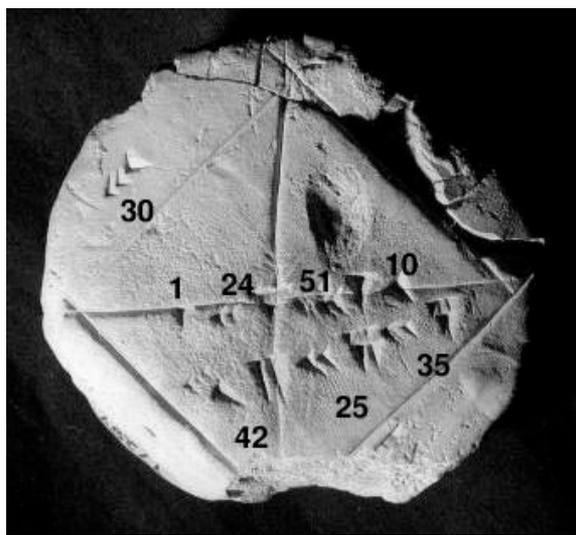


计算方法 (A)



童伟华 管理科研楼1205室

E-mail: tongwh@ustc.edu.cn

中国科学技术大学 数学科学学院

<http://math.ustc.edu.cn/>





第七章 计算矩阵的特征值与特征向量

特征值与特征向量



- 在实际工程计算中，经常会遇到特征值和特征向量的计算，如：机械、结构或电磁振动中的固有值问题；物理学中的各种临界值等
- 定义：设有 n 阶矩阵 A ，若有数 λ 及非零向量 v 满足方程 $Av = \lambda v$ ，则称 λ 为 A 的特征值， v 为属于特征值 λ 的特征向量
- 线性代数中特征值和特征向量的计算步骤：
 - 计算矩阵的特征多项式： $\det(\lambda I - A) = \lambda^n + \dots + (-1)^n \det(A)$
 - 求解齐次线性方程组的解空间： $(A - \lambda_i I)v = 0, i = 1, 2, \dots, n$
- 困难：求解高次多项式的根

特征值与特征向量



■ 求解特征值与特征向量的方法按求解方法的特性可以分为：

- 直接法：可用来计算矩阵的全部特征值和特征向量，一般用于稠密矩阵，计算代价为 $O(n^3)$ 且对矩阵的元素是相对不敏感的（注意：此处的直接法必须作迭代，因为求特征值问题等价于求多项式的根，而后者不可能存在非迭代的方法。如果经验表明，用一个固定的迭代次数收敛（几乎）从不失败，则称该方法是直接的）
- 迭代法：一般只提供特征值和特征向量的一个子集的近似，一般用于稀疏矩阵或者只适合执行矩阵-向量乘法的矩阵。为了得到几个足够精确的特征值可能需要运行足够长的时间，收敛性强烈的依赖于矩阵的元素

■ 求解特征值与特征向量的方法按矩阵的对称性可以分为：

- 非对称特征值问题：非对称矩阵的特征值问题
- 对称特征值问题：对称矩阵的特征值问题，不仅内容非常丰富和优美，而且在实际问题中经常遇到

幂法



- 在实际问题中，矩阵按模最大的特征值往往起重要的作用，譬如矩阵的谱半径决定了迭代矩阵是否收敛
- 幂法：计算按模最大特征值及相应特征向量的数值方法
- 要求：矩阵 A 具有完备的特征向量系，即 A 有 n 个线性无关的特征向量。在实际问题中，常遇到的实对称矩阵或具有 n 个互不相同特征值的矩阵就具有这种性质

- 设矩阵 A 的特征值和特征向量如下:

特征值: $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$

特征向量: $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

A 是非亏损的, 即特征值的几何重数等于代数重数

幂法可以求 λ_1, \mathbf{v}_1

- 基本思想: 不断利用矩阵向量乘法, 分析得到的向量序列, 计算出矩阵按模最大特征值及相应特征向量
- 取初值 $\mathbf{x}^{(0)}$, 做迭代

$$\mathbf{x}^{(k+1)} = A\mathbf{x}^{(k)}$$

$$\Rightarrow \mathbf{x}^{(k+1)} = A^k \mathbf{x}^{(0)}, \mathbf{x}^{(0)} = \alpha_1 \mathbf{v}_1 + \alpha_2 \mathbf{v}_2 + \dots + \alpha_n \mathbf{v}_n$$

$$\Rightarrow \mathbf{x}^{(k+1)} = A^k (\alpha_1 \mathbf{v}_1 + \alpha_2 \mathbf{v}_2 + \dots + \alpha_n \mathbf{v}_n)$$

$$= \alpha_1 A^k \mathbf{v}_1 + \alpha_2 A^k \mathbf{v}_2 + \dots + \alpha_n A^k \mathbf{v}_n$$

$$= \alpha_1 \lambda_1^k \mathbf{v}_1 + \alpha_2 \lambda_2^k \mathbf{v}_2 + \dots + \alpha_n \lambda_n^k \mathbf{v}_n$$

幂法



■ 情形1: 若 $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$, 则有

$$\mathbf{x}^{(k)} = \lambda_1^k \left(\alpha_1 \mathbf{v}_1 + \left(\frac{\lambda_2}{\lambda_1} \right)^k \alpha_2 \mathbf{v}_2 + \dots + \left(\frac{\lambda_n}{\lambda_1} \right)^k \alpha_n \mathbf{v}_n \right)$$

$$\alpha_1 \neq 0 \Rightarrow \begin{cases} \mathbf{x}^{(k)} \approx \lambda_1^k (\alpha_1 \mathbf{v}_1) \\ \mathbf{x}^{(k+1)} \approx \lambda_1^{k+1} (\alpha_1 \mathbf{v}_1) \end{cases}, k \rightarrow \infty$$

$$\Rightarrow \begin{cases} \lambda_1 \approx \mathbf{x}^{(k+1)} / \mathbf{x}^{(k)} \\ \mathbf{v}_1 \approx \mathbf{x}^{(k+1)} \end{cases}$$

收敛速度取决于: $\frac{|\lambda_2|}{|\lambda_1|}$

- 情形2: 若 $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \dots \geq |\lambda_n|, \lambda_1 = -\lambda_2$

$$\mathbf{x}^{(k)} = \lambda_1^k \left(\alpha_1 \mathbf{v}_1 + (-1)^k \alpha_2 \mathbf{v}_2 + \dots + \left(\frac{\lambda_n}{\lambda_1} \right)^k \alpha_n \mathbf{v}_n \right)$$

$$\alpha_1 \neq 0 \Rightarrow \mathbf{x}^{(k)} \approx \lambda_1^k \left(\alpha_1 \mathbf{v}_1 + (-1)^k \alpha_2 \mathbf{v}_2 \right), k \rightarrow \infty$$

$$\Rightarrow \begin{cases} \mathbf{x}^{(2k+2)} / \mathbf{x}^{(2k)} \approx \lambda_1^2 \\ \mathbf{x}^{(2k+1)} / \mathbf{x}^{(2k-1)} \approx \lambda_1^2 \end{cases}, \begin{cases} \mathbf{v}_1 \approx \mathbf{x}^{(k+1)} + \lambda_1 \mathbf{x}^{(k)} \\ \mathbf{v}_2 \approx \mathbf{x}^{(k+1)} - \lambda_1 \mathbf{x}^{(k)} \end{cases}$$

- 其它情形……

- 在幂法中，我们构造的序列

$$\mathbf{x}^{(k)} = \lambda_1^k \left(\alpha_1 \mathbf{v}_1 + \left(\frac{\lambda_2}{\lambda_1} \right)^k \alpha_2 \mathbf{v}_2 + \cdots + \left(\frac{\lambda_n}{\lambda_1} \right)^k \alpha_n \mathbf{v}_n \right)$$

可以看出，当 $k \rightarrow +\infty$

$$\mathbf{x}^{(k)} \rightarrow \begin{cases} 0 & , \quad |\lambda_1| < 1 \\ \infty & , \quad |\lambda_1| > 1 \end{cases}$$

- 为避免 $\mathbf{x}^{(k)}$ 分量过大（上溢）或过小（下溢），在实际运算中采用规范运算：

$$\begin{cases} \mathbf{x}^{(k+1)} = \mathbf{A} \mathbf{y}^{(k)} \\ \mathbf{y}^{(k+1)} = \mathbf{x}^{(k+1)} / \left\| \mathbf{x}^{(k+1)} \right\|_{\infty} \end{cases}$$

■ 不难发现:

$$\begin{cases} \mathbf{y}^{(k)} = \mathbf{A} \mathbf{y}^{(k-1)} / \|\mathbf{x}^{(k)}\|_{\infty} = \cdots = \mathbf{A}^k \mathbf{y}^{(0)} / \left(\|\mathbf{x}^{(k)}\|_{\infty} \cdots \|\mathbf{x}^{(1)}\|_{\infty} \right) \\ \|\mathbf{y}^{(k)}\|_{\infty} = 1 \end{cases}$$

■ 从而有

$$\mathbf{y}^{(k)} = \mathbf{A}^k \mathbf{y}^{(0)} / \|\mathbf{A}^k \mathbf{y}^{(0)}\|_{\infty}$$
$$\mathbf{y}^{(k)} = \frac{\lambda_1^k \left(\alpha_1 \mathbf{v}_1 + \left(\frac{\lambda_2}{\lambda_1} \right)^k \alpha_2 \mathbf{v}_2 + \cdots + \left(\frac{\lambda_n}{\lambda_1} \right)^k \alpha_n \mathbf{v}_n \right)}{|\lambda_1|^k \cdot \left\| \left(\alpha_1 \mathbf{v}_1 + \left(\frac{\lambda_2}{\lambda_1} \right)^k \alpha_2 \mathbf{v}_2 + \cdots + \left(\frac{\lambda_n}{\lambda_1} \right)^k \alpha_n \mathbf{v}_n \right) \right\|_{\infty}}$$

■ 情形1: 若 $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$, 则有

$$\mathbf{y}^{(k)} \approx \frac{\lambda_1^k (\alpha_1 \mathbf{v}_1)}{|\lambda_1^k| \cdot \|(\alpha_1 \mathbf{v}_1)\|_\infty} \Rightarrow \mathbf{y}^{(k)} \rightarrow \begin{cases} \pm \frac{\alpha_1 \mathbf{v}_1}{\|\alpha_1 \mathbf{v}_1\|_\infty}, & \lambda_1 < 0 \\ \frac{\alpha_1 \mathbf{v}_1}{\|\alpha_1 \mathbf{v}_1\|_\infty}, & \lambda_1 > 0 \end{cases}$$

$$\mathbf{x}^{(k+1)} = \mathbf{A} \mathbf{y}^{(k)} = \lambda_1 \mathbf{y}^{(k)}$$

$$\|\mathbf{x}^{(k+1)}\|_\infty = |\lambda_1 \mathbf{y}^{(k)}|_\infty \approx |\lambda_1|$$

A) 若 $\lambda_1 > 0$, 则 $\{\mathbf{y}^{(k)}\}$ 收敛, $\lambda_1 \approx \|\mathbf{x}^{(k+1)}\|_\infty$, $\mathbf{v}_1 \approx \mathbf{y}^{(k)}$

B) 若 $\lambda_1 < 0$, 则 $\{\mathbf{y}^{(2k)}\}, \{\mathbf{y}^{(2k+1)}\}$ 分别收敛于互为反号的向量, $\lambda_1 \approx -\|\mathbf{x}^{(k+1)}\|_\infty$, $\mathbf{v}_1 \approx \mathbf{y}^{(k)}$

- 情形2: 若 $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \dots \geq |\lambda_n|$, $\lambda_1 = -\lambda_2$, 则有

$$\mathbf{y}^{(k)} = \frac{\lambda_1^k \left(\alpha_1 \mathbf{v}_1 + (-1)^k \alpha_2 \mathbf{v}_2 + \dots + \left(\frac{\lambda_n}{\lambda_1} \right)^k \alpha_n \mathbf{v}_n \right)}{|\lambda_1|^k \cdot \left\| \left(\alpha_1 \mathbf{v}_1 + (-1)^k \alpha_2 \mathbf{v}_2 + \dots + \left(\frac{\lambda_n}{\lambda_1} \right)^k \alpha_n \mathbf{v}_n \right) \right\|_\infty}$$

$\{\mathbf{y}^{(2k)}\}, \{\mathbf{y}^{(2k-1)}\}$ 分别收敛于两个向量, 且不是互为反号

令 $\mathbf{x}^{(k+1)} = \mathbf{A}\mathbf{x}^{(k)}$, 则 $\lambda_1 \approx \sqrt{\mathbf{x}^{(k+1)} / \mathbf{y}^{(k-1)}}$, $\lambda_2 = -\lambda_1$

$$\begin{cases} \mathbf{v}_1 \approx \mathbf{x}^{(k+1)} + \lambda_1 \mathbf{x}^{(k)} \\ \mathbf{v}_2 \approx \mathbf{x}^{(k+1)} - \lambda_1 \mathbf{x}^{(k)} \end{cases}$$

- 其它情形……

- (Rayleigh商) 设 $A \in R^{n \times n}$, 对 $\mathbf{x} \in R^n$ 定义

$$R(\mathbf{x}) = \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}}$$

称为矩阵 A 在 \mathbf{x} 的Rayleigh商

- 性质: 对于给定的 \mathbf{x} , 最小二乘问题

$$\min_{\alpha \in \mathbb{R}} \|\mathbf{A}\mathbf{x} - \alpha\mathbf{x}\|_2$$

有显示解, 即 $\alpha = R(\mathbf{x})$

当 \mathbf{x} 趋于但未必等于特征向量时,
 $R(\mathbf{x})$ 就是一个自然的特征值估计

- 不难发现 $\nabla R(\mathbf{x}) = 2(\mathbf{A}\mathbf{x} - R(\mathbf{x})\mathbf{x}) / \mathbf{x}^T \mathbf{x}$, 因此: 若 \mathbf{x} 是 A 的特征向量, 那么 $R(\mathbf{x})$ 的梯度为零向量; 反之, 若 $\nabla R(\mathbf{x}) = 0$ 且 $\mathbf{x} \neq 0$, 则 \mathbf{x} 是特征向量, $R(\mathbf{x})$ 是相对应的特征值
- 利用函数 $R(\mathbf{x})$ 的光滑性, 可以得出一个重要结论:

$$R(\mathbf{x}) - R(\mathbf{q}) = O(\|\mathbf{x} - \mathbf{q}\|^2), \quad \mathbf{x} \rightarrow \mathbf{q}$$

■ 幂法迭代算法

Algorithm 18 Power Iteration Algorithm

Input:

$n, \mathbf{A}, \mathbf{x}, M, \varepsilon$

1: $\mathbf{v} \leftarrow \mathbf{x}/\|\mathbf{x}\|$;

2: $\mu \leftarrow \mathbf{v}^T \mathbf{A} \mathbf{v}$;

3: **for** $k = 1$ to M **do**

4: $\mathbf{x} \leftarrow \mathbf{A} \mathbf{v}$;

5: $\mathbf{u} \leftarrow \mathbf{x}/\|\mathbf{x}\|$;

6: $\lambda \leftarrow \mathbf{u}^T \mathbf{A} \mathbf{u}$;

7: **if** $|\lambda - \mu| < \varepsilon$ **or** $\|\mathbf{u} - \mathbf{v}\| < \varepsilon$ **then**

8: $\mathbf{x} \leftarrow \mathbf{u}$;

9: **break**;

10: **else**

11: $\mathbf{v} \leftarrow \mathbf{u}$;

12: $\mu \leftarrow \lambda$;

13: **end if**

14: **end for**

Output:

λ, \mathbf{x}

反幂法



- 基本思想：计算按模最小特征值及相应特征向量的数值方法

- 矩阵 \mathbf{A} 与 \mathbf{A}^{-1} 特征值之间的关系：

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v} \Rightarrow \mathbf{A}^{-1}\mathbf{v} = \frac{1}{\lambda}\mathbf{v}$$

$$\mathbf{A} : |\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| > 0$$

$$\mathbf{A}^{-1} : |\mu_1| \leq |\mu_2| \leq \dots \leq |\mu_n|$$

- 反幂法迭代：

$$\begin{cases} \mathbf{x}^{(k+1)} = \mathbf{A}^{-1}\mathbf{y}^{(k)} \\ \mathbf{y}^{(k+1)} = \mathbf{x}^{(k+1)} / \|\mathbf{x}^{(k+1)}\|_{\infty} \end{cases}$$

- 为避免求 \mathbf{A}^{-1} ，可用分解法求解线性方程组：

$$\mathbf{x}^{(k+1)} = \mathbf{A}^{-1}\mathbf{y}^{(k)} \Leftrightarrow \mathbf{A}\mathbf{x}^{(k+1)} = \mathbf{y}^{(k)}$$

反幂法



- 设 $\tilde{\lambda}_i$ 是矩阵 A 特征值 λ_i 的近似，带原点位移 $\tilde{\lambda}_i$ 的反幂法迭代进行如下：

$$\begin{cases} \mathbf{x}^{(k+1)} = (\mathbf{A} - \tilde{\lambda}_i \mathbf{I})^{-1} \mathbf{y}^{(k)} \\ \mathbf{y}^{(k+1)} = \mathbf{x}^{(k+1)} / \|\mathbf{x}^{(k+1)}\|_{\infty} \end{cases}$$

- 若 $|\lambda_i - \tilde{\lambda}_i| < |\lambda_j - \tilde{\lambda}_i|, \forall j \neq i$ ，则迭代收敛
- 用途：用于求矩阵 A 的最接近给定初值 $\tilde{\lambda}_i$ 的特征值 λ_i 及相应的特征向量

■ 带原点位移的反幂法迭代算法

Algorithm 19 Shifted Inverse Power Iteration Algorithm

Input:

$n, \mathbf{A}, \mathbf{x}, s, M, \varepsilon$

- 1: $\mathbf{v} \leftarrow \mathbf{x} / \|\mathbf{x}\|;$
- 2: $\mathbf{x} \leftarrow (\mathbf{A} - s\mathbf{I})^{-1}\mathbf{v};$
- 3: $\mu \leftarrow \mathbf{v}^T \mathbf{x};$
- 4: **for** $k = 1$ to M **do**
- 5: $\mathbf{u} \leftarrow \mathbf{x} / \|\mathbf{x}\|;$
- 6: $\mathbf{x} \leftarrow (\mathbf{A} - s\mathbf{I})^{-1}\mathbf{u};$
- 7: $\lambda \leftarrow \mathbf{u}^T \mathbf{x};$
- 8: **if** $|\lambda - \mu| < \varepsilon$ **or** $\|\mathbf{u} - \mathbf{v}\| < \varepsilon$ **then**
- 9: $\mathbf{x} \leftarrow \mathbf{u};$
- 10: $\lambda = 1/\lambda + s;$
- 11: **break;**
- 12: **else**
- 13: $\mathbf{v} \leftarrow \mathbf{u};$
- 14: $\mu \leftarrow \lambda;$
- 15: **end if**
- 16: **end for**

Output:

λ, \mathbf{x}

实对称矩阵的Jacobi方法

- 特征值与特征向量的计算：要充分考虑矩阵的特性，譬如对称性、正定性、稀疏性等，来选用合适的算法
- 定理：若矩阵 $A \in \mathbb{R}^{n \times n}$ 是对称的，则存在正交阵 $Q \in \mathbb{R}^{n \times n}$ 使得

$$Q^{-1}AQ = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix}, \lambda_i \in \mathbb{R}, i = 1, 2, \dots, n$$

- 基本思想：对于一般的矩阵，不可能直接找到 Q ，而是构造一系列特殊形式的正交阵对 A 进行正交变换，使得对角元比重逐渐增加，非对角元变小。

实对称矩阵的Jacobi方法



■ Givens旋转变换:

$$\mathbf{Q}(p, q, \theta) = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & \cos \theta & & \sin \theta & \\ & & & \ddots & & \\ & & -\sin \theta & & \cos \theta & \\ & & \uparrow & & \uparrow & \ddots \\ & & p\text{列} & & q\text{列} & \\ & & & & & 1 \end{pmatrix}$$

实对称矩阵的Jacobi方法

■ 设矩阵 $\mathbf{A} = (a_{ij})_{n \times n}$ ，则 $\mathbf{B} = \mathbf{Q}^T(p, q, \theta) \mathbf{A} \mathbf{Q}(p, q, \theta) = (b_{ij})_{n \times n}$

$$\begin{cases} b_{ip} = b_{pi} = a_{pi} \cos \theta - a_{qi} \sin \theta, i \neq p, q \\ b_{iq} = b_{qi} = a_{pi} \sin \theta + a_{qi} \cos \theta, i \neq p, q \\ b_{pp} = a_{pp} \cos^2 \theta + a_{qq} \sin^2 \theta - a_{pq} \sin 2\theta \\ b_{qq} = a_{pp} \sin^2 \theta + a_{qq} \cos^2 \theta + a_{pq} \sin 2\theta \\ b_{pq} = b_{qp} = a_{pq} \cos 2\theta + \frac{a_{pp} - a_{qq}}{2} \sin 2\theta \end{cases}$$

■ 目标:

$$b_{pq} = b_{qp} = a_{pq} \cos 2\theta + \frac{a_{pp} - a_{qq}}{2} \sin 2\theta = 0$$

实对称矩阵的Jacobi方法



■ 若记

$$s = \frac{a_{qq} - a_{pp}}{2a_{pq}}, \quad t = \tan \theta$$

■ 则有

$$t = \begin{cases} t^2 + 2ts - 1 = 0 \text{按模的较小根}, & s \neq 0 \\ 1, & s = 0 \end{cases}$$

实对称矩阵的Jacobi方法



■ 若记

$$\begin{cases} \cos \theta = \frac{1}{\sqrt{1+t^2}} \equiv c \\ \sin \theta = \frac{t}{\sqrt{1+t^2}} \equiv d \end{cases}$$

■ 则有

$$\begin{cases} b_{ip} = b_{pi} = ca_{pi} - da_{qi}, i \neq p, q \\ b_{iq} = b_{qi} = da_{pi} + ca_{qi}, i \neq p, q \\ b_{pp} = a_{pp} - ta_{pq} \\ b_{qq} = a_{qq} + ta_{pq} \\ b_{pq} = b_{qp} = 0 \end{cases}$$

$$\Rightarrow \begin{cases} \sum_{i \neq j} b_{ij}^2 = \sum_{i \neq j} a_{ij}^2 - 2a_{pq}^2 \\ \sum_{i=1}^n b_{ii}^2 = \sum_{i=1}^n a_{ii}^2 + 2a_{pq}^2 \end{cases}$$

实对称矩阵的Jacobi方法

- Jacobi方法：取 p, q 使得 $|a_{pq}| = \max_{i \neq j} |a_{ij}|$ ，连续实施Givens变换，即

$$\mathbf{A}^{(k+1)} = \mathbf{Q}^T(p, q, \theta) \mathbf{A}^{(k)} \mathbf{Q}(p, q, \theta)$$

- 定理：若对称矩阵 $\mathbf{A} \in \mathbb{R}^{n \times n}$ ，则

$$\mathbf{A}^{(k+1)} \rightarrow \text{diag}\{\lambda_1, \dots, \lambda_n\}$$

- 优点：特征值计算结果精度一般比较高，特征向量的正交性也较好，适合于并行计算
- 缺点：当计算稀疏矩阵的特征值时，Givens变换后不能保持原矩阵稀疏的性质
- 用途：计算对称稠密矩阵的全部特征值及相应的特征向量

实对称矩阵的Jacobi方法



■ 实对称矩阵的Jacobi迭代算法

Algorithm 20 Jacobi Algorithm

Input:

$n, \mathbf{A}, M, \varepsilon$

- 1: $\mathbf{V} \leftarrow \mathbf{I}_n$;
- 2: $\delta \leftarrow \varepsilon \cdot \|\mathbf{A}\|_F$;
- 3: **for** $k = 1$ to M **do**
- 4: Choose (p, q) such that $|a_{pq}| = \max_{i \neq j} |a_{ij}|$;
- 5: **if** $a_{pq} \neq 0$ **then**
- 6: $s \leftarrow (a_{qq} - a_{pp}) / (2 * a_{pq})$;
- 7: **if** $s \geq 0$ **then**
- 8: $t \leftarrow 1 / (s + \sqrt{1 + s^2})$;
- 9: **else**
- 10: $t \leftarrow 1 / (s - \sqrt{1 + s^2})$;
- 11: **end if**
- 12: $c \leftarrow 1 / \sqrt{1 + t^2}$;
- 13: $d \leftarrow t * c$;
- 14: **else**
- 15: $c \leftarrow 1$;
- 16: $d \leftarrow 0$;
- 17: **end if**
- 18: $\mathbf{A} \leftarrow \mathbf{Q}(p, q, \theta)^T \mathbf{A} \mathbf{Q}(p, q, \theta)$;
- 19: $\mathbf{V} \leftarrow \mathbf{V} \mathbf{Q}(p, q, \theta)$;
- 20: $\text{off}(\mathbf{A}) \leftarrow \|\mathbf{A} - \text{diag}(\mathbf{A})\|_F$
- 21: **if** $\text{off}(\mathbf{A}) < \delta$ **then**
- 22: **break**;
- 23: **end if**
- 24: **end for**

Output:

\mathbf{V}, \mathbf{D}

- Householder矩阵：若 $\mathbf{v} \in \mathbb{R}^n$, $\|\mathbf{v}\|=1$, 则矩阵

$$\mathbf{H} = \mathbf{I} - 2\mathbf{v}\mathbf{v}^T$$

称为 Householder矩阵, 且满足以下的性质:

(1) $\det(\mathbf{H}) = -1$;

(2) \mathbf{H} 为对称正交矩阵;

(3) 若 $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, $\|\mathbf{x}\| = \|\mathbf{y}\|$, 记 $\mathbf{v} = \frac{\mathbf{y} - \mathbf{x}}{\|\mathbf{y} - \mathbf{x}\|}$, $\mathbf{H} = \mathbf{I} - 2\mathbf{v}\mathbf{v}^T$,

则 $\mathbf{H}\mathbf{x} = \mathbf{y}$

- Householder变换: 镜像变换的推广

- (QR分解定理)：设 $A \in \mathbb{C}^{n \times n}$ 是列满秩的，则矩阵 A 能分解为 $A = QR$ 的形式，其中 $Q \in \mathbb{C}^{n \times n}$ 是酉矩阵， $R \in \mathbb{C}^{n \times n}$ 是非奇异上三角阵
推论：设 $A \in \mathbb{R}^{n \times n}$ 是列满秩的，则矩阵 A 能唯一分解为 $A = QR$ 的形式，其中 $Q \in \mathbb{R}^{n \times n}$ 是正交阵， $R \in \mathbb{R}^{n \times n}$ 是非奇异上三角阵
- 对于非满秩的矩阵或一般形式的 $m \times n$ 阶矩阵，有类似的结论
- QR方法：一种求解完全特征值问题的算法，属于变换方法，与Gram-Schmidt正交化过程密切相关

QR方法



■ Householder变换: 设 $0 \neq \mathbf{x} \in \mathbb{R}^n$, 如何构造 \mathbf{v} 及 \mathbf{H} 使得 $\mathbf{H}\mathbf{x} = \alpha \mathbf{e}_1$?

■ 取 $\mathbf{v} = \frac{\mathbf{x} - \|\mathbf{x}\|_2 \mathbf{e}_1}{\|\mathbf{x} - \|\mathbf{x}\|_2 \mathbf{e}_1\|_2}$, $\mathbf{H} = \mathbf{I} - 2\mathbf{v}\mathbf{v}^T$, 则有 $\mathbf{H}\mathbf{x} = \|\mathbf{x}\|_2 \mathbf{e}_1 = \alpha \mathbf{e}_1$

■ 计算QR分解的算法: 设 $\mathbf{A} = [\alpha_1, \alpha_2, \dots, \alpha_n]$

第一步: 取 $\mathbf{v}_1 = (\mathbf{e}_1 - \alpha_1 / \rho_1) / \rho_1$, $\mathbf{Q}_1 = \mathbf{I} - 2\mathbf{v}_1\mathbf{v}_1^T$, 则

$\mathbf{Q}_1\mathbf{A} = [\alpha_1 \mathbf{e}_1, \alpha_2^{(2)}, \dots, \alpha_n^{(2)}]$, $\alpha_1 = -\text{sign}(a_{11}) \cdot \|\mathbf{a}_1\|_2$, $\rho_1 = \|\mathbf{e}_1 - \alpha_1 / \rho_1\|_2$

第 k 步: 记 $\mathbf{a}_k^{(k)} = [a_{1k}^{(k)}, \dots, a_{k-1,k}^{(k)}, \tilde{a}_k^{(k)T}]^T$, 取

$\mathbf{v}_k = (\tilde{\mathbf{a}}_k^{(k)} - \alpha_k \mathbf{e}_k) / \rho_k$, $\mathbf{Q}_k = \begin{bmatrix} \mathbf{I} & \\ & \mathbf{I} - 2\mathbf{v}_k\mathbf{v}_k^T \end{bmatrix}$

$\mathbf{Q}_k \cdots \mathbf{Q}_1 \mathbf{A} = [\alpha_1 \mathbf{e}_1, a_{12}^{(2)} \mathbf{e}_1 + \alpha_2 \mathbf{e}_2, \dots, \sum_{i=1}^{k-1} a_{ik}^{(k)} \mathbf{e}_i + \alpha_k \mathbf{e}_k, \alpha_{k+1}^{(k+1)}, \dots, \alpha_n^{(k+1)}]$

- 不断利用Householder变换:

$$\mathbf{Q}_{n-1} \cdots \mathbf{Q}_1 \mathbf{A} = \begin{bmatrix} \alpha_1 & a_{12}^{(2)} & a_{13}^{(3)} & \cdots & a_{1n}^{(n)} \\ & \alpha_2 & a_{23}^{(3)} & \cdots & a_{2n}^{(n)} \\ & & \alpha_3 & \cdots & a_{3n}^{(n)} \\ & & & \ddots & \vdots \\ & & & & \alpha_n \end{bmatrix} = \mathbf{R}$$

$$\Rightarrow \mathbf{A} = (\mathbf{Q}_{n-1} \cdots \mathbf{Q}_1)^{-1} \mathbf{R} = (\mathbf{Q}_1^{-1} \cdots \mathbf{Q}_{n-1}^{-1}) \mathbf{R} = (\mathbf{Q}_1 \cdots \mathbf{Q}_{n-1}) \mathbf{R} = \mathbf{QR}$$

- 算法时间复杂度: $O(4n^3 / 3)$

- 设矩阵 A 的特征值满足: $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$, 作以下相似正交序列:

$$\begin{cases} \mathbf{A}_k = \mathbf{Q}_k \mathbf{R}_k \\ \mathbf{A}_{k+1} = \mathbf{R}_k \mathbf{Q}_k \end{cases}$$

则当 $k \rightarrow \infty$ 时, 有

$$\begin{cases} a_{ij}^{(k)} \rightarrow 0, 1 \leq j < i \leq n \\ a_{ii}^{(k)} \rightarrow \lambda_i, i = 1, 2, \dots, n \end{cases}$$

收敛于上三角矩阵, 对角线元素即为矩阵的特征值

- 有以下关系:

$$\mathbf{A}_{k+1} = \mathbf{Q}_k^{-1} \mathbf{A}_k \mathbf{Q}_k = \mathbf{Q}_k^{-1} \dots \mathbf{Q}_1^{-1} \mathbf{A} \mathbf{Q}_1 \dots \mathbf{Q}_k$$

$$\tilde{\mathbf{Q}}_k = \mathbf{Q}_1 \dots \mathbf{Q}_k, \tilde{\mathbf{R}}_k = \mathbf{R}_k \dots \mathbf{R}_1$$

$$\Rightarrow \mathbf{A}^{k+1} = \tilde{\mathbf{Q}}_{k+1} \tilde{\mathbf{R}}_{k+1}$$

- QR方法的收敛性证明较困难，参见《Matrix Computations》，4th ed., p368 - p372.
- QR方法的改进
 - 带位移的QR算法
 - 实矩阵的双重步QR算法
- 广义特征值问题
 - QZ算法 (QR算法的推广形式)

特征值与特征向量



■ 三对角阵

- 二分法
- Sturm序列方法
- 分而治之方法

■ 高阶稀疏矩阵

- Lanczos方法
- Arnoldi with restarting
- Implicit restarting
- Jacobi-Davidson方法

■ SVD分解

- Jacobi方法
- QR方法
- Golub-Kahan方法