Computational Methods

计算方法

郑咸义 编著

华南理工大学出版社

计 算 方 法

郑咸义　编著

华南理工大学出版社

内 容 简 介

“计算方法”也可称“数值分析”。

本书内容包括绪论、解线性方程组的直接法与迭代法、一元方程求根的迭代法、函数近似计算的插值方法、曲线拟合的最小二乘法、微积分数值计算方法和常微分方程初值问题的数值解法等共8章。

本书的特点是：“课文”部分简明，“练习”部分丰富，从而使本书具有可读性、可学性。每章提供的复习题、例题讲解、习题（其中奇数题给出简答，偶数题给出答案）有助于培养学生的解题能力和创造性能力。本书具有清晰的积木式结构，因此教师容易取舍，构成不同层次、不同要求的教学方案。

本书既适用于本科计算机专业和其他理工科高年级学生，也适用于研究生中的工学硕士、工程硕士和申请同等学力硕士学位考试的人员。

图书在版编目（CIP）数据

计算方法/郑咸义编著 .—广州：华南理工大学出版社，2002.9
ISBN 7-5623-1882-4

Ⅰ. 计… Ⅱ. 郑… Ⅲ. 计算方法-高等学校-教材 Ⅳ. O241

中国版本图书馆 CIP 数据核字（2002）第 066403 号

总 发 行：华南理工大学出版社
（广州五山华南理工大学 17 号楼，邮编 510640）
发行部电话：020－87113487 87111048（传真）
E-mail：scut202@scut.edu.cn **http**：//www 2.scut.edu.cn/press
责任编辑：胡 元
印 刷 者：广东农垦总局印刷厂
开 本：787×1092 1/16 **印张**：18 **字数**：470 千
版 次：2002 年 9 月第 1 版第 1 次印刷
印 数：1～4 000 册
定 价：29.00 元

前　言

关于书名　“计算方法”也可称“数值分析”，近年来还被称为“科学计算”。

对　象　本教材的服务对象，一是本科计算机专业和其他理工专业的高年级学生（他们常称为“计算方法”），二是研究生中的工程硕士、工学硕士或申请同等学力硕士学位考试的人员（他们常称为“数值分析”）。无疑，这是一个以“使用”数值算法为自己专业服务的群体。当然，也鼓励他们对数值算法的“创造”作出贡献。不管怎样，通过本课程，我们既可看到计算机如何求解数学问题的机理，也可感受到数值计算如何为数学问题的求解开辟了另一条康庄大道。

特　点　本教材最明显的特点是：“课文”部分力求写得简明，“练习”部分尽量写得丰富。这种写法基于我们对这门课程的教学理念。我们认为，只有老师“讲”，学生不“做”，那是学不到多少东西的，更谈不上培养具有真实能力和创新能力的人才。简明带来可读性，我们用新的眼光对传统的教学内容精心取舍，并力求用现代风格的语言加以演绎；练习部分更是经过精心的综合、加工、链接和再创作的结果，我们希望把它制作成一个培养、训练学生独立思考能力、分析处理问题能力和创造能力的“平台”。我们相信，这种写法对学生的学习是有帮助的，对教师的教学是方便的。

内　容　如果用一句话来概括本教材的内容，那就是“《计算方法》：六大问题，七类方法，写成八章书”（详见目录）。大致说来，第1章是全书的引论和基本概念，然后纵横展开。从横向看，除第2、3章包含同一类问题外，后续每章各为一类问题；从纵向看，每章的第1节是问题的提法及相关概念，接着第2节、第3节……则由浅入深地介绍各种计算方法及其相应的理论结果。由此可见，这是一本结构清晰，便于“截断、舍入”的教材。教师很容易用“纵横增减法”来组织针对不同层次、不同学时的教学方案，而学生则可以按自己的需要和实力安排可行的学习计划。

本教材是在国家工科基础课程数学教学基地（华南理工大学）的资助下完成的。同时也得到了华南理工大学研究生院、教务处、网络教育学院和理学院应用数学系等多方面的支持。

致 谢

本教材承我校计算机学院副院长、博士生导师韩国强教授主审，作者对韩教授提出的许多宝贵和精辟的改进、指导意见表示衷心的谢意。作者多年教学采用和参考清华等多所著名大学的多本教材，受益匪浅，本教材也无不受这些优秀教材的启发和指引，故谨向李庆扬、关治、陆金甫、王能超、易大义等多位教授致崇高敬意。作为同一个项目组的同事陶志穗、雷秀仁、陆子强老师，他们与作者共同研究的心得在本教材中得到充分的体现。作者感谢美国西弗吉尼亚大学（West Virginia University）教授方伟夫（Weifu Fang）博士为本书提供了有关参考资料，并传递了当前美国的一些教学信息。

结 语

作为一本希望做些新尝试的教材，尽管编者作了认真的努力，但疏漏和不妥之处一定还不少，恳请使用本教材的老师、同学以及其他专家批评指正。

郑咸义
2002年暑假于华南理工大学应用数学系
mazhengx@scut.edu.cn

目　录

1 计算方法的基本概念

1.1 《计算方法》的内容、意义和学习

“计算方法”是研究数学问题的数值计算方法(或称近似计算方法)及其相关理论的课程。“计算方法”这个名称更完整的叫法应该是“数学数值计算方法”，但由于数学的一般性，通常就简称为“数值计算方法”或“数值方法”或“计算方法”。另外，“计算方法”课程与另一门称为“数值分析”的课程，可以说是大同小异。这类课程不论叫“计算方法”还是“数值分析”，其主要差异在于内容的多、少、深、浅，是突出方法，淡化理论，还是既突出方法，也强调理论，特别是课程的教学对象定位在哪个层次、哪些群体。

根据“计算方法”课程的任务，“计算方法”课程的基本框架是：

①给出一类类典型数学问题的数值求解提法(包括其应用背景和理论背景)；

②构造求解该类问题数值解(而不是解析解)的各种数值计算方法，并作其误差分析；

③进一步把计算方法设计成计算机算法，考察其数值稳定性以及上机计算。

只有充分理解每类数值问题的提法及其有关背景，才能理解这类数值问题要解决的是什么问题，可用哪些数值计算方法。只有熟练掌握解决不同类型问题的不同数值计算方法及其相关理论结果，才有可能最终有效地解决所提出的问题。也只有在上述基础上，才能把数值计算方法应用到具体的科学/工程计算中去，解决实际的问题。至于把数值计算方法设计成计算机算法，对于常用的一些方法，并不需要每一个都去研究其算法设计，因为已有大量的算法汇编的专著和现成的数值软件可供使用。目前，已经相当流行的数学/数值软件包有 Mathematica，Matlab，Maple 等，但这并不意味着有了现成的软件包，就不用学习“计算方法”这门课了。事实上，如果没有为具体问题选择和使用数值计算方法的能力和知识；如果不会充分利用商品化的数值软件工具，或必要时自己也能设计一定的算法和编写相应的程序，那么，你所能解决的问题在范围、深度和效率方面，将是极其有限的。

有两点需要说明的是：

①既然“计算方法”是解决各类数学问题的数值求解方法，那么，各类数学问题的应用也就是“计算方法”的应用。虽然不少数学问题的解析解往往不一定能求得，但通过“计算方法”，其数值解一般都能求到。

②现代计算方法的特点是以现代计算机系统作为处理平台。利用数值计算方法和计算机来解决科学/工程中的问题，通常称为“科学/工程计算”或简称“科学计算”。由于科学计算本身的迅速发展及其不断取得成效，使得“科学计算”与传统的“理论研究”和“实验研究”并列成为当今科学发展的三大研究方法。而科学计算与具体学科的交叉发展，又形成了诸如计算力学、计算物理、计算化学、计算生物等等新的计算工程学科。这些学科

无疑给理工科学生和理工科专业工作者提供了诱人的发展空间。

1.2 误差的基本概念

1. 误差

数值计算实质上是数学的一种近似处理，所以最基础的概念就是误差的概念。所谓误差，就是一个量的准确值与其近似值之差。

在这里，“误差”用数值来表示。“数值”这个术语在数值计算中有时叫“数”，有时也叫“值”，指的就是我们熟知的实数值(或复数值)。这与计算机科学中使用的“型”与“值”的概念并不相同。

在数值计算中，针对不同的对象，引进不同的误差概念。其中研究和使用的最基本的概念有刻画近似值近似程度的“绝对误差”、“相对误差”和“有效数字”，还有描述数值计算方法构造时产生的“截断误差”和进行数值计算方法实际计算时存在的“舍入误差”，不妨称为“误差五大基本概念”。

2. 绝对误差/相对误差

一个准确值(也称精确值)可能由多个不同的近似值表示，而一个近似值总是对应某个准确值而言。近似值的近似程度需要有一个表述。

定义 1.2.1 设 x 为准确值，x^* 为 x 的一个近似值，定义

$$e(x^*)=x-x^* \tag{1.2.1}$$

为近似值 x^* 的**绝对误差**或简称**误差**。而在 $x\neq 0$(或 $x^*\neq 0$)时，再定义

$$e_r(x^*)=\frac{x-x^*}{x} \quad 或 \quad e_r(x^*)=\frac{x-x^*}{x^*} \tag{1.2.2}$$

为近似值 x^* 的**相对误差**。

在不引起混淆时，$e(x^*)$和 $e_r(x^*)$也可简写为 e 和 e_r。

从定义可见，相对误差是绝对误差在准确值或近似值中所占的比率，因此，用相对误差比用绝对误差在一定情况下能更好地反映近似值的准确程度。

例 1.2.1 设近似值 5 000 的绝对误差为 1，近似值 5 的绝对误差为 0.1，说哪个近似值比较准确呢？当然，如果不比较，前者误差大，后者误差小；但如果比较，虽然前者的绝对误差是后者的绝对误差的 10 倍，但考虑到所讨论的值本身的大小，在 5000 之中有误差 1 比在 5 中有误差 0.1，显然前者准确程度更高。用相对误差表示，可得 5000 与 5 的相对误差分别为

$$e_r(5000)=\frac{1}{5000}=0.0002=0.02\%$$

$$e_r(5)=\frac{0.1}{5}=0.02=2\%$$

在实际情况中，e 与 e_r 有时可以具体计算出来，有时无法具体计算出来。在后一种情况下，我们就不说去计算误差，而只是说去作误差估计，即估计出绝对误差或相对误差的一个范围，也即所谓**界**或**限**。

定义 1.2.2 设 x 为准确值，x^* 为 x 的一个近似值，如果能对 x^* 的绝对误差作出

估计

$$|e| = |x - x^*| \leqslant \varepsilon \tag{1.2.3}$$

则称 ε 为 x^* 的**绝对误差界**，简称**误差界**。如果能对 x^* 的相对误差作出估计

$$|e_r| = \frac{|x - x^*|}{|x^*|} \leqslant \varepsilon_r \tag{1.2.4}$$

或直接取

$$\varepsilon_r = \frac{\varepsilon}{|x^*|}$$

则称 ε_r 为 x^* 的**相对误差界**。

满足不等式(1.2.3)和(1.2.4)的 ε 和 ε_r 可以很多。按定义的实质，误差估计的任务就是提供式中尽可能小的 ε、ε_r。引入误差界 ε 之后，可以把无法明明白白写出来的准确值 x 表示为

$$x^* - \varepsilon \leqslant x \leqslant x^* + \varepsilon \quad 或 \quad x = x^* \pm \varepsilon$$

这样，准确值 x 虽不具体，但有“踏踏实实”的感觉。

通过误差界，还引进了“有效数字”的概念。

3. 有效数字

我们已经熟悉用“四舍五入”的原则取近似值的方法。例如，最熟悉的 $\pi = 3.1415926\cdots$，若取 $x^* = 3.14$，则有

$$|e| = |\pi - x^*| = |0.0015926\cdots| \leqslant 0.002 \leqslant 0.005 = \frac{1}{2} \times 10^{-2}$$

若取 $x^* = 3.1416$，则有

$$|e| = |\pi - x^*| = |-0.0000073\cdots| \leqslant 0.000008 \leqslant 0.00005 = \frac{1}{2} \times 10^{-4}$$

这就是说，用“四舍五入”原则取得的近似值的特点是，其绝对误差界不超过(即$\leqslant$)近似值被保留的最末位数的半个单位。因此，一般地，若近似值 x^* 的绝对误差界不超过自身某一位数的半个单位，便称该位数为近似值 x^* 的所谓“准确数字”；同时，若该位数到 x^* 的第一位非零数字共有 n 位，便称这 n 位数字为**有效数字**，或称近似值 x^* 有 n 位**有效数字**。严格的定义如下：

定义 1.2.3 设 x 的一个近似值 x^*，把 x^* 写成规范化的科学记数形式(也称规范化的浮点形式)

$$x^* = \pm 0.a_1a_2a_3\cdots a_n\cdots \times 10^k \quad (有限或无限)$$

其中 k 为整数，a_1，a_2，…，a_n，… 是 0，1，…，9 中的一个数字，且 $a_1 \neq 0$，如果有

$$|x - x^*| \leqslant \frac{1}{2} \times 10^{k-n} \tag{1.2.5}$$

则称 x^* 为 x 的具有 n 位**有效数字**的近似值，简称 x^* 有 n 位**有效数字**。

例 1.2.2 设 $x = 10^3 \times 0.7136\cdots$

①若取 $x_1^* = 10^3 \times 0.714$ 作为 x 的近似值，则由于

$$|x - x_1^*| = 10^3 \times 0.0003\cdots \leqslant 10^3 \times 0.0005 = \frac{1}{2} \times 10^{3-3}$$

故 x_1^* 作为 x 的近似值有 3 位有效数字。

②若取 $x_2^* = 10^3 \times 0.715$，则由于

$$|x-x_2^*|=10^3\times0.0013\cdots\leqslant10^3\times0.005=\frac{1}{2}\times10^{3-2}$$

故 x_2^* 作为 x 的近似值有 2 位有效数字。

③若取 $x_3^*=10^3\times0.7132$，则由于

$$|x-x_3^*|=10^3\times0.0004\cdots\leqslant10^3\times0.0005=\frac{1}{2}\times10^{3-3}$$

故 x_3^* 作为 x 的近似值有 3 位有效数字。要注意的是，这里是指用 $x_3^*=10^3\times0.7132$ 作 x 的近似值有 3 位有效数字；如果取 $x_4^*=10^3\times0.7131$ 作 x 的近似值，则由于

$$|x-x_4^*|=10^3\times0.0005\cdots\leqslant10^3\times0.005=\frac{1}{2}\times10^{3-2}$$

可知 x_4^* 作为 x 的近似值只有 2 位有效数字。

关于有效数字还有以下几点说明：

①由准确值用“四舍五入”法则取得的近似值必为有效数字；但近似值除了用“四舍五入”取得外，也可由其他计算方法取得，如用分数$\frac{22}{7}$作为 π 的近似值，它具有 3 位有效数字($\frac{22}{7}=3.142857\cdots$)。

②将任何数乘以 $10^k(k=0,\ \pm1,\ \pm2,\ \cdots)$，相当于移动该数的小数点，并不影响它的有效数字的位数，所以说，有效数字的位数与小数点无关。

③如无特别声明，科技文本的数字、实验报告的数字以及媒体公布的统计数字，对于写者或读者，均认同是有效数字；至于准确值则通常被认为具有无穷位有效数字。

4. 截断误差/舍入误差

事实上，在处理科学/工程计算问题的整个过程中，例如，建立数学模型、确定模型中各种参数、构造数值计算方法、设计计算机算法，直至编写程序上机计算或直接使用现成软件包计算等步骤，都可能出现误差。在数值计算中，主要关注下列两类误差。

第一类是所谓**截断误差**(也称**方法误差**)，它是指在构造数值计算方法时，用有限过程代替无限过程或用容易计算的方法代替不容易计算的方法，其计算结果所存在的误差。例如，研究 $\sin x$ 的计算，用无穷级数表示

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

当$|x|$较小时取前三项作为 $\sin x$ 的计算公式，这样计算的结果(近似值)就与无穷级数本来的值产生了误差，这种误差就是截断误差，或称它为计算公式的截断误差或余项。又如，用差商$\frac{\Delta y}{\Delta x}$作为导数$\frac{\mathrm{d}y}{\mathrm{d}x}$的近似值，这时也产生了截断误差。由于截断误差来自方法处理过程，故也称方法误差。

第二类是所谓**舍入误差**(或称**计算误差**)。计算机所能表示的数字位数总是有限的，因此，对原始数据、中间计算结果和最后计算结果，都只能取有限位表示，这就要求进行“舍入”(即使手算，也只能取有限位数进行计算)，这时所产生的误差就是舍入误差。舍入误差存在于计算过程中，故也称计算误差。

此外，有时也考虑所谓**初始数据误差**或称**输入数据误差**，它可能是物理数据测量不准确带来的，也可能是初始数据的值只能取近似值进行实际计算时引起的(如公式中有

$\sqrt{2}$，取 1.41 进行实际计算）。为了简明，我们把这些误差归入舍入误差范围处理。

总之，在构造数值计算方法时，需要研究截断误差；就算法实现而言，需要注意舍入误差和初始数据误差。

1.3 误差分析初步、Taylor 公式与大 O 记号

Taylor 公式在数值计算方法的构造和误差分析中起着极其重要的作用。我们先来复习一下 Taylor 公式并补充数学中的大 O 记号。针对本课程的实际需要，有关 Taylor 公式的假设条件总是假设得略为宽松。

1. Taylor(泰勒)公式

设 f 在含有点 x_0 的某个开区间(a, b)内具有 $n+1$ 阶导数，则 $\forall x\in(a, b)$有

$$f(x)=f(x_0)+f'(x_0)(x-x_0)+\frac{f''(x_0)}{2!}(x-x_0)^2+\cdots+\frac{f^{(n)}(x_0)}{n!}(x-x_0)^n+\frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)^{n+1}$$

其中 ξ 在 x_0 与 x 之间，前 $n+1$ 项称为 n 次 Taylor 多项式，最后一项称为 n 次 Taylor 多项式的余项(即截断误差)。为了方便作误差估计，有时还假定 f 的 $n+1$ 阶导数连续。

Taylor 公式可以有多种表达形式，原理是一样的，要熟练掌握。

2. 多元函数 Taylor 公式

以二元函数 $f(x, y)$为例。设 $f\in C^n(D)$，$D=\{(x, y)\mid |x-x_0|<a, |y-y_0|<b\}$，且 f 在 D 上存在 $n+1$ 阶偏导数，$(x_0, y_0)\in D$，则 $\forall (x, y)\in D$ 有

$$f(x, y)=f(x_0, y_0)+\sum_{k=1}^{n}\frac{1}{k!}\left[(x-x_0)\frac{\partial}{\partial x}+(y-y_0)\frac{\partial}{\partial y}\right]^k f(x_0, y_0)+R_n(x, y)$$

其中 $R_n(x, y)=\frac{1}{(n+1)!}\left[(x-x_0)\frac{\partial}{\partial x}+(y-y_0)\frac{\partial}{\partial y}\right]^{n+1} f(\xi, \eta)$，$\xi$ 在 x_0 与 x 之间，η 在 y_0 与 y 之间。

多元 Taylor 公式也有不同的表达形式。

3. 大 O 记号

大 O 记号是数学、计算机科学和技术文献中广泛使用的符号，是为表示近似值而允许我们用 = 号代替 ≈ 号的方便符号。

设变量 X，Y(其中 $X\neq 0$)，如果在变化过程的某一时刻以后，有

$$\left|\frac{Y}{X}\right|\leqslant M \quad 或 \quad |Y|\leqslant M|X| \quad (M 为大于 0 的常数)$$

便记成

$$Y=O(X)$$

这就是说，记号 $O(X)$表示这样一个量，我们并不明显地知道它，也不必说出定义中的 M 是多少，它的出现即意味着，当 X 变化到某一时刻后，$|O(X)|$ 总是不会超过 $M|X|$。

例如，$1^2+2^2+\cdots+n^2=\frac{1}{3}n\left(n+\frac{1}{2}\right)(n+1)=\frac{1}{3}n^3+\frac{1}{2}n^2+\frac{1}{6}n$

$$(可以记为)=\frac{1}{3}n^3+\frac{1}{2}n^2+O(n)$$

$$(或)=\frac{1}{3}n^3+O(n^2)$$

$$(或)=O(n^3)$$

又例如，利用 O 大记号，Taylor 公式也可表示为

$$f(x)=f(x_0)+f'(x_0)(x-x_0)+\cdots+\frac{f^{(n)}(x_0)}{n!}(x-x_0)^n+O((x-x_0)^{n+1})$$

必须注意，大 O 记号具有单向相等性，可以写为 $\frac{1}{2}n^2+n=O(n^2)$，但决不能写成 $O(n^2)=\frac{1}{2}n^2+n$；含大 O 记号的等式的右端只是左端的"粗略化"，右端不能提供比左端更多的信息。大 O 记号的一些简单运算如下：

① $O(X)\pm O(X)=O(X)$

② $mO(X)=O(X)$，m 为不等于 0 的常数

③ $O(X)\cdot O(X)=O(X^2)$

④ $O(X(X))=O(X)$

4. 函数计算的误差估计

计算函数 $f(x)$在 x 处的值时，往往用 x 的近似值x^*计算出近似值 $f(x^*)$，这就要估计误差

$$e(f(x^*))=f(x)-f(x^*)$$

设 f 在x^*的邻域上 2 阶连续可微，利用 2 阶 Taylor 公式

$$f(x)=f(x^*)+f'(x^*)(x-x^*)+\frac{f''(\xi)}{2!}(x-x^*)^2$$

把 $f(x^*)$移到等号左边，并两边取绝对值，可得近似值 $f(x^*)$的误差估计

$$|f(x)-f(x^*)|\leqslant|f'(x^*)||x-x^*|+\frac{|f''(\xi)|}{2}|x-x^*|^2$$

或在 $f'(x^*)\neq0$，$|f''(\xi)|$与$|f'(x^*)|$相比不会太大时，略去 2 阶项，可取误差界的一个估计

$$\varepsilon(f(x^*))\approx|f'(x^*)||x-x^*| \quad 或 \quad \varepsilon(f(x^*))\approx|f'(x^*)|\varepsilon(x^*)$$

同理，对二元函数 $f(x,\ y)$计算，利用

$$f(x,\ y)=f(x^*,\ y^*)+\frac{\partial f(x^*,\ y^*)}{\partial x}(x-x^*)+\frac{\partial f(x^*,\ y^*)}{\partial y}(y-y^*)+\cdots$$

或

$$f(x,\ y)-f(x^*,\ y^*)\approx\frac{\partial f(x^*,\ y^*)}{\partial x}(x-x^*)+\frac{\partial f(x^*,\ y^*)}{\partial y}(y-y^*)$$

可取近似值 $f(x^*,\ y^*)$的误差界的一个估计

$$\varepsilon(f(x^*-y^*))\approx\left|\frac{\partial f(x^*,\ y^*)}{\partial x}\right||x-x^*|+\left|\frac{\partial f(x^*,\ y^*)}{\partial y}\right||y-y^*|$$

或

$$\varepsilon(f(x^*,\ y^*))\approx\left|\frac{\partial f(x^*,\ y^*)}{\partial x}\right|\varepsilon(x^*)+\left|\frac{\partial f(x^*,\ y^*)}{\partial y}\right|\varepsilon(y^*)$$

由上述误差估计式可进一步得到对应的相对误差估计式。

5. 算术运算的误差估计

两个近似数的算术运算的误差估计，可直接按定义推导，也可看做二元函数 $f(x, y)=x+y$，$x-y$，$x\cdot y$，$\frac{x}{y}$ 的计算(除法时 $y\neq 0$)，按上述公式推导。如按后者可导出估计式

$$\varepsilon(x^*\pm y^*)\approx\varepsilon(x^*)+\varepsilon(y^*)\quad\text{(按定义推导又得}\approx\text{号为}\leqslant\text{号)}$$

$$\varepsilon(x^*\cdot y^*)\approx|x^*|\varepsilon(y^*)+|y^*|\varepsilon(x^*)$$

$$\varepsilon\left(\frac{x^*}{y^*}\right)\approx\frac{|x^*|\varepsilon(y^*)+|y^*|\varepsilon(x^*)}{|y^*|^2}\quad(y^*\neq 0)$$

6. 用差商近似代替导数的误差估计

用差商近似代替导数的误差估计可利用 Taylor 公式

$$f(x+h)=f(x)+f'(x)h+\frac{f''(\xi)}{2!}h^2\quad(\xi\text{ 在 }x\text{ 与 }x+h\text{ 之间})$$

将它改写为

$$f'(x)=\frac{f(x+h)-f(x)}{h}-\frac{f''(\xi)}{2}h=\frac{f(x+h)-f(x)}{h}+O(h)$$

可得导数 $f'(x)$ 用差商 $\frac{f(x+h)-f(x)}{h}$(称向前差商)近似代替，截断误差(余项)为

$$R=\frac{M}{2}h\qquad M=\max|f''(x)|$$

或记为 $R=O(h)$，并称这种近似有 1 阶精度。

类似上述推导，也可得导数 $f'(x)$ 用差商 $\frac{f(x)-f(x-h)}{h}$(称向后差商)近似代替，截断误差 $R=O(h)$，也有 1 阶精度。

又利用 Taylor 公式：

$$f(x+h)=f(x)+f'(x)h+\frac{f''(x)}{2!}h^2+\frac{f'''(\xi)}{3!}h^3$$

$$f(x-h)=f(x)-f'(x)h+\frac{f''(x)}{2!}h^2-\frac{f'''(\eta)}{3!}h^3$$

两式相减有

$$f(x+h)-f(x-h)=2f'(x)h+\frac{1}{3!}[f'''(\xi)+f'''(\eta)]h^3$$

再改写成

$$f'(x)=\frac{f(x+h)-f(x-h)}{2h}-\frac{1}{12}[f'''(\xi)+f'''(\eta)]h^2$$

$$=\frac{f(x+h)-f(x-h)}{2h}+O(h^2)$$

可得导数 $f'(x)$ 用差商 $\frac{f(x+h)-f(x-h)}{2h}$(称中心差商)近似代替，截断误差为 $O(h^2)$，称有 2 阶精度。

再利用 Taylor 公式：

$$f(x+h)=f(x)+f'(x)h+\frac{f''(x)}{2!}h^2+\frac{f'''(x)}{3!}h^3+\frac{f^{(4)}(\xi)}{4!}h^4$$

$$f(x-h)=f(x)-f'(x)h+\frac{f''(x)}{2!}h^2-\frac{f'''(x)}{3!}h^3+\frac{f^{(4)}(\eta)}{4!}h^4$$

两式相加有

$$f(x+h)+f(x-h)=2f(x)+f''(x)h^2+\frac{1}{4!}[f^{(4)}(\xi)+f^{(4)}(\eta)]h^4$$

再改写成

$$f''(x)=\frac{f(x+h)-2f(x)+f(x-h)}{h^2}-\frac{1}{4!}[f^{(4)}(\xi)+f^{(4)}(\eta)]h^2$$

$$=\frac{f(x+h)-2f(x)+f(x-h)}{h^2}+O(h^2)$$

可得 2 阶导数 $f''(x)$用 2 阶差商$\frac{f(x+h)-2f(x)+f(x-h)}{h^2}$近似代替，截断误差为 $O(h^2)$，称有 2 阶精度。

1.4* 计算机中数的表示和舍入误差

使用计算机(包括计算器)进行数值计算时，需要把原始数据输入计算机，令计算机进行运算以及从计算机读取计算结果。因此，需要了解计算机如何表示(存储)数，以及计算机作运算时是否存在误差。

通常，计算机表示数总是有限位的，并且采用二进制实数系统(或非本质地变形为十六进制系统)。更具体地说，还分为定点表示和浮点表示，相应地有定点数和浮点数之称；浮点表示中又有单精度和双精度之分。

1. 定点表示

定点表示即约定小数点固定在数的某个位置(如中间或“两头”)，一般表示为

$$x=\pm a_m a_{m-1}\cdots a_1 a_0 . a_{-1}a_{-2}\cdots a_{-n}$$

即约定小数点前有 $m+1$ 位整数，小数点后有 n 位小数。计算机中所表示的全体定点数称为定点数系，常记为 D。若 D 采用β 进制(如 $\beta=2$，10，16，…)，则上述 x 可表示为

$$x=\pm a_m\beta^m+a_{m-1}\beta^{m-1}+\cdots+a_1\beta+a_0\beta^0+a_{-1}\beta^{-1}+a_{-2}\beta^{-2}+\cdots+a_{-n}\beta^{-n}$$

β 称为基底，每个 $a_i\in\{0,1,\cdots,\beta-1\}$，$m$ 和 n 为自然数。

例如，考虑这样的定点数系，其中 $\beta=10$(即十进制)，$m=3$(即有 4 位整数)，$n=5$(即有 5 位小数)，则下列 3 个十进制数

$$2.53 \qquad 0.045 \qquad -23.0001$$

在该数系中将分别表示为

$$0002.53000 \qquad 0000.04500 \qquad -0023.00010$$

而且可以看出，对属于该数系的任何 x，有

$$|x|_{\max}=9999.99999 \qquad |x|_{\min}=0000.00001 \quad (x\neq 0)$$

两数间的最小距离为 0.00001。

由此看来，用定点方法可能表示的数是相当有限的。计算机中一般仅用定点数表示整数，即取 $n=0$，常用于作精确运算或控制循环次数等。

2. 浮点表示

浮点表示的数(即浮点数)的一般形式为

$$x=\pm(a_1\beta^{-1}+a_2\beta^{-2}+\cdots+a_n\beta^{-n})\times\beta^m$$

其中，与定点表示相仿，β 称为基底，通常取 $\beta=2$，10，16，…；每个 $a_i\in\{0,1,\cdots,\beta-1\}$；$n$ 为自然数，是计算机的字长；m 为阶码，有随不同计算机系统而定的下限 L 和上限 U，即 $L\leqslant m\leqslant U$；又$(a_1\beta^{-1}+a_2\beta^{-2}+\cdots+a_n\beta^{-n})$称为浮点数 x 的尾数。特别地，若限定 $a_1\neq0$，则称 x 为**规格化浮点数**。显然，$x=0$ 不可能用规格化形式表示；用浮点数表示的每一个数都是有理数。计算机中所表示的全体浮点数称为浮点数系，常记为 F。F 是实数系 $\mathbf{R}$ 的一个离散子集，而 $\mathbf{R}$ 则是一个"连续统"。

例如，对十进制浮点数 $x=0.005678\times10^6$，其基底 $\beta=10$，阶码 $m=6$，小数点后面部分(005678)为尾数，字长 6 位；这个数的规格化浮点形式为 $x=0.567800\times10^4$。

由浮点数的一般形式可推出具体的 F 中的数的最大/最小绝对值。如对 $\beta=10$，$n=7$(字长 7 位)，阶码 $-4\leqslant m\leqslant4$，则

$$|x|_{\max}=(9\times10^{-1}+9\times10^{-2}+\cdots+9\times10^{-7})\times10^4=9999.999$$

$$|x|_{\min}=1\times10^{-7}\times10^{-4}=10^{-11}\quad(x\neq0)$$

在计算机表示或运算过程中，出现绝对值大于 $|x|_{\max}$ 的数便产生"溢出"；出现绝对值小于 $|x|_{\min}$ 的数机器便作零处理，称"机器零"。

显然，进入计算机的每一个实数 x，只能在数系 F 中寻找最靠近 x 的浮点数来表示。寻找的原则通常就是"四舍五入"原则(也有的用"只舍不入"原则)。

例如，设使用的计算机系统为 $\beta=10$(即十进制系统，以十进制系统为例，是为了说明方便，对二进制系统原理也是一样的)，尾数为 n。这时，若输入的数 x 为

$$x=\pm0.a_1a_2\cdots a_na_{n+1}\cdots\times10^s\quad(0\leqslant a_i\leqslant9,\ a_1\neq0)$$

则机器取 F 中的 x^*：

$$x^*=\begin{cases}\pm0.a_1a_2\cdots a_n\times10^s & (0\leqslant a_{n+1}\leqslant4)\\ \pm0.a_1a_2\cdots(a_n+10^{-n})\times10^s & (a_{n+1}\geqslant5)\end{cases}$$

作为 x 的近似值，通常记 $x^*=fl(x)$。这时，$fl(x)$的绝对误差和相对误差分别估计为

$$|x-fl(x)|\leqslant\frac{1}{2}\times10^{s-n}$$

$$\left|\frac{x-fl(x)}{x}\right|\leqslant\frac{\frac{1}{2}\times10^{s-n}}{0.1\times10^s}=\frac{1}{2}\times10^{-n+1}$$

其中令 $\delta=\dfrac{x-fl(x)}{x}$，即$|\delta|\leqslant\dfrac{1}{2}\times10^{-n+1}$称为计算机的精度或机器中的浮点数(简称机器数)的精度，它由尾数的长度 n 决定。

3. 单精度/双精度

机器数还有单精度和双精度之分。常用的一种规定是单精度 32 位，双精度 64 位，它们是数符(+或-)、阶码(s)、尾数(n)三者所占二进制的总长度。在单精度，尾数 n