数值分析

郑继明 朱 伟 刘 勇 方长杰 编

清华大学出版社

数值分析

郑继明 朱 伟 刘 勇 方长杰 编

清华大学出版社
北京

内 容 简 介

本书从实用和简明的角度介绍了数值分析的基本概念和方法，并对误差估计、方法的收敛性和稳定性以及优缺点等作了适当分析. 全书共分 8 章，内容包括：绪论，插值法，曲线拟合与函数逼近，线性方程组的数值解法，数值积分与数值微分，非线性方程与方程组的数值解法，常微分方程初值问题的数值解法，矩阵特征值问题的数值方法. 附录中给出了 MATLAB 简介. 书中配有典型例题、习题和实验题，书后给出了部分习题答案.

本书可作为理工科各专业研究生和高年级本科生的教材或教学参考书，也可供从事科学与工程计算的科技工作者参考.

图书在版编目(CIP)数据

数值分析/郑继明等编. —北京：清华大学出版社，2016
ISBN 978-7-302-45903-3

Ⅰ. ①数…　Ⅱ. ①郑…　Ⅲ. ①数值分析－研究生－教材　Ⅳ. ①O241

中国版本图书馆 CIP 数据核字(2016)第 302414 号

责任编辑：陈　明
封面设计：傅瑞学
责任校对：赵丽敏
责任印制：刘海龙

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 装 者：保定市中画美凯印刷有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：14.25　　**字　　数**：344 千字
版　　次：2016 年 12 月第 1 版　　**印　　次**：2016 年 12 月第1次印刷
印　　数：1～2500
定　　价：29.80 元

产品编号：070498-01

前言

FOREWORD

科学与工程计算是伴随着计算机的出现而迅速发展并获得广泛应用的一门新兴交叉科学.随着科学技术的发展,作为科学计算基础的数值分析越来越显示出它的重要性.在自然科学和工程应用中,已先后产生了计算力学、计算物理等一系列计算性的分支学科.科学计算利用先进的计算能力认识和解决复杂的科学工程问题,是计算机实现其在高科技领域应用的必不可少的纽带和工具.计算方法是科学与工程计算的核心,构造好的计算方法与研制高性能计算机及高效率计算软件同等重要.科学计算、理论和实验方法一并成为科学技术创新的主要方式.

数值分析也称计算方法,是理工科大多数专业学生的一门重要基础课程,主要介绍工程数学问题(数学模型)中数值计算的一些基本概念和方法,基本内容是数值算法的设计与分析.数值分析既有数学的高度抽象性与严密科学性,又有与计算机技术结合密切、应用广泛的特点.

本书是为理工科各专业研究生和高年级本科生编写的教材,内容包括函数逼近、数值代数、数值微积分和微分方程数值解法等.本书从实用和简明的角度,着重讲清数值算法构造的基本思想与原理,并对误差估计,方法的收敛性、稳定性、适用范围以及优缺点等作了适当分析.教材力求通过分析问题求解的基本算法和典型例题,帮助读者提高解决实际问题的能力,在算法实现方面努力将数值分析理论学习与数学软件编程结合.每章配有适量的习题和上机实验题目,书后给出了习题的参考答案,并推荐使用 MATLAB 软件完成所列实验题目,进一步加深对算法的理解.

本书由郑继明编写了第 3、4、5 章和实验题目,朱伟编写了第 7、8 章,刘勇编写了第 1、6 章和附录,方长杰编写了第 2 章和部分习题,郑继明完成了全书的统稿.本书在编写过程中得到了重庆邮电大学理学院部分师生的指导与帮助,也参考了许多相关教材或著作,在此表示衷心的感谢.另外,本书得到了重庆市“三特行动计划”信息与计算科学专业建设项目和重庆邮电大学文峰骨干教师培养项目,以及清华大学出版社的大力支持,特别是陈明编辑为教材的顺利出版付出了辛劳,在此一并表示感谢.

由于编者水平有限,书中难免出现疏漏甚至错误,敬请广大专家、同行和读者批评指正.

编　者

2016 年 9 月

目录

CONTENTS

第1章 绪 论

随着科学技术的快速发展，科学计算的范围扩大到了几乎所有的科学领域，已成为工程设计与科学研究的重要手段. 因此熟练地运用计算机进行科学计算，已成为科技人员的一项基本技能. 本章除了对数值分析的内容与特点作概述外，还介绍了与误差有关的概念和问题.

1.1 数值分析的内容与特点

自然科学、工程技术、社会经济等领域中产生的许多实际问题都可通过建立数学模型来处理. 然而，可用解析方法精确求解的数学问题只是些很特殊的类型，对于许多数学问题来说我们无法得到它的准确解，从而需要进行数值求解，即需要研究获得实际问题解的近似值的数值方法.

数值分析也称计算方法，是科学计算的一门基础课程. 它是计算数学的重要组成部分，主要介绍各种数学模型及其算法，这些数学模型是为了解决各类实际问题，特别是科学与工程计算领域的实际问题而提出的. 它研究利用计算机求解各种数学问题的数值计算方法及其计算机实现. 数值分析的方法和理论主要包括各种数值方法的构造与应用、误差估计、收敛性和稳定性等.

一般地讲，算法是指解决问题的步骤，即由一些基本运算及运算顺序的规定构成的一个完整的解题步骤. 它可用框图、算法语言、数学语言或自然语言描述. 数值分析的任务，就是为各种数学问题的数值解答提供最有效的算法. 而最有效的算法应当适用范围广、运算工作量少、需要存储单元少、逻辑结构简单、便于编写计算机程序、计算结果可靠等. 因此，概括地说，数值分析有以下特点：

(1) 面向计算机，要根据计算机特点提供切实可行的有效算法.

(2) 有可靠的理论分析，实际计算精度高. 算法要具有收敛性和数值稳定性.

(3) 计算复杂性尽可能小，包括好的时间复杂性(计算时间少)和好的空间复杂性(占用储存单元少). 对很多数值问题使用不同算法，其计算复杂性将会大不一样，这也是数值算法要研究的问题，它关系到算法能否在计算机上实现.

(4) 要有数值实验，即任何一个算法除了从理论上要满足上述三点外，还要通过数值实验证明其是行之有效的.

用数值方法来解决工程实际和科学技术中的问题时，首先必须将具体问题抽象为数学

问题，即建立起能描述并等价代替该实际问题的数学模型，例如各种微分方程、积分方程、代数方程等，然后选择合适的计算方法（算法），编制出计算机程序，最后上机调试并进行运算，以得出欲求解的结果来. 基本过程如下：

实际问题 → 数学模型 → 数值方法 → 程序设计 → 上机调试 → 结果分析

其中根据数学模型提出求解的数值算法直到编出程序、上机计算求出结果，这一过程是计算数学的任务，也是数值分析研究的对象和内容. 其核心是为解决某类问题而设计算法，并对算法的收敛性、稳定性和误差进行分析、计算的全过程. 本书主要介绍工程技术中常用的数值方法.

数值分析是一门紧密联系实际问题、数学理论与计算机的课程，是科学计算的主要组成部分. 学习数值分析这门课程，首先要注意掌握方法的基本原理和基本思想，在此基础上注意和计算机结合进行一些数值计算的训练.

现在，数值分析的实践发生了巨大的变化，几乎很少有人再去写那些繁杂的代码和程序，人们更多的是用像 MATLAB 那样的数学软件包. 同时人们所解决问题的规模也越来越大，从这一方向说，要得益于高级的计算机硬件以及强有力的软件. 但即使这样，也还要进一步了解算法背后的基本思想以及怎样使这些算法在合理时间内收敛，其中包括选择适当的初值、各种方法的结合及调整算法的参数等. 因此，我们推荐用 MATLAB 等软件做数值实验，在本书最后也简要介绍了 MATLAB.

最后要说明的是，数值分析中对数值方法的构造和分析是密切相关、不可分割的. 对于给定的数学问题，常常可以提出各种各样的数值方法（算法）. 这里所说的“算法”，不仅是单纯的数学公式，而且是指由基本运算和运算顺序的规定所组成的整个解题方案和步骤. 例如，当计算多项式

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

的值时，若直接计算 $a_i x^i (i=0,1,\cdots,n)$，再逐项相加，共需做

$$1 + 2 + \cdots + (n-1) + n = \frac{n(n+1)}{2}$$

次乘法和 n 次加法. 当 $n=100$ 时，需做 5050 次乘法和 100 次加法. 若用著名的秦九韶（我国宋代数学家）算法，将多项式 $P(x)$ 改写成

$$P(x) = ((\cdots((a_n x + a_{n-1})x + a_{n-2})x + \cdots + a_2)x + a_1)x + a_0$$

来计算时，只要做 n 次乘法和 n 次加法即可. 如当 $n=100$ 时，只要做 100 次乘法和 100 次加法. 两个算法不同，计算量可以节省几十倍，可见算法的优劣直接影响计算的速度和效率.

另外，算法选得不恰当，不仅影响到计算的速度和效率，还会由于计算机计算的近似性和误差的传播、积累而直接影响到计算结果的精度，有时甚至直接影响到计算的成败. 不合适的算法会导致计算误差达到不能容许的地步，而使计算最终失败，这就是算法的数值稳定性问题.

1.2 误差及有效数字

对数学问题进行数值求解，求得的结果一般都会有误差. 这可能是由实际问题的数学模型化产生的，也可能是计算工作者的疏忽造成的. 因此误差分析和估计是计算方法的重要

内容.

1.2.1 误差的来源

在科学计算中误差来源一般有以下 4 个方面：模型误差、观测误差、截断误差和舍入误差.

(1) **模型误差** 在建模过程中，欲将复杂的物理现象抽象、归结为数学模型，往往只得忽略一些次要因素的影响，进而对问题作某些必要的简化. 这样建立起来的数学模型实际上必定只是所研究的复杂客观现象的一种近似的描述，它与真正客观存在的实际问题之间有一定的差别，这种误差称为模型误差.

(2) **观测误差** 在建模和具体运算过程中所用的一些初始数据往往都是通过人们实际观察、测量得来的，由于受到所用测量工具的限制或在数据的获取时受到随机因素的影响，这些数据都只能是近似的，即存在着误差，这种误差称为观测误差.

(3) **截断误差** 在不少数值运算中常遇到超越计算，如微分、积分和无穷级数求和等，它们需用极限或无穷过程来求得. 然而计算机却只能完成有限次算术运算和逻辑运算，因此需将解题过程化为一系列有限的算术运算和逻辑运算. 这样就要对某种无穷过程进行“截断”. 这种用有限过程代替无限过程所引起的误差，称为截断误差或方法误差. 例如，函数 $\sin x$ 和 $\ln(1+x)$ 可分别展开为 x 的幂级数：

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots$$

若取级数的起始若干项的部分和作为 $|x|<1$ 时函数值的近似计算公式，例如取

$$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!}$$

$$\ln(1+x) \approx x - \frac{x^2}{2} + \frac{x^3}{3}$$

则由于它们的第 4 项和以后各项都舍弃了，自然产生了所谓的截断误差.

一般地，函数 $f(x)$ 用泰勒(Taylor)多项式

$$P_n(x) = f(0) + \frac{f'(0)}{1!}x + \frac{f''(0)}{2!}x^2 + \cdots + \frac{f^{(n)}(0)}{n!}x^n$$

近似代替，则截断误差是

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

其中 ξ 在 0 与 x 之间.

例如，用多项式 $x-\frac{x^3}{3!}+\frac{x^5}{5!}$ 计算 $\sin x$ 的截断误差是

$$|R_4(x)| \leqslant \frac{x^7}{7!}$$

(4) **舍入误差** 在数值计算过程中还会用到一些无穷小数，如

$$\pi = 3.141\,592\,65\cdots$$

$$\sqrt{2} = 1.414\,213\,56\cdots$$

$$\frac{1}{3!} = \frac{1}{6} = 0.166\,666\cdots$$

等. 而计算机受机器字长的限制，它所能表示的数只能有一定的有限位数，这时无穷小数和位数很多的数必须舍入成一定的位数. 由此产生的误差称为舍入误差.

例如，地球的表面积可以通过半径为 r 的球体表面积公式

$$A = 4\pi r^2$$

计算，这个公式包括以下几种近似：将地球看成一个球体，这是理想化的形状；半径近似等于 6370km 可由经验测量和前面计算得到.

因此在计算机或计算器上计算时，输入数据的值以及运算结果的值都可能有一定的舍入. 计算结果的精确度与所有这些近似有关.

综上所述，数值计算中除了可以完全避免的过失误差外，还存在难以回避的模型误差、观测误差、截断误差和舍入误差. 本书主要考虑后两种误差.

1.2.2 绝对误差、相对误差和有效数字

定义 1.1 设某一个量的准确值(称为真值)为 x，其近似值为 x^*，称

$$e(x^*) = x - x^* \tag{1.1}$$

为近似值 x^* 的绝对误差，简称误差.

由于真值 x 往往是未知或无法知道的，因此 $e(x^*)$ 的准确值也就无法求出. 但一般可估计出此绝对误差 $e(x^*)$ 的上限，也即如果可以求出一个正数 ε，使

$$|e(x^*)| = |x - x^*| \leqslant \varepsilon \tag{1.2}$$

则称 ε 为近似值 x^* 的绝对误差限，或称为精度. 通常亦记 ε 为 $\varepsilon(x^*)$. 例如，$\pi = 3.141\,592\,65\cdots$，取 $\pi^* = 3.14$，则

$$|\pi - \pi^*| < 0.002$$

即近似值 π^* 的绝对误差限 $\varepsilon = 0.002$.

通常用

$$x = x^* \pm \varepsilon$$

来表示近似值的精度. 正数 ε 越小，表示该近似值 x^* 的精度越高.

在实际问题中，判断一个近似值的精确度大小不仅要观察绝对误差大小，还要考虑该近似值本身的大小. 这就需要引进相对误差的概念.

定义 1.2 设 x^* 为 x 的近似值，称

$$e_r(x^*) = \frac{e(x^*)}{x} = \frac{x - x^*}{x} \quad (x \neq 0) \tag{1.3}$$

为近似值 x^* 的相对误差. 若已知 $|e_r(x^*)|$ 的一个上界，即存在 $\varepsilon_r > 0$，使得

$$|e_r(x^*)| \leqslant \varepsilon_r$$

则称 ε_r 为 x^* 的相对误差限. 通常亦记 ε_r 为 $\varepsilon_r(x^*)$.

例如测量 10m 的长度时产生 1cm 的误差与测量 1m 的长度时产生 1cm 的误差是大有区别的. 虽然两者的绝对误差相同，都是 1cm，但是前一种测量的相对误差为 $\frac{1}{1000}$，而后一

种测量的相对误差则为$\frac{1}{100}$，是前一种的10倍.

由式(1.3)可得

$$e(x^*) = x \cdot e_r(x^*) \tag{1.4}$$

相对误差不仅能表示出绝对误差来，而且在估计近似值运算结果的误差时，它比绝对误差更能反映出误差的特性. 因此在误差分析中，相对误差比绝对误差更为重要.

注 (1) 相对误差没有量纲，而绝对误差有量纲.

(2) 在实际计算中，由于真值x总是无法知道的，因此往往取

$$e_r^*(x^*) = \frac{e(x^*)}{x^*} \tag{1.5}$$

作为相对误差的另一定义.

1.2.3 有效数字

在表示一个近似值时，为了同时反映其准确程度，常常用到“有效数字”的概念. 例如对无穷小数或循环小数，可用四舍五入的办法来取其近似值.

例1.1 我们知道$\pi=3.141\,592\,65\cdots$是一个无理数，按四舍五入考虑π的不同近似值：

取一位小数，$x_1^*=3$，有$|\pi-x_1^*|\leqslant 0.5=\frac{1}{2}$；

取四位小数，$x_2^*=3.1416$，有$|\pi-x_2^*|\leqslant 0.000\,05=\frac{1}{2}\times 10^{-4}$；

取五位小数，$x_3^*=3.141\,59$，有$|\pi-x_3^*|\leqslant 0.000\,005=\frac{1}{2}\times 10^{-5}$.

这种近似值取法的特点是误差限为其末位数的半个单位. 当近似值x^*的绝对误差限是其某一位上的半个单位时，我们就称其“准确”到这一位，且从该位起直到前面第一位非零数字为止的所有数字都称为有效数字.

定义1.3 设x的近似值x^*的规格化形式为

$$x^* = \pm 0.\alpha_1\alpha_2\cdots\alpha_n \times 10^m \tag{1.6}$$

其中$\alpha_1,\alpha_2,\cdots,\alpha_n$都是0～9中的任一数字，且$\alpha_1\neq 0$；$n$是正整数，$m$是整数. 若$x^*$的(绝对)误差限为

$$|e(x^*)| = |x-x^*| \leqslant \frac{1}{2}\times 10^{m-n} \tag{1.7}$$

则称x^*为具有n位有效数字的有效数，或称它精确到10^{m-n}，其中每一位数字$\alpha_1,\alpha_2,\cdots,\alpha_n$都是$x^*$的有效数字.

注 (1) 若式(1.6)中的x^*是x经四舍五入得到的近似值，则x^*具有n位有效数字. 例如，3.1416是π的具有五位有效数字的近似值，它精确到0.0001(即精确到小数点后第$n-m=4$位).

(2) 有效数尾部的零不可随意省去，以免损失精度.

(3) 另一种情况，例如$x=0.1524$，$x^*=0.154$. 这时x^*的误差$e(x)=-0.0016$，其绝对值超过了0.0005(第三位小数的半个单位)，但却没有超过0.005(第二位小数的半个单

位)，即

$$0.0005 < |x - x^*| \leqslant 0.005$$

显然 x^* 虽有三位小数但却只精确到第二位小数，因此它只具有二位有效数字. 对于这个例子，$\alpha_1 = 1, \alpha_2 = 5$ 都是准确数字，而第三位数字 $\alpha_3 = 4$ 就不再是准确数字了，我们称它为存疑数字.

另外，由式(1.7)可知，从有效数字可以算出近似数的绝对误差限；有效数字的位数越多，其绝对误差限也就越小. 不但如此，还可以从有效数字求出其相对误差限.

当用式(1.6)表示的近似值 x^* 具有 n 位有效数字时，显然有

$$|x^*| \geqslant \alpha_1 \times 10^{m-1}$$

故由式(1.7)可知，其相对误差的绝对值

$$|e_r(x^*)| = \left|\frac{e(x^*)}{x^*}\right| \leqslant \frac{\frac{1}{2} \times 10^{m-n}}{\alpha_1 \times 10^{m-1}} = \frac{1}{2\alpha_1} \times 10^{-n+1}$$

故相对误差限为

$$\varepsilon_r \leqslant \frac{1}{2\alpha_1} \times 10^{-n+1} \tag{1.8}$$

式(1.8)说明 x^* 的有效数字位数越多，其相对误差限也越小. 由此可见，有效数字的位数反映了近似值的相对精确度. 事实上，关于相对误差限与有效数字的关系，我们有下面的定理.

定理 1.1 若 x 的近似值 $x^* = \pm 0.\alpha_1\alpha_2\cdots\alpha_n \times 10^m (\alpha_1 \neq 0)$ 有 n 位有效数字，则相对误差限 $\varepsilon_r \leqslant \frac{1}{2\alpha_1} \times 10^{-n+1}$；反之，若 x^* 的相对误差限

$$\varepsilon_r \leqslant \frac{1}{2(\alpha_1 + 1)} \times 10^{-n+1} \tag{1.9}$$

则 x^* 至少有 n 位有效数字.

证 定理的前一部分结论已证，下面证明后一部分结论.

如果 $\varepsilon_r \leqslant \frac{1}{2(\alpha_1+1)} \times 10^{-n+1}$，则由式(1.5)有

$$\begin{aligned} |e(x^*)| &= |x^* e_r^*(x^*)| = |x^*| e_r^*(x^*) \\ &\leqslant |x^*| \varepsilon_r < (\alpha_1 + 1) \times 10^{m-1} \times \frac{1}{2(\alpha_1+1)} \times 10^{-n+1} = \frac{1}{2} \times 10^{m-n} \end{aligned}$$

由式(1.7)知 x^* 至少有 n 位有效数字.

例 1.2 当用 3.1416 来表示 π 的近似值时，它的相对误差限是多少？

解 3.1416 具有五位有效数字，$\alpha_1 = 3$，由式(1.8)有

$$\varepsilon_r = \frac{1}{2\alpha_1} \times 10^{-n+1} = \frac{1}{2 \times 3} \times 10^{-5+1} = \frac{1}{6} \times 10^{-4}$$

例 1.3 为了使 $x = \sqrt{20}$ 的近似值 x^* 的相对误差小于 0.1%，问至少取几位有效数字？

解 因为 $\sqrt{20} = 4.472\,13\cdots$，则近似值 x^* 中 $\alpha_1 = 4$. 由式(1.8)知

$$\frac{1}{2 \times 4} \times 10^{-n+1} \leqslant 0.1\%$$

可解出 $n = 4$. 即只要取 4 位有效数字，此时 $x^* = 4.472$ 就能满足要求.

1.2.4 计算机机器数系与浮点运算

假定我们提供给计算机的数 x 只是有限位小数，则它可以表示成

$$x = \pm 10^J \sum_{k=1}^{t} d_k 10^{-k} \tag{1.10}$$

其中 J 是整数，$d_1, d_2, \cdots, d_t$ 都是 0,1,2,…,9 中的一个数字. 例如

$$312.74 = 10^3(3\times10^{-1}+1\times10^{-2}+2\times10^{-3}+7\times10^{-4}+4\times10^{-5})$$

式(1.10)是通常的数的十进制系统计数法，其中的 10 称为十进制系统的基数. 在计算机中，还采用二进制、八进制和十六进制等系统表示的方法，其基数分别为 2、8 和 16. 在大多数计算机中，实数是以二进制形式表示的，并且在二进制实数系统中进行运算.

对于一般实数 x，将 x 展开成

$$x = \pm(b_{J-1}\times2^{J-1}+\cdots+b_1\times2^1+b_0\times2^0+b_{-1}\times2^{-1}+b_{-2}\times2^{-2}+\cdots+b_{-n}\times2^{-n}+\cdots)_2$$

这样 x 的二进制表示为

$$x = \pm(b_{J-1}\cdots b_1 b_0 . b_{-1}b_{-2}\cdots b_{-n}\cdots)_2$$

其中 $b_j(j=J-1,\cdots,1,0,-1,-2,\cdots,-n,\cdots)$ 是 1 或 0.

例如，$18.25 = 1\times2^4+0\times2^3+0\times2^2+1\times2^1+0\times2^0+0\times2^{-1}+1\times2^{-2} = (10\,010.01)_2$

在计算机中，一个非零数 x 通常被表示为如下规格化的二进制浮点形式：

$$x = \pm 2^J \sum_{k=1}^{t} b_k 2^{-k} = \pm 0.b_1 b_2\cdots b_t \times 2^J \tag{1.11}$$

其中 $b_j(j=2,3,\cdots,t)$ 是 1 或 0，$b_1=1$. 小数部分 $\pm 0.b_1b_2\cdots b_t$ 称为尾数，正整数 t 称为计算机的字长；J 是整数，称为数 x 的阶.

在各种计算机中，有各自规定的字长 t，以及阶 J 的范围：$L\leqslant J\leqslant U$. L 和 U 的大小表明计算机中表示的数的范围大小. 这种形式的数称为机器数. 由于机器数的字长与阶码有限，因此计算机中的数是有限的. 把计算机中的全体机器数组成的集合记为 F 或 $F(2,t,L,U)$，称为计算机机器数系. 机器数系 F 不是连续统，它是一个有限的、离散的、分布不均匀的集合. 机器数有单精度和双精度之分，字长 t 的值规定了机器数的精度. 阶码 J 的值规定了机器数的绝对值范围. 当数据的绝对值不在范围之内时，称为产生溢出.

绝大多数实数输入计算机时，要转换为有限字长的二进制机器数，只能按舍入原则近似表示. 设实数 x 对应的机器数记为 $fl(x)$，其末位数字 b_t 可能有半位误差，即绝对误差

$$|x-fl(x)|\leqslant\frac{1}{2}\times2^{J-t}$$

相对误差

$$\frac{|x-fl(x)|}{|x|}\leqslant\frac{1}{2}\times2^{-(t-1)}$$

例 1.4 将实数 $x=2.656\,25$ 与 $y=0.1$ 分别表示为 $F(2,8,-19,19)$ 中的机器数.

解 因为 $x=2.656\,25=(0.101\,010\,1)_2\times2^2\in F$，所以

$$fl(x)=x=(0.101\,010\,1)_2\times2^2$$

而 $y=0.1=(0.0\overline{0011})_2=(0.\overline{1100})_2\times 2^{-3}\notin F$，但 $2^{-20}\leqslant|y|\leqslant 2^{20}$.

按舍入法

$$fl(y)=(0.110\,011\,01)_2\times 2^{-3}=0.100\,097\,656$$

按截断法

$$fl(y)=(0.110\,011\,00)_2\times 2^{-3}=0.099\,609\,375$$

下面讨论计算机中浮点数的运算. 在计算机中，规格化的浮点数运算之后仍视为规格化的浮点数存储，且机器数系对四则运算并不封闭. 因此计算机的每次运算都可能产生舍入误差.

设 x 和 y 都是机器数，即 $x,y\in F(10,t,L,U)$，它们的算术运算符合下述规则：

(1) 加减法　先对阶(靠高阶)，后运算，再舍入；

(2) 乘除法　先运算，再舍入.

在运算中，不妨假定计算机具有双精度累加寄存器，即在运算时先保留 $2t$ 位，最后再把第 $t+1$ 位的数进行四舍五入.

例 1.5　设有 4 位十进制数 $x=0.1995\times 10^4$，$y=0.4270\times 10^{-1}$，按舍入法，计算 $x+y$.

解　$fl(x+y)=fl(0.1995\times 10^4+0.000\,004\,270\times 10^4)$　(对阶，靠高阶)

$=0.1995\times 10^4$

结果小数被大数“吃掉”，产生误差.

1.3　数值运算的误差估计

在实际的数值计算中，参与运算的数据往往都是些近似值，带有误差. 这些数据误差在多次运算过程中会进行传播，使计算结果产生误差. 而确定计算结果所能达到的精度显然是十分重要的，但这往往也是件很困难的事. 不过，我们对计算误差作出一定的定量估计还是可以做到的. 这里介绍一种常用的误差估计的公式，它是利用函数的泰勒(Taylor)展开式得到的.

若两个近似值 x_1^* 与 x_2^* 的误差限分别为 $\varepsilon(x_1^*)$ 与 $\varepsilon(x_2^*)$，则它们进行四则运算得到的误差限分别为

$$\varepsilon(x_1^*\pm x_2^*)=\varepsilon(x_1^*)+\varepsilon(x_2^*) \tag{1.12}$$

$$\varepsilon(x_1^*x_2^*)\approx|x_1^*|\varepsilon(x_2^*)+|x_2^*|\varepsilon(x_1^*) \tag{1.13}$$

$$\varepsilon\left(\frac{x_1^*}{x_2^*}\right)\approx\frac{|x_1^*|\varepsilon(x_2^*)+|x_2^*|\varepsilon(x_1^*)}{|x_2^*|^2}\quad(x_2^*\neq 0) \tag{1.14}$$

由式(1.13)和式(1.14)可知，当乘数很大时，乘积的绝对误差限可能很大，应设法避免；当除数 x_2^* 的绝对值很小，接近于零时，商的绝对误差限可能会很大，甚至造成计算机的“溢出”错误，故应设法避免让绝对值太小的数作为除数.

下面考虑误差的传播问题.

当自变量有误差时，计算函数值时也会产生误差. 设一元函数 $f(x)$ 具有二阶导数，自变量 x 的一个近似值 x^*，$f(x)$ 的近似值 $f(x^*)$，用 $f(x)$ 在 x^* 处的泰勒展开式估计误差，可得