格拉姆-施密特过程及相关算法的误差分析

邹秦萌　编著

GELAMU-SHIMITE GUOCHENG JI
XIANGGUAN SUANFA DE WUCHA FENXI

北京邮电大学出版社
www.buptpress.com

格拉姆-施密特过程及相关算法的误差分析

邹秦萌　编著

北京邮电大学出版社
www.buptpress.com

内 容 简 介

格拉姆-施密特过程在线性方程组求解、特征值计算、最小二乘问题中应用广泛。本书深入讨论了矩阵误差分析的思想和理论，主要内容包括误差分析基础知识、传统和改进的格拉姆-施密特过程的算法和误差分析、重正交化技术、极小残差法、分块格拉姆-施密特过程等，证明过程用到的相关算法也都在有限精度下进行了分析。本书适合计算数学相关专业的研究生和科研工作者阅读，也可作为从事科学与工程计算的广大技术人员的参考书。

图书在版编目(CIP)数据

格拉姆-施密特过程及相关算法的误差分析 / 邹秦萌编著. -- 北京：北京邮电大学出版社，2022.6

ISBN 978-7-5635-6648-8

Ⅰ. ①格… Ⅱ. ①邹… Ⅲ. ①线性代数计算法 Ⅳ. ①O241.6

中国版本图书馆 CIP 数据核字(2022)第 087543 号

策划编辑：彭 楠 **责任编辑**：王晓丹 陶 恒 **封面设计**：七星博纳

出版发行：北京邮电大学出版社
社　　址：北京市海淀区西土城路 10 号
邮政编码：100876
发 行 部：电话：010-62282185 传真：010-62283578
E-mail：publish@bupt.edu.cn
经　　销：各地新华书店
印　　刷：唐山玺诚印务有限公司
开　　本：720 mm×1 000 mm 1/16
印　　张：7.25
字　　数：122 千字
版　　次：2022 年 6 月第 1 版
印　　次：2022 年 6 月第 1 次印刷

ISBN 978-7-5635-6648-8 定价：42.00 元

前　言

本书讨论的对象主要是格拉姆-施密特过程。格拉姆-施密特是一类正交化技术，是实现矩阵 QR 分解的一类算法，其英文为 Gram-Schmidt，由两位数学家的姓氏组成，这两位数学家分别是 J. P. 格拉姆（J. P. Gram）和 E. 施密特（E. Schmidt）。

格拉姆（1850—1916 年）生于丹麦，1873 年获得硕士学位，从 1875 年起在哈夫尼亚保险公司从事精算科学工作，1879 年获得博士学位。他在 1884 年创建了自己的保险公司，担任总裁，1895 年成为原公司的董事会成员。自 1910 年起，格拉姆担任丹麦保险协会主席，业余时间他积极参加丹麦皇家科学院的数学活动，研究概率论和数值分析，并用其解决实际问题，同时也在数论领域取得了一些成果。施密特（1876—1959 年）生于爱沙尼亚，1905 年在哥廷根大学获得博士学位，是数学家希尔伯特的学生，1917 年获得柏林大学的教授职位。在柏林大学工作期间，施密特担任过数学系的共同主任、学院院长和副校长，参与创建了应用数学中心。1946—1958 年，施密特担任德国科学院数学研究中心主任。1948 年，施密特参与创建数学期刊 *Mathematische Nachrichten*，并成为首位主编。施密特在泛函分析领域做出了重要贡献。

格拉姆在 1879 年用丹麦语写成的文章中，借助最小二乘法研究了实函数的级数展开问题。该文章关注函数的线性组合，提出了一种更新策略来寻找函数的最佳逼近，该策略用到了正交函数。在这篇文章中，格拉姆首先讨论了离散内积下的正交函数，然后讨论了非离散情况。该文章于 1883 年用德语发表，在积分方程领域有一定的重要性。该文章传达了一个重要思想，那就是给定一组函数，可以构建一组正交函数，其中新的正交函数能够表示为新的原始函数和旧的原始函数的线性组合。受到格拉姆工作的启发，施密特在

1907 年发表了一篇文章，研究积分方程的求解问题。该文章提出了一种算法，能够将积分方程的特征函数正交化，其中新的正交函数由新的原始函数和旧的正交函数组合而成。施密特在文中声明其算法与格拉姆的算法等价，因此两人的姓氏被合在一起，共同用来命名这一算法。实际上，早在 1820 年，法国数学家拉普拉斯就已在著作中提出了连续正交投影。不过直到 1907 年，施密特的文章才令这类方法广为人知。拉普拉斯（1749—1827 年）比格拉姆和施密特名气更大，他在数学和物理学领域都做出了重要贡献。

一般称施密特算法的向量版本为传统的格拉姆-施密特算法，或简称为格拉姆-施密特算法，而称拉普拉斯算法为改进的格拉姆-施密特算法。除此之外，格拉姆-施密特算法还有多种变体，在数值意义上各不相同。即使算法在数学上等价，用计算机进行计算时也会产生很大差别，原因在于舍入误差的影响。广义极小残差法是格拉姆-施密特过程的一个应用实例，用来求解非对称线性方程组，早在 1983 年就以报告的形式受到关注，1986 年正式发表。虽然在实验中能够观察到广义极小残差法是数值稳定的，但由于长期缺少严格的舍入误差分析，直到 2006 年其数值稳定性才得到证明。在此之前，正交化过程往往采用更稳定的豪斯霍尔德变换来实现。可以看出，舍入误差分析对数值算法有重要意义。

本书关注格拉姆-施密特过程的误差分析。第 1 章为绪论，第 2 章介绍传统和改进的格拉姆-施密特过程的算法和误差分析，第 3 章研究重正交化技术，第 4 章研究极小残差法，第 5 章介绍分块格拉姆-施密特算法，第 6 章为总结与展望。本书的出版得到国家自然科学基金（项目编号：12101071）的资助，在此表示感谢。由于作者水平有限，书中错误与片面之处在所难免，请读者不吝指正。

作　者

2022 年 2 月于北京

目　录

第1章 绪 论

本章主要介绍误差的概念，给出基本的符号和定义，并简要介绍格拉姆-施密特过程的背景。概括而言，格拉姆-施密特是一类经典的正交化技术，在最小二乘问题及线性方程组求解、特征值计算问题等方面有广泛的应用。误差分析能够揭示格拉姆-施密特算法的稳定性，从而在实际应用中避免误差危害。

1.1 QR 分解

给定矩阵 $\boldsymbol{A}=(a_{i,j})\in\mathbb{R}^{m\times n}$，即

$$\boldsymbol{A}=(a_1,\cdots,a_n)=\begin{pmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{pmatrix},$$

其中 $m\geqslant n$，计算列向量两两正交的矩阵 $\boldsymbol{Q}\in\mathbb{R}^{m\times n}$ 和上三角矩阵 $\boldsymbol{R}\in\mathbb{R}^{n\times n}$，使得 $\boldsymbol{A}=\boldsymbol{QR}$。该过程被称为瘦 QR 分解，或简称 QR 分解。$\boldsymbol{A}$ 的 QR 分解必定存在，且在一定条件下具有唯一性。下面的定理描述了 QR 分解的唯一性。

定理 1.1 设 $\boldsymbol{A}\in\mathbb{R}^{m\times n}$，其中 $m\geqslant n$。若 $\boldsymbol{A}$ 的秩为 n，且 $\boldsymbol{Q}\in\mathbb{R}^{m\times n}$ 的各列为两两正交的单位向量，$\boldsymbol{R}\in\mathbb{R}^{n\times n}$ 为具有正对角元的上三角矩阵，则 $\boldsymbol{A}$ 的 QR 分解 $\boldsymbol{A}=\boldsymbol{QR}$ 是唯一的。

定理 1.1 可用具体的 QR 算法过程加以证明。

QR 分解可用多种方法计算。本书主要关注格拉姆-施密特过程（后文经常用

Gram-Schmidt 的首字母缩写“GS”来表示)。若给定一个线性无关的向量组$\{\boldsymbol{a}_1,\cdots,\boldsymbol{a}_n\}$,希望构建一个正交向量组$\{\boldsymbol{q}_1,\cdots,\boldsymbol{q}_n\}$,格拉姆-施密特过程可以简述如下:

① $\boldsymbol{q}_1=\boldsymbol{a}_1$;

② $\boldsymbol{q}_k=\boldsymbol{a}_k-\sum_{j=1}^{k-1}\frac{(\boldsymbol{a}_k,\boldsymbol{q}_j)}{\|\boldsymbol{q}_j\|^2}\boldsymbol{q}_j,k=2,\cdots,n$。

这里$(\boldsymbol{u},\boldsymbol{v})=\boldsymbol{v}^{\mathrm{T}}\boldsymbol{u}$是点积运算,$\|\boldsymbol{u}\|=\sqrt{(\boldsymbol{u},\boldsymbol{u})}$是欧式范数。所得正交向量组构成子空间的一个正交基。若需要标准正交基,可令$\boldsymbol{q}_k=\boldsymbol{q}_k/\|\boldsymbol{q}_k\|$,即

① $\boldsymbol{q}_1=\boldsymbol{a}_1/\|\boldsymbol{a}_1\|$;

② $\boldsymbol{w}_k=\boldsymbol{a}_k-\sum_{j=1}^{k-1}(\boldsymbol{a}_k,\boldsymbol{q}_j)\boldsymbol{q}_j,\boldsymbol{q}_k=\boldsymbol{w}_k/\|\boldsymbol{w}_k\|,k=2,\cdots,n$。

虽然没有正交化的向量组也可作为一组基,但在数值计算中,误差会不断积累,使得计算结果不稳定。而正交基不会产生该问题。然而,受误差影响,格拉姆-施密特过程的计算结果不会是严格正交的,正交性损失的程度因算法实现的不同而有显著差异,这也是本书的主要研究内容。

格拉姆-施密特过程可用来计算 QR 分解。由上述内容,可令

$$\boldsymbol{a}_k=r_{1,k}\boldsymbol{q}_1+\cdots+r_{k,k}\boldsymbol{q}_k, \tag{1.1}$$

其中$r_{i,j}\in\mathbb{R}$。若$\boldsymbol{q}_1,\cdots,\boldsymbol{q}_{k-1}$已知,由式(1.1)可得

$$r_{k,k}\boldsymbol{q}_k=\boldsymbol{a}_k-\sum_{j=1}^{k-1}r_{j,k}\boldsymbol{q}_j, \tag{1.2}$$

其中$r_{j,k}=(\boldsymbol{a}_k,\boldsymbol{q}_j)$。令$\boldsymbol{w}_k=r_{k,k}\boldsymbol{q}_k$,则每步迭代都能由新的原始向量和旧的正交向量算出$\boldsymbol{w}_k$,继而得到

$$r_{k,k}=\|\boldsymbol{w}_k\|,\quad \boldsymbol{q}_k=\frac{\boldsymbol{w}_k}{r_{k,k}}。 \tag{1.3}$$

写成矩阵形式,则有

$$\boldsymbol{A}=(\boldsymbol{q}_1,\cdots,\boldsymbol{q}_n)\begin{pmatrix} r_{1,1} & r_{1,2} & r_{1,3} & \cdots & r_{1,n}\\ 0 & r_{2,2} & r_{2,3} & \cdots & r_{2,n}\\ 0 & 0 & r_{3,3} & \cdots & r_{3,n}\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & \cdots & 0 & 0 & r_{n,n}\end{pmatrix}=\boldsymbol{QR},$$

其中 $\boldsymbol{Q}\in\mathbb{R}^{m\times n}$，$\boldsymbol{R}\in\mathbb{R}^{n\times n}$。这里 $r_{j,k}\boldsymbol{q}_j=(\boldsymbol{a}_k,\boldsymbol{q}_j)\boldsymbol{q}_j=\boldsymbol{q}_j\boldsymbol{q}_j^{\mathrm{T}}\boldsymbol{a}_k$，其中 $\boldsymbol{q}_j\boldsymbol{q}_j^{\mathrm{T}}$ 是正交投影算子，而式(1.2)每次用 $\boldsymbol{a}_k$ 减去 $\boldsymbol{q}_j\boldsymbol{q}_j^{\mathrm{T}}\boldsymbol{a}_k$ 相当于将 $\boldsymbol{a}_k$ 正交投影到 $\boldsymbol{q}_j$ 的正交补空间。若令 $\boldsymbol{R}$ 的对角元 $r_{1,1},\cdots,r_{n,n}$ 为正，如式(1.3)所示，则容易看出 QR 分解是唯一的。定理 1.1 得证。

具体算法将在后文详细介绍。除格拉姆-施密特过程之外，豪斯霍尔德(Householder)反射和吉文斯(Givens)旋转也可用来实现 QR 分解。豪斯霍尔德反射定义如下：

$$\widetilde{\boldsymbol{H}}_k=\boldsymbol{I}-2\frac{\widetilde{\boldsymbol{v}}_k\widetilde{\boldsymbol{v}}_k^{\mathrm{T}}}{\widetilde{\boldsymbol{v}}_k^{\mathrm{T}}\widetilde{\boldsymbol{v}}_k},\quad \widetilde{\boldsymbol{v}}_k=\boldsymbol{a}_k\pm\|\boldsymbol{a}_k\|\boldsymbol{e}_k,$$

其中 $\boldsymbol{I}$ 是单位矩阵，$\boldsymbol{e}_k$ 是单位向量的第 k 列，$\widetilde{\boldsymbol{H}}_k$ 又称豪斯霍尔德矩阵，$\widetilde{\boldsymbol{v}}_k$ 称作豪斯霍尔德向量。豪斯霍尔德变换能将 $\boldsymbol{a}_k$ 变换到 $\boldsymbol{e}_k$ 的方向，即

$$\widetilde{\boldsymbol{H}}_k\boldsymbol{a}_k=\mp\|\boldsymbol{a}_k\|\boldsymbol{e}_k,$$

变换前后的两个向量关于以 $\widetilde{\boldsymbol{v}}_k$ 为法向量的超平面对称。由构造可知 $\widetilde{\boldsymbol{H}}_k^{\mathrm{T}}=\widetilde{\boldsymbol{H}}_k$，$\widetilde{\boldsymbol{H}}_k^{\mathrm{T}}\widetilde{\boldsymbol{H}}_k=\boldsymbol{I}$，因此 $\widetilde{\boldsymbol{H}}_k=\widetilde{\boldsymbol{H}}_k^{-1}$。用豪斯霍尔德变换实现 QR 分解，相当于依次保留矩阵 $\boldsymbol{A}$ 第 k 列的前 k 个元素，并将剩余元素清零。令 $\boldsymbol{H}_k$ 为待求的豪斯霍尔德矩阵，$\boldsymbol{A}^{(1)}=\boldsymbol{A}$，$\boldsymbol{A}^{(k)}=\boldsymbol{H}_{k-1}\cdots\boldsymbol{H}_1\boldsymbol{A}$，不妨设前 $k-1$ 列已经上三角化。取 m 维列向量

$$\boldsymbol{x}_k=(0,\cdots,0,a_{k,k}^{(k)},\cdots,a_{m,k}^{(k)})^{\mathrm{T}},$$

然后令

$$\boldsymbol{H}_k=\boldsymbol{I}-2\frac{\boldsymbol{v}_k\boldsymbol{v}_k^{\mathrm{T}}}{\boldsymbol{v}_k^{\mathrm{T}}\boldsymbol{v}_k},\quad \boldsymbol{v}_k=\boldsymbol{x}_k+\mathrm{sign}(a_{k,k}^{(k)})\|\boldsymbol{x}_k\|\boldsymbol{e}_k,$$

可知 $\boldsymbol{A}^{(k+1)}=\boldsymbol{H}_k\cdots\boldsymbol{H}_1\boldsymbol{A}$ 的前 k 列完成上三角化，其中符号函数 sign 的目的是减少数值误差的影响。于是有 $\boldsymbol{A}=\boldsymbol{Q}\boldsymbol{R}$，其中 $\boldsymbol{R}$ 等于 $\boldsymbol{A}^{(n+1)}$ 的前 n 行，$\boldsymbol{Q}$ 等于 $\boldsymbol{H}_1\cdots\boldsymbol{H}_n$ 的前 n 列。本质上，格拉姆-施密特方法是将向量组正交化，顺便得到上三角矩阵，这一过程称作“三角正交化”；而豪斯霍尔德方法是将矩阵上三角化，顺便得到正交向量组，这一过程称作“正交三角化”。

吉文斯旋转定义如下：

$$\boldsymbol{G}_{i,j}=\begin{pmatrix}1 & \cdots & 0 & \cdots & 0 & \cdots & 0\\ \vdots & \ddots & \vdots & & \vdots & & \vdots\\ 0 & \cdots & c & \cdots & s & \cdots & 0\\ \vdots & & \vdots & \ddots & \vdots & & \vdots\\ 0 & \cdots & -s & \cdots & c & \cdots & 0\\ \vdots & & \vdots & & \vdots & \ddots & \vdots\\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1\end{pmatrix},$$

其中 $c=\cos(\theta)$，$s=\sin(\theta)$。由该定义可以看出单位矩阵的(i,i)、(i,j)、(j,i)、(j,j)4 个位置被分别替换成了 c、s、$-s$、c。用 $\boldsymbol{G}_{i,j}^{\mathrm{T}}$左乘向量 $\boldsymbol{a}_k$ 相当于在(i,j)平面上逆时针旋转 θ 度。可以看出 $\boldsymbol{G}_{i,j}$是正交矩阵。令

$$c=\frac{a_{i,k}}{\sqrt{a_{i,k}^2+a_{j,k}^2}},\quad s=\frac{a_{j,k}}{\sqrt{a_{i,k}^2+a_{j,k}^2}},$$

则

$$\boldsymbol{G}_{i,j}\boldsymbol{a}_k=\boldsymbol{G}_{i,j}\begin{pmatrix}a_{1,k}\\ \vdots\\ a_{i,k}\\ \vdots\\ a_{j,k}\\ \vdots\\ a_{m,k}\end{pmatrix}=\begin{pmatrix}a_{1,k}\\ \vdots\\ ca_{i,k}+sa_{j,k}\\ \vdots\\ -sa_{i,k}+ca_{j,k}\\ \vdots\\ a_{m,k}\end{pmatrix}=\begin{pmatrix}a_{1,k}\\ \vdots\\ \sqrt{a_{i,k}^2+a_{j,k}^2}\\ \vdots\\ 0\\ \vdots\\ a_{m,k}\end{pmatrix},$$

因此吉文斯旋转能够消去指定元素。用吉文斯旋转计算 QR 分解相当于由左至右-由下至上地消去对角线下方元素，最终得到上三角矩阵。吉文斯旋转的一个重要应用是海森伯格(Hessenberg)矩阵三角化。在上海森伯格矩阵中，所有次对角线以下元素都为零，因此对每一列只需计算一次吉文斯旋转即可，n 步之后便得到上三角矩阵。

1.2 舍入误差

依照来源进行分类，可将误差大致分为4类，分别是模型误差、数据误差、截断误差和舍入误差。模型误差是数学模型与实际问题之间的误差；数据误差是输入数据与真实数据之间的误差，也称观测误差；截断误差是数值方法的近似解与数学模型的精确解之间的误差，也称方法误差；舍入误差是由有限精度运算所造成的误差。向量组正交化问题本身已是数值代数问题，QR分解所得的就是精确解，因此不存在截断误差；而模型误差和数据误差是数值算法研究人员无法独立解决的。本书只关注舍入误差。

令 x 为准确值，$\hat{x}$ 为 x 的近似值，绝对误差 e 被定义为

$$e=\hat{x}-x,$$

而相对误差 e_r 被定义为

$$e_r=\frac{\hat{x}-x}{x}。$$

如果误差的正负不重要，则 e 和 e_r 的表达式可以加上绝对值。误差的上界称作误差限 ε，满足

$$|\hat{x}-x|\leqslant\varepsilon,$$

相对误差限 ε_r 满足

$$\frac{|\hat{x}-x|}{|x|}\leqslant\varepsilon_r。$$

由于 x 一般是未知的，相对误差分母中的 x 常用 $\hat{x}$ 替代。

大部分十进制数不能用二进制浮点数精确表示，令 $\mathrm{fl}(x)$ 表示 x 在计算机中的存储值，将机器精度定义为1.0和大于1.0的最小浮点数的距离，记作 ε_m。令 ε_u 为单位舍入误差(unit roundoff)，其值为 $(1/2)\varepsilon_m$。若 x 位于浮点数的表示范围内，则 $\mathrm{fl}(x)$ 的相对误差不超过 ε_u。

引理1.2 若 x 在浮点数表示范围内，则

$$\mathrm{fl}(x)=x(1+\delta),\quad |\delta|\leqslant\varepsilon_u。$$

证明：根据 IEEE 754 标准，浮点数可用二进制表示为

$$\mathrm{fl}(x)=\pm m\cdot 2^{e-t}, \tag{1.4}$$

其中 t 是精度，$e\in[e_{\min},e_{\max}]$ 称为指数或阶数，m 是位于 $[2^{t-1},2^{t}-1]$ 之间的整数。不妨设 $x>0$。当 $m=2^{t-1}$，$e=1$ 时，$\mathrm{fl}(x)=1.0$。因此，

$$\varepsilon_{\mathrm{m}}=(2^{t-1}+1)\cdot 2^{1-t}-1=2^{1-t}。$$

令 $x=\mu\cdot 2^{e-t}$，其中 $\mu\in[2^{t-1},2^{t}-1]$。于是有 $\mathrm{fl}(x)=\lfloor\mu\rfloor\cdot 2^{e-t}$ 或 $\mathrm{fl}(x)=\lceil\mu\rceil\cdot 2^{e-t}$，其中 $\lfloor\cdot\rfloor$ 和 $\lceil\cdot\rceil$ 分别表示向下、向上取整，则

$$\frac{|x-\mathrm{fl}(x)|}{|x|}\leqslant\frac{\frac{1}{2}2^{e-t}}{\mu\cdot 2^{e-t}}\leqslant\frac{1}{2}\cdot 2^{1-t}=\varepsilon_{\mathrm{u}}。$$

因此 ε_{u} 是 $\mathrm{fl}(x)$ 的相对误差限。证毕。

若 x 不在浮点数的表示范围内，则会造成下溢(underflow)或上溢(overflow)。二进制浮点数还可表示为

$$\mathrm{fl}(x)=\pm .d_1d_2\cdots d_t\cdot 2^{e}, \tag{1.5}$$

其中 $d_1,d_2,\cdots,d_t$ 为 0 或 1，称作尾数，其位数又称精度。在教科书中式(1.5)比式(1.4)更常见，两个公式都以 2 为底数。显然，也可选整数 β 作为底数，由此可得 β 进制浮点数系统，但最常用的还是二进制。在 1985 年制定的 IEEE 754 标准中，浮点数由二进制表示，该标准定义了 32 位的单精度浮点数(fp32)和 64 位的双精度浮点数(fp64)，2008 年的修订版增加了 16 位的半精度浮点数(fp16)和 128 位的四倍精度浮点数(fp128)。这些浮点数的主要参数总结在表 1-1 中。IEEE 标准还定义了扩展格式，这里不做讨论。

表 1-1　IEEE 754-2008 标准

浮点数类型	精度	指数位数	ε_{u}	$e_{\min}$	$e_{\max}$	最大数
fp16	11	5	4.88×10^{-4}	−14	15	65 504
fp32	24	8	5.96×10^{-8}	−126	127	3.40×10^{38}
fp64	53	11	1.11×10^{-16}	−1 022	1 023	1.80×10^{308}
fp128	113	15	9.63×10^{-35}	−16 382	16 383	1.19×10^{4932}

若浮点运算的结果不能精确表示，则会产生误差，该误差称为舍入误差。一般将十进制转换成二进制时产生的误差也归为舍入误差。浮点运算的舍入误差一般

采用下述模型：

$$\mathrm{fl}(x\odot y)=(x\odot y)(1+\delta),\quad |\delta|\leqslant\varepsilon_{\mathrm{u}},\tag{1.6}$$

其中$\odot$表示加、减、乘、除任意一种运算。由于ε_{u}很小，一般认为上述浮点运算足够精确。根据IEEE标准，平方根运算也满足式(1.6)的相对误差限。另一个常用模型为：

$$\mathrm{fl}(x\odot y)=\frac{x\odot y}{1+\delta},\quad |\delta|\leqslant\varepsilon_{\mathrm{u}}。$$

所有实现了IEEE标准的计算机都满足上述模型。

舍入误差在运算过程中会不断积累。令$\boldsymbol{x}$和$\boldsymbol{y}$为3维实向量，即$\boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^3$。若以点积运算为例，则准确值为$\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}=x_1y_1+x_2y_2+x_3y_3$。设浮点运算从左向右进行，且每个运算的相对误差都为δ，则近似值为

$$\begin{aligned}\mathrm{fl}(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y})&=\mathrm{fl}(\mathrm{fl}(\mathrm{fl}(x_1y_1)+\mathrm{fl}(x_2y_2))+\mathrm{fl}(x_3y_3))\\&=\mathrm{fl}(\mathrm{fl}(x_1y_1(1+\delta)+x_2y_2(1+\delta))+\mathrm{fl}(x_3y_3))\\&=\mathrm{fl}((x_1y_1+x_2y_2)(1+\delta)^2+x_3y_3(1+\delta))\\&=((x_1y_2+x_2y_2)(1+\delta)^2+x_3y_3(1+\delta))(1+\delta)\\&=x_1y_1(1+\delta)^3+x_2y_2(1+\delta)^3+x_3y_3(1+\delta)^2。\end{aligned}$$

依此类推，当$\boldsymbol{x}$和$\boldsymbol{y}$为n维向量时，计算结果为

$$\mathrm{fl}(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y})=x_1y_1(1+\delta)^n+x_2y_2(1+\delta)^n+x_3y_3(1+\delta)^{n-1}+\cdots+x_ny_n(1+\delta)^2。$$

令

$$\gamma_n=\frac{n\varepsilon_{\mathrm{u}}}{1-n\varepsilon_{\mathrm{u}}},\tag{1.7}$$

下面的结论有助于简化浮点运算的误差分析。

引理1.3 若对任意$i=1,2,\cdots,n$，有$|\delta_i|\leqslant\varepsilon_{\mathrm{u}}$，$\rho_i=\pm1$，且$n\varepsilon_{\mathrm{u}}<1$，则

$$\prod_{i=1}^{n}(1+\delta_i)^{\rho_i}=1+\theta_n,\quad |\theta_n|\leqslant\gamma_n。\tag{1.8}$$

证明：当$\rho_n=1$时，有

$$\prod_{i=1}^{n}(1+\delta_i)^{\rho_i}=(1+\delta_n)\prod_{i=1}^{n-1}(1+\delta_i)^{\rho_i}=(1+\delta_n)(1+\theta_{n-1})。$$

故

$$\theta_n=\delta_n+(1+\delta_n)\theta_{n-1}。$$

用数学归纳法，可得

$$|\theta_n| \leqslant \varepsilon_u + (1+\varepsilon_u)\frac{(n-1)\varepsilon_u}{1-(n-1)\varepsilon_u}$$

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

$$= \frac{n\varepsilon_u}{1-(n-1)\varepsilon_u} \leqslant \gamma_n 。$$

同理，当 $\rho_n=-1$ 时也可得到类似结论。证毕。

在引理 1.3 的假设下，可得

$$\mathrm{fl}(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}) = x_1y_1(1+\theta_n)+x_2y_2(1+\theta_n')+x_3y_3(1+\theta_{n-1})+\cdots+x_ny_n(1+\theta_2)$$

$$= \boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}+x_1y_1\theta_n+x_2y_2\theta_n'+x_3y_3\theta_{n-1}+\cdots+x_ny_n\theta_2 。$$

观察 $\mu=x_iy_i(1+\theta_{n+2-i})$，$i=3,\cdots,n$，可以看到

$$\left|\frac{\mu-x_iy_i}{x_iy_i}\right| = |\theta_{n+2-i}| \leqslant \gamma_{n+2-i} \leqslant \gamma_n ,$$

即 $\mu=x_iy_i(1+\theta_n)$，从而有 $\mathrm{fl}(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y})=\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}(1+\theta_n)$。也可直接计算

$$|\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}-\mathrm{fl}(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y})| \leqslant |x_1y_1|\gamma_n+|x_2y_2|\gamma_n+|x_3y_3|\gamma_{n-1}+\cdots+|x_ny_n|\gamma_2$$

$$\leqslant \gamma_n\sum_{i=1}^{n}|x_iy_i| = \gamma_n|\boldsymbol{x}|^{\mathrm{T}}|\boldsymbol{y}| ,$$

其中 $|\boldsymbol{x}|$ 表示对向量 $\boldsymbol{x}$ 逐项取绝对值。

1.3 前向误差与后向误差

前向误差(forward error)与后向误差(backward error)用来描述计算结果的好坏。令 $y=f(x)$，其中 x 是实数。受舍入误差影响，计算结果为 $\hat{y}$。前向误差就是 1.2 节中定义的误差，是准确函数值与近似函数值的差，后文将其定义为误差的绝对值

$$e_f = \frac{|y-\hat{y}|}{|y|} 。$$

前向误差将 $\hat{y}$ 看作“准确问题的近似解”，而后向误差将其看作“近似问题的准确解”，即引入 Δx 使得 $\hat{y}=f(x+\Delta x)$，并将其最小值定义为后向误差

$$e_b=\min\left\{\frac{|\Delta x|}{|x|}:\hat{y}=f(x+\Delta x)\right\}。$$

后向误差小意味着为准确得到 $\hat{y}$ 所需要的扰动小。误差分析即给出 e_f 或 e_b 的上界 ε_f、ε_b。前向误差分析最直观，但有时无法得到，且控制前向误差有时意义不大。后向误差分析是一个重要工具，目的是将舍入误差看作数据的扰动，并给出扰动的上界。如果后向误差在量级上等同于输入数据的不确定性或单位舍入误差，那么计算结果显然是可以接受的。有些算法的后向误差无法求得，那么可以考虑混合前向-后向误差分析，即

$$\hat{y}+\Delta y=f(x+\Delta x),\quad \frac{|\Delta y|}{|y|}\leqslant\varepsilon_f,\quad \frac{|\Delta x|}{|x|}\leqslant\varepsilon_b。$$

后向误差小的算法被称作是后向稳定的。以向量点积运算为例。令 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 为 n 维向量，由 1.2 节的分析可知

$$\mathrm{fl}(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y})=(\boldsymbol{x}+\Delta\boldsymbol{x})^{\mathrm{T}}\boldsymbol{y},\quad |\Delta\boldsymbol{x}|\leqslant\gamma_n|\boldsymbol{x}|。\tag{1.9}$$

已知 $|\gamma_n|$ 很小，因此认为向量点积运算是后向稳定的。这里假设的是 $\boldsymbol{x}$ 受到扰动，若 $\boldsymbol{y}$ 受到扰动也可得到相同结果。相应地，前向误差结果已在 1.2 节中给出，即

$$|\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}-\mathrm{fl}(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y})|\leqslant\gamma_n|\boldsymbol{x}|^{\mathrm{T}}|\boldsymbol{y}|,$$

该结果只有当 $|\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}|$ 与 $|\boldsymbol{x}|^{\mathrm{T}}|\boldsymbol{y}|$ 相差不大时才表明前向误差足够小。

前向误差与后向误差可通过条件数(condition number)来建立联系。前向误差可看作后向误差经问题放大后的结果，而条件数则是放大系数，用来描述问题本身的好坏，与算法无关。给定标量函数 $y=f(x)$，条件数定义为

$$\mathrm{cond}(x)=\lim_{\varepsilon\to0^+}\sup_{|\Delta x|\leqslant|x|}\frac{|f(x+\Delta x)-f(x)|}{\varepsilon|f(x)|},$$

或采用更简洁的定义

$$\mathrm{cond}(x)=\left|\frac{xf'(x)}{f(x)}\right|,$$

其中 $f'(x)$ 为导数。若给定方阵 $\boldsymbol{A}$，将求 $\boldsymbol{A}^{-1}$ 看作问题，则条件数是 $\kappa(\boldsymbol{A})=\|\boldsymbol{A}\|\,\|\boldsymbol{A}^{-1}\|$。可知 $\kappa(\boldsymbol{A})\geqslant1$。当 $\kappa(\boldsymbol{A})$ 趋于无穷时，$\boldsymbol{A}$ 是奇异矩阵。一般将 $\kappa(\boldsymbol{A})$ 简称为矩阵 $\boldsymbol{A}$ 的条件数，此外 $\kappa(\boldsymbol{A})$ 也是线性方程组 $\boldsymbol{Ax}=\boldsymbol{b}$ 的条件数。前向误差、后向误差、条件数的关系是

$$e_f\lesssim\mathrm{cond}(x)e_b,\tag{1.10}$$

其中≲表明不等关系在 $\Delta x \to 0$ 时成立。由于 Δx 足够小，通常可认为式(1.10)的不等关系成立。

这里使用2-范数，但也可采用其他范数。在本书中，2-范数是最常用到的诱导范数，矩阵的2-范数等于矩阵的最大奇异值；另一个常用到的范数是F-范数

$$\| \boldsymbol{A} \|_{\mathrm{F}} = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} a_{i,j}^2 \right)^{\frac{1}{2}},$$

也称作Frobenius范数或欧式范数。2-范数具有一致性和正交不变性，即 $\| \boldsymbol{AB} \| \leqslant \| \boldsymbol{A} \| \, \| \boldsymbol{B} \|$，$\| \boldsymbol{UAV} \| = \| \boldsymbol{A} \|$，其中 $\boldsymbol{U}$ 和 $\boldsymbol{V}$ 是正交矩阵。易证明，F-范数同样具有这两个性质。此外，$|\boldsymbol{A}|$ 表示对矩阵 $\boldsymbol{A}$ 逐项取绝对值。若将上述误差分析中的标量换成向量，可将绝对值符号换成范数符号，也可逐项取绝对值；对于矩阵范数，也可选择逐列进行误差分析。后文将在具体问题中进行讨论。

若一个问题的条件数很大，则称该问题为病态问题，即输入数据的小扰动会令输出数据产生大误差。由式(1.10)还可看到，后向稳定的算法一定前向稳定，反之则不一定。如果后向误差上界为 ε_{b} 的算法是后向稳定的，那么前向误差上界为 $\mathrm{cond}(x)\varepsilon_{\mathrm{b}}$ 的算法就是前向稳定的。

通常认为后向误差分析是由威尔金森(J. H. Wilkinson)于20世纪50年代提出的，但他本人将其归功于冯·诺依曼(J. von Neumann)和戈德斯坦(H. Goldsteine)于1947年发表的一篇文章。威尔金森为数值分析的发展做出了重要的贡献，1970年获得图灵奖。