Northeastern University

数值分析

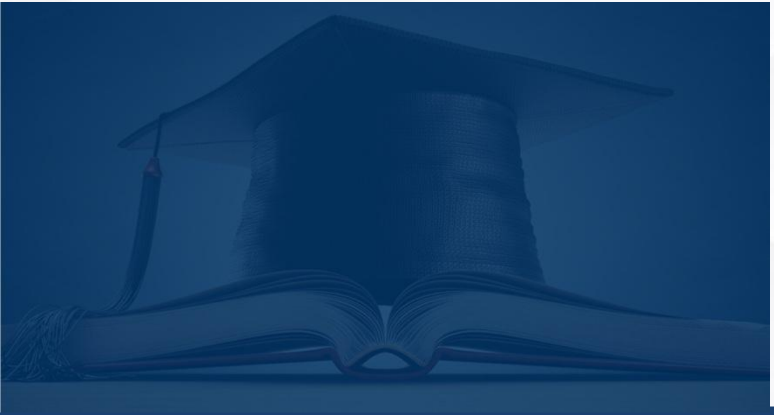
理学院　　数学系

计算数学教研室

2017年12月13日

目 录
Contents

01 绪　论



绪　论

数值分析研究的对象和内容

20世纪数学最大的变化是数学应用，美国科学工程和公共事务政策委员会报告《美国的现在和未来》（1986年）指出：“今天，在技术科学中最有用的数学领域是数值分析和数学建模”。

数值分析是研究科学计算中各种数学问题求解的数值计算方法.

实例分析

绪　论

天气预报： 天气会受各种因素的影响，稍微一些因素发生改变就会产生很大的变化，所以天气预报其实是一件比较困难的工作，古代人们用占卜或者经验总结等方式来预计天气状况，这是统计学。有了计算机，可以通过数值模拟来预报天气。

具体过程：
1. 根据大气运动列出数学物理方程；
2. 对空间进行网格划分；
3. 通过观测数据给出初值条件，通过数值方法求解这些方程得到网格点处的数值解。这也是为什么主持人总是说大概在...地区大致在...时段，可能有...量级的降水...因为时空是连续的，而网格划分不可能无限密，所得的数值解也存在误差 。

实例分析

绪　论

供水计划和生产调度计划的制定

如何充分地利用这些数据建立数学模型，预测2007年1月份城市的用水量，以制定相应的供水计划和生产调度计划？

表1　2000-2006年1月某城市的总用水量（万吨/日）

年份	2000	2001	2002	2003	2004	2005	2006
用水量	4032．41	4186．025	4296．986	4374．852	4435．234	4505．427	4517．699

如果能建立该城市的日用水量随时间变化的函数关系，则用该函数来进行预测非常方便。但是这一函数关系的解析表达式是没办法求出来的，那么能否根据历史数据求出该函数的近似函数呢？根据未知函数的已有数据信息求出其近似函数的常用方法有插值法和数据拟合。

实例分析

绪　　论

湘江水流量估计的实际意义

水流量是水文特征值的一个重要指标，而水文特征值对于水资源的合理利用，防洪以及抗旱具有指导性的作用，因此湘江水流量估计对于湘江流域的社会经济和人民生活具有重大的影响。现根据实际测量得到湘江某处河宽700m，其横截面不同位置某一时刻的水深如表2所示。若此刻湘江的流速为0.5m/s，试估计湘江此刻的流量。要计算湘江水流量就需要知道其横截面面积，如果知道此处江的水深曲线函数，则其横截面面积为 $\int_a^b h(x)dx$ 但是在实际中是不可能精确得到的，那么怎样求出足够高精度的横截面面积的近似值。

表2 湘江某处横截面不同位置的水深数据　单位：m

x	0	50	100	150	200	250	300	350	400	450	500	550	600	650	700
h(x)	4.2	5.9	5.8	5.2	4.5	5.7	5	5.5	4.8	5.9	4.1	5.1	4.6	5.7	4.7

实例分析

绪　　论

用计算机进行科学计算解决实际问题的过程如下：

实际问题 → 数学模型 → 数值计算方法 → 程序设计 → 计算机计算求出结果

对数学模型建立数值计算方法，并对方法进行理论分析，直到编程上机计算出结果，以及对结果的分析，这就是数值分析研究的对象和任务。

对象内容

好算法应具有的特点

Tip：学习中，要注意掌握数值方法的基本原理和思想，要注意方法处理的技巧及其与计算机的结合，要重视误差分析、收敛性和稳定性的基本理论。

1. 结构简单，易于计算机实现
2. 理论上要保证方法的收敛性和数值稳定性
3. 计算效率高：计算速度快，节省存储量
4. 经过数值实验检验，证明行之有效

数值分析应用范畴

随着计算机的飞速发展，数值分析方法已深入到计算物理、计算力学、计算化学、计算生物学、计算经济学等各个领域。本课仅限介绍最常用的数学模型的最基本的数值分析方法。

误差的来源和分类

误差分类

模型误差

观测误差

截断误差

舍入误差

误差是描述数值计算之中近似值的精确程度，在数值计算中十分重要，误差按来源可分为模型误差、观测误差、截断误差和舍入误差四种。

1. 模型误差　数学模型通常是由实际问题抽象得到的，一般带有误差，这种误差称为模型误差。

模型误差

误差的来源和分类

误差分类

模型误差

观测误差

截断误差

舍入误差

2. 观测误差 数学模型中包含的一些物理参数通常是通过观测和实验得到的，难免带有误差，这种误差称为**观测误差**。

3. 截断误差　求解数学模型所用的数值方法通常是一种近似方法，这种因方法产生的误差称为**截断误差或方法误差。**

例如，利用ln（x+1）的Taylor公式：

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

实际计算时只能截取有限项代数和计算，如取前5项有：

$$\ln 2\approx 1-\frac{1}{2}+\frac{1}{3}-\frac{1}{4}+\frac{1}{5}$$

这里产生误差（记作R5）

$$R_5=-\frac{1}{6}+\frac{1}{7}-\frac{1}{8}+\frac{1}{9}-\frac{1}{10}+\ldots$$

4. 舍入误差　由于计算机只能对有限位数进行运算，在运算中像 $e, \sqrt{2}, \frac{1}{3}$ 等都要按舍入原则保留有限位，这时产生的误差称为**舍入误差或计算误差**。

在数值分析中，我们总假定数学模型是准确的，因而不考虑模型误差和观测误差，主要研究截断误差和舍入误差对计算结果的影响。

误差的来源和分类

绝对误差

相对误差

设x是精确值x*的一个近似值，记

$$e=x^*-x$$

称e为近似值x的**绝对误差**，简称**误差**。 如果ε满足

$$|e|\leqslant\varepsilon$$

则称ε为近似值x的**绝对误差限**，简称**误差限**。

精确值x* 、近似值x和误差限ε之间满足： $x-\varepsilon\leq x^*\leqslant x+\varepsilon$

通常记为 $x^*=x\pm\varepsilon$

绝对误差有时并不能很好地反映近似程度的好坏，如

$x^*=10$， $\varepsilon_x=1$ ， $y^*=10000$， $\varepsilon_y=5$

虽然ε_y是ε_x的5倍，但在10000内差5显然比10内差1好。

绝对误差

相对误差

记

$$e_r = \frac{e}{x^*} = \frac{x^* - x}{x^*}$$

称e_r为近似值x的**相对误差**。

由于x^*未知，实际使用时总是将x的相对误差取为

$$e_r = \frac{e}{x} = \frac{x^* - x}{x}$$

$\varepsilon_r = \varepsilon / |x|$ 称为近似值x的**相对误差限**。$|e_r| \le \varepsilon_r$.

例1　设x=1.24是由精确值x^*经过四舍五入得到的近似值，求x的绝对误差限和相对误差限。

解　由已知可得：$1.235 \leqslant x^* < 1.245$

所以

$$\varepsilon=0.005,\quad \varepsilon_r = 0.005 \div 1.24 \approx 0.4\%$$

Tip：一般地,凡是由精确值经过四舍五入得到的近似值，其绝对误差限等于该近似值末位的半个单位

有效数字

定义1　设数x是数x*的近似值，如果x的绝对误差限是它的某一数位的半个单位，并且从x左起第一个非零数字到该数位共有n位，则称这n个数字为x的**有效数字**，也称用x近似x*时具有**n位有效数字**。

例2　已知下列近似值的绝对误差限都是0.005，问它们具有几位有效数字？a=12.175, b=-0.10, c=0.1, d=0.0032

解　由于0.005是小数点后第2数位的半个单位，所以a有4位有效数字1、2、1、7，b有2位有效数字1、0，c有1位有效数字1，d没有有效数字。

有效数字

数x总可以写成如下形式

$$x = \pm 0.a_1 a_2 \cdots a_k \times 10^m$$

其中m是整数，a_k是0到9中的一个数字，$a_1 \neq 0$.

x作为x*的近似值，具有n位(n≤k)有效数字当且仅当

$$\left| x^* - x \right| \leq \frac{1}{2} \times 10^{m-n}$$

注：近似值的有效数字越多，其绝对误差越小。

有效数字

例1 为了使 $x^* = \sqrt{2}$ 的近似值的绝对误差小于10^{-5}，问应取几位有效数字？

解：由于$\sqrt{2} = 1.414...$，则近似值x可写为 $x = \pm 0.a_1 a_2 \cdots a_k \times 10^1, a_1 = 1 \neq 0.$

$$令\left|\sqrt{2} - x\right| \leq \frac{1}{2} \times 10^{1-n} < 10^{-5}$$

故取n=6，即取6位有效数字。此时x=1. 41421。

注：精确值的有效数字可认为有无限多位。

有效数字与相对误差的关系

若x有n位有效数字，则其相对误差限为

$$\varepsilon_r \le \frac{1}{2a_1} \times 10^{-n+1}.$$

反之，若x的相对误差限

$$\varepsilon_r \le \frac{1}{2(a_1+1)} \times 10^{-n+1}$$

则x至少有n位有效数字.

数值计算中的若干原则

为了减少舍入误差的影响，设计算法时应遵循如下的一些原则。

1. 避免两个相近的数相减。

如果x* , y* 的近似值分别为x, y，则z=x−y是z* =x* −y*的近似值. 此时，相对误差满足估计式

$$|e_r(z)| = \left|\frac{z^* - z}{z}\right| \le \left|\frac{x}{x-y}\right||e_r(x)| + \left|\frac{y}{x-y}\right||e_r(y)|$$

可见，当x与y很接近时，z的相对误差有可能很大。

在数值计算中，如果遇到两个相近的数相减运算，可考虑改变一下算法以避免两数相减。例如：

当$x_1 \approx x_2$时，有$\log x_1 - \log x_2 = \log\dfrac{x_1}{x_2}$

当$x\approx 0$时，有$1-\cos x=2\sin^2\frac{x}{2}$　当$x>>1$时，有$\sqrt{x+1}-\sqrt{x}=\frac{1}{\sqrt{x+1}+\sqrt{x}}$

例4 求方程$x^2-64x+1=0$的两个根，使它们至少具有四位有效数字（$\sqrt{1023}\approx 31.984$）

解　由求根公式有$x_1=32+\sqrt{1023}\approx 63.984$

若由$x_2=32-\sqrt{1023}\approx 0.016$，仅有两位有效数字。

再由根与系数的关系得$x_2=1/x_1\approx 0.01563$。则有四位有效数

对两个相近的数相减，若找不到适当方法代替，只能在计算机上采用双倍字长计算，以提高精度。

2. 防止大数“吃掉”小数

因为计算机上只能采用有限位数计算，若参加运算的数量级差很大，在它们的加、减运算中，绝对值很小的数往往被绝对值较大的数“吃掉”，造成计算结果失真。

例如，用八位十进制浮点计算A=26358713+0.8+0.2

按照加法浮点运算的对阶规则，应有

$$A=0.26358713\times10^8+0.000000008\times10^8+0.000000002\times10^8$$

由于采用八位数运算，于是有

$$A=26358713$$

若改变计算顺序，计算0.2+0.8+26358713，则有

$$A=0.00000001\times10^8+0.26358713\times10^8=26358714$$

可见，在求和或差的过程中应采用由小到大的运算过程。

3. 绝对值太小的数不宜作除数

由于除数很小, 将导致商很大, 有可能出现“溢出”现象. 另外, 设x^* , y^*的近似值分别为x, y，则$z=x\div y$是$z^*=x^*\div y^*$的近似值. 此时，z的绝对误差满足估计式

$$\left|e(z)\right| = \left|z^* - z\right| = \left|\frac{(x^* - x)y + x(y - y^*)}{yy^*}\right| \approx \frac{|y||e(x)| + |x||e(y)|}{y^2}$$

可见, 若除数太小, 则可能导致商的绝对误差很大。

4. 注意简化计算程序，减少计算次数

首先，若算法计算量太大，实际计算无法完成，例如用Cramer法则求n元线性方程组Ax=b的解，需要计算n+1个n阶行列式，而每个n阶行列式按定义：

$D=\sum_{p_1p_2\cdots p_n}(-1)^t a_{1p_1}a_{2p_2}\cdots a_{np_n}$ 要计算(n−1)n!次乘法，则Cramer法则至少需要(n^2−1)n!

次乘法，当n=20时，有(20^2−1)20!≈9.7×10^{20}次乘法运算。如果用每秒钟计算1百万次乘除运算的计算机，约需要：

$$9.7\times10^{20}\div10^{6}\div60\div60\div24\div365\approx3000\text{万年}$$

其次，即使是可行算法，则计算量越大积累的误差也越大。

例如计算n次多项式：

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

若直接逐项计算，大约需要乘法运算次数为

$$n + (n-1) + \ldots + 2 + 1 = \frac{n(n+1)}{2} \text{次}$$

若将多项式改写为：

$$p_n(x) = (\ldots((a_n x + a_{n-1})x + a_{n-2})x + \ldots)x + a_0$$

则只需n次乘法和n次加法运算。

5. 选用数值稳定性好的算法

一种数值算法，如果其计算舍入误差积累是可控制的，则称其为**数值稳定的**，反之称为**数值不稳定的**。例如积分 $I_n = \int_0^1 x^n e^{x-1} dx$

利用分部积分法可得计算I_n的递推公式

$$I_n = 1 - nI_{n-1}，\quad n=1, 2. . .$$

由于, n=0时，取I_0具有四位有效数字的近似值$I_0 \approx 0.6321$, 递推可得:

$$I_0 = \int_0^1 e^{x-1} dx = 1 - e^{-1} = 0.632120558.....$$

I_0	0.6321	I_2	0.2642	I_4	0.1704	I_6	0.1120	I_8	-0.7280
I_1	0.3679	I_3	0.2074	I_5	0.1480	I_7	0.2160	I_9	7.5520

数值计算中的若干原则

对任何n都应有$I_n>0$, 但计算结果显示$I_8<0$, 可见, 虽然I_0的近似误差不超过0.5×10^{-4}, 但随着计算步数的增加, 误差明显增大。这说明这里的递推公式是数值不稳定的。

事实上，由于$I_n=1-nI_{n-1}$，和$I^*_n=1-nI^*_{n-1}$ ，n=1，2，…

可得$I_n-I^*_n=-n（I_{n-1}-I^*_{n-1}）=\ldots=(-1)^n n!（I_0-I^*_0）$

可见, 随着计算步数的增加, 误差迅速放大，使结果失真。

若将计算公式改写为

$$I_{n-1}=\frac{1}{n}(1-I_n),\quad n=k,k-1,\ldots 2,1$$

类似地可得

$$I_k-I_k^*=(-1)^{n-k}\frac{k!}{n!}(I_n-I_n^*),\quad k=n,n-1,\ldots,1,0$$

可见，近似误差$I_k-I^*_k$是可控制的，算法是数值稳定的。

例如，由于

$$\frac{e^{-1}}{10} = \int_0^1 x^9 e^{-1} dx \le I_9 \le \int_0^1 x^9 dx \le \frac{1}{10}$$

取近似值 $I_9 \approx \frac{1}{2}\left(\frac{e^{-1}}{10} + \frac{1}{10}\right) = 0.0684$，递推可得：

I_9	0.0684	I_7	0.1121	I_5	0.1455	I_3	0.2073	I_1	0.3679
I_8	0.1035	I_6	0.1268	I_4	0.1709	I_2	0.2642	I_0	0.6321

可见，I_0已精确到小数点后四位。