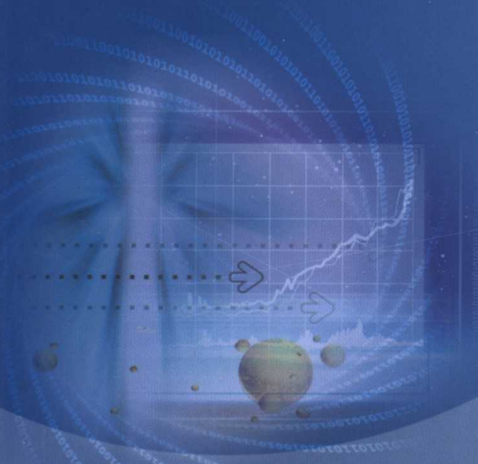
高等学校计算机科学与技术教材

计算方法

主　编　吴筑筑
副主编　于江明



清华大学出版社
http://www.tup.tsinghua.edu.cn
北京交通大学出版社
http://press.bjtu.edu.cn

高等学校计算机科学与技术教材

计　算　方　法

主　编　吴筑筑
副主编　于江明
参　编　黄玉昌　成　舜

清华大学出版社
北京交通大学出版社
·北京·

内 容 简 介

本书着重介绍计算机上常用的数值计算方法。内容包括误差和范数的知识、一元非线性方程的解法、线性方程组的解法、矩阵特征值的计算、插值法、曲线拟合和B样条曲线、数值积分和微分、常微分方程和方程组初值问题数值解法等方面的基础知识。

全书分为7章。常用算法给出编程计算步骤，并有用C语言编写的参考程序，便于上机实验。各章有较多例题和习题，附录中给出习题答案及用MATLAB解决常用数值计算问题的例子。全书叙述由浅入深，文字通俗，便于自学。本书适合作为高等学校计算机专业开设计算方法课程的教材，也适合其他理工科专业计算方法课程使用。

图书在版编目（CIP）数据

计算方法/吴筑筑主编；于江明副主编．—北京：清华大学出版社；北京交通大学出版社，2004.2

（高等学校计算机科学与技术教材）

ISBN 7-81082-218-7

Ⅰ．计…　Ⅱ．①吴…　②于…　Ⅲ．电子计算机－计算方法－高等学校－教材
Ⅳ．TP301.6

中国版本图书馆CIP数据核字（2003）第096628号

责任编辑：谭文芳
出 版 者：清 华 大 学 出 版 社　　邮编：100084　　电话：010－62776969
　　　　　北京交通大学出版社　　邮编：100044　　电话：010－51686045，62237564
印 刷 者：北京瑞达方舟印务有限公司
发 行 者：新华书店总店北京发行所
开　　本：787×1 092　1/16　　印张：10　　字数：256千字
版　　次：2004年2月第1版　　2004年2月第1次印刷
印　　数：1～5 000册　　定价：15.00元

前　言

伴随着现代社会的高速发展，在工程技术、自然科学和社会科学中涌现出越来越多的数值计算问题。把繁杂的计算问题交给计算机去完成，这已是人们的共识。因此，如何选择或构造合适的计算方法，并正确地在计算机上实现，以及如何估计计算结果的可靠程度，这都是科技工作者需要掌握的基本知识。

本教材是根据新时期普通高等学校计算机专业发展的需要而编写的。内容安排上强调各种数值算法的构造思想及用法，在力求严谨的同时，注意简化理论性的论证过程。书中提供了较多的例题和习题，并给出大多数习题的答案，便于学习时参考。本书注意突出应用技能的训练，各章主要算法都给出编程计算步骤，并设计了较多的上机实验题目，可供选用。附录A中给出了一些基本数值方法的Turbo C程序，并有文字说明，以利于学生理解数值算法的编程特点。附录B中介绍了如何应用国际流行的数学和工程计算软件MATLAB解决本教材涉及的数值问题的例子，希望这有助于学生提高解决数值计算问题的能力。

全书分为7章及3个附录。第1章介绍数值方法的特点、误差和范数等有关预备知识；第2章介绍一元非线性方程的几种基本解法；第3章为线性代数方程组的解法；第4章为矩阵特征值问题的算法；第5章介绍插值法、数据拟合及B样条曲线；第6章介绍数值积分法和基本的数值微分法；第7章介绍常微分方程（组）初值问题的数值解法。附录A为上机实验参考程序；附录B是用MATLAB进行数值计算的例子，可作为课外上机实验的内容。对于学时数安排较少的专业，可根据情况不讲第4章，或自行选择少讲一些理论性较强的内容。

本书由吴筑筑任主编，于江明任副主编。第2章由成舜编写，第6章由黄玉昌编写，第4、7章由于江明编写。吴筑筑编写第1、3、5章及附录，并统编全书。插图由吴筑筑等绘制。

本书的编写得到骆耀祖、苗雪兰、谭信民等同志的热情支持和鼓励，邓秀勤和谭信民为本书提供了部分资料，曾浦华打印了部分书稿并绘制部分插图。在此一并表示诚挚的谢意。

由于水平所限，必有疏漏之处，敬请读者指正。

编　者

2004年1月

目　录

第1章 预备知识

1.1 数值计算方法引论

电子计算机是20世纪最伟大、影响最深远的发明。几十年来,它的应用已深入到人类活动的各个领域。随着科学技术的高速发展,越来越多的实际问题需要通过数值计算进行模拟或定量分析,而且问题的规模越来越大,对计算精度的要求越来越高。如何利用计算机高效率地处理这些数值计算问题,是计算机应用学科的一项重要任务。

数值计算方法也简称计算方法,其目的是用简单的运算解决复杂的数值问题,它是利用计算机求解实际应用中提出的数学问题近似解的方法。

通常,用计算机解决科学计算问题经历以下过程:

$\boxed{\text{实际问题}} \to \boxed{\text{建立数学模型}} \to \boxed{\text{构造数值计算方法}} \to \boxed{\text{程序设计}} \to \boxed{\text{上机计算结果}}$

计算方法是从数学模型到计算结果之间的关键环节。一些科学家指出,人类的计算能力等于计算工具的效率与计算方法的效率的乘积[1]。例如,如果没有高效率的计算方法及由其转化而来的计算软件,计算机辅助设计技术的产生和发展是不可能的。又如从20世纪50年代初到90年代中期,计算机硬件的运算速度大致提高了1亿倍,而求解在工程中大量出现的椭圆型偏微分方程的算法的速度却提高了1万亿倍,这显然得益于计算方法效率的提高。可见计算方法的进步和计算机硬件的发展对于提高人的计算能力具有同等重要的地位。

计算方法课程主要讨论如何构造求数学模型近似解的算法,讨论算法的数学原理、误差和复杂性,配合程序设计进行计算试验并分析试验结果。本书只介绍一些常用的、基本的数值计算方法。例如一元非线性方程的解法,求解线性方程组的算法,矩阵特征值的算法,插值法、B样条曲线和曲线拟合法,定积分和导数的近似计算,常微分方程的近似解法等。

与纯数学的理论方法不同,用数值计算方法所求出的答案一般不是解的精确值或者准确的解析表达式,而是所求真解的某些近似值或近似曲线。例如方程

$$x^2 = 2\sin x$$

在区间(1,2)内有惟一根,但找不出求根的解析式,只能用数值计算方法求其近似解。又如湖水的水温直接影响鱼类的生存,环境工程师发现湖水的水温与深度有关,经观测得到某湖在一些不同深处的水温,希望能算出未观测到的任意深度的水温。用理论方法无法得到水温与深度的函数关系式,而运用计算方法中的插值法或曲线拟合法就可以在计算机上编程算出所需要的水温近似值。有些数学问题虽有理论上准确的公式解,但不一定实用。例如,求解线性方程组的Cramer法则,由于计算量太大而并不适宜作为数值计算公式。

在计算机上进行算术运算与严格的精确计算很不相同。

例如,在纯数学中,众所周知只有 $b=0$ 才能满足 $a+b=a$,而在一台10位十进制计算机上计算时,却出现 $10^{10}+4=10^{10}$,$10^{10}+1=10^{10}$,……等“大数吃掉小数”的现象。因此,有必要了解计算机上进行算术运算的特点。

一般,一台 t 位10进制计算机上能表示的实数 x 都可以写成**规格化浮点数**的形式:

$$x = \pm 0.d_1d_2\cdots d_t \times 10^n$$

其中 10 称为**基底**，$d_1,d_2,\cdots,d_t$ 都是 0 到 9 中的数字，当 $x\neq 0$ 时 $d_1\neq 0$。称 $0.d_1d_2\cdots d_t$ 为数 x 的**尾数**，t 为计算机的**字长**。整数 n 称为数 x 的**阶**，它表示数 x 的数量级，若将尾数的小数点移动 n 位就得到数 x。数 0 在计算机中用尾数为 0 的浮点数表示，其阶可以是允许范围内的任意值。

每种计算机有各自规定的基底、字长及阶的范围。所以，计算机中能表示的实数全体是一个有限集。当输入数据或中间结果不在这个集合内时，机器自动将它舍入成集合内最接近它的数，因而产生**舍入误差**。当一个数的阶 n 超出规定的范围就造成"溢出"。

例如，无限小数 e 在 4 位十进制计算机上规格化为 $e=0.2718\times 10^1$，其尾数是 0.2718，阶为 1。而在 10 位十进制机上 $e=0.271\,828\,182\,8\times 10^1$。

两个数量级不同的数相加减时需要**对阶**，将阶码统一为较大者，然后才能将尾数相加减。如前面提过的例子：在 10 位十进制计算机上计算

$$\begin{aligned}10^{10}+4 &= 0.100\,000\,000\,0\times 10^{11} + 0.400\,000\,000\,0\times 10^1 \quad（规格化）\\ &= 0.100\,000\,000\,0\times 10^{11} + 0.000\,000\,000\,0\times 10^{11} \quad（对阶）\\ &= 10^{10}\end{aligned}$$

由于计算机按其限定的字长进行运算，数 4 在对阶后被当做 0"吃掉"了。

例 1-1　在 3 位十进制计算机上计算 0.0275，0.0329，12.7 的和（准确和是 12.7604）。

算法 1　$(0.275\times 10^{-1}+0.329\times 10^{-1})+0.127\times 10^2$

$$\begin{aligned}&=0.604\times 10^{-1}+0.127\times 10^2\\ &=0.001\times 10^2+0.127\times 10^2 \quad（对阶）\\ &=0.128\times 10^2\end{aligned}$$

算法 2　$0.275\times 10^{-1}+(0.329\times 10^{-1}+0.127\times 10^2)$

$$\begin{aligned}&=0.275\times 10^{-1}+(0.000\times 10^2+0.127\times 10^2) \quad（对阶）\\ &=0.275\times 10^{-1}+0.127\times 10^2\\ &=0.000\times 10^2+0.127\times 10^2 \quad（对阶）\\ &=0.127\times 10^2\end{aligned}$$

此例说明，通常实数运算的加法结合律在计算机上不成立。同样可举出例子说明乘法对加法的分配律也不成立。

例 1-2　在8位十进制计算机上求解方程

$$x^2-10^5x+1=0$$

算法 1　直接用求根公式计算，由于 $\sqrt{10^{10}-4}=10^5$，所以求得

$$x_1=(10^5+\sqrt{10^{10}-4})/2=100\,000.00,\ x_2=(10^5-\sqrt{10^{10}-4})/2=0$$

算法 2　x_1 同上，计算 x_2 利用根与系数的关系 $x_1x_2=1$，得

$$x_2=1/x_1=0.000\,010\,000\,000$$

两个算法得出了不同结果。事实上方程的两根取 11 位有效数字是

$$x_1=99\,999.999\,990,\ x_2=0.000\,010\,000\,000\,001$$

方程显然并无 0 根，可见算法 1 的结果是不能接受的。原因是计算 x_2 时发生了相近两数 10^5 和 $\sqrt{10^{10}-4}$ 相减的现象，而根号下的计算"大数吃掉小数"又不能避免，以致计算结果失真。算法 2 避免了相近两数相减，所以得到较为准确的结果。此例也说明有些问题尽管在纯数学

中早已有公式解，却未必适合用做计算机算法，舍入误差的影响是不可忽视的。

计算量的大小也是计算方法中需要考虑的。例如计算多项式

$$P_n(x)=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0$$

的值，若直接计算共需做$\frac{1}{2}n(n+1)$次乘法和 n 次加法。若采用“秦九韶算法”：

$$P_n(x)=(\cdots((a_nx+a_{n-1})x+a_{n-2})x+\cdots+a_1)x+a_0$$

令
$$\begin{cases} u_k=u_{k-1}x+a_{n-k}, & (k=1,2,\cdots,n) \\ u_0=a_n \end{cases}$$

于是 $P_n(x)=u_n$，则只需 n 次乘法和 n 次加法。

总之，计算方法要面向计算机应用，因此必须考虑算法在计算机上是否容易实现，计算过程的误差是否可控制，算法的收敛性和稳定性，以及计算的复杂度即计算量和存储空间的大小等问题。

1.2 误差

1.2.1 科学计算问题中误差的来源

用计算机解决科学计算问题的每一步都可能产生误差。误差的来源主要有模型误差、观测误差、截断误差和舍入误差。

模型误差是由于数学建模过程中往往要忽略一些次要的因素而产生的，它是实际问题的客观量与其理论数学模型精确解之间的差。在数学模型中的数据如果是从观测得到的，由此产生的误差叫做**观测误差**。这两类误差在计算方法中一般不予讨论。

当一个数学模型得不到精确解时，要用数值计算方法求它的近似解，数学模型的精确解与近似算法的解之间的差称为该算法的**截断误差**或**方法误差**。例如：

$$\sin x=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}+\cdots,\qquad (-\infty<x<\infty)$$

当$|x|$很小时，常用近似公式 $\sin x\approx x$ 计算，于是产生截断误差 $\sin x-x$，按绝对值约为$\frac{1}{6}|x|^3$。在计算方法中，构造任一个近似算法都应考虑其截断误差。

舍入误差是由于计算机字长有限，所有输入数据及每步运算的结果都按字长的限定进行舍入而产生的误差。在 1.1 节中已提到过。

例如，在 4 位十进制计算机上用 2.718 作为 e 的近似值，产生舍入误差 $e-2.718=0.00028\cdots$。而 $10e\approx10\times2.718=27.18$。与准确值 $10e=27.1828\cdots$ 相比较，计算值有误差 $10e-27.18=0.0028\cdots$。初始数据的舍入误差经计算过程积累而被扩大 10 倍。

在数值计算中往往要进行成千上万次乃至亿万次四则运算，舍入误差的积累可能对结果造成很大影响，在构造算法时应十分注意。

1.2.2 误差的估计方式

估计误差的方式主要是绝对误差(限)、相对误差(限)和有效数字位数。

定义 1-1 设某量的准确值为 x^*，近似值为 x，则称差 $\varepsilon=x^*-x$ 为近似值 x 的**绝对误**

差(简称**误差**)。

由于准确值 x^* 一般不能得到,绝对误差也就无法具体确定。通常用 $|\varepsilon|$ 的适当小的上界 ξ 来度量绝对误差,称 ξ 为近似值 x 的**绝对误差限**,简称**误差限**。显然 ξ 不惟一,它满足

$$|\varepsilon| = |x^* - x| \leqslant \xi \tag{1-1}$$

在衡量近似值的精度时,常取 ξ 为满足上式的形如 0.5×10^k(k 是整数)的数中较小者。

利用近似值 x 及其绝对误差限 ξ,可以将准确值 x^* 的所在范围表示成

$$x-\xi\leqslant x^*\leqslant x+\xi, \quad x^* = x\pm\xi$$

后一种写法常出现在实际应用中。例如,某商品外包装标出其重量为 24 kg ± 0.2 kg,则商品重量大约为 24 kg,绝对误差限为 0.2 kg,商品实际重量在 23.8 kg 到 24.2 kg 的范围内。

在同一个量的不同近似值中,$|\varepsilon|$ 越小者精度越高。但对于不同量的近似值,绝对误差限不能反映近似值的准确程度。例如,测得两物长度分别为 100 cm 和 50 cm,误差限都是 1 cm,显然前者的误差占准确数的比例较小,因而更准确些。为此需要相对误差的概念。

定义 1-2 设准确值 x^* 的近似值为 x,则称

$$\varepsilon_r = \frac{\varepsilon}{x^*} = \frac{x^* - x}{x^*}, \qquad (x^*\neq0)$$

为 x 的**相对误差**。

在同一个量或不同量的几个近似值中,$|\varepsilon_r|$ 较小者精度较高。x 的相对误差的绝对值的较小上界称为 x 的**相对误差限**,记为 η,它满足:

$$|\varepsilon_r| = \frac{|\varepsilon|}{|x^*|} = \frac{|x^* - x|}{|x^*|}\leqslant\eta \tag{1-2}$$

相对误差(限)常用百分数表示。由于准确值 x^* 一般得不到,实际应用中用近似值 x 代替分母中的 x^* 估计相对误差。相对误差限可用下式估计:

$$\eta = \frac{\xi}{|x|} \tag{1-3}$$

例 1-3 设 $x=1000\pm1$ cm, $y=-50\pm0.5$ cm,问哪个量的近似值较为准确?

解 $x\approx1000$ cm, $\quad \xi(x)=1$ cm, $\quad \eta(x)=1/1000=0.001$

$y\approx-50$ cm, $\quad \xi(y)=0.5$ cm, $\quad \eta(y)=0.5/50=0.01$

由于 $\eta(x)<\eta(y)$,所以 x 的近似值较为准确。

近似值 x 的绝对误差可看做 x 的增量,因此可以利用微分来估计误差。即

$$\varepsilon = x^* - x\approx \mathrm{d}x \tag{1-4}$$

例 1-4 测得某正方体边长为 20 mm,误差限 0.5 mm。试估计其体积的误差限及相对误差限。

解 设边长为 x,体积为 $S=x^3$,由已知有 $x=20$, $|\varepsilon(x)|\leqslant0.5$,因

$$\varepsilon(S)\approx\mathrm{d}S=3x^2\mathrm{d}x\approx3x^2\varepsilon(x)$$

于是

$$|\varepsilon(S)|\approx3x^2|\varepsilon(x)|\leqslant3\times20^2\times0.5=600\ \mathrm{mm}^3$$

$$|\varepsilon_r(S)|=|\varepsilon(S)|/|S|\leqslant600/20^3=0.075=7.5\%$$

所以体积的误差限为 600 mm³,相对误差限为 7.5%。

定义 1-3 若近似值 x 的绝对误差限是某一数位上的半个单位,则说 x **精确到该位**,若从该位到 x 的左起第一位非零数字一共有 n 位,则称近似值 x 有 n 位**有效数字**。

准确数被认为有无穷多位有效数字。如果 x 是将准确数直接四舍五入到某个数位而得，则绝对误差限是该数位的半个单位，所以 x 精确到该位。

需注意近似值末尾的 0 不能省去。例如，设近似值 0.008 300，0.0083 分别由准确值四舍五入而得，则前者精确到 10^{-6}，有 4 位有效数字，后者精确到 10^{-4}，只有 2 位有效数字。

例 1-5 设分别取 $x_1=3.14, x_2=3.1415$ 作为 π 的近似值，试估计绝对误差，并求它们各有多少有效数字？

解 准确值 $x^*=\pi=3.141\,592\,65\cdots$，$x_1=3.14$，$|\varepsilon_1|=|\pi-3.14|=0.001\,59\cdots\leqslant 0.5\times 10^{-2}$，绝对误差限为 0.5×10^{-2}，它是 10^{-2}位上的半个单位，所以 $x_1=3.14$ 精确到 10^{-2}，有 3 位有效数字。

$x_2=3.1415$，$|\varepsilon_2|=|\pi-3.1415|=0.000\,092\cdots\leqslant 0.5\times10^{-3}$，绝对误差限为 0.5×10^{-3}，所以 $x_2=3.1415$ 精确到 10^{-3}，有 4 位有效数字。

定义 1-3 的一种等价说法是定义1-3′。

定义 1-3′ 设 x^* 的近似值 $x=\pm 0.\alpha_1\alpha_2\cdots\alpha_n\cdots\times 10^p$，其中 p 为整数，各 α_i 是 0 到 9 中的数字，$\alpha_1\neq 0$。若 x 的绝对误差限为 $0.5\times10^{p-n}$，即

$$|\varepsilon|=|x^*-x|\leqslant 0.5\times 10^{p-n} \tag{1-5}$$

则称近似值 x 精确到 10^{p-n}位，具有 n 位有效数字。

事实上，若式(1-5)成立，则 x 的绝对误差限不超过 10^{p-n}这个数位上的半个单位，根据定义 1-3 可知 x 精确到 10^{p-n}位。而 x 的 10^{p-n}位正是α_n 所在的数位，从这一位到左起第一位非零数字 α_1 共有 n 位，所以 x 具有 n 位有效数字。

例 1-6 近似数 $x=0.020\,570\,2$ 的绝对误差限为 0.5×10^{-5}，它具有多少位有效数字？

解法 1 根据定义1-3。由绝对误差限可知 x 精确到 10^{-5}，x 具有 4 位有效数字。

解法 2 根据定义1-3′。近似数写成 $x=0.205\,702\times10^{-1}$，于是 $p=-1$，$p-n=-5$，所以 $n=4$，x 具有 4 位有效数字。

有效数字位数与相对误差限有下面的关系。

定理 1-1 设近似值 x 的左起第一位非零数字是α_1。若 x 具有 n 位有效数字，则

$$\frac{1}{2\alpha_1}\times 10^{-n+1}$$

为 x 的相对误差限。

证 将 x 写成$x=\pm 0.\alpha_1\alpha_2\cdots\alpha_n\cdots\times10^p$，已知 x 具有 n 位有效数字，则

$$|\varepsilon|=|x^*-x|\leqslant 0.5\times10^{p-n}$$

$$|\varepsilon_r|=\frac{|\varepsilon|}{|x|}\leqslant\frac{0.5\times10^{p-n}}{0.\alpha_1\alpha_2\cdots\times10^p}\leqslant\frac{0.5\times10^{-n}}{0.\alpha_1}=\frac{1}{2\alpha_1}\times10^{-n+1}$$

所以定理成立。

上述定理说明有效数字位数越多，相对误差限越小。反之也可以证明，相对误差限越小，有效数字位数越多。所以，**有效数字位数可作为近似值精度的一种衡量标准。**

例 1-7 用四舍五入法取 π 的近似值，使相对误差不超过 0.52%。

解 设近似值取 n 位有效数字，则相对误差限 $\eta=\frac{1}{2\alpha_1}\times10^{-n+1}$，其中 $\alpha_1=3$，令 $\eta\leqslant$ 0.52%，即

$$\frac{1}{2\times3}\times10^{-n+1}\leqslant 0.005\,2,\qquad 10^n\geqslant 1/0.003\,12\approx 320.5$$

故正整数 $n \geqslant 3$。π 的近似值取 3 位有效数字可以满足要求。所以取 $\pi \approx 3.14$。

有些数值计算问题指定“用 t 位有效数字计算”(t 是给定的正整数)。这是指计算过程中,每个数都当做准确值并四舍五入到从左边第一位非零数字起向右数的第 t 位。换言之,即在虚设的字长为 t 的计算机上进行计算。而实际计算结果的有效数字往往少于 t 位。

例如,用 3 位有效数字计算时,$0.123\,456-0.112\,324 \approx 0.123-0.112=0.011\,0$。计算结果“用 3 位有效数字”形式输出。而准确值是 0.011 132,所得近似值的绝对误差限 $|0.011\,132-0.011\,0|=0.000\,132<0.5\times10^{-3}$,因此实际计算结果 0.0110 只有 2 位有效数字。

1.3 算法的数值稳定性

若一个算法的计算过程中,误差的传播可以控制,计算结果受误差传播的影响小,则称该算法的数值稳定性较好,算法是数值稳定的,否则称它的数值稳定性不好,算法是数值不稳定的。

例如,在例 1-2 中算法 2 是数值稳定的,算法 1 的数值稳定性不好。为防止误差的传播使结果失真,要注意选用数值稳定的算法。以下给出设计算法的一些原则。

1. 要避免相近两数相减

由于两数差的绝对误差 $\varepsilon(x-y) \approx \mathrm{d}(x-y)=\mathrm{d}x-\mathrm{d}y \approx \varepsilon(x)-\varepsilon(y)$,故

$$|\varepsilon_r(x-y)|=\frac{|\varepsilon(x)-\varepsilon(y)|}{|x-y|} \leqslant \frac{|x|}{|x-y|}\frac{|\varepsilon(x)|}{|x|}+\frac{|y|}{|x-y|}\frac{|\varepsilon(y)|}{|y|}$$

可见当数 $x \approx y$ 时,差 $x-y$ 的相对误差限可能会很大,造成有效数字严重受损。

例如,设 $x^*=0.012\,346$,$y^*=0.012\,342$,在 4 位计算机上两数很接近,它们的近似值 $x=0.012\,35$,$y=0.012\,34$ 都有 4 位有效数字。计算 x^*-y^* 的近似值得

$$x-y=0.012\,35-0.012\,34=0.000\,010\,00$$

而准确值为 $x^*-y^*=0.000\,004$,所以近似值的绝对误差限:

$$|0.000\,004-0.000\,010\,00|=0.000\,006<0.5\times10^{-4}$$

这说明近似值 0.000 010 00 连 1 位有效数字也没有。原因在于相近两数相减使有效数字丢失了。例 1-2 中也遇到过类似情况。

现举例说明避免相近两数相减的一些办法。

(1)对于一元二次方程 $ax^2+bx+c=0$,为避免发生相近两数相减造成数值不稳定,用以下算法较好:

$$x_1=\frac{-b+\mathrm{sign}(-b)\sqrt{b^2-4ac}}{2a},\quad x_2=\frac{c}{ax_1}$$

(2)当 $x \gg y>0$ 时,算式 $\sqrt{x+y}-\sqrt{x}$ 改为 $\dfrac{y}{\sqrt{x+y}+\sqrt{x}}$ 较好。

(3)当 x 接近于 0 时,算式 $1-\cos x$ 可改为 $2\sin^2\dfrac{x}{2}$。

(4)当 $|x| \ll 1$ 时,计算 e^x-1 可用 Taylor 级数 $x+\dfrac{x^2}{2}+\dfrac{x^3}{6}+\cdots$,取前面若干项计算。

2. 要防止“大数吃掉小数”

例如,将数量级相差很大的一些数相加时,为了避免小数被大数“吃掉”,应尽可能调整顺

序，先计算数量级低的小数之和，再与大数相加。

3. 避免用绝对值很小的数做除数

由于 $\varepsilon(\frac{x}{y})\approx d(\frac{x}{y})\approx\frac{y\varepsilon(x)-x\varepsilon(y)}{y^2}$，当除数的绝对值很小时，$\left|\varepsilon(\frac{x}{y})\right|$会很大，影响结果的精度，所以要尽量避免。例如：

当 $x\gg a>0$ 时，算式$\frac{1}{\sqrt{x+a}-\sqrt{x}}$改为$\frac{\sqrt{x+a}+\sqrt{x}}{a}$较好。

4. 注意简化计算步骤，减少运算次数

秦九韶算法就是一例。减少运算次数可加快计算速度并减少误差积累。

5. 注意控制递推公式中误差的传播

例如：计算积分 $I_n=\int_0^1\frac{x^n}{x+10}dx$，$n=0,1,\cdots,10$。容易得到递推公式

$$I_0=\ln 1.1,\qquad I_n=\frac{1}{n}-10I_{n-1},\qquad (n=1,2,\cdots,10)$$

若直接用上述递推公式计算，误差的传递规律为

$$\varepsilon(I_n)=-10\varepsilon(I_{n-1})$$

于是

$$\varepsilon(I_{10})=-10\varepsilon(I_9)=10^2\quad\varepsilon(I_8)=\cdots=10^{10}\varepsilon(I_0)$$

计算 I_0 时的误差被放大到 10^{10}倍，显然算法是数值不稳定的。

注意到 I_n 单调下降且

$$\frac{1}{11}\int_0^1 x^n dx<I_n<\frac{1}{10}\int_0^1 x^n dx$$

有

$$\frac{1}{11\times 11}<I_{10}<\frac{1}{10\times 11}$$

取近似值 $I_{10}=\frac{1}{2}(\frac{1}{11\times 11}+\frac{1}{10\times 11})$，再递推计算

$$I_{n-1}=(1-nI_n)/(10n),\qquad (n=10,\ 9,\ \cdots,\ 1)$$

则这个算法的误差传递规律为 $\varepsilon(I_{n-1})=-\varepsilon(I_n)/10$，即每计算一步的误差的绝对值是上一步的十分之一，误差的传播得到很好的控制，这个算法是数值稳定的。

1.4 向量和矩阵的范数

在线性方程组求解等问题和一些算法的误差分析、收敛性分析中常常要计算向量或矩阵。为了从整体上估计一个向量或矩阵的误差，需要有一种度量向量或矩阵大小的方法。因此引进范数的概念。

下面用记号 R^n 表示实数域 R 上全体 n 维向量构成的线性空间（简称为 n 维向量空间），用 $R^{n\times n}$表示实数域 R 上全体 n 阶矩阵构成的线性空间。

定义 1-4 设在 n 维向量空间 R^n 上定义一个实值函数，记做 $\|\boldsymbol{x}\|$，其中 $\boldsymbol{x}\in R^n$。若满足

(1)**非负性**：$\|\boldsymbol{x}\|\geqslant 0,\ \forall\,\boldsymbol{x}\in R^n$；等号当且仅当 $\boldsymbol{x}=\boldsymbol{0}$时成立；

(2)**齐次性**：$\|\lambda\boldsymbol{x}\|=|\lambda|\,\|\boldsymbol{x}\|,\ \forall\,\boldsymbol{x}\in R^n,\ \forall\,\lambda\in R$；

(3)**三角不等式**：即 $\|\boldsymbol{x}+\boldsymbol{y}\|\leqslant\|\boldsymbol{x}\|+\|\boldsymbol{y}\|,\ \forall\,\boldsymbol{x},\boldsymbol{y}\in R^n$。

则称函数 $\|x\|$ 为向量 x 的**范数**。

这样的函数一定存在。事实上，1 维向量空间就是实数域 R，绝对值函数 $|x|$ 显然满足定义中 3 条性质。因此，实数 x 的绝对值可以被称为实数 x 的一种范数。3 维向量空间 R^3 就是几何空间，熟知的向量长度是向量的实值函数，令

$$\|x\|_2=\sqrt{x_1^2+x_2^2+x_3^2},\qquad (\forall x\in R^3)$$

显然它满足定义的条件，所以 $\|x\|_2$ 是 3 维向量的一种范数。

将向量的长度推广到 n 维向量空间。令

$$\|x\|_p=\sum_{i=1}^{n}(|x_i|^p)^{\frac{1}{p}},\forall x=[x_1,x_2,\cdots,x_n]^{\mathrm{T}}\in R^n \tag{1-6}$$

则可以证明它满足范数的定义，称 $\|x\|_p$ 为 n 维向量 x 的 **l_p 范数**。常用的 l_p 范数有：

$$\|x\|_1=|x_1|+|x_2|+\cdots+|x_n|$$

$$\|x\|_2=\sqrt{x_1^2+x_2^2+\cdots+x_n^2}=\sqrt{x^{\mathrm{T}}x}$$

$$\|x\|_\infty=\lim_{p\to\infty}\|x\|_p=\max\{|x_1|,|x_2|,\cdots,|x_n|\}$$

分别称为向量的 **l_1 范数，l_2 范数，l_∞ 范数**。易见当 $n=3$ 时，l_2 范数就是向量的几何长度。

例如，设 $x=[1,-2,3]^{\mathrm{T}}$，则容易算出：

$$\|x\|_1=1+2+3=6,\ \|x\|_2=\sqrt{1+4+9}=\sqrt{14},\ \|x\|_\infty=\max\{1,2,3\}=3$$

定理 1-2　R^n 上任意两种范数 $\|\cdot\|_a$ 和 $\|\cdot\|_b$ 是等价的，即存在与 x 无关的数 m、M，使

$$m\|x\|_a\leqslant\|x\|_b\leqslant M\|x\|_a,\qquad \forall x\in R^n$$

例如，常用的 l_1 范数、l_2 范数、l_∞ 范数存在下列等价性关系式：

$$\|x\|_\infty\leqslant\|x\|_1\leqslant n\|x\|_\infty,\ \frac{1}{\sqrt{n}}\|x\|_1\leqslant\|x\|_2\leqslant\|x\|_1,\ \|x\|_\infty\leqslant\|x\|_2\leqslant\sqrt{n}\|x\|_\infty \tag{1-7}$$

在计算方法中，常常需要利用范数来讨论向量序列的收敛性。

定义 1-5　设有向量序列 $\{x^{(k)}\}$，其中 $x^{(k)}=(x_1{}^{(k)},x_2{}^{(k)},\cdots,x_n{}^{(k)})^{\mathrm{T}},k=0,1,2,\cdots$。若 $x^{(k)}$ 的每个分量都存在极限，$\lim\limits_{k\to\infty}x_i^{(k)}=x_i^*\ (i=1,2,\cdots,n)$，则称向量序列 $\{x^{(k)}\}$ 收敛于向量 $x^*=(x_1^*,x_2^*,\cdots,x_n^*)^{\mathrm{T}}$。

定理 1-3　向量序列 $\{x^{(k)}\}$ 收敛于向量 x^* 的充要条件是存在一种范数 $\|\cdot\|$，使

$$\lim_{k\to\infty}\|x^{(k)}-x^*\|=0$$

证　必要性：当序列 $\{x^{(k)}\}$ 收敛于向量 x^*，则 $x^{(k)}-x^*$ 收敛于 $\mathbf{0}$，因此有

$$\lim_{k\to\infty}\|x^{(k)}-x^*\|_\infty=\lim_{k\to\infty}\max_{1\leqslant i\leqslant n}|x_i^{(k)}-x_i^*|=0$$

充分性：当存在一种范数 $\|\cdot\|$，使 $\lim\limits_{k\to\infty}\|x^{(k)}-x^*\|=0$ 成立时，由向量范数的等价性，总存在常数 m、M，使

$$m\|x^{(k)}-x^*\|\leqslant\|x^{(k)}-x^*\|_\infty\leqslant M\|x^{(k)}-x^*\|$$

根据夹逼定理可知 $\lim\limits_{k\to\infty}\|x^{(k)}-x^*\|_\infty=0$，所以 $x^{(k)}-x^*$ 收敛于 $\mathbf{0}$，即 $\{x^{(k)}\}$ 收敛于向量 x^*。

定义 1-6　矩阵 A 的所有特征值的模中之最大者称做矩阵 A 的谱半径，记为 $\rho(A)$。

例如：$A=\begin{pmatrix}2&-3\\1&2\end{pmatrix}$，其特征值为 $2+\sqrt{3}i,2-\sqrt{3}i$，所以 $\rho(A)=\sqrt{7}$。

矩阵的谱半径有如下性质：

(1)$\rho(\boldsymbol{0})=0$，　$\rho(\boldsymbol{I})=1$

(2)$\rho(\boldsymbol{A}^m)=(\rho(\boldsymbol{A}))^m$

(3)当 $\boldsymbol{A}$ 为对称半正定矩阵时，其特征值全为非负实数，$\rho(\boldsymbol{A})$等于其最大特征值。

关于矩阵范数的一般定义与向量范数类似，但由于在数值分析中常需对矩阵乘积、矩阵与向量的乘积进行度量，因此要求矩阵范数满足以下性质。

(1)**非负性**：$\|\boldsymbol{A}\|\geqslant 0, \forall \boldsymbol{A}\in R^{n\times n}$；等号当且仅当 $\boldsymbol{A}=\boldsymbol{0}$(零矩阵)时成立。

(2)**齐次性**：$\|\lambda\boldsymbol{A}\|=|\lambda|\,\|\boldsymbol{A}\|, \forall \boldsymbol{A}\in R^{n\times n}, \forall\lambda\in R$。

(3)**三角不等式**：$\|\boldsymbol{A}+\boldsymbol{B}\|\leqslant\|\boldsymbol{A}\|+\|\boldsymbol{B}\|, \forall \boldsymbol{A},\boldsymbol{B}\in R^{n\times n}$。

(4)**与向量范数的相容性**：存在某种向量范数 $\|\cdot\|_a$ 使得 $\|\boldsymbol{A}\boldsymbol{x}\|_a\leqslant\|\boldsymbol{A}\|\,\|\boldsymbol{x}\|_a, \forall \boldsymbol{A}\in R^{n\times n}, \forall \boldsymbol{x}\in R^n$。

(5)**对于矩阵乘法的相容性**：$\|\boldsymbol{A}\boldsymbol{B}\|\leqslant\|\boldsymbol{A}\|\,\|\boldsymbol{B}\|, \forall \boldsymbol{A},\boldsymbol{B}\in R^{n\times n}$。

如下给出的矩阵范数定义便满足上述性质(证明从略)。

定义 1-7　设矩阵 $\boldsymbol{A}\in R^{n\times n}$，$\|\cdot\|_a$ 为R^n 中定义的某种向量范数，则称

$$\|\boldsymbol{A}\|_a=\max_{\|\boldsymbol{x}\|_a=1}\|\boldsymbol{A}\boldsymbol{x}\|_a,\qquad(\boldsymbol{x}\in R^n)$$

为矩阵 $\boldsymbol{A}$ 的从属于向量范数 $\|\cdot\|_a$ 的**矩阵范数**。

简言之，矩阵 $\boldsymbol{A}$ 的范数 $\|\boldsymbol{A}\|_a$ 是指函数 $\|\boldsymbol{A}\boldsymbol{x}\|_a$ 在约束条件 $\|\boldsymbol{x}\|_a=1$ 下的最大值。在上述定义下，零矩阵 $\boldsymbol{0}$ 的范数 $\|\boldsymbol{0}\|=\boldsymbol{0}$，单位矩阵 $\boldsymbol{I}$ 的范数 $\|\boldsymbol{I}\|=\boldsymbol{1}$。

定理 1-4　设 $\boldsymbol{A}=(a_{ij})_n\in R^{n\times n}$，则从属于向量 l_1,l_2,l_∞范数的矩阵范数分别为

$$\|\boldsymbol{A}\|_1=\max_{\|\boldsymbol{x}\|_1=1}\|\boldsymbol{A}\boldsymbol{x}\|_1=\max_{1\leqslant j\leqslant n}\sum_{i=1}^{n}|a_{ij}|\tag{1-8}$$

$$\|\boldsymbol{A}\|_2=\max_{\|\boldsymbol{x}\|_2=1}\|\boldsymbol{A}\boldsymbol{x}\|_2=\sqrt{\rho(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})}\tag{1-9}$$

$$\|\boldsymbol{A}\|_\infty=\max_{\|\boldsymbol{x}\|_\infty=1}\|\boldsymbol{A}\boldsymbol{x}\|_\infty=\max_{1\leqslant i\leqslant n}\sum_{j=1}^{n}|a_{ij}|\tag{1-10}$$

它们分别被称为矩阵 $\boldsymbol{A}$ 的 1-范数、2-范数、∞-范数。根据各个表达式的特点，1-范数又称为列范数，2-范数又称为谱范数，∞-范数又称为行范数。

证　只证明式(1-8)。设 $\boldsymbol{A}$ 按列分块为$\boldsymbol{A}=(\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_n)$，其中 $\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_n$ 为列向量。记

$$M=\max_{1\leqslant j\leqslant n}\sum_{i=1}^{n}|a_{ij}|=\max_{1\leqslant j\leqslant n}\|\boldsymbol{a}_j\|_1$$

若 $\|\boldsymbol{x}\|_1=1, \boldsymbol{x}=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}\in R^n$，则有

$$\begin{aligned}\|\boldsymbol{A}\boldsymbol{x}\|_1&=\|x_1\boldsymbol{a}_1+x_2\boldsymbol{a}_2+\cdots+x_n\boldsymbol{a}_n\|_1\leqslant|x_1|\,\|\boldsymbol{a}_1\|_1+|x_2|\,\|\boldsymbol{a}_2\|_1+\cdots+|x_n|\,\|\boldsymbol{a}_n\|_1\\&\leqslant(|x_1|+|x_2|+\cdots+|x_n|)\max_{1\leqslant j\leqslant n}\|\boldsymbol{a}_j\|_1=\|\boldsymbol{x}\|_1\max_{1\leqslant j\leqslant n}\|\boldsymbol{a}_j\|_1=M\end{aligned}$$

这说明 M 是函数 $\|\boldsymbol{A}\boldsymbol{x}\|_1$ 的上界。另一方面，设

$$M=\max_{1\leqslant j\leqslant n}\|\boldsymbol{a}_j\|_1=\|\boldsymbol{a}_k\|_1$$

取 $\boldsymbol{e}_k=(0,\cdots,0,1,0,\cdots,0)^{\mathrm{T}}$，其第 k 个元素为 1，其余元素为 0，于是 $\|\boldsymbol{e}_k\|_1=1$，且

$$\|\boldsymbol{A}\boldsymbol{e}_k\|_1=\|(\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_n)(0,\cdots,0,1,0,\cdots,0)^{\mathrm{T}}\|_1=\|\boldsymbol{a}_k\|_1=M$$

即当 $\boldsymbol{x}=\boldsymbol{e}_k$ 时，函数 $\|\boldsymbol{A}\boldsymbol{x}\|_1$ 的值等于其上界 M。

所以 M 是函数 $\|\boldsymbol{Ax}\|_1$ 在约束条件 $\|\boldsymbol{x}\|_1=1$ 下的最大值。式(1-8)成立。

例 1-8 求下面矩阵 $\boldsymbol{A}$ 的 1-范数，∞-范数，2-范数：

$$\boldsymbol{A}=\begin{pmatrix}8 & -2\\ 1 & 4\end{pmatrix}$$

解 $\|\boldsymbol{A}\|_1=\max\{|8|+1,|-2|+|4|\}=9$，$\|\boldsymbol{A}\|_\infty=\max\{|8|+|-2|,1+|4|\}=10$，由 $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}=\begin{pmatrix}65 & -12\\ -12 & 20\end{pmatrix}$ 求得其特征值为 68 和 17，故 $\|\boldsymbol{A}\|_2=\sqrt{68}\approx 8.25$。

定理 1-5 设 $\boldsymbol{A}$ 为 n 阶矩阵，则对任意矩阵范数 $\|\cdot\|$ 有 $\rho(\boldsymbol{A})\leqslant\|\boldsymbol{A}\|$。

证 设 λ 为 $\boldsymbol{A}$ 的特征值，$\boldsymbol{x}$ 是对应于 λ 的特征向量，则 $\lambda\boldsymbol{x}=\boldsymbol{Ax}$。上式两端取范数，利用范数的有关性质可得 $|\lambda|\,\|\boldsymbol{x}\|\leqslant\|\boldsymbol{A}\|\,\|\boldsymbol{x}\|$，由于 $\boldsymbol{x}\neq 0$，故 $|\lambda|\leqslant\|\boldsymbol{A}\|$，从而结论成立。

定理 1-6 $\forall\varepsilon>0$，必存在某种矩阵范数 $\|\cdot\|$，使对任意 n 阶矩阵 $\boldsymbol{A}$ 有 $\|\boldsymbol{A}\|\leqslant\rho(\boldsymbol{A})+\varepsilon$。

习题一

1. 在 3 位十进制计算机上分别从左到右及从右到左计算：$34.53+0.035\,24+0.046\,219+0.048\,9+0.032\,7$，并说明哪个结果较为准确。

2. 用秦九韶算法计算 $p(x)=2x^3-3x^2+5x-4$ 在 $x=2$ 处的值。

3. 设 $\boldsymbol{u},\boldsymbol{v},\boldsymbol{x}$ 都是 n 维向量，$\boldsymbol{I}$ 是单位矩阵，试分析用下面两种算法计算 $\boldsymbol{y}$ 的乘法计算量：

(1) $\boldsymbol{y}=(\boldsymbol{I}-\boldsymbol{uu}^{\mathrm{T}})(\boldsymbol{I}-\boldsymbol{vv}^{\mathrm{T}})\boldsymbol{x}$；

(2) $\boldsymbol{y}_1=\boldsymbol{x}-(\boldsymbol{v}^{\mathrm{T}}\boldsymbol{x})\boldsymbol{v}$，$\boldsymbol{y}=\boldsymbol{y}_1-(\boldsymbol{u}^{\mathrm{T}}\boldsymbol{y}_1)\boldsymbol{u}$。

4. 用最小刻度为 mm 的卷尺量得桌面的长为 120 cm，宽 60 cm，试估计桌面的面积会有多大的误差限和相对误差限。

5. 设 n 个近似值 $x_i(i=1,2,\cdots,n)$ 的误差限都是 δ，试求它们之和的误差限。

6. 试确定下列近似值的误差限和有效数字位数。

(1) $x^*=1/6$，x^* 的近似值 $x=0.166$。　(2) 圆周率 π 的近似值 355/113。

(3) $x^*=\mathrm{e}/100$，x^* 的近似值 $x=0.027\,18$。

7. 用四舍五入法求 π 的近似值，使其相对误差限 $\eta\leqslant 0.15\%$。

8. 设近似值 $0.008\,215\,7$ 有 4 位有效数字，求其误差限和相对误差限。

9. 试改变下列表达式，使其计算结果比较精确。

(1) $\dfrac{1}{x+2}-\dfrac{1}{x}$，　$x\gg 1$　　(2) $\dfrac{1}{1+2x}-\dfrac{1-x}{1+x}$，　$|x|\ll 1$

(3) $\sqrt{x+\dfrac{1}{x}}-\sqrt{x-\dfrac{1}{x}}$，　$x\gg 1$　　(4) $\lg x_1-\lg x_2$，　$x_1\approx x_2$

10. 用较好的算法计算下列一元二次方程的根(用 4 位有效数字计算)。

(1) $x^2-100x+1=0$　　(2) $x^2+56x+1=0$

11. 若 $y_0=\sqrt{2}\approx 1.41$(3 位有效数字)，按递推关系

$$y_n=10y_{n-1}-1,\qquad (n=1,2,\cdots)$$

计算到 y_{10} 时误差有多大？这个计算过程稳定吗？

12. 计算向量 $\boldsymbol{x}=(-8,3,-5)^{\mathrm{T}}$ 的 l_1-范数，l_2-范数，l_∞-范数。