第一讲　引论与预备知识

— 数值计算中的误差

误差与误差来源

误差是人们用来描述数值计算中近似解的精确程度.

- 模型误差: 从实际问题中抽象出数学模型, 往往是抓住主要因素, 忽略次要因素, 因此, 数学模型与实际问题之间总会存在一定的误差.
- 观测误差: 模型中往往包含各种数据或参量, 这些数据一般都是通过测量和实验得到的, 也会存在一定的误差.
- 截断误差: 也称方法误差, 是指对数学模型进行数值求解时产生的误差.
- 舍入误差: 由于计算机的机器字长有限, 做算术运算时存在一定的精度限制, 也会产生误差.

✍ 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.

例 近似计算 $\int_0^1 e^{-x^2}\,\mathrm{d}x$ 的值.

解. 这里我们利用 Taylor 展开, 即

$$\begin{aligned}\int_0^1 e^{-x^2}\,\mathrm{d}x &= \int_0^1 \left(x - x^2 + \frac{x^4}{2!} - \frac{x^6}{3!} + \frac{x^8}{4!} - \cdots\right)\,\mathrm{d}x\\ &= 1 - \frac{1}{3} + \frac{1}{2!}\times\frac{1}{5} - \frac{1}{3!}\times\frac{1}{7} + \frac{1}{4!}\times\frac{1}{9} - \cdots\\ &\triangleq S_4 + R_4,\end{aligned}$$

如果我们以 S_4 作为定积分的近似值, 则 R_4 就是由此而产生的误差, 这种误差就称为截断误差, 它是由我们的近似方法所造成的.

在计算 S_4 的值时, 假定我们保留小数点后 4 位有效数字, 则

$$S_4 = 1 - \frac{1}{3} + \frac{1}{10} - \frac{1}{42} \approx 1.00000 - 0.33333 + 0.10000 - 0.023810 \approx 0.7429$$

这就是我们最后得到的近似值. 这里, 在计算 S_4 时所产生的误差就是舍入误差. □

绝对误差

定义 设 $\tilde{x}$ 是 x 的近似值, 则称

$$\epsilon \triangleq \tilde{x} - x$$

为近似值 $\tilde{x}$ 的**绝对误差**, 简称**误差**.

绝对误差

定义 设 $\tilde{x}$ 是 x 的近似值, 则称

$$\epsilon \triangleq \tilde{x} - x$$

为近似值 $\tilde{x}$ 的**绝对误差**, 简称**误差**. 若存在 $\varepsilon > 0$ 使得

$$|\epsilon| = |\tilde{x} - x| \leq \varepsilon,$$

则称 ε 为**绝对误差限**, 简称**误差限**.

绝对误差

定义 设 $\tilde{x}$ 是 x 的近似值, 则称

$$\epsilon \triangleq \tilde{x} - x$$

为近似值 $\tilde{x}$ 的**绝对误差**, 简称**误差**. 若存在 $\varepsilon > 0$ 使得

$$|\epsilon| = |\tilde{x} - x| \leq \varepsilon,$$

则称 ε 为**绝对误差限**, 简称**误差限**.

✍ 在工程中, 通常用 $x = \tilde{x} \pm \varepsilon$ 表示 $\tilde{x}$ 的误差限为 ε.

关于误差和误差限的几点说明

- 绝对误差不是误差的绝对值, 可能是正的, 也可能是负的.
- 由于精确值通常是不知道的, 因此绝对误差一般也是不可知的.
- 在做误差估计时, 我们所求的通常是误差限;
- 误差限不唯一, 越小越好, 一般是指所能找到的最小上界;
- 近似值的精确程度不能仅仅看绝对误差, 更重要的是看相对误差.

相对误差

定义 设 $\tilde{x}$ 是 x 的近似值, 称

$$\epsilon_r \triangleq \frac{\tilde{x} - x}{x} \qquad \text{或} \qquad \epsilon_r \triangleq \frac{\tilde{x} - x}{\tilde{x}}$$

为近似值 $\tilde{x}$ 的 **相对误差**.

相对误差

定义 设 $\tilde{x}$ 是 x 的近似值, 称

$$\epsilon_r \triangleq \frac{\tilde{x}-x}{x} \qquad \text{或} \qquad \epsilon_r \triangleq \frac{\tilde{x}-x}{\tilde{x}}$$

为近似值 $\tilde{x}$ 的 **相对误差**. 若存在 $\varepsilon_r > 0$ 使得

$$|\epsilon_r| \leq \varepsilon_r,$$

则称 ε_r 为**相对误差限**.

相对误差

定义 设 $\tilde{x}$ 是 x 的近似值, 称

$$\epsilon_r \triangleq \frac{\tilde{x}-x}{x} \qquad \text{或} \qquad \epsilon_r \triangleq \frac{\tilde{x}-x}{\tilde{x}}$$

为近似值 $\tilde{x}$ 的 **相对误差**. 若存在 $\varepsilon_r > 0$ 使得

$$|\epsilon_r| \leq \varepsilon_r,$$

则称 ε_r 为**相对误差限**.

- 近似值的精确程度通常取决于**相对误差**的大小;
- 实际计算中我们所能得到的通常是相对误差限 (所能找到的最小上界);

有效数字

例 已知精确值 $\pi = 3.14159265\cdots$, 则近似值 $x_1 = 3.14$ 有 3 位有效数字, 近似值 $x_2 = 3.1416$ 有 5 位有效数字.

从上面的例子中可以看出, 我们在计算有效数字的个数时, 是 **从最小的有效数字位开始往前数, 直至第一个非零数字为止**, 如果总共有 n 个数字, 那么我们就称其有 n 位有效数字.

有效数字的判断

定理 设 $\tilde{x}$ 是 x 的近似值, 若 $\tilde{x}$ 可表示为

$$\tilde{x} = \pm a_1.a_2 \cdots a_n \cdots \times 10^m,$$

其中 a_i 是 0 到 9 中的数字, 且 $a_1 \neq 0$. 若

$$0.5 \times 10^{m-n} < |\tilde{x} - x| \leq 0.5 \times 10^{m-n+1},$$

则 $\tilde{x}$ 恰好有 n 位有效数字.

等价描述

若 $0.5 \times 10^{k-1} < |\tilde{x} - x| \leq 0.5 \times 10^k$, 则 $\tilde{x}$ 恰好有 $m-k+1$ 位有效数字.

四舍五入

例 根据四舍五入原则写出下列各数的具有 5 位有效数字的近似值:

$$187.9325, \quad 0.03785551, \quad 8.000033.$$

✍ 按四舍五入原则得到的数字都是有效数字.

✍ 一个数末尾的 0 不可以随意添加或省略.

有效数字个数

实际计算中, 往往只知道误差的上限, 因此一般只能判断有效数字的最少个数.

推论 设 $\tilde{x}$ 是 x 的近似值, 若 $\tilde{x}$ 可表示为

$$\tilde{x} = \pm a_1.a_2 \cdots a_n \cdots \times 10^m,$$

其中 a_i 是 0 到 9 中的数字, 且 $a_1 \neq 0$. 若

$$|\tilde{x} - x| \leq 0.5 \times 10^{m-n+1},$$

则 $\tilde{x}$ 至少有 n 位有效数字.

例 已知精确值 $x = 2.7182818\cdots$, 则近似值 $x_1 = 2.7182$ 和 $x_2 = 2.7183$ 分别有几位有效数字?

有效数字与相对误差

定理 (有效数字与相对误差限) 设 $\tilde{x}$ 是 x 的近似值, 若 $\tilde{x}$ 可表示为

$$\tilde{x} = \pm a_1.a_2 \dots a_n \dots \times 10^m,$$

其中 a_i 是 0 到 9 中的数字, 且 $a_1 \neq 0$. 若 $\tilde{x}$ 具有 n 位有效数字, 则相对误差满足

$$|\epsilon_r| \leq \frac{1}{2a_1} \times 10^{-n+1}.$$

反之, 若 $\tilde{x}$ 的相对误差满足

$$|\epsilon_r| \leq \frac{1}{2(a_1+1)} \times 10^{-n+1}.$$

则 $\tilde{x}$ 至少有 n 位有效数字. (板书)

☞ 有效数字越多, 相对误差越小. 同样, 相对误差越小, 则有效数字越多.

误差估计：四则运算

设 $\tilde{x}_1$ 和 $\tilde{x}_2$ 的误差限分别为 $\varepsilon(\tilde{x}_1)$ 和 $\varepsilon(\tilde{x}_2)$，则

$$\varepsilon(\tilde{x}_1 \pm \tilde{x}_2) \le \varepsilon(\tilde{x}_1) + \varepsilon(\tilde{x}_2),$$

$$\varepsilon(\tilde{x}_1 \tilde{x}_2) \le |\tilde{x}_2|\varepsilon(\tilde{x}_1) + |\tilde{x}_1|\varepsilon(\tilde{x}_2) + \varepsilon(\tilde{x}_1)\varepsilon(\tilde{x}_2) \lessapprox |\tilde{x}_2|\varepsilon(\tilde{x}_1) + |\tilde{x}_1|\varepsilon(\tilde{x}_2),$$

$$\varepsilon\left(\frac{\tilde{x}_1}{\tilde{x}_2}\right) \lessapprox \frac{|\tilde{x}_2|\varepsilon(\tilde{x}_1) + |\tilde{x}_1|\varepsilon(\tilde{x}_2)}{|\tilde{x}_2|^2}.$$

误差估计：单变量可微函数

一般地, 设 $\tilde{x}$ 是 x 的近似值, 若 $f(x)$ 可导, 则有

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

由于 $\tilde{x} - x$ 相对较小, 所以当 $|f''(x)|$ 与 $|f'(x)|$ 的比值不是很大时, 我们可以忽略二阶项, 即

$$|f(\tilde{x}) - f(x)| \approx |f'(x)| \cdot |\tilde{x} - x|.$$

因此, 可得函数值的误差限

$$\varepsilon(f(\tilde{x})) \approx |f'(x)| \, \varepsilon(\tilde{x}) \approx |f'(\tilde{x})| \, \varepsilon(\tilde{x}).$$

例 设 $x>0$, x 的相对误差是 δ, 试估计 $\ln x$ 的误差.

解. 设 $\tilde{x}$ 是 x 的近似值. 由题意可知, 相对误差为

$$\left|\frac{\tilde{x}-x}{\tilde{x}}\right|=\delta.$$

所以误差 $|\varepsilon(\tilde{x})|=|\tilde{x}-x|=|\tilde{x}|\cdot\delta$. 设 $f(x)=\ln(x)$, 则

$$\varepsilon(f(\tilde{x}))\approx|f'(\tilde{x})|\varepsilon(\tilde{x})=\left|\frac{1}{\tilde{x}}\right|\cdot|\tilde{x}|\cdot\delta=\delta,$$

即 $\ln x$ 的误差约为 δ. □

多变量可微函数

关于多元函数 $f(x_1, x_2, \ldots, x_n)$, 我们可以得到类似的结论:

$$\varepsilon(f(\tilde{x})) \approx \sum_{k=1}^{n} \left| \frac{\partial f(\tilde{x})}{\partial x_k} \right| \varepsilon(\tilde{x}_k),$$

其中 $\tilde{x} = [\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n]^{\mathsf{T}}$ 是 $x = [x_1, x_2, \ldots, x_n]^{\mathsf{T}}$ 的近似值.

例 测得某场地的长 x 和宽 y 分别为: $\tilde{x}=110\text{m}$, $\tilde{y}=80\text{m}$, 其测量误差限分别为 0.2m 和 0.1m. 试求面积 $S=x\times y$ 的绝对误差限和相对误差限.

解. 由于 $\varepsilon(\tilde{x})=0.2\text{m}$, $\varepsilon(\tilde{y})=0.1\text{m}$, 故

$$\begin{aligned}\varepsilon(\tilde{S}) &\approx \left|\frac{\partial S(\tilde{x},\tilde{y})}{\partial x}\right|\varepsilon(\tilde{x})+\left|\frac{\partial S(\tilde{x},\tilde{y})}{\partial y}\right|\varepsilon(\tilde{y})\\ &=|\tilde{y}|\cdot\varepsilon(\tilde{x})+|\tilde{x}|\cdot\varepsilon(\tilde{y})\\ &=80\times 0.2+110\times 0.1=27(\text{m}^2).\end{aligned}$$

相对误差限

$$\varepsilon_r(\tilde{S})=\frac{\varepsilon(\tilde{S})}{|\tilde{S}|}\approx\frac{27}{110\times 80}\approx 0.0031.$$

□

误差分析

由于在进行数值计算时, 不仅仅原始数据可能带有误差, 在计算过程中也会出现舍入误差, 而且误差会不断 **传播** 和 **积累**, 也会**对消**.

因此误差分析通常比较复杂.

目前的误差分析方法可分为定量分析和定性分析.

定量分析与定性分析

- 主要方法有: 向后误差分析法, 向前误差分析法, 概率分析法等
- 向后误差分析法比较有效

✍ 实际计算中的运算次数通常都在千万次以上, 如果对每一步运算都做误差估计的话, 工作量会非常大, 而且得到的误差界也往往不太实用.

定性分析

- 目前在数值计算中更关注的是误差的定性分析;
- 定性分析包括数值算法的稳定性, 数学问题与原问题的相容性, 数学问题的适定性, 避免扩大误差的准则等;
- 定性分析的核心是原始数据的误差和计算过程中产生的误差对最终计算结果的影响.

✍ 算法有 "优劣" 之分, 问题有 "好坏" 之别, 即使不能定量地估计出最终误差, 但是若能确保计算过程中 **误差不会被任意放大**, 那就能放心地实施计算, 这就是误差定性分析的初衷.

数学问题的适定性

定义 如果数学问题满足

(1) 对任意满足一定条件的输入数据，存在一个解，

(2) 对任意满足一定条件的输入数据，解是唯一的，

(3) 问题的解关于输入数据是连续的，

则称该数学问题是**适定的** (well-posed)，否则就称为**不适定的** (ill-posed).

病态问题与条件数

定义 如果输入数据的微小扰动会引起输出数据 (即计算结果) 的很大变化 (误差), 则称该数学问题是**病态**的, 否则就是**良态**的.

例 解线性方程组 $\begin{cases} x + \alpha y = 1 \\ \alpha x + y = 0 \end{cases}$

解. 易知当 $\alpha = 1$ 时, 方程组无解. 当 $\alpha \neq 1$ 时, 解为

$$x = \frac{1}{1-\alpha^2}, \quad y = \frac{-\alpha}{1-\alpha^2}.$$

如果 $\alpha \approx 1$, 则 α 的微小误差可能会引起解的很大变化. 比如当 $\alpha = 0.999$ 时, $x \approx 500.25$. 假定输入数据 α 带有 0.0001 的误差, 即实际输入数据为 $\tilde{\alpha} = 0.9991$, 则此时有 $\tilde{x} \approx 555.81$, 解的误差约为 55.56, 是输入数据误差的五十多万倍, 因此该问题的病态的. □

单变量函数的条件数

设 $f(x)$ 可导, $\tilde{x}$ 是精确值 x_* 的近似值, 则由 Taylor 公式可知

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

当 $\tilde{x}$ 很接近 x_* 时, $(\tilde{x}-x_*)^2$ 非常小. 假定 $f''(x)$ 在 x_* 附近的邻域内有界且不是很大, 则有

$$\left|\frac{f(\tilde{x})-f(x_*)}{f(x_*)}\right|\approx\left|\frac{x_*f'(x_*)}{f(x_*)}\right|\times\left|\frac{\tilde{x}-x_*}{x_*}\right|,$$

即函数值的相对误差大约是输入数据相对误差的 $\left|\frac{x_*f'(x_*)}{f(x_*)}\right|$ 倍. 这个值就定义为函数 $f(x)$ 在 x_* 处的 **条件数**, 记为 $C_p(x_*)$, 其中

$$\boxed{C_p(x)\triangleq\left|\frac{xf'(x)}{f(x)}\right|}$$

几点说明

- 如果条件数比较大, 则就认为问题是病态的, 而且条件数越大问题病态就越严重;
- 病态是问题本身固有的性质, 与数值算法无关;
- 对于病态问题, 选择数值算法时需要更加谨慎.

算法的稳定性 I

如果误差不增长或能得到有效控制, 则称该算法是**稳定**的, 否则为**不稳定**的.

例 近似计算 (Demo11_Stablity.m)

$$S_n = \int_0^1 \frac{x^n}{x+5}\,\mathrm{d}x, \quad n = 1, 2, \ldots, 8.$$

解 通过观察可知

$$S_n + 5S_{n-1} = \int_0^1 \frac{x^n + 5x^{n-1}}{x+5}\,\mathrm{d}x = \int_0^1 x^{n-1} = \frac{1}{n},$$

因此,

$$S_n = \frac{1}{n} - 5S_{n-1}. \tag{1.1}$$

算法的稳定性 II

易知 $S_0 = \ln 6 - \ln 5 \approx 0.182$ (保留三位有效数字), 利用上面的递推公式可得 (保留三位有效数字)

$$S_1 = 0.0900, \quad S_2 = 0.0500, \quad S_3 = 0.0833, \quad S_4 = -0.166,$$
$$S_5 = 1.03, \quad S_6 = -4.98, \quad S_7 = 25.0, \quad S_8 = -125.$$

另一方面, 我们有

$$\frac{1}{6(n+1)} = \int_0^1 \frac{x^n}{6}\,\mathrm{d}x \leq \int_0^1 \frac{x^n}{x+5}\,\mathrm{d}x \leq \int_0^1 \frac{x^n}{5}\,\mathrm{d}x = \frac{1}{5(n+1)}. \tag{1.2}$$

因此, 上面计算的 $S_4, \ldots, S_8$ 显然是不对的. 原因是什么呢? 误差!

设 $\tilde{S}_n$ 是 S_n 的近似值, 则

$$\epsilon(\tilde{S}_n) = \tilde{S}_n - S_n = \left(\frac{1}{n} - 5\tilde{S}_{n-1}\right) - \left(\frac{1}{n} - 5S_{n-1}\right) \approx -5(\tilde{S}_{n-1} - S_{n-1}) = -5\epsilon(\tilde{S}_{n-1}).$$

算法的稳定性 III

即误差是以 5 倍速度增长, 这说明计算过程是不稳定的, 因此我们不能使用该算法.

事实上, 递推公式 (1.1) 可以改写为

$$S_{n-1} = \frac{1}{5n} - \frac{1}{5} S_n.$$

因此, 我们可以先估计 S_8 的值, 然后通过反向递推, 得到其它值.

我们可以根据 (1.2) 对 S_8 做简单的估计, 即

$$S_8 \approx \frac{1}{2} \left(\int_0^1 \frac{x^n}{6} \, \mathrm{d}x + \int_0^1 \frac{x^n}{5} \, \mathrm{d}x \right) \approx 0.0204.$$

于是

$$S_7 = 0.0209, \quad S_6 = 0.0244, \quad S_5 = 0.0285, \quad S_4 = 0.0343,$$
$$S_3 = 0.0431, \quad S_2 = 0.0580, \quad S_1 = 0.0884, \quad S_0 = 0.182.$$

通过误差分析可知, 误差是以 $\frac{1}{5}$ 的速度减小, 因此计算过程是稳定的.

几点说明

- ✍ 在数值计算中, 误差不可避免, 算法的稳定性是一个非常重要的性质.
- ✍ 在数值计算中, 不要采用不稳定的算法!
- ✍ 用计算机进行整数之间的加减和乘法运算时, 没有误差. (不考虑溢出情况)

减小误差危害

为了尽量减小误差给计算结果带来的危害, 在数值计算过程中, 我们需注意以下几点.

(1) 避免相近的数相减

如果两个相近的数相减, 则会损失有效数字, 如 $0.12346 - 0.12345 = 0.00001$, 两个操作数都有 5 位有效数字, 但计算结果却只有 1 位有效数字.

例 计算 $\sqrt{9.01}-3$, 计算过程中保留 3 位有效数字.

解. 如果直接计算的话, 可得

$$\sqrt{9.01}=3.0016662039607\cdots\approx 3.00.$$

所以 $\sqrt{9.01}-3\approx 0.00$, 一个有效数字都没有!

但如果换一种计算方法, 如

$$\sqrt{9.01}-3=\frac{9.01-3^2}{\sqrt{9.01}+3}\approx\frac{0.01}{3.00+3}\approx 0.00167.$$

通过精确计算可知 $\sqrt{9.01}-3=0.0016662039607\cdots$. 因此第二种计算能得到三位有效数字! □

如何避免相近的数相减

通过各种等价公式来计算两个相近的数相减，是避免有效数字损失的有效手段之一.

下面给出几个常用的等价公式:

$$\sqrt{x+\varepsilon}-\sqrt{x}=\frac{\varepsilon}{\sqrt{x+\varepsilon}+\sqrt{x}}$$

$$\ln(x+\varepsilon)-\ln(x)=\ln\left(1+\frac{\varepsilon}{x}\right)$$

$$1-\cos(x)=2\sin^2\frac{x}{2},\quad |x|\ll 1$$

$$\mathrm{e}^x-1=x\left(1+\frac{1}{2}x+\frac{1}{6}x^2+\cdots\right),\quad |x|\ll 1$$

例 计算 $y=\left(\dfrac{\sqrt{101}-10}{\sqrt{101}+10}\right)^2$.

方法一：分母有理化 $y=\left(\frac{\sqrt{101}-10}{\sqrt{101}+10}\right)^2=80801-8040\sqrt{101}$;

方法二：分子有理化 $y=\left(\frac{\sqrt{101}-10}{\sqrt{101}+10}\right)^2=\frac{1}{80801+8040\sqrt{101}}$;

方法三：直接将 $\sqrt{101}$ 的近似值代入计算.

已知 $\sqrt{101}=10.0498756\cdots$, 分别取近似值 $10.04, 10.05, 10.06$, 计算结果如下:

$\sqrt{101}$	方法一	方法二	方法三
10.04	79.400	6.1911E-6	3.9840E-6
10.05	-1.0000	6.1880E-6	6.2189E-6
10.06	-81.400	6.1849E-6	8.9462E-6

实际值为 $y=6.188042227\cdots\times 10^{-6}$.

例 在 MATLAB 中用双精度数计算 (Demo12_Significance.m)

$$E_1 = \frac{1-\cos(x)}{\sin^2(x)} \quad 和 \quad E_2 = \frac{1}{1+\cos(x)}.$$

x	E_1	E_2
1.0000000000	0.649223205204762	0.649223205204762
0.1000000000	0.501252086288577	0.501252086288571
0.0100000000	0.500012500208481	0.500012500208336
0.0010000000	0.500000124992189	0.500000125000021
0.0001000000	0.499999998627931	0.500000001250000
0.0000100000	0.500000041386852	0.500000000012500
0.0000010000	0.500044450291337	0.500000000000125
0.0000001000	0.499600361081322	0.500000000000001
0.0000000100	0.000000000000000	0.500000000000000

由此可见，当 x 趋于 0 是，E_1 的计算结果显然是错的，而 E_2 则能很好地计算出近似值.

例 计算 $y = \ln 2$. (Demo13_ln.m)

方法一: 利用 $f(x) = \ln(1+x)$ 的 Taylor 展开

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

将 $x = 1$ 代入后计算结果为

n	1	2	3	4	5	10	50	100
S_n	1	0.5	0.833	0.583	0.783	0.646	0.683	0.688
误差	3.1E-1	1.9E-1	1.4E-1	1.1E-1	9.0E-2	4.8E-2	9.9E-3	5.0E-3

计算到第 100 项, 误差仍有 0.05.

方法二：利用 $f(x)=\ln\left(\dfrac{1+x}{1-x}\right)$ 的 Taylor 展开

$$\ln\left(\frac{1+x}{1-x}\right)=2\left(x+\frac{1}{3}x^3+\frac{1}{5}x^5+\cdots+\frac{1}{2n-1}x^{2n-1}+\cdots\right).$$

将 $x=1/3$ 代入后计算结果为

n	1	2	3	4	5	10
S_n	0.667	0.691	0.693	0.693	0.693	0.693
$\lvert S_n-\ln 2\rvert$	2.6E-2	1.8E-3	1.4E-4	1.2E-5	1.1E-6	1.0E-11

计算到第 10 项，误差已经小于 10^{-10}！实际值为 $\ln 2=0.693147180559945$.

(2) 避免数量级相差很大的数相除

可能会产生溢出, 即超出计算机所能表示的数的范围. 特别需要注意的是, 尽量不要用很小的数作为除数, 否则为放大分子的误差.

✍ 如果两个数相除, 一般情况下建议把绝对值小的数作为分子, 这在后面的算法中会经常遇到.

(3) 避免大数吃小数

如 $(10^9+10^{-9}-10^9)/10^{-9}$, 直接计算的话, 结果为 0.

另外, 在对一组数求和时, 建议按照绝对值从小到大求和.

例 计算 $S=1+2+3+\cdots+100+10^{16}$. (`Demo14_Sum.m`)

从小到大计算, 结果为

$$S=1+2+3+\cdots+100+10^{16}=1.0000000000005050\times10^{16}.$$

从大到小计算, 结果为

$$S=10^{16}+100+99+\cdots+2+1=1.0000000000005100\times10^{16}.$$

(4) 简化计算

尽量减少运算次数, 从而减少误差的积累.

例 多项式计算. 设多项式 (Demo15_Poly.m)

$$p(x) = 5x^5 + 4x^4 + 3x^3 + 2x^2 + 2x + 1.$$

试计算 $p(3)$ 的值.

方法一: 直接计算

$$p(3) = 5 \times 3^5 + 4 \times 3^4 + 3 \times 3^3 + 2 \times 3^2 + 2 \times 3 + 1.$$

需要做 15 次乘法和 5 次加法.

方法二: 当计算 x^k 时, 由于前面已经计算出 x^{k-1}, 因此只需做一次乘法. 这样整个计算过程可以减少到 9 次乘法和 5 次加法.

方法三: 有没有更快的?

✍ 在计算多项式的值时，我们都是将多项式改写成

$$
\begin{aligned}
p(x) &= a_n x^n + a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \cdots + a_1 x + a_0 \\
&= ((\cdots((a_n x + a_{n-1})x + a_{n-2})x + \cdots)x + a_1)x + a_0.
\end{aligned}
$$

这里利用了嵌套思想，如果直接计算的话，需要 $\frac{n(n+1)}{2}$ 次乘法和 n 次加法. 但如果采用秦九韶算法的话，只需做 n 次乘法和 n 次加法.

这种计算方法就是著名的 秦九韶算法 (1247)，五百多年后，英国数学家 Horner (1819) 重新发现了该公式，因此西方也称为 Horner 算法.

方法三：我们可以将多项式改写为

$$
p(x) = ((((5x + 4)x + 3)x + 2)x + 2)x + 1.
$$

这样就只需做 5 次乘法和 5 次加法. 显然这是更佳的计算方案.