x_k) = \delta_{jk} = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases}$$

根据上面的条件, 对于 $k=0, \dots, j-1, j+1, \dots, n$ 有

$\beta_j(x_k) = \beta'_j(x_k) = 0$, 因此 x_k 是 $\beta_j(x)$ 的二重根,

另外, $\beta_j(x_j) = 0$, 所以 $\beta_j(x)$ 有因式

$$(x-x_j)(x-x_0)^2 \dots (x-x_{j-1})^2 (x-x_{j+1})^2 \dots (x-x_n)^2$$

根据 $l_j(x) = \frac{(x-x_0) \dots (x-x_{j-1})(x-x_{j+1}) \dots (x-x_n)}{(x_j-x_0) \dots (x_j-x_{j-1})(x_j-x_{j+1}) \dots (x_j-x_n)}$

也可以认为 $\beta_j(x)$ 有因式 $(x-x_j)l_j^2(x)$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

94

$\beta_j(x)$ 的确定

$\beta_j(x)$ 有因式 $(x-x_j)l_j^2(x)$, 这已经是一个 $2n+1$ 次的多项式,

因此可以设 $\beta_j(x) = A(x-x_j)l_j^2(x)$

由 $1 = \beta'_j(x_j) = Al'_j(x_j) = A$, 得到

$$\beta_j(x) = (x-x_j)l_j^2(x)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

95

$\alpha_j(x)$ 的确定

$$\alpha_j(x_k) = \delta_{jk} = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases}, \alpha'_j(x_k) = 0$$

根据上面的条件, 对于 $k=0, \dots, j-1, j+1, \dots, n$ 有

$\alpha_j(x_k) = \alpha'_j(x_k) = 0$, 因此 x_k 是 $\alpha_j(x)$ 的二重根, 所以

$\alpha_j(x)$ 有因式 $l_j^2(x)$. 设 $\alpha_j(x) = (ax+b)l_j^2(x)$, 有

$$1 = \alpha_j(x_j) = (ax_j+b)l_j^2(x_j) = ax_j+b$$

$$0 = \alpha'_j(x_j) = al'_j(x_j) + 2l_j(x_j)l'_j(x_j)(ax_j+b) = a + 2l'_j(x_j)$$

从而得到 $a = -2l'_j(x_j), b = 1 + 2x_j l'_j(x_j)$

$$\alpha_j(x) = (1 - 2(x-x_j) \sum_{\substack{k=0 \\ k \neq j}}^n \frac{1}{x_j - x_k}) l_j^2(x)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

96

Hermite插值的唯一性

已知 x_0, \dots, x_n 处有 y_0, \dots, y_n 和 m_0, \dots, m_n , 求 $H_{2n+1}(x)$ 满足 $H_{2n+1}(x_i) = y_i, H'_{2n+1}(x_i) = m_i$.

问题: 这样的插值多项式是否唯一?

解答: 板书.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

97

插值余项

利用和Lagrange插值类似的思想, 可以得到 Hermite插值多项式的余项

$$R(x) = f(x) - H_{2n+1}(x) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!} \omega_{n+1}^2(x)$$

其中

$$\omega_{n+1}(x) = (x - x_0)(x - x_1) \cdots (x - x_n)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

98

两点三次Hermite插值

如果是两个点的Hermite插值, 则插值函数是三次的. 若要求

$H(x_k) = y_k, H(x_{k+1}) = y_{k+1}, H'(x_k) = m_k, H'(x_{k+1}) = m_{k+1}$,
则 $H(x) = y_k \alpha_k(x) + y_{k+1} \alpha_{k+1}(x) + m_k \beta_k(x) + m_{k+1} \beta_{k+1}(x)$

其中

$$\alpha_k(x) = (1 + 2 \frac{x - x_k}{x_{k+1} - x_k}) (\frac{x - x_{k+1}}{x_k - x_{k+1}})^2 \quad \beta_k(x) = (x - x_k) (\frac{x - x_{k+1}}{x_k - x_{k+1}})^2$$

$$\alpha_{k+1}(x) = (1 + 2 \frac{x - x_{k+1}}{x_k - x_{k+1}}) (\frac{x - x_k}{x_{k+1} - x_k})^2 \quad \beta_{k+1}(x) = (x - x_{k+1}) (\frac{x - x_k}{x_{k+1} - x_k})^2$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

99

利用均差作Hermite插值(补充)

上面的插值方法比较复杂, 下面介绍用均差进行Hermite插值的简单方法

问题: 求三次多项式 $H(x)$ 满足

$$H(x_k) = f(x_k), H(x_{k+1}) = f(x_{k+1}),$$

$$H'(x_k) = f'(x_k), H'(x_{k+1}) = f'(x_{k+1}).$$

解: 函数 $P_1(x) = f(x_k) + f'(x_k)(x - x_k)$ 显然满足

$$P_1(x_k) = f(x_k), P_1'(x_k) = f'(x_k).$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

100

利用均差作Hermite插值(补充)

我们先求函数 $P_2(x)$ (不超过二次), 使得

$$P_2(x_k) = f(x_k), P_2'(x_k) = f'(x_k), \text{ 且 } P_2(x_{k+1}) = f(x_{k+1}).$$

由于 $P_2(x_k) = P_1(x_k), P_2'(x_k) = P_1'(x_k)$,

因此 $P_2(x) - P_1(x)$ 有因式 $(x - x_k)^2$,

设 $P_2(x) - P_1(x) = A(x - x_k)^2$. 令 $x = x_{k+1}$ 得

$$f(x_{k+1}) = f(x_k) + f'(x_k)(x_{k+1} - x_k) = A(x_{k+1} - x_k)^2$$

$$\text{于是 } A = \frac{f(x_{k+1}) - f(x_k)}{(x_{k+1} - x_k)^2} - \frac{f'(x_k)}{x_{k+1} - x_k}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

101

利用均差作Hermite插值(补充)

$$\text{因此, } P_2(x) = f(x_k) + f'(x_k)(x - x_k) + \left[\frac{f(x_{k+1}) - f(x_k)}{(x_{k+1} - x_k)^2} - \frac{f'(x_k)}{x_{k+1} - x_k} \right] (x - x_k)^2$$

如果令 $f'(x_k) = f[x_k, x_k]$, 则上式可记为

$$P_2(x) = f(x_k) + f[x_k, x_k](x - x_k) + \frac{f[x_k, x_{k+1}] - f[x_k, x_k]}{x_{k+1} - x_k} (x - x_k)^2$$

$$\text{再令 } f[x_k, x_k, x_{k+1}] = \frac{f[x_k, x_{k+1}] - f[x_k, x_k]}{x_{k+1} - x_k}$$

$$P_2(x) = f(x_k) + f[x_k, x_k](x - x_k) + f[x_k, x_k, x_{k+1}](x - x_k)^2$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

102

利用均差作Hermite插值(补充)

令 $z_0=x_k, z_1=x_k, z_2=x_{k+1}$, 上式子可以记为
 $P_2(x) = f(z_0) + f[z_0, z_1](x-z_0) + f[z_0, z_1, z_2](x-z_0)(x-z_1)$.
 该式与Newton插值多项式的形式一致.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

103

利用均差作Hermite插值(补充)

求三次多项式 $H(x)$ 满足

$$H(x_k) = f(x_k), H(x_{k+1}) = f(x_{k+1}), \\ H'(x_k) = f'(x_k), H'(x_{k+1}) = f'(x_{k+1}).$$

显然 $H(x_k) = P_2(x_k), H'(x_k) = P_2'(x_k), H(x_{k+1}) = P_2(x_{k+1})$
 因此可以设 $H(x) = P_2(x) + a_3(x-x_k)^2(x-x_{k+1})$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

104

利用均差作Hermite插值(补充)

$H(x) = P_2(x) + a_3(x-x_k)^2(x-x_{k+1})$
 令 $z_0=z_1=x_k, z_2=z_3=x_{k+1}$, 上式子可以记为
 $H(x) = f(z_0) + f[z_0, z_1](x-z_0) + f[z_0, z_1, z_2](x-z_0)(x-z_1) \\ + a_3(x-z_0)(x-z_1)(x-z_2)$
 我们可以猜想上述结果与Newton插值多项式一致, 即有 $a_3 = f[z_0, z_1, z_2, z_3] = f[x_k, x_k, x_{k+1}, x_{k+1}]$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

105

$$f[x_k, x_k, x_{k+1}, x_{k+1}]?$$

首先的问题是 $f[x_k, x_k, x_{k+1}, x_{k+1}]$ 如何计算?
 按照均差的定义, 应该有

$$f[x_k, x_k, x_{k+1}, x_{k+1}] = \frac{f[x_k, x_{k+1}, x_{k+1}] - f[x_k, x_k, x_{k+1}]}{x_{k+1} - x_k} \\ f[x_k, x_k, x_{k+1}] \text{ 我们已经得到, } f[x_k, x_{k+1}, x_{k+1}] \text{ 可以类似计算.} \\ = \frac{f[x_{k+1}, x_{k+1}] - f[x_k, x_{k+1}]}{x_{k+1} - x_k} \\ = \frac{f'(x_{k+1}) - f[x_k, x_{k+1}]}{x_{k+1} - x_k}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

106

$$a_3 = f[x_k, x_k, x_{k+1}, x_{k+1}]? \\ H(x) = f(z_0) + f[z_0, z_1](x-z_0) + f[z_0, z_1, z_2](x-z_0)(x-z_1) \\ + a_3(x-z_0)(x-z_1)(x-z_2) \\ = f(x_k) + f[x_k, x_k](x-x_k) + f[x_k, x_k, x_{k+1}](x-x_k)^2 \\ + a_3(x-x_k)^2(x-x_{k+1})$$

我们利用 $H(x)$ 在 x_k 的函数值, 导数值以及 x_{k+1} 处的函数值得到了上述结果, $H(x)$ 在 x_{k+1} 处的导数值可以用来确定 a_3 .

我们求出的 a_3 与前面定义的 $f[x_k, x_k, x_{k+1}, x_{k+1}]$ 是相等的(过程略).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

107

利用均差作Hermite插值(补充)

根据上面的推导, 有下面的结论

$$H(x) = f(z_0) + f[z_0, z_1](x-z_0) + f[z_0, z_1, z_2](x-z_0)(x-z_1) \\ + f[z_0, z_1, z_2, z_3](x-z_0)(x-z_1)(x-z_2)$$

即进行Hermite插值时, 我们可以采用Newton插值公式. 其中的均差计算在遇到相同的节点时用导数代替.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

108

利用均差作Hermite插值(算例)

求一次数最低的多项式 $H(x)$,使得

$$H(-1)=-1, H'(-1)=1, H(1)=-3, H'(1)=-3, H''(1)=14, H(2)=29.$$

解:一共6个已知条件,可以唯一确定一个不超过5次的多项式

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

109

利用均差作Hermite插值(算例)

首先计算均差表

x	$H(x)$	一阶均差
-1	-1	
-1	-1	1
1	-3	-1
1	-3	-3
1	-3	-3
2	29	32

$$f[-1,-1] = f'(-1) = 1$$

$$f[-1,1] = \frac{f[1] - f[-1]}{1 - (-1)} = \frac{-3 - (-1)}{1 - (-1)} = -1$$

$$f[1,1] = f'(1) = -3$$

$$f[1,2] = \frac{f[2] - f[1]}{2 - 1} = \frac{29 - (-3)}{2 - 1} = 32$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

110

利用均差作Hermite插值(算例)

x	$H(x)$	一阶均差	二阶均差
-1	-1		
-1	-1	1	
1	-3	-1	-1
1	-3	-3	-1
1	-3	-3	
2	29	32	

$$f[-1,-1,1] = \frac{f[-1,1] - f[-1,-1]}{1 - (-1)} = \frac{-1 - 1}{1 - (-1)} = -1$$

$$f[-1,1,1] = \frac{f[1,1] - f[-1,1]}{1 - (-1)} = \frac{-3 - (-1)}{1 - (-1)} = -1$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

111

利用均差作Hermite插值(算例)

x	$H(x)$	一阶均差	二阶均差
-1	-1		
-1	-1	1	
1	-3	-1	-1
1	-3	-3	-1
1	-3	-3	7
2	29	32	35

$$f[1,1,1] = \frac{f''(1)}{2} = \frac{14}{2} = 7$$

一般的, $n+1$ 个相同节点 x_0 的 n 阶均差为 $\frac{f^{(n)}(x_0)}{n!}$

$$f[1,1,2] = \frac{f[1,2] - f[1,1]}{2 - 1} = \frac{32 - (-3)}{2 - 1} = 35$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

112

利用均差作Hermite插值(算例)

x	$H(x)$	一阶均差	二阶均差	三阶均差
-1	-1			
-1	-1	1		
1	-3	-1	-1	
1	-3	-3	-1	0
1	-3	-3	7	4
2	29	32	35	

$$f[-1,-1,1,1] = \frac{f[-1,1,1] - f[-1,-1,1]}{1 - (-1)} = \frac{-1 - (-1)}{1 - (-1)} = 0$$

$$f[-1,1,1,1] = \frac{f[1,1,1] - f[-1,1,1]}{1 - (-1)} = \frac{7 - (-1)}{1 - (-1)} = 4$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

113

利用均差作Hermite插值(算例)

x	$H(x)$	一阶均差	二阶均差	三阶均差
-1	-1			
-1	-1	1		
1	-3	-1	-1	
1	-3	-3	-1	0
1	-3	-3	7	4
2	29	32	35	28

$$f[1,1,1,2] = \frac{f[1,1,2] - f[1,1,1]}{2 - 1} = \frac{35 - 7}{2 - 1} = 28$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

114

利用均差作Hermite插值(算例)

x	$H(x)$	一阶均差	二阶均差	三阶均差	四阶均差
-1	-1				$f[-1,-1,1,1,1]$
-1	-1	1			$= \frac{f[-1,1,1,1] - f[-1,-1,1,1]}{1 - (-1)}$
1	-3	-1	-1		$= \frac{4-0}{1-(-1)} = 2$
1	-3	-3	-1	0	
1	-3	-3	7	4	2
2	29	32	35	28	

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

115

利用均差作Hermite插值(算例)

x	$H(x)$	一阶均差	二阶均差	三阶均差	四阶均差
-1	-1				$f[-1,1,1,1,2]$
-1	-1	1			$= \frac{f[-1,1,2] - f[-1,1,1,1]}{2 - (-1)}$
1	-3	-1	-1		$= \frac{28-4}{2-(-1)} = 8$
1	-3	-3	-1	0	
1	-3	-3	7	4	2
2	29	32	35	28	8

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

116

利用均差作Hermite插值(算例)

x	$H(x)$	一阶均差	二阶均差	三阶均差	四阶均差	五阶均差
-1	-1					$f[-1,-1,1,1,1,2]$
-1	-1	1				$= \frac{f[-1,1,1,1,2] - f[-1,-1,1,1,1]}{2 - (-1)}$
1	-3	-1	-1			$= \frac{8-2}{2-(-1)} = 2$
1	-3	-3	-1	0		
1	-3	-3	7	4	2	
2	29	32	35	28	8	2

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

117

我们得到 $H(x) = -1 + 1 \cdot (x+1) + (-1) \cdot (x+1)^2 + 0 \cdot (x+1)^2(x-1) + 2 \cdot (x+1)^2(x-1)^2 + 2(x+1)^2(x-1)^3$

$$= 2x^5 - 4x^3 - x^2 + x - 1$$

-1	-1					
-1	-1	1				
1	-3	-1	-1			
1	-3	-3	-1	0		
1	-3	-3	7	4	2	
2	29	32	35	28	8	2

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

118

例5: 设 $x_0 \neq x_1 \neq x_2$, 已知 $f(x_0), f(x_1), f(x_2)$ 和 $f'(x_1)$, 求多项式 $P(x)$ 满足 $P(x_i) = f(x_i), i = 0, 1, 2$, 且 $P'(x_1) = f'(x_1)$, 并估计误差。

解: 首先, P 的阶数=3

模仿 Lagrange 多项式的思想, 设

$$P_3(x) = \sum_{i=0}^2 f(x_i) \alpha_i(x) + f'(x_1) \beta_1(x)$$

其中 $\alpha_i(x_j) = \delta_{ij}$, $\alpha_i'(x_1) = 0, \beta_1(x_i) = 0, \beta_1'(x_1) = 1$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

119

$\alpha_0(x)$ $\alpha_0(x_1) = \alpha_0(x_2) = 0$, 因此, $\alpha_0(x)$ 有根 x_1, x_2 , 又 $\alpha_0'(x_1) = 0 \Rightarrow x_1$ 是重根。

因此

又: $\alpha_0(x_0) = 1 \Rightarrow C_0 \Rightarrow \alpha_0(x) = \frac{(x-x_1)^2(x-x_2)}{(x_0-x_1)^2(x_0-x_2)}$

$\alpha_2(x)$ 与 $\alpha_0(x)$ 完全类似。

$$\alpha_2(x) = \frac{(x-x_1)^2(x-x_0)}{(x_2-x_1)^2(x_2-x_0)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

120

$\alpha_1(x)$ 有根 $x_0, x_2 \Rightarrow$

$$\alpha_1(x) = (Ax + B)(x - x_0)(x - x_2)$$

由余下条件 $\alpha_1(x_1) = 1$ 和 $\alpha_1'(x_1) = 0$ 可解。

$\beta_1(x)$ 有根 $x_0, x_1, x_2 \Rightarrow$

$$\beta_1(x) = C_1(x - x_0)(x - x_1)(x - x_2)$$

又, $\beta_1'(x_1) = 1 \Rightarrow C_1$ 可解。

$$K(x) = \frac{f^{(4)}(\xi_x)}{4!}$$

与 Lagrange 分析
完全类似

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

121

方法2

利用均差的方法也可以构造出满足题中条件的多项式——作业

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

122

求 Hermite 多项式的基本步骤 (基函数方法)

- ① 写出相应于条件的基函数的组合形式;
- ② 对每一个基函数找出尽可能多的条件给出的根;
- ③ 根据多项式的总阶数和根的个数写出表达式;
- ④ 根据尚未利用的条件解出表达式中的待定系数;
- ⑤ 最后完整写出 $H(x)$ 。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

123

求 Hermite 多项式的基本步骤 (均差方法)

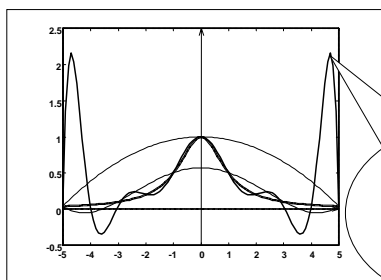
- ① 计算均差表
对于不同的节点,直接按照定义计算,
对于相同的节点,用导数代替。
- ② 根据 Newton 插值公式计算插值多项式

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

124

分段低次插值

例:在 $[-5, 5]$ 上考察函数 $f(x) = 1/(1+x^2)$ 的 Lagrange 插值函数,节点取为等距的。



$L_n(x) \approx f(x)$

n 越大,
端点附近抖动
越大,称为
Runge 现象



分段低次插值

华制作 njuptshumo2006@126.com

125

分段线性插值

所谓分段线性插值,就是将插值点用折线段相连来逼近被插值的函数。

设已知节点 $a \leq x_0 < x_1 < \dots < x_n \leq b$ 上的函数值 f_0, f_1, \dots, f_n , 记 $h_k = x_{k+1} - x_k$, $h = \max h_k$, 求一折线函数 $I_h(x)$ 满足: (1)

(2) $I_h(x_k) = f_k (k=0, 1, \dots, n)$

(3) $I_h(x)$ 在每个区间 $[x_k, x_{k+1}]$ 上是线性函数
满足上述条件的 $I_h(x)$ 称为分段线性插值函数。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

126

分段线性插值

显然, $I_h(x)$ 在每个小区间 $[x_k, x_{k+1}]$ 上可以表示为

$$I_h(x) = \frac{x - x_{k+1}}{x_k - x_{k+1}} f_k + \frac{x - x_k}{x_{k+1} - x_k} f_{k+1} \quad (x_k \leq x \leq x_{k+1})$$

$I_h(x)$ 也可以用插值基函数表示 $I_h(x) = \sum_{j=0}^n f_j l_j(x)$

其中 $l_j(x)$ 满足 $l_j(x_k) = \delta_{jk}$

$$l_j(x) = \begin{cases} \frac{x - x_{j-1}}{x_j - x_{j-1}}, & x_{j-1} \leq x \leq x_j \\ \frac{x - x_j}{x_{j+1} - x_j}, & x_j \leq x \leq x_{j+1} \\ 0 & x \in [a, b], x \notin [x_{j-1}, x_{j+1}] \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

127

分段线性插值的收敛性

类似于Lagrange插值基函数,分段线性插值基函数的和也是1,即 $\sum_{j=0}^n l_j(x) = 1$

当 x 位于区间 $[x_k, x_{k+1}]$ 时,基函数中只有 $l_k(x), l_{k+1}(x)$ 不为零,此时有

$$1 = l_k(x) + l_{k+1}(x), \quad f(x) = (l_k(x) + l_{k+1}(x))f(x)$$

另一方面,由于基函数中只有 $l_k(x), l_{k+1}(x)$ 不为零,
 $I_h(x) = f_k l_k(x) + f_{k+1} l_{k+1}(x)$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

128

分段线性插值的收敛性

$$f(x) = (l_k(x) + l_{k+1}(x))f(x) \quad I_h(x) = f_k l_k(x) + f_{k+1} l_{k+1}(x)$$

$$\text{因此 } |f(x) - I_h(x)| = |l_k(x)(f(x) - f_k) + l_{k+1}(x)(f(x) - f_{k+1})|$$

$$\leq l_k(x)|f(x) - f_k| + l_{k+1}(x)|f(x) - f_{k+1}|$$

$$\leq l_k(x)\omega(h_k) + l_{k+1}(x)\omega(h_k) \leq \omega(h)$$

这里 $\omega(h)$ 称为 $f(x)$ 在区间 $[a, b]$ 的连续模.

$$(|u - v| \leq h, \text{ 则有 } |f(u) - f(v)| \leq \omega(h))$$

如果 $f(x)$ 是区间 $[a, b]$ 上的连续函数,则

$h \rightarrow 0$ 时 $\omega(h) \rightarrow 0$. 因此 $I_h(x) \rightarrow f(x)$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

129

分段三次Hermite插值简介

已知 x_k, f_k, f'_k ($k=0, 1, \dots, n$), 求 $I_h(x)$, 使得

1) $I_h(x)$ 一阶导数连续

2) $I_h(x_k) = f_k, I_h'(x_k) = f'_k$ ($k=0, 1, \dots, n$)

3) $I_h(x)$ 在每个区间 $[x_k, x_{k+1}]$ 上是三次函数

类似于分段线性插值,可以写出分段三次埃尔米特插值函数,而且当区间长度的最大值 h 趋于零时, $I_h(x)$ 一致收敛到 $f(x)$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

130

三次样条插值

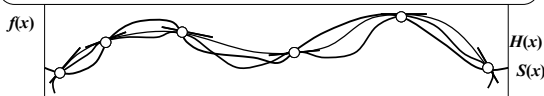
定义设 $a \leq x_0 < x_1 < \dots < x_n \leq b$, 三次样条函数

即在每个 $[x_i, x_{i+1}]$ 上为三次多项式.

若它同时还满足 $S(x_i) = f(x_i)$ ($i=0, 1, \dots, n$), 则称为 f 的

三次样条插值函数.

注: 三次样条与分段 Hermite 插值的根本区别在于 $S(x)$ 自身光滑, 不需要知道 f 的导数值 (除了在2个端点可能需要); 而 Hermite 插值依赖于 f 在所有插值点的导数值.



131

三弯矩法

在 $[x_j, x_{j+1}]$ 上, 记 $h_j = x_{j+1} - x_j$, 在此区间内, $S''(x)$ 为1次多项式, 记 $S''(x_j) = M_j, S''(x_{j+1}) = M_{j+1}$, 则

$$S''(x) = M_j \frac{x_{j+1} - x}{h_j} + M_{j+1} \frac{x - x_j}{h_j}$$

积分2次, 可得 $S'(x)$ 和 $S(x)$

$$S'(x) = -M_j \frac{(x_{j+1} - x)^2}{2h_j} + M_{j+1} \frac{(x - x_j)^2}{2h_j} + A_j$$

$$S(x) = M_j \frac{(x_{j+1} - x)^3}{6h_j} + M_{j+1} \frac{(x - x_j)^3}{6h_j} + A_j x + B_j$$

$$A_j = \frac{y_{j+1} - y_j}{h_j} - \frac{M_{j+1} - M_j}{6} h_j$$

$$B_j = \frac{x_{j+1} y_j - x_j y_{j+1}}{h_j} + \frac{x_j M_{j+1} - x_{j+1} M_j}{6} h_j$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

132

下面解决 M_j : 利用 S' 在 x_j 的连续性

$$[x_{j-1}, x_j]: S^{[j]}(x) = -M_{j-1} \frac{(x_j - x)^2}{2h_{j-1}} + M_j \frac{(x - x_{j-1})^2}{2h_{j-1}} + f[x_{j-1}, x_j] - \frac{M_j - M_{j-1}}{6} h_{j-1}$$

$$[x_j, x_{j+1}]: S^{[j+1]}(x) = -M_j \frac{(x_{j+1} - x)^2}{2h_j} + M_{j+1} \frac{(x - x_j)^2}{2h_j} + f[x_j, x_{j+1}] - \frac{M_{j+1} - M_j}{6} h_j$$

因此, $S'(x_j - 0) = \frac{h_{j-1}}{6} M_{j-1} + \frac{h_{j-1}}{3} M_j + \frac{y_j - y_{j-1}}{h_{j-1}}$

$$S'(x_j + 0) = -\frac{h_{j-1}}{6} M_{j+1} - \frac{h_j}{3} M_j + \frac{y_{j+1} - y_j}{h_j}$$

利用 $S'(x_j - 0) = S'(x_j + 0)$, 可以得到关于 M_{j-1}, M_j, M_{j+1} 的线性方程

$$\mu_j M_{j-1} + 2M_j + \lambda_j M_{j+1} = d_j \quad (j = 1, 2, \dots, n-1)$$

其中, $\mu_j = \frac{h_j}{h_{j-1} + h_j}, \lambda_j = \frac{h_{j-1}}{h_{j-1} + h_j}, d_j = 6f[x_{j-1}, x_j, x_{j+1}]$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

133

一共 $n+1$ 个未知数, $n-1$ 个方程, 还需要两个条件
第一类边界条件, $S'(x_0) = f'_0, S'(x_n) = f'_n$.

此时容易得到 $2M_0 + M_1 = \frac{6}{h_0} (f[x_0, x_1] - f'_0)$

$$M_{n-1} + 2M_n = \frac{6}{h_{n-1}} (f'_n - f[x_{n-1}, x_n])$$

最终得到如下形式的方程组

$$\begin{bmatrix} 2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ \mu_1 & 2 & \lambda_1 & \cdots & 0 & 0 & 0 \\ 0 & \mu_2 & 2 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 2 & \lambda_{n-1} & 0 \\ 0 & 0 & 0 & \cdots & \mu_{n-1} & 2 & \lambda_{n-1} \\ 0 & 0 & 0 & \cdots & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} M_0 \\ M_1 \\ M_2 \\ \vdots \\ M_{n-2} \\ M_{n-1} \\ M_n \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{n-2} \\ d_{n-1} \\ d_n \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

134

追赶法求解三对角方程组

上面的方程组可以利用追赶法求解(P185).

对于下面形式的方程组

$$\begin{bmatrix} b_1 & c_1 & & & & \\ a_2 & b_2 & c_2 & & & \\ & & & \ddots & & \\ & & & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & & & a_n & b_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n \end{bmatrix}$$

将系数矩阵进行三角分解

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

135

$$\begin{bmatrix} b_1 & c_1 & & & & \\ a_2 & b_2 & c_2 & & & \\ & & & \ddots & & \\ & & & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & & & a_n & b_n \end{bmatrix} = \begin{bmatrix} \alpha_1 & & & & & \\ r_2 & \alpha_2 & & & & \\ & r_3 & \alpha_3 & & & \\ & & & \ddots & & \\ & & & & \alpha_{n-1} & \\ & & & & r_n & \alpha_n \end{bmatrix} \begin{bmatrix} 1 & \beta_1 & & & & \\ & 1 & \beta_2 & & & \\ & & \ddots & \ddots & & \\ & & & 1 & \beta_{n-1} & \\ & & & & 1 & \beta_n \end{bmatrix}$$

比较两边对应元素可以得到

$$\begin{aligned} b_1 &= \alpha_1, \quad c_1 = \alpha_1 \beta_1, \quad a_i = r_i \quad (i = 2, \dots, n), \\ b_i &= r_i \beta_{i-1} + \alpha_i \quad (i = 2, \dots, n), \quad c_i = \alpha_i \beta_i \quad (i = 2, \dots, n-1). \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

136

$$\begin{aligned} b_1 &= \alpha_1, \quad c_1 = \alpha_1 \beta_1, \quad a_i = r_i \quad (i = 2, \dots, n), \\ b_i &= r_i \beta_{i-1} + \alpha_i \quad (i = 2, \dots, n), \quad c_i = \alpha_i \beta_i \quad (i = 2, \dots, n-1). \end{aligned}$$

因此有

$$\beta_i = \frac{c_i}{\alpha_i} = \frac{c_i}{b_i - r_i \beta_{i-1}} = \frac{c_i}{b_i - a_i \beta_{i-1}} \quad (i = 2, \dots, n-1)$$

又 $\beta_1 = \frac{c_1}{\alpha_1} = \frac{c_1}{b_1}$

因此所有 β_i 的可递推出, 进一步可求出 α_i, r_i .

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

137

在得到系数矩阵的分解后, 原方程组转化为

$$LUx = f, \quad \text{先求解 } Ly = f$$

$$\begin{bmatrix} \alpha_1 & & & & \\ r_2 & \alpha_2 & & & \\ & r_3 & \alpha_3 & & \\ & & & \ddots & \\ & & & & \alpha_{n-1} & \alpha_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n \end{bmatrix}$$

显然有 $y_1 = f_1 / \alpha_1$,

$$y_i = (f_i - r_i y_{i-1}) / \alpha_i = (f_i - \alpha_i y_{i-1}) / (b_i - a_i \beta_{i-1}) \quad (i = 2, \dots, n)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

138

再求解 $Ux=y$,

$$\begin{bmatrix} 1 & \beta_1 & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \beta_{n-1} \\ & & & & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix}$$

显然有 $x_n=y_n$, $x_i=y_i-b_i x_{i+1} (i=n-1, \dots, 1)$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

139

插值法小结

- ◆ Lagrange: 给出 $y_0 \dots y_n$, 选基函数 $L_i(x)$, 其次数为节点数 -1 .
- ◆ Newton: $N_n(x) \equiv L_n(x)$, 只是形式不同; 节点等距或渐增节点时方便处理.
- ◆ Hermite: 给出 y_i 及 y_i' , 选基函数或用均差方法
- ◆ 样条: 分段低次, 自身光滑, f 的导数只在边界给出。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

140

Matlab中的插值命令

一元插值格式: `interp1(x,y,x1,method)`

其中: x : 插值节点, y : 对应函数值

$x1$: 要计算函数值的点

$method$: 选用的插值方法

'nearest': 0次插值,

'linear': 分段线性(缺省时),

'spline': 样条,

'pchip': 分段三次(保证一阶导数连续)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

141

Matlab中的插值命令

二元插值格式: `interp2(x,y,z,x1,y1,method)`

缺省时的方法为双线性

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

142

第三章 函数逼近与计算

曲线拟合 (P68)

仍然是已知 $x_1 \dots x_m; y_1 \dots y_m$, 求一个简单易算的近似函数 $P(x) \approx f(x)$ 。

但是 ① m 很大;

② y_i 本身是测量值, 不准确, 即 $y_i \neq f(x_i)$

这时没必要取 $P(x_i) = y_i$, 而要使 $P(x_i) - y_i$ 总体上尽可能小。

常 使 $\max_{1 \leq i \leq m} |P(x_i) - y_i|$ 最小

见 使 $\sum_{i=1}^m |P(x_i) - y_i|$ 最小

做法 使 $\sum_{i=1}^m |P(x_i) - y_i|^2$ 最小 (最小二乘法)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

144

最小二乘拟合多项式

确定多项式 $P(x) = a_0 + a_1x + \dots + a_nx^n$ ，对于一组数据 (x_i, y_i) ($i = 1, 2, \dots, m$) 使得 $E = \sum_{i=1}^m [P(x_i) - y_i]^2$ 达到极小，这里 $n \ll m$ 。

E 实际上是 a_0, a_1, \dots, a_n 的多元函数，即

$$E(a_0, a_1, \dots, a_n) = \sum_{i=1}^m [a_0 + a_1x_i + \dots + a_nx_i^n - y_i]^2$$

在 E 的极值点应有

$$\frac{\partial E}{\partial a_k} = 0, \quad k = 0, \dots, n$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

145

$$E(a_0, a_1, \dots, a_n) = \sum_{i=1}^m [P(x_i) - y_i]^2$$

$$0 = \frac{\partial E}{\partial a_k} = 2 \sum_{i=1}^m [P(x_i) - y_i] \frac{\partial P(x_i)}{\partial a_k} = 2 \sum_{i=1}^m [\sum_{j=0}^n a_j x_i^j - y_i] x_i^k$$

$$= 2 \{ \sum_{j=0}^n [\sum_{i=1}^m x_i^{j+k}] a_j - \sum_{i=1}^m y_i x_i^k \}$$

$$\text{记 } b_k = \sum_{i=1}^m x_i^k \quad d_k = \sum_{i=1}^m y_i x_i^k$$

$$\text{得法方程组} \begin{bmatrix} b_{0+0} & \dots & b_{0+n} \\ b_{1+0} & \dots & b_{1+n} \\ \vdots & \vdots & \vdots \\ b_{n+0} & \dots & b_{n+n} \end{bmatrix} \begin{bmatrix} a_0 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} d_0 \\ \vdots \\ d_n \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

146

最小二乘算例(P70例9)

在某化学反应里,根据实验所得生成物的浓度和时间关系如下表,求浓度 y 与时间 t 的拟合曲线 $y = F(t)$ 。

时间 t		1	2	3	4	5	6	7
浓度 y		4.00	6.40	8.00	8.80	9.22	9.50	9.70
8	9	10	11	12	13	14	15	16
9.86	10.00	10.20	10.32	10.42	10.50	10.55	10.58	10.60

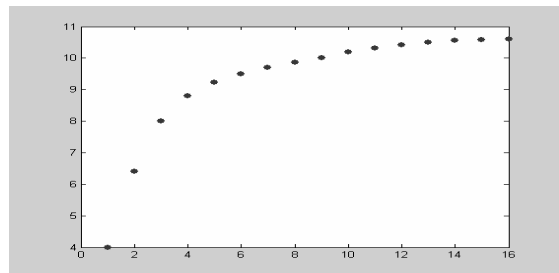
南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

147

将数据在图形上画出。

可以发现数据有一定的规律性。

但是函数的类型我们无法确定。



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

148

方案一

$$y \approx F(t) = \frac{t}{at+b}$$

求 a 和 b 使得 $\varphi(a, b) = \sum_{i=1}^m (\frac{t_i}{at_i+b} - y_i)^2$ 最小。

直接求解很困难

线性化

令 $Y = 1/y, T = 1/t$, 则有 $Y \approx a + bT$, 成为线性问题

由 (t_i, y_i) 容易求得 (T_i, Y_i) , 然后可解出 a, b 。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

149

$$\sum_{i=1}^m T_i^0 = 16, \sum_{i=1}^m T_i^1 = 3.38073, \sum_{i=1}^m T_i^2 = 1.58435,$$

$$\sum_{i=1}^m Y_i T_i^0 = 1.83291, \sum_{i=1}^m Y_i T_i^1 = 0.52886.$$

$$\text{得到法方程组为} \begin{cases} 16a + 3.38073b = 1.83291 \\ 3.38073a + 1.58435b = 0.52886 \end{cases}$$

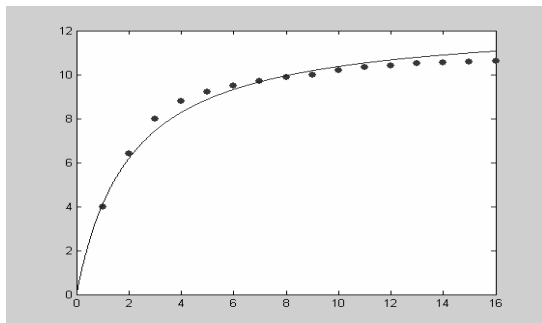
$$\text{解得} \begin{cases} a = 0.0801736 \\ b = 0.1627258 \end{cases}$$

$$\text{得到 } y = \frac{t}{0.0801736t + 0.1627258}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

150

下面是所得的函数的图形



151

方案二 $y \approx F(t) = ae^{\frac{b}{t}}$

线性化

对函数取对数得到, $\ln y \approx \ln a + b/t$

令 $Y = \ln y, T = 1/t, A = \ln a$, 则有 $Y \approx A + bT$, 成为线性问题

由 (t_i, y_i) 容易求得 (T_i, Y_i) , 然后可解出 a, b .

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

152

$$\sum_{i=1}^m T_i^0 = 16, \sum_{i=1}^m T_i^1 = 3.38073, \sum_{i=1}^m T_i^2 = 1.58435,$$

$$\sum_{i=1}^m Y_i T_i^0 = 35.26017, \sum_{i=1}^m Y_i T_i^1 = 6.53099.$$

得到法方程组为 $\begin{cases} 16A + 3.38073b = 35.2602 \\ 3.38073A + 1.58435b = 6.53099 \end{cases}$

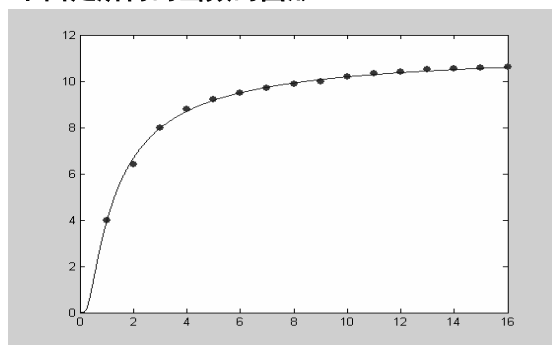
解得 $\begin{cases} A = 2.4270 \\ b = -1.0567 \end{cases} \quad \begin{cases} a = e^A = 11.3253 \\ b = -1.0567 \end{cases}$

得到 $y = 11.3253e^{-1.0567/t}$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

153

下面是所得的函数的图形



154

从图形上可以看出,第二个函数拟合的效果较好.

我们可以比较两者的均方误差.

$$\sqrt{\sum_{i=1}^m (\delta_i^{(1)})^2} = 1.25, \sqrt{\sum_{i=1}^m (\delta_i^{(2)})^2} = 0.34$$

第二个函数的均方误差小一些.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

155

离散问题 已知 $x_1 \dots x_m; y_1 \dots y_m$, 求一个简单易算的近似函数 $P(x) \approx f(x)$ 使得 $\sum_{i=1}^m (P(x_i) - y_i)^2$ 最小.

连续问题 已知 $[a, b]$ 上定义的 $f(x)$, 求一个简单易算的近似函数 $P(x)$ 使得 $\int_a^b [P(x) - f(x)]^2 dx$ 最小.

最常见的近似函数是多项式

$$P(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

有时我们会考虑其它类型的函数

$$P(x) = \sum_{j=0}^n \alpha_j \varphi_j(x)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

156

线性无关函数族

考虑函数 $1+\cos^2x+2\sin^2x$ 以及 $2\cos^2x+3\sin^2x$
可以发现这两个函数是相等的.

原因: $\cos^2x+\sin^2x=1$

即 $1, \sin^2x, \cos^2x$ 是线性相关的.

线性无关函数族 $\{\varphi_0(x), \varphi_1(x), \dots, \varphi_n(x), \dots\}$

满足条件: 其中任意函数的线性组合

$a_0\varphi_0(x)+a_1\varphi_1(x)+\dots+a_n\varphi_n(x)=0$ 对任意 $x \in [a, b]$

成立当且仅当 $a_0=a_1=\dots=a_n=0$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

157

考虑一般的线性无关函数族 $\Phi=\{\varphi_0(x), \varphi_1(x), \dots, \varphi_n(x), \dots\}$, 其有限项的线性组合 $P(x)=\sum_{j=0}^n \alpha_j \varphi_j(x)$ 称为广义多项式.

常见广义多项式:

- $\{\varphi_j(x)=x^j\}$ 对应代数多项式
- $\{\varphi_j(x)=\cos jx\}, \{\psi_j(x)=\sin jx\} \Rightarrow \{\varphi_j(x), \psi_j(x)\}$ 对应三角多项式
- $\{\varphi_j(x)=e^{kjx}, k_i \neq k_j\}$ 对应指数多项式

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

158

权函数

① 离散型(P69)

根据一系列离散点 $(x_i, y_i) (i=1, \dots, m)$ 拟合时, 在每一误差前乘一正数 w_i , 即误差函数 $E=\sum_{i=1}^m w_i [P(x_i)-y_i]^2$, 这个 w_i 就称作权, 反映该点的重要程度.

② 连续型(P50)

在 $[a, b]$ 上用广义多项式 $P(x)$ 拟合连续函数 $f(x)$ 时, 定义权函数 $\rho(x) \in C[a, b]$, 即误差函数 $E=\int_a^b \rho(x) [P(x)-y(x)]^2 dx$ 权函数必须 $\rho(x)$ 满足: 非负, 可积, 且在 $[a, b]$ 的任何子区间上 $\rho(x) \neq 0$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

159

广义最小二乘拟合

① 离散型(P69)

在点集 $\{x_1 \dots x_m\}$ 上测得 $\{y_1 \dots y_m\}$, 在一组权重系数 $\{w_1 \dots w_m\}$ 下求广义多项式 $P(x)$ 使得误差函数 $E=\sum_{i=1}^m w_i [P(x_i)-y_i]^2$ 最小.

② 连续型(P53)

已知 $y(x) \in C[a, b]$ 以及权函数 $\rho(x)$, 求广义多项式 $P(x)$ 使得误差函数 $E=\int_a^b \rho(x) [P(x)-y(x)]^2 dx$ 最小.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

160

离散型广义最小二乘问题求解(P69)

$P(x)=a_0\varphi_0(x)+a_1\varphi_1(x)+\dots+a_n\varphi_n(x)$ $E=\sum_{i=1}^m w_i [P(x_i)-y_i]^2$
为求误差函数 E 的最小值, 令其对 $a_k (k=0, 1, \dots, n)$ 的导数为 0.

$$\begin{aligned} 0 &= \frac{\partial E}{\partial a_k} = 2 \sum_{i=1}^m w_i [P(x_i) - y_i] \frac{\partial P(x_i)}{\partial a_k} \\ &= 2 \sum_{i=1}^m w_i \left[\sum_{j=0}^n a_j \varphi_j(x_i) - y_i \right] \varphi_k(x_i) \\ &= 2 \left(\sum_{j=0}^n \left[a_j \sum_{i=1}^m w_i \varphi_j(x_i) \varphi_k(x_i) \right] - \sum_{i=1}^m w_i y_i \varphi_k(x_i) \right) \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

161

离散型广义最小二乘问题求解

因此有 $\sum_{j=0}^n \left[a_j \sum_{i=1}^m w_i \varphi_j(x_i) \varphi_k(x_i) \right] = \sum_{i=1}^m w_i y_i \varphi_k(x_i)$

引进记号 $(\varphi_j, \varphi_k) = \sum_{i=1}^m w_i \varphi_j(x_i) \varphi_k(x_i)$
 $(f, \varphi_k) = \sum_{i=1}^m w_i f(x_i) \varphi_k(x_i) = \sum_{i=1}^m w_i y_i \varphi_k(x_i)$

则上面的方程可以化为 $\sum_{j=0}^n (\varphi_j, \varphi_k) a_j = (f, \varphi_k)$
我们得到法方程组

$$\begin{bmatrix} (\varphi_0, \varphi_0) & (\varphi_0, \varphi_1) & \cdots & (\varphi_0, \varphi_n) \\ (\varphi_1, \varphi_0) & (\varphi_1, \varphi_1) & \cdots & (\varphi_1, \varphi_n) \\ \vdots & \vdots & \ddots & \vdots \\ (\varphi_n, \varphi_0) & (\varphi_n, \varphi_1) & \cdots & (\varphi_n, \varphi_n) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} (f, \varphi_0) \\ (f, \varphi_1) \\ \vdots \\ (f, \varphi_n) \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

162

连续型广义最小二乘问题求解(P53)

$$P(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \dots + a_n\varphi_n(x) \quad E = \int_a^b \rho(x)(P(x) - f(x))^2 dx$$

为求误差函数 E 的最小值,令其对 $a_k(k=0,1,\dots,n)$ 的导数为0.

$$\begin{aligned} 0 = \frac{\partial E}{\partial a_k} &= 2 \int_a^b \rho(x)(P(x) - f(x)) \frac{\partial P(x)}{\partial a_k} dx \\ &= 2 \int_a^b \rho(x) \left(\sum_{j=0}^n a_j \varphi_j(x) - f(x) \right) \varphi_k(x) dx \\ &= 2 \left(\sum_{j=0}^n a_j \int_a^b \rho(x) \varphi_j(x) \varphi_k(x) dx - \int_a^b \rho(x) f(x) \varphi_k(x) dx \right) \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

163

连续型广义最小二乘问题求解

$$\text{因此有 } \sum_{j=0}^n a_j \int_a^b \rho(x) \varphi_j(x) \varphi_k(x) dx = \int_a^b \rho(x) f(x) \varphi_k(x) dx$$

$$\text{引进记号 } (\varphi_j, \varphi_k) = \int_a^b \rho(x) \varphi_j(x) \varphi_k(x) dx$$

$$(f, \varphi_k) = \int_a^b \rho(x) f(x) \varphi_k(x) dx$$

则上面的方程可以化为 $\sum_{j=0}^n (\varphi_j, \varphi_k) a_j = (f, \varphi_k)$
我们得到法方程组

$$\begin{bmatrix} (\varphi_0, \varphi_0) & (\varphi_0, \varphi_1) & \dots & (\varphi_0, \varphi_n) \\ (\varphi_1, \varphi_0) & (\varphi_1, \varphi_1) & \dots & (\varphi_1, \varphi_n) \\ \vdots & \vdots & \ddots & \vdots \\ (\varphi_n, \varphi_0) & (\varphi_n, \varphi_1) & \dots & (\varphi_n, \varphi_n) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} (f, \varphi_0) \\ (f, \varphi_1) \\ \vdots \\ (f, \varphi_n) \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

164

内积与范数(P51)

$$\text{内积 } (f, g) = \begin{cases} \sum_{i=1}^m w_i f(x_i) g(x_i) & \text{离散型} \\ \int_a^b \rho(x) f(x) g(x) dx & \text{连续型} \end{cases}$$

$(f, g)=0$ 表示 f 与 g 正交

$$\text{范数 } (f, f) = \begin{cases} \sum_{i=1}^m w_i (f(x_i))^2 \\ \int_a^b \rho(x) (f(x))^2 dx \end{cases}$$

广义最小二乘问题可叙述为: 求广义多项式 $P(x)$ 使得 $E=(P-f, P-f)=\|P-f\|^2$ 最小.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

165

法方程组解的唯一性

$$\begin{bmatrix} (\varphi_0, \varphi_0) & (\varphi_0, \varphi_1) & \dots & (\varphi_0, \varphi_n) \\ (\varphi_1, \varphi_0) & (\varphi_1, \varphi_1) & \dots & (\varphi_1, \varphi_n) \\ \vdots & \vdots & \ddots & \vdots \\ (\varphi_n, \varphi_0) & (\varphi_n, \varphi_1) & \dots & (\varphi_n, \varphi_n) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} (f, \varphi_0) \\ (f, \varphi_1) \\ \vdots \\ (f, \varphi_n) \end{bmatrix}$$

定理 法方程组有唯一解 $\Leftrightarrow \varphi_0(x), \varphi_1(x), \dots, \varphi_n(x)$ 线性无关(参见P53定理6).

若存在一组常数使得 $\alpha_0 \varphi_0 + \alpha_1 \varphi_1 + \dots + \alpha_n \varphi_n = 0$

则等式两边分别与 $\varphi_0, \varphi_1, \dots, \varphi_n$ 作内积,得到:

$$\begin{aligned} \text{证明思路} \quad \begin{cases} \alpha_0 (\varphi_0, \varphi_0) + \alpha_1 (\varphi_1, \varphi_0) + \dots + \alpha_n (\varphi_n, \varphi_0) = 0 \\ \alpha_0 (\varphi_0, \varphi_1) + \alpha_1 (\varphi_1, \varphi_1) + \dots + \alpha_n (\varphi_n, \varphi_1) = 0 \\ \vdots \\ \alpha_0 (\varphi_0, \varphi_n) + \alpha_1 (\varphi_1, \varphi_n) + \dots + \alpha_n (\varphi_n, \varphi_n) = 0 \end{cases} \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

166

例:连续型拟合中,取

$$\varphi_j(x) = x^j, \quad \rho(x) \equiv 1, \quad y(x) \in C[0, 1]$$

$$\text{则 } (\varphi_i, \varphi_j) = \int_0^1 x^i x^j dx = \frac{1}{i+j+1} \quad \text{Hilbert阵!}$$

考虑系数矩阵为Hilbert阵的三阶方程组(P194)

$$\begin{bmatrix} 1 & 1/2 & 1/3 \\ 1/2 & 1/3 & 1/4 \\ 1/3 & 1/4 & 1/5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 11/6 \\ 13/12 \\ 47/60 \end{bmatrix} \quad \text{其精确解为}(1,1,1)^T$$

系数取三位有效数字后,方程组变为

$$\begin{bmatrix} 1.00 & 0.500 & 0.333 \\ 0.500 & 0.333 & 0.250 \\ 0.333 & 0.250 & 0.200 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1.83 \\ 1.08 \\ 0.783 \end{bmatrix}$$

其精确解为(1.0895125, 0.4879671, 1.4910028)^T

方程组病态!

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

167

$$\begin{bmatrix} (\varphi_0, \varphi_0) & (\varphi_0, \varphi_1) & \dots & (\varphi_0, \varphi_n) \\ (\varphi_1, \varphi_0) & (\varphi_1, \varphi_1) & \dots & (\varphi_1, \varphi_n) \\ \vdots & \vdots & \ddots & \vdots \\ (\varphi_n, \varphi_0) & (\varphi_n, \varphi_1) & \dots & (\varphi_n, \varphi_n) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} (f, \varphi_0) \\ (f, \varphi_1) \\ \vdots \\ (f, \varphi_n) \end{bmatrix}$$

改进

若能取函数族 $\Phi = \{ \varphi_0(x), \varphi_1(x), \dots, \varphi_n(x), \dots \}$, 使得任意一对 $\varphi_i(x)$ 和 $\varphi_j(x)$ 两两(带权)正交, 则系数矩阵就化为对角阵!

$$\text{这时直接可算出 } a_k = \frac{(f, \varphi_k)}{(\varphi_k, \varphi_k)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

168

正交化方法(补充)

将正交函数族中的 φ_k 取为 k 阶多项式, 为简单起见,可取 φ_k 的首项系数为1.

递推关系式:

$$\begin{aligned} \varphi_0(x) &= 1, \quad \varphi_1(x) = (x - \alpha_1)\varphi_0(x) \\ \varphi_{k+1}(x) &= (x - \alpha_{k+1})\varphi_k(x) - \beta_k \varphi_{k-1}(x) \end{aligned}$$

其中 $\alpha_{k+1} = \frac{(x\varphi_k, \varphi_k)}{(\varphi_k, \varphi_k)}, \quad \beta_k = \frac{(x\varphi_k, \varphi_{k-1})}{(\varphi_{k-1}, \varphi_{k-1})}$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

169

Legendre多项式(P55)

当区间为 $[-1,1]$,权函数 $\rho(x)=1$ 时,由 $\{1, x, \dots, x^n, \dots\}$ 正交化得到的多项式称为Legendre多项式.

$$P_0(x)=1 \quad P_1(x)=(x-\alpha_1)P_0(x)$$

令 $P_1(x)$ 与 $P_0(x)$ 正交,得到

$$0 = \int_{-1}^1 (x - \alpha_1) dx = 2\alpha_1 \quad \text{因此} \alpha_1=0, P_1(x)=x.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

170

Legendre多项式

$$P_2(x)=(x-\alpha_2)P_1(x)-\beta_1P_0(x)$$

分别令 $P_2(x)$ 与 $P_1(x)$ 及 $P_0(x)$ 正交,得到

$$\begin{aligned} 0 &= \int_{-1}^1 ((x - \alpha_2)(P_1(x))^2 - \beta_1 P_0(x) P_1(x)) dx \\ &= \int_{-1}^1 (x - \alpha_2) x^2 dx = -\frac{2\alpha_2}{3} \quad \text{因此} \alpha_2=0 \\ 0 &= \int_{-1}^1 (x P_1(x) P_0(x) - \alpha_2 P_1(x) P_0(x) - \beta_1 (P_0(x))^2) dx \\ &= \int_{-1}^1 (x^2 - \beta_1) dx = \frac{2}{3} - 2\beta_1 \quad \text{因此} \beta_1=1/3 \end{aligned}$$

最终得到 $P_2(x)=x^2-1/3$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

171

Legendre多项式

前几个(首一)的Legendre多项式为

$$P_0(x)=1, P_1(x)=x, P_2(x)=x^2-1/3, P_3(x)=x^3-3/5 x$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

172

连续型最小二乘的例

在 $[0,1]$ 区间上求 $f(x)=1/(1+x)$ 的最佳平方逼近二次多项式.

首先将 $[0,1]$ 转化为区间 $[-1,1]$,为此,令 $x=(t+1)/2$ (即 $t=2x-1$),则

$$F(t) = f(x) = f\left(\frac{t+1}{2}\right) = \frac{1}{1 + \frac{t+1}{2}} = \frac{2}{t+3}$$

设 $P(t)$ 为最佳平方逼近二次多项式,

$$P(t)=a_0P_0(t)+a_1P_1(t)+a_2P_2(t)$$

其中 $P_0(t), P_1(t), P_2(t)$ 为首一的Legendre多项式

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

173

连续型最小二乘的算例

$$\text{式中的系数为 } a_0 = \frac{(F(t), P_0(t))}{(P_0(t), P_0(t))} = \frac{\int_{-1}^1 \frac{2}{t+3} dt}{\int_{-1}^1 1 dt} = \ln 2 \approx 0.693147$$

$$a_1 = \frac{(F(t), P_1(t))}{(P_1(t), P_1(t))} = \frac{\int_{-1}^1 \frac{2t}{t+3} dt}{\int_{-1}^1 t^2 dt} = 6 - 9 \ln 2 \approx -0.238325$$

$$a_2 = \frac{(F(t), P_2(t))}{(P_2(t), P_2(t))} = \frac{\int_{-1}^1 \frac{2(t^2-1/3)}{t+3} dt}{\int_{-1}^1 (t^2-1/3)^2 dt} = -\frac{135}{2} + \frac{195}{2} \ln 2 \approx 0.081850$$

$$\begin{aligned} \text{因此 } P(t) &= 0.693147 - 0.238325t + 0.081850(t^2 - 1/3) \\ &= 0.081850t^2 - 0.238325t + 0.665864 \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

174

连续型最小二乘的算例

$$P(t)=0.081850t^2-0.238325t+0.665864$$

将 $t=2x-1$ 带入得到

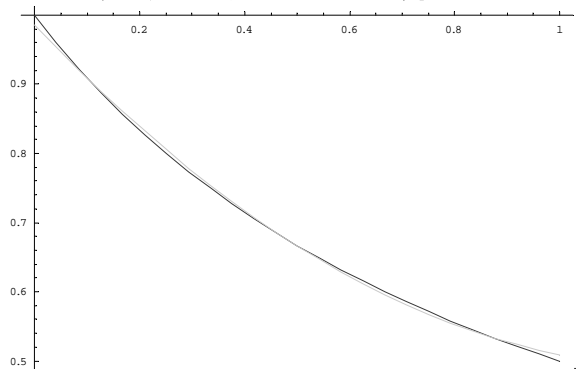
$$P(2x-1)=0.327400x^2-0.804050x+0.986039$$

这就是 $[0,1]$ 区间上 $f(x)=1/(1+x)$ 的最佳平方逼近二次多项式.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

175

连续型最小二乘的算例



76

第四章 数值积分与数值微分

§ 1 引言

依据微积分基本定理, 只要找到被积函数 $f(x)$ 的原函数 $F'(x)=f(x)$, 便有Newton-Leibniz公式 $\int_a^b f(x)dx = F(b) - F(a)$

由于大量的被积函数找不到用初等函数表示的原函数, 而实验测量或数值计算给出的通常是一张函数表, 所以Newton-Leibniz公式往往不能直接运用. 因此有必要研究积分的数值计算问题.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

178

数值求积的基本思想

依据积分中值定理, $\int_a^b f(x)dx = (b-a)f(\xi)$

就是说, 底为 $b-a$ 而高为 $f(\xi)$ 的矩形面积恰恰等于所求曲边梯形的面积.

取 $[a,b]$ 内若干个节点 x_k 处的高度 $f(x_k)$, 通过加权平均的方法生成平均高度 $f(\xi)$, 这类求积公式称机械求积公式:

$$\int_a^b f(x)dx \approx \sum_{k=0}^n A_k f(x_k)$$

式中 x_k 称为求积节点, A_k 称为求积系数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

179

代数精度的概念

数值求积方法是近似方法, 为保证精度, 自然希望所提供求积公式对于“尽可能多”的函数是准确的.

如果机械求积公式对 $f(x)=x^k (k=0, 1, \dots, m)$ 均能准确成立, 但对 $f(x)=x^{m+1}$ 不准确, 则称机械求积公式具有 m 次代数精度.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

180

例:对于 $[a, b]$ 上1次插值,有 $L_1(x) = \frac{x-b}{a-b} f(a) + \frac{x-a}{b-a} f(b)$

$$\int_a^b f(x) dx \approx \int_a^b L_1(x) dx = f(a) \int_a^b \frac{x-b}{a-b} dx + f(b) \int_a^b \frac{x-a}{b-a} dx$$

$$\Rightarrow A_1 = A_2 = \frac{b-a}{2} \Rightarrow \int_a^b f(x) dx \approx \frac{b-a}{2} [f(a) + f(b)]$$

考察其代数精度.

解: 逐次检查公式是否精确成立

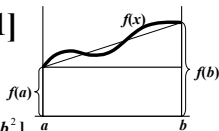
$$P_0(x) = 1, \int_a^b 1 dx = b-a = \frac{b-a}{2} [1+1]$$

$$P_1(x) = x, \int_a^b x dx = \frac{b^2-a^2}{2} = \frac{b-a}{2} [a+b]$$

$$P_2(x) = x^2, \int_a^b x^2 dx = \frac{b^3-a^3}{3} \neq \frac{b-a}{2} [a^2+b^2]$$

代数精度 = 1

梯形公式



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

181

代数精度的概念

事实上,令求积公式对 $f(x) = x^k (k=0,1,\dots,m)$ 准确成立,则有

$$\begin{cases} A_0 + A_1 + \dots + A_n = b-a \\ A_0 x_0 + A_1 x_1 + \dots + A_n x_n = (b^2 - a^2)/2 \\ \dots \\ A_0 x_0^m + A_1 x_1^m + \dots + A_n x_n^m = (b^{m+1} - a^{m+1})/(m+1) \end{cases}$$

可见,在求积公式节点给定的情况下,求积公式的构造问题本质上是解线性方程组的代数问题.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

182

插值型的求积公式

设已给 $f(x)$ 在节点 $x_k (k=0,1,\dots,n)$ 的函数值,作插值多项式

$$L_n(x) = \sum_{k=0}^n f(x_k) l_k(x) \quad \text{其中} \quad l_k(x) = \prod_{j \neq k} \frac{x-x_j}{x_k-x_j}$$

由于多项式的求积是容易的,令

$$\int_a^b f(x) dx \approx \int_a^b L_n(x) dx$$

这样得到的求积公式称为插值型的求积公式.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

183

插值型的求积公式

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

$$\int_a^b f(x) dx \approx \sum_{k=0}^n f(x_k) \int_a^b l_k(x) dx = A_k$$

$$A_k = \int_a^b \prod_{j \neq k} \frac{(x-x_j)}{(x_k-x_j)} dx$$

由节点决定, 与 $f(x)$ 无关.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

184

对于插值型的求积公式,当 $f(x)$ 为不超过 n 次的多项式时,显然有 $f(x) \equiv L_n(x)$,求积公式精确成立.

反之,若一个机械求积公式至少具有 n 次代数精度,则当 $f(x)=l_j(x)$ 时,公式精确成立.

$$\text{因此} \quad \int_a^b l_j(x) dx = \sum_{k=0}^n A_k l_j(x_k) = A_j$$

形如 $\sum_{k=0}^n A_k f(x_k)$ 的求积公式至少有 n 次代数精度 \Leftrightarrow 该公式为插值型,即: $A_k = \int_a^b l_k(x) dx$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

185

§ 2 Newton-Cotes公式

当节点等距分布时 $x_i = a + ih, h = \frac{b-a}{n}, i = 0, 1, \dots, n$

$$A_k = \int_{x_0}^{x_n} \prod_{j \neq k} \frac{(x-x_j)}{(x_k-x_j)} dx \quad \text{令} \quad x = a + th$$

$$A_k = \int_0^n \prod_{j \neq k} \frac{(t-j)}{(k-j)} \frac{h}{h} \times h dt = \frac{(b-a)}{n} \frac{(-1)^{n-k}}{k!(n-k)!} \int_0^n \prod_{j \neq k} (t-j) dt$$

Cotes系数仅取决于 n 和 k ,可查表得到,与 $f(x)$ 及区间 $[a, b]$ 均无关.

Cotes系数 $C_k^{(n)}$

$$\int_a^b f(x) dx \approx (b-a) \sum_{k=0}^n C_k^{(n)} f(x_k) \quad \text{称为Newton-Cotes公式}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

186

梯形公式

$$n=1: C_0^{(1)} = \frac{1}{2}, C_1^{(1)} = \frac{1}{2}$$

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

梯形公式

代数精度 = 1

$$R[f] = \int_a^b \frac{f''(\xi_x)}{2!} (x-a)(x-b) dx$$

$$= -\frac{1}{12} (b-a)^3 f''(\eta), \eta \in [a, b]$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

187

Simpson公式

$$n=2: C_0^{(2)} = \frac{1}{6}, C_1^{(2)} = \frac{2}{3}, C_2^{(2)} = \frac{1}{6}$$

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

Simpson公式

代数精度 = 3

$$R[f] = -\frac{b-a}{180} \left(\frac{b-a}{2}\right)^4 f^{(4)}(\eta), \eta \in (a, b)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

188

高阶公式

$n=3$: Simpson 3/8公式

$$\int_a^b f(x) dx \approx \frac{b-a}{8} [f(a) + 3f(\frac{2a+b}{3}) + 3f(\frac{a+2b}{3}) + f(b)]$$

代数精度 = 3 $R[f] = -\frac{b-a}{80} \left(\frac{b-a}{3}\right)^4 f^{(5)}(\eta)$

$n=4$: Cotes公式

$$\int_a^b f(x) dx \approx \frac{b-a}{90} [7f(a) + 32f(\frac{3a+b}{4}) + 12f(\frac{a+b}{2}) + 32f(\frac{a+3b}{4}) + 7f(b)]$$

代数精度 = 5 $R[f] = -\frac{2(b-a)}{945} \left(\frac{b-a}{4}\right)^6 f^{(6)}(\eta)$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

189

Example

The Trapezoidal rule for a function f on the interval $[0,1]$ is $\int_0^1 f(x) dx \approx f(0) + f(1)$

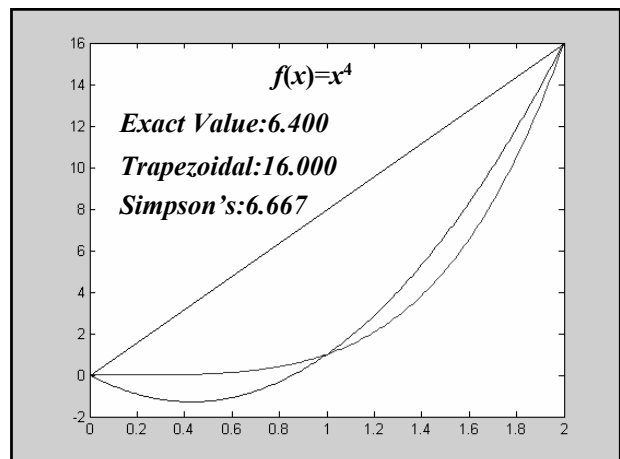
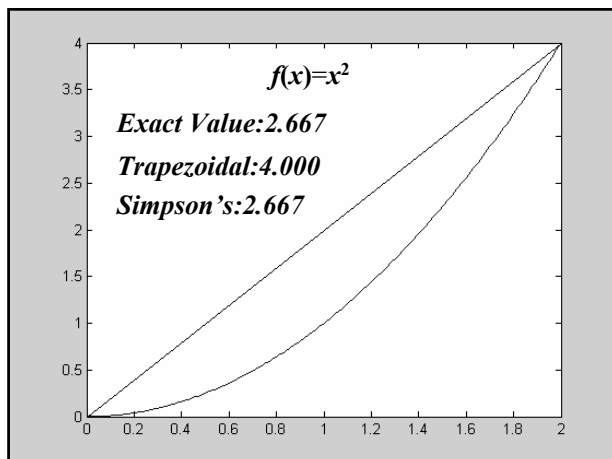
and Simpson's rule for f on $[0,2]$ is

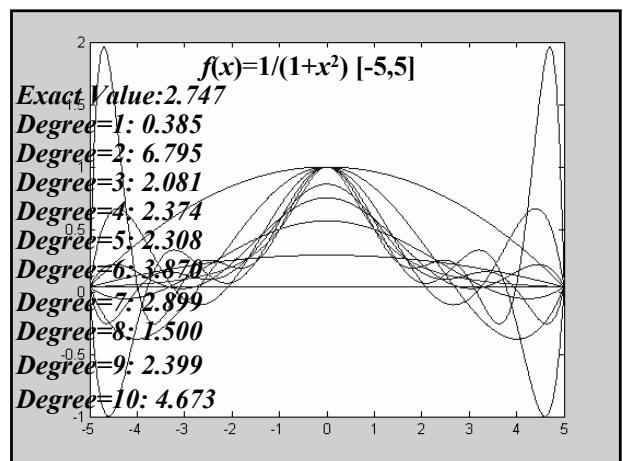
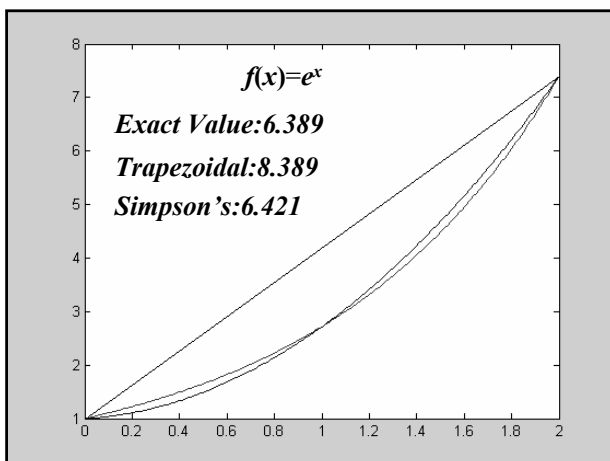
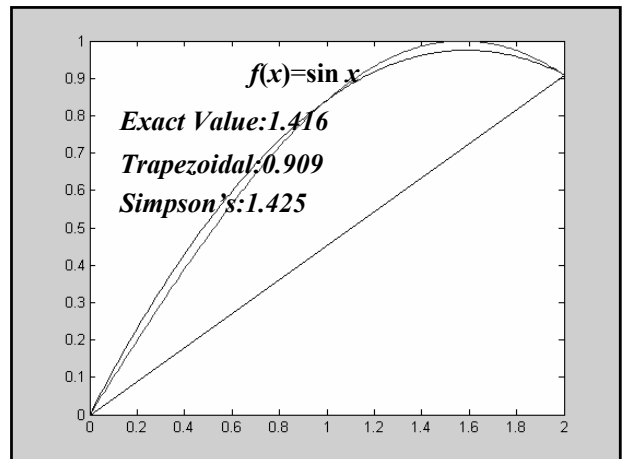
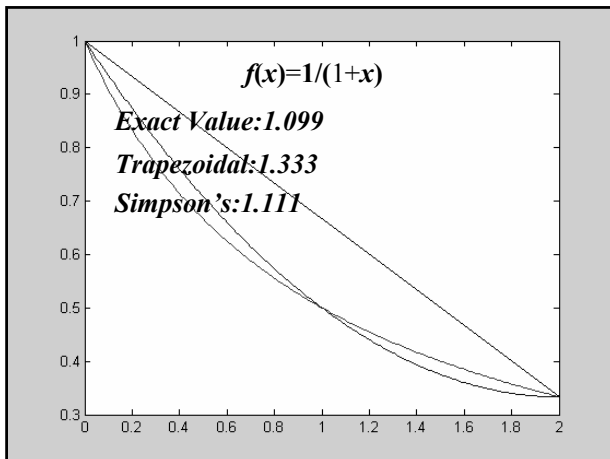
$$\int_0^2 f(x) dx \approx \frac{1}{3} [f(0) + 4f(1) + f(2)]$$

The results to three places for some elementary functions are shown in the following figures.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

190





偶阶求积公式的代数精度

对于 n 阶的Newton-Cotes公式,其至少具有 n 阶的代数精度.

分析已知的2阶公式(Simpson)和4阶公式(Cotes),可以发现它们的代数精度分别为3和5.

事实上,偶阶求积公式的代数精度都比阶数大1.

定理: n 为偶数时,Newton-Cotes公式至少具有 $n+1$ 阶代数精度.

偶阶求积公式的代数精度

证明:只需证明当 $f(x)=x^{n+1}$ 时Newton-Cotes公式的余项为0.

由Lagrange插值的余项公式,得到Newton-Cotes公式的余项为

$$R[f] = \int_a^b \prod_{j=0}^n (x - x_j) dx$$

令 $x=a+th$,由 $x_j=a+jh$ 可以得到

$$R[f] = h^{n+2} \int_a^b \prod_{j=0}^n (t - j) dx$$

偶阶求积公式的代数精度

再令 $t = u + n/2$ (注意到 n 为偶数), 有

$$\begin{aligned} R[f] &= h^{n+2} \int_{-n/2}^{n/2} \prod_{j=0}^n (u + \frac{n}{2} - j) du \\ &= h^{n+2} \int_{-n/2}^{n/2} \prod_{j=-n/2}^{n/2} (u - j) du \end{aligned}$$

由于被积函数是奇偶函数, 积分为零.
从而结论成立

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

199

复化求积公式

高次插值有Runge 现象, 故采用分段低次插值得到Newton-Cotes复化求积公式.

复化梯形公式 $h = \frac{b-a}{n}$, $x_k = a + kh$ ($k = 0, \dots, n$)

在每个 $[x_{k-1}, x_k]$ 上用梯形公式 

$$\int_{x_{k-1}}^{x_k} f(x) dx \approx \frac{x_k - x_{k-1}}{2} [f(x_{k-1}) + f(x_k)], \quad k = 1, \dots, n \quad \Rightarrow$$

$$\int_a^b f(x) dx \approx \sum_{k=1}^n \frac{h}{2} [f(x_{k-1}) + f(x_k)] = \frac{h}{2} \left[f(a) + 2 \sum_{k=1}^{n-1} f(x_k) + f(b) \right] = T_n$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

200

复化梯形公式的余项及收敛性

$$R[f] = \sum_{k=1}^n \left[-\frac{h^3}{12} f''(\xi_k) \right] = -\frac{h^2}{12} (b-a) f''(\xi), \quad \xi \in (a, b)$$

另一方面

$$\begin{aligned} \frac{R(f)}{h^2} &= -\frac{1}{12} \sum_{k=1}^n h f''(\xi_k) \rightarrow -\frac{1}{12} \int_a^b f''(x) dx \\ &= -\frac{1}{12} (f'(b) - f'(a)) \end{aligned}$$

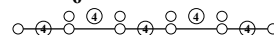
南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

201

复化Simpson公式

$$h = \frac{b-a}{n}, \quad x_k = a + kh \quad (k = 0, \dots, n) \quad \overbrace{x_k \quad x_{k+\frac{1}{2}} \quad x_{k+1}}$$

$$\int_{x_k}^{x_{k+1}} f(x) dx \approx \frac{h}{6} [f(x_k) + 4f(x_{k+\frac{1}{2}}) + f(x_{k+1})]$$



梯形法的递推化

设 T_n 表示复化梯形公式求得的积分值, 其下标 n 是等份数.

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

$$T_{2n} = \frac{h}{4} \left[f(a) + 2 \sum_{k=1}^{n-1} f(x_k) + 2 \sum_{k=1}^{n-1} f(x_{k+\frac{1}{2}}) + f(b) \right]$$

则有递推公式

$$T_{2n} = \frac{1}{2} T_n + \frac{h}{2} \sum_{k=1}^{n-1} f(x_{k+\frac{1}{2}})$$

其中

$$h = \frac{b-a}{n}, x_{k+\frac{1}{2}} = a + \left(k + \frac{1}{2}\right)h$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

205

迭代步数的判断

给定精度 ε , 如何取 n ?

要求 $|I - T_n| < \varepsilon$, 如何判断 $n = ?$

$$R(f) \approx -\frac{h^2}{12} [f'(b) - f'(a)]$$

上例中若要求 $|I - T_n| < 10^{-6}$, 则

$$|R_n[f]| \approx \frac{h^2}{12} |f'(1) - f'(0)| = \frac{h^2}{6} < 10^{-6}$$

$$\Rightarrow h < 0.00244949 \quad \text{即取 } n = 409$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

206

通常采取将区间不断对分的方法, 即取 $n = 2^k$

上例中令 $2^k \geq 409$, 得 $k = 9, n = 512, T_{512} = 3.14159202$

注意到区间再次对分时

$$R_{2n}[f] \approx -\frac{1}{12} [f'(b) - f'(a)] \left(\frac{h}{2}\right)^2 \approx \frac{1}{4} R_n[f]$$

$$S_4 = 3.141592502$$

$$\Rightarrow \frac{I - T_{2n}}{I - T_n} \approx \frac{1}{4}$$

$$\Rightarrow I - T_{2n} \approx \frac{1}{3} (T_{2n} - T_n)$$

可用来判断迭代是否停止。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

207

Romberg算法

$$\text{例: 计算 } \pi = \int_0^1 \frac{4}{1+x^2} dx$$

$\varepsilon = 10^{-6}$ 时须将区间等份9次, 得到 $T_{512} = 3.14159202$

$$\text{考察 } \frac{I - T_{2n}}{I - T_n} \approx \frac{1}{4}$$

由 $I \approx \frac{4T_{2n} - T_n}{4-1} = \frac{4}{3}T_{2n} - \frac{1}{3}T_n$ 来计算 I 效果是否好些?

$$\frac{4}{3}T_8 - \frac{1}{3}T_4 = 3.141592502 = S_4$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

208

$$\frac{4}{3}T_{2n} - \frac{1}{3}T_n = S_n?$$

$$\frac{4}{3}T_{2n} - \frac{1}{3}T_n = \frac{4}{3} \left(\frac{1}{2}T_n + \frac{h}{2} \sum_{k=1}^{n-1} f(x_{k+\frac{1}{2}}) \right) - \frac{1}{3}T_n = \frac{1}{3}T_n + \frac{2h}{3} \sum_{k=1}^{n-1} f(x_{k+\frac{1}{2}})$$

$$= \frac{h}{6} [f(a) + 2 \sum_{k=1}^{n-1} f(x_k) + f(b)] + \frac{2h}{3} \sum_{k=1}^{n-1} f(x_{k+\frac{1}{2}})$$

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

$$= S_n$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

209

Romberg算法

一般有

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

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

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

$$\textcircled{1} T_1 = T_0^{(0)}$$

$$\diamond < \varepsilon ?$$

$$\textcircled{2} T_2 = T_0^{(1)} \rightarrow \textcircled{3} S_1 = T_1^{(0)}$$

$$\diamond < \varepsilon ?$$

$$\textcircled{4} T_4 = T_0^{(2)} \rightarrow \textcircled{5} S_2 = T_1^{(1)}$$

$$\textcircled{6} C_1 = T_2^{(0)}$$

Romberg 公式

$$\textcircled{7} T_8 = T_0^{(3)} \rightarrow \textcircled{8} S_4 = T_1^{(2)}$$

$$\textcircled{9} C_2 = T_2^{(1)}$$

$$\textcircled{10} R_1 = T_3^{(0)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

210

Richardson外推加速法

目的:利用低阶公式产生高精度的结果.

设对于某一 $h \neq 0$, 有梯形公式 $T_0(h)$ 近似计算定积分值 I . 可以证明:

$$T_0(h) = I + a_1 h^2 + a_2 h^4 + a_3 h^6 + \dots + a_k h^{2k} + \dots$$

现将 h 对分, 得:

$$T_0\left(\frac{h}{2}\right) - I = a_1 \left(\frac{h}{2}\right)^2 + a_2 \left(\frac{h}{2}\right)^4 + a_3 \left(\frac{h}{2}\right)^6 + \dots$$

如何将公式精度由 $O(h^2)$ 提高?

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

211

$$T_0(h) - I = a_1 h^2 + a_2 h^4 + a_3 h^6 + \dots + a_k h^{2k} + \dots$$

$$T_0\left(\frac{h}{2}\right) - I = a_1 \left(\frac{h}{2}\right)^2 + a_2 \left(\frac{h}{2}\right)^4 + a_3 \left(\frac{h}{2}\right)^6 + \dots$$

$$\frac{4T_0\left(\frac{h}{2}\right) - T_0(h)}{4-1} - I = \beta_1 h^4 + \beta_2 h^6 + \beta_3 h^8 + \dots$$

$$T_1(h) = \frac{4}{3} T_0\left(\frac{h}{2}\right) - \frac{1}{3} T_0(h) = I + \beta_1 h^4 + \beta_2 h^6 + \dots$$

$$T_2(h) = \frac{16}{15} T_1\left(\frac{h}{2}\right) - \frac{1}{15} T_1(h) = I + \gamma_1 h^6 + \gamma_2 h^8 + \dots$$

\Rightarrow

$$T_m(h) = \frac{4^m}{4^m - 1} T_{m-1}\left(\frac{h}{2}\right) - \frac{1}{4^m - 1} T_{m-1}(h) \\ = I + \delta_1 h^{2(m+1)} + \delta_2 h^{2(m+2)} + \dots$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

212

Gaussian Quadrature

For the integral $\int_0^2 f(x) dx$, we can see that the error of Trapezoidal Rule is rather large.

However, we can choose another formula:

$$\int_0^2 f(x) dx \approx f\left(1 - \frac{\sqrt{3}}{3}\right) + f\left(1 + \frac{\sqrt{3}}{3}\right) \approx f(0.4226) + f(1.5774)$$

We only compute the values of the function twice, which is same to Trapezoidal Rule.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

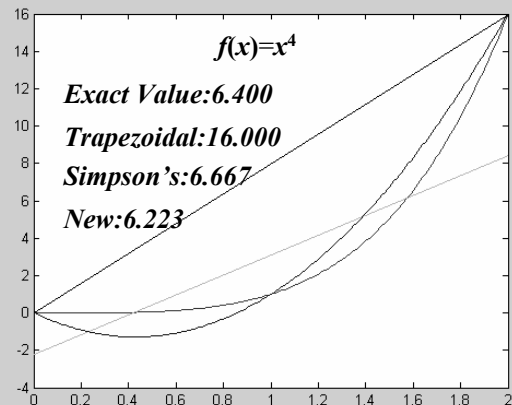
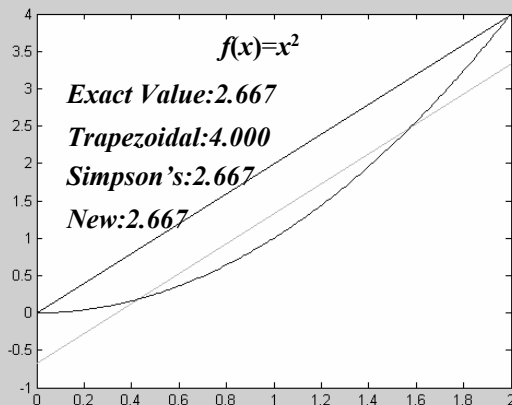
213

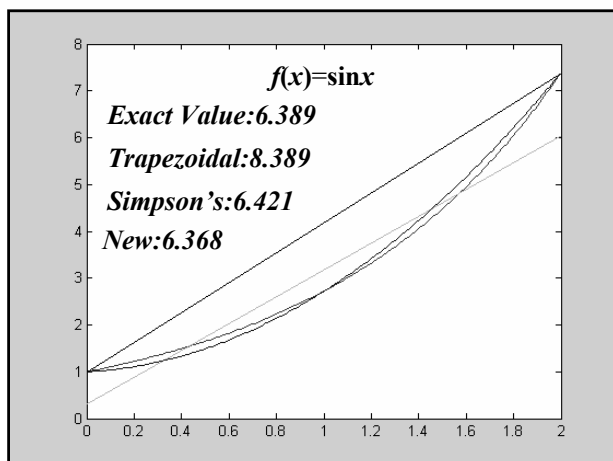
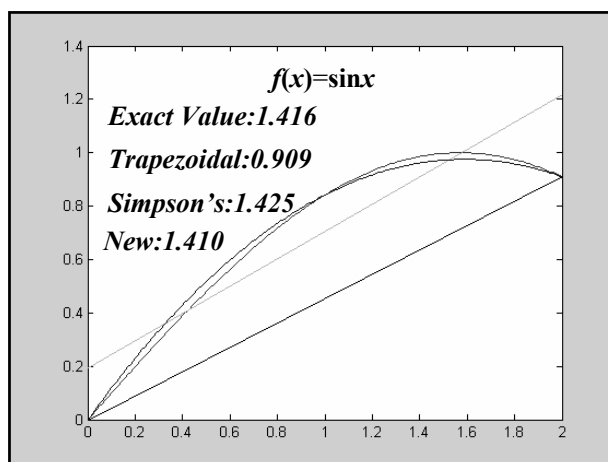
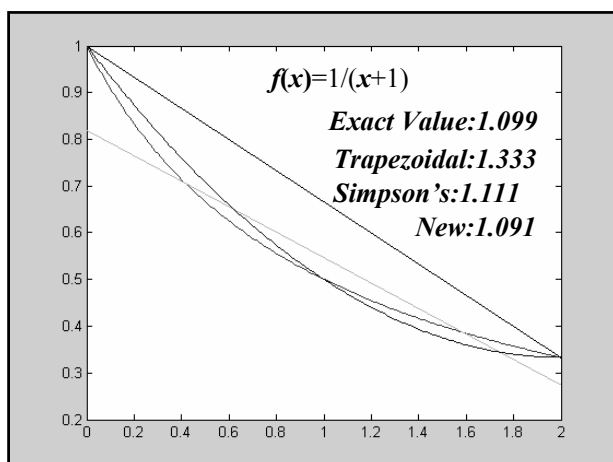
Gaussian Quadrature

$f(x)$	Exact Value	T	S	new formula
x^2	2.667	4.000	2.667(0)	2.667(0)
x^4	6.400	16.000	6.667(0.267)	6.223(0.177)
$1/(x+1)$	1.099	1.333	1.111(0.012)	1.091(0.008)
$\sqrt{1+x^2}$	2.958	3.326	2.964(0.006)	2.953(0.005)
$\sin x$	1.416	0.909	1.425(0.009)	1.410(0.006)
e^x	6.389	8.389	6.421(0.032)	6.368(0.021)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

214





§ 4 Gauss公式

$$\int_a^b f(x)dx \approx \sum_{k=0}^n A_k f(x_k)$$

若在机械求积公式中,节点待定,则有 $2n+2$ 个未知数,可以用 $2n+2$ 个等式来确定,因此可以构造具有 $2n+1$ 次代数精度的求积公式.

将节点 x_0, \dots, x_n 以及系数 A_0, \dots, A_n 都作为待定系数.令 $f(x) = 1, x, x^2, \dots, x^{2n+1}$ 代入可求解,得到的公式具有 $2n+1$ 次代数精度.这样的节点称为Gauss点,公式称为Gauss型求积公式.

220

Gauss公式

例: 求 $\int_0^1 f(x)dx$ 的 2 点高斯公式.

解: 设 $\int_0^1 f(x)dx \approx A_0 f(x_0) + A_1 f(x_1)$, 应有 3 次代数精度.

令 $f(x) = 1, x, x^2, x^3$ 时公式精确成立.

$$\begin{cases} 1 = A_0 + A_1 \\ \frac{1}{2} = A_0 x_0 + A_1 x_1 \\ \frac{1}{3} = A_0 x_0^2 + A_1 x_1^2 \\ \frac{1}{4} = A_0 x_0^3 + A_1 x_1^3 \end{cases} \Rightarrow \begin{cases} x_0 \approx 0.211325 \\ x_1 \approx 0.788675 \\ A_0 \approx 0.5 \\ A_1 \approx 0.5 \end{cases}$$

方程组非线性,不易求解.

221

Gauss公式

设 $x_0 \dots x_n$ 为Gauss点,令 $\omega(x) = \prod_{k=0}^n (x - x_k)$

考虑不超过 n 次的多项式 $P(x)$,则 $P(x)\omega(x)$ 为不超过 $2n+1$ 次的多项式.

因此 $\int_a^b P(x)\omega(x)dx = \sum_{k=0}^n A_k P(x_k)\omega(x_k) = 0$

所以 $\omega(x)$ 与任意不超过 n 次的多项式 $P(x)$ 都是正交的.

222

Gauss公式

反之,若 $\omega(x) = \prod_{k=0}^n (x - x_k)$ 与任意不超过 n 次的多项式都是正交的.

以该多项式的根为节点构造插值型积分公式.

$$\int_a^b f(x) dx \approx \sum_{k=0}^n A_k f(x_k)$$

设 $f(x)$ 为不超过 $2n+1$ 次的多项式,且

其中 $P(x), Q(x)$ 都是不超过 n 次的多项式.

用上面的公式计算 $f(x)$ 的积分.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

223

Gauss公式

$$\int_a^b f(x) dx \approx \sum_{k=0}^n A_k f(x_k) \quad f(x) = P(x)\omega(x) + Q(x)$$

$$\sum_{k=0}^n A_k f(x_k) = \sum_{k=0}^n [A_k P(x_k)\omega(x_k) + A_k Q(x_k)] = \sum_{k=0}^n A_k Q(x_k)$$

由于积分公式是插值型的, $Q(x)$ 是不超过 n 次的多项式. 因此

$$\int_a^b Q(x) dx = \sum_{k=0}^n A_k Q(x_k)$$

$$\int_a^b f(x) dx = \int_a^b P(x)\omega(x) dx + \int_a^b Q(x) dx = \int_a^b Q(x) dx$$

我们得到
$$\int_a^b f(x) dx = \sum_{k=0}^n A_k f(x_k)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

224

Gauss公式

总结上面的结论,有以下定理

定理4(P97)对于插值型求积公式

$$\int_a^b f(x) dx \approx \sum_{k=0}^n A_k f(x_k)$$

其节点 $x_k (k=0,1,\dots,n)$ 是 Gauss 点的充要条件是以这些点为零点的多项式

$$\omega(x) = \prod_{k=0}^n (x - x_k)$$

与任意次数不超过 n 的多项式 $P(x)$ 正交.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

225

正交多项式族 $\{\varphi_0, \varphi_1, \dots, \varphi_n, \dots\}$ 有性质: 任意次数不大于 n 的多项式 $P(x)$ 必与 φ_{n+1} 正交. 若取 $\omega(x)$ 为其中的 φ_{n+1} , 则 φ_{n+1} 的根就是 Gauss 点.

再解上例: $\int_0^1 f(x) dx \approx A_0 f(x_0) + A_1 f(x_1)$

构造正交多项式 $\varphi_0, \varphi_1, \varphi_2$

设 $\varphi_0(x)=1, \varphi_1(x)=x-\alpha_1$, 由正交性得到

$$0 = \int_0^1 \varphi_0(x)\varphi_1(x) dx = \int_0^1 (x-\alpha_1) dx = \frac{1}{2} - \alpha_1 \Rightarrow \alpha_1 = \frac{1}{2}$$

再设 $\varphi_2(x)=(x-\alpha_2) \varphi_1(x)-\beta_1 \varphi_0(x)$, 由正交性可得

$$\alpha_2 = \frac{1}{2} \quad \beta_1 = \frac{1}{12} \quad \varphi_2(x) = (x - \frac{1}{2})^2 - \frac{1}{12}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

226

Gauss公式

$\varphi_2(x) = (x - \frac{1}{2})^2 - \frac{1}{12}$ 的两个根为

$$x_0 = \frac{1}{2} - \frac{1}{2\sqrt{3}} \approx 0.211325 \quad x_1 = \frac{1}{2} + \frac{1}{2\sqrt{3}} \approx 0.788675$$

再根据插值型求积公式的结论容易求得求积系数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

227

Gauss-Legendre公式

Legendre 多项式是 $[-1,1]$ 上的正交多项式, 其零点可以作为 Gauss 点, 得到的 Gauss 型求积公式, 称为 Gauss-Legendre 公式.

例如, 首一的二次 Legendre 多项式为 $x^2 - 1/3$, 其根为 $-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}$

因此得到两点 Gauss-Legendre 公式

$$\int_{-1}^1 f(x) dx = A_0 f(-\frac{1}{\sqrt{3}}) + A_1 f(\frac{1}{\sqrt{3}})$$

容易求得两个系数均为 1.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

228

Gauss公式的余项

$R[f] = \int_a^b f(x)dx - \sum_{k=0}^n A_k f(x_k)$ 设 P 为 f 的过 $x_0 \dots x_n$ 的插值多项式

$$= \int_a^b f(x)dx - \sum_{k=0}^n A_k P(x_k)$$

只要 P 的阶数不大于 $2n+1$, 则下一步等式成立

$$= \int_a^b f(x)dx - \int_a^b P(x)dx = \int_a^b [f(x) - P(x)]dx$$

设 $H(x_k) = f(x_k)$, $H'(x_k) = f'(x_k)$

$$\Rightarrow R[f] = \int_a^b [f(x) - H(x)]dx$$

$$= \int_a^b \frac{f^{(2n+1)}(\xi_x)}{(2n+2)!} \omega^2(x)dx = \frac{f^{(2n+1)}(\xi)}{(2n+2)!} \int_a^b \omega^2(x)dx, \quad \xi \in (a, b)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

229

Gauss公式的稳定性

构造具有 $2n+1$ 次代数精度的求积公式

$$\int_a^b f(x)dx \approx \sum_{k=0}^n A_k f(x_k)$$

由于该公式对 $2n$ 次的多项式精确成立, 将 $f(x) = l_k^2(x)$ 代入得到

$$\int_a^b l_k^2(x)dx = \sum_{i=0}^n A_i l_k^2(x_i) = A_k > 0$$

因此上述 Gauss 公式的求积系数是正的。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

230

Gauss公式的稳定性

如果计算机计算时有舍入误差, 实际计算得到的值为 $I_n^* = \sum_{k=0}^n A_k f_k^*$

其中 f_k^* 为 $f(x_k)$ 的近似值。

如果计算时 n 选取比较大, 误差会不会也比较大?

$$|I_n - I_n^*| = \left| \sum_{k=0}^n A_k (f_k^* - f_k) \right| \quad \text{Gauss公式稳定!}$$

$$\leq \sum_{k=0}^n A_k |f_k^* - f_k| \leq \left(\sum_{k=0}^n A_k \right) \max_{0 \leq k \leq n} |f_k^* - f_k|$$

$$= (b-a) \max_{0 \leq k \leq n} |f_k^* - f_k|$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

231

§5 数值微分

导数 $f'(a)$ 是均差 $\frac{f(a+h) - f(a)}{h}$ 在 $h \rightarrow 0$ 时的极限。

因此可以取差商(均差)作为导数的近似值。

一般有以下三种方法

$$\text{向前差商 } f'(a) \approx \frac{f(a+h) - f(a)}{h}$$

$$\text{向后差商 } f'(a) \approx \frac{f(a) - f(a-h)}{h}$$

$$\text{中心差商 } f'(a) \approx \frac{f(a+h) - f(a-h)}{2h}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

232

中点方法的误差

为分析中点公式 $G(h) = \frac{f(a+h) - f(a-h)}{2h}$ 的误差, 我们将 $f(a+h)$ 及 $f(a-h)$ 在 a 处作 Taylor 展开。

$$f(a \pm h) = f(a) \pm hf'(a) + \frac{h^2}{2} f''(a) \pm \frac{h^3}{3!} f'''(a) + \frac{h^4}{4!} f^{(4)}(a) \pm \frac{h^5}{5!} f^{(5)}(a) + \dots$$

$$\text{因此有 } G(h) = f'(a) + \frac{h^2}{3!} f'''(a) + \frac{h^4}{5!} f^{(5)}(a) + \dots$$

所以, 从截断误差角度考虑, h 越小, 结果越精确。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

233

中点方法的误差

从舍入误差考虑, 当 h 很小时, $f(a+h)$ 与 $f(a-h)$ 很接近, 会造成有效数字的损失。

因此 h 不能取得太小。

例 用中点公式求 $f(x) = \sqrt{x}$ 在 $x=2$ 处的一阶导数。

$$\text{计算公式为 } g(h) = \frac{\sqrt{2+h} - \sqrt{2-h}}{2h}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

234

中点方法的误差

精确到小数点后4位计算得到的结果如右表.

可以看出,在 $h=0.1$ 是效果最好(精确值0.353553)

h 过小或过大都使得误差较大.

h	$G(h)$
1	0.3660
0.5	0.3564
0.1	0.3535
0.05	0.3540
0.01	0.3500
0.005	0.3600
0.001	0.3500
0.0005	0.4000
0.0001	0.5000

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

235

合适步长的选取(补充)

究竟以什么标准来选取合适的 h ?

在用中点公式计算时,计算所得到的结果为

$$\tilde{f}(a+h) - \tilde{f}(a-h)$$

其中 $\tilde{f}(a+h), \tilde{f}(a-h)$ 分别为 $f(a+h)$ 及 $f(a-h)$ 带舍入误差的近似值.

下面我们分析截断误差和舍入误差的总误差

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

236

合适步长的选取(补充)

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

$$\text{令 } f(a+h) = \tilde{f}(a+h) + e(a+h)$$

$$f(a-h) = \tilde{f}(a-h) + e(a-h)$$

$$\text{则 } f'(a) - \frac{\tilde{f}(a+h) - \tilde{f}(a-h)}{2h} = \frac{e(a+h) - e(a-h)}{2h} - \frac{h^2}{6} f'''(\xi)$$

设舍入误差限为 ε ,三阶导数的上界为 M ,则计算结果的误差限为

$$\varepsilon(h) = \frac{\varepsilon}{h} + \frac{h^2}{6} M$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

237

合适步长的选取(补充)

为使得误差最小,我们求误差限的最小值.

$$\varepsilon(h) = \frac{\varepsilon}{h} + \frac{h^2}{6} M = \frac{\varepsilon}{2h} + \frac{\varepsilon}{2h} + \frac{h^2}{6} M \geq 3 \sqrt[3]{\frac{\varepsilon}{2h} \cdot \frac{\varepsilon}{2h} \cdot \frac{h^2}{6} M}$$

当 $\frac{\varepsilon}{2h} = \frac{h^2}{6} M$, 即 $h = \sqrt[3]{\frac{3\varepsilon}{M}}$ 时, $\varepsilon(h)$ 取最小值

在上例中,取 $h \leq 0.31 M = \max_{x \in [1.69, 2.31]} |-\frac{1}{4}x^{-\frac{3}{2}}| = 0.1138$

$$\varepsilon = 5 \cdot 10^{-5}$$

$$\text{因此 } h^* = \sqrt[3]{\frac{3\varepsilon}{M}} = \sqrt[3]{\frac{3 \cdot 5 \cdot 10^{-5}}{0.1138}} = 0.1096$$

利用0.11为步长算得的结果为0.3536.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

238

插值型的求导公式

利用函数在节点 (x_0, x_1, \dots, x_n) 处的值 (y_0, y_1, \dots, y_n) ,作插值多项式 $P_n(x)$,再用 $P_n(x)$ 的导数值代替函数的导数值,所得的数值公式称为插值型的求导公式.

$$\text{余项 } f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \omega_{n+1}(x)$$

$$f'(x) - P'_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \omega'_{n+1}(x) + \frac{\omega_{n+1}(x)}{(n+1)!} \frac{d}{dx} f^{(n+1)}(\xi)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

239

插值型的求导公式

$$f'(x) - P'_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \omega'_{n+1}(x) + \frac{\omega_{n+1}(x)}{(n+1)!} \frac{d}{dx} f^{(n+1)}(\xi)$$

等式右端第二项中 $\frac{d}{dx} f^{(n+1)}(\xi)$ 的值一般无法估计

当 x 取节点处的值时, $\omega_{n+1}(x)=0$,第二项消失.

因此,插值型的求导公式一般只用来计算节点处的导数值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

240

两点公式

取两个点 x_0, x_1 为插值节点,插值函数为

$$P_1(x) = \frac{x-x_1}{x_0-x_1} f(x_0) + \frac{x-x_0}{x_1-x_0} f(x_1)$$

记 $x_1-x_0=h$,对上式两边求导 $P_1'(x) = \frac{1}{h}[f(x_1) - f(x_0)]$

因此 $P_1'(x_0) = \frac{1}{h}[f(x_1) - f(x_0)]$ $P_1'(x_1) = \frac{1}{h}[f(x_1) - f(x_0)]$

带余项的两点公式为

$$f_1'(x_0) = \frac{1}{h}[f(x_1) - f(x_0)] - \frac{h}{2} f''(\xi)$$

$$f_1'(x_1) = \frac{1}{h}[f(x_1) - f(x_0)] + \frac{h}{2} f''(\xi)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

241

三点公式

取三个点 x_0, x_1, x_2 为插值节点,作插值多项式 $P_2(x)$,再令 $x=x_0+th$,对 $P_2(x)$ 求导可得

$$P_2'(x_0+th) = \frac{1}{2}(t-1)(t-1)f(x_0) - t(t-2)f(x_1) + \frac{1}{2}t(t-1)f(x_2)$$

分别取 $t=0,1,2$ 再考虑余项,可以得到以下的公式

$$P_2'(x_0) = \frac{1}{2h}[-3f(x_0) + 4f(x_1) - f(x_2)] + \frac{h^2}{3} f'''(\xi)$$

$$P_2'(x_1) = \frac{1}{2h}[-f(x_0) + f(x_2)] - \frac{h^2}{6} f'''(\xi)$$

$$P_2'(x_2) = \frac{1}{2h}[f(x_0) - 4f(x_1) + 3f(x_2)] + \frac{h^2}{3} f'''(\xi)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

242

二阶导数公式

$$P_2'(x_0+th) = \frac{1}{2}(t-1)(t-1)f(x_0) - t(t-2)f(x_1) + \frac{1}{2}t(t-1)f(x_2)$$

将上式再对 t 求导一次

$$P_2''(x_0+th) = \frac{1}{h^2}[f(x_0) - 2f(x_1) + f(x_2)]$$

一般以它为 x_1 处的二阶导数公式

$$f''(x_1) = \frac{1}{h^2}[f(x_1-h) - 2f(x_1) + f(x_1+h)] - \frac{h^2}{12} f^{(4)}(\xi)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

243

作业:以二阶导数公式计算 $f(x) = \sqrt{x}$ 在 $x=2$ 处的二阶导数的近似值,若计算数据精确到小数点后4位,试求步长 h 取何值时计算误差最小.此时误差界为多少?

将此步长带入进行计算,并将实际误差与误差界进行比较.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

244

第五章 常微分方程数值解法

作业

1.用Euler公式,后退的Euler公式,梯形公式,Euler两步公式,改进的Euler公式,梯形-Euler两步预测-校正公式,4阶Runge-Kutta公式求解初值问题(步长取为0.1,计算到 y_2 为止)

$$\begin{cases} y' = x + y(0 < x < 1) \\ y(0) = 1 \end{cases}$$

2.分析改进的Euler公式与梯形公式求解上述问题的收敛性.

3.研究改进的Euler公式与梯形公式的稳定性.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

246

作业

4.对如下形式的线性多步法,在 $r=2$ 时,分别构造精度最高的显式与隐式公式.

$$y_{n+1} = \sum_{k=0}^r \alpha_k y_{n-k} + h \sum_{k=-1}^r \beta_k y'_{n-k}$$

5. 用改进的Euler方法求解下面的微分方程(步长取为1,算一步) $\begin{cases} y'' - 3y' + 2y = 0 \\ y(0) = 1, y'(0) = 1 \end{cases}$

6. P142 第14题

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

247

常微分方程数值解

许多形式的常微分方程无法给出解析解,一般要用数值的方法给出解.

一般考虑一阶常微分方程的初值问题 $\begin{cases} \frac{dy}{dx} = f(x, y) & x \in [a, b] \\ y(a) = y_0 \end{cases}$

只要 $f(x, y)$ 在 $[a, b] \times R^1$ 上连续,且关于 y 满足Lipschitz条件,即存在与 x, y 无关的常数 L 使 $|f(x, y_1) - f(x, y_2)| \leq L|y_1 - y_2|$

对任意定义在 $[a, b]$ 上的 $y_1(x)$ 和 $y_2(x)$ 都成立,则上述微分方程存在唯一解。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

248

常微分方程数值解

所谓求微分方程的数值解,实际上是要计算出解函数 $y(x)$ 在一系列节点 $a = x_0 < x_1 < \dots < x_m = b$ 处的近似值 $y_n \approx y(x_n)$

节点间距 $h_n = x_{n+1} - x_n$ 为步长,通常采用等距节点,即取 $h_n = h$ (常数).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

249

Euler公式

向前差商近似导数

$$\Rightarrow y'(x_0) \approx \frac{y(x_1) - y(x_0)}{h}$$

$$y(x_1) \approx y(x_0) + hy'(x_0) = y_0 + hf(x_0, y_0)$$

记为

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



定义在假设 $y_n = y(x_n)$,即第 n 步计算是精确的前提下,考虑的截断误差 $R_n = y(x_{n+1}) - y_{n+1}$ 称为局部截断误差。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

250

Euler公式

定义若某算法的局部截断误差为 $O(h^{p+1})$,则称该算法有 p 阶精度.

Euler法的局部截断误差:

$$\begin{aligned} R_n &= y(x_{n+1}) - y_{n+1} \\ &= [y(x_n) + hy'(x_n) + \frac{h^2}{2}y''(x_n) + O(h^3)] - [y_n + hf(x_n, y_n)] \\ &= \frac{h^2}{2}y''(x_n) + O(h^3) \end{aligned}$$

Euler法具有1阶精度.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

251

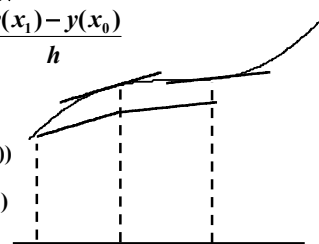
后退的Euler公式

向后差商近似导数

$$\Rightarrow y'(x_1) \approx \frac{y(x_1) - y(x_0)}{h}$$

$$y(x_1) \approx y_0 + hf(x_1, y(x_1))$$

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



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

252

后退的Euler公式

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

由于未知数 y_{n+1} 同时出现在等式的两边,不能直接得到,故称为隐式Euler公式,而前者称为显式Euler公式.

一般先用显式计算一个初值,再迭代求解.

隐式Euler法的局部截断误差:

$$R_n = y(x_{n+1}) - y_{n+1} = -\frac{h^2}{2} y''(x_n) + O(h^3)$$

即隐式Euler公式具有 1 阶精度.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

253

梯形公式

Euler公式的局部截断误差为 $-\frac{h^2}{2} y''(x_i) + O(h^3)$

后退Euler公式局部截断误差 $-\frac{h^2}{2} y''(x_i) + O(h^3)$
取平均可以得到高阶的梯形公式

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

注:的确有局部截断误差 $O(h^3)$,即梯形公式具有 2 阶精度,比Euler公式有了进步.但注意到该公式是隐式公式,计算时不得不用到迭代法.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

254

Euler两步公式

中心差商近似导数 $\longrightarrow y'(x_1) \approx \frac{y(x_2) - y(x_0)}{2h}$

$$\longrightarrow y(x_2) \approx y(x_0) + 2h f(x_1, y(x_1))$$

$$\longrightarrow y_{n+1} = y_{n-1} + 2hf(x_n, y_n)$$

假设 $y_{n-1} = y(x_{n-1}), y_n = y(x_n)$,则可以导出

$$R_n = y(x_{n+1}) - y_{n+1} = O(h^3),$$

即Euler两步公式具有 2 阶精度.

Euler两步公式的计算需要两个初值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

255

改进的Euler公式

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

考虑梯形公式的迭代求解(求解 y_{n+1}),它已经是 $y_{n+1} = \varphi(y_{n+1})$ 的形式,只需找一个初值进行迭代.

我们用显式Euler公式求出的值作为 y_{n+1} 的初值

$$y_p = y_n + hf(x_n, y_n)$$

然后用上面的公式迭代一次(不求出上面方程的精确解)

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

上述公式称为改进的Euler公式.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

256

改进的Euler公式

改进的Euler公式可以看作一个显式公式和一个隐式公式的结合.

先用显式公式作预测(predictor),给出隐式公式的一个初值.

然后用隐式公式作校正(corrector),相当于只迭代一步,得到的值作为隐式公式的近似解.

注:此法亦称为预测-校正法,可以证明该算法具有 2 阶精度,同时可以看到它是个单步递推格式,比隐式公式的迭代求解过程简单.后面将看到,它的稳定性高于显式Euler法.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

257

算例(Euler公式)

求解初值问题 $\begin{cases} y' = y - \frac{2x}{y} & (0 < x < 1) \\ y(0) = 1 \end{cases}$

解: 取步长 $h=0.1$.

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

此处的具体形式为

$$y_{n+1} = y_n + h(y_n - \frac{2x_n}{y_n})$$

该方程的精确解为 $y = \sqrt{1+2x}$

下表给出了Euler公式的计算结果及误差.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

258

算例(Euler公式)

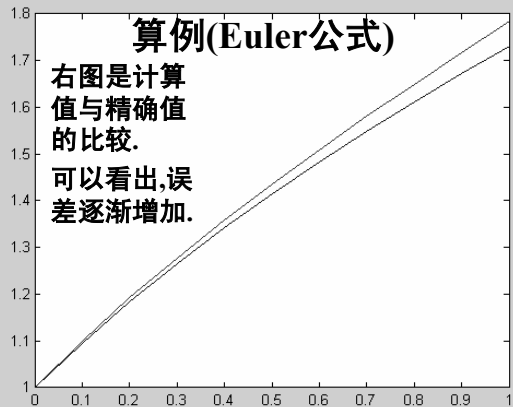
x_n	y_n	$y(x_n)$	R_n	x_n	y_n	$y(x_n)$	R_n
0.1	1.1000	1.0954	-0.0046	0.6	1.5090	1.4832	-0.0257
0.2	1.1918	1.1832	-0.0086	0.7	1.5803	1.5492	-0.0311
0.3	1.2774	1.2649	-0.0125	0.8	1.6498	1.6125	-0.0373
0.4	1.3582	1.3416	-0.0166	0.9	1.7178	1.6733	-0.0445
0.5	1.4351	1.4142	-0.0209	1.0	1.7848	1.7321	-0.0537

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

259

算例(Euler公式)

右图是计算
值与精确值
的比较。
可以看出,误
差逐渐增加。



算例(后退的Euler公式)

(2)后退的Euler公式 $y_{n+1}=y_n+h f(x_{n+1}, y_{n+1})$,
此处的具体形式为 $y_{n+1} = y_n + h(y_{n+1} - \frac{2x_{n+1}}{y_{n+1}})$

可以得到 $y_{n+1} = \frac{y_n + \sqrt{-8hx_{n+1} + 8h^2x_{n+1}^2 + y_n^2}}{2(1-h)}$

计算结果见下表

实际求解时,一般得不到 y_{n+1} 的具体表达式。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

261

算例(后退的Euler公式)

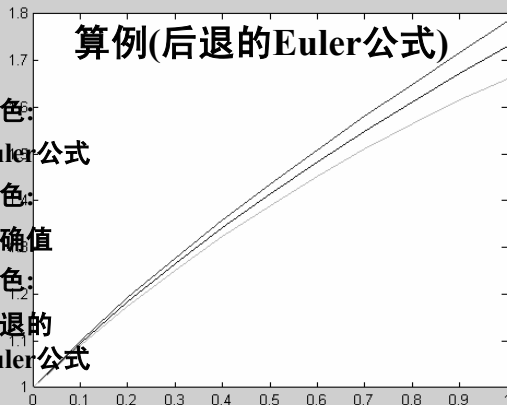
x_n	y_n	$y(x_n)$	R_n	x_n	y_n	$y(x_n)$	R_n
0.1	1.0907	1.0954	0.0046	0.6	1.4529	1.4832	0.0303
0.2	1.1741	1.1832	0.0091	0.7	1.5114	1.5492	0.0378
0.3	1.2512	1.2649	0.0137	0.8	1.5658	1.6125	0.0467
0.4	1.3231	1.3416	0.0185	0.9	1.6160	1.6733	0.0573
0.5	1.3902	1.4142	0.0240	1.0	1.6618	1.7321	0.0702

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

262

算例(后退的Euler公式)

红色:
Euler公式
蓝色:
精确值
绿色:
后退的
Euler公式



算例(梯形公式)

(3)梯形公式 $y_{n+1}=y_n+h[f(x_n, y_n)+f(x_{n+1}, y_{n+1})]/2$,
此处的具体形式为 $y_{n+1} = y_n + h(\frac{y_n}{2} + \frac{y_{n+1}}{2} - \frac{x_n}{y_n} - \frac{x_{n+1}}{y_{n+1}})$

$$y_{n+1} = \frac{-2hx_n + (2+h)y_n^2 + \sqrt{-8h(2-h)x_{n+1}y_n^2 + (2hx_n - (2+h)y_n^2)^2}}{2(2-h)y_n}$$

计算结果见下表

实际求解时,一般得不到 y_{n+1} 的具体表达式。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

264

算例(梯形公式)

x_n	y_n	$y(x_n)$	R_n	x_n	y_n	$y(x_n)$	R_n
0.1	1.0957	1.0954	-0.0002	0.6	1.4843	1.4832	-0.0010
0.2	1.1836	1.1832	-0.0004	0.7	1.5504	1.5492	-0.0012
0.3	1.2654	1.2649	-0.0005	0.8	1.6139	1.6125	-0.0015
0.4	1.3423	1.3416	-0.0007	0.9	1.6751	1.6733	-0.0018
0.5	1.4151	1.4142	-0.0008	1.0	1.7341	1.7321	-0.0021

由于误差较小,已无法用图形来显示.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

265

算例(Euler两步公式)

(4)Euler两步公式 $y_{n+1}=y_{n-1}+2hf(x_n, y_n)$,

此处的具体形式为 $y_{n+1} = y_{n-1} + 2h(y_n - \frac{2x_n}{y_n})$

取 $y_0=1.0, y_1=y(0.1)=1.0954$ (精确值)来计算

计算结果见下表

实际求解时,一般得不到 y_1 的精确值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

266

算例(Euler两步公式)

x_n	y_n	$y(x_n)$	R_n	x_n	y_n	$y(x_n)$	R_n
0.1	1.0954	1.0954	-	0.6	1.4816	1.4832	0.0017
0.2	1.1826	1.1832	0.0007	0.7	1.5473	1.5492	0.0019
0.3	1.2643	1.2649	0.0006	0.8	1.6101	1.6125	0.0024
0.4	1.3405	1.3416	0.0011	0.9	1.6706	1.6733	0.0027
0.5	1.4130	1.4142	0.0012	1.0	1.7287	1.7321	0.0034

误差与梯形公式大致相当.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

267

算例(改进的Euler公式)

(4)改进的Euler公式

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

此处的具体形式为

$$y_{n+1} = y_n + \frac{h}{2} \left[(2+h)y_n - \frac{2(1+h)x_n}{y_n} - \frac{2(x_n+h)}{y_n+h(y_n-\frac{2x_n}{y_n})} \right]$$

计算结果见下表

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

268

算例(改进的Euler公式)

x_n	y_n	$y(x_n)$	R_n	x_n	y_n	$y(x_n)$	R_n
0.1	1.0959	1.0954	-0.0005	0.6	1.4860	1.4832	-0.0027
0.2	1.1841	1.1832	-0.0009	0.7	1.5525	1.5492	-0.0033
0.3	1.2662	1.2649	-0.0013	0.8	1.6155	1.6125	-0.0040
0.4	1.3434	1.3416	-0.0017	0.9	1.6782	1.6733	-0.0048
0.5	1.4164	1.4142	-0.0022	1.0	1.7379	1.7321	-0.0058

误差比梯形公式稍大.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

269

算例(Euler两步公式-梯形公式)

(5)预测-校正公式(Euler两步公式-梯形公式)

在改进的Euler公式中,采取Euler公式进行预测,梯形公式进行校正.

但是这两个公式的精度不一致(Euler公式1阶,梯形公式2阶).下面改用Euler两步公式进行预测,而用梯形公式进行校正.

$$y_p = y_{n-1} + 2hf(x_n, y_n),$$

$$y_{n+1} = y_n + h[f(x_n, y_n) + f(x_n + h, y_p)]/2.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

270

算例(Euler两步公式-梯形公式)

x_n	y_n	$y(x_n)$	R_n	x_n	y_n	$y(x_n)$	R_n
0.1	1.0954	1.0954	-	0.6	1.4832	1.4832	0.0000
0.2	1.1833	1.1832	-0.0000	0.7	1.5490	1.5492	0.0002
0.3	1.2650	1.2649	-0.0001	0.8	1.6121	1.6125	0.0003
0.4	1.3417	1.3416	-0.0001	0.9	1.6727	1.6733	0.0006
0.5	1.4143	1.4142	-0.0000	1.0	1.7312	1.7321	0.0009

误差与梯形公式相当(此处多了一个精确值).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

271

各种方法的优劣

方法	优点	缺点
显式Euler	简单	精度低
隐式Euler	稳定性最好	精度低, 计算量大
梯形公式	精度提高	计算量大
Euler两步公式	精度提高, 显式	多一个初值, 可能影响精度
改进的Euler公式	精度有所提高, 显式	精度比梯形公式略差
梯形-Euler两步	精度最高, 显式	多一个初值

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

272

Taylor级数法

$$y(x_{n+1}) \approx y(x_n) + hy'(x_n) = y_n + hf(x_n, y_n)$$

Euler公式可以看作由1阶Taylor展式而来.

如果将Taylor展式多写一项, 可以得到

$$y(x_{n+1}) \approx y(x_n) + hy'(x_n) + \frac{h^2}{2} y''(x_n) \triangleq y_{n+1}$$

$y'(x_n)$ 可以直接得到 $y'(x_n)=f(x_n, y_n)$.

$y''(x_n)$ 是否可以计算?

$$\begin{aligned} \frac{dy}{dx} &= f(x, y) & \frac{d^2 y}{dx^2} &= \frac{d}{dx} f(x, y) = \frac{\partial f}{\partial x}(x, y) + \frac{\partial f}{\partial y}(x, y) \cdot \frac{dy}{dx} \\ & & &= \frac{\partial f}{\partial x} + f \frac{\partial f}{\partial y} \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

273

Taylor级数法

类似的可以利用高阶的Taylor级数

$$y(x_{n+1}) = y(x_n) + hy'(x_n) + \frac{h^2}{2!} y''(x_n) + \frac{h^3}{3!} y'''(x_n) + \dots$$

y 的各阶导数的求法

$$y' = f \triangleq f^{(0)} \quad y'' = \frac{\partial f^{(0)}}{\partial x} + f \frac{\partial f^{(0)}}{\partial y} \triangleq f^{(1)}$$

$$\text{一般的} \quad y^{(j)} = \frac{\partial f^{(j-2)}}{\partial x} + f \frac{\partial f^{(j-2)}}{\partial y} \triangleq f^{(j-1)}$$

若使用 p 阶的Taylor展式, 则局部截断误差为 $O(h^{p+1})$, 从而公式是 p 阶精度.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

274

算例(Taylor级数法)

求解初值问题 $\begin{cases} y' = y - \frac{2x}{y} & (0 < x < 1) \\ y(0) = 1 \end{cases}$

解: 直接求得

$$\begin{aligned} y' &= y - \frac{2x}{y} & y'' &= y' - \frac{2y - 2xy'}{y^2} = y' - \frac{2}{y} + \frac{2xy'}{y^2} \\ y''' &= y'' + \frac{2y'}{y^2} - \frac{4x(y')^2}{y^3} + \frac{2(y' + xy'')}{y^2} = y'' + \frac{4y' + 2xy''}{y^2} - \frac{4x(y')^2}{y^3} \\ y^{(4)} &= y''' + \frac{2xy'' + 6y''}{y^2} - \frac{12xy' y'' + 12(y')^2}{y^3} + \frac{12(y')^3}{y^4} \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

275

算例(Taylor级数法)

由 $x_0=0, y_0=1$, 得到

$$y'(x_0) = 1 \quad y''(x_0) = -1 \quad y'''(x_0) = 3 \quad y^{(4)}(x_0) = -15$$

取 $h=0.1$, 有

$$\begin{aligned} y_1 &= y(x_0) + hy'(x_0) + \frac{h^2}{2!} y''(x_0) + \frac{h^3}{3!} y'''(x_0) + \frac{h^4}{4!} y^{(4)}(x_0) \\ &= 1 + 0.1 + \frac{0.01}{2}(-1) + \frac{0.001}{6}3 + \frac{0.0001}{24}(-15) = 1.0954 \end{aligned}$$

下表给出了Taylor级数方法的计算结果及误差.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

276

算例(四阶Taylor级数方法)

x_n	y_n	$y(x_n)$	R_n	x_n	y_n	$y(x_n)$	R_n
0.1	1.0954	1.0954	0	0.6	1.4832	1.4832	0
0.2	1.1832	1.1832	0	0.7	1.5491	1.5492	0.0001
0.3	1.2649	1.2649	0	0.8	1.6123	1.6125	0.0002
0.4	1.3416	1.3416	0	0.9	1.6731	1.6733	0.0002
0.5	1.4142	1.4142	0	1.0	1.7318	1.7321	0.0003

误差是相当小的.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

277

Runge-Kutta方法

Taylor级数方法虽然可以达到比较高的精度,但是计算相当复杂.
是否有比较简单的算法,但是精度也比较高呢?
考虑改进的Euler公式

$$y_{n+1} = y_n + h[f(x_n, y_n) + f(x_n + h, y_n + hf(x_n, y_n))]/2,$$

它可以
改写为

$$y_{n+1} = y_n + \frac{h}{2}[K_1 + K_2]$$

$$K_1 = f(x_n, y_n), K_2 = f(x_n + h, y_n + hK_1)$$

斜率
一定取 K_1, K_2
的平均值吗?

步长一定是一个 h 吗?

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

278

将改进Euler法推广为

$$\begin{cases} y_{n+1} = y_n + h[\lambda_1 K_1 + \lambda_2 K_2] \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + ph, y_n + phK_1) \end{cases}$$

$$y''(x) = \frac{d}{dx} f(x, y)$$

$$= f_x(x, y) + f_y(x, y) \frac{dy}{dx}$$

$$= f_x(x, y) + f_y(x, y) f(x, y)$$

首先希望能确定系数 λ_1, λ_2, p ,使得到的算法格式有2阶精度,即在 $y_n = y(x_n)$ 的前提下假设下,使得

$$R_i = y(x_{i+1}) - y_{i+1} = O(h^3)$$

将 K_2 在 (x_n, y_n) 点作 Taylor 展开

$$K_2 = f(x_n + ph, y_n + phK_1)$$

$$= f(x_n, y_n) + phf_x(x_n, y_n) + phK_1f_y(x_n, y_n) + O(h^2)$$

$$= y'(x_n) + ph y''(x_n) + O(h^2)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

279

$$\begin{cases} y_{n+1} = y_n + h[\lambda_1 K_1 + \lambda_2 K_2] \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + ph, y_n + phK_1) \end{cases} \quad K_2 = y'(x_n) + ph y''(x_n) + O(h^2)$$

将 K_2 代入计算公式,得到

$$y_{n+1} = y_n + h\{\lambda_1 + y'(x_n) + \lambda_2[y'(x_n) + ph y''(x_n) + O(h^2)]\}$$

$$= y_n + [(\lambda_1 + \lambda_2)h]y'(x_n) + [\lambda_2 ph^2]y''(x_n) + O(h^3)$$

将 y_{n+1} 与 $y(x_{n+1})$ 在 x_n 点的Taylor展开作比较

$$y(x_{n+1}) = y(x_n) + [h]y'(x_n) + \frac{h^2}{2}y''(x_n) + O(h^3)$$

要求 $y(x_{n+1}) - y_{n+1} = O(h^3)$,则必须有

$$\lambda_1 + \lambda_2 = 1,$$

$$\lambda_2 p = \frac{1}{2}$$

这里有3个未知数,2个方程.

存在无穷多个解.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

280

二阶Runge-Kutta方法

$$\begin{cases} y_{n+1} = y_n + h[\lambda_1 K_1 + \lambda_2 K_2] \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + ph, y_n + phK_1) \end{cases}$$

$$\lambda_1 + \lambda_2 = 1, \quad \lambda_2 p = \frac{1}{2}$$

所有满足上式的格式统称为2阶Runge-Kutta公式.

注意到 $p=1, \lambda_1 = \lambda_2 = \frac{1}{2}$ 就是改进的Euler法.

为获得更高的精度,应该如何进一步推广?

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

281

三阶Runge-Kutta方法

为了获得更高的精度,我们多取一个点 x_{n+q}
($p \leq q \leq 1$),用三个点的斜率的线性组合来作为平均斜率,即考虑如下的公式

$$\begin{cases} y_{n+1} = y_n + h(\lambda_1 K_1 + \lambda_2 K_2 + \lambda_3 K_3) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + ph, y_n + phK_1) \\ K_3 = f(x_n + qh, y_n + qh(rK_1 + sK_2)) \end{cases}$$

我们的目的是选取合适的 $\lambda_1, \lambda_2, \lambda_3, p, q, r, s$ 使得公式的精度尽量的高.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

282

三阶Runge-Kutta方法

$$\begin{cases} y_{n+1} = y_n + h(\lambda_1 K_1 + \lambda_2 K_2 + \lambda_3 K_3) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + ph, y_n + phK_1) \\ K_3 = f(x_n + qh, y_n + qh(rK_1 + sK_2)) \end{cases}$$

根据较为复杂的推导,在参数满足下列关系时,上面的公式具有三阶精度(不可能达到四阶)

$$\begin{cases} r + s = 1 \\ \lambda_1 + \lambda_2 + \lambda_3 = 1 \\ \lambda_2 p + \lambda_3 q = 1/2 \\ \lambda_2 p^2 + \lambda_3 q^2 = 1/3 \\ \lambda_3 pqs = 1/6 \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

283

四阶Runge-Kutta方法

通过类似的方法可以得到下面常用的四阶Runge-Kutta公式

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

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

284

算例(四阶Runge-Kutta方法)

再次考虑问题 $\begin{cases} y' = y - \frac{2x}{y} (0 < x < 1) \\ y(0) = 1 \end{cases}$

令 $h=0.2$,用四阶Runge-Kutta公式求解此问题. 具体迭代格式为

具体计算结果见下表.

$$\begin{cases} y_{n+1} = y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \\ K_1 = y_n - \frac{2x_n}{y_n} \\ K_2 = y_n + \frac{h}{2}K_1 - \frac{2x_n + h}{y_n + hK_1/2} \\ K_3 = y_n + \frac{h}{2}K_2 - \frac{2x_n + h}{y_n + hK_2/2} \\ K_4 = y_n + hK_3 - \frac{2(x_n + h)}{y_n + hK_3} \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

285

算例(四阶Runge-Kutta方法)

x_n	y_n	$y(x_n)$
0.2	1.1832	1.1832
0.4	1.3417	1.3416
0.6	1.4833	1.4832
0.8	1.6125	1.6125
1.0	1.7321	1.7321

可以看出,在小数点后4位几乎没有误差.

和改进的Euler公式相比,四阶的Runge-Kutta公式每步的计算量是两倍.

计算到 $x=1$ 为止,总的计算量是一致的.但是四阶Runge-Kutta公式在 $x=1$ 处的误差要小得多.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

286

Runge-Kutta的主要运算在于计算 K_i 的值,即计算 f 的值. Butcher 于1965年给出了计算量与可达到的最高精度阶数的关系:

每步须算 K_i 的个数	2	3	4	5	6	7	$n \geq 8$
可达到的最高精度	$O(h^2)$	$O(h^3)$	$O(h^4)$	$O(h^5)$	$O(h^6)$	$O(h^7)$	$O(h^{n-2})$

由于Runge-Kutta方法的导出基于Taylor展开,故精度主要受解函数的光滑性影响.对于光滑性不太好的解,最好采用低阶算法而将步长 h 取小.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

287

单步法的收敛性

定义 若某算法对于任意固定的 $x_n = x_0 + nh$, 当 $h \rightarrow 0$ (同时 $n \rightarrow \infty$) 时有 $y_n \rightarrow y(x_n)$, 则称该算法是收敛的.

例:就初值问题 $\begin{cases} y' = \lambda y \\ y(0) = y_0 \end{cases}$ 考察Euler公式的收敛性.

解: 该问题的精确解为 $y(x) = y_0 e^{\lambda x}$

Euler公式为 $y_{n+1} = y_n + h\lambda y_n = (1 + \lambda h)y_n \Leftrightarrow y_n = (1 + \lambda h)^n y_0$

对任意固定的 $x_n = nh$, 有

$$y_n = y_0 (1 + \lambda h)^{x_n/h} = y_0 [(1 + \lambda h)^{1/\lambda h}]^{\lambda x_n}$$

$$\rightarrow y_0 e^{\lambda x_n} = y(x_n) \quad \checkmark$$

$$\lim_{h \rightarrow 0} (1 + \lambda h)^{1/\lambda h} = e$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

288

单步法的收敛性

关于单步法,有下面的收敛性定理.

定理1 假设单步法 $y_{n+1}=y_n+h\varphi(x_n, y_n, h)$ 具有 p 阶精度,且增量函数 $\varphi(x, y, h)$ 关于 y 满足Lipschitz条件

$$|\varphi(x, y, h) - \varphi(x, \bar{y}, h)| \leq L_\varphi |y - \bar{y}|$$

又设初值 y_0 是准确的,即 $y_0=y(x_0)$,则整体截断误差

$$y(x_n) - y_n = O(h^p)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

289

单步法的收敛性

由于前面所研究的方法最低是1阶的,根据上述定理,若增量函数关于 y 满足Lipschitz条件,则方法收敛.

$$|\varphi(x, y, h) - \varphi(x, \bar{y}, h)| \leq L_\varphi |y - \bar{y}|$$

增量函数关于 y 的偏导数有界,则上面的条件一定成立.

在实际应用中,有时用这一增强的条件.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

290

单步法的稳定性

例考察初值问题 $\begin{cases} y'(x) = -30y(x) \\ y(0) = 1 \end{cases}$ 在区间 $[0, 0.5]$ 上的解.分别用Euler显,隐式格式和改进的Euler格式计算数值解。

节点 x_n	Euler显式	Euler隐式	改进Euler法	精确解 $y = e^{-30x}$
0.0	1.0000	1.0000	1.0000	1.0000
0.1	-2.0000	2.5000×10^{-1}	2.5000	4.9787×10^{-2}
0.2	4.0000	6.2500×10^{-2}	6.2500	2.4788×10^{-3}
0.3	-8.0000	1.5625×10^{-2}	1.5626×10^1	1.2341×10^{-4}
0.4	1.6000×10^1	3.9063×10^{-3}	3.9063×10^1	6.1442×10^{-6}
0.5	-3.2000×10^1	9.7656×10^{-4}	9.7656×10^1	3.0590×10^{-7}

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

291

定义若某算法在节点 y_n 上大小为 δ 的扰动(误差)于以后各节点值 $y_m(m>n)$ 上产生的偏差均不超过 δ ,则称该算法是稳定的.

一般分析时,只考虑模型方程 $y' = \lambda y (\lambda < 0)$

将Euler公式用于模型方程有

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

设 y_n 上有扰动 ε_n , $y_n^* = y_n + \varepsilon_n$,假设Euler公式计算时不再产生新的扰动(不考虑截断误差),则有

$$\begin{aligned} y_{n+1} + \varepsilon_{n+1} &= y_{n+1}^* = (1 + h\lambda)y_n^* = (1 + h\lambda)(y_n + \varepsilon_n) \\ &= y_{n+1} + (1 + h\lambda)\varepsilon_n \quad \text{因此} \quad \varepsilon_{n+1} = (1 + h\lambda)\varepsilon_n \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

292

$$y_{n+1} = (1 + h\lambda)y_n \quad \varepsilon_{n+1} = (1 + h\lambda)\varepsilon_n$$

$$|\varepsilon_{n+1}| \leq |\varepsilon_n| \quad \text{的充要条件为} \quad |y_{n+1}| \leq |y_n|$$

此处,上述条件在 h 充分小时成立 $|1 + h\lambda| \leq 1$

这说明Euler公式是条件稳定的,稳定性条件为

$$h \leq -\frac{2}{\lambda}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

293

再考虑后退的Euler公式

对于模型方程,后退的Euler公式为

$$y_{n+1} = y_n + h\lambda y_{n+1} \quad y_{n+1} = \frac{1}{1 - h\lambda} y_n$$

类似的可以说明

$$|\varepsilon_{n+1}| \leq |\varepsilon_n| \quad \text{的充要条件为} \quad |y_{n+1}| \leq |y_n|$$

由于 $\lambda < 0$,这时恒成立 $\frac{1}{1 - h\lambda} < 1$

这说明后退的Euler公式是无条件稳定的.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

294

线性多步法

Euler两步公式

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

Euler两步公式每次只需计算一个函数(f)值,但其精度是两阶的。

即,该公式精度与改进的Euler公式一致,但计算量减少了一半。

一般的多步法都有这一优点。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

295

基于数值积分的构造方法

前面的方法都是基于数值微分或者Taylor级数.数值积分公式也可以构造常微分方程数值解法.

将方程 $y'=f(x,y)$ 在区间 $[x_n, x_{n+1}]$ 上进行积分,得

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f(x, y(x)) dx$$

对上面的积分采用数值的方法进行近似计算就可以得到 $y(x_{n+1})$ 的近似值。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

296

基于数值积分的构造方法

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f(x, y(x)) dx$$

左矩形公式

$$\int_{x_n}^{x_{n+1}} f(x, y(x)) dx \approx hf(x_n, y_n)$$

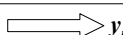


$$y_{n+1} = y_n + hf(x_n, y_n)$$

Euler公式

右矩形公式

$$\int_{x_n}^{x_{n+1}} f(x, y(x)) dx \approx hf(x_{n+1}, y_{n+1})$$



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

后退的Euler公式

梯形公式

$$\int_{x_n}^{x_{n+1}} f(x, y(x)) dx \approx \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, y_{n+1})]$$



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

梯形公式

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

297

基于数值积分的构造方法

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f(x, y(x)) dx$$

如果直接在 $[x_n, x_{n+1}]$ 上利用已知的数值积分公式,一般要利用区间内部的点(除了前面的三个公式),而这样又要计算函数 f 的值。

为例如避免这一情况,我们借用区间外部的点.若已知 $(x_n, f_n), (x_{n-1}, f_{n-1}), \dots, (x_{n-r}, f_{n-r})$,构造插值多项式近似代替 $f(x, y(x))$,得到数值积分,进一步可以得到 $y(x_{n+1})$ 的近似值。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

298

Adams显式公式

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} P_r(x) dx$$

已知 $(x_n, f_n), (x_{n-1}, f_{n-1}), \dots, (x_{n-r}, f_{n-r})$,构造插值多项式 $P_r(x)$ (利用Newton后插公式, $t = (x - x_n)/h$):

$$P_r(x_n + th) = \sum_{j=0}^r (-1)^j \binom{-t}{j} \Delta^j f_{n-j}$$

将 $P_r(x)$ 表达式带入前面的公式,得到下面的Adams显式公式

$$y_{n+1} = y_n + h \sum_{j=0}^r \alpha_j \Delta^j f_{n-j} \quad \text{或} \quad y_{n+1} = y_n + h \sum_{i=0}^r \beta_i f_{n-i}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

299

Adams显式公式

对 $(x_n, f_n), (x_{n-1}, f_{n-1}), (x_{n-2}, f_{n-2})$ 三个数据点构造的多项式为

$$\begin{aligned} P_2(x_n + th) &= f_n + t(f_n - f_{n-1}) + \frac{t(t+1)}{2}(f_n - 2f_{n-1} + f_{n-2}) \\ \int_{x_n}^{x_{n+1}} P_r(x) dx &= h \int_0^1 [f_n + t(f_n - f_{n-1}) + \frac{t(t+1)}{2}(f_n - 2f_{n-1} + f_{n-2})] dt \\ &= h [f_n + \frac{1}{2}(f_n - f_{n-1}) + \frac{5}{12}(f_n - 2f_{n-1} + f_{n-2})] \\ &= h [\frac{23}{12}f_n - \frac{16}{12}f_{n-1} + \frac{5}{12}f_{n-2}] \end{aligned}$$

因此有 $y_{n+1} = y_n + h [\frac{23}{12}f_n - \frac{16}{12}f_{n-1} + \frac{5}{12}f_{n-2}]$

其它各阶公式的系数见书中表5-7.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

300

Adams隐式公式

如果在构造插值多项式时,考虑数据点 (x_{n+1}, f_{n+1}) ,
则得到如下形式的Adams隐式公式

$$y_{n+1} = y_n + h \sum_{i=0}^r \beta_i f_{n-i+1}$$

$r=0$ 时得到的是后退的Euler公式, $r=1$ 时得到的是梯形公式.

$r=2$ 时得到公式 $y_{n+1} = y_n + h \left[\frac{5}{12} f_{n+1} + \frac{8}{12} f_n - \frac{1}{12} f_{n-1} \right]$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

301

Adams公式的截断误差

考虑Adams显式公式

$$y_{n+1} = y_n + \frac{h}{24} [55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}]$$

若 $y_{n-k}=y(x_{n-k})(k=0,1,2,3)$,则 $f_{n-k}=y'(x_{n-k})$,带入上式

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

将右端各项在 x_n 处进行Taylor展开,可以得到

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

302

Adams公式的截断误差

$$\begin{aligned} y_{n+1} &= y(x_n) + \frac{h}{24} [55y'(x_n) - 59y'(x_{n-1}) + 37y'(x_{n-2}) - 9y'(x_{n-3})] \\ y'(x_{n-1}) &= y'(x_n) - hy''(x_n) + \frac{h^2}{2} y'''(x_n) - \frac{h^3}{6} y^{(4)}(x_n) + \frac{h^4}{24} y^{(5)}(x_n) - \dots \\ y'(x_{n-2}) &= y'(x_n) - 2hy''(x_n) + \frac{4h^2}{2} y'''(x_n) - \frac{8h^3}{6} y^{(4)}(x_n) + \frac{16h^4}{24} y^{(5)}(x_n) - \dots \\ y'(x_{n-3}) &= y'(x_n) - 3hy''(x_n) + \frac{9h^2}{2} y'''(x_n) - \frac{27h^3}{6} y^{(4)}(x_n) + \frac{81h^4}{24} y^{(5)}(x_n) - \dots \\ y_{n+1} &= y(x_n) + hy'(x_n) + \frac{h^2}{2} y''(x_n) + \frac{h^3}{6} y'''(x_n) + \frac{h^4}{24} y^{(4)}(x_n) - \frac{49}{144} y^{(5)}(x_n) + \dots \\ y(x_{n+1}) &= y(x_n) + hy'(x_n) + \frac{h^2}{2} y''(x_n) + \frac{h^3}{6} y'''(x_n) + \frac{h^4}{24} y^{(4)}(x_n) + \frac{h^5}{120} y^{(5)}(x_n) + \dots \end{aligned}$$

$$y(x_{n+1}) - y_{n+1} \approx \frac{251}{720} h^5 y^{(5)}(x_n)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

303

Adams公式的截断误差

$$y_{n+1} = y_n + \frac{h}{24} [55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}]$$

$$y(x_{n+1}) - y_{n+1} \approx \frac{251}{720} h^5 y^{(5)}(x_n)$$

四阶公式

$$y_{n+1} = y_n + \frac{h}{24} [9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2}]$$

$$y(x_{n+1}) - y_{n+1} \approx -\frac{19}{720} h^5 y^{(5)}(x_n)$$

四阶公式

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

304

Adams预测-校正系统

Adams隐式公式一般不用迭代法求解,而是用显式公式进行预测之后,用它来进行校正,即有如下的Adams预测-校正系统

$$\begin{aligned} \text{预测} \quad \bar{y}_{n+1} &= y_n + \frac{h}{24} [55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}] \\ \bar{f}_{n+1} &= f(x_{n+1}, \bar{y}_{n+1}) \end{aligned}$$

$$\begin{aligned} \text{校正} \quad y_{n+1} &= y_n + \frac{h}{24} [9\bar{f}_{n+1} + 19f_n - 5f_{n-1} + f_{n-2}] \\ f_{n+1} &= f(x_{n+1}, y_{n+1}) \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

305

算例(Adams四阶预测-校正系统)

$$\text{求解初值问题} \begin{cases} y' = y - \frac{2x}{y} \quad (0 < x < 1) \\ y(0) = 1 \end{cases}$$

解 取 $h=0.1$,前三个初值 y_1, y_2, y_3 采用四阶Runge-Kutta方法计算.

计算结果见下表.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

306

算例(四阶Adams预测-校正系统)

x_n	y_n (预测)	y_n (校正)	精确值	$R_n(A)$	$R_n(R-K)$
0.1		1.09544553	1.09544512	-4.17	-4.17
0.2		1.18321675	1.18321596	-7.89	-7.89
0.3		1.26491223	1.26491106	-11.64	-11.64
0.4	1.34155176	1.34164136	1.34164079	-5.71	-15.67
0.5	1.41416568	1.41421383	1.41421356	-2.71	-20.16

表中误差的结果应乘以 10^{-7}

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

307

算例(四阶Adams预测-校正系统)

x_n	y_n (预测)	y_n (校正)	精确值	$R_n(A)$	$R_n(R-K)$
0.6	1.48321206	1.48323982	1.48323970	-1.27	-25.25
0.7	1.54917625	1.54919338	1.54919334	-0.42	-31.14
0.8	1.61244047	1.61245154	1.61245155	0.13	-38.00
0.9	1.67331257	1.67332000	1.67332005	0.54	-46.06
1.0	1.73204556	1.73205072	1.73205081	0.88	-55.58

表中误差的结果应乘以 10^{-7}

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

308

对于此例,可以看出,Adams的稳定性比R-K方法好,
 计算到 $x=5$ 时,四阶Adams预测-校正方法的
 误差为 1.88×10^{-4} ,
 而四阶Runge-Kutta的误差为 -93.36×10^{-4} .
 另外,其计算量也小(每步函数值只计算两次,
 而Runge-Kutta方法计算四次).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

309

基于Taylor展开的方法

在Adams方法中, y_{n+1} 的计算都是用 y_n 加上 f 的
 函数的线性组合.

在Euler两步方法中, y_{n+1} 的计算是用 y_{n-1} 加上 f
 的函数的线性组合.

事实上, y_{n+1} 的计算可以采用如下形式.

$$y_{n+1} = \sum_{k=0}^r \alpha_k y_{n-k} + h \sum_{k=-1}^r \beta_k y'_{n-k}$$

当 $\beta_{-1}=0$ 时是显式公式,否则是隐式公式.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

310

基于Taylor展开的方法

$$y_{n+1} = \sum_{k=0}^r \alpha_k y_{n-k} + h \sum_{k=-1}^r \beta_k y'_{n-k}$$

对于这种形式的公式,一般采用Taylor展开的
 方法确定其中的系数.

假设 $y_{n-k} = y(x_{n-k}), y'_{n-k} = y'(x_{n-k})$

然后在 x_n 处进行Taylor展开,就可以确定待定
 系数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

311

基于Taylor展开的方法

考察 $r=1$ 时的公式.

$$\begin{aligned}
 y_{n+1} &= y(x_{n+1}) = y_n + hy'_n + \frac{h^2}{2} y''_n + \frac{h^3}{6} y'''_n + \frac{h^4}{24} y^{(4)}_n + \frac{h^5}{120} y^{(5)}_n + \dots \\
 &= \alpha_0 y_n + \alpha_1 y_{n-1} + h(\beta_{-1} y'_{n+1} + \beta_0 y'_n + \beta_1 y'_{n-1}) \\
 &= \alpha_0 y_n + \alpha_1 y_n - hy'_n + \frac{h^2}{2} y''_n - \frac{h^3}{6} y'''_n + \frac{h^4}{24} y^{(4)}_n - \frac{h^5}{120} y^{(5)}_n + \dots \\
 &\quad + hy'_n + h^2 y''_n + \frac{h^3}{2} y'''_n + \frac{h^4}{6} y^{(4)}_n + \frac{h^5}{24} y^{(5)}_n + \dots \\
 &\quad + hy'_n - h^2 y''_n + \frac{h^3}{2} y'''_n - \frac{h^4}{6} y^{(4)}_n + \frac{h^5}{24} y^{(5)}_n + \dots
 \end{aligned}$$

$$1 = \alpha_0 + \alpha_1$$

$$1 = -\alpha_1 + \beta_{-1} + \beta_0 + \beta_1$$

$$1 = \alpha_1 + 2\beta_{-1} - 2\beta_1$$

$$1 = -\alpha_1 + 3\beta_{-1} + 3\beta_1$$

$$1 = \alpha_1 + 4\beta_{-1} - 4\beta_1$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

312

基于Taylor展开的方法

如果要求公式有4阶精度,则右边等式成立

解得

$$\alpha_0 = 0, \alpha_1 = 1, \beta_{-1} = \frac{1}{3}, \beta_0 = \frac{4}{3}, \beta_1 = \frac{1}{3}$$

得到如下公式

$$y_{n+1} = y_{n-1} + \frac{h}{3}(y'_{n+1} + 4y'_n + y'_{n-1})$$

$$1 = \alpha_0 + \alpha_1$$

$$1 = -\alpha_1 + \beta_{-1} + \beta_0 + \beta_1$$

$$1 = \alpha_1 + 2\beta_{-1} - 2\beta_1$$

$$1 = -\alpha_1 + 3\beta_{-1} + 3\beta_1$$

$$1 = \alpha_1 + 4\beta_{-1} - 4\beta_1$$

Simpson公式

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

313

基于Taylor展开的方法

若要求是显式公式,则 $\beta_{-1}=0$,

划去右边最后一个等式

解得

$$\alpha_0 = -4, \alpha_1 = 5, \beta_{-1} = 0, \beta_0 = 4, \beta_1 = 2$$

得到如下公式

$$y_{n+1} = -4y_n + 5y_{n-1} + h(4y'_n + 2y'_{n-1})$$

该公式具有3阶精度

$$1 = \alpha_0 + \alpha_1$$

$$1 = -\alpha_1 + \beta_{-1} + \beta_0 + \beta_1$$

$$1 = \alpha_1 + 2\beta_{-1} - 2\beta_1$$

$$1 = -\alpha_1 + 3\beta_{-1} + 3\beta_1$$

$$1 = \alpha_1 + 4\beta_{-1} - 4\beta_1$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

314

$$\begin{aligned} 1 &= \alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 \\ 1 &= -\alpha_1 - 2\alpha_2 - 3\alpha_3 + \beta_0 + \beta_1 + \beta_2 + \beta_3 \\ 1 &= \alpha_1 + 4\alpha_2 + 9\alpha_3 - 2\beta_1 - 4\beta_2 - 6\beta_3 \\ 1 &= -\alpha_1 - 8\alpha_2 - 27\alpha_3 + 3\beta_1 + 12\beta_2 + 27\beta_3 \\ 1 &= \alpha_1 + 16\alpha_2 + 81\alpha_3 - 4\beta_1 - 32\beta_2 - 108\beta_3 \end{aligned}$$

$$\begin{aligned} y_n &= y_n \\ y_{n-1} &= y_n - hy'_n + \frac{h^2}{2} y''_n - \frac{h^3}{6} y'''_n + \frac{h^4}{24} y^{(4)}_n - \frac{h^5}{120} y^{(5)}_n + \dots \\ y_{n-2} &= y_n - 2hy'_n + \frac{4h^2}{2} y''_n - \frac{8h^3}{6} y'''_n + \frac{16h^4}{24} y^{(4)}_n - \frac{32h^5}{120} y^{(5)}_n + \dots \\ y_{n-3} &= y_n - 3hy'_n + \frac{9h^2}{2} y''_n - \frac{27h^3}{6} y'''_n + \frac{81h^4}{24} y^{(4)}_n - \frac{243h^5}{120} y^{(5)}_n + \dots \\ hy'_n &= hy'_n \\ hy'_{n-1} &= hy'_n - h^2 y''_n + \frac{h^3}{2} y'''_n - \frac{h^4}{6} y^{(4)}_n + \frac{h^5}{24} y^{(5)}_n + \dots \\ hy'_{n-2} &= hy'_n - 2h^2 y''_n + \frac{4h^3}{2} y'''_n - \frac{8h^4}{6} y^{(4)}_n + \frac{16h^5}{24} y^{(5)}_n + \dots \\ hy'_{n-3} &= hy'_n - 3h^2 y''_n + \frac{9h^3}{2} y'''_n - \frac{27h^4}{6} y^{(4)}_n + \frac{81h^5}{24} y^{(5)}_n + \dots \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

315

基于Taylor展开的方法

考察 $r=3$ 时的显式公式.

$$\begin{aligned} y_{n+1} &= y(x_{n+1}) = y_n + hy'_n + \frac{h^2}{2} y''_n + \frac{h^3}{6} y'''_n + \frac{h^4}{24} y^{(4)}_n + \frac{h^5}{120} y^{(5)}_n + \dots \\ y_n &= y_n \\ y_{n-1} &= y_n - hy'_n + \frac{h^2}{2} y''_n - \frac{h^3}{6} y'''_n + \frac{h^4}{24} y^{(4)}_n - \frac{h^5}{120} y^{(5)}_n + \dots \\ y_{n-2} &= y_n - 2hy'_n + \frac{4h^2}{2} y''_n - \frac{8h^3}{6} y'''_n + \frac{16h^4}{24} y^{(4)}_n - \frac{32h^5}{120} y^{(5)}_n + \dots \\ y_{n-3} &= y_n - 3hy'_n + \frac{9h^2}{2} y''_n - \frac{27h^3}{6} y'''_n + \frac{81h^4}{24} y^{(4)}_n - \frac{243h^5}{120} y^{(5)}_n + \dots \\ hy'_n &= hy'_n \\ hy'_{n-1} &= hy'_n - h^2 y''_n + \frac{h^3}{2} y'''_n - \frac{h^4}{6} y^{(4)}_n + \frac{h^5}{24} y^{(5)}_n + \dots \\ hy'_{n-2} &= hy'_n - 2h^2 y''_n + \frac{4h^3}{2} y'''_n - \frac{8h^4}{6} y^{(4)}_n + \frac{16h^5}{24} y^{(5)}_n + \dots \\ hy'_{n-3} &= hy'_n - 3h^2 y''_n + \frac{9h^3}{2} y'''_n - \frac{27h^4}{6} y^{(4)}_n + \frac{81h^5}{24} y^{(5)}_n + \dots \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

316

基于Taylor展开的方法

如果要求公式有4阶精度,则下列等式成立

$$1 = \alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 \quad \alpha_0 = 1, \beta_0 = \frac{55}{24}, \beta_1 = -\frac{59}{24}$$

$$1 = -\alpha_1 - 2\alpha_2 - 3\alpha_3 + \beta_0 + \beta_1 + \beta_2 + \beta_3 \quad \beta_2 = \frac{37}{24}, \beta_3 = -\frac{9}{24}$$

$$1 = \alpha_1 + 4\alpha_2 + 9\alpha_3 - 2\beta_1 - 4\beta_2 - 6\beta_3$$

$$1 = -\alpha_1 - 8\alpha_2 - 27\alpha_3 + 3\beta_1 + 12\beta_2 + 27\beta_3$$

$$1 = \alpha_1 + 16\alpha_2 + 81\alpha_3 - 4\beta_1 - 32\beta_2 - 108\beta_3$$

该方程组有8个未知数,5个式子的Adams公式.

令 $\alpha_1=\alpha_2=\alpha_3=0$,得

根据Taylor展式,可以得到余项公式.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

317

Milne公式

$$1 = \alpha_0 + \alpha_1 + \alpha_2 + \alpha_3$$

$$1 = -\alpha_1 - 2\alpha_2 - 3\alpha_3 + \beta_0 + \beta_1 + \beta_2 + \beta_3 \quad \text{此时得到公式}$$

$$1 = \alpha_1 + 4\alpha_2 + 9\alpha_3 - 2\beta_1 - 4\beta_2 - 6\beta_3 \quad \text{为Milne公式}$$

$$1 = -\alpha_1 - 8\alpha_2 - 27\alpha_3 + 3\beta_1 + 12\beta_2 + 27\beta_3$$

$$1 = \alpha_1 + 16\alpha_2 + 81\alpha_3 - 4\beta_1 - 32\beta_2 - 108\beta_3$$

令 $\alpha_0=\alpha_1=\alpha_2=0$,得

$$\alpha_3 = 1, \beta_0 = \frac{8}{3}, \beta_1 = -\frac{4}{3}$$

$$\beta_2 = \frac{8}{3}, \beta_3 = 0$$

$$y_{n+1} = y_{n-3} + \frac{4h}{3}(2y'_n - y'_{n-1} + 2y'_{n-2})$$

Milne公式常与Simpson公式构成预测-校正系统.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

318

Hamming公式

类似的方法可以推出隐式的Adams公式,及常用的Hamming公式(该公式稳定性较好)

$$y_{n+1} = \frac{1}{8}(9y_n - y_{n-2}) + \frac{3h}{8}(y'_{n+1} + 2y'_n - y'_{n-1})$$

Milne公式可与Hamming公式构成预测-校正系统.

事实上 $r=3$ 时,可以选择系数达到更高的精度,不过在实际应用中,一般最多选取4阶的公式.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

319

方程组的求解

考察一阶微分方程组 初始条件为

$$\begin{cases} y'_1 = f_1(x, y_1, y_2, \dots, y_N) \\ y'_2 = f_2(x, y_1, y_2, \dots, y_N) \\ \dots \\ y'_N = f_N(x, y_1, y_2, \dots, y_N) \end{cases} \begin{cases} y_1(x_0) = y_1^0 \\ y_2(x_0) = y_2^0 \\ \dots \\ y_N(x_0) = y_N^0 \end{cases}$$

采用向量的记号

$$y = (y_1, y_2, \dots, y_N)^T; y^0 = (y_1^0, y_2^0, \dots, y_N^0)^T, f = (f_1, f_2, \dots, f_N)^T$$

上面的方程组可以记为 $\begin{cases} y' = f(x, y) \\ y(x_0) = y_0 \end{cases}$

求解方程的各种方法都可以用来求解方程组.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

320

高阶方程的求解

考察 m 阶微分方程 初始条件为

$$y^{(m)} = f(x, y, y', \dots, y^{(m-1)}) \begin{cases} y(x_0) = y_0 \\ y'(x_0) = y'_0 \\ \dots \\ y^{(m-1)}(x_0) = y_0^{(m-1)} \end{cases}$$

引进新的变量

$$y_1 = y, y_2 = y', \dots, y_m = y^{(m-1)}$$

则 m 阶方程可以化为一阶方程组

$$\begin{cases} y'_1 = y_2 \\ y'_2 = y_3 \\ \dots \\ y'_{m-1} = y_m \\ y'_m = f(x, y_1, y_2, \dots, y_m) \end{cases} \quad \text{再求解该方程组即可求得原微分方程的解.}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

321

算例(高阶方程)

用4阶Runge-Kutta方法求解方程.

$$\begin{cases} y'' = 3y' - 2y + x + 1, 0 < x \leq 10 \\ y(0) = 2, y'(0) = 1 \end{cases}$$

解: 设 $y_1=y, y_2=y'$, 则有 再设 $Y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}, Y_0 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$

$$\begin{cases} y'_1 = y_2 \\ y'_2 = -2y_1 + 3y_2 + x + 1 \end{cases} \quad f(x, Y) = \begin{pmatrix} y_2 \\ -2y_1 + 3y_2 + x + 1 \end{pmatrix}$$

$$\begin{cases} y_1(0) = 2 \\ y_2(0) = 1 \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

322

算例(高阶方程)

$$Y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}, Y_0 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, f(x, Y) = \begin{pmatrix} y_2 \\ -2y_1 + 3y_2 + x + 1 \end{pmatrix}$$

Runge-Kutta公式为 取 $h=0.2$, 第一步计算过程为

$$Y_{n+1} = Y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4)$$

其中 $K_1 = f(x_n, Y_n)$

$$K_2 = f(x_n + \frac{h}{2}, Y_n + \frac{h}{2}K_1)$$

$$K_3 = f(x_n + \frac{h}{2}, Y_n + \frac{h}{2}K_2)$$

$$K_4 = f(x_n + h, Y_n + hK_3)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

323

算例(高阶方程)

x_n	Y_n	$y(x_n)$
0.2	(2.1984667, 0.9755333)	(2.1984466, 0.97549041)
0.4	(2.3855009, 0.8791838)	(2.3854395, 0.87905423)
0.6	(2.5422297, 0.6623530)	(2.5420896, 0.66206034)
0.8	(2.6375660, 0.2496112)	(2.6372828, 0.24902472)
1.0	(2.6215527, -0.4751457)	(2.6210178, -0.47624622)

该方程的精确解为 $y(x) = \frac{1}{4}(5 + 4e^x - e^{2x} + 2x)$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

324

边值问题

考虑二阶微分方程 $y''=f(x,y,y')$,

若定解条件由 $y(a)=\alpha, y'(a)=m$ 给出,称为初值问题.

在实际应用中,若在区间 $[a,b]$ 上求解上述方程,常给出边界条件 $y(a)=\alpha, y(b)=\beta$,称为边值问题.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

325

试射法

对于常微分方程边值问题,一种比较初等的方法是试射法.先猜测出 $y'(a)$ 值的下界 m_1 ,上界 m_2 ,然后分别在 $y'(a)=m_1$ 与 $y'(a)=m_2$ 的条件下求解初值问题的微分方程.

获得 $y(b)$ 的两个值后,与已知的 $y(b)$ 的值进行比较.然后再反复调整,得到最终的解.

这一方法显然有很大的局限性.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

326

线性常微分方程的差分方程方法

下面主要研究线性方程 $y''+p(x)y'+q(x)y=r(x)$ 的解法.

取一定的步长 h ,将导数用差商去近似.

$$y_n'' \approx \frac{y_{n+1} - 2y_n + y_{n-1}}{h^2}, \quad y_n' \approx \frac{y_{n+1} - y_{n-1}}{2h},$$

得到以下的方程

$$\frac{y_{n+1} - 2y_n + y_{n-1}}{h^2} + p_n \frac{y_{n+1} - y_{n-1}}{2h} + q_n y_n = r_n \quad (n = 1, 2, \dots, N-1)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

327

线性常微分方程的差分方程方法

$$\frac{y_{n+1} - 2y_n + y_{n-1}}{h^2} + p_n \frac{y_{n+1} - y_{n-1}}{2h} + q_n y_n = r_n \quad (n = 1, 2, \dots, N-1)$$

将边界条件 $y_0=\alpha, y_N=\beta$ 代入,可以得到方程组

$$\begin{cases} (-2 + h^2 q_1) y_1 + (1 + \frac{h}{2} p_1) y_2 = h^2 r_1 - (1 - \frac{h}{2} p_1) \alpha \\ (1 - \frac{h}{2} p_{n-1}) y_{n-1} + (-2 + h^2 q_n) y_n + (1 + \frac{h}{2} p_n) y_{n+1} = h^2 r_n \quad (n = 2, 3, \dots, N-2) \\ (1 - \frac{h}{2} p_{N-1}) y_{N-1} + (-2 + h^2 q_{N-1}) y_{N-1} = h^2 r_{N-1} - (1 + \frac{h}{2} p_{N-1}) \beta \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

328

线性常微分方程的差分方程方法

前面的方程组的系数矩阵为三对角矩阵

$$\begin{bmatrix} -2 + h^2 q_1 & 1 + \frac{h}{2} p_1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 - \frac{h}{2} p_2 & -2 + h^2 q_2 & 1 + \frac{h}{2} p_2 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 - \frac{h}{2} p_3 & -2 + h^2 q_3 & 1 + \frac{h}{2} p_3 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 - \frac{h}{2} p_{N-2} & -2 + h^2 q_{N-2} & 1 + \frac{h}{2} p_{N-2} \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 - \frac{h}{2} p_{N-1} & -2 + h^2 q_{N-1} \end{bmatrix}$$

可用追赶法求解方程组.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

329

算例(边值问题)

用差分方法求解边值问题 解 取步长 $h=0.2$, 节点

$$\begin{cases} y'' - y = x, 0 < x < 1 \\ y(0) = 0, y(1) = 1 \end{cases} \quad x_n = 0, 0.2, 0.4, 0.6, 0.8, 1.0.$$

得到的方程组为

$$\begin{bmatrix} -2.04 & 1 & 0 & 0 \\ 1 & -2.04 & 1 & 0 \\ 0 & 1 & -2.04 & 1 \\ 0 & 0 & 1 & -2.04 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 0.008 \\ 0.016 \\ 0.024 \\ -0.968 \end{bmatrix}$$

解此方程组的到的解见下表.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

330

算例(边值问题)

x_n	y_n	$y(x_n)$	$R_n(10^{-4})$
0.2	0.142811	0.142641	-1.70
0.4	0.299335	0.299033	-3.02
0.6	0.483832	0.483480	-3.52
0.8	0.711682	0.711411	-2.71

该方程的精确解为 $y(x) = \frac{2(e^x - e^{-x})}{e - e^{-1}} - x$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

331

本章小结

建立常微分方程(初值)数值公式的四种方法:

数值微分 \Rightarrow Euler方法 后退的Euler方法 Euler两步方法

数值积分 \Rightarrow 梯形方法 Adams方法 Simpson方法

Taylor级数 \Rightarrow Runge-Kutta方法 Hamming方法 Milne方法

预测-校正 \Rightarrow 改进的Euler方法 梯形-Euler两步

Adams系统 Milne-Hamming

Simpson-Hamming

边值问题的方法 差分方法

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

332

本章小结

各种方法的比较

Taylor级数方法:具有一般性,不利于应用

Runge-Kutta方法:精度高,计算量大,适用于光滑性好的函数

Euler,改进Euler:计算量小,精度底,适用于光滑性差的函数

多步方法:计算量小,必须用其它方法提供初值

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

333

第六章 方程求根

作业

一.教材P164第9题

二.分别有Newton法与改进的Newton法求解方程 $1 - \cos x = 0$. (取初值 $x_0 = 0.5$, 计算到 x_2)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

335

作业

三.用下列方法求 $f(x) = x^3 - 2x - 1 = 0$ 在区间 $[1, 2]$ 内的根.

(1)二分法(精确到小数点后2位)

(2)Newton法($x_0 = 1$, 求 x_2)

(3)弦截法($x_0 = 1, x_1 = 2$, 求 x_3)

(4)抛物线法($x_0 = 1, x_1 = 2, x_2$ 为弦截法中 x_2 , 求 x_3)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

336

问题的提出

实际应用中很多问题常常归结为解方程 $f(x)=0$.

这里 $f(x)$ 是单变量 x 的函数,它可以是代数多项式,也可以是超越函数.

满足方程 $f(x)=0$ 的 x 值通常叫做方程的根或解,也叫函数 $f(x)$ 的零点.

如果 $f(x)=(x-\alpha)^m g(x)$ 且 $g(\alpha) \neq 0$,则称 α 为 $f(x)=0$ 的 m 重根.

$m=1$ 称为单根, $m>1$ 称为重根.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

337

求根步骤

方程的求根问题一般分两步进行:

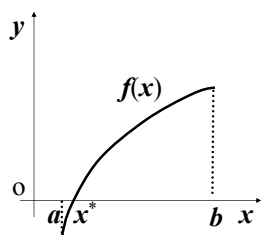
1.根的搜索 确定根所在的区间,使方程在这个小区间内有且仅有一个根,这一过程称为根的搜索,做好根的搜索工作,就可以获得方程各个根的近似值.

2.根的精确化 已知一个根的近似值后,再用一种方法把此近似值精确化,使其满足给定的精度要求.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

338

设函数 $f(x)$ 在 $[a, b]$ 内连续, 严格单调, 且有 $f(a)f(b) < 0$, 则在 $[a, b]$ 内方程 $f(x) = 0$ 有且仅有一个实根.



$f(a)f(b) < 0 \rightarrow$ 至少一个根

严格单调 \rightarrow 最多一个根

设函数 $f(x)$ 在 $[a, b]$ 上连续, 且 $f(a)f(b) < 0$, 这时称 $[a, b]$ 是 $f(x)=0$ 的有根区间.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

339

有根区间的搜索

有根区间的搜索可以采用如下方法:

(1)作 $y=f(x)$ 的草图,由 $y=f(x)$ 与横轴交点的大致位置来确定根的搜索区间.

(2)逐步搜索,在连续区间 $[a, b]$ 内,选择一系列的 x 值(等步长或不等步长),当出现两个相邻点上函数值反号时,在此小区间内至少有一个实根.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

340

二分法

二分法的基本思想:

用对分区间的方法,通过判别函数 $f(x)$ 的符号,逐步将有根区间缩小,使在足够小的区间内,方程有且仅有一根.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

341

二分法

考察 $f(x)$ 的有根区间 $[a, b]$,用中点 $x_0=(a+b)/2$ 将区间分成两半,分别考虑以下三种情况:

(1) $f(x_0)=0$,此时 x_0 是所求的根;

(2) $f(x_0)$ 与 $f(a)$ 异号,此时令 $a_1=a, b_1=x_0$,得到有根区间 $[a_1, b_1]$,其长度是原来区间的一半;

(3) $f(x_0)$ 与 $f(b)$ 异号,此时令 $a_1=x_0, b_1=b$,也得到有根区间 $[a_1, b_1]$,长度也是原区间的一半.

注:由于 $f(a)$ 和 $f(b)$ 异号,如果 $f(x_0) \neq 0$, (2)(3)两种情况必居其一.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

342

上述的过程可以一直进行下去,或者求出了根,或者求出一系列的有根区间:

$$[a, b] \supset [a_1, b_1] \supset [a_2, b_2] \supset \cdots \supset [a_k, b_k] \supset \cdots$$

如果我们不考虑中间求出根的情况,由于根 x^* 在区间 $[a_k, b_k]$ 内,令 $x_k = (a_k + b_k)/2$,我们有

$$|x^* - x_k| \leq (b_k - a_k)/2 = (b - a)/2^{k+1}$$

只要二分足够多次(k 足够大),就有

$$|x^* - x_k| < \varepsilon, \text{ 其中 } \varepsilon \text{ 是预先设定的精度.}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

343

二分法的计算步骤

step1准备计算 $f(x)$ 在有根区间 $[a, b]$ 端点处的值

step2二分计算 $f(x)$ 在区间中点处的函数值

step3判断若 $f(\frac{a+b}{2})=0$,则 $\frac{a+b}{2}$ 即是根,计算过程结束,否则检验:

•若 $f(\frac{a+b}{2})$ 与 $f(a)$ 异号,则根位于区间 $[a, \frac{a+b}{2}]$ 内,以 $\frac{a+b}{2}$ 代替 b ;

•若 $f(\frac{a+b}{2})$ 与 $f(a)$ 同号,则根位于区间 $[\frac{a+b}{2}, b]$ 内,以 $\frac{a+b}{2}$ 代替 a ;

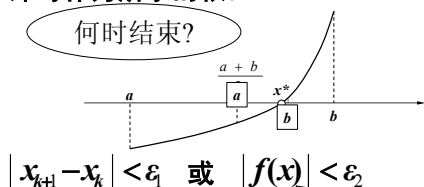
南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

344

二分法的计算步骤

反复执行step2和step3,直到区间 $[a, b]$ 长度缩小到允许误差范围之内,此时区间中点即可作为所求的根。

何时结束?



$$|x_{k+1} - x_k| < \varepsilon_1 \quad \text{或} \quad |f(x)| < \varepsilon_2$$

不能保证 x 的精度

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

345

例2 求方程 $f(x)=x^3-x-1=0$ 在区间 $[1.0, 1.5]$ 内的一个实根,精确到小数点后第二位。

解:这里 $a=1.0, b=1.5$,而 $f(a)<0, f(b)>0$,取区间的中点 $x_0=1.25$,由于 $f(x_0)<0$,根在 $[x_0, b]$ 内,逐渐二分得到下面的表。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

346

例2 求方程 $f(x)=x^3-x-1=0$ 在区间 $[1.0, 1.5]$ 内的一个实根,精确到小数点后第二位。

k	a_k	b_k	x_k
0	1.0(-)	1.5(+)	1.25(-)
1	1.25(-)		1.375(+)
2		1.375(+)	1.3125(-)
3	1.3125(-)		1.3438(+)
4		1.3438(+)	1.3281(+)
5		1.3281(+)	1.3203(-)
6	1.3203(-)		1.3242(-)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

347

例2 求方程 $f(x)=x^3-x-1=0$ 在区间 $[1.0, 1.5]$ 内的一个实根,精确到小数点后第二位。

k	a_k	b_k	x_k
0	1.0(-)	1.5(+)	1.25(-)

最终我们求得根的近似之为 $x_6=1.3242$,满足 $|x^*-x_6|<0.005$

3	1.3125(-)		1.3438(+)
4		1.3438(+)	1.3281(+)
5		1.3281(+)	1.3203(-)
6	1.3203(-)		1.3242(-)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

348

误差分析

第0步产生的 x_1 , 有误差 $|x_0 - x^*| \leq \frac{b-a}{2}$

第 k 步产生的 x_k 有误差 $|x_k - x^*| \leq \frac{b-a}{2^{k+1}}$

对于给定的精度 ε , 可估计二分法所需的步数 k :

$$\frac{b-a}{2^{k+1}} \leq \varepsilon \Rightarrow k \geq \left[\frac{\ln(b-a) - \ln \varepsilon}{\ln 2} \right] - 1$$



1. 计算简单, 方法可靠;
2. 对 $f(x)$ 要求不高(只要连续即可);
3. 收敛性总能得到保证。



1. 无法求复根及偶重根
2. 收敛慢

一般常用来为其它方法求近似根时提供初值。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

349

迭代法

迭代法的基本思想:

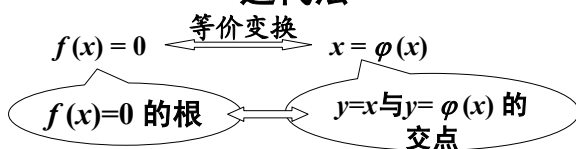
迭代法是一种逐次逼近的方法, 首先给定一个粗糙的初始值, 然后用同一个迭代公式, 反复校正这个初值, 直到满足预先给定的精度要求为止。

迭代法的关键在于如何构造一个合适的迭代公式。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

350

迭代法



从一个初值 x_0 出发, 计算 $x_1 = \phi(x_0)$, $x_2 = \phi(x_1)$, ..., $x_{k+1} = \phi(x_k)$, ... 若 $\{x_k\}_{k=0}^{\infty}$ 收敛, 即存在 x^* 使得 $\lim_{k \rightarrow \infty} x_k = x^*$ 若 ϕ 连续, 则由 $\lim_{k \rightarrow \infty} x_{k+1} = \lim_{k \rightarrow \infty} \phi(x_k)$ 可知 $x^* = \phi(x^*)$, 则 x^* 即为 $x = \phi(x)$ 的根, 也就是 $f(x) = 0$ 的根。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

351

例3 求方程 $f(x)=x^3-x-1=0$ 在 $x_0=1.5$ 附近的根 x^*

将方程改写为 $x=(x+1)^{1/3}$. 得到迭代公式

$$x_{k+1} = (x_k + 1)^{1/3}.$$

根据 $x_0=1.5$ 得到

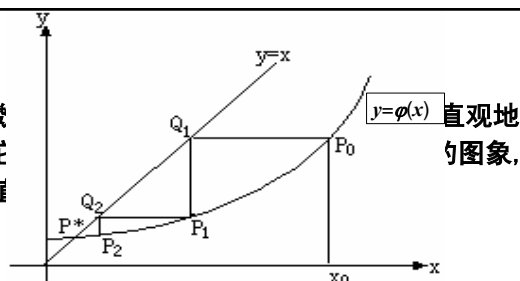
$$x_1=1.35721, \dots, x_7=1.32472, x_8=1.32472$$

由于 x_7 和 x_8 已经前六位相等, 终止迭代。我们求得方程的近似根为1.32472。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

352

对函数
显示它
对初值



过 P_0 引平行 x 轴的直线交直线 $y=x$ 于点 Q_1 , 再过 Q_1 作平行 y 轴的直线, 它与曲线的交点记为 P_1 , 重复上面的过程就得到一系列点 P_1, P_2, \dots , 这些点逐步趋向于曲线 $y=\phi(x)$ 与直线 $y=x$ 的交点 P^* 。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

353

迭代图形

对函数的迭代过程, 我们可以用几何图象来直观地显示它。在平面上, 先作出函数 $y=\phi(x)$ 与 $y=x$ 的图象, 对初值 x_0 , 在曲线上可确定一点 P_0 。

过 P_0 引平行 x 轴的直线交直线 $y=x$ 于点 Q_1 , 再过 Q_1 作平行 y 轴的直线, 它与曲线的交点记为 P_1 , 重复上面的过程就得到一系列点 P_1, P_2, \dots , 这些点逐步趋向于曲线 $y=\phi(x)$ 与直线 $y=x$ 的交点 P^* 。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

354

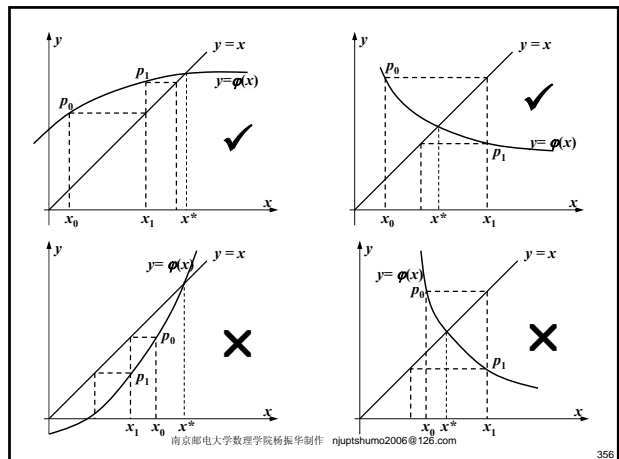
迭代法的收敛性

对于前面的方程 $f(x)=x^3-x-1=0$, 如果取迭代公式 $x_{k+1}=x_k^3-1$, 仍然取初值 $x_0=1.5$, 则有 $x_1=2.375, x_2=12.39, x_3=1901, \dots$, 显然迭代是发散的.

下面的图形给出了迭代收敛和发散的几种情况.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

355



356

迭代法的收敛性

定理考虑方程 $x = \varphi(x)$, $\varphi(x) \in C[a, b]$, 若

(I) 当 $x \in [a, b]$ 时, $\varphi(x) \in [a, b]$;

(II) 存在 $0 \leq L < 1$ 使得 $|\varphi'(x)| \leq L < 1$ 对任意 $x \in [a, b]$ 成立.

则迭代过程 $x_{k+1} = \varphi(x_k)$ 对于任意初值 $x_0 \in [a, b]$ 均收敛于 $x = \varphi(x)$ 的根 x^* , 且有误差估计式:

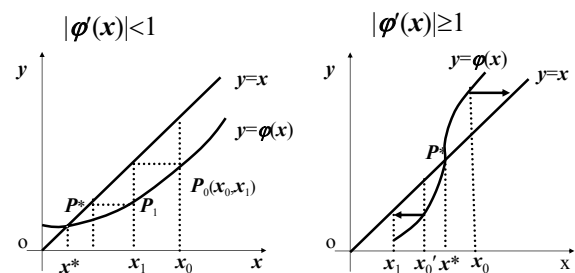
$$\textcircled{1} \quad |x^* - x_k| \leq \frac{L^k}{1-L} |x_1 - x_0|$$

$$\textcircled{2} \quad |x^* - x_k| \leq \frac{1}{1-L} |x_{k+1} - x_k|$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

357

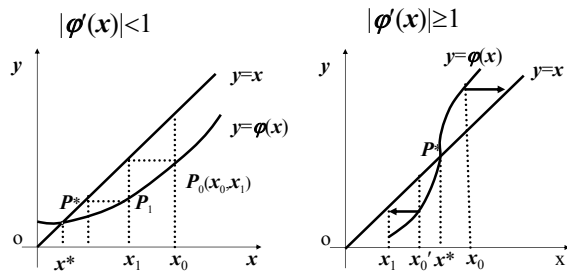
迭代法的收敛性



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

358

迭代法的收敛性



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

359

证明: ① $f(x)$ 在 $[a, b]$ 上有根?

令 $f(x) = \varphi(x) - x$ $\because a \leq \varphi(x) \leq b$

$\therefore f(a) = \varphi(a) - a \geq 0, f(b) = \varphi(b) - b \leq 0$

② 根唯一? $\Rightarrow f(x)$ 有根 \checkmark

反证: 若不然, 设还有 $\tilde{x} = \varphi(\tilde{x})$

则 $x^* - \tilde{x} = \varphi(x^*) - \varphi(\tilde{x}) = \varphi'(\xi)(x^* - \tilde{x})$,

$\Rightarrow (x^* - \tilde{x})(1 - \varphi'(\xi)) = 0$ 而 $|\varphi'(\xi)| < 1 \therefore x^* = \tilde{x} \checkmark$

③ 当 $k \rightarrow \infty$ 时, x_k 收敛到 x^* ?

$|x^* - x_k| = |\varphi(x^*) - \varphi(x_{k-1})| = |\varphi'(\xi_{k-1})| \cdot |x^* - x_{k-1}|$

$\leq L |x^* - x_{k-1}| \leq \dots \leq L^k |x^* - x_0| \rightarrow 0 \checkmark$

$L < 1$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

360

④ $|x_k - x^*| \leq \frac{1}{1-L} |x_{k+1} - x_k|$? 可用 $|x_{k+1} - x_k|$ 来控制收敛精度

$|x_k - x^*| = |x_{k+1} - x^* + x_k - x_{k+1}| \leq |x_{k+1} - x^*| + |x_k - x_{k+1}| \leq L|x_k - x^*| + |x_{k+1} - x_k|$ ✓

⑤ $|x^* - x_k| \leq \frac{L^k}{1-L} |x_1 - x_0|$? L 越小收敛越快

$|x_{k+1} - x_k| = |\varphi(x_k) - \varphi(x_{k-1})| = |\varphi'(\xi_k)(x_k - x_{k-1})| \leq L|x_k - x_{k-1}| \leq \dots \leq L^k |x_1 - x_0|$ ✓

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com 361

注：定理条件非必要条件,可将 $[a, b]$ 缩小,定义局部收敛性:若在 x^* 的某 δ 邻域

$B_\delta = \{x \mid |x - x^*| \leq \delta\}$

有 $\varphi \in C^1[a, b]$ 且 $|\varphi'(x^*)| < 1$, 则由 $\forall x_0 \in B_\delta$ 开始的迭代收敛. 即调整初值可得到收敛的结果.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com 362

迭代法的局部收敛性

定义：对于方程 $x = \varphi(x)$, 若在 x^* 的某个邻域 $R = \{x \mid |x - x^*| \leq \delta\}$ 内, 对任意初值 $x_0 \in R$, 迭代格式 $x_{k+1} = \varphi(x_k)$ 都收敛, 则称该迭代格式在 x^* 的附近是局部收敛的。

定理：设方程 $x = \varphi(x)$ 有根 x^* , 且在 x^* 的某个邻域 $R = \{x \mid |x - x^*| \leq \delta\}$ 内 $\varphi(x)$ 存在一阶连续导数, 则

- (1) 当 $|\varphi'(x^*)| < 1$ 时, 迭代格式局部收敛;
- (2) 当 $|\varphi'(x^*)| > 1$ 时, 迭代格式发散。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com 363

迭代法的计算步骤

step1 准备 提供迭代初值 x_0 ;

step2 迭代 计算迭代值 $x_1 = \varphi(x_0)$

step3 控制 检查 $|x_1 - x_0|$: 若 $|x_1 - x_0| > \varepsilon$, (ε 为预先指定的精度), 则以 x_1 替换 x_0 转 step2 继续迭代;

当 $|x_1 - x_0| \leq \varepsilon$ 时终止计算, 取 x_1 作为所求的结果.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com 364

Steffensen 加速法

$\hat{x} = x_0 - \frac{(x_1 - x_0)^2}{x_0 - 2x_1 + x_2}$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com 365

Steffensen 加速法

一般地, 有 $\hat{x}_k = x_k - \frac{(x_{k+1} - x_k)^2}{x_k - 2x_{k+1} + x_{k+2}}$

$x_0, \tilde{x}_1 = \varphi(x_0), \bar{x}_1 = \varphi(\tilde{x}_1)$

$x_1 = \bar{x}_1 - \frac{(\bar{x}_1 - \tilde{x}_1)^2}{\bar{x}_1 - 2\tilde{x}_1 + x_0}$

$\tilde{x}_2 = \varphi(x_1), \bar{x}_2 = \varphi(\tilde{x}_2)$

$x_2 = \bar{x}_2 - \frac{(\bar{x}_2 - \tilde{x}_2)^2}{\bar{x}_2 - 2\tilde{x}_2 + x_1}, \dots$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com 366

例: 求方程 $x^3-x^2-1=0$ 在 $x_0=1.5$ 附近的根, 设建立下列相应的迭代公式, 分析收敛性, 并求近似根.

解:

$$1) x = 1 + \frac{1}{x^2} \quad \varphi(x)$$

$$|\varphi'(x)| = \left| -2 \frac{1}{x^3} \right| = 2 \frac{1}{1.5^3} \Big|_{x_0=1.5} = 0.59 < 1 \text{ (收敛)}$$

$$2) x = \sqrt[3]{1+x^2}$$

$$|\varphi'(x)| = \left| \frac{1}{3} (1+x^2)^{-\frac{2}{3}} 2x \right| = \left| \frac{2}{3} x (1+x^2)^{-\frac{2}{3}} \right|_{x_0=1.5} = 0.4557 < 1 \text{ (收敛)}$$

$$3) x = \sqrt{\frac{1}{x-1}}$$

$$|\varphi'(x)| = \left| \frac{1}{2} (x-1)^{-\frac{3}{2}} \right|_{x_0=1.5} = 1.4142 > 1 \text{ (不收敛)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

367

用Steffensen法求方程 $x^3-x^2-1=0$ 在1.5附近的根

$$\text{方法 (1)} \quad x_{k+1} = \sqrt[3]{1+x_k^2}$$

$$\begin{cases} x_0 = 1.5 \\ \tilde{x}_1 = \sqrt[3]{1+1.5^2} = 1.48125 \\ \bar{x}_1 = \sqrt[3]{1+1.48125^2} = 1.47271 \end{cases}$$

$$\begin{cases} x_1 = \frac{(\bar{x}_1 - \tilde{x}_1)^2}{x_0 - 2\tilde{x}_1 + \bar{x}_1} = 1.46557 \\ \tilde{x}_2 = \sqrt[3]{1+1.46557^2} = 1.46557 \\ \bar{x}_2 = \sqrt[3]{1+1.46557^2} = 1.46557 \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

368

$$\text{方法 (2)} \quad x_{k+1} = 1 + \frac{1}{x_k^2}$$

$$\begin{cases} x_0 = 1.5 \\ \tilde{x}_1 = 1 + \frac{1}{1.5^2} = 1.44444 \\ \bar{x}_1 = 1 + \frac{1}{1.44444^2} = 1.479585 \end{cases}$$

$$\begin{cases} x_1 = \frac{1.5 * 1.479585 - 1.44444^2}{1.5 - 2 * 1.44444 + 1.479585} = 1.46597 \\ \tilde{x}_2 = 1 + \frac{1}{1.46597^2} = 1.46532 \\ \bar{x}_2 = 1 + \frac{1}{1.46532^2} = 1.46573 \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

369

$$x_2 = \frac{1.46597 * 1.46573 - 1.46532^2}{1.46597 - 2 * 1.46532 + 1.46573} = 1.46557$$

$$\tilde{x}_3 = 1 + \frac{1}{1.46557^2} = 1.46557$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

370

Newton法

原理: 将非线性方程逐步化为线性方程来求解.

设 $f(x)=0$ 有近似根 x_k , 将 $f(x)$ 在 x_k 处作一阶

Taylor展开 $f(x^*) = f(x_k) + f'(x_k)(x^* - x_k) + \frac{f''(\xi)}{2!}(x^* - x_k)^2$

ξ 在 x_k 和 x^* 之间.

将 $(x^* - x_k)^2$ 看成高阶小量, 则有:

$$f(x_k) + f'(x_k)(x^* - x_k) \approx 0$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

371

Newton法

$$f(x_k) + f'(x_k)(x^* - x_k) \approx 0$$

$$x^* \approx x_k - \frac{f(x_k)}{f'(x_k)}$$

线性方程

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

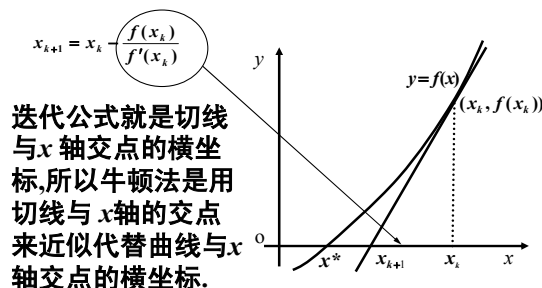
Newton公式

只要 $f \in C_1$, 每一步迭代都有 $f'(x_k) \neq 0$, 而且 $\{x_k\}$ 有极限 x^* , 则 x^* 就是 f 的根.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

372

Newton法的几何解释



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

373

Newton迭代法的计算步骤

step1 准备选定初始值 x_0 , 计算 $f_0=f(x_0)$ 和 $f'_0=f'(x_0)$

step2 迭代按公式 $x_1=x_0-f_0/f'_0$ 迭代一次, 得新的近似值 x_1 , 计算 $f_1=f(x_1)$ 和 $f'_1=f'(x_1)$.

step3 控制如果 x_1 满足 $|\delta| < \varepsilon_1$ 或 $|f_1| < \varepsilon_2$, 则终止迭代, 以 x_1 作为所求的根; 否则转 step4. 此处 $\varepsilon_1, \varepsilon_2$ 是允许误差, 而

$$\delta = \begin{cases} |x_1 - x_0| & \text{if } |x_1| < C \\ \frac{|x_1 - x_0|}{|x_1|} & \text{if } |x_1| \geq C \end{cases}$$

其中 C 是取绝对误差或相对误差的控制常数, 一般可取 $C=1$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

374

Newton迭代法的计算步骤

step4 修改 如果迭代次数达到预先指定的次数 N , 或者 $f'_1=0$, 则方法失败;

否则以 (x_1, f_1, f'_1) 代替 (x_0, f_0, f'_0) 继续迭代。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

375

迭代法的收敛速度

定义: 设序列 $\{x_k\}$ 收敛于 x^* , 并记

$$e_k = x_k - x^*, (k=0, 1, 2, \dots)$$

如果存在非零常数 c 和正常数 p , 使得

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

则称序列 $\{x_k\}$ 是 p 阶收敛的。

注: 上面地极限通常换为 $\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^p} = c$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

376

定理若 $\varphi(x)$ 在 x^* 附近的某个邻域内有 p 阶连续导数,

且 $\varphi(x^*) = x^*, \varphi'(x^*) = 0, \dots, \varphi^{(p-1)}(x^*) = 0, \varphi^{(p)}(x^*) \neq 0$

则对一个任意靠近 x^* 的初始值, 迭代公式 $x_{k+1} = \varphi(x_k)$ 是 p 阶收敛的, 且有

$$\lim_{k \rightarrow \infty} \frac{x_{k+1} - x^*}{(x_k - x^*)^p} = \frac{\varphi^{(p)}(x^*)}{p!}$$

注如果 $p=1$, 要求 $|\varphi'(x^*)| < 1$

证明 $x_{k+1} = \varphi(x_k) = \varphi(x^*) + \varphi'(x^*)(x_k - x^*) + \dots + \frac{\varphi^{(p)}(\xi_k)}{p!}(x_k - x^*)^p$

$$\frac{x_{k+1} - x^*}{(x_k - x^*)^p} \rightarrow \frac{\varphi^{(p)}(x^*)}{p!} \triangleq C$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

377

Newton法的局部收敛性

➤ 一般 迭代, 若 $\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|} = |\varphi'(x^*)| \neq 0$, 称为线性收敛, 这时 $p=1, 0 < C < 1$.

➤ Steffensen加速有 $\lim_{k \rightarrow \infty} \frac{|\hat{x}_k - x^*|}{|x_k - x^*|} = 0, \lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|} = 0$ 称为超线性收敛。

➤ Newton迭代 $\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^2} = \left| \frac{f''(x^*)}{2f'(x^*)} \right|$, 只要 $f'(x^*) \neq 0$ 就有 $p \geq 2$. 重根是线性收敛的。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

378

有根

根唯一

定理 (收敛的充分条件) 设 $f \in C^2[a, b]$, 若

(1) $f(a)f(b) < 0$; (2) 在整个 $[a, b]$ 上 f' 不变号且 $f'(x) \neq 0$; 序列单调有界, 保证收敛。

(3) 选取 $x_0 \in [a, b]$ 使得 $f(x_0)f''(x_0) > 0$;

则牛顿法产生的序列 $\{x_k\}$ 收敛到 $f(x)$ 在 $[a, b]$ 的唯一根。

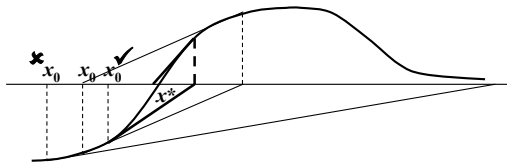
$$\varphi(x) = x - \frac{f(x)}{f'(x)} \quad \varphi'(x) = \frac{f(x)f''(x)}{[f'(x)]^2} \quad f'(x^*) \neq 0$$

$\Rightarrow \varphi'(x^*) = 0 \Rightarrow$ Newton法在根附近至少二阶收敛

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

379

注: Newton法收敛性依赖于 x_0 的选取。



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

380

下山法——Newton法局部微调

原理: 若由 x_k 得到的 x_{k+1} 不能使 $|f|$ 减小, 则在 x_k 和 x_{k+1} 之间找一个更好的点 x_{k+1} , 使得 $|f(x_{k+1})| < |f(x_k)|$ 。

$$\begin{aligned} x_{k+1} &= \lambda [x_k - \frac{f(x_k)}{f'(x_k)}] + (1-\lambda)x_k \\ &= x_k - \lambda \frac{f(x_k)}{f'(x_k)} \end{aligned}$$

$\lambda x_{k+1} + (1-\lambda)x_k, \lambda \in [0, 1]$

注: $\lambda = 1$ 时就是Newton公式。

当 $\lambda = 1$ 代入效果不好时, 将 λ 减半计算。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

381

改进Newton迭代公式(补充)

在 $f'(x^*)=0$ 时, Newton法是线性收敛的, 收敛速度不快。此时 x^* 是 $f(x)=0$ 的重根, 设 $f(x)=(x-x^*)^m q(x)$, 且 $q(x^*) \neq 0$ 。令

$$\begin{aligned} \mu(x) &= \frac{f(x)}{f'(x)} = \frac{(x-x^*)^m q(x)}{m(x-x^*)^{m-1} q(x) + (x-x^*)^m q'(x)} \\ &= (x-x^*) \frac{q(x)}{mq(x) + (x-x^*)q'(x)} \end{aligned}$$

由于 $q(x^*) \neq 0$, 所以 $\frac{q(x^*)}{mq(x^*) + (x^*-x^*)q'(x^*)} = \frac{1}{m} \neq 0$

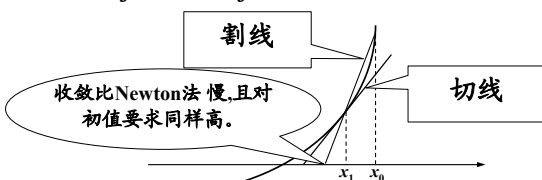
因此, x^* 是 $\mu(x)$ 的单根, 对 $\mu(x)$ 用牛顿法, 就可以求出 x^* , 是二阶而且收敛的。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

382

弦截法

Newton法要计算 f 和 f' , 计算量较大, 比较费时。现用 f 的值近似 f' , 可减少计算量。



收敛比Newton法慢, 且对初值要求同样高。

$$\text{切线斜率} \approx \text{割线斜率} \Rightarrow f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$

$$\Rightarrow x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})} \quad \text{需要2个初值 } x_0 \text{ 和 } x_1.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

383

抛物线法

如果已知 $f(x)=0$ 的三个近似根, 以这三个点为插值节点构造二次插值多项式 $P_2(x)$, 然后以 $P_2(x)$ 的一个零点作为新的近似根。

这样的迭代过程称为抛物线方法。

设已知的三点为 x_{k-2}, x_{k-1}, x_k , 则插值多项式为

$$P_2(x) = f[x_k] + f[x_k, x_{k-1}](x - x_k) + f[x_k, x_{k-1}, x_{k-2}](x - x_k)(x - x_{k-1})$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

384

抛物线法

$P_2(x)$ 有

两个零点 $x_{k+1} = x_k - \frac{2f(x_k)}{\omega \pm \sqrt{\omega^2 - 4f(x_k)f[x_k, x_{k-1}, x_{k-2}]}}$
其中 $\omega = f[x_k, x_{k-1}] + f[x_k, x_{k-1}, x_{k-2}](x_k - x_{k-1})$

在实际计算时,我们选取更接近于 x_k 的一个零点作为 x_{k+1} .
即分母中的正负号以 ω 为准,这样可使分母的绝对值较大.

$$x_{k+1} = x_k - \frac{2f(x_k)}{\omega + \text{sign}(\omega)\sqrt{\omega^2 - 4f(x_k)f[x_k, x_{k-1}, x_{k-2}]}}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

385

例: 分别用迭代法,Newton法,弦截法,抛物线法($x_0=2, x_1=1.9$)求方程 $x^3-3x-1=0$ 在 $x=2$ 附近的根

解: (1) 迭代法

因为 $x^3=3x+1 \therefore x = \sqrt[3]{1+3x}$

$$|\phi'(x)| = \left| \frac{1}{3}(1+3x)^{-\frac{2}{3}} \right| = \left| (1+3x)^{-\frac{2}{3}} \right|_{x=2} = 0.27 < 1$$

$$\therefore x_{n+1} = \sqrt[3]{1+3x_n}$$

$$x_0=2 \quad x_1 = \sqrt[3]{1+3 \cdot 2} = 1.91293$$

$$x_2 = \sqrt[3]{1+1.91293 \cdot 2} = 1.88883 \quad x_3 = \sqrt[3]{1+1.88883 \cdot 2} = 1.88203$$

$$x_4=1.88014 \quad x_5=1.87960 \quad x_6=1.87945$$

$$x_7=1.87940 \quad x_8=1.87939 \quad x_9=1.87939$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

386

(2) Newton法

$$x_{n+1} = x_n - \frac{x_n^3 - 3x_n - 1}{3(x_n^2 - 1)} = \frac{x_n \cdot 3(x_n^2 - 1) - (x_n^3 - 3x_n - 1)}{3(x_n^2 - 1)}$$

$$= \frac{3x_n^3 - 3x_n - x_n^3 + 3x_n + 1}{3(x_n^2 - 1)} = \frac{2x_n^3 + 1}{3(x_n^2 - 1)}$$

第一步: 形成迭代函数
 $f(x_0) = 2^3 - 3 \cdot 2 - 1 = 1$

第二步: 确定初值
 $f'(x_0) = (f'(x_0))' = (3(x_0^2 - 1))' = 6x_0 = 6 \cdot 2 = 12$

取初值 $x_0=2$ $x_1 = \frac{2 \cdot 2^3 + 1}{3 \cdot (2^2 - 1)} = 1.88889$

第三步: 迭代计算
 $x_2=1.87945 \quad x_3=1.87938 \quad x_4=1.87938$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

387

(3) 弦截法 $x_0=2, x_1=1.9$

$$x_0=2 \quad f(x_0) = 2^3 - 3 \cdot 2 - 1 = 1$$

$$x_1=1.9 \quad f(x_1) = 1.9^3 - 3 \cdot 1.9 - 1 = 0.159$$

$$x_2 = \frac{x_0 f(x_1) - x_1 f(x_0)}{f(x_1) - f(x_0)} = \frac{2 \cdot 0.159 - 1.9 \cdot 1}{0.159 - 1} = 1.88109 \quad f(x_2) = 0.01297$$

$$x_3 = \frac{1.9 \cdot 0.01297 - 1.88109 \cdot 0.159}{0.01297 - 0.159} = 1.87941 \quad f(x_3) = 0.00019$$

$$x_4 = \frac{1.88109 \cdot 0.00019 - 1.87941 \cdot 0.01297}{0.00019 - 0.01297} = 1.87939$$

$$x_5 = \dots = 1.87939$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

388

(3) 抛物线法 $x_0=2, x_1=1.9, x_2=1.88109$

$$f(x_0) = 1 \quad f(x_1) = 0.159 \quad f(x_2) = 0.011297$$

$$f[x_1, x_0] = 8.41 \quad f[x_2, x_1] = 7.72487 \quad f[x_2, x_1, x_0] = 5.76226$$

$$\omega = f[x_2, x_1] + f[x_2, x_1, x_0](x_2 - x_1) = 7.61599$$

$$x_3 = x_2 - \frac{2f(x_2)}{\omega + \sqrt{\omega^2 - 4f(x_2)f[x_2, x_1, x_0]}} = 1.87939$$

$$x_4 = \dots = 1.87939$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

389

例: $x=0$ 是 $e^x-x-1=0$ 的二重根,比较用Newton法和改进的Newton法求解方程 $e^x-x-1=0$ (取初值为1)

Newton法: 设 $f(x)=e^x-x-1$,Newton法的迭代公式为

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} = x_n - \frac{e^{x_n} - x_n - 1}{e^{x_n} - 1}$$

$$\text{设 } \mu(x) = \frac{f(x)}{f'(x)} = \frac{e^x - x - 1}{e^x - 1}$$

改进的Newton法的迭代公式为

$$x_{n+1} = x_n - \frac{\mu(x_n)}{\mu'(x_n)} = \frac{e^{x_n}(x_n^2 + 2) - e^{2x_n} - 1}{e^{x_n}(x_n - 1) + 1}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

390

用Newton法和改进的Newton法进行迭代得到的结果见下面的表格

Newton法	n	x_n	改进的Newton法	n	x_n
	0	1.0		0	1.0
	1	0.581977		1	-2.34211 10^{-1}
	2	0.319055		2	-8.45828 10^{-3}
	3	0.167996		3	-1.18895 10^{-4}
	4	0.0863489		4	-6.86382 10^{-6}
	5	0.0437957		5	-2.80852 10^{-7}

可以看出,改进的Newton法的收敛速度比Newton法要快的多.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

391

多项式求值

多项式求值的秦九韶方法

设 $f(x)=a_0x^n+a_1x^{n-1}+\dots+a_{n-1}x+a_n$ ($a_0\neq 0$),其系数均为实数.

要计算 $f(x)$ 在一点 x_0 的函数值.

若直接带入计算,则乘法的计算次数为

$$n+n-1+\dots+1=n(n+1)/2$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

392

多项式求值

将多项式重新写为

$$f(x)=a_0x^n+a_1x^{n-1}+\dots+a_{n-1}x+a_n$$

$$=a_n+x(a_{n-1}+x(\dots+x(a_1+a_0x)))$$

则乘法计算次数减少为 n .

上述计算过程可以写为

$$b_0=a_0, b_i=a_i+x_0b_{i-1} (i=1,2,\dots,n),$$

$$f(x_0)=b_n.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

393

多项式求值

$$b_0=a_0, b_i=a_i+x_0b_{i-1} (i=1,2,\dots,n), f(x_0)=b_n.$$

$$\text{设 } p(x)=b_0x^{n-1}+b_1x^{n-2}+\dots+b_{n-2}x+b_{n-1},$$

$$\text{则 } (x-x_0)p(x)$$

$$=b_0x^n+b_1x^{n-1}+b_2x^{n-2}+\dots+b_{n-2}x^2+b_{n-1}x$$

$$-x_0b_0x^{n-1}-x_0b_1x^{n-2}-\dots-x_0b_{n-2}x-x_0b_{n-1}$$

$$=a_0x^n+a_1x^{n-1}+a_2x^{n-2}+\dots+a_{n-1}x+a_n-b_n$$

$$=f(x)-f(x_0)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

394

多项式求值

$$p(x)=b_0x^{n-1}+b_1x^{n-2}+\dots+b_{n-2}x+b_{n-1},$$

$$(x-x_0)p(x)=f(x)-f(x_0)$$

$$\text{因此 } f(x)=(x-x_0)p(x)+f(x_0)$$

即 $p(x)$ 是 $f(x)$ 除以 $(x-x_0)$ 的商式,

$f(x_0)$ 是 $f(x)$ 除以 $(x-x_0)$ 的商式(数).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

395

多项式求值

$$b_0=a_0, b_i=a_i+x_0b_{i-1} (i=1,2,\dots,n), f(x_0)=b_n.$$

例 求 $f(x)=x^5+4x^4-6x^2+3x-9$ 在 $x_0=2$ 处的函数值.

解:将多项式的系数及 x_0 写为如下形式.

$$\begin{array}{r|rrrrrr}
 2 & 1 & 4 & 0 & -6 & 3 & -9 \\
 \hline
 x_0 * b_0 & \rightarrow & 2 & 12 & 24 & 36 & 78 \\
 b_0 = a_0 & \rightarrow & 1 & 6 & 12 & 18 & 39 & 69 \\
 & & & \uparrow & \uparrow & & & \\
 & & & b_1 = a_1 + x_0 * b_0 & & \text{商式 } p(x) \text{ 的系数} & &
 \end{array}$$

类似可求以后的各个数值.
该数即为 $b_5=f(x_0)$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

396

导数求值

对 n 次多项式 $f(x)$,下面的Taylor展式精确成立.

$$f(x) = f(x_0) + f'(x_0)(x-x_0) + \frac{f''(x_0)}{2!}(x-x_0)^2 + \cdots + \frac{f^{(n)}(x_0)}{n!}(x-x_0)^n$$

显然, $f(x)$ 除以 $(x-x_0)$ 的商式为

$$p_1(x) = f'(x_0) + \frac{f''(x_0)}{2!}(x-x_0) + \cdots + \frac{f^{(n)}(x_0)}{n!}(x-x_0)^{n-1}$$

$p_1(x)$ 除以 $(x-x_0)$ 的余式为 $f'(x_0)$,商式设为 $p_2(x)$.

$p_2(x)$ 除以 $(x-x_0)$ 的余式为 $f''(x_0)/2$.

该过程进行下去可以求得 $f(x)$ 在 x_0 的各阶导数值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

397

导数求值

例求

$f(x)=x^5+4x^4-6x^2+3x-9$ 在
 $x_0=2$ 处的各
阶导数值.

$$\begin{array}{rcl} 2 & \begin{array}{r} 1 \quad 4 \quad 0 \quad -6 \quad 3 \quad -9 \\ 2 \quad 12 \quad 24 \quad 36 \quad 78 \end{array} & \longleftarrow f(x_0) \\ 2 & \begin{array}{r} 1 \quad 6 \quad 12 \quad 18 \quad 39 \quad 69 \\ 2 \quad 16 \quad 56 \quad 148 \end{array} & \longleftarrow f'(x_0) \\ 2 & \begin{array}{r} 1 \quad 8 \quad 28 \quad 74 \quad 187 \\ 2 \quad 20 \quad 96 \end{array} & \longleftarrow f''(x_0) \\ 2 & \begin{array}{r} 1 \quad 10 \quad 48 \quad 170 \\ 2 \quad 24 \end{array} & \longleftarrow \frac{f'''(x_0)}{3!} \\ 2 & \begin{array}{r} 1 \quad 12 \quad 72 \\ 2 \end{array} & \longleftarrow \frac{f^{(4)}(x_0)}{4!} \\ & \begin{array}{r} 1 \quad 14 \\ 1 \end{array} & \longleftarrow \frac{f^{(5)}(x_0)}{5!} \end{array}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

398

导数求值

由以上的过程可以得到

$$f(2)=69, f'(2)=187, f''(2)=340, f'''(2)=432,$$

$$f^{(4)}(2)=336, f^{(5)}(2)=120.$$

另外还可以给出 $f(x)$ 在 $x_0=2$ 处的Taylor展式

$$f(x)=69+187(x-2)+170(x-2)^2+72(x-2)^3+14(x-2)^4+(x-2)^5$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

399

第七章 解线性方程组的直接方法

§ 1 引言

解线性方程组的两种方法:

直接法:通过有限步算术运算直接求出方程组的精确解,最常用的是消元结合代入的方法.

实际上除非是采用无穷位精度计算,一般都得不到精确解.

直接方法适用于解低阶稠密矩阵方程组.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

401

§ 1 引言

迭代法 类似于方程求根的迭代法,用一个迭代过程逐步逼近方程组的解.

迭代有可能不收敛,或虽然收敛,但收敛速度慢.

迭代法适用于求解高阶稀疏矩阵方程组.

稀疏矩阵:矩阵非零元素较少,且在固定的位置上.

稀疏矩阵一般是人为构造的,例如36页三转角插值时方程组(8.12),(8.15)的系数矩阵.

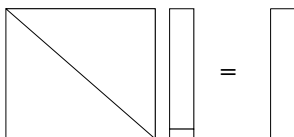
南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

402

§ 2 Gauss消去法

问题:求解 $Ax=b$.

思路:将 A 化为上三角阵,再回代求解.



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

403

例题

例1 用消去法解方程组

$$\begin{cases} x_1 + x_2 + x_3 = 6 \\ 4x_2 - x_3 = 5 \\ 2x_1 - 2x_2 + x_3 = 1 \end{cases}$$

Step1 用第一个方程消去后两个方程的 x_1 .得

$$\begin{cases} x_1 + x_2 + x_3 = 6 \\ 4x_2 - x_3 = 5 \\ -4x_2 - x_3 = -11 \end{cases}$$

Step2 用第二个方程消去第三个方程的 x_2 .得

$$\begin{cases} x_1 + x_2 + x_3 = 6 \\ 4x_2 - x_3 = 5 \\ -2x_3 = -6 \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

404

例题

$$\begin{cases} x_1 + x_2 + x_3 = 6 \\ 4x_2 - x_3 = 5 \\ -2x_3 = -6 \end{cases}$$

Step3 回代求解

根据第三个方程得 $x_3=3$,代入第二个方程得 $x_2=2$.

再一起代入第一个方程得到 $x_1=1$.

于是得到原方程组的解 $(1,2,3)^T$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

405

Gauss消去法(第一次消元)

考虑方程组

$$A^{(1)}x=b^{(1)} \quad \begin{cases} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \cdots + a_{1n}^{(1)}x_n = b_1^{(1)} \\ a_{21}^{(1)}x_1 + a_{22}^{(1)}x_2 + \cdots + a_{2n}^{(1)}x_n = b_2^{(1)} \\ \vdots \\ a_{n1}^{(1)}x_1 + a_{n2}^{(1)}x_2 + \cdots + a_{nn}^{(1)}x_n = b_n^{(1)} \end{cases}$$

第一次消元用第一个方程将后面方程的 x_1 消去.

计算乘数 $m_{i1} = \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} \leftarrow \text{条件: } a_{11}^{(1)} \neq 0$

用 $-m_{i1}$ 乘以第一个方程加到第 i 个($i=1, \dots, n$)方程上,则消去了第 i 个方程中的 x_1 .

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

406

Gauss消去法(第一次消元)

经过上述过程,得到方程组 $A^{(2)}x=b^{(2)}$,

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_n^{(2)} \end{bmatrix}$$

其中

$$a_{ij}^{(2)} = a_{ij}^{(1)} - m_{i1}a_{1j}^{(1)},$$

$$b_i^{(2)} = b_i^{(1)} - m_{i1}b_1^{(1)}, \quad (i, j = 2, 3, \dots, n)$$

$$m_{i1} = \frac{a_{i1}^{(1)}}{a_{11}^{(1)}}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

407

Gauss消去法(第 k 次消元)

假设已完成 $k-1$ 次消元,得到方程组 $A^{(k)}x=b^{(k)}$.

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1,k-1}^{(1)} & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2,k-1}^{(2)} & a_{2k}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{k-1,k-1}^{(k-1)} & a_{k-1,k}^{(k-1)} & \cdots & a_{k-1,n}^{(k-1)} \\ 0 & 0 & \cdots & 0 & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} \end{bmatrix}$$

第 k 次消元的目的是将 $a_{kk}^{(k)}$ (称为主元)下面的元素变为0.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

408

Gauss消去法(第k次消元)

对 $A^{(k)}$ 右下角的矩阵

$$\begin{bmatrix} a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ \vdots & \ddots & \vdots \\ a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} \end{bmatrix}$$

计算乘数 $m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} \leftarrow$ 条件: $a_{kk}^{(k)} \neq 0$

用 $-m_{ik}$ 乘以第 k 个方程加到第 i 个($i=k+1, \dots, n$)方程上,则消去了第 i 个方程中的 x_k ,得到方程组 $A^{(k+1)}x=b^{(k+1)}$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

409

Gauss消去法(第k次消元)

第一步消元的计算公式

$$a_{ij}^{(2)} = a_{ij}^{(1)} - m_{i1}a_{1j}^{(1)}, \quad b_i^{(2)} = b_i^{(1)} - m_{i1}b_1^{(1)}, \quad m_{i1} = \frac{a_{i1}^{(1)}}{a_{11}^{(1)}}$$

类似可以得到第 k 步消元的计算公式

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik}a_{kj}^{(k)},$$

$$b_i^{(k+1)} = b_i^{(k)} - m_{ik}b_k^{(k)}, \quad (i, j = k+1, \dots, n)$$

$$m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

410

Gauss消去法

消去法完成后最终得到与原方程组等价的三角形方程组 $A^{(n)}x=b^{(n)}$.

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn}^{(n)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_n^{(n)} \end{bmatrix}$$

一共需进行 $n-1$ 步

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

411

Gauss消去法(算法)

$k=1, \dots, n-1$

$i=k+1, \dots, n$

$$m_{ik} = a_{ik}^{(k)} / a_{kk}^{(k)}$$

$i, j=k+1, \dots, n$

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik}a_{kj}^{(k)}$$

$i=k+1, \dots, n$

$$b_i^{(k+1)} = b_i^{(k)} - m_{ik}b_k^{(k)}$$

$$x_n = b_n^{(n)} / a_{nn}^{(n)}$$

$k=n-1, \dots, 1$

$$x_k = (b_k^{(k)} - \sum_{j=k+1}^n a_{kj}^{(k)} x_j) / a_{kk}^{(k)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

412

完成Gauss消去法的条件

Gauss消去法可以完成的充要条件为

$$a_{ii}^{(i)} \neq 0 (i=1, 2, \dots, n-1)$$

如何用原始的系数矩阵 A 来表示上面的条件?

考察消去法进行 $k-1$ 步后的系数矩阵 $A^{(k)}$

$$\begin{bmatrix} a_{11}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} \end{bmatrix} \quad \text{其第} k \text{个顺序主子式为}$$

$$a_{11}^{(1)} a_{22}^{(2)} \cdots a_{kk}^{(k)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

413

完成Gauss消去法的条件

在Gauss消去法的过程中我们对矩阵 A 实施的是矩阵的第二种初等行变换(一行乘以一个数加到另一行),不改变任何子式的值.

因此,若设 D_k 为矩阵 A 的第 k 个顺序主子式,则

$$D_k = a_{11}^{(1)} a_{22}^{(2)} \cdots a_{kk}^{(k)}$$

显然 $a_{ii}^{(i)} \neq 0 (i=1, 2, \dots, n-1)$ 的充要条件为

$$D_i \neq 0 (i=1, 2, \dots, n-1)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

414

完成Gauss消去法的条件

定理2(P169)如果 n 阶矩阵 A 的所有顺序主子式均不为零,即 $D_i \neq 0 (i=1,2,\dots,n)$,则可通过Gauss消去法(不进行交换两行的初等变换),将方程组 $Ax=b$ 约化为三角形方程组.

注:实际求解方程组时,通常要求 $|A| \neq 0$,因此,定理2中条件虽然比前面的充要条件稍强一些,却是比较自然的.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

415

完成Gauss消去法的条件

定理2中的条件是相当强的,一般不能满足,比如第一个方程的 x_1 的系数为零.

此时通常的做法是交换两行,使得消去法可以进行.

定理1(P168) 如果 A 为非奇异矩阵,则可以通过Gauss消去法及交换两行的初等变换将原方程组化为三角形方程组.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

416

完成Gauss消去法的条件

$$\begin{bmatrix} a_{11}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} \\ & \ddots & & & \\ & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} \end{bmatrix}$$

在遇到主元 $a_{kk}^{(k)}$ 为零时,可以选择它下面的非零元素作为主元.

如果 $a_{kk}^{(k)}$ 以及它下方的元素全为零,则矩阵是奇异的.

因此在矩阵非奇异时,结合换行,Gauss消去法总是可以完成的.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

417

矩阵的三角分解

Gauss消去法第 k 步消元时,以 $-m_{ik}$ 乘以矩阵 $A^{(k)}$ 第 k 行加到第 i 行($i=k+1,\dots,n$)得到矩阵 $A^{(k+1)}$,因此 $A^{(k+1)}=L_k A^{(k)}$

其中

$$L_k = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -m_{k+1,k} & 1 & \\ & & \vdots & & \ddots \\ & & -m_{nk} & & & 1 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

418

矩阵的三角分解

根据 $A^{(k+1)}=L_k A^{(k)} (k=1,\dots,n-1)$,可以得到 $A^{(n)}=L_{n-1} \cdots L_2 L_1 A^{(1)}$

因此 $A=A^{(1)}=L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1} A^{(n)}$

令 $L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1}=L$, $A^{(n)}=U$,有 $A=LU$.

若Gauss消去法可以完成,则 $U(A^{(n)})$ 为上三角矩阵.

另一方面,我们可以求出 L 的形式.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

419

由 $L_k = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -m_{k+1,k} & 1 & \\ & & \vdots & & \ddots \\ & & -m_{nk} & & & 1 \end{bmatrix}$ 可得 $L_k^{-1} = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & m_{k+1,k} & 1 & \\ & & \vdots & & \ddots \\ & & m_{nk} & & & 1 \end{bmatrix}$

于是

$$L=L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1} = \begin{bmatrix} 1 & & & & \\ m_{21} & 1 & & & \\ m_{31} & m_{32} & 1 & & \\ \vdots & \vdots & \vdots & \ddots & \\ m_{n-1,1} & m_{n-1,2} & m_{n-1,3} & \cdots & 1 \\ m_{n1} & m_{n2} & m_{n3} & \cdots & m_{n,n-1} & 1 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

420

矩阵的三角分解

根据前面的推导,在Gauss消去法可以进行时,矩阵可以分解为一个单位下三角矩阵(对角线元素为1的下三角矩阵)和一个上三角矩阵的乘积.

即,若矩阵 A 的顺序主子式 $D_i \neq 0 (i=1,2,\dots,n-1)$,则 A 可以分解为单位下三角矩阵 L 和一个上三角矩阵 U 的乘积.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

421

矩阵的三角分解

那么,上述分解是否具有唯一性?

即若 $A=LU=L_1U_1$ (L, L_1 为单位下三角矩阵, U, U_1 为上三角矩阵),那么是否 $L=L_1, U=U_1$?

若 A 非奇异,则有 $L^{-1}L_1=UU_1^{-1}$ (U_1 非奇异).

等式右端为上三角矩阵,左端为单位下三角矩阵,因此等式两边为单位阵, $L=L_1, U=U_1$.

在 A 为奇异时,唯一性的证明作为作业.

最终我们得到定理3(Page170).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

422

矩阵的三角分解

定理3 设 A 为 n 阶矩阵,如果矩阵 A 的顺序主子式 $D_i \neq 0 (i=1,2,\dots,n-1)$,则 A 可以分解为单位下三角矩阵 L 和一个上三角矩阵 U 的乘积,且这种分解是唯一的.

例 对于矩阵

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 4 & -1 \\ 2 & -2 & 1 \end{bmatrix}$$

由前面的消元过程可知

$m_{21}=0, m_{31}=2, m_{32}=-1$, 因此

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 4 & -1 \\ 0 & 0 & -2 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

423

计算量(P170)

下面是第 k 次消元时所需要完成的计算,我们考察其乘除法的计算次数.

$$m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}, \quad (i = k+1, \dots, n) \quad n-k$$

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik} a_{kj}^{(k)}, \quad (i, j = k+1, \dots, n) \quad (n-k)^2$$

$$b_i^{(k+1)} = b_i^{(k)} - m_{ik} b_k^{(k)}, \quad (i = k+1, \dots, n) \quad n-k$$

消元时总的乘除工作量为

$$\sum_{k=1}^{n-1} [(n-k)^2 + 2(n-k)]$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

424

计算量

在得到三角形方程组后,回代求解的计算公式为

$$x_n = \frac{b_n^{(n)}}{a_{nn}^{(n)}} \quad x_k = \frac{1}{a_{kk}^{(k)}} (b_k^{(k)} - \sum_{j=k+1}^n a_{kj}^{(k)} x_j) \quad (k = n-1, \dots, 1)$$

乘除法的计算量为

$$1 + \sum_{k=1}^{n-1} (n-k+1)$$

乘除法的总的计算量为

$$\sum_{k=1}^{n-1} [(n-k)^2 + 2(n-k)] + 1 + \sum_{k=1}^{n-1} (n-k+1) = \frac{n^3}{3} + n^2 - \frac{n}{3}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

425

§ 3 Gauss主元素消去法

考虑方程组

$$\begin{bmatrix} 0.001 & 2.000 & 3.000 \\ -1.000 & 3.712 & 4.623 \\ -2.000 & 1.072 & 5.643 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1.000 \\ 2.000 \\ 3.000 \end{bmatrix}$$

如果将系数都看成精确的,可以得到有理数解,取4位有效数字为 $(-0.4904, -0.05104, 0.3675)^T$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

426

Gauss主元素消去法

Gauss消去法求解

$$\begin{bmatrix} 0.001 & 2.000 & 3.000 & 1.000 \\ -1.000 & 3.712 & 4.623 & 2.000 \\ -2.000 & 1.072 & 5.643 & 3.000 \end{bmatrix}$$

$$\begin{array}{l} r_1 \cdot 1000 + r_2 \rightarrow r_2 \\ r_1 \cdot 2000 + r_3 \rightarrow r_3 \end{array} \begin{bmatrix} 0.001 & 2.000 & 3.000 & 1.000 \\ 0 & 2004 & 3005 & 1002 \\ 0 & 4001 & 6006 & 2003 \end{bmatrix}$$

$$\begin{array}{l} r_2 \cdot \left(-\frac{4001}{2004}\right) + r_3 \rightarrow r_3 \end{array} \begin{bmatrix} 0.001 & 2.000 & 3.000 & 1.000 \\ 0 & 2004 & 3005 & 1002 \\ 0 & 0 & 5.000 & 2.000 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

427

$$\begin{bmatrix} 0.001 & 2.000 & 3.000 & 1.000 \\ 0 & 2004 & 3005 & 1002 \\ 0 & 0 & 5.000 & 2.000 \end{bmatrix}$$

因此有 $x_3 = \frac{2.000}{5.000} = 0.4000$,

$$x_2 = \frac{1002 - 3005 \cdot 0.4000}{2004} = -0.09980,$$

$$x_1 = \frac{1.000 - 2.000 \cdot (-0.09980) - 3.000 \cdot 0.4000}{0.001} = -0.400,$$

得到解为 $\bar{x} = (-0.400, -0.09980, 0.4000)^T$.

精确解为 $(-0.4904, -0.05104, 0.3675)^T$, 误差很大!

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

428

分析

$$\begin{bmatrix} 0.001 & 2.000 & 3.000 & 1.000 \\ -1.000 & 3.712 & 4.623 & 2.000 \\ -2.000 & 1.072 & 5.643 & 3.000 \end{bmatrix} \begin{array}{l} r_1 \cdot 1000 + r_2 \rightarrow r_2 \\ r_1 \cdot 2000 + r_3 \rightarrow r_3 \end{array} \begin{bmatrix} 0.001 & 2.000 & 3.000 & 1.000 \\ 0 & 2004 & 3005 & 1002 \\ 0 & 4001 & 6006 & 2003 \end{bmatrix}$$

在右面的矩阵中,2004处的值应该是

$$3.712 + 2.000 \cdot \left(-\frac{1.000}{0.001}\right) = 3.712 + 2000 \approx 2004$$

事实上,3.712处的值换成3.500到4.499间的任何值时,变换后相应位置的值都是2004.

变换后的方程组不能完全反映原来的方程组!

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

429

分析

$$3.712 + 2.000 \cdot \left(-\frac{1.000}{0.001}\right) = 3.712 + 2000 \approx 2004$$

之所以造成这种情况,主要是因为2000相对于3.712而言太大.

进一步分析可以发现,2000太大的一个关键原因是主元0.001太小.

在求解时,我们尽量选取大的主元.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

430

Gauss主元素消去法

Gauss选主元素消去法求解

$$\begin{bmatrix} 0.001 & 2.000 & 3.000 & 1.000 \\ -1.000 & 3.712 & 4.623 & 2.000 \\ -2.000 & 1.072 & 5.643 & 3.000 \end{bmatrix}$$

$$\begin{array}{l} r_1 \leftrightarrow r_3 \\ r_1 \cdot \left(-\frac{1.000}{-2.000}\right) + r_2 \rightarrow r_2 \\ r_1 \cdot (0.001/2.000) + r_3 \rightarrow r_3 \end{array} \begin{bmatrix} -2.000 & 1.072 & 5.643 & 3.000 \\ -1.000 & 3.712 & 4.623 & 2.000 \\ 0.001 & 2.000 & 3.000 & 1.000 \end{bmatrix}$$

$$\begin{bmatrix} -2.000 & 1.072 & 5.643 & 3.000 \\ 0 & 3.176 & 1.801 & 0.5000 \\ 0 & 2.001 & 3.003 & 1.002 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

431

$$\begin{bmatrix} -2.000 & 1.072 & 5.643 & 3.000 \\ 0 & 3.176 & 1.801 & 0.5000 \\ 0 & 2.001 & 3.003 & 1.002 \end{bmatrix}$$

$$\begin{array}{l} r_2 \cdot \left(-\frac{2.001}{3.176}\right) + r_3 \rightarrow r_3 \end{array} \begin{bmatrix} -2.000 & 1.072 & 5.643 & 3.000 \\ 0 & 3.176 & 1.801 & 0.5000 \\ 0 & 0 & 1.868 & 0.6870 \end{bmatrix}$$

因此有

$$x_3 = \frac{0.6870}{1.868} = 0.3678,$$

$$x_2 = \frac{0.5000 - 1.801 \cdot 0.3678}{3.176} = -0.05113,$$

$$x_1 = \frac{3.000 - 1.072 \cdot (-0.05113) - 5.643 \cdot 0.3678}{2.000} = -0.4900.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

432

Gauss主元素消去法

得到解 $x = (-0.4900, -0.05113, 0.3678)^T$.

精确解为 $(-0.4904, -0.05104, 0.3675)^T$, 误差很小!

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

433

Gauss主元素消去法

Gauss主元素消去法分为列主元素消去法和全主元素消去法.

列选主元素消去法

如果已经进行了 $k-1$ 步消元, 得到右边的增广矩阵.

$$\begin{bmatrix} a_{11}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ & \ddots & & \ddots & & \\ & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} & b_k^{(k)} \\ & & & \ddots & & \\ & & & & a_{nk}^{(k)} & b_n^{(k)} \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

434

Gauss列选主元素消去法

$$\begin{bmatrix} a_{11}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ & \ddots & & \ddots & & \\ & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} & b_k^{(k)} \\ & & & \ddots & & \\ & & & & a_{nk}^{(k)} & b_n^{(k)} \end{bmatrix} \Rightarrow$$

在 $a_{kk}^{(k)}, \dots, a_{nk}^{(k)}$ 中选取绝对值最大的元素

$$|a_{i_k,k}^{(k)}| = \max_{k \leq i \leq n} |a_{ik}^{(k)}|$$

交换第 i_k 行与第 k 行,

然后将主元以下元素消为零, 即完成第 k 步消去.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

435

算法

$k = 1, \dots, n-1$	
	$ a_{i_k,k} = \max_{k \leq i \leq n} a_{ik} $
YES	$i_k = k$
NO	
	$j = k, \dots, n$
	$a_{kj} \leftrightarrow a_{i_k,j} \quad b_k \leftrightarrow b_{i_k}$
$i = k+1, \dots, n$	$m_{ik} = a_{ik}^{(k)} / a_{kk}^{(k)}$
$i, j = k+1, \dots, n$	$a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik} a_{kj}^{(k)}$
$i = k+1, \dots, n$	$b_i^{(k+1)} = b_i^{(k)} - m_{ik} b_k^{(k)}$
	$x_n = b_n^{(n)} / a_{nn}^{(n)}$
$k = n-1, \dots, 1$	$x_k = (b_k^{(k)} - \sum_{j=k+1}^n a_{kj}^{(k)} x_j) / a_{kk}^{(k)}$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

436

Gauss主元素消去法

全选主元素消去法

另一种主元素消去法称为全选主元素消去法, 第 k 步消去时, 选取系数矩阵的第 k 行至第 n 行, 第 k 列至第 n 列中绝对值最大的元素作为主元.

$$\begin{bmatrix} a_{11}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ & \ddots & & \ddots & & \\ & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} & b_k^{(k)} \\ & & & \ddots & & \\ & & & & a_{nk}^{(k)} & b_n^{(k)} \end{bmatrix}$$

注: 在全选主元时, 必须对矩阵进行列变换, 此时必须记录列交换的过程.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

437

Gauss-Jordan消去法

在消元过程中, 如果在不仅消去对角线下方的元素, 同时消去对角线上方的元素, 同时使对角线元素变为1, 这种方法称为Gauss-Jordan消去法.

$$\begin{bmatrix} 1 & 1 & 1 & 6 \\ 0 & 4 & -1 & 5 \\ 2 & -2 & 1 & 1 \end{bmatrix} \xrightarrow{r_1 \cdot (-2) + r_3 \rightarrow r_3} \begin{bmatrix} 1 & 1 & 1 & 6 \\ 0 & 4 & -1 & 5 \\ 0 & -4 & -1 & -11 \end{bmatrix}$$

$$\xrightarrow{\begin{matrix} r_2 \cdot (-1/4) + r_1 \rightarrow r_1 \\ r_2 \cdot (-(-4)/4) + r_3 \rightarrow r_3 \end{matrix}} \begin{bmatrix} 1 & 0 & 5/4 & 19/4 \\ 0 & 4 & -1 & 5 \\ 0 & 0 & -2 & -6 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

438

Gauss-Jordan消去法

$$\begin{bmatrix} 1 & 0 & 5/4 & 19/4 \\ 0 & 4 & -1 & 5 \\ 0 & 0 & -2 & -6 \end{bmatrix} \xrightarrow{r_2 \cdot (1/4) \rightarrow r_2} \begin{bmatrix} 1 & 0 & 5/4 & 19/4 \\ 0 & 1 & -1/4 & 5/4 \\ 0 & 0 & -2 & -6 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & -2 & -6 \end{bmatrix} \xrightarrow{\begin{matrix} r_3 \cdot (-5/4)/(-2) + r_1 \rightarrow r_1 \\ r_3 \cdot (-(-1/4)/(-2)) + r_2 \rightarrow r_2 \end{matrix}} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

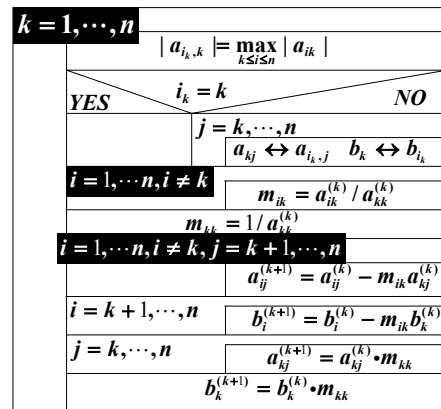
于是得到解为 $x=(1,2,3)^T$.

注:求解时,也可以先将主元变为1,再消元.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

439

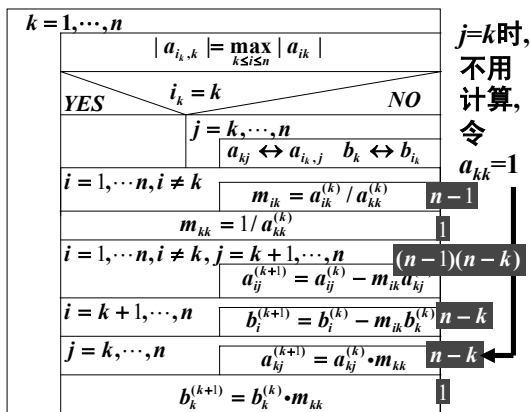
算法



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

440

计算量



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

441

计算量

我们可以得到Gauss-Jordan消去法的总的乘除法的计算量为

$$\sum_{k=1}^n [(n+1)(n-k) + n + 1] = \frac{n^3}{2} + n^2 + \frac{n}{2} = \frac{n^3}{2} + O(n^2)$$

Gauss消去法的计算量为 $= \frac{n^3}{3} + O(n^2)$

可以看出, Gauss-Jordan消去法的计算量大约是Gauss消去法的1.5倍.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

442

Gauss-Jordan消去法求逆矩阵

Gauss-Jordan消去法的计算量较大,一般不直接用来求解单个的方程组,而用来求解有着相同系数矩阵的多个方程组.

已知矩阵 A ,其逆矩阵 A^{-1} 满足 $AA^{-1}=I$.

令 $A^{-1}=[x_1, x_2, \dots, x_n]$,则有

$Ax_i = e_i$ (第 i 个分量为1,其余为零的向量).

因此,求逆矩阵的问题可以转化为求解 n 个系数矩阵均为 A 的线性方程组问题.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

443

用Gauss-Jordan方法求 $A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 6 \end{bmatrix}$ 的逆 A^{-1}

$$C = \begin{bmatrix} 1 & 2 & 3 & 1 & 0 & 0 \\ 2 & 4 & 5 & 0 & 1 & 0 \\ 3 & 5 & 6 & 0 & 0 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 2 & 3 & 1 & 0 & 0 \\ 2 & 4 & 5 & 0 & 1 & 0 \\ 3 & 5 & 6 & 0 & 0 & 1 \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} 1 & 5/3 & 2 & 0 & 0 & 1/3 \\ 0 & 2/3 & 1 & 0 & 1 & -2/3 \\ 0 & 1/3 & 1 & 1 & 0 & -1/3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & -1/2 & 0 & -5/2 & 2 \\ 0 & 1 & 3/2 & 0 & 3/2 & -1 \\ 0 & 0 & 1/2 & 1 & -1/2 & 0 \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} 1 & 0 & 0 & 1 & -3 & 2 \\ 0 & 1 & 0 & -3 & 3 & -1 \\ 0 & 0 & 1 & 2 & -1 & 0 \end{bmatrix} = (I_3 | A^{-1}) \quad A^{-1}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

444

§ 4 Gauss消去法的变形

根据Gauss消去法,可以得到矩阵的 LU 分解.

另一方面,我们也可以直接根据矩阵的元素写出矩阵 L, U 的元素的计算公式.

若 $A=LU$,求解则方程组 $Ax=b$ 可以转化为求解两个三角形方程组 $Ly=b$ (求 y), $Ux=y$ (求 x).

直接三角分解法

设 A 为 n 阶非奇异矩阵,且有分解式 $A=LU$,其中 L 为单位下三角矩阵, U 为上三角矩阵,即

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

考虑 A 一共有 n^2 个元素,可以确立 n^2 个等式,在一定条件下可以解出上式中的 n^2 个未知数.

直接三角分解法

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

根据上式有

$$a_{i1} = u_{i1} \quad (i=1, \cdots, n), \quad a_{i1} = l_{i1}u_{11}, l_{i1} = a_{i1}/u_{11} \quad (i=2, \cdots, n)$$

可以求出 U 的第一行以及 L 的第一列.

直接三角分解法

假设 U 的前 $r-1$ 行和 L 的前 $r-1$ 列已经求出. 考虑 A 的第 r 行的元素

$$a_{ri} = \sum_{k=1}^n l_{rk}u_{ki} = \sum_{k=1}^{r-1} l_{rk}u_{ki} + l_{rr}u_{ri} + \sum_{k=r+1}^n l_{rk}u_{ki}$$

由于 L 为单位下三角矩阵,因此 $l_{rr}=1, l_{rk}=0(k>r)$,

$$a_{ri} = \sum_{k=1}^{r-1} l_{rk}u_{ki} + u_{ri}, \quad u_{ri} = a_{ri} - \sum_{k=1}^{r-1} l_{rk}u_{ki}$$

从而可以求出 U 的第 r 行元素 $u_{ri}(i=r, \cdots, n)$.

直接三角分解法

再考虑 A 的第 r 列元素

$$a_{ir} = \sum_{k=1}^n l_{ik}u_{kr} = \sum_{k=1}^{r-1} l_{ik}u_{kr} + l_{ir}u_{rr} + \sum_{k=r+1}^n l_{ik}u_{kr}$$

由于 U 为上三角矩阵,因此 $u_{kr}=0(k>r)$,

$$a_{ir} = \sum_{k=1}^{r-1} l_{ik}u_{kr} + l_{ir}u_{rr}, \quad l_{ir} = (a_{ir} - \sum_{k=1}^{r-1} l_{ik}u_{kr})/u_{rr},$$

从而可以求出 L 的第 r 列元素 $l_{ir}(i=r+1, \cdots, n)$.

直接三角分解解方程组(算法)

$$(1) i=1, \cdots, n: u_{i1} = a_{i1}; \quad (2) i=2, \cdots, n: l_{i1} = a_{i1}/u_{11};$$

$$(3) r=2, \cdots, n-1: \quad i=r, \cdots, n: u_{ri} = a_{ri} - \sum_{k=1}^{r-1} l_{rk}u_{ki};$$

$$i=r+1, \cdots, n: l_{ir} = (a_{ir} - \sum_{k=1}^{r-1} l_{ik}u_{kr})/u_{rr};$$

$$(3') u_{nn} = a_{nn} - \sum_{k=1}^{n-1} l_{nk}u_{kn};$$

$$(4) y_1 = b_1, \quad i=2, \cdots, n: y_i = b_i - \sum_{k=1}^{i-1} l_{ik}y_k;$$

$$(5) x_n = y_n/u_{nn}, \quad i=n-1, \cdots, 1: x_i = (y_i - \sum_{k=i+1}^n l_{ik}x_k)/u_{ii}.$$

直接三角分解解方程组(计算量)

$$\begin{aligned}
 (1) & i = 1, \dots, n: u_{i1} = a_{i1}; \quad (2) i = 2, \dots, n: l_{i1} = a_{i1} / u_{11}; \quad n-1 \\
 (3) & r = 2, \dots, n-1: i = r, \dots, n: u_{ri} = a_{ri} - \sum_{k=1}^{r-1} l_{rk} u_{ki}; \quad (r-1)(n-r+1) \\
 & i = r+1, \dots, n: l_{ir} = (a_{ir} - \sum_{k=1}^{r-1} l_{ik} u_{kr}) / u_{rr}; \quad r(n-r) \\
 (3') & u_{nn} = a_{nn} - \sum_{k=1}^{n-1} l_{nk} u_{kn}; \quad n-1 \\
 (4) & y_1 = b_1, \quad i = 2, \dots, n: y_i = b_i - \sum_{k=1}^{i-1} l_{ik} y_k; \quad i-1 \\
 (5) & x_n = y_n / u_{nn}, \quad i = n-1, \dots, 1: x_i = (y_i - \sum_{k=i+1}^n u_{ik} x_k) / u_{ii}; \quad n-i+1 \\
 & 2n-1 + \sum_{r=2}^{n-1} [(r-1)(n-r+1) + r(n-r)] + \sum_{i=2}^n (i-1) + \sum_{i=1}^{n-1} (n-i+1)
 \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

451

直接三角分解解方程组(计算量)

通过计算可以得到

$$\begin{aligned}
 & 2n-1 + \sum_{r=2}^{n-1} [(r-1)(n-r+1) + r(n-r)] + \sum_{i=2}^n (i-1) + \sum_{i=1}^{n-1} (n-i+1) \\
 & = \frac{n^3}{3} + n^2 - \frac{n}{3}
 \end{aligned}$$

该计算量和Gauss消去法解方程组的计算量是完全一致的。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

452

直接三角分解法

前面所述的直接求矩阵的三角分解方法称为Doolittle分解。

注:(1)教材中介绍了直接三角分解方法在程序计算时节省存储的方法,这一类方法现在已经不具备重要的意义。

(2)类似于选主元的Gauss消去法,直接三角分解也有选主元的方法。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

453

对称矩阵的三角分解

对于一般的矩阵,在一定条件下可以实现其LU分解。

对于对称矩阵,这种分解没有利用其对称性,计算量不是最省的。我们下面来利用对称性,讨论其三角分解的算法。

在实际应用中,对称正定矩阵是广泛出现的,因此讨论其三角分解有着实用价值。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

454

对称矩阵的三角分解

设A是对称矩阵,且其所有顺序主子式均不为零,于是A有如下的三角分解

$$A = \begin{bmatrix} 1 & & & \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \dots & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ & u_{22} & \dots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

455

对称矩阵的三角分解

将上面分解中的U再进行分解

$$U = \begin{bmatrix} u_{11} & & & \\ & u_{22} & & \\ & & \ddots & \\ & & & u_{nn} \end{bmatrix} \begin{bmatrix} 1 & \frac{u_{12}}{u_{11}} & \dots & \frac{u_{1n}}{u_{11}} \\ & 1 & \dots & \vdots \\ & & \ddots & \frac{u_{n-1,n}}{u_{n-1,n-1}} \\ & & & 1 \end{bmatrix} = DU_0$$

将U分解为对角阵D与单位上三角阵U₀的乘积。

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

456

对称矩阵的三角分解

于是 $A=LU=LDU_0$.

根据 A 的对称性 $A=A^T=U_0^T(DL^T)$.

由分解的唯一性 (U_0^T 是单位下三角矩阵), 有 $U_0^T=L$.

因此 $A=LDL^T$.

定理7 设 A 为 n 阶对称矩阵, 其所有顺序主子式均不为零, 则 A 可唯一的分解为 $A=LDL^T$, 其中 L 为单位下三角矩阵, D 为对角阵.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

457

Cholesky分解

若 A 为正定矩阵, A 有上述形式的分解 $A=LDL^T$.

设 d_i 为 D 的第 i 个对角线元素, D_i 为 A 的顺序主子式下面说明 D 的对角元均为正数.

由于 L 为单位下三角阵, 容易证明

$D_i=d_1d_2\cdots d_i$ (将 A, L, D 写为分块矩阵的形式来证, 或根据 P169 推论)

因此 $d_1=D_1, d_i=D_i/D_{i-1}>0 (i=2, 3, \dots, n-1)$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

458

Cholesky分解

因此

$$D = \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \end{bmatrix}$$

$$= \begin{bmatrix} \sqrt{d_1} & & & \\ & \sqrt{d_2} & & \\ & & \ddots & \\ & & & \sqrt{d_n} \end{bmatrix} \begin{bmatrix} \sqrt{d_1} & & & \\ & \sqrt{d_2} & & \\ & & \ddots & \\ & & & \sqrt{d_n} \end{bmatrix} = D^{\frac{1}{2}} D^{\frac{1}{2}}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

459

Cholesky分解

因此有 $A=LDL^T=LD^{\frac{1}{2}}D^{\frac{1}{2}}L^T=(LD^{\frac{1}{2}})(LD^{\frac{1}{2}})^T=L_1L_1^T$.

定理8 如果 A 为 n 阶对称正定矩阵, 则存在一个实的非奇异下三角阵 L 使得 $A=LL^T$, 当限定 L 的对角元素为正时, 这种分解是唯一的.

注: 这一分解称为对称正定阵的 Cholesky 分解. 利用顺序主子式的值可以证明唯一性.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

460

平方根法

$$A = \begin{bmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & l_{21} & \cdots & l_{n1} \\ & l_{22} & \cdots & l_{n2} \\ & & \ddots & \vdots \\ & & & l_{nn} \end{bmatrix}$$

设 $i \geq j$, 根据 A 的 Cholesky 分解式,

$$a_{ij} = \sum_{k=1}^n l_{ik} l_{jk} = \sum_{k=1}^{j-1} l_{ik} l_{jk} + l_{ij} l_{ij} + \sum_{k=j+1}^n l_{ik} l_{jk}$$

$j < k$ 时, $l_{jk}=0$, 因此

$$a_{ij} = \sum_{k=1}^{j-1} l_{ik} l_{jk} + l_{ij} l_{ij}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

461

平方根法

$$a_{ij} = \sum_{k=1}^{j-1} l_{ik} l_{jk} + l_{ij} l_{ij}$$

在上式中, 令 $i=j$ 得 $a_{jj} = \sum_{k=1}^{j-1} l_{jk}^2 + l_{jj}^2$

设 $l_{ii} \geq 0$, 得

$$l_{jj} = (a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2)^{\frac{1}{2}}$$

在 $i=j+1, \dots, n$ 时有

$$l_{ij} = (a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}) / l_{jj}$$

对 $j=1, 2, \dots, n$ 用上面公式, 就得 Cholesky 分解.

乘除法计算量: 约为 $n^3/6$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

462

改进的平方根法

在平方根法中,需要进行开方计算,工作量较大. 为避免开方运算,我们常计算 LDL^T 分解.

$$A = \begin{bmatrix} 1 & & & \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \end{bmatrix} \begin{bmatrix} 1 & l_{21} & \cdots & l_{n1} \\ & 1 & \cdots & l_{n2} \\ & & \ddots & \vdots \\ & & & 1 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

463

改进的平方根法

LDL^T 分解得计算公式为:对 $i=1,2,\dots,n$,

$$l_{ij} = (a_{ij} - \sum_{k=1}^{j-1} l_{ik} d_k l_{jk}) / d_j \quad (j=1,2,\dots,i-1) \quad d_i = a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2 d_k$$

在上面得计算中,需重复计算 $l_{ik} d_k$,我们可以设 $t_{ij} = l_{ij} d_j$, 得到下面得公式:对 $i=1,2,\dots,n$,

$$(1) t_{ij} = a_{ij} - \sum_{k=1}^{j-1} t_{ik} l_{jk} \quad (j=1,2,\dots,i-1)$$

$$(2) l_{ij} = t_{ij} / d_j \quad (j=1,2,\dots,i-1)$$

$$(3) d_i = a_{ii} - \sum_{k=1}^{i-1} t_{ik} l_{ik}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

464

§ 5 向量和矩阵的范数

定义1 (1) 设 $x=(x_1, x_2, \dots, x_n)^T, y=(y_1, y_2, \dots, y_n)^T \in R^n$ (或 C^n). 将实数

$$(x, y) = y^T x = \sum_{i=1}^n x_i y_i$$

或复数

$$(x, y) = y^H x = \sum_{i=1}^n x_i \bar{y}_i$$

称为向量 x, y 的数量积(内积).

(2) 将非负实数 $\|x\|_2 = (x, x)^{\frac{1}{2}} = (\sum_{i=1}^n |x_i|^2)^{\frac{1}{2}}$

称为 x 的欧式范数(2范数).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

465

内积的性质

定理10 设 $x, y \in R^n$ (或 C^n), 则

1. $(x, x) = 0$ 当且仅当 $x=0$ 时成立;
2. $(\alpha x, y) = (x, \alpha y) = \alpha (x, y)$, α 为实数 (或 $(\alpha x, y) = (x, \alpha y) = \bar{\alpha} (x, y)$, α 为复数);
3. $(x, y) = (y, x)$ (或 $(x, y) = \overline{(y, x)}$);
4. $(x_1 + x_2, y) = (x_1, y) + (x_2, y)$;
5. (Cauchy-Schwarz 不等式) $|(x, y)| \leq \|x\|_2 \|y\|_2$, 等式当且仅当 x 与 y 线性相关时成立;
6. 三角不等式 $\|x+y\|_2 \leq \|x\|_2 + \|y\|_2$;

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

466

向量的范数

定义2 如果向量 $x \in R^n$ (或 C^n)的某个实值函数 $N(x) = \|x\|$, 满足条件

(1) $\|x\| \geq 0$ ($\|x\| = 0$ 当且仅当 $x=0$) (正定性);

(2) $\|\alpha x\| = |\alpha| \|x\|$ (对于任意 $\alpha \in R$ 或 C);

(3) $\|x+y\| \leq \|x\| + \|y\|$ (三角不等式);

则称 $N(x)$ 为 R^n (或 C^n)上的一个向量范数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

467

常用的向量范数

1-范数 $\|x\|_1 = \sum_{i=1}^n |x_i|$

2-范数(欧式范数) $\|x\|_2 = (\sum_{i=1}^n |x_i|^2)^{\frac{1}{2}}$

∞ -范数 $\|x\|_{\infty} = \max_{1 \leq i \leq n} |x_i|$

p -范数 $\|x\|_p = (\sum_{i=1}^n |x_i|^p)^{\frac{1}{p}}$

∞ -范数是 p -范数的极限 $\|x\|_{\infty} = \lim_{p \rightarrow \infty} \|x\|_p$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

468

常用的向量范数

对向量 $x=(1,-2,3)^T$, 有

$$\|x\|_1 = 6,$$

$$\|x\|_2 = \sqrt{14} \approx 3.74166,$$

$$\|x\|_3 = \sqrt[3]{36} \approx 3.30193,$$

$$\|x\|_\infty = 3.$$

$\|x\|_p$ 是 p 的单调递减函数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

469

向量范数的性质

定理11 (范数的连续性) 设非负函数 $N(x)=\|x\|$ 为 R^n 上任一向量范数, 则 $N(x)$ 是 x 分量 x_1, x_2, \dots, x_n 的连续函数

定理12 (向量范数的等价性) 设 $\|x\|_s, \|x\|_t$ 为 R^n 上向量的任意两种范数, 则存在常数 $c_1, c_2 > 0$, 使得对任意 $x \in R^n$ 有 $c_1 \|x\|_s \leq \|x\|_t \leq c_2 \|x\|_s$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

470

向量序列的极限

设 $\{x^{(k)}\}$ 是 R^n 中的向量序列, $x^* \in R^n$, 如果

$$\lim_{k \rightarrow \infty} x_i^{(k)} = x_i^* (i=1, 2, \dots, n)$$

则称 $\{x^{(k)}\}$ 收敛于 x^* , 记为 $\lim_{k \rightarrow \infty} x^{(k)} = x^*$.

定理13 设 $\|\cdot\|$ 为向量的任意一种范数, 则

$$\lim_{k \rightarrow \infty} x^{(k)} = x^* \Leftrightarrow \lim_{k \rightarrow \infty} \|x^{(k)} - x^*\| = 0.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

471

矩阵范数

定义4 如果矩阵 $A \in R^{n \times n}$ 的某个实值函数 $N(A)=\|A\|$, 满足条件

- (1) $\|A\| \geq 0$ ($\|A\|=0$ 当且仅当 $A=0$) (正定性);
- (2) $\|cA\| = |c| \cdot \|A\|$ (对于任意 $c \in R$);
- (3) $\|A+B\| \leq \|A\| + \|B\|$ (三角不等式);
- (4) $\|AB\| \leq \|A\| \cdot \|B\|$ (相容性)

则称 $N(A)$ 为 $R^{n \times n}$ 上的一个矩阵范数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

472

Frobenius范数

$$\|A\|_F = \left(\sum_{i,j=1}^n a_{ij}^2 \right)^{\frac{1}{2}}$$

将 $R^{n \times n}$ 中的元素看成 n^2 维的向量, 再取向量的 2-范数, 就得到上述的矩阵范数, 称为 Frobenius 范数, 简称 F-范数.

F-范数满足矩阵范数定义的前三条性质是容易证明的 (根据 n^2 维的向量范数的性质), 而相容性的证明比较困难, 在此略去.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

473

矩阵的算子范数

定义5 设 $x \in R^n, A \in R^{n \times n}$, 给出一种向量范数 $\|x\|_v$ ($v=1, 2$ 或 ∞), 相应地定义一个矩阵的非负函数

$$\|A\|_v = \max_{x \neq 0} \frac{\|Ax\|_v}{\|x\|_v}.$$

可以验证 $\|A\|_v$ 是矩阵范数, 称为 A 的算子范数.

算子范数的定义可以改写为等价于

$$\|A\|_v = \max_{\|x\|_v=1} \|Ax\|_v.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

474

矩阵的算子范数

定理14 设 $\|x\|_v$ 是 R^n 上一个向量范数,则 $\|A\|_v$ 是 $R^n \times R^n$ 上矩阵的范数,且对任意的 $x \in R^n$ 满足相容条件

$$\|Ax\|_v \leq \|A\|_v \|x\|_v.$$

证明 根据定义 $\|A\|_v = \max_{x \neq 0} \frac{\|Ax\|_v}{\|x\|_v}.$

因此,对任意的 $x(\neq 0) \in R^n$, $\|A\|_v \geq \frac{\|Ax\|_v}{\|x\|_v},$
从而 $\|Ax\|_v \leq \|A\|_v \|x\|_v.$

矩阵的算子范数

关于范数定义的正定性和齐次性,证明较简单,三角不等式的证明作为作业.

下面证明第四条性质,即 $\|AB\| \leq \|A\| \cdot \|B\|.$

由于 $\|ABx\|_v \leq \|A\| \cdot \|Bx\|_v \leq \|A\|_v \cdot \|B\|_v \cdot \|x\|_v$

因此, $\|AB\|_v = \max_{\|x\|_v=1} \|ABx\|_v \leq \|A\|_v \cdot \|B\|_v.$

常见的算子范数

定理15 设 $A \in R^{n \times n}$, 则

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

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

(3) $\|A\|_2 = \sqrt{\lambda_{\max}(A^T A)}$ (称为 A 的2-范数)

其中 $\lambda_{\max}(A^T A)$ 表示 $A^T A$ 的最大特征值.

常见的算子范数

证明 我们仅证明2,3

$$(2) \|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$$

我们证明两个方面:

<1>若 $x \in R^n, \|x\|_1=1$, 则 $\|Ax\|_1 \leq \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$

<2>存在 $x \in R^n, \|x\|_1=1$, 使得 $\|Ax\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$

于是 $\max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}| = \max_{\|x\|_1=1} \|Ax\|_1 = \|A\|_1.$

常见的算子范数

<1>若 $x \in R^n, \|x\|_1=1$, 则 $\|Ax\|_1 \leq \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$

设 $\sum_{i=1}^n |a_{ij_0}| = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$, 若 $\|x\|_1=1$, 则

$$\begin{aligned} \|Ax\|_1 &= \sum_{i=1}^n \left| \sum_{j=1}^n a_{ij} x_j \right| \leq \sum_{i=1}^n \sum_{j=1}^n |a_{ij}| |x_j| = \sum_{j=1}^n \sum_{i=1}^n |a_{ij}| |x_j| \\ &= \sum_{j=1}^n \left(|x_j| \sum_{i=1}^n |a_{ij}| \right) \leq \sum_{j=1}^n \left(|x_j| \sum_{i=1}^n |a_{ij_0}| \right) \\ &= \left(\sum_{i=1}^n |a_{ij_0}| \right) \cdot \left(\sum_{j=1}^n |x_j| \right) = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}| \end{aligned}$$

常见的算子范数

<2>存在 $x \in R^n, \|x\|_1=1$, 使得 $\|Ax\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$

仍然设 $\sum_{i=1}^n |a_{ij_0}| = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$,

令 e_{j_0} 为第 j_0 个分量为1, 其余分量为零的向量, 则

$$\|Ae_{j_0}\|_1 = \sum_{i=1}^n \left| \sum_{j=1}^n a_{ij} e_{j_0} \right| = \sum_{i=1}^n |a_{ij_0}| = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|.$$

常见的算子范数

$$(3) \|A\|_2 = \sqrt{\lambda_{\max}(A^T A)}$$

仍然证明两个方面:

<1> 若 $x \in R^n, \|x\|_2=1$, 则 $\|Ax\|_2 \leq \sqrt{\lambda_{\max}(A^T A)}$

<2> 存在 $x \in R^n, \|x\|_2=1$, 使得 $\|Ax\|_2 = \sqrt{\lambda_{\max}(A^T A)}$

于是 $\sqrt{\lambda_{\max}(A^T A)} = \max_{\|x\|_2=1} \|Ax\|_2 = \|A\|_2$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

481

常见的算子范数

容易证明, $A^T A$ 为半正定实对称矩阵, 有 n 个相互正交的标准特征向量 u_1, u_2, \dots, u_n , 假设其对应的特征值 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ 是从大到小排列的.

<1> 若 $x \in R^n, \|x\|_2=1$, 则 $\|Ax\|_2 \leq \sqrt{\lambda_{\max}(A^T A)}$

$$\begin{aligned} \text{设 } x &= \sum_{i=1}^n c_i u_i, \text{ 则 } \|Ax\|_2^2 = (Ax)^T (Ax) = x^T A^T A x \\ &= \left(\sum_{i=1}^n c_i u_i \right)^T \left(\sum_{j=1}^n c_j A^T A u_j \right) = \left(\sum_{i=1}^n c_i u_i \right)^T \left(\sum_{j=1}^n c_j \lambda_j u_j \right) = \sum_{i=1}^n c_i^2 \lambda_i \\ &\leq \lambda_1 \sum_{i=1}^n c_i^2 = \lambda_1 \quad (\|x\|_2^2 = \sum_{i=1}^n c_i^2 = 1) \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

482

常见的算子范数

<2> 存在 $x \in R^n, \|x\|_2=1$, 使得 $\|Ax\|_2 = \sqrt{\lambda_{\max}(A^T A)}$

对向量 u_1 , 有 $\|u_1\|_2=1$, 且

$$\|Au_1\|_2^2 = u_1^T A^T A u_1 = u_1^T \lambda_1 u_1 = \lambda_1 u_1^T u_1 = \lambda_1$$

注: 对于复矩阵, 有 $\|A\|_2 = \sqrt{\lambda_{\max}(A^H A)}$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

483

常见的算子范数

$$\text{对矩阵 } A = \begin{bmatrix} 1 & -2 \\ -3 & 4 \end{bmatrix},$$

$$\|A\|_1 = \max\{4, 6\} = 6, \quad \|A\|_{\infty} = \max\{3, 7\} = 7,$$

$$\|A\|_F = \sqrt{1+4+9+16} = \sqrt{30} \approx 5.477,$$

$$A^T A = \begin{bmatrix} 10 & -14 \\ -14 & 20 \end{bmatrix}, \quad \lambda(A^T A) = 15 \pm \sqrt{221},$$

$$\|A\|_2 = \sqrt{15 + \sqrt{221}} \approx 5.465.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

484

谱半径

定义6 设 $A \in R^{n \times n}$ 的特征值为 $\lambda_i (i=1, 2, \dots, n)$, 称

$$\rho(A) = \max_{1 \leq i \leq n} |\lambda_i|$$

为 A 的谱半径.

定理16 设 $A \in R^{n \times n}$, 则 $\rho(A) \leq \|A\|$, 即 A 的谱半径不超过 A 的任何一种算子范数(或F-范数).

证明思路

$$Ax = \lambda x \Rightarrow \lambda \cdot \|x\| = \|\lambda x\| = \|Ax\| \leq \|A\| \cdot \|x\| \Rightarrow \lambda \leq \|A\|$$

注: $\|A\|_2 \leq \|A\|_F$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

485

谱半径

作业: 证明下面的定理

定理17 如果 $A \in R^{n \times n}$ 为对称矩阵, 则 $\rho(A) = \|A\|_2$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

486

算子范数

定理18 如果 $\|B\| < 1$, 则 $I \pm B$ 为非奇异矩阵, 且

$$\|(I \pm B)^{-1}\| \leq \frac{1}{1 - \|B\|}.$$

其中 $\|\cdot\|$ 为矩阵的算子范数.

证明 仅就 $I-B$ 来证.

若 $\det(I-B)=0$, 则存在非零向量 x , 使得 $Bx=x$.

于是 $\|B\| = \max_{\|x\| \neq 0} \frac{\|Bx\|}{\|x\|} \geq 1$, 矛盾. 因此 $\det(I-B) \neq 0$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

487

算子范数

又 $(I-B)(I-B)^{-1} = I$, 得

$$(I-B)^{-1} = I + B(I-B)^{-1},$$

$$\|(I-B)^{-1}\| \leq \|I\| + \|B(I-B)^{-1}\|$$

$$\leq 1 + \|B\| \|(I-B)^{-1}\|$$

因此

$$\|(I-B)^{-1}\| \leq \frac{1}{1 - \|B\|}.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

488

§ 6 误差分析

对于数值分析中的许多算法, 我们有必要研究问题中的数据出现小的误差对于结果的影响.

本节研究线性方程组 $Ax=b$ 中系数矩阵 A 与右端向量 b 出现小的误差时, 求出的解会出现怎样的误差.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

489

最小二乘问题算例

在区间 $[0, 1]$ 求函数 $f(x) = \sin \pi x$ 的最佳平方逼近二次函数.

解: 设二次函数为 $P_2(x) = a_2 x^2 + a_1 x + a_0$ 法方程组为

$$a_0 \int_0^1 1 dx + a_1 \int_0^1 x dx + a_2 \int_0^1 x^2 dx = \int_0^1 \sin \pi x dx$$

$$a_0 \int_0^1 x dx + a_1 \int_0^1 x^2 dx + a_2 \int_0^1 x^3 dx = \int_0^1 x \sin \pi x dx$$

$$a_0 \int_0^1 x^2 dx + a_1 \int_0^1 x^3 dx + a_2 \int_0^1 x^4 dx = \int_0^1 x^2 \sin \pi x dx$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

490

最小二乘问题算例

计算得到

$$a_0 + \frac{1}{2}a_1 + \frac{1}{3}a_2 = \frac{2}{\pi} \quad \frac{1}{2}a_0 + \frac{1}{3}a_1 + \frac{1}{4}a_2 = \frac{1}{\pi} \quad \frac{1}{3}a_0 + \frac{1}{4}a_1 + \frac{1}{5}a_2 = \frac{\pi^2 - 4}{\pi^3}$$

解得

$$a_0 = \frac{12\pi^2 - 120}{\pi^3} \approx -0.050465,$$

$$a_1 = -a_2 = \frac{720 - 60\pi^2}{\pi^3} \approx 4.12251,$$

因此 $P_2(x) = -4.12251x^2 + 4.12251x - 0.050456$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

491

最小二乘问题算例

$$a_0 + \frac{1}{2}a_1 + \frac{1}{3}a_2 = \frac{2}{\pi} \quad \frac{1}{2}a_0 + \frac{1}{3}a_1 + \frac{1}{4}a_2 = \frac{1}{\pi} \quad \frac{1}{3}a_0 + \frac{1}{4}a_1 + \frac{1}{5}a_2 = \frac{\pi^2 - 4}{\pi^3}$$

如果我们采用3位有效数字计算, 上面的方程组变为

$$1.00a_0 + 0.500a_1 + 0.333a_2 = 0.637$$

$$0.500a_0 + 0.333a_1 + 0.250a_2 = 0.318$$

$$0.333a_0 + 0.250a_1 + 0.200a_2 = 0.189$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

492

最小二乘问题算例

$$\begin{pmatrix} 1.00 & 0.500 & 0.333 & 0.637 \\ 0.500 & 0.333 & 0.250 & 0.318 \\ 0.333 & 0.250 & 0.200 & 0.189 \end{pmatrix}$$

$$\begin{aligned} r_1 \cdot (-0.500) + r_2 &\rightarrow r_2 \begin{pmatrix} 1.00 & 0.500 & 0.333 & 0.637 \\ 0 & 0.083 & 0.083 & -0.001 \\ 0 & 0.083 & 0.089 & -0.023 \end{pmatrix} \\ r_1 \cdot (-0.333) + r_3 &\rightarrow r_3 \end{aligned}$$

$$\begin{aligned} r_2 \cdot (-1.000) + r_3 &\rightarrow r_3 \begin{pmatrix} 1.00 & 0.500 & 0.333 & 0.637 \\ 0 & 0.083 & 0.083 & -0.001 \\ 0 & 0 & 0.006 & -0.022 \end{pmatrix} \end{aligned}$$

于是 $a_2 = -0.022/0.006 = -3.67$,
 $a_1 = (-0.001 - 0.083a_2)/0.083 = 3.65$,
 $a_0 = (0.637 - 0.500a_1 - 0.333a_2) = 0.027$. 误差相当大!

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

493

病条件方程组

在上面的线性方程组中,系数矩阵和右端向量出现了微小误差(舍入误差是很小的),引起了方程组的解的巨大变化,这样的方程组称为病条件方程组.(P192 定义7)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

494

右端向量的扰动

设方程组 $Ax=b$ 的精确解为 x , 现在假设 A 精确(没有误差), 右端向量 b 有误差 δb , 带误差的方程组的解为 $x+\delta x$, 即 $A(x+\delta x)=b+\delta b$,

由于 $Ax=b$, 易得 $\delta x=A^{-1}\delta b$.

取算子范数 $\|\delta x\| \leq \|A^{-1}\| \|\delta b\|$

考虑相对误差

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\|b\|}{\|x\|} \frac{\|A^{-1}\| \|\delta b\|}{\|b\|}$$

$$= \frac{\|Ax\|}{\|x\|} \frac{\|A^{-1}\| \|\delta b\|}{\|b\|} \leq \|A\| \|A^{-1}\| \frac{\|\delta b\|}{\|b\|}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

495

右端向量的扰动

因此,在右端向量有扰动时,解的相对误差与右端向量的相对误差的关系为

$$\frac{\|\delta x\|}{\|x\|} \leq \|A\| \|A^{-1}\| \frac{\|\delta b\|}{\|b\|}$$

即 $\|A^{-1}\| \|A\|$ 为相对误差放大倍数的界.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

496

系数矩阵的扰动

设方程组 $Ax=b$ 的精确解为 x , 现在假设 b 精确(没有误差), 系数矩阵 A 有误差 δA , 带误差的方程组的解为 $x+\delta x$, 即 $(A+\delta A)(x+\delta x)=b$,

由于 $Ax=b$, 易得

$$(A+\delta A)\delta x = -\delta A x.$$

如果 δA 不受限制, $A+\delta A$ 可能奇异.

由于 $A+\delta A=A(I+A^{-1}\delta A)$.

在 $\|A^{-1}\delta A\|<1$ 时, $(I+A^{-1}\delta A)^{-1}$ 存在.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

497

系数矩阵的扰动

$$\delta x = -(I+A^{-1}\delta A)^{-1} A^{-1}(\delta A)x$$

取算子范数

$$\|\delta x\| \leq \frac{\|A^{-1}\| \|\delta A\| \|x\|}{1-\|A^{-1}\delta A\|}$$

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\|A^{-1}\| \|A\| \frac{\|\delta A\|}{\|A\|}}{1-\|A^{-1}\delta A\|} \leq \frac{\|A^{-1}\| \|A\| \frac{\|\delta A\|}{\|A\|}}{1-\|A^{-1}\| \|A\| \frac{\|\delta A\|}{\|A\|}}$$

(右边的不等式必须假设 $\|A^{-1}\| \|\delta A\|<1$)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

498

系数矩阵的扰动

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\|A^{-1}\| \|\delta A\|}{1 - \|A^{-1}\| \|\delta A\|} \frac{\|\delta A\|}{\|A\|}$$

由上式,解的相对误差较系数矩阵的误差放大的倍数的界比 $\|A^{-1}\| \|A\|$ 要大一些.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

499

同时扰动

在右端向量与系数矩阵同时扰动的情况下,有下面的结果

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\|A^{-1}\| \|\delta A\|}{1 - \|A^{-1}\| \|\delta A\|} \left(\frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|b\|} \right)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

500

条件数

从上面几种情况的分析可以看出, $\|A^{-1}\| \|A\|$ 刻划了解对原始数据变化的灵敏程度,即刻划了方程组的病态程度.

定义8 设 A 为非奇异矩阵,称数

$\text{cond}(A)_v = \|A^{-1}\|_v \|A\|_v$ ($v=1,2$ 或 ∞)为矩阵的条件数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

501

条件数

最常用的条件数是无穷条件数与2条件数(称为谱条件数).

对谱条件数有 $\text{cond}(A)_2 = \sqrt{\frac{\lambda_{\max}(A^T A)}{\lambda_{\min}(A^T A)}}$

A 为对称矩阵时 $\text{cond}(A)_2 = \frac{|\lambda_1|}{|\lambda_n|}$

其中 λ_1, λ_n 分别为 A 的绝对值最大与绝对值最小的特征值

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

502

条件数的性质

1 若 A 为非奇异矩阵, $\text{cond}(A)_v \geq 1$

$\text{cond}(A)_v = \|A^{-1}\|_v \|A\|_v \geq \|A^{-1}A\|_v \geq \|I\|_v = 1$

2 若 A 为非奇异矩阵,对常数 $c \neq 0$,有 $\text{cond}(cA)_v = \text{cond}(A)_v$

3 若 A 为正交矩阵,则 $\text{cond}(A)_2 = 1$;若 A 为非奇异矩阵, R 为正交矩阵,则

$\text{cond}(RA)_2 = \text{cond}(AR)_2 = \text{cond}(A)_2$.

2,3的证明作为作业.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

503

Hilbert矩阵的条件数

$$H_n = \begin{bmatrix} 1 & \frac{1}{2} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \cdots & \frac{1}{2n-1} \end{bmatrix}$$

上述矩阵称为Hilbert矩阵.

下面计算 H_3 的条件数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

504

Hilbert矩阵的条件数

$$H_3 = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{bmatrix} \quad H_3^{-1} = \begin{bmatrix} 9 & -36 & 30 \\ -36 & 192 & -180 \\ 30 & -180 & 180 \end{bmatrix}$$

$$\|H_3\| = \frac{11}{6}, \quad \|H_3^{-1}\| = 408,$$

因此 $\text{cond}(H_3)_\infty = 748$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

505

Hilbert矩阵的条件数

前8个Hilbert矩阵的条件数为

1	1
2	27
3	748
4	2.84e4
5	9.44e5
6	2.91e7
7	9.85e8
8	3.39e10

可见,随着阶数的增加,Hilbert矩阵的条件数增长很快.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

506

Hilbert矩阵的条件数

在前面的最小二乘问题中,系数矩阵与右端向量的相对误差为

$$\frac{\|\delta A\|_\infty}{\|A\|_\infty} = 1.82 \cdot 10^{-4}, \quad \frac{\|\delta b\|_\infty}{\|b\|_\infty} = 5.97 \cdot 10^{-4},$$

$$\text{解的相对误差为 } \frac{\|\delta x\|_\infty}{\|x\|_\infty} = 0.115$$

相对误差大约扩大了200倍.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

507

病条件的判断

若根据条件数的定义来判断病条件,必须计算 A^{-1} ,这通常计算量较大(如果是病条件, A^{-1} 本身计算就有较大误差),一般改用别的方法大致地判断是否病条件.

(1)三角约化时,(尤其是选主元情况下)出现小主元,一般可以判断矩阵病条件.

(2)矩阵的绝对值最大特征值与绝对值最小特征值的比较大时,矩阵病条件.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

508

病条件的判断

(3)矩阵的行列式很小,或某些行近似线性相关,则矩阵可能病态.

(4)矩阵的各元素数量级相差很大,且无一定规则,则矩阵可能病态.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

509

第八章 解线性方程组的迭代法

§ 1 引言

在处理一元方程 $f(x)=0$ 时,我们将其转化为 $x=\varphi(x)$ 的形式,然后用不动点迭代的方法进行求解.

对于线性方程组 $Ax=b$,我们也可以将其转化为类似的形式: $x=Bx+f$,

任取初始向量 $x^{(0)}$,令 $x^{(k+1)}=Bx^{(k)}+f(k=0,1,\dots)$,则得到一个向量的序列 $\{x^{(k)}\}$.

若该序列收敛于向量 x^* ,对 $x^{(k+1)}=Bx^{(k)}+f$ 两边取极限得到 $x^*=Bx^*+f$,即 x^* 是方程组的解.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

511

§ 2 Jacobi迭代法与Gauss-Seidel迭代法

对于方程组
$$\begin{cases} 8x_1 - 3x_2 + 2x_3 = 20 \\ 4x_1 + 11x_2 - x_3 = 33 \\ 6x_1 + 3x_2 + 12x_3 = 36 \end{cases}$$

我们将其改写为
$$\begin{cases} x_1 = \frac{1}{8}(3x_2 - 2x_3 + 20) \\ x_2 = \frac{1}{11}(-4x_1 + x_3 + 33) \\ x_3 = \frac{1}{12}(-6x_1 - 3x_2 + 36) \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

512

Jacobi迭代法

写成矩阵的形式为 $x=B_0x+f$,其中

$$B_0 = \begin{bmatrix} 0 & \frac{3}{8} & -\frac{2}{8} \\ -\frac{4}{11} & 0 & \frac{1}{11} \\ -\frac{6}{12} & -\frac{3}{12} & 0 \end{bmatrix} \quad f = \begin{bmatrix} \frac{20}{8} \\ \frac{33}{11} \\ \frac{36}{12} \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

513

Jacobi迭代法

利用 $x^{(k+1)}=Bx^{(k)}+f$ 进行迭代,得到结果如下

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$\ x^{(k)} - x^{(k-1)}\ _\infty$
0	0	0	0	
1	2.5	3.0	3.0	3.0
2	2.87500000	2.36363636	1.00000000	2.0
8	3.00020012	2.00063786	0.99983051	3.30e-3
9	3.00028157	1.99991182	0.99974048	7.26e-4
10	3.00003181	1.99987402	0.99988126	2.50e-4

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

514

Jacobi迭代法

从上表可以看出,迭代序列逐步逼近方程组的精确解 $(3,2,1)^T$.

注:在迭代中,我们不可能得到 $x^{(k)}$ 和精确解之间的误差,一般我们用 $\|x^{(k)} - x^{(k-1)}\|$ (通常用无穷范数)的值来判断是否终止迭代.

在上面的例子中,我们将第 i 个方程变形为左边是 x_i ,右边是其它分量和常数的线性组合,然后进行迭代,这一方法称为Jacobi迭代.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

515

Jacobi迭代法

一般的,对于方程组 $Ax=b$,设 A 非奇异且 $a_{ii} \neq 0(i=1,2,\dots,n)$,将 A 改写为 $A=D-L-U$,其中

$$D = \begin{bmatrix} a_{11} & & & & \\ & a_{22} & & & \\ & & a_{33} & & \\ & & & \ddots & \\ & & & & a_{nn} \end{bmatrix} \quad L = - \begin{bmatrix} 0 & & & & \\ a_{21} & 0 & & & \\ \vdots & a_{32} & \ddots & & \\ a_{n1} & a_{n2} & & a_{nn} & 0 \end{bmatrix}$$

$$U = - \begin{bmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1n} \\ & 0 & a_{23} & \cdots & a_{2n} \\ & & 0 & \ddots & a_{3n} \\ & & & \ddots & \\ & & & & 0 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

516

Jacobi迭代法

将方程组改写为

$$Dx=(L+U)x+b$$

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

令 $B_0=D^{-1}(L+U)$ (称为Jacobi迭代矩阵), $f=D^{-1}b$,
上式简记为 $x=B_0x+f$.

我们得到Jacobi迭代公式 $x^{(k+1)}=B_0x^{(k)}+f$.

写成分量的形式为
$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j^{(k)} \right)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

517

Gauss-Seidel迭代法

$$\begin{cases} x_1^{(k+1)} = \frac{1}{8}(3x_2^{(k)} - 2x_3^{(k)} + 20) \\ x_2^{(k+1)} = \frac{1}{11}(-4x_1^{(k+1)} + x_3^{(k)} + 33) \\ x_3^{(k+1)} = \frac{1}{12}(-6x_1^{(k+1)} - 3x_2^{(k+1)} + 36) \end{cases}$$

在上面的例子中,
我们计算 $x_1^{(k+1)}$,用
的是第 k 步的 x_2, x_3 ;

计算 $x_2^{(k+1)}$,用的是第 k 步的 x_1, x_3 ,我们有理由
认为已经计算出的第 $k+1$ 步的 x_1 比第 k 步的
“好”.因此,我们应该用第 $k+1$ 步的 x_1 和第 k 步
的 x_3 来计算 x_2 .

类似地,我们也应该用新信息计算 x_3 .

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

518

Gauss-Seidel迭代法

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j^{(k)} \right)$$

我们可以将上面一般的Jacobi迭代公式改写为

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

这一迭代方法称为Gauss-Seidel迭代.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

519

Gauss-Seidel迭代法(算例)

对于方程组
$$\begin{cases} 8x_1 - 3x_2 + 2x_3 = 20 \\ 4x_1 + 11x_2 - x_3 = 33 \\ 6x_1 + 3x_2 + 12x_3 = 36 \end{cases}$$

其Gauss-Seidel迭代公式为

$$\begin{cases} x_1^{(k+1)} = \frac{1}{8}(3x_2^{(k)} - 2x_3^{(k)} + 20) \\ x_2^{(k+1)} = \frac{1}{11}(-4x_1^{(k+1)} + x_3^{(k)} + 33) \\ x_3^{(k+1)} = \frac{1}{12}(-6x_1^{(k+1)} - 3x_2^{(k+1)} + 36) \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

520

Gauss-Seidel迭代法(算例)

同样取 $x^{(0)}=(0,0,0)^T$,迭代结果如下

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$\ x^{(k)} - x^{(k-1)}\ _\infty$
0	0	0	0	
1	2.50000000	2.09090909	1.22727273	2.5
2	2.97272727	2.02892562	1.00413223	0.477
3	3.00981405	1.99680691	0.99589125	3.25e-2
4	2.99982978	1.99968838	1.00016302	9.98e-3
5	2.99984239	2.00007213	1.00006077	3.84e-4

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

521

Gauss-Seidel迭代法

一般而言,如果Jacobi迭代与Gauss-Seidel迭代
都收敛,那么Gauss-Seidel迭代收敛的速度比
Jacobi迭代收敛的速度快(不总是如此!).

不过,有时Jacobi迭代收敛,而Gauss-Seidel迭代
不收敛.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

522

Gauss-Seidel迭代法(矩阵表示)

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

上面的Gauss-Seidel迭代方法等价于

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

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

即

$$(D-L)x^{(k+1)} = b + Ux^{(k)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

523

Gauss-Seidel迭代法(矩阵表示)

$$(D-L)x^{(k+1)} = b + Ux^{(k)}$$

$$x^{(k+1)} = (D-L)^{-1} Ux^{(k)} + (D-L)^{-1} b$$

因此Gauss-Seidel迭代方法的矩阵形式为

$$x^{(k+1)} = Gx^{(k)} + f$$

其中 $G = (D-L)^{-1} U$ (称为Gauss-Seidel迭代矩阵),
 $f = (D-L)^{-1} b$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

524

§ 3 迭代法的收敛性

定义2 设有矩阵序列 $A_k = (a_{ij}^{(k)})_n (k=1,2,\dots)$ 及 $A = (a_{ij})_n$, 如果

$$\lim_{k \rightarrow \infty} a_{ij}^{(k)} = a_{ij} (i, j = 1, 2, \dots, n)$$

成立, 则称 $\{A_k\}$ 收敛于 A , 记为

定理1 $\lim_{k \rightarrow \infty} A_k = A$ 的充要条件是 $\|A_k - A\| \rightarrow 0 (k \rightarrow \infty)$.
 其中的范数是任何一种矩阵范数.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

525

迭代法的收敛性

对于矩阵 $A = \begin{bmatrix} \lambda & 1 \\ 0 & \lambda \end{bmatrix}$, 有 $A^k = \begin{bmatrix} \lambda^k & k\lambda^{k-1} \\ 0 & \lambda^k \end{bmatrix}$,

因此, 当 $|\lambda| < 1$ 时, $A^k \rightarrow \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} (k \rightarrow \infty)$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

526

迭代法收敛的条件(P204)

根据方程组的解 x^* , 可以定义误差向量

$$\varepsilon^{(k)} = x^{(k)} - x^*$$

由 $x^{(k+1)} = Bx^{(k)} + f$, $x^* = Bx^* + f$ 可以得到

$$\varepsilon^{(k+1)} = B\varepsilon^{(k)}.$$

因此 $\varepsilon^{(k)} = B^k \varepsilon^{(0)}$,

要迭代收敛, 必须 $\varepsilon^{(k)} \rightarrow 0$ (零向量), 即 $B^k \varepsilon^{(0)} \rightarrow 0$.

我们希望对任意 $\varepsilon^{(0)}$ 上式都成立, 必须

$B^k \rightarrow O$ (零矩阵).

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

527

迭代法收敛的条件

性质(习题6) $\lim_{k \rightarrow \infty} A_k = A$ 的充要条件是对于任意的向量 x 均有 $\lim_{k \rightarrow \infty} A_k x = Ax$.

证明思路 必要性: 根据下面的不等式得到

$$\|A_k x - Ax\| \leq \|A_k - A\| \|x\|$$

充分性: 取单位向量 e_1, e_2, \dots, e_n , 根据下式可证

$$\|A_k - A\|_1 = \max_{1 \leq i \leq n} \|(A_k - A)e_i\|$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

528

迭代法的收敛性

对于迭代格式 $x^{(k+1)} = Bx^{(k)} + f$, 要使它对于任意的初始向量 $x^{(0)}$ 都收敛, 必须 $B^k \rightarrow O (k \rightarrow \infty)$.

如果 B 有 n 个线性无关的特征向量 u_1, u_2, \dots, u_n , 对应的特征值为 $\lambda_1, \lambda_2, \dots, \lambda_n$.

对于任意的 n 维向量 x , x 可以写为

$$x = \sum_{i=1}^n \alpha_i u_i$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

529

迭代法的收敛性

由于 $Bu_i = \lambda_i u_i$,

可以证明 $B^k u_i = \lambda_i^k u_i (k=1, 2, \dots)$.

因此 $B^k x = \sum_{i=1}^n \alpha_i B^k u_i = \sum_{i=1}^n \alpha_i \lambda_i^k u_i$

若 $|\lambda_i| < 1 (i=1, 2, \dots, n)$, 显然当 $k \rightarrow \infty$ 时, $\lambda_i^k \rightarrow 0$,

对于任意向量 x , $B^k x = \sum_{i=1}^n \alpha_i \lambda_i^k u_i \rightarrow 0$.

因此 $B^k \rightarrow O$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

530

迭代法的收敛性

反之, 若 $B^k \rightarrow 0$, 则 $B^k u_i = \lambda_i^k u_i \rightarrow 0$, 因此, 对 $i=1, \dots, n$, 均有 $|\lambda_i| < 1$.

所以在 B 有 n 个线性无关的特征向量的条件下, $B^k \rightarrow O (k \rightarrow \infty)$ 的充要条件是 $\rho(B) < 1$.

事实上, 有下面的定理

定理2 设 $B = (b_{ij})_n$, 则 $B^k \rightarrow O (k \rightarrow \infty)$ 的充要条件是 $\rho(B) < 1$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

531

迭代法基本定理

定理3 设有方程组

$$x = Bx + f$$

对于任意的初始向量 $x^{(0)}$ 及任意的 f , 解此方程组的迭代法 ($x^{(k+1)} = Bx^{(k)} + f$) 收敛的充要条件是 $\rho(B) < 1$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

532

迭代法基本定理

在例1中, Jacobi 迭代矩阵为

$$B_0 = \begin{bmatrix} 0 & \frac{3}{8} & -\frac{2}{8} \\ -\frac{4}{11} & 0 & \frac{1}{11} \\ -\frac{6}{12} & -\frac{3}{12} & 0 \end{bmatrix}$$

B_0 的特征方程为

$$\det(\lambda I - B_0) = \lambda^3 + 0.034090909\lambda + 0.039772727 = 0,$$

解得 $\lambda_1 = -0.3082, \lambda_{2,3} = 0.1541 \pm 0.3245i$,

显然有 $\rho(B_0) < 1$, Jacobi 迭代收敛.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

533

迭代法的收敛速度

如果 B 有 n 个线性无关的特征向量 u_1, u_2, \dots, u_n , 对应的特征值为 $\lambda_1, \lambda_2, \dots, \lambda_n$. 设初始误差向量为

$$\varepsilon^{(0)} = \sum_{i=1}^n \alpha_i u_i$$

则第 k 步的误差向量为

$$\varepsilon^{(k)} = B^k \varepsilon^{(0)} = \sum_{i=1}^n \alpha_i B^k u_i = \sum_{i=1}^n \alpha_i \lambda_i^k u_i$$

若 $\rho(B) < 1$, 则 $\lambda_i^k \rightarrow 0$. 收敛速度可以用 $\max |\lambda_i|$ 即 $\rho(B)$ 来衡量.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

534

迭代法的收敛速度

$$\varepsilon^{(k)} = \sum_{i=1}^n \alpha_i \lambda_i^k u_i$$

一般,在要求误差小于 $\varepsilon=10^{-s}$ 时,我们直接要求

$$[\rho(B)]^k \leq 10^{-s}$$

即

$$k \geq \frac{s \ln 10}{-\ln \rho(B)}$$

有时称 $-\ln \rho(B)$ 为迭代法的收敛速度.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

535

迭代法的收敛性

特征值是比较难以计算的,因此,用 $\rho(B) < 1$ 这一条件来判断迭代是否收敛较为困难.

由于 $\rho(B) \leq \|B\|_v$ ($v=1, 2, \infty$, 或 F), 所以 $\|B\|_v < 1$ 是迭代收敛的一个充分条件.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

536

迭代法的收敛性

设 $\|B\|_v = q < 1$, 则

$$\begin{aligned} & \|x^{(k+1)} - x^{(k)}\|_v \\ &= \|(Bx^{(k)} + f) - (Bx^{(k-1)} + f)\|_v \\ &= \|B(x^{(k)} - x^{(k-1)})\|_v \\ &\leq \|B\|_v \|x^{(k)} - x^{(k-1)}\|_v \\ &= q \|x^{(k)} - x^{(k-1)}\|_v \end{aligned}$$

因此

$$\|x^{(k+1)} - x^{(k)}\|_v \leq q \|x^{(k)} - x^{(k-1)}\|_v$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

537

迭代法的收敛性

设 x^* 为方程组的解($x^* = Bx^* + f$)

$$\begin{aligned} & \|x^{(k+1)} - x^*\|_v \\ &= \|(Bx^{(k)} + f) - (Bx^* + f)\|_v \\ &= \|B(x^{(k)} - x^*)\|_v \\ &\leq \|B\|_v \|x^{(k)} - x^*\|_v \\ &= q \|x^{(k)} - x^*\|_v \end{aligned}$$

因此

$$\|x^{(k+1)} - x^*\|_v \leq q \|x^{(k)} - x^*\|_v$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

538

迭代法的收敛性

$$\begin{aligned} \|x^{(k+1)} - x^{(k)}\|_v &= \|(x^{(k+1)} - x^*) - (x^{(k)} - x^*)\|_v \\ &\geq \|x^{(k)} - x^*\|_v - \|x^{(k+1)} - x^*\|_v \\ &\geq (1 - q) \|x^{(k)} - x^*\|_v \\ &\geq \|x^{(k)} - x^*\|_v - q \|x^{(k)} - x^*\|_v \end{aligned}$$

$$\|x^{(k)} - x^*\|_v \leq \frac{1}{1-q} \|x^{(k+1)} - x^{(k)}\|_v \leq \frac{q}{1-q} \|x^{(k)} - x^{(k-1)}\|_v$$

根据上式和 $\|x^{(k+1)} - x^{(k)}\|_v \leq q \|x^{(k)} - x^{(k-1)}\|_v$, 可以得到

$$\|x^{(k)} - x^*\|_v \leq \frac{q^k}{1-q} \|x^{(1)} - x^{(0)}\|_v$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

539

迭代法的收敛性

$$\|x^{(k)} - x^*\|_v \leq \frac{q}{1-q} \|x^{(k)} - x^{(k-1)}\|_v$$

我们在实际计算时,常根据 $\|x^{(k)} - x^{(k-1)}\|_v$ 的值来判断是否终止计算.

上面的不等式表明这样终止计算有一定的根据.

在 q 的值接近于1时,可能会出现 $\|x^{(k)} - x^{(k-1)}\|_v$ 较小,而 $\|x^{(k)} - x^*\|_v$ 较大的情况.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

540

迭代法的收敛性

定理4 如果方程组 $x=Bx+f$ 的迭代公式为 $x^{(k+1)}=Bx^{(k)}+f$ ($x^{(0)}$ 为任意初始向量),且迭代矩阵的某一种算子范数 $\|B\|_v=q<1$,则

1. 迭代法收敛(F 范数小于1,此结论也成立)

$$2. \|x^{(k)} - x^*\|_v \leq \frac{q}{1-q} \|x^{(k)} - x^{(k-1)}\|_v$$

$$3. \|x^{(k)} - x^*\|_v \leq \frac{q^k}{1-q} \|x^{(1)} - x^{(0)}\|_v$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

541

迭代法的收敛性

$$B_0 = \begin{bmatrix} 0 & \frac{3}{8} & -\frac{2}{8} \\ -\frac{4}{11} & 0 & \frac{1}{11} \\ -\frac{6}{12} & -\frac{3}{12} & 0 \end{bmatrix}$$

对于前面的算例中的Jacobi迭代矩阵 B_0 ,易见 $\|B_0\|_\infty = \max\{5/8, 5/11, 9/12\} = 9/12 < 1$,因此, Jacobi迭代收敛.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

542

迭代法的收敛性

存在这样的矩阵,其谱半径小于1,但各种范数大于1,因此不能根据范数大于1来判断迭代法发散.

$$B = \begin{bmatrix} 0.9 & 0 \\ 0.3 & 0.8 \end{bmatrix} \quad \lambda_1 = 0.9, \lambda_2 = 0.8$$

$$\|B\|_1 = 1.2, \|B\|_\infty = 1.1, \|B\|_2 = 1.021, \|B\|_F = \sqrt{1.54}.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

543

对角优势阵

定义4 设 $A=(a_{ij})_n \in R^{n \times n}$ (或 $C^{n \times n}$),

(1)如果矩阵 A 满足条件

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \quad (i = 1, 2, \dots, n)$$

即 A 的每一行的对角元素的绝对值都严格大于同行其它元素绝对值之和,则称 A 为严格对角优势矩阵.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

544

对角优势阵

(2)如果 $|a_{ii}| \geq \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \quad (i = 1, 2, \dots, n)$

且至少有一个不等式严格成立,称 A 为弱对角优势矩阵.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

545

对角占优定理

定理6 如果 $A=(a_{ij})_n \in R^{n \times n}$ (或 $C^{n \times n}$)为严格对角优势阵(或为不可约弱对角优势阵),则 A 为非奇异矩阵.

证明 若 $\det(A)=0$,则 $Ax=0$ 有非零解,记为

$$x=(x_1, x_2, \dots, x_n)^T. \text{ 又记 } |x_k| = \max_{1 \leq i \leq n} |x_i| \neq 0$$

$$Ax=0 \text{ 的第 } k \text{ 个方程为 } \sum_{j=1}^n a_{kj}x_j = 0,$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

546

对角占优定理

$$|a_{kk}x_k| = \left| \sum_{j=1, j \neq k}^n a_{kj}x_j \right| \leq \sum_{j=1, j \neq k}^n |a_{kj}| |x_j| \leq |x_k| \sum_{j=1, j \neq k}^n |a_{kj}|$$

$$\text{因此 } |a_{kk}| \leq \sum_{j=1, j \neq k}^n |a_{kj}|,$$

与假设矛盾,故 $\det(A) \neq 0$,即 A 非奇异.

迭代法收敛的充分条件

定理7 如果 $A \in R^{n \times n}$ 为严格对角优势矩阵(或为不可约弱对角优势矩阵),则对任意的 $x^{(0)}$,解方程组 $Ax=b$ 的Jacobi迭代法,Gauss-Seidel迭代法均收敛.

证明 我们证明Jacobi迭代法收敛.

由假设可知 $a_{ii} \neq 0 (i=1,2,\dots,n)$,解方程组 $Ax=b$ 的Jacobi迭代法的迭代矩阵为

$$B_0 = D^{-1}(L+U).$$

迭代法收敛的充分条件

$$\begin{aligned} \det(\lambda I - B_0) &= \det(\lambda I - D^{-1}(L+U)) \\ &= \det(D^{-1}) \det(\lambda D - (L+U)) \end{aligned}$$

设 $B = \lambda D - (L+U)$

下面说明 $|\lambda| \geq 1$ 时, $\det(B) \neq 0$,从而 $\det(\lambda I - B_0) \neq 0$,于是 $\rho(B_0) < 1$,Jacobi迭代法收敛.

$$B = \begin{bmatrix} \lambda a_{11} & a_{12} & a_{1n} \\ a_{21} & \lambda a_{22} & a_{2n} \\ \vdots & \vdots & \ddots \\ a_{n1} & a_{n2} & \cdots & \lambda a_{nn} \end{bmatrix}$$

迭代法收敛的充分条件

$$B = \lambda D - (L+U) = \begin{bmatrix} \lambda a_{11} & a_{12} & a_{1n} \\ a_{21} & \lambda a_{22} & a_{2n} \\ \vdots & \vdots & \ddots \\ a_{n1} & a_{n2} & \cdots & \lambda a_{nn} \end{bmatrix}$$

A 为严格对角优势阵, $|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}| (i=1,2,\dots,n)$

若 $|\lambda| \geq 1$,显然 $|\lambda a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}| (i=1,2,\dots,n)$

因此 B 为严格对角优势阵, $\det(B) \neq 0$.

§ 4 超松弛迭代(SOR)方法

$$x_i^{(k)} \xrightarrow{\quad} x_{i(G)}^{(k+1)} \quad (1-\omega)x_i^{(k)} + \omega x_{i(G)}^{(k+1)}$$

沿着从 $x_i^{(k)}$ 到 $x_{i(G)}^{(k+1)}$ 的方向再向前走,就得到超松弛迭代(SOR)方法.

假设已知第 k 步的迭代向量 $x^{(k)}$ 以及第 $k+1$ 步迭代向量 $x^{(k+1)}$ 的前 $i-1$ 个分量已知,Gauss-Seidel迭代法取

$$\tilde{x}_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right)$$

超松弛迭代方法

我们定义新的 $x_i^{(k+1)}$ 为 $x_i^{(k)}$ 与 $\tilde{x}_i^{(k+1)}$ 的加权平均.

$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \omega\tilde{x}_i^{(k+1)} = x_i^{(k)} + \omega(\tilde{x}_i^{(k+1)} - x_i^{(k)})$$

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

在 $\omega=1$ 时,上述方法就是Gauss-Seidel方法, $\omega < 1$ 时,称为低松弛法, $\omega > 1$ 时称为超松弛法(有时不管 ω 的范围,统称为超松弛方法).

超松弛迭代方法(算例)

对于方程组
$$\begin{bmatrix} -4 & 1 & 1 & 1 \\ 1 & -4 & 1 & 1 \\ 1 & 1 & -4 & 1 \\ 1 & 1 & 1 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

松弛方法迭代格式为

$$\begin{cases} x_1^{(k+1)} = x_1^{(k)} - \omega(1 + 4x_1^{(k)} - x_2^{(k)} - x_3^{(k)} - x_4^{(k)})/4 \\ x_2^{(k+1)} = x_2^{(k)} - \omega(1 - x_1^{(k+1)} + 4x_2^{(k)} - x_3^{(k)} - x_4^{(k)})/4 \\ x_3^{(k+1)} = x_3^{(k)} - \omega(1 - x_1^{(k+1)} - x_2^{(k+1)} + 4x_3^{(k)} - x_4^{(k)})/4 \\ x_4^{(k+1)} = x_4^{(k)} - \omega(1 - x_1^{(k+1)} - x_2^{(k+1)} - x_3^{(k+1)} + 4x_4^{(k)})/4 \end{cases}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

553

超松弛迭代方法(算例)

取 $x^{(0)}=0, \omega=1.3$, 终止准则为 $\|x^{(k)} - x^{(k-1)}\|_{\infty} < 10^{-5}$.

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$x_4^{(k)}$	$\ x^{(k)} - x^{(k-1)}\ _{\infty}$
0	0	0	0	0	
1	-0.32500000	-0.43062500	-0.57057813	-0.75601602	0.756
2	-0.79858622	-0.88649937	-0.94718783	-0.95368731	0.474
10	-1.00000717	-0.99999179	-1.00000289	-1.00000170	3.45e-5
11	-0.99999667	-1.00000287	-0.99999954	-0.99999919	1.11e-5
12	-1.00000152	-0.99999922	-1.00000012	-1.00000052	4.85e-6

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

554

超松弛迭代方法(算例)

我们来观察松弛因子 ω 对收敛速度的影响.

ω	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
步数	301	156	104	76	59	47	38	31	26	21
ω	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
步数	17	12	12	15	18	24	35	55	114	**

步数表示 $\|x^{(k)} - x^{(k-1)}\|_{\infty} < 10^{-5}$ 时的迭代步数, $\omega=2.0$ 时, 500步以内不收敛.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

555

超松弛迭代方法(矩阵表示)

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

超松弛迭代格式可以写为

$$a_{ii} x_i^{(k+1)} = (1 - \omega) a_{ii} x_i^{(k)} + \omega \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right)$$

用矩阵可以表示为

$$\begin{aligned} Dx^{(k+1)} &= (1 - \omega) Dx^{(k)} + \omega(b + Lx^{(k+1)} + Ux^{(k)}) \\ (D - \omega L)x^{(k+1)} &= ((1 - \omega)D + \omega U)x^{(k)} + \omega b \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

556

超松弛迭代方法(矩阵表示)

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

在 $a_{ii} \neq 0 (i=1, 2, \dots, n)$ 时 $D - \omega L$ 非奇异,

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

超松弛方法迭代格式为 $x^{(k+1)} = L_{\omega} x^{(k)} + f$

其中 $L_{\omega} = (D - \omega L)^{-1} ((1 - \omega)D + \omega U)$

$$f = \omega(D - \omega L)^{-1} b$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

557

超松弛迭代方法(收敛的必要条件)

定理9 设方程组系数矩阵的对角元非零, 求解方程组的超松弛迭代方法收敛, 则 $0 < \omega < 2$.

证明 由于迭代法收敛, 显然 $\rho(L_{\omega}) < 1$.

设 L_{ω} 的特征值为 $\lambda_1, \lambda_2, \dots, \lambda_n$, 则

$$|\det(L_{\omega})| = |\lambda_1 \lambda_2 \cdots \lambda_n| \leq (\rho(L_{\omega}))^n, |\det(L_{\omega})|^{1/n} \leq \rho(L_{\omega}) < 1.$$

而 $\det(L_{\omega}) = \det((D - \omega L)^{-1} \det((1 - \omega)D + \omega U))$

$$= \det(D^{-1}) \det((1 - \omega)D + \omega U)$$

$$= \det((1 - \omega)I + \omega D^{-1}U) = (1 - \omega)^n$$

因此 $|1 - \omega| < 1, 0 < \omega < 2$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

558

超松弛迭代方法(收敛的充分条件)

对于一般的系数矩阵,我们较难给出超松弛迭代方法的充分条件.在 A 为对称正定矩阵时,有下面的结果.

定理10 如果 A 为对称正定矩阵,且 $0 < \omega < 2$, 则求解以 A 为系数矩阵的超松弛迭代法收敛.
证明略.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

559

第九章 矩阵的特征值 与特征向量计算

作业

- (1) Page 252 1
(2) 用Householder变换的方法对矩阵
进行QR分解(用精确值计算).

答案:

$$Q = \begin{bmatrix} -1/3 & 2/3 & 2/3 \\ 2/3 & 2/3 & -1/3 \\ 2/3 & -1/3 & 2/3 \end{bmatrix} \quad R = \begin{bmatrix} 3 & 7 & -5/3 \\ 0 & 5 & 1/3 \\ 0 & 0 & 10/3 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

561

§ 1 引言

定理1 如果 $\lambda_i (i=1,2,\dots,n)$ 是矩阵 A 的特征值,则有

$$1 \quad \sum_{i=1}^n \lambda_i = \sum_{i=1}^n a_{ii} = \text{tr}(A);$$

$$2 \quad \det(A) = \lambda_1 \lambda_2 \cdots \lambda_n$$

定理2 设 A 与 B 为相似矩阵(存在可逆矩阵 T ,使得 $B=T^{-1}AT$),则

1 A 与 B 有相同的特征值

2 若 x 是 B 的一个特征向量,则 Tx 是 A 的特征向量.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

562

Gerschgorin's 定理

定理3 设 $A=(a_{ij})_{n \times n}$, 则 A 的每一特征值必属于下述某个圆盘之中:

$$|\lambda - a_{ii}| \leq \sum_{j=1, j \neq i}^n |a_{ij}| \quad (i=1,2,\dots,n)$$

注: 我们只能断定任何一个特征值在 n 个圆盘的并集之中,不能说 n 个特征值分别在 n 个圆盘之中.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

563

Gerschgorin's 定理

证明 设 λ 为 A 的任意一个特征值, x 为对应的特征向量,则有

$$(\lambda I - A)x = 0.$$

记 $x=(x_1, x_2, \dots, x_n)^T \neq 0$ 及 $|x_i| = \max\{|x_k| \mid 1 \leq k \leq n\}$, 方程组的第 i 个方程为

$$(\lambda - a_{ii})x_i = \sum_{j=1, j \neq i}^n a_{ij}x_j$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

564

Gerschgorin's 定理

$$(\lambda - a_{ii})x_i = \sum_{j=1, j \neq i}^n a_{ij}x_j$$

因此有

$$|\lambda - a_{ii}| = \left| \sum_{j=1, j \neq i}^n a_{ij} \frac{x_j}{x_i} \right| \leq \sum_{j=1, j \neq i}^n |a_{ij}| \left| \frac{x_j}{x_i} \right| \leq \sum_{j=1, j \neq i}^n |a_{ij}|.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

565

Gerschgorin's 定理

注:并不是 n 个特征值分属于 n 个圆盘,定理只是说明特征值一定在特征向量中绝对值最大的分量对应的圆盘中.

例如:矩阵 $\begin{bmatrix} 79 & 133 & 208 \\ -46 & -84 & -164 \\ 2 & 7 & 33 \end{bmatrix}$

的三个特征值为-7,14,21.

矩阵第三行对应的圆盘为 $|\lambda - 33| \leq 9$.
该圆盘中无特征值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

566

Rayleigh商

定义1 设 A 为 n 阶实对称矩阵,对于任一非零向量 x ,称 $R(x) = x^T A x / x^T x$ 为对应于向量 x 的Rayleigh商.

定理4 设 A 为 n 阶实对称矩阵,其特征值次序记为 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$,对应的特征向量 x_1, x_2, \dots, x_n 组成规范化正交组,即 $(x_i, x_j) = \delta_{ij}$,则

$$1 \quad \lambda_n \leq \frac{x^T A x}{x^T x} \leq \lambda_1 (\forall x \in R^n, x \neq 0)$$

$$2 \quad \lambda_1 = \max_{\substack{x \in R^n \\ x \neq 0}} \frac{x^T A x}{x^T x} \quad \lambda_n = \min_{\substack{x \in R^n \\ x \neq 0}} \frac{x^T A x}{x^T x}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

567

§ 2 幂法及反幂法

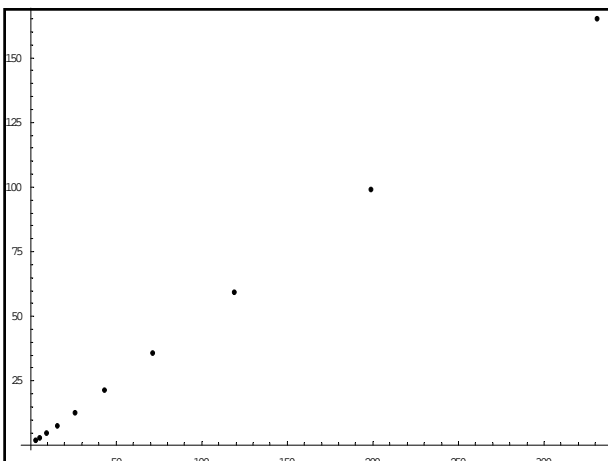
对于一个 n 阶矩阵 A ,一个 n 维向量 u , Au 是一个 n 维向量,如果再用 A 连续地进行左乘,得到一个向量序列,该序列有什么性质?

$$\text{设 } A = \begin{bmatrix} 4 & 2 \\ 3 & 3 \\ 1 & 1 \\ 3 & 1 \end{bmatrix}, u = \begin{bmatrix} 1 \\ 2 \end{bmatrix},$$

$A^k u (k=1, 2, \dots, 10)$ 这10个向量所代表的点可以用下面的图形来表示.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

568



幂法

可以看出,上面这10个点近似的在一条直线上.

下表给出了10个向量的单位化向量以及它们的长度(无穷范数).

k	$A^k u / \max(A^k u)$	$\max(A^k u)$	k	$A^k u / \max(A^k u)$	$\max(A^k u)$
1	(1., 0.875000)	2.67	6	(1., 0.503078)	42.78
2	(1., 0.630435)	5.11	7	(1., 0.501230)	71.39
3	(1., 0.549587)	8.96	8	(1., 0.500492)	119.04
4	(1., 0.519449)	15.23	9	(1., 0.500197)	198.43
5	(1., 0.507720)	25.59	10	(1., 0.500079)	330.75

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

570

幂法

可以看出,向量的方向趋于一致.

向量的长度逐渐扩大,扩大的倍数是下面几个数:

1.91667, 1.75362, 1.69972, 1.67963, 1.67181,

1.66872, 1.66749, 1.66699, 1.6668

长度扩大的倍数也趋于一致.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

571

幂法

设 $\frac{A^k u}{\max(A^k u)} \rightarrow x, \frac{\max(A^k u)}{\max(A^{k-1} u)} \rightarrow \lambda,$

则 $A^{k-1} u \approx \max(A^{k-1} u)x$

$$A^k u \approx \max(A^k u)x \approx \lambda \max(A^{k-1} u)x$$

$$\text{又 } A^k u = A(A^{k-1} u) \approx A(\max(A^{k-1} u)x) = \max(A^{k-1} u)(Ax)$$

因此 $Ax \approx \lambda x$

x 是 A 的近似特征向量, λ 是 A 的近似特征值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

572

幂法

问题:计算矩阵的主特征值及对应的特征向量

条件: A 有特征根 $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n| \geq 0$, 对

应 n 个线性无关的特征向量 x_1, \dots, x_n .

思路:从任意 $v_0 \neq 0$ 出发

$$v_0 = \sum_{i=1}^n a_i x_i, \quad a_1 \neq 0$$

$$v_1 = Av_0 = \sum_{i=1}^n a_i \lambda_i x_i$$

$$v_2 = Av_1 = \sum_{i=1}^n a_i \lambda_i^2 x_i$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

573

幂法

$$v_k = Av_{k-1} = \sum_{i=1}^n a_i \lambda_i^k x_i = \lambda_1^k \sum_{i=1}^n a_i \left(\frac{\lambda_i}{\lambda_1} \right)^k x_i$$

当 k 充分大时, 有 $v_k \approx \lambda_1^k a_1 x_1$, $v_{k-1} \approx \lambda_1^{k-1} a_1 x_1$

$$v_{k+1} = Av_k \approx \lambda_1^{k+1} a_1 x_1 \approx \lambda_1 v_k$$

求得近似特征向量

$$\frac{(\bar{v}^{(k)})_i}{(\bar{v}^{(k-1)})_i} \approx \lambda_1$$

求得近似特征值

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

574

幂法

定理5 设 $A \in R^{n \times n}$ 有 n 个线性无关的特征向量

x_1, x_2, \dots, x_n , 主特征值 λ_1 满足 $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$,

对于初始向量 $v_0 = a_1 x_1 + a_2 x_2 + \dots + a_n x_n$ ($a_1 \neq 0$),

$v_k = A^k v_0$, 则有 $\lim_{k \rightarrow \infty} \frac{v_k}{\lambda_1^k} = a_1 x_1$,

$$\lim_{k \rightarrow \infty} \frac{(v_{k+1})_i}{(v_k)_i} = \lambda_1.$$

注 若绝对值最大的特征值只有一个, 且为实数, 则当其为重根时, 上面的结论依然成立.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

575

幂法

在上面的方法中, 向量长度逐渐扩大(扩大的倍

数是 $|\lambda_1|$), 有可能导致计算溢出(趋于无穷或零).

为避免溢出, 计算时, 可以每次将向量规范化. 任

取向量 $u_0 = v_0 \neq 0$ ($a_1 \neq 0$), 构造向量序列

$$v_1 = Au_0 = Av_0, \quad u_1 = \frac{v_1}{\max(v_1)} = \frac{Av_0}{\max(Av_0)},$$

$$v_2 = Au_1 = \frac{A^2 v_0}{\max(Av_0)}, \quad u_2 = \frac{v_2}{\max(v_2)} = \frac{A^2 v_0}{\max(A^2 v_0)},$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

576

幂法

根据
$$v_k = \frac{A^k v_0}{\max(A^{k-1} v_0)}, u_k = \frac{A^k v_0}{\max(A^k v_0)},$$

可得
$$u_k = \frac{A^k v_0}{\max(A^k v_0)} = \frac{\lambda_1^k [a_1 x_1 + \sum_{i=2}^n a_i (\frac{\lambda_i}{\lambda_1})^k x_i]}{\lambda_1^k [a_1 x_1 + \sum_{i=2}^n a_i (\frac{\lambda_i}{\lambda_1})^k x_i]}$$

$$= \frac{a_1 x_1 + \sum_{i=2}^n a_i (\frac{\lambda_i}{\lambda_1})^k x_i}{\max(a_1 x_1 + \sum_{i=2}^n a_i (\frac{\lambda_i}{\lambda_1})^k x_i)} \rightarrow \frac{x_1}{\max(x_1)} (k \rightarrow \infty)$$

规范化向量序列收敛到主特征值对应的特征向量.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

577

幂法

类似可以得到

$$v_k = \frac{\lambda_1^k [a_1 x_1 + \sum_{i=2}^n a_i (\frac{\lambda_i}{\lambda_1})^k x_i]}{\max[\lambda_1^{k-1} (a_1 x_1 + \sum_{i=2}^n a_i (\frac{\lambda_i}{\lambda_1})^{k-1} x_i)]}$$

$$\max(v_k) = \frac{\lambda_1 \max[a_1 x_1 + \sum_{i=2}^n a_i (\frac{\lambda_i}{\lambda_1})^k x_i]}{\max[a_1 x_1 + \sum_{i=2}^n a_i (\frac{\lambda_i}{\lambda_1})^{k-1} x_i]} \rightarrow \lambda_1$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

578

幂法

定理6 设 $A \in R^{n \times n}$ 有 n 个线性无关的特征向量 x_1, x_2, \dots, x_n , 主特征值 λ_1 满足 $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$, 对于初始向量 $v_0 = u_0 = a_1 x_1 + a_2 x_2 + \dots + a_n x_n (a_1 \neq 0)$, 按如下方法构造的向量序列

有
$$v_0 = u_0 \neq 0, v_k = A u_{k-1}, u_k = \frac{v_k}{\max(v_k)},$$

$$\lim_{k \rightarrow \infty} u_k = \frac{x_1}{\max(x_1)}, \lim_{k \rightarrow \infty} \max(v_k) = \lambda_1.$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

579

幂法

注:上述定理是错误的!

错误的原因: \max 运算不具备连续性.

若 $u_k \rightarrow u$, 不能推出 $\max(u_k) \rightarrow \max(u)$.

若加上条件: 主特征值对应的特征向量只有一个分量的绝对值最大, 则定理成立.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

580

幂法(算例)

用幂法计算右边矩阵的主特征值和相应的特征向量.

$$A = \begin{bmatrix} 1.0 & 1.0 & 0.5 \\ 1.0 & 1.0 & 0.25 \\ 0.5 & 0.25 & 2.0 \end{bmatrix}$$

解: 取初始向量 $v_0 = u_0 = (1, 1, 1)^T$.

$v_1 = A u_0 = (2.5, 2.25, 2.75)^T, \max(v_1) = 2.75,$

$u_1 = v_1 / \max(v_1) = (0.9091, 0.8182, 1)^T.$

下面的计算过程见下表.

(终止准则: $|\max(v_k) - \max(v_{k-1})| < 5 \times 10^{-5}$)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

581

幂法(算例)

k	u_k^T	$\max(v_k)$	$ \max(v_k) - \max(v_{k-1}) $
0	(1, 1, 1)		
1	(0.9091, 0.8182, 1)	2.75000	0.1818
2	(0.8376, 0.7346, 1)	2.65909	7.46e-2
14	(0.7484, 0.6498, 1)	2.53670	9.82e-5
15	(0.7483, 0.6497, 1)	2.53662	7.13e-5
16	(0.7483, 0.6497, 1)	2.53658	4.16e-5

因此, 主特征值约为 2.5367, 主特征向量约为 (0.7483, 0.6497, 1)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

582

加速方法(原点平移法)

应用幂法计算矩阵 A 的主特征值的收敛速度主要由比值 $r=\lambda_1/\lambda_2$ 来确定.当 r 接近于1时,收敛速度较慢.

令 $B=A-pI$,

若 A 的特征值为 $\lambda_1, \lambda_2, \dots, \lambda_n$,则 B 的特征值为 $\lambda_1-p, \lambda_2-p, \dots, \lambda_n-p$,且对应特征向量相同.

选择 p 的目的: λ_1-p 仍然是 B 的主特征值,且

$$\left| \frac{\lambda_2-p}{\lambda_1-p} \right| < \left| \frac{\lambda_2}{\lambda_1} \right|$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

583

加速方法(原点平移法)

例2 若四阶矩阵 A 的特征值为11,12,13,14,则对应的比值 $r=\lambda_1/\lambda_2=0.9286$.

作变换 $B=A-12I$.

则 B 的特征值为2,1,0,-1.

应用幂法求 B 的主特征值的收敛速度对应的比值为0.5.

显然收敛速度得到加快.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

584

加速方法(原点平移法)

如果 A 的特征值满足 $\lambda_1 > \lambda_2 \geq \dots \geq \lambda_n$,

则 $B=A-pI$ 的主特征值为 λ_1-p 或 λ_n-p .

选取 p 的目的是使

$|\lambda_1-p| > |\lambda_n-p|$,且使下式取得最小

$$\omega = \max \left\{ \left| \frac{\lambda_2-p}{\lambda_1-p} \right|, \left| \frac{\lambda_n-p}{\lambda_1-p} \right| \right\}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

585

加速方法(原点平移法)

显然,当 $\lambda_2-p=-(\lambda_n-p)$,即

$p=(\lambda_2+\lambda_n)/2=p^*$ 时 ω 为最小,

这时对应的比值为

$$\frac{\lambda_2-p^*}{\lambda_1-p^*} = -\frac{\lambda_n-p^*}{\lambda_1-p^*} = \frac{\lambda_2-\lambda_n}{2\lambda_1-\lambda_2-\lambda_n}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

586

原点平移法算例

$$A = \begin{bmatrix} 1.0 & 1.0 & 0.5 \\ 1.0 & 1.0 & 0.25 \\ 0.5 & 0.25 & 2.0 \end{bmatrix}$$

对上面的矩阵作变换 $B=A-pI$,取 $p=0.75$,则

$$B = \begin{bmatrix} 0.25 & 1.0 & 0.5 \\ 1.0 & 0.25 & 0.25 \\ 0.5 & 0.25 & 1.25 \end{bmatrix}$$

对 B 应用幂法,得到下表的结果.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

587

原点平移法算例

k	u_k^T	$\max(v_k)$	$ \max(v_k) - \max(v_{k-1}) $
0	(1,1,1)		
1	(0.8750,0.7500,1)	2.00000	0.2500
2	(0.7833,0.7000,1)	1.87500	9.17e-2
9	(0.7483,0.6497,1)	1.78666	1.69e-4
10	(0.7482,0.6497,1)	1.78659	7.35e-5
11	(0.7482,0.6497,1)	1.78655	4.36e-5

因此, B 的主特征值约为1.78655, A 的主特征值约为1.7866+0.75=2.53655.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

588

Rayleigh商加速

根据定理4 $\lambda_1 = \max_{x \in R^n, x \neq 0} \frac{x^T A x}{x^T x}$ $\lambda_n = \min_{x \in R^n, x \neq 0} \frac{x^T A x}{x^T x}$

对称矩阵的绝对值最大,最小的特征值可以用Rayleigh商的极值来表示.

我们可以将Rayleigh商应用到幂法计算对称矩阵的主特征值的加速收敛上来.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

589

Rayleigh商加速

定理7 设 $A \in R^{n \times n}$ 为对称矩阵,特征值满足 $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$,应用幂法计算 A 的主特征值 λ_1 ,则规范化向量 u_k 的Rayleigh商给出 λ_1 的较好的近似

$$\frac{(Au_k, u_k)}{(u_k, u_k)} = \lambda_1 + O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2k}\right)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

590

Rayleigh商加速(算例)

对 $A = \begin{bmatrix} 1.0 & 1.0 & 0.5 \\ 1.0 & 1.0 & 0.25 \\ 0.5 & 0.25 & 2.0 \end{bmatrix}$

我们采用幂法及Rayleigh商加速来计算 A 的主特征值.取初始向量 $v_0 = u_0 = (1, 1, 1)^T$.

$$v_1 = Au_0 = (2.5, 2.25, 2.75)^T, \max(v_1) = 2.75,$$

$$u_1 = v_1 / \max(v_1) = (0.9091, 0.8182, 1)^T.$$

$$\frac{(Au_1, u_1)}{(u_1, u_1)} = 2.52483$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

591

Rayleigh商加速(算例)

k		$\left \frac{(Au_k, u_k)}{(u_k, u_k)} - \frac{(Au_{k-1}, u_{k-1})}{(u_{k-1}, u_{k-1})} \right $
1	2.52483	
2	2.53252	7.68e-3
3	2.53516	2.64e-3
4	2.53606	9.02e-4
5	2.53637	3.08e-4
6	2.53647	1.05e-4
7	2.53651	3.57e-5

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

592

反幂法

反幂法用来计算矩阵的按模最小的特征值及对应的特征向量.

设 $A \in R^{n \times n}$ 为非奇异矩阵,特征值满足 $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$,相应的特征向量为 x_1, x_2, \dots, x_n ,则 A^{-1} 的特征值为 $1/|\lambda_n| \geq 1/|\lambda_{n-1}| \geq \dots \geq 1/|\lambda_1|$,对应的特征向量为 x_n, x_{n-1}, \dots, x_1 .

因此计算 A 的模最小的特征值转化为计算 A^{-1} 的主特征值.

对 A^{-1} 应用幂法进行迭代就称为反幂法.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

593

反幂法

定理8 设(1) $A \in R^{n \times n}$ 有 n 个线性无关的特征向量 x_1, x_2, \dots, x_n , (2) A 非奇异且特征值满足 $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| > 0$,对于初始向量 $v_0 = u_0 = a_1 x_1 + a_2 x_2 + \dots + a_n x_n$ ($a_n \neq 0$),按如下反幂法

$$v_k = A^{-1} u_{k-1}, u_k = \frac{v_k}{\max(v_k)} \quad (k = 1, 2, \dots)$$

构造的向量序列 $\{v_k\}, \{u_k\}$ 满足

$$\lim_{k \rightarrow \infty} u_k = \frac{x_n}{\max(x_n)}, \lim_{k \rightarrow \infty} \max(v_k) = \frac{1}{\lambda_n}.$$

注:本定理有类似于定理6的问题.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

594

反幂法

如果已知矩阵的某个特征值 λ_j 的近似值 p ,满足 $|\lambda_j - p| < |\lambda_i - p| (i \neq j)$,于是 $1/(\lambda_j - p)$ 是 $(A - pI)^{-1}$ 的主特征值.我们可以用反幂法来求出该特征值及对应的特征向量.

在计算过程中,有 $v_k = (A - pI)^{-1}u_{k-1}$.

我们通过求解方程组 $(A - pI)v_k = u_{k-1}$ 得到 v_k .

由于要反复求解同一个系数矩阵对应的方程组,我们先对 $(A - pI)$ 进行三角分解,然后求解.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

595

反幂法(算例)

例4 用反幂法求 A 的对应计算特征值 $\lambda=1.2679$ 的特征向量.

$$A = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 3 & 1 \\ 0 & 1 & 4 \end{bmatrix}$$

对应的精确特征值为 $\lambda = 3 - \sqrt{3}$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

596

反幂法(算例)

解:先对 $(A - pI)$ 用列选主元方法进行三角分解,得到 $P(A - pI) = LU$,其中

$$A - pI = \begin{bmatrix} 0.7321 & 1 & 0 \\ 1 & 1.7321 & 1 \\ 0 & 1 & 2.7321 \end{bmatrix} \quad P = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0.7321 & -0.26807 & 1 \end{bmatrix} \quad U = \begin{bmatrix} 1 & 1.7321 & 1 \\ 0 & 1 & 2.7321 \\ 0 & 0 & 0.29517 \times 10^{-3} \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

597

反幂法(算例)

取 $Uv_1 = (1, 1, 1)^T$,得

$$v_1 = (12644, -9255.1, 3387.9)^T,$$

$$u_1 = (1, -0.73198, 0.26795)^T,$$

由 $LUv_2 = Pu_1$,得

$$v_2 = (20328, -14881, 5446.8)^T,$$

$$u_2 = (1, -0.73205, 0.26795)^T.$$

而 A 的对应的特征向量的精确值取前5位有效数字与 u_2 是一致的.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

598

§ 3 Jacobi方法

本节考虑的问题是计算实对称矩阵的全部特征值与对应的特征向量.

若 A 为实对称矩阵,则存在正交阵 P ,使得

$$PAP^T = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_n] \triangleq D.$$

D 的对角元就是 A 的特征值, P^T 的列向量就是 A 的特征向量.

正交阵 P 是指 $PP^T = P^T P = I$ 或 P 的列向量为规范向量组.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

599

Jacobi方法(二阶)

对于二阶对称阵 $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$

我们要寻找二阶正交阵 P ,将 A 对角化.

P 可以写为如下形式 $P = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix}$

此时可以求得 $PAP^T = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$

其中 $c_{11} = a_{11} \cos^2 \theta + a_{22} \sin^2 \theta + a_{21} \sin 2\theta$

$$c_{22} = a_{11} \sin^2 \theta + a_{22} \cos^2 \theta - a_{21} \sin 2\theta$$

$$c_{12} = c_{21} = \frac{1}{2}(a_{22} - a_{11}) \sin 2\theta + a_{21} \cos 2\theta$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

600

Jacobi方法(二阶)

$$PAP^T = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

$c_{12} = c_{21} = \frac{1}{2}(a_{22} - a_{11})\sin 2\theta + a_{21}\cos 2\theta$
为使得 c_{12} 与 c_{21} 为零,可以选择 θ 使得

$$\frac{1}{2}(a_{22} - a_{11})\sin 2\theta + a_{21}\cos 2\theta = 0$$

因此 $\tan 2\theta = \frac{2a_{12}}{a_{11} - a_{22}}$

当 $a_{11}=a_{22}$ 时选取 $\theta=\pi/4$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

601

Jacobi方法

在 R^n 空间中,我们针对其中的两个分量实施旋转变换 $y=Px$,其中

$$P = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & & \cos \theta & \sin \theta \\ & & & -\sin \theta & \cos \theta \\ & & & & 1 & \ddots \\ & & & & & 1 & \ddots \\ & & & & & & 1 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

602

Jacobi方法

具体的坐标变换格式为 $\begin{cases} y_i = x_i \cos \theta + x_j \sin \theta \\ y_j = -x_i \sin \theta + x_j \cos \theta \\ y_k = x_k, \quad k \neq i, j \end{cases}$
变换矩阵 P 具有如下性质

1. P 为正交阵
2. P 与单位阵只在 (i,i) , (j,i) , (i,j) , (j,j) 4个位置上的元素不一样
3. PA 只改变 A 的第 i 行与第 j 行元素, AP^T 只改变 A 的第 i 列与第 j 列元素, PAP^T 只改变 A 的第 i 行,第 j 行,第 i 列,第 j 列元素.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

603

Jacobi方法

定理10 设 A 为 n 阶对称矩阵, $C=PAP^T$,其中 P 为正交矩阵,则 $\|C\|_F^2 = \|A\|_F^2$.

证明: $\|A\|_F^2 = \sum_{i=1}^n \sum_{j=1}^n a_{ij}^2 = \text{tr}(A^T A)$

$$(A^T A)_{jj} = \sum_{i=1}^n (A^T)_{ji} (A)_{ij} = \sum_{i=1}^n (A)_{ij} (A)_{ij} = \sum_{i=1}^n a_{ij}^2$$

$$\text{tr}(A^T A) = \text{tr}(A^2) = \sum_{i=1}^n \lambda_i^2(A) \quad \text{同理} \quad \|C\|_F^2 = \sum_{i=1}^n \lambda_i^2(C)$$

A 与 C 的特征值一致,因此 $\|C\|_F^2 = \|A\|_F^2$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

604

Jacobi方法

对 $A \in R^{n \times n}$, $P(i,j)$ 为一平面旋转矩阵,则 $C=PAP^T$ 的计算公式为

$$c_{ii} = a_{ii} \cos^2 \theta + a_{jj} \sin^2 \theta + a_{ij} \sin 2\theta$$

$$c_{jj} = a_{ii} \sin^2 \theta + a_{jj} \cos^2 \theta - a_{ij} \sin 2\theta$$

$$c_{ij} = c_{ji} = \frac{1}{2}(a_{jj} - a_{ii})\sin 2\theta + a_{ij}\cos 2\theta$$

$$c_{ik} = c_{ki} = a_{ik} \cos \theta + a_{jk} \sin \theta (k \neq i, j)$$

$$c_{jk} = c_{kj} = a_{jk} \cos \theta - a_{ik} \sin \theta (k \neq i, j)$$

$$c_{lk} = a_{lk} (l, k \neq i, j)$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

605

Jacobi方法

若 A 的非对角元 $a_{ij} \neq 0$,可以选择平面旋转矩阵 $P(i,j)$,使得

$$c_{ij} = c_{ji} = \frac{1}{2}(a_{jj} - a_{ii})\sin 2\theta + a_{ij}\cos 2\theta = 0$$

即选择 θ 使得 $\tan 2\theta = \frac{2a_{ij}}{a_{ii} - a_{jj}} (|\theta| \leq \frac{\pi}{4})$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

606

Jacobi方法

设 $A \in R^{n \times n}$ 为对称矩阵, $a_{ij} \neq 0$ 为 A 的一个非对角元素, 则可选择一平面旋转阵 $P(i, j)$, 使 $C = PAP^T$ 的非对角元素 $c_{ij} = c_{ji} = 0$ 且 C 与 A 的元素满足下列关系

$$1. c_{ik}^2 + c_{jk}^2 = a_{ik}^2 + a_{jk}^2 \quad (k \neq i, j)$$

$$2. c_{ii}^2 + c_{jj}^2 = a_{ii}^2 + a_{jj}^2 + 2a_{ij}^2$$

$$3. c_{ik}^2 = a_{ik}^2 \quad (i, k \neq j).$$

C 的对角元的平方和比 A 的对角元的平方和增加了 $2a_{ij}^2$, 而非对角元的平方和减少了 $2a_{ij}^2$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

607

Jacobi方法

Jacobi方法首先在 A 的非对角线元素中选取绝对值最大的元素, 然后选择一个平面旋转矩阵 P_1 , 使得 $A_1 = P_1 A P_1^T$ 在该处的元素变为 0, 然后再对 A_1 实施类似的变换. 最终使得矩阵的非对角线元素的绝对值充分的小, 从而求得 A 的全部近似特征值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

608

Jacobi方法

对 A 实施了 m 次平面旋转变换后, 有

$$P_m \cdots P_2 P_1 A P_1^T P_2^T \cdots P_m^T \approx D$$

其中 D 是一个对角矩阵.

令 $R_m = P_m \cdots P_2 P_1$, 则有

$$R_m A R_m^T \approx D.$$

于是, R_m^T 的列向量就是 A 的近似特征向量.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

609

Jacobi方法的收敛性

定理 13 设 $A = (a_{ij})_n$ 为实对称矩阵, 对 A 实施上述一系列平面旋转变换 $A_m = P_m A_{m-1} P_m^T$ ($m=1, 2, \dots$), 则 $\lim_{m \rightarrow \infty} A_m = D$ (对角矩阵)

证明 记 $A_m = (a_{ik}^{(m)})_n$, $S_m = \sum_{i \neq k} (a_{ik}^{(m)})^2$

$$\text{则 } S_{m+1} = S_m - 2(\max_{i \neq k} |a_{ik}^{(m)}|)^2$$

$$\text{又 } S_m = \sum_{i \neq k} (a_{ik}^{(m)})^2 \leq n(n-1)(\max_{i \neq k} |a_{ik}^{(m)}|)^2$$

$$\text{得 } S_{m+1} \leq S_m - \frac{2S_m}{n(n-1)}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

610

Jacobi方法的收敛性

$$S_{m+1} \leq S_m - \frac{2S_m}{n(n-1)} = S_m \left(1 - \frac{2}{n(n-1)}\right)$$

$$\text{因此 } S_m \leq S_0 \left(1 - \frac{2}{n(n-1)}\right)^m$$

$$\lim_{m \rightarrow \infty} S_m = 0$$

上式说明 A 的非对角元趋于零.

关于对角元的收敛性证明略去.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

611

Jacobi方法计算步骤

(1) 先选取 A 的绝对值最大非对角线元素, 设为 a_{ij} .

(2) 计算平面旋转矩阵对应的角度的三角函数

$$d = \frac{a_{ii} - a_{jj}}{2a_{ij}} \quad \tan \theta = \frac{s(d)}{|d| + \sqrt{d^2 + 1}} \triangleq t$$

$$\cos \theta = \frac{1}{\sqrt{1+t^2}} \triangleq c \quad \sin \theta = \cos \theta * t = c * t \triangleq s$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

612

Jacobi方法计算步骤

(3)计算 A 矩阵变换后的第 i,j 行和第 i,j 列元素

$$\begin{aligned}c_{ii} &= a_{ii} + ta_{ij} \\c_{jj} &= a_{jj} - ta_{ij} \\c_{ij} &= c_{ji} = 0 \\c_{ik} &= c_{ki} = ca_{ik} + sa_{jk} (k \neq i, j) \\c_{jk} &= c_{kj} = ca_{kj} - sa_{ki} (k \neq i, j)\end{aligned}$$

(4)计算 R 矩阵变换后的第 i,j 列元素

$$\begin{aligned}\tilde{r}_{ki} &= cr_{ki} + sr_{kj} (k = 1, 2, \dots, n) \\ \tilde{r}_{kj} &= -sr_{ki} + cr_{kj} (k = 1, 2, \dots, n)\end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

613

Jacobi方法(算例)

例5 用Jacobi方法计算对称矩阵 $A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$ 的特征值和特征向量.

解:第1步 $A_0 = A$, 选取 A_0 的绝对值最大的元素 $a_{12} = -1, i=1, j=2$.

$d=0, t=1, c=s=0.7071068$,

$$A_1 = P_1 A P_1^T = \begin{bmatrix} 1 & 0 & -0.7071068 \\ 0 & 3 & -0.7071068 \\ -0.7071068 & -0.7071068 & 2 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

614

Jacobi方法(算例)

$$R = \begin{bmatrix} 0.7071068 & -0.7071068 & 0 \\ 0.7071068 & 0.7071068 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

设终止准则为非对角线平方和小于 10^{-10} , 可以得到下面的计算结果.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

615

$$A_2 = \begin{bmatrix} 0.6339746 & -0.3250576 & 0 \\ -0.3250576 & 3 & -0.6279630 \\ 0 & -0.6279630 & 2.366025 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0.6339746 & -0.2768366 & -0.1703642 \\ -0.2768366 & 3.386446 & 0 \\ -0.1703642 & 0 & 1.979579 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 0.6064072 & 0 & -0.1695258 \\ 0 & 3.414013 & 0.01688139 \\ -0.1695258 & 0.01688139 & 1.979579 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} 0.5857879 & 0.002038248 & 0 \\ 0.002038248 & 3.414013 & 0.01675789 \\ 0 & 0.01675789 & 2.000199 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

616

$$A_6 = \begin{bmatrix} 0.5857879 & 0.002038105 & -0.00002415418 \\ 0.002038105 & 3.414212 & 0 \\ -0.00002415418 & 0 & 2.000000 \end{bmatrix}$$

$$A_7 = \begin{bmatrix} 0.5857864 & 0 & -0.00002415418 \\ 0 & 3.414214 & 1.740500 \times 10^{-8} \\ -0.00002415418 & 1.740500 \times 10^{-8} & 2.000000 \end{bmatrix}$$

$$A_8 = \begin{bmatrix} 0.5857864 & 0 & 0 \\ 0 & 3.414214 & 1.740500 \times 10^{-8} \\ 0 & 1.740500 \times 10^{-8} & 2.000000 \end{bmatrix}$$

A_8 的非对角元平方和为 6.06×10^{-16} .

特征值精确值: 3.41421356, 2, 0.585786438,

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

617

Jacobi方法(算例)

最终得到的矩阵 R 为

$$R = \begin{bmatrix} 0.500000000 & -0.500000009 & -0.707106775 \\ 0.707106781 & 0.707106781 & -8.7 \times 10^{-9} \\ 0.500000000 & -0.499999991 & 0.707106787 \end{bmatrix}$$

精确值为

$$\begin{bmatrix} 0.5 & -0.5 & -0.707106781 \\ 0.707106781 & 0.707106781 & 0 \\ 0.5 & -0.5 & 0.707106781 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

618

Jacobi过关法

在Jacobi方法每一步迭代过程中,矩阵的计算为 $O(n)$.

但是每一步迭代时需要进行 $n(n-1)/2$ 个比较,会浪费太多的时间.

实际计算中,我们将Jacobi方法改为Jacobi过关法.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

619

Jacobi过关法

$$\text{设 } v_0 = \left(2 \sum_{j=1}^n \sum_{k=1}^{l-1} a_{jk}^2 \right)^{\frac{1}{2}} = (S(A))^{\frac{1}{2}}$$

设置关口 $v_1=v_0/n$,在 A 的(上三角部分)非对角线元素中按行扫描,一旦有非对角元 a_{ij} 的绝对值大于等于 v_1 ,则实施平面旋转变换将 a_{ij} 的化为零.重复这一过程,直到所有非对角元都“过关”.

再设第2道关口 $v_2=v_1/n$.

重复类似的步骤,直至满足精度要求.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

620

Jacobi过关法(算例)

对于前面的算例,改用Jacobi过关法,前两步得到的 A 矩阵是一致的

$$A_2 = \begin{bmatrix} 0.6339746 & -0.3250576 & 0 \\ -0.3250576 & 3 & -0.6279630 \\ 0 & -0.6279630 & 2.366025 \end{bmatrix}$$

在得到 A_2 后,由于 a_{12} 不能过关,我们直接对该元素进行变换,因此 A_3 矩阵与Jacobi方法中不再一致.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

621

Jacobi过关法(算例)

$$A_3 = \begin{bmatrix} 0.5901289 & 0 & -0.08394315 \\ -0.08394315 & 3.0438457 & -0.6223272 \\ -0.08394315 & -0.6223272 & 2.366025 \end{bmatrix}$$

实施8次变换最终得到的 A 矩阵为

$$A_8 = \begin{bmatrix} 0.5857864 & 0 & -4.330069 \times 10^{-8} \\ 0 & 3.414214 & 8.6 \times 10^{-13} \\ -4.330069 \times 10^{-8} & 8.6 \times 10^{-13} & 2.000000 \end{bmatrix}$$

对角元平方和为 3.75×10^{-15} ,与Jacobi方法计算结果基本一致.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

622

Householder方法

对于实对称矩阵 A ,虽然在理论上存在正交矩阵 P ,使得 PAP^T 为对角矩阵,但是矩阵 P 一般不易直接求得.

不过我们容易实现一个实对称矩阵与三对角矩阵的正交相似.

对于一般的矩阵,我们可以将其与一个上Hessenberg矩阵实现正交相似.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

623

上Hessenberg矩阵

如下形式的矩阵称为上Hessenberg矩阵.

$$\begin{bmatrix} * & * & * & \cdots & * & * & * \\ * & * & * & \cdots & * & * & * \\ 0 & * & * & \cdots & * & * & * \\ 0 & 0 & * & \cdots & * & * & * \\ 0 & 0 & 0 & \cdots & * & * & * \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \vdots & 0 & * & * \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

624

矩阵的上Hessenberg化

对于一般的矩阵,我们将其正交相似与一个上Hessenberg矩阵,然后再使用QR方法(下一节)可以求出其特征值与特征向量.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

625

初等反射阵

给定 R^n 空间中的一点(用向量 v 表示),给定一个平面 $S: w^T x = 0$ (w 为其单位法向量).

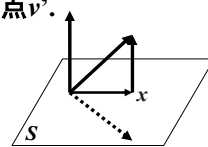
现在要求出 v 关于平面 S 的对称点 v' .

设 y 为 v 在 w 方向上的投影.

即 $v = x + \alpha w, x^T w = 0$. 则有

$$w^T v = w^T (x + \alpha w) = \alpha w^T w = \alpha$$

$$\text{因此 } \alpha = w^T v, \quad v' = v - 2y = v - 2ww^T v = (I - 2ww^T)v.$$



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

626

初等反射阵

$$v' = (I - 2ww^T)v.$$

称矩阵 $H = I - 2ww^T$ 为初等反射阵,记为 $H(w)$.

$$H(w) = \begin{bmatrix} 1 - 2w_1^2 & -2w_1w_2 & \cdots & -2w_1w_n \\ -2w_1w_2 & 1 - 2w_2^2 & \cdots & -2w_2w_n \\ \vdots & \vdots & \ddots & \vdots \\ -2w_nw_1 & -2w_nw_2 & \cdots & 1 - 2w_n^2 \end{bmatrix}$$

对于非零向量 $u, u/\|u\|$ 为单位向量,因此矩阵 $H = I - 2uu^T/\|u\|^2$ 为初等反射阵.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

627

初等反射阵的性质

定理15 初等反射阵 H 是对称阵($H^T = H$),正交阵($H^T H = I$),对合阵($H^2 = I$).

证明: $H = I - 2ww^T, H^T = (I - 2ww^T)^T = I - 2ww^T = H$.

$$\begin{aligned} H^T H &= H^2 = (I - 2ww^T)(I - 2ww^T) \\ &= I - 4ww^T + 4ww^T ww^T \\ &= I \end{aligned}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

628

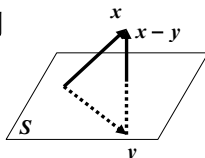
初等反射阵

问题:对于 R^n 空间中的任意两个长度相等的向量 $x \neq y$,能否找到初等反射阵 H ,使得 $Hx = y$?

显然,将向量 $x - y$ 单位化得到向量 w ,其对应的初等反射阵 H ,能够使得 $Hx = y$.

即

$$H = I - 2 \frac{(x - y)(x - y)^T}{\|x - y\|^2}$$



南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

629

初等反射阵

$$H = I - 2 \frac{(x - y)(x - y)^T}{\|x - y\|^2}$$

$$Hx = x - 2 \frac{(x - y)(x - y)^T x}{\|x - y\|^2}$$

$$\|x - y\|^2 = (x - y)^T (x - y)$$

$$= \|x\|^2 - 2x^T y + \|y\|^2$$

$$= x^T x - 2x^T y + y^T y$$

$$= 2x^T x - 2x^T y$$

$$= 2(x - y)^T x$$

因此

$$Hx = x - (x - y) = y$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

630

矩阵的上Hessenberg化

对于一般的矩阵 A ,我们希望借助于通过初等反射阵将 A 通过相似变换(称为Householder变换)进行简化,即使得 $A_0=H_1AH_1^T$ 尽可能的简化.

我们可以找到一个初等反射阵 H 先使得 A 的第一列的零元素尽可能的多.

$$\text{即 } H[a_1, a_2, \dots, a_n] = [\sigma e_1, b_2, \dots, b_n],$$

其中 σ 为 a_1 的长度.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

631

矩阵的上Hessenberg化

$$H[a_1, a_2, \dots, a_n] = [\sigma e_1, b_2, \dots, b_n]$$

我们目的是求矩阵的特征值,因此必须对矩阵实施相似变换.

因此用 H 左乘时,必须用 $H(=H^{-1})$ 也进行右乘.

得到的矩阵为

$$HAH^T = [\sigma e_1, b_2, \dots, b_n]H^T$$

此时矩阵的第一列一般不再稀疏.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

632

矩阵的上Hessenberg化

要使得矩阵右乘一个矩阵 H 保持第一列不变, H 的第一行必须为 $[1, 0, \dots, 0]$.

因此实施Householder变换的初等反射阵只能是下面的形式

$$H = \begin{bmatrix} 1 & 0 \\ 0 & H_0 \end{bmatrix}$$

其中 H_0 为 $n-1$ 阶的方阵.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

633

矩阵的上Hessenberg化

$$H = \begin{bmatrix} 1 & 0 \\ 0 & H_0 \end{bmatrix} \quad A = \begin{bmatrix} a_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

将矩阵 A 进行类似

的分块.可以得到 $HAH = \begin{bmatrix} a_{11} & A_{12}H_0 \\ H_0A_{21} & H_0A_{22}H_0 \end{bmatrix}$

因此,我们只能选取 $n-1$ 阶的

初等反射阵 H_0 ,使得 H_0A_{21} 成

为 σe_1 ($n-1$ 维),通过对应的矩

阵 H 对 A 实施Householder变

换后得到如下形式的矩阵.

$$HAH = \begin{bmatrix} * & * & * & \dots & * \\ * & * & * & \dots & * \\ 0 & * & * & \dots & * \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & * & * & \dots & * \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

634

矩阵的上Hessenberg化

$$HAH = \begin{bmatrix} * & * & * & \dots & * \\ * & * & * & \dots & * \\ 0 & * & * & \dots & * \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & * & * & \dots & * \end{bmatrix}$$

对于右下角的 $n-1$ 阶矩阵,我们类似的可以找到一个Householder变换将其第一列的后面 $n-3$ 个元素变为0.

这样两步变换后矩阵的形式成为

$$\begin{bmatrix} * & * & * & * & \dots & * \\ * & * & * & * & \dots & * \\ 0 & * & * & * & \dots & * \\ 0 & 0 & * & * & \dots & * \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & * & * & \dots & * \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

635

矩阵的上Hessenberg化

最终我们可将矩阵变换为上Hessenberg矩阵.

$$\begin{bmatrix} * & * & * & \dots & * & * & * \\ * & * & * & \dots & * & * & * \\ 0 & * & * & \dots & * & * & * \\ 0 & 0 & * & \dots & * & * & * \\ 0 & 0 & 0 & \dots & * & * & * \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \vdots & 0 & * & * \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

636

矩阵的上Hessenberg化

$$A = \begin{bmatrix} a_{11} & A_{12}^{(1)} \\ A_{21}^{(1)} & A_{22}^{(1)} \end{bmatrix}$$

设 $A_{21}^{(1)} = [a_{21}, \dots, a_{n1}] \neq 0, \sigma_1 = \pm \|A_{21}^{(1)}\|_2$

选取初等反射阵 $H(n-1)$ 阶, 使得 $Hx = -\sigma_1 e_1 (n-1 \text{ 维})$,

显然有 $H = I - 2ww^T, w = \frac{A_{21}^{(1)} + \sigma_1 e_1}{\|A_{21}^{(1)} + \sigma_1 e_1\|_2}$

记 $u_1 = A_{21}^{(1)} + \sigma_1 e_1 = (a_{21} + \sigma_1, a_{31}, \dots, a_{n1})$

若 a_{21} 与 σ_1 异号, 会造成有效数字的损失, 因此取

$$\sigma_1 = \text{sign}(a_{21}) \|A_{21}^{(1)}\|_2$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

637

矩阵的上Hessenberg化

$$A = \begin{bmatrix} a_{11} & A_{12}^{(1)} \\ A_{21}^{(1)} & A_{22}^{(1)} \end{bmatrix}$$

第1步

$$\sigma_1 = \text{sign}(a_{21}) \|A_{21}^{(1)}\|_2 \quad u_1 = A_{21}^{(1)} + \sigma_1 e_1$$

$$\rho_1 = \frac{1}{2} \|u_1\|_2^2 \quad R_1 = I - \frac{1}{\rho} u_1 u_1^T \quad U_1 = \begin{bmatrix} 1 & 0 \\ 0 & R_1 \end{bmatrix}$$

$$A_2 = U_1 A U_1 = \begin{bmatrix} a_{11} & A_{12}^{(1)} R_1 \\ R_1 A_{21}^{(1)} & R_1 A_{22}^{(1)} R_1 \end{bmatrix} \triangleq \begin{bmatrix} A_{11}^{(2)} & A_{12}^{(2)} & A_{13}^{(2)} \\ 0 & A_{22}^{(2)} & A_{23}^{(2)} \end{bmatrix} \begin{bmatrix} k \\ n-k \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

638

假设对 A 已经进行了 $k-1$ 步正交相似约化, 得

$$A_k = U_{k-1} A_{k-1} U_{k-1} = \begin{bmatrix} a_{11} & a_{12}^{(2)} & \cdots & a_{1k}^{(k)} & a_{1,k+1}^{(k)} & \cdots & a_{1n}^{(k)} \\ -\sigma_1 & a_{22}^{(2)} & \cdots & a_{2k}^{(k)} & a_{2,k+1}^{(k)} & \cdots & a_{2n}^{(k)} \\ & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ & & -\sigma_{k-1} & a_{kk}^{(k)} & a_{k,k+1}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & & a_{k+1,k}^{(k)} & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ & & & \vdots & \vdots & \ddots & \vdots \\ & & & a_{nk}^{(k)} & a_{n,k+1}^{(k)} & \cdots & a_{nn}^{(k)} \end{bmatrix} \triangleq \begin{bmatrix} A_{11}^{(k)} & A_{12}^{(k)} & A_{13}^{(k)} \\ 0 & A_{22}^{(k)} & A_{23}^{(k)} \end{bmatrix} \begin{bmatrix} k \\ n-k \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

639

$$A_k = \begin{bmatrix} A_{11}^{(k)} & A_{12}^{(k)} & A_{13}^{(k)} \\ 0 & A_{22}^{(k)} & A_{23}^{(k)} \end{bmatrix} \begin{bmatrix} k \\ n-k \end{bmatrix}$$

第 k 步

$$\sigma_k = \text{sign}(a_{k+1,k}^{(k)}) \|A_{22}^{(k)}\|_2 \quad u_k = A_{22}^{(k)} + \sigma_k e_1$$

$$\rho_k = \frac{1}{2} \|u_k\|_2^2 \quad R_k = I - \frac{1}{\rho} u_k u_k^T \quad U_k = \begin{bmatrix} I & 0 \\ 0 & R_k \end{bmatrix}$$

$$A_{k+1} = U_k A_k U_k = \begin{bmatrix} A_{11}^{(k)} & A_{12}^{(k)} & A_{13}^{(k)} R_k \\ 0 & R_k A_{22}^{(k)} & R_k A_{23}^{(k)} R_k \end{bmatrix}$$

将此过程进行 $n-2$ 步, 即得到初等反射阵 U_1, \dots, U_{n-2} , 使得 $U_{n-2} \cdots U_1 A U_1 \cdots U_{n-2} = C$ (上Hessenberg阵)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

640

对称矩阵的上Hessenberg化

若 A 为对称矩阵, 根据前面结论, 存在初等反射阵初等反射阵 U_1, \dots, U_{n-2} , 使得 $U_{n-2} \cdots U_1 A U_1 \cdots U_{n-2} = C$ (上Hessenberg阵).

显然此时 C 为对称矩阵.

对称的上Hessenberg阵显然为三对角矩阵.

我们得到定理18.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

641

对称矩阵的上Hessenberg化

定理18 如果 $A \in R^{n \times n}$ 为对称矩阵, 存在初等反射阵 U_1, \dots, U_{n-2} , 使得

$$U_{n-2} \cdots U_2 U_1 A U_1 U_2 \cdots U_{n-2} = \begin{bmatrix} c_1 & b_1 & & & \\ b_1 & c_2 & b_2 & & \\ & c_2 & b_2 & \ddots & \\ & & \ddots & \ddots & c_{n-1} & b_{n-1} \\ & & & & b_{n-1} & c_n \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

642

上Hessenberg化(算例)

例6 用Householder方法将下述矩阵化为上Hessenberg阵.

$$A_1 = A = \begin{bmatrix} -4 & -3 & -7 \\ 2 & 3 & 2 \\ 4 & 2 & 7 \end{bmatrix}$$

解:确定变换

$$U_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & R_1 \\ 0 & \end{bmatrix} \quad a_{21}^{(1)} = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$$

其中 R_1 为初等反射阵,且 $R_1 a_{21}^{(1)} = -\sigma_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

643

上Hessenberg化(算例)

$$\sigma_1 = \text{sign}(2) \|a_{21}^{(1)}\| = \sqrt{20} = 4.472136$$

$$u_1 = a_{21}^{(1)} + \sigma_1 e_1 = \begin{bmatrix} 2 + \sqrt{20} \\ 4 \end{bmatrix} = \begin{bmatrix} 6.472136 \\ 4 \end{bmatrix}$$

$$\rho_1 = \sigma_1(\sigma_1 + a_{21}) = \sqrt{20}(\sqrt{20} + 2) = 28.94427$$

于是可以得到 R_1 的数值. $R_1 = I_2 - \rho_1^{-1} u_1 u_1^T$

根据 R_1, U_1 就确定下来,可以得到 $A_2 = U_1 A U_1$.

$$A_2 = \begin{bmatrix} -4 & 7.602634 & -0.447212 \\ -4.472136 & 7.800003 & -0.399999 \\ 0 & -0.399999 & 2.200000 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

644

QR算法

若 A 为非奇异矩阵,则存在正交矩阵 Q 及上三角矩阵 R ,使得 $A=QR$.

如果限定 R 的对角元为正数,则这种分解是唯一的.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

645

Schmidt正交化方法

矩阵的QR分解有许多方法.

方法一 Schmidt正交化方法(参见矩阵论教材).

对线性无关的向量组 a_1, a_2, \dots, a_n ,通过下面的Schmidt正交化方法可以得到正交的向量组 b_1, b_2, \dots, b_n ,且这两组向量可以相互线性表示.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

646

Schmidt正交化方法

$$b_1 = a_1$$

$$b_2 = a_2 - \frac{(a_2, b_1)}{(b_1, b_1)} a_1$$

$$b_3 = a_3 - \frac{(a_3, b_1)}{(b_1, b_1)} a_1 - \frac{(a_3, b_2)}{(b_2, b_2)} a_2$$

.....

$$b_n = a_n - \frac{(a_n, b_1)}{(b_1, b_1)} a_1 - \dots - \frac{(a_n, b_{n-1})}{(b_{n-1}, b_{n-1})} a_{n-1}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

647

Schmidt正交化方法

$$\text{令 } c_i = \frac{b_i}{\|b_i\|} \quad a_1 = \|b_1\| c_1$$

$$a_2 = (a_2, c_1) c_1 + \|b_2\| c_2$$

$$a_3 = (a_3, c_1) c_1 + (a_3, c_2) c_2 + \|b_3\| c_3$$

$$\dots\dots$$

$$a_n = (a_n, c_1) c_1 + \dots + (a_n, c_{n-1}) c_{n-1} + \|b_n\| c_n$$

因此 $A = [a_1, \dots, a_n] = [c_1, \dots, c_n] R$ 其中 C 为正交阵

$$R = \begin{bmatrix} \|b_1\| & (a_2, c_1) & (a_3, c_1) & \dots & (a_n, c_1) \\ 0 & \|b_2\| & (a_3, c_2) & \dots & (a_n, c_2) \\ 0 & 0 & \|b_3\| & \dots & (a_n, c_3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \|b_n\| \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

648

方法二 平面旋转变换

引理1 对于向量 $x=(\alpha_1, \dots, \alpha_i, \dots, \alpha_j, \dots, \alpha_n)^T$,可以选择一个平面旋转矩阵 $P(i,j)$ 使得

$$Px=y=(\alpha_1, \dots, \alpha_i', \dots, \alpha_j', \dots, \alpha_n)^T,$$

其中 $\alpha_i'=(\alpha_i^2+\alpha_j^2)^{1/2}, \alpha_j'=0$.

根据引理1,我们可以用一个平面旋转矩阵使得一个向量的指定位置变为0.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

649

平面旋转变换

在 R^n 空间中,我们针对其中的两个分量实施旋转变换 $y=P_x$,其中

$$P = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & & \cos \theta & \sin \theta \\ & & & -\sin \theta & \cos \theta \\ & & & & 1 & \ddots \\ & & & & & 1 & \ddots \\ & & & & & & 1 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

650

平面旋转变换

具体的坐标变换格式为
$$\begin{cases} y_i = x_i \cos \theta + x_j \sin \theta \\ y_j = -x_i \sin \theta + x_j \cos \theta \\ y_k = x_k, \quad k \neq i, j \end{cases}$$

变换矩阵 P 具有如下性质

1. P 为正交阵
2. P 与单位阵只在 $(i,i), (i,j), (j,i), (j,j)$ 4个位置上的元素不一样
3. 对于向量 x, Px 只改变 x 的第 i 个及第 j 个分量.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

651

平面旋转变换

对于矩阵 A ,我们选择平面旋转矩阵 P_1 ,使得 a_{21} 变为0,再选择平面旋转矩阵 P_2 ,使得 a_{31} 变为0, ..., 选择平面旋转矩阵 P_{n-1} ,使得 a_{n1} 变为0.

于是 A 的第一列成为 σe_1 的形式.

类似的再变化第二列, ..., 第 $n-1$ 列.

最终得到定理19.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

652

平面旋转变换

定理19 如果 A 为非奇异矩阵,则存在正交矩阵 P_1, P_2, \dots, P_{n-1} (即一系列平面旋转矩阵)使

$$P_{n-1} \cdots P_2 P_1 A = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{bmatrix} \triangleq R$$

且 $r_{ii} > 0 (i=1, 2, \dots, n-1)$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

653

矩阵的QR分解

定理20 如果 $A \in R^{n \times n}$ 为非奇异矩阵,则 A 可分解为一正交阵与上三角矩阵 R 的乘积,即 $A=QR$,且当对角元素都为正数时分解唯一.

证明: 根据定理19,存在正交阵 P_1, P_2, \dots, P_{n-1} ,使得 $P_{n-1} \cdots P_2 P_1 A = R$ 为上三角阵.

令 $Q=(P_{n-1} \cdots P_2 P_1)^T$,则 $Q^T A = R$,即 $A=QR$.

唯一性: 若 $A=Q_1 R_1=Q_2 R_2$,其中 Q_1, Q_2 为正交阵, R_1, R_2 为非奇异上三角阵.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

654

矩阵的QR分解

$$A=Q_1R_1=Q_2R_2,$$

$$\text{于是, } Q_2^T Q_1 = R_2 R_1^{-1}.$$

从而上三角阵 $R_2 R_1^{-1}$ 是正交阵.

因此该矩阵为对角矩阵(根据正交矩阵的定义以及矩阵相乘的定义).

$$\text{即 } R_2 R_1^{-1} = D = \text{diag}[d_1, d_2, \dots, d_n].$$

由于该矩阵为正交阵, 所以 $D_2 = I$.

又 R_1, R_2 的对角元为正数, $d_i > 0$, 所以 $d_i = 1, D = I$.

因此 $R_1 = R_2$, 进一步有 $Q_1 = Q_2$.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

655

Householder变换

对于一般的矩阵 A , 直接采用Householder变换(初等反射阵相乘)的方法可以将矩阵的第一列化成为 σe_1 的形式.

$$\text{即 } H[a_1, a_2, \dots, a_n] = [-\text{sgn}(a_{11})\sigma e_1, b_2, \dots, b_n],$$

其中 σ 为 a_1 的长度.(计算时添上 $-\text{sgn}(a_{11})$ 是为了保证舍入误差尽量小)

通过一系列的Householder变换可以将矩阵进行QR分解.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

656

Householder方法算例

用Householder方法对矩阵 A 进行QR分解.

$$A = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 3 & 1 \\ 0 & 1 & 4 \end{bmatrix}$$

$$\text{解: } a_1 = (2, 1, 0)^T, \sigma_1 = \sqrt{5} \approx 2.236, a_{11} = 2 > 0,$$

选取 H_1 , 使得 $H_1 a_1 = -\sigma_1 e_1 = (-\sqrt{5}, 0, 0)$,

$$\text{令 } w_1 = \frac{a_1 + \sigma_1 e_1}{\|a_1 + \sigma_1 e_1\|} = \frac{(4.236, 1, 0)^T}{4.352} = (0.973, 0.230, 0)^T$$

$$H_1 = I - 2w_1 w_1^T = \begin{bmatrix} -0.895 & -0.447 & 0 \\ -0.447 & -0.895 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

657

Householder方法算例

$$A_2 = H_1 A_1 = \begin{bmatrix} -2.237 & -2.237 & -0.447 \\ 0 & 2.236 & 0.894 \\ 0 & 1 & 4 \end{bmatrix}$$

下面求二阶初等反射阵约化 A_2 右下角的二阶矩阵.

$$a_2 = (2.236, 1)^T, \sigma_2 \approx 2.449, a_{22} > 0,$$

选取 H_2 , 使得 $H_2 a_2 = -\sigma_2 e_2 = (-2.449, 0)$,

$$\text{令 } w_2 = \frac{a_2 + \sigma_2 e_2}{\|a_2 + \sigma_2 e_2\|} = \frac{(4.685, 1)^T}{4.791} = (0.978, 0.209)^T$$

$$H_2 = I - 2w_2 w_2^T = \begin{bmatrix} -0.912 & -0.408 \\ -0.408 & 0.912 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

658

Householder方法算例

$$A_3^{(2)} = H_2 A_2^{(2)} = \begin{bmatrix} -2.449 & -2.449 \\ 0 & 3.287 \end{bmatrix}$$

$$\text{因此 } H = \begin{bmatrix} 1 & 0 \\ 0 & H_2 \end{bmatrix} H_1$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & -0.912 & -0.408 \\ 0 & -0.408 & 0.912 \end{bmatrix} \begin{bmatrix} -0.895 & -0.447 & 0 \\ -0.447 & -0.895 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -0.895 & -0.447 & 0 \\ 0.408 & 0.816 & -0.408 \\ 0.182 & 0.365 & 0.912 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

659

Householder方法算例

根据上面的过程得到 $HA = R$, 其中

$$H = \begin{bmatrix} -0.895 & -0.447 & 0 \\ 0.408 & 0.816 & -0.408 \\ 0.182 & 0.365 & 0.912 \end{bmatrix} R = \begin{bmatrix} -2.237 & -2.237 & -0.447 \\ 0 & -2.449 & -2.449 \\ 0 & 0 & 3.287 \end{bmatrix}$$

因此 $A = H^{-1} R = H^T R = QR$, 其中

$$Q = H^T = \begin{bmatrix} -0.895 & 0.408 & 0.182 \\ -0.447 & 0.816 & 0.365 \\ 0 & -0.408 & 0.912 \end{bmatrix}$$

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

660

求特征值的QR算法

给定矩阵 A ,对 $A_1=A$ 进行QR分解,得 $A_1=Q_1R_1$,
令 $A_2=R_1Q_1$,再对 A_2 进行QR分解得 $A_2=Q_2R_2$,
...

对 A_k 进行QR分解得 $A_k=Q_kR_k$,令 $A_{k+1}=R_kQ_k$.
在一定条件下(定理22), $k \rightarrow \infty$ 时, A_k 的下三角部分趋于0,对角线元素趋于 A 的特征值.

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

661

算例

用QR算法求矩阵
的全部特征值.

$$A = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 3 & 1 \\ 0 & 1 & 4 \end{bmatrix}$$

解:采用matlab软件进行计算,终止准则为

$$|a_{n,n-1}^{(k)}| \leq \varepsilon \min(|a_{nn}^{(k)}|, |a_{n-1,n-1}^{(k)}|), \varepsilon = 10^{-5}$$

程序见下页.(注:Matlab中QR分解不保证R
的对角元为正)

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

662

```
a=[2,1,0;1,3,1;0,1,4];
n=3; l=1; eps=1e-5; kmax=100; k=0; tic;
while l==1 & k<kmax
    k=k+1; [q,r]=qr(a); a=r*q;
    if abs(a(n,n-1))<eps*min(abs(a(n,n)),abs(a(n-1,n-1)))
        l=0;
    end
end
toc;
```

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

663

运行结果

程序运行15步终止.得

$$A^{(15)} = \begin{bmatrix} 4.73203589027303 & 0.00508303962188 & 0 \\ 0.00508303962188 & 3.00001491721805 & -0.00001160807538 \\ 0 & -0.00001160807538 & 1.26794919250892 \end{bmatrix}$$

得 A 的三个特征值的近似值为

4.7320359,3.0000149,1.2679492

精确值为

4.7320508,3.0000000,1.2679492

南京邮电大学数理学院杨振华制作 njuptshumo2006@126.com

664