计算物理讲义

冯旭

1 数值计算的基础

2 线性方程组的直接解法

本章我们将主要讨论求解线性方程组的数值方法中的比较“简单”的几类。线性方程组的求解在物理学的众多数值应用中都会遇到。因此, 这是我们这个课程的一个基础。后面的许多章都会涉及。由于这是一个非常大的一个课题, 本章中我们将仅仅涉及其中的一些基础的内容, 主要包括: 高斯消元法、LU 分解、Cholesky 分解等。更为复杂的内容 (比如奇异值分解等) 将放在后面的第六章讨论。

2.1 线性代数知识的回顾

本节中我们需要回顾一下线性代数中最为基本的知识。这些知识对于我们理解本章以及后面各章中的基本算法是十分重要的。我们仅仅回顾那些与我们数值计算密切相关的知识, 这包括对矩阵的分类, 以及一些重要的定理, 我们会对这些定理加以说明。另外, 由于舍入误差的存在, 我们往往需要对相关的算法的稳定性进行估计。这时候就需要对矢量以及矩阵的模进行讨论。这部分内容往往在通常的线性代数的课程中是没有的。我们在本节中也会介绍。由于我们后面 (参见第六章) 还会讨论矩阵的对角化以及本征值的问题, 因此有些线性代数的知识 (特别是与本征值直接相关的知识) 将会在那里进行回顾。首先是一些符号。一个数域 K 上面的 n 行 m 列的矩阵 A 我们一般记为：$A \in K^{n\times m}$, 其中数域 K 最为常见的情形是复数域 C 和实数域 R。矩阵可以视为两个矢量空间 K^n 和 K^m 之间的一个线性映射。

2.1.1 矩阵的迹和行列式

当 $n = m$, 我们称该矩阵为一个方阵。对于方阵我们可以定义它的迹和行列式。一个矩阵的迹就是该矩阵对角元之和：

$$\mathrm{Tr}(A) = \sum_{i=1}^{n} a_{ii} \tag{1}$$

矩阵的行列式则在求解线性方程中具有重要的意义。行列式可以表达为

$$\det(A) = \sum_{j=1}^{n} \Delta_{ij} a_{ij} \tag{2}$$

其中 $i \in [1, n]$ 是任意一个行指标而 Δ_{ij} 是矩阵元 a_{ij} 的代数余子式。事实上, 矩阵 A 的逆矩阵可以表达为

$$A^{-1} = \frac{1}{\det(A)}\Delta \tag{3}$$

其中 Δ 是以 Δ_{ij} 为矩阵元的矩阵。由此我们看到，一个方阵的逆矩阵存在的充分必要条件是它的行列式不为零。

从上面的公式出发我们可以很容易获得求解线性方程 $Ax = b$ 的方法，

$$x_j = \Delta_j / \det(A), \quad j = 1, \cdots, n \tag{4}$$

其中 Δ_j 是将原矩阵中的第 j 列换为矢量 b 所得到的矩阵之行列式的值。这个法则一般称为克莱默法则。

但是，克莱默法则的公式并不能直接用于数值计算。因为按照这些公式，行列式的计算涉及到 $O(n!)$ 的计算量。这对于即使不太大的矩阵来说也过于庞大了。例如，即使对于大约 100 阶的矩阵来说 (或者说求解 100 个联立的线性方程组)，按照这些公式计算的话在我们有生之年都不太可能算出结果。为了获得数值上近似的解，我们需要更聪明的计算方法。

2.1.2 矩阵的秩和核

矩阵 $A \in K^{m\times n}$ 的秩 –记为 rank(A)–可以定义为从矩阵 A 中能够抽取的非奇异的子矩阵 (行列式不为零) 的最大的阶数。当 A 被视为 $K^n \to K^m$ 的线性映射时，我们可以定义其值域为:

$$\text{range}(A) = \{y \in K^m : \, y = A \cdot x, \quad x \in K^n\} \tag{5}$$

而矩阵的秩也可以定义为其值域空间的维数：$\text{rank}(A) = \dim(\text{range}(A))$。另一个重要的概念是矢量的线性相关。一个矩阵按照行的秩与其按照列的秩定义为线性无关的矢量的数目。严格来说，我们需要区分矩阵按照行的秩和按照列的秩。但是线性代数的基础知识告诉我们这两个是一致的。线性映射 A 的核定义为：

$$\ker(A) = \{x \in K^n : \, A \cdot x = 0\} \tag{6}$$

它其实是满足 $Ax = 0$ 的矢量构成的子空间。

那么下列关系是成立的：

- rank(A)=rank(A^T)
- rank(A)+dim(ker(A))=n

2.1.3 矢量与矩阵的模

下面我们来讨论矢量与矩阵的模。数学中可以对一般的模进行定义。一个矢量空间 V 上的模 $||\cdot||$ 一般来说可以定义为满足下列条件的非负函数:

- 非负性：$||\vec{v}|| \geq 0, \forall \vec{v} \in V$ 且 $||\vec{v}|| = 0$ 当且仅当 $\vec{v} = 0$。
- 均匀性：$||\alpha\vec{v}|| = |\alpha| \cdot ||\vec{v}||; \forall \alpha \in K, \forall \vec{v} \in V$
- 三角不等式：$||\vec{v} + \vec{w}|| \leq ||\vec{v}|| + ||\vec{w}||, \forall \vec{v}, \vec{w} \in V$

一个常用的模是所谓的 p-模，又称为 Hölder 模，它由下式定义：

$$||\vec{x}||_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}, \forall \vec{x} \in V, 1 \leq p < \infty \tag{7}$$

对于 p-模，如果我们取极限 $p \to \infty$，就得到了无穷模，它实际上仅仅挑选出矢量 $\vec{x}$ 的分量中模最大的那个：

$$||\vec{x}||_\infty = \max_{1\le i\le n} |x_i| \tag{8}$$

另外一个经常用到的是 $p = 2$ 的情形。对于我们讨论的实空间和复空间来说，这个模称为相应空间的欧氏模

$$||\vec{x}||_2 = (x, x)^{1/2} = (x^\dagger x)^{1/2} = \left(\sum_{i=1}^{n} |x_i|^2\right)^{1/2} \tag{9}$$

既然矢量空间中模的定义可以有很多种，一个自然的问题是它们是否都等价呢? 按照定义，矢量空间 V 上的两个模 $||\cdot||_p$ 和 $||\cdot||_q$ 被称为等价，如果存在两个正的与 $\vec{x}$ 无关的常数 $c > 0$ 和 $C > 0$ 使得：

$$c||\vec{x}||_q \le ||\vec{x}||_p \le C||\vec{x}||_q, \quad \forall \vec{x} \in V \tag{10}$$

也就是说其中对于任意的矢量，其中一个模一定夹在另外一个模的两个正常数倍数之间。相应的这些常数 c 和 C 被称为等价常数。可以证明前面给出的三种不同的 p-模 $(p = 1, 2, \infty)$ 都是等价的。我们下面来一个简单粗暴的证明

$$||\vec{x}||_p = ||\vec{x}||_\infty \tag{11}$$

引入一个" 单位圆"，记为 $S = \{\vec{x}|\, ||\vec{x}||_\infty = 1\}$，明显 S 是个有界闭集。由于 $f(x) = ||\vec{x}||_p$ 是 S 上的连续函数，$f(x)$ 在 S 上可以取到最小值和最大值。设最小值为 c，最大值为 C。那么 $c \le f(x) \le C$。另一方面，$\forall \vec{x} \in K^n$ 且 $\vec{x} \ne 0$，则 $\vec{x}/||\vec{x}||_\infty \in S$，则有

$$c \le \left|\left|\frac{\vec{x}}{||\vec{x}||_\infty}\right|\right|_p \le C \quad \Rightarrow \quad c \le \frac{||\vec{x}||_p}{||\vec{x}||_\infty} \le C \tag{12}$$

事实上，有限维矢量空间中的任何模都是等价的。

我们花了一些时间来回顾矢量的模的概念，这些概念同学们在线性代数课上应该学到过。之所以要强调这个概念，我们不妨来看一下解线性方程的问题

$$\begin{aligned} A\vec{x} = \vec{b} \quad &\xrightarrow{\text{数学上}} \quad \vec{x} = A^{-1}\vec{b} \\ A\vec{x} = \vec{b} \quad &\xrightarrow{\text{计算物理中}} \quad \min ||A\vec{x} - \vec{b}|| \end{aligned} \tag{13}$$

定义了矢量空间上的模之后就可以随之定义矩阵的模。对于 $K^{m\times n}$ 上的矩阵，它的模 $||\cdot||$ 定义为：

- 非负性：$||A|| \ge 0, \forall A \in K^{m\times n}$ 且 $||A|| = 0$ 当且仅当 $A = 0$(所有矩阵元为 0)
- 均匀性：$||\alpha A|| = |\alpha| \cdot ||A||; \forall \alpha \in K, \ \forall A \in K^{m\times n}$
- 三角不等式：$||A + B|| \le ||A|| + ||B||, \forall A, B \in K^{m\times n}$

如果矩阵的模和矢量的模满足

$$||A\vec{x}|| \le ||A|| \cdot ||\vec{x}||, \forall \vec{x} \in K^n, \forall A \in K^{m\times n} \tag{14}$$

我们就称相应的矩阵模与矢量模兼容。另一方面，一个矩阵模 $||\cdot||$ 被称为服从乘法模，如果它满足

$$||AB|| \le ||A|| \cdot ||B||, \forall A \in K^{n\times m}, \forall B \in K^{m\times q} \tag{15}$$

值得指出的是，并不是所有的矩阵模都是服从乘法的模。一个简单的例子是所谓的最大模，其定义为 $||A||_\Delta = \max(|a_{ij}|)$。我们可以验证它满足矩阵模的所有条件因而构成一个矩阵模。但是对于下面的矩阵：

$$A = B = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \tag{16}$$

我们可以很容易验证 $||AB||_\Delta = 2 > ||A||_\Delta ||B||_\Delta = 1$。它不是一个服从乘法模。

从一个矢量空间的模出发，我们可以定义一个矩阵模如下:

$$||A|| = \sup_{\vec{x}\neq 0} \frac{||A\vec{x}||}{||\vec{x}||} \tag{17}$$

这称为诱导矩阵模或自然矩阵模。这里的 sup 表示集合的上确界，首先它代表了集合的上界 (upper bound)，而且这个上界确实可以取到的。

诱导矩阵模有几个性质：

- 可以证明诱导矩阵模是与诱导它的矢量模兼容的，同时也是服从乘法的。
- 存在一个非 0 矢量 $\vec{y}$，使得 $||A\vec{y}|| = ||A|| \cdot ||\vec{y}||$
  我们可以看到，在诱导矩阵模的定义里，$\vec{x} \to \alpha\vec{x}$ 是不改变模的定义的。于是我们可以把诱导矩阵模写成

  $$||A|| = \sup_{||\vec{x}||=1} ||A\vec{x}|| \tag{18}$$

  我们一定可以找到一个矢量 $\vec{y}$，这里 $\vec{y}$ 满足 $||\vec{y}|| = 1$，并且使得

  $$\sup_{||\vec{x}||=1} ||A\vec{x}|| = ||A\vec{y}|| \tag{19}$$

  于是我们就是证明了

  $$||A|| \cdot ||\vec{y}|| = ||A\vec{y}|| \tag{20}$$

对于由矢量的 p-模所诱导的矩阵模，我们也会用同样的符号来标记，例如

$$||A||_p = \sup_{\vec{x}\neq 0} \frac{||A\vec{x}||_p}{||\vec{x}||_p}, \quad \forall \vec{x} \in V, \quad \vec{x} \neq 0 \tag{21}$$

$p = 1$ 和 $p = \infty$ 的诱导矩阵模其实形式并不复杂

$$||A||_1 = \max_{j=1,\cdots,n} \sum_{i=1}^{m} |a_{ij}|, \quad ||A||_\infty = \max_{i=1,\cdots,m} \sum_{j=1}^{n} |a_{ij}| \tag{22}$$

一个是对列向量的元素进行求和的模，一个是对行向量的元素进行求和的模。所以，我们有 $||A||_1 = ||A^T||_\infty$。

下面，我们主要考察 $p = 2$ 的诱导矩阵模，它有一些比较特殊的性质。我们比较方便地引入方阵的谱半径 (spectral radius) 的概念，定义为

$$\rho(A) := \max_{1\leq i\leq n} |\lambda_i| \tag{23}$$

这里 λ_i 是矩阵的本征值。那么一个矩阵的诱导矩阵模一定大于等于该矩阵的谱半径

$$||A|| \geq \rho(A) \tag{24}$$

这是因为，对于 $\forall\lambda$，$A\vec{x}=\lambda\vec{x}$，我们一定有

$$|\lambda|\cdot||\vec{x}||=||\lambda\vec{x}||=||A\vec{x}||\leq||A||\cdot||\vec{x}|| \tag{25}$$

如果 A 恰好是个厄米矩阵，那么 $||A||=\rho(A)$。
Proof: 这一点我们可以这么来看，对于任意方阵 B，我们应该有

$$||B||=\left[\rho(B^{\dagger}B)\right]^{\frac{1}{2}} \tag{26}$$

然后观察

$$\frac{||B\vec{x}||}{||\vec{x}||}=\frac{(\vec{x}^{\dagger}B^{\dagger}B\vec{x})^{\frac{1}{2}}}{(\vec{x}^{\dagger}\vec{x})^{\frac{1}{2}}}=\left(\frac{\sum_{i=1}^{n}|c_i|^2v_i}{\sum_{j=1}^{n}|c_j|^2}\right)^{\frac{1}{2}} \tag{27}$$

这里，我们把矢量 $\vec{x}$ 写成了 $\vec{x}=\sum_{i=1}^{n}c_i\vec{\alpha}_i$ 的形式。这里 $\vec{\alpha}_i$ 是 $B^{\dagger}B$ 的正交本征矢。v_i 是与 α_i 对应的本征值，其中最大值取为 v_n。明显

$$\frac{||B\vec{x}||}{||\vec{x}||}\leq\sqrt{v_n}=\left[\rho(B^{\dagger}B)\right]^{\frac{1}{2}} \tag{28}$$

并且当 $\vec{x}\propto\vec{\alpha}_n$ 时，等号成立。于是有 $||B||=\left[\rho(B^{\dagger}B)\right]^{\frac{1}{2}}$。有了这个条件以后，我们很容易得到对于厄米矩阵 A，有

$$||A||^2=\rho(A^{\dagger}A)=\rho(A^2)=\rho^2(A) \tag{29}$$

所以厄米矩阵的 $p=2$ 诱导矩阵模就等于该矩阵最大的本征值。但是非厄米矩阵不具备这样的性质。

2.1.4 问题的敏感性和矩阵条件数

有了向量和矩阵范数的概念，下面分析线性方程组求解问题的敏感性和条件数，研究初始数据误差对解的影响。我们考虑方程组

$$A\vec{x}=\vec{b} \tag{30}$$

这里 A 是个矩阵，$\vec{b}$ 和 $\vec{x}$ 都是向量。我们考虑方程右端发生扰动 $\Delta\vec{b}$ 的情况，相应的解变为 $\vec{x}+\Delta\vec{x}$，则

$$A(\vec{x}+\Delta\vec{x})=\vec{b}+\Delta\vec{b} \tag{31}$$

可推导出如下两个不等式

$$\begin{aligned}&A\Delta\vec{x}=\Delta\vec{b}\quad\Rightarrow\quad\Delta\vec{x}=A^{-1}\Delta\vec{b}\quad\Rightarrow\quad||\Delta\vec{x}||\leq||A^{-1}||\cdot||\Delta\vec{b}||\\&A\vec{x}=\vec{b}\quad\Rightarrow\quad||\vec{b}||\leq||A||\cdot||\vec{x}||\end{aligned} \tag{32}$$

根据问题条件数的定义，得到

$$\text{cond}=\frac{||\Delta\vec{x}||/||\vec{x}||}{||\Delta\vec{b}||/||\vec{b}||}=\frac{||\Delta\vec{x}||\cdot||\vec{b}||}{||\Delta\vec{b}||\cdot||\vec{x}||}\leq||A||\cdot||A^{-1}|| \tag{33}$$

如果 A 是个厄米矩阵，而我们用的范数是 $p=2$ 的欧氏模，那么

$$\text{cond}=\rho(A)\cdot\rho(A^{-1})=|\lambda|_{\max}/|\lambda|_{\min} \tag{34}$$

所以一个厄米矩阵最大本征值和最小本征值的比值决定了问题本身的病态程度。如果问题非常病态, 那么，要求得一个精确解是非常困难的。这个时候，我们就需要采取一些手段，把病态问题变成良态问题。

我们知道，本征值的数目与矩阵维数一致。一般来讲，一个超大矩阵 ($n \sim 10^6$)，这 10^6 个本征值的分布会出现下疏上密的情形。如果我们能计算出最小的若干个本征值和本征矢量，比方说前 100 个，然后把它们从原有系统中刨去，那么，就能使 $|\lambda|_{\min}$ 大大增加，从而使得条件数大大减小。

事实上我们可以这样来玩。把矩阵 A 用正交归一本征矢量表示出来

$$A = \sum_{\lambda} \lambda |\lambda\rangle\langle\lambda|, \quad 1 = \sum_{\lambda} |\lambda\rangle\langle\lambda| \tag{35}$$

假设我们已知 $|\lambda| \leq |\lambda_0|$ 时的所有本征值和本征矢量，并且 $|\lambda_0| \gg |\lambda|_{\min}$。我们可以构造投影算符

$$P_0 = \sum_{|\lambda|<|\lambda_0|} |\lambda\rangle\langle\lambda|, \quad 1 - P_0 = \sum_{|\lambda|>|\lambda_0|} |\lambda\rangle\langle\lambda| \tag{36}$$

然后我们把解方程组的事分成两步来完成

$$A\vec{x}_H = (1 - P_0)\vec{b}, \quad A\vec{x}_L = P_0\vec{b} \tag{37}$$

把 $\vec{x}_H$ 和 $\vec{x}_L$ 加起来就得到了方程组的解。对于 $\vec{x}_L$ 来讲，因为我们已经知道了最小的前 n 个本征值, 所以我们可以直接得到

$$\vec{x}_L = A^{-1}P_0\vec{b} = \sum_{|\lambda|<|\lambda_0|} \lambda^{-1}|\lambda\rangle\langle\lambda|b\rangle \tag{38}$$

对于 $\vec{x}_H$ 来讲，我们需要解

$$A\vec{x}_H = (1 - P_0)\vec{b}, \quad \Rightarrow \quad A\Delta\vec{x}_H = (1 - P_0)\Delta\vec{b}, \quad \Rightarrow \quad ||\Delta\vec{x}_H|| \leq ||A^{-1}(1 - P_0)|| \cdot ||\Delta\vec{b}|| \leq |\lambda_0|^{-1}||\Delta\vec{b}|| \tag{39}$$

这样条件数就变成了

$$\text{cond} = |\lambda|_{\max}/|\lambda_0| \tag{40}$$

这种方法叫做 low mode deflation (低模式紧缩)。所以线性方程组求解，或者矩阵求逆问题经常会与本征值和本征矢量的求解联系起来。我们在第 6 章会详细讲解本征值问题。

2.1.5 一些特殊形状的矩阵

本小节我们罗列在数值计算中经常接触到的一些特殊的矩阵。

对角矩阵是指仅仅对角元 a_{ii} 不为零的矩阵。通常意义下是指方阵，但是此定义也适用于长方阵。一个矩阵 $A \in K^{m\times n}$，如果对 $i > j$ 就有 $a_{ij} = 0$，我们就称矩阵 A 为上梯形矩阵。相应的，如果对 $i < j$ 就有 $a_{ij} = 0$，我们就称矩阵 A 为下梯形矩阵。大家可以验证，如果 $m < n$ 的话，上梯形矩阵的非零矩阵元恰好构成一个梯形。

$$\begin{pmatrix} x & x & x & x \\ & x & x & x \\ & & x & x \end{pmatrix}_{3\times 4} \tag{41}$$

对于 $m = n$ 的方阵而言，上/下梯形矩阵分别称为上/下三角矩阵。

上下三角矩阵的一些性质是容易验证的。它的行列式就是对角元的乘积。而且它的逆矩阵仍然维持原矩阵的上下三角的性质。如果上/下三角矩阵的对角元都等于 1，这样的上/下三角矩阵称为单位上/下三角矩阵。容易验证，两个单位上下三角矩阵的乘积仍然是单位上下三角矩阵。

三角矩阵的概念可以稍加推广到所谓的带型矩阵。一般来说, 对于 $A \in K^{m\times n}$, 我们称其具有上带 p，如果对 $i > j + p$ 必定有 $a_{ij} = 0$; 相应的，我们称其具有下带 q 如果对于 $j > i + q$ 必定有 $a_{ij} = 0$。

$$\begin{pmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ & x & x & x & x & x \\ & & x & x & x & x \\ & & & x & x & x \end{pmatrix}_{i>j+1}, \quad \begin{pmatrix} x & x & & & \\ x & x & x & & \\ & x & x & x & \\ & & x & x & x \\ & & & x & x & x \end{pmatrix}_{p,q=1} \tag{42}$$

利用这个概念我们可以统一上面提及的几种矩阵。例如，对角矩阵是 $p = q = 0$ 的带型矩阵; 下梯形矩阵是具有 $p = m - 1, q = 0$ 的带型矩阵; 上梯形矩阵则是具有 $p = 0, q = n - 1$ 的带型矩阵。如果带型矩阵的 $p = q = 1$，则该带状矩阵称为三对角矩阵。另外两种情形是上双对角 (p = 0,q = 1) 和下双对角 (p = 1,q = 0) 矩阵。另外一类我们后面会用到的矩阵是所谓的上/下 Hessenberg 矩阵。下 Hessenberg 矩阵具有 $p = m - 1, q = 1$ 而上 Hessenberg 矩阵则具有 $p = 1, q = n - 1$。

2.1.6 二次型与正定矩阵

另外一类重要的矩阵称为正定矩阵。它们与相应的二次型密切联系在一起。二次型起源于矢量空间中的标量积运算。矢量空间 V 中的标量积可以视为 $V \times V$ 到 K 的一个映射 $(\cdot, \cdot)$，它满足：

- 双线性：$(\alpha\vec{x} + \beta\vec{y}, \vec{z}) = \alpha(\vec{x}, \vec{z}) + \beta(\vec{y}, \vec{z})$, $\forall \vec{x}, \vec{y}, \vec{z} \in V$, $\forall \alpha, \beta \in K$
- 厄米性：$(\vec{x}, \vec{y}) = (\vec{y}, \vec{x})^*$, $\forall \vec{x}, \vec{y} \in V$
- 正定性：$(\vec{x}, \vec{x}) > 0$, $\forall \vec{x} \in V$ 除非 $\vec{x} = 0$

对于空间 C^n 来说，我们可以定义内积为

$$(\vec{x}, \vec{y}) = \vec{y}^\dagger \cdot \vec{x} = \sum_{i=1}^{n} y_i^* x_i \tag{43}$$

我们显然有

$$(A\vec{x}, \vec{y}) = (\vec{x}, A^\dagger \vec{y}) \tag{44}$$

如果对于 $C^{n\times n}$(或 $R^{n\times n}$) 中的矩阵 A 以及任意的非零矢量 $\vec{x} \in V$ 都有 $(A\vec{x}, \vec{x})$ 是正的实数，我们就称矩阵 A 是正定的。如果 $>$ 换为 $\geq$ 且等号有可能成立，我们就称 A 是半正定的。对于 $C^{n\times n}$ 中的复矩阵 A，它是正定的条件要求 A 必定是厄米的 (从而其本征值均为实数) 并且所有本征值都是正的。这个结论的一个推论是，正定的复矩阵必定是不奇异的。最后，让我们提及关于复厄米正定矩阵的下列性质。令 $A \in C^{n\times n}$ 为厄米矩阵。那么它是正定矩阵当且仅当下列等价的条件之一获得满足：

- $(A\vec{x}, \vec{x}) > 0$, $\forall \vec{x} \neq 0$, $\vec{x} \in C^n$
- A 的主子矩阵的本征值都是正的

- A 的主子矩阵的行列式都是正的 (又称 Sylvester 判据)
- 存在一个非奇异矩阵 $H \in C^{n\times n}$ 使得 $A = H^{\dagger}H$

(所谓 n 阶主子矩阵，是指任意取 n 行，再选取相同行号的列，所构成的 $n \times n$ 矩阵。) 正是这最后一个条件使得我们对于正定的厄米矩阵可以采用所谓的 Cholesky(乔利斯基) 分解。事实上，非奇异的 H 不仅仅是存在的，我们还可以将其选为上三角矩阵 (从而 $H^{\dagger}$ 为下三角矩阵)。

2.2 高斯消元法

我们希望求解的线性系统可以统一表达为

$$A\vec{x} = \vec{b} \tag{45}$$

数值上求解这类方程其实与我们在初中时学习的步骤基本一致。这实际上是一个古老的方法，称为高斯消元法 (Gauss Elimination Method, GEM)。我们首先写出这些方程，然后对它们进行一系列的所谓的初等变换 –即将某个方程乘以一定的数然后将其与另一个方程相加减 –其目的是消去某个变量前面的系数。通过这样的变换，将原先的方程化为如下形式的方程:

$$U\vec{x} = \vec{c}, \quad U = \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & & u_{2n} \\ & & \ddots & \\ 0 & & & u_{nn} \end{pmatrix} \tag{46}$$

其中 U 是一个上三角矩阵。如果矩阵 A 是非奇异的，那么上三角矩阵 U 一定也是如此，这意味着所有的对角元 $u_{ii} \neq 0$。因此这个上三角系统可以运用所谓反代的方法来求解，即首先从最后一个方程解出 x_n，反代到倒数第二个方程解出 x_{n-1}; 然后将这两者反代到倒数第三个方程解出 x_{n-2}，等等。具体写出来，就是

$$\begin{aligned} x_n &= \frac{c_n}{u_{nn}} \\ x_{n-1} &= \frac{c_{n-1} - u_{n-1,n}x_n}{u_{n-1,n-1}} \end{aligned} \tag{47}$$

以此类推，用公式来表达就是,

$$x_i = \frac{c_i - \sum_{k=i+1}^{n} u_{ik}x_k}{u_{ii}}, \quad i = n, n-1, \cdots, 1 \tag{48}$$

给出反代部分的伪码，就是

$$\begin{aligned} &\text{For } i = n, n-1, \cdots, 1 \\ &\qquad \text{If } u_{ii} = 0, \text{ then stop} \\ &\qquad x_i := c_i \\ &\qquad \text{For } k = n, n-1, \cdots, i+1 \\ &\qquad\qquad x_i := x_i - u_{ik}x_k \\ &\qquad \text{End} \\ &\qquad x_i := x_i/u_{ii} \\ &\text{End} \end{aligned} \tag{49}$$

这个算法的乘除法次数为 $n+\cdots+2+1=\frac{1}{2}n^2+\frac{1}{2}n$。加减法次数为 $(n-1)+\cdots+2+1=\frac{1}{2}n^2-\frac{1}{2}n$。如果忽略低次项，反代过程中的浮点数乘除法 + 加减法的计算总次数为 n^2。

为了要将线性系统化为上三角的形式，我们将原先的矩阵 A 和 $\vec{b}$ 合并为一个 $n\times(n+1)$ 的矩阵，它称为原来线性系统的增广矩阵，我们将用 $(A,\vec{b})$ 来标记它，

$$(A,\vec{b})=\begin{pmatrix} a_{11} & \cdots & a_{1n} & b_1 \\ \cdots & & \cdots & \cdots \\ a_{n1} & \cdots & a_{nn} & b_n \end{pmatrix} \tag{50}$$

我们随后的线性变换都是直接对增广矩阵 (A,b) 来进行操作的。最终的目的是将其变换为上三角的形式。我们首先去寻找一个 $a_{r1}\neq 0$。对所有的 $1\leq r\leq n$ 来说，这样的矩阵元总是存在的，否则矩阵 A 将是奇异的，与我们的初始假设矛盾。如果我们找到了这样的一个矩阵元，我们首先可以做的是将第 1 行与第 r 行进行互换。这可以通过一个交换矩阵来实现。假设通过交换行 1 和行 r 将 (A,b) 变为 $(\bar{A},\bar{b})$，那么这个变换可以表达为，

$$(\bar{A},\bar{b})=P^{1r}\cdot(A,b) \tag{51}$$

其中 $n\times n$ 的置换矩阵 $P^{(1r)}$ 的矩阵元为，

$$\begin{aligned} &[P^{(1r)}]_{1r}=[P^{(1r)}]_{r1}=1,\quad [P^{(1r)}]_{11}=[P^{(1r)}]_{rr}=0 \\ &[P^{(1r)}]_{ij}=\delta_{ij},\quad (ij)\neq(1r)\neq(r1) \end{aligned} \tag{52}$$

也就是说，除了第 1 和第 r 行、第 1 和第 r 列这四个矩阵元之外，其他的矩阵元与单位矩阵的相应矩阵元完全一样；而在这四个矩阵元的位置，它看起来就像 $\begin{pmatrix}0&1\\1&0\end{pmatrix}$ 一样。大家很容易验证，这个矩阵是一个非奇异的矩阵并且它的逆矩阵就是它本身。当我们用它左乘上矩阵 (A,b) 时，其作用是交换该矩阵的第 1 行和第 r 行。经过这个变换，(A,b) 变换到了 $(\bar{A},\bar{b})$，用数学表达式来写就是：$(\bar{A},\bar{b})=P^{(1r)}\cdot(A,b)$。

另外一类初等变换的矩阵的相貌这样的，

$$G^{(1)}=\begin{pmatrix} 1 & 0 & \cdots & 0 \\ -l_{21} & 1 & \cdots & 0 \\ \cdots & \cdots & \ddots & \cdots \\ -l_{n1} & 0 & \cdots & 1 \end{pmatrix} \tag{53}$$

它是一个下三角矩阵，并且仅仅在一列 (具体到这个例子是第一列) 与单位矩阵不同，其他地方与单位矩阵相同。这样的矩阵在数学上称为 Frobinius 矩阵。它的作用是，只要我们适当地选取 l_{r1} 的数值，就可以将增广矩阵 $(\bar{A},\bar{b})$ 中第一列中除了第一个元素 $\bar{a}_{11}$ 之外的其他元素统统变换为零。这个具体的选择是 $l_{r1}=\bar{a}_{r1}/\bar{a}_{11}, r=2,\cdots,n$，其中我们假定了 $\bar{a}_{11}\neq 0$。容易证明，这个矩阵也是非奇异的，事实上它的逆矩阵也是一个 Frobinius 矩阵，只不过其中的 $-l_{r1}$ 都要换成 $+l_{r1}$。

因此，经过上述的两个变换，我们总可以将原先的最一般的增广矩阵 (A,b) 变换为如下的形式，

$$(A',b')=\begin{pmatrix} a'_{11} & a'_{12} & \cdots & a'_{1n} & b'_1 \\ 0 & a'_{22} & \cdots & a'_{2n} & b'_2 \\ \cdots & \cdots & \ddots & \cdots & \cdots \\ 0 & a'_{n2} & \cdots & a'_{nn} & b'_n \end{pmatrix},\quad (A',b')=G^{(1)}(\bar{A},\bar{b})=G^{(1)}P^{(1r)}(A,b) \tag{54}$$

其中的第一个矩阵 $P^{(1r)}$ 的作用是，首先选择一个不为零的元素 a_{r1} 并且将它调到第一行，当然如果 a_{11} 本身就不等于零，这一步原则上也可以省略；第二个矩阵则将矩阵 $(\bar{A},\bar{b}) = P^{(1r)}(A,b)$ 的第一列从第二个元素开始往下的矩阵元全部清零。

经过上述两个步骤原先的增广矩阵 (A,b) 被约化了。具体来说，第一个待求的变量 x_1 仅仅出现在第一个方程之中，后面的 $(n-1)$ 个方程仅仅包含其余的 $n-1$ 个变量：$x_2,\cdots,x_n$。显然，我们可以对剩余的这 $(n-1)$ 个变量的增广矩阵继续利用上面提及的那两类变换，即将两行对掉 (注意，由于这时的对掉不牵涉第一行，因此不会破坏整个矩阵的结构，也就是说，除了 a'_{11} 之外，第一列都是零) 以及将某行乘以一个数与另一行相加，我们将可以将其变为仅仅包含变量 $(x_3,\cdots,x_n)$ 的增广矩阵。这个过程可以一直迭代地做下去。最终的结果就是我们将增广矩阵变换为了我们希望的上三角的形式并进一步利用反代的方法获得线性系统的解。

前面提及的寻找的不为零的矩阵元 $\bar{a}_{11} = a_{r1}$ 有个专门的名称，叫做支点元，或者简称支点。而这个过程称为支点遴选。虽然任意的不为零的元素都可以作为支点，但是直觉告诉我们它应当尽可能地远离零，因此一个自然的选择是

$$\bar{a}_{11} = \max_r |a_{r1}| \tag{55}$$

的确，数学上可以证明这样的选择造成的误差比起随便胡乱选择的支点来说是比较小的。这个选择一般称为部分支点遴选。与之相比，我们还可以进行所谓的完全支点遴选。在完全支点遴选中，我们不仅仅局限于第一列，而是在所有矩阵元中挑选模最大的作为支点，即，

$$\bar{a}_{11} = \max_{r,s} |a_{rs}| \tag{56}$$

随后我们将矩阵的第 1 行与第 r 行，第 1 列与第 s 列对换 (这相当于将原先的解 x 的第 1 个分量与第 s 个分量进行了一次对换)。这两个操作也可以用矩阵来表达：令 $P^{(1r)}\cdot(A,b) = (A'',b'')$，那么 $(\bar{A},\bar{b}) = (A''\cdot P^{1s},\bar{b})$。同时记住，$x_1$ 与 x_s 进行了对换。经过这两步操作，原先的增广矩阵 (A,b) 就变为了一个新的增广矩阵 $(\bar{A},\bar{b})$。然后可以进一步利用 $G^{(1)}$ 变换将其化为 (54) 的形式。我们可以将其简记为

$$(A',b') = \begin{pmatrix} a'_{11} & a'^{T} & b'_1 \\ 0 & \tilde{A} & \tilde{b} \end{pmatrix} \tag{57}$$

其中 a' 和 $\tilde{b}$ 都是具有 $(n-1)$ 个分量的矢量，$\tilde{A}$ 则是一个 $(n-1)\times(n-1)$ 的复矩阵。下面的步骤就是对 $(n-1)$ 阶的增广矩阵 $(\tilde{A},\tilde{b})$ 继续重复上面的步骤就可以将其最终化为上三角形式。

在第一步的支点遴选的过程中的置换矩阵 $P^{(1r)}$ 的上标 r 其实并不是必须的，它只是临时从各个行中计算出来的一个具有最大模的矩阵元的行指标而已。因此为了下面描述的方便，我们将这一步的置换矩阵记为 $P^{(1)}$，即隐去其上标 r。并且我们将约化后的矩阵记为 $(A^{(1)},b^{(1)})$。原始的增广矩阵则给它一个上标 0，即 $(A^{(0)},b^{(0)}) \equiv (A,b)$。这样一来第一步的约化可以表达为：

$$(A^{(1)},b^{(1)}) = G^{(1)}P^{(1)}(A^{(0)},b^{(0)}) \tag{58}$$

利用这个记号，我们可以比较方便地将整个约化过程表述为，

$$\begin{aligned} &(A^{(j)},b^{(j)}) = G^{(j)}P^{(j)}(A^{(j-1)},b^{(j-1)}), \quad j = 1,2,\cdots,(n-1) \\ &(A^{(n-1)},b^{(n-1)}) \equiv (U,c) \end{aligned} \tag{59}$$

或者更为明确地写出，

$$(U,c) = G^{(n-1)}P^{(n-1)}G^{(n-2)}P^{(n-2)}\cdots G^{(1)}P^{(1)}(A,b) \tag{60}$$

这个过程就是高斯消元法的完整过程。经过这个约化，原先的线性系统被成功约化为一个上三角线性系统。

下面给出消元过程得到上三角矩阵的伪码，为方便起见，我们假设支点很容易选取

$$
\begin{aligned}
&\text{For } k = 1, \cdots, n-1 \\
&\qquad \text{If } a_{kk} = 0, \text{ then stop} \\
&\qquad \text{For } i = k+1, k+2, \cdots, n \\
&\qquad\qquad l := -a_{ik}/a_{kk} \quad \text{计算倍数因子} \\
&\qquad\qquad \text{For } j = k+1, k+2, \cdots, n \\
&\qquad\qquad\qquad a_{ij} := a_{ij} + l \cdot a_{kj} \quad \text{更新矩阵元素} \\
&\qquad\qquad \text{End} \\
&\qquad\qquad b_i := b_i + l \cdot b_k \quad \text{更新右端项} \\
&\qquad \text{End} \\
&\text{End}
\end{aligned} \tag{61}
$$

我们下面分析一下这部分算法的复杂度。最外层循环的第 k 步计算 $n-k$ 次除法，$(n-k)(n-k+1)$ 次乘法，因此除法、乘法的总数是

$$
\begin{aligned}
&\text{除法：} (n-1) + (n-2) + \cdots + 1 = \frac{n(n-1)}{2} \\
&\text{乘法：} n(n-1) + (n-1)(n-2) + \cdots + 2 \times 1 = \frac{(n+1)n(n-1)}{3}
\end{aligned} \tag{62}
$$

忽略掉低次项，消元过程中的浮点数乘除法的计算次数约为 $\frac{1}{3}n^3$。加减法的次数也是 $\frac{1}{3}n^3$。这就把行列式计算的复杂度 $O(n!)$ 降为高斯消元的 $\frac{2}{3}n^3$。这对于量级为 $n \sim O(100)$ 的矩阵是没有任何问题的。当然，如果矩阵是 10^6 或者更大，我们还需要其他更好的方法。

提到这里我们顺便提一下算法中的 P 与 NP 问题。这里的 P 指的是多项式时间 (Polynomial)。一个复杂问题如果存在一个算法，能在多项式时间内 (比方说我们的例子是 $O(n^3)$) 解决，那么它便被称为 P 问题。NP 是指非确定性多项式时间 (nondeterministic polynomial)，一个复杂问题不能确定能否在多项式时间内解决。这类问题的复杂度往往是 $n!$ 或者 n^n，比方说大整数质因子分解，被广泛用于加密算法和银行系统。之所以说非确定性，是指对于 NP 问题，当我们在验证答案正确与否时，只需要多项式时间就可以做到。比方说如果有人告诉我们，数 13717421 可以因子分解为 3607 乘上 3803，那么你可以通过一个计算器马上去验证它。但如果有人告诉我们，这个数可以写成两个较小的数的乘积，你就不知道对不对了。这就是 NP 问题的特点。克雷数学研究所高额悬赏的七个千禧年难题，其中一个就是 P 是否等于 NP 的问题。也就是说，能用多项式时间验证解的问题能否在多项式时间内找出解，这是计算机与算法方面的重大问题。如果 P=NP，那我们的密钥将是不安全的。

这里，需要提醒大家注意的是，由于计算复杂度较高，在实际应用中应尽量避免对矩阵求逆。事实上，很多数学表达式中虽然包含了矩阵的逆，如 $A^{-1}b$，但实际计算时并不需要真正计算 A^{-1}，比如求解方程 $Ax = b$ 就会得到 $A^{-1}b$ 的结果，这比求出 A^{-1} 再做矩阵乘法更有效 (事实上，光是矩阵乘法就需要 $O(n^3)$ 次乘法运算)，也更准确。实际问题中真正需要 A 的逆的情况很少，因此一旦见到公式中的 A^{-1}，首先应该想到的是解方程组而不是求矩阵的逆。

2.3 矩阵的 LU 分解

一个方阵 $A \in C^{n\times n}$ 的 LU 分解是指将其分解为一个下三角和一个上三角矩阵的乘积:

$$A = LU \tag{63}$$

其中 $L(U)$ 分别是下 (上) 三角矩阵，L 和 U 分别指代 lower 和 upper。如果一个矩阵 A 的 LU 分解可以获得，那么求解它的线性方程可以转化为先后求解两个三角形矩阵的线性问题，而这个可以利用反代的方法解出。例如下面的两步走的方程的解 x 恰好就是 $Ax = b$ 的解，只要 $A = LU$:

$$y = Ux, \quad Ly = b \tag{64}$$

我们下面论证，前一节的高斯消元法恰好给出了矩阵 A 的一个 LU 分解。

按照上一节的讨论，利用一系列的置换矩阵与 Frobinius 矩阵的乘积，我们可以将 (A, b) 约化为上三角的形式 (U, c)。这些 Frobinius 矩阵的一般形式为,

$$G^{(j)} = \begin{pmatrix} 1 & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \cdots & \cdots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & \cdots & 0 \\ 0 & \cdots & 0 & 1 & \cdots & \cdots & 0 \\ \vdots & \cdots & \vdots & -l_{j+1,j} & 1 & \cdots & 0 \\ \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & -l_{n,j} & 0 & \cdots & 1 \end{pmatrix}$$

注意各个 $G^{(j)}$ 矩阵，如果我们遍历所有的 $j = 1, \cdots, n-1$，这些矩阵中非零的元素恰好填满一个 $n \times n$ 矩阵的左下三角区域，也就是主对角线的左下方的所有矩阵元。另一方面，根据高斯消元过程，我们有上三角矩阵 U

$$U = G^{(n-1)}P^{(n-1)}G^{(n-2)}P^{(n-2)}\cdots G^{(1)}P^{(1)}A \tag{65}$$

如果我们定义矩阵

$$M = G^{(n-1)}P^{(n-1)}G^{(n-2)}P^{(n-2)}\cdots G^{(1)}P^{(1)} \tag{66}$$

可以证明，矩阵

$$L = PM^{-1}, \quad P = P^{(n-1)}\cdots P^{(1)} \tag{67}$$

是下三角矩阵，并且对角元都是 1。于是，我们得到

$$PA = PM^{-1}U = LU \tag{68}$$

这个表达式实际上给出了矩阵 PA 的一个所谓的 LU 分解。事实上任意方阵都可以分解为这种形式，其中 $P = P^{(n-1)}\cdots P^{(1)}$ 是一系列置换矩阵的乘积。需要的说明是，由于下三角矩阵的对角元都已设成 1，所以 LU 分解具有唯一性。

利用 LU 分解可以解线性方程组。假定非奇异矩阵 A 可以分解一个下三角矩阵 L 与一个上三角矩阵 U 的乘积，即 $A = LU$，那么线性方程 $Ax = b$ 可以通过两个步骤来求解:

$$y = Ux, \quad Ly = b \tag{69}$$

其中每一步都是求解一个三角矩阵的线性方程组，这可以通过反代的方法给出。

于是一个关键的问题是，什么样的矩阵允许做这样的 LU 分解。显然，我们前面讨论的高斯消元法恰好给出了一个矩阵的 LU 分解，如果它存在的话。事实上，如果我们允许支点遴选，LU 分解应该是容易做到的。如果我们不做支点遴选，那么一个矩阵 A 是否存在 LU 分解，要取决于矩阵 A 以及它的各个子矩阵的秩。虽然对于任意矩阵的 LU 分解的定理有些复杂，但是对于一个实矩阵来说，它的判别标准还是比较简单的，这就是下面的定理。

定理：对于任意的实矩阵 $A \in R^{n\times n}$ 来说，它具有唯一的 LU 分解：$A = LU$，其中 L 是下三角矩阵且 $l_{ii} = 1$，$i = 1, \cdots, n$，U 为上三角矩阵的充要条件为 A 的所有主子矩阵 A_i，$i = 1, \cdots, (n-1)$ 都是非奇异的。

特别值得注意的是，即使是一个奇异的 $n \times n$ 矩阵也可以有唯一的 LU 分解，只要它的各个主子矩阵一直到 $(n-1)$ 阶都是非奇异的即可。例如，对于奇异的矩阵 $A = \begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix}$ 来说，我们可以获得它的标准的、唯一的 LU 分解：$A = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 0 & 0 \end{pmatrix}$。对于不满足定理条件的矩阵来说，可能根本就没有 LU 分解的存在。例如大家可以验证，Pauli 矩阵 $\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ 就不存在任何的 LU 分解。

这个定理的证明可以利用对 i 的数学归纳法展开。其步骤非常类似于我们下面要讲述的 Cholesky 分解中的证明，我们这里不再赘述。

2.4 Cholesky 分解

正如前面提到的, 对于正定的厄米矩阵 $A \in C^{n\times n}$ 来说, 我们可以找到一个矩阵 H 使得 $A = H^\dagger H$。事实上，我们可以要求矩阵 H 是上三角矩阵。这个分解一般称为 Cholesky 分解。一旦这样的 H 求得之后，求解线性方程 $Ax = b$ 的问题就可以分为两步进行: $Hx = y$，$H^\dagger y = b$，每一步都只涉及三角矩阵的线性系统。因此，对于求解正定厄米矩阵的线性方程问题就化为寻找矩阵的 Cholesky 分解问题。

其实 Cholesky 分解可以从 LU 分解中得到。我们先对正定的厄米矩阵进行 LU 分解，并把分解中的 U 矩阵写成对角矩阵乘以单位上三角矩阵的形式

$$U = \begin{pmatrix} u_{11} & & & \\ & u_{22} & & \\ & & \ddots & \\ & & & u_{nn} \end{pmatrix} \begin{pmatrix} 1 & u_{12}/u_{11} & \cdots & u_{1n}/u_{11} \\ & 1 & \cdots & u_{2n}/u_{22} \\ & & \ddots & \cdots \\ & & & 1 \end{pmatrix} = DU_0 \tag{70}$$

于是我们得到

$$A = LDU_0 \tag{71}$$

又因为 A 是厄米矩阵，则

$$A = A^\dagger = U_0^\dagger D L^\dagger \tag{72}$$

根据 LU 分解的唯一性，那么，我们有 $U_0^\dagger = L$，于是 A 的分解变为

$$A = LDL^\dagger = LD^{\frac{1}{2}}D^{\frac{1}{2}}L^\dagger = H^\dagger H \tag{73}$$

其中 $H^\dagger = LD^{\frac{1}{2}}$ 是下三角矩阵。

当然，实际我们在做 Cholesky 分解的时候，并不需要按照 LU 分解的步骤。由于对称性的存在，Cholesky 分解要比普通的 LU 分解更省时。

按照我们前面提及的线性代数的结果，正定的厄米矩阵的所有主子矩阵也都是正定的。因此，寻找 Cholesky 分解可以按照数学归纳法的思路进行。对于 n = 1 的一阶矩阵，问题的解是平庸的。令 A_i，$i = 1, \cdots, n$ 是原矩阵的第 i 阶的主子矩阵。它们显然也都是正定的厄米矩阵。假定我们已经找到了 $A_{i-1} \in C^{(i-1)\times(i-1)}$ 的分解矩阵 H_{i-1}，即 $A_{i-1} = H_{i-1}^\dagger H_{i-1}$，我们试图来寻找 A_i 的分解矩阵 H_i。为此，我们将矩阵 A_i 表达为

$$A_i = \begin{pmatrix} A_{i-1} & \vec{v} \\ \vec{v}^\dagger & \alpha \end{pmatrix} \tag{74}$$

其中 α 是一个正的实数，$\vec{v} \in C^{i-1}$ 为一矢量。事实上我们有，$\vec{v} = (a_{1i}, a_{2i}, \cdots, a_{i-1,i})^T$。我们希望矩阵 A_i 具有的分解为：

$$A_i = H_i^\dagger H_i = \begin{pmatrix} H_{i-1}^\dagger & 0 \\ \vec{h}^\dagger & \beta \end{pmatrix} \begin{pmatrix} H_{i-1} & \vec{h} \\ 0^\dagger & \beta \end{pmatrix} \tag{75}$$

其中的 $\vec{h} \in C^{i-1}$ 为一待定矢量而 β 为一个待定实数。将这个式子的右边乘出来我们就发现：

$$H_{i-1}^\dagger \cdot \vec{h} = \vec{v} \tag{76}$$

同时 $\vec{h}^\dagger\vec{h} + \beta^2 = \alpha$。由于 $H^\dagger$, $\vec{v}$ 已知，并且 $H^\dagger$ 是一个下三角矩阵，我们当然可以轻易求解出矢量 $\vec{h}$。另一方面，$\beta = \sqrt{\alpha - \vec{h}^\dagger\vec{h}}$ 则给出了参数 β 的数值。

正定厄米矩阵的 Cholesky 分解可以通过下面的算法获得。为了方便起见，我们在考虑一个实空间而非复空间的矩阵。这样厄米矩阵就变成了实对称矩阵。记下三角矩阵 $H_{i-1}^\dagger$ 的矩阵元为 $h_{\alpha\beta}$, $\beta \le \alpha \le i-1$。$\vec{h}^\dagger$ 的向量元素为 $\vec{h}^\dagger = (h_{i1}, h_{i2}, \cdots, h_{i(i-1)})$，而 $\vec{h}$ 的向量元素为 $\vec{h} = \begin{pmatrix} h_{i1} \\ h_{i2} \\ \cdots \\ h_{i(i-1)} \end{pmatrix}$。向量 $\vec{v}$ 的元素为 $\vec{v} = \begin{pmatrix} a_{i1} \\ a_{i2} \\ \cdots \\ a_{i(i-1)} \end{pmatrix}$。这个反代的前几步可以写成

$$\begin{aligned} h_{1,1}h_{i,1} = a_{i,1} \quad &\Rightarrow \quad h_{i,1} = \frac{a_{i,1}}{h_{1,1}} \\ h_{2,1}h_{i,1} + h_{2,2}h_{i,2} = a_{i,2} \quad &\Rightarrow \quad h_{i,2} = \frac{1}{h_{2,2}}\left(a_{i,2} - h_{2,1}h_{i,1}\right) \end{aligned} \tag{77}$$

整个反代这个公式实际上告诉我们

$$h_{ij} = \frac{1}{h_{jj}}\left(a_{ij} - \sum_{k=1}^{j-1} h_{ik}h_{jk}\right), \quad j = 1, \cdots, i-1 \tag{78}$$

算法如下

$$
\begin{aligned}
&\text{For } j=1,2,\ldots,n\\
&\quad \text{For } k=1,\cdots,(j-1)\\
&\qquad a_{jj}:=a_{jj}-a_{jk}^2\\
&\quad \text{End}\\
&\quad a_{jj}:=\sqrt{a_{jj}}\quad (\text{这一步是为了实现 } \beta=h_{jj}=\sqrt{a_{jj}-\textstyle\sum_{k=1}^{j-1}h_{jk}^2})\\
&\quad \text{For } i=j+1,j+2,\cdots,n\\
&\qquad \text{For } k=1,2,\cdots,j-1\\
&\qquad\quad a_{ij}:=a_{ij}-a_{ik}a_{jk}\\
&\qquad \text{End}\\
&\qquad a_{ij}:=a_{ij}/a_{jj}\quad (\text{这一步是为了实现 } h_{ij}=\tfrac{1}{h_{jj}}\left(a_{ij}-\textstyle\sum_{k=1}^{j-1}h_{ik}h_{jk}\right))\\
&\quad \text{End}\\
&\text{End}
\end{aligned}
\tag{79}
$$

这里有个求平方根的运算，大家可以想想怎么快速求平方根。如果我们忽略 n 次求平方根运算，那么，这个算法的需要 $\frac{1}{6}n^3$ 次乘除法和 $\frac{1}{6}n^3$ 次加减法，比 LU 分解要节省大约一半。除此之外，因为对称性的缘故，所需的内存也减半。从稳定性上来说，Cholesky 分解的稳定性极佳，只要矩阵确实是正定的厄米矩阵。

2.5 三对角矩阵线性方程组的求解

本节我们讨论一个三对角矩阵所给出的线性方程组的求解问题。这类问题会出现在不少的数值应用之中，例如后面会讨论到的利用三次样条函数进行拟合的问题。

我们将问题中的三对角矩阵的 LU 分解写为如下的形式

$$
A=\begin{pmatrix} a_1 & c_1 & \cdots & 0\\ b_2 & a_2 & \ddots & \\ & \ddots & & c_{n-1}\\ 0 & & b_n & a_n \end{pmatrix}=LU \tag{80}
$$

其中的矩阵 L 和 U 分别是下和上双对角矩阵:

$$
L=\begin{pmatrix} 1 & & \cdots & 0\\ \beta_2 & 1 & \ddots & \\ & \ddots & \ddots & \\ 0 & & \beta_n & 1 \end{pmatrix},\quad U=\begin{pmatrix} \alpha_1 & c_1 & \cdots & 0\\ & \alpha_2 & \ddots & \\ & & \ddots & c_{n-1}\\ 0 & & & \alpha_n \end{pmatrix} \tag{81}
$$

根据高斯消元过程，很容易证明矩阵 U 的副对角线元素与 A 的相同。另外参数 α_i 以及 β_i 与原先的系数之间的关系有下式给出，

$$
\alpha_1=a_1,\quad \beta_i=\frac{b_i}{\alpha_{i-1}},\quad \alpha_i=a_i-\beta_i c_{i-1},\quad i=2,\cdots,n \tag{82}
$$

这个算法又称为 Thomas 算法。具体如下

$$\begin{aligned}&\alpha_1 = a_1\\&\text{For } i = 2, \cdots, n\\&\qquad \beta_i = \frac{b_i}{\alpha_{i-1}}\\&\qquad \alpha_i = a_i - \beta_i c_{i-1}\\&\text{End}\end{aligned} \tag{83}$$

经过这个分解之后，我们可以利用该分解求解方程 $Ax = b$。具体可以分成两步：$Ly = b$, $Ux = y$。

$$\begin{aligned}&y_1 = b_1, \quad y_i = b_i - \beta_i y_{i-1}, \quad i = 2, \cdots, n\\&x_n = \frac{y_n}{\alpha_n}, \quad x_i = (y_i - c_i x_{i+1})/\alpha_i, \quad i = n-1, \cdots, 1\end{aligned} \tag{84}$$

这个过程的计算量大约是 $8n - 7$。具体来说，上述分解本身需要 $3(n-1)$，而求解两个三角系统的计算量为 $5n - 4$。

我们下面来看一个三对角矩阵的变种。设

$$A = \begin{pmatrix} a_1 & c_1 & & b_1 \\ b_2 & a_2 & \ddots & \\ & \ddots & \ddots & c_{n-1} \\ c_n & & b_n & a_n \end{pmatrix}, \quad Ax = f \tag{85}$$

我们可以先解

$$A_1 \begin{pmatrix} u_2 \\ u_3 \\ \cdots \\ u_n \end{pmatrix} = \begin{pmatrix} f_2 \\ f_3 \\ \cdots \\ f_n \end{pmatrix}, \quad A_1 = \begin{pmatrix} a_2 & \ddots & \\ \ddots & \ddots & c_{n-1} \\ & b_n & a_n \end{pmatrix} \tag{86}$$

然后再解

$$A_1 \begin{pmatrix} v_2 \\ v_3 \\ \cdots \\ v_n \end{pmatrix} = \begin{pmatrix} -b_2 \\ 0 \\ \cdots \\ -c_n \end{pmatrix} \tag{87}$$

定义 $x_i = u_i + x_1 v_i$, $i = 2, \cdots, n$。很明显，这样定义列向量 x 以后，与 $f_2, \cdots, f_n$ 对应的 $n-1$ 个方程自动满足。然后我们再来看与 f_1 对应的方程

$$a_1 x_1 + c_1 x_2 + b_1 x_n = f_1 \quad \Rightarrow \quad x_1 = \frac{f_1 - c_1 u_1 - b_1 u_n}{a_1 + c_1 v_1 + b_1 v_n} \quad \Rightarrow \quad x_i = u_i + x_1 v_i \tag{88}$$

2.6 高斯消元过程中的舍入误差估计

我们来看 LU 分解中的舍入误差。假设 $A \in K^{n\times n}$。我们计算得到了下上三角矩阵 $\hat{L}$ 和 $\hat{U}$。$\hat{L}$ 和 $\hat{U}$ 肯定是带有舍入误差的。于是

$$\hat{L}\hat{U} = A + H \tag{89}$$

这里 H 是用来衡量 LU 分解中舍入误差的大小的。这里我们有一个结论

$$|H| \leq 2(n-1)u\left(|A| + |\hat{L}||\hat{U}|\right) + O(u^2), \quad u = \epsilon_M/2 \tag{90}$$

这里的 $|H|$ 还是一个矩阵，表示 $|H|_{ij} = |H_{ij}|$。

这件事情在 $n = 1$ 的时候是显然成立的。对于 $n \geq 2$ 时，我们可以把 A 写成

$$A = \begin{pmatrix} \alpha & \omega^T \\ v & B \end{pmatrix} \tag{91}$$

这里的 $B \in K^{(n-1)\times(n-1)}$。根据高斯消元的算法，我们会有

$$\hat{z} = fl(v/\alpha), \quad \hat{C} = fl(\hat{z}\omega^T), \quad \hat{A}_1 = fl(B - \hat{C}) \tag{92}$$

然后得到 $\begin{pmatrix} \alpha & \omega^T \\ 0 & \hat{A}_1 \end{pmatrix}$。这里的舍入误差有

$$\begin{aligned} &\hat{z} = v/\alpha + f, \quad |f| \leq u|v/\alpha| \\ &\hat{C} = \hat{z}\omega^T + F_1, \quad |F_1| \leq u|\hat{z}||\omega^T| \\ &\hat{A}_1 = B - (\hat{z}\omega^T + F_1) + F_2, \quad |F_2| \leq u(|B| + |\hat{z}||\omega^T|) + O(u^2) \end{aligned} \tag{93}$$

而另外一方面，对于 $\hat{A}_1$ 做 LU 分解，会得到

$$\hat{L}_1\hat{U}_1 = \hat{A}_1 + H_1 \tag{94}$$

根据数学归纳法，我们假定已经得到了

$$H_1 \leq 2(n-2)u\left(|\hat{A}_1| + |\hat{L}_1||\hat{U}_1|\right) \tag{95}$$

这时，我们把 $\hat{L}$ 和 $\hat{U}$ 写出来就是

$$\hat{L} = \begin{pmatrix} 1 & 0 \\ \hat{z} & \hat{L}_1 \end{pmatrix}, \quad \hat{U} = \begin{pmatrix} \alpha & \omega^T \\ 0 & \hat{U}_1 \end{pmatrix} \tag{96}$$

然后根据 $\hat{L}\hat{U} = A + H$，我们可以得到

$$H = \begin{pmatrix} 0 & 0 \\ \alpha f & H_1 - F_1 + F_2 \end{pmatrix} \tag{97}$$

我们需要证明的是

$$|H| \leq 2(n-1)u \begin{pmatrix} 2|\alpha| & 2|\omega^T| \\ |v| + |\alpha||f| & |B| + |\hat{L}_1||\hat{U}_1| + |\hat{z}||\omega^T| + O(u^2) \end{pmatrix} \tag{98}$$

事实上我们只需要证明

$$|H_1| + |F_1| + |F_2| \leq 2(n-1)u\left(|B| + |\hat{L}_1||\hat{U}_1| + |\hat{z}||\omega^T|\right) + O(u^2) \tag{99}$$

我们只需要把关于 $|H_1|, |F_1|$ 和 $|F_2|$ 的不等式

$$\begin{aligned} |H_1| &\leq 2(n-2)u\left(|B| + |\hat{z}||\omega^T| + |\hat{L}_1||\hat{U}_2|\right) + O(u^2) \\ |F_1| + |F_2| &\leq u\left(|B| + 2|\hat{z}||\omega^T|\right) + O(u^2) \end{aligned} \tag{100}$$

代入即可得到。

我们总结一下，这个误差分析告诉我们：第一，对于 LU 分解，如果矩阵越大，也就是 n 越大，那么误差越大，这个误差正比于 n。第二，这个误差的大小与 L 和 U 这两个矩阵有关。假如我们不采用支点遴选，当我们的支点很接近 0 的时候，就意味着 Frobinius 矩阵的矩阵元很大，这个矩阵元会传递到下三角矩阵 L 中，从而使得整个 LU 的误差变大。

对于三对角矩阵分解，Thomas 算法，刘川老师的讲义里给出，它的舍入误差为

$$\hat{L}\hat{U} = A + H \quad \Rightarrow \quad |H| \leq (4u + 3u^2 + u^3)|\hat{L}| \cdot |\hat{U}| \tag{101}$$

这里，我们并不要求证明。有一点需要说明的是，对于普通的矩阵元，它的非零矩阵元个数为 $O(n^2)$。因此，舍入误差在 LU 分解中通过多次运算，误差累加到 $O(n)$ 的量级。而对于三对角矩阵，它的非零矩阵元个数为 $O(n)$，因此它是一个稀疏矩阵，无论在内存的要求还是运算的步骤，都大为缩减，而且它的舍入误差是 $O(1)$ 的量级。因此 Thomas 算法的稳定性是非常不错的。这里 H 的上限基本上由 $|\hat{L}|$ 和 $|\hat{U}|$ 矩阵中的最大矩阵元决定。只要各个系数 β_i 以及 α_i 不会太大。这些系数变大的一种可能是其中某个 α_i 非常接近于零。因此，如果矩阵的确是非奇异的，那么各个 α_i 必定不能等于零。但是如果矩阵接近于奇异，那么有可能造成算法出现不稳定的情况。如果矩阵 $A \in R^{n\times n}$ 是一个正定实对称矩阵，那么我们可以获得更好的估计，

$$|H| \leq \frac{4u + 3u^2 + u^3}{1-u}|A|. \tag{102}$$

这个分解更为稳定。