

第八章 矩阵特征值与特征向量的计算

中国科学技术大学 数学学院

chenxjin@ustc.edu.cn

在线性代数中，对给定的 n 阶方阵 A ，若存在一个(复)数 λ 及一个 n 维非零向量 v 满足

$$Av = \lambda v,$$

则称 λ 为 A 的特征值， v 为属于特征值 λ 的特征向量.

在线性代数中, 对给定的 n 阶方阵 A , 若存在一个(复)数 λ 及一个 n 维非零向量 v 满足

$$Av = \lambda v,$$

则称 λ 为 A 的特征值, v 为属于特征值 λ 的特征向量. 显然:

$$\lambda \text{ 为 } A \text{ 的特征值} \iff \lambda \text{ 满足其特征多项式 } |\lambda I_n - A| = 0$$

但是, 一般高次多项式(非线性方程)的求根通常也很困难, 实际计算中很难通过特征多项式的定义来计算矩阵的特征值。

在线性代数中, 对给定的 n 阶方阵 A , 若存在一个(复)数 λ 及一个 n 维非零向量 v 满足

$$Av = \lambda v,$$

则称 λ 为 A 的特征值, v 为属于特征值 λ 的特征向量. 显然:

$$\lambda \text{ 为 } A \text{ 的特征值} \iff \lambda \text{ 满足其特征多项式 } |\lambda I_n - A| = 0$$

但是, 一般高次多项式(非线性方程)的求根通常也很困难, 实际计算中很难通过特征多项式的定义来计算矩阵的特征值。

比如, 在用迭代法解线性代数方程组时, 其收敛性就与其迭代矩阵 G 的谱半径 $\rho(G)$ 密切相关, 即要求 $\rho(G) < 1$; 而求一个矩阵的谱半径往往需要计算其特征值。

在线性代数中, 对给定的 n 阶方阵 A , 若存在一个(复)数 λ 及一个 n 维非零向量 v 满足

$$Av = \lambda v,$$

则称 λ 为 A 的特征值, v 为属于特征值 λ 的特征向量. 显然:

$$\lambda \text{ 为 } A \text{ 的特征值} \iff \lambda \text{ 满足其特征多项式 } |\lambda I_n - A| = 0$$

但是, 一般高次多项式(非线性方程)的求根通常也很困难, 实际计算中很难通过特征多项式的定义来计算矩阵的特征值。

比如, 在用迭代法解线性代数方程组时, 其收敛性就与其迭代矩阵 G 的谱半径 $\rho(G)$ 密切相关, 即要求 $\rho(G) < 1$; 而求一个矩阵的谱半径往往需要计算其特征值。

本章将介绍一些简单有效的计算矩阵特征值和特征向量的数值方法。

矩阵的按模最大特征值往往具有特殊的重要性，如矩阵的谱半径. 幂法(power method)就是一种最经典的求矩阵按模最大特征值与相应的特征向量的方法. 模最大特征值求出后，其他模较小特征值可以由Wielandt压缩法和幂法依次再求.

矩阵的按模最大特征值往往具有特殊的重要性，如矩阵的谱半径. 幂法(power method)就是一种最经典的求矩阵按模最大特征值与相应的特征向量的方法. 模最大特征值求出后，其他模较小特征值可以由Wielandt压缩法和幂法依次再求.

幂法要求 A 有 n 个线性无关的特征向量，或者说能相似到对角阵. 比如，实对称矩阵或具有互不相同的特征值的矩阵就具有这种性质. 设 A 的特征值和特征向量如下：

$$\begin{array}{ll} \text{特征值:} & |\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n| \\ \text{特征向量:} & v_1 \quad v_2 \quad \cdots \quad v_n \end{array}$$

冪法

幂法

设 A 的 n 个线性无关的特征向量为 $\{v_i\}_{i=1}^n$. 任取非0初始向量 $x^{(0)}$, 设

$$x^{(0)} = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n$$

幂法

设 A 的 n 个线性无关的特征向量为 $\{v_i\}_{i=1}^n$. 任取非0初始向量 $x^{(0)}$, 设

$$x^{(0)} = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n$$

按幂法迭代格式 $x^{(k)} = Ax^{(k-1)}$

幂法

设 A 的 n 个线性无关的特征向量为 $\{v_i\}_{i=1}^n$. 任取非0初始向量 $x^{(0)}$, 设

$$x^{(0)} = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n$$

按幂法迭代格式 $x^{(k)} = Ax^{(k-1)} \Rightarrow x^{(k)} = A^k x^{(0)}$,
得迭代序列 $\{x^{(k)}\}$

$$\begin{aligned} x^{(k)} &= A^k (\alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n) \\ &= \alpha_1 A^k v_1 + \alpha_2 A^k v_2 + \cdots + \alpha_n A^k v_n \\ &= \alpha_1 \lambda_1^k v_1 + \alpha_2 \lambda_2^k v_2 + \cdots + \alpha_n \lambda_n^k v_n \end{aligned}$$

幂法

设 A 的 n 个线性无关的特征向量为 $\{v_i\}_{i=1}^n$. 任取非0初始向量 $x^{(0)}$, 设

$$x^{(0)} = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n$$

按幂法迭代格式 $x^{(k)} = Ax^{(k-1)} \Rightarrow x^{(k)} = A^k x^{(0)}$,
得迭代序列 $\{x^{(k)}\}$

$$\begin{aligned} x^{(k)} &= A^k (\alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n) \\ &= \alpha_1 A^k v_1 + \alpha_2 A^k v_2 + \cdots + \alpha_n A^k v_n \\ &= \alpha_1 \lambda_1^k v_1 + \alpha_2 \lambda_2^k v_2 + \cdots + \alpha_n \lambda_n^k v_n \end{aligned}$$

$x^{(k)}$ 的变化趋势与特征值的分布有关, 幂法根据 $x^{(k)}$ 的变化趋势计算矩阵按模最大的特征值与特征向量.

下面讨论两种比较最简单的情况.

下面讨论两种比较最简单的情况.

① 按模最大的特征值只有一个(即为单实根), $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$.

下面讨论两种比较最简单的情况.

① 按模最大的特征值只有一个(即为单实根), $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$. 则:

$$\begin{aligned}x^{(k)} &= \alpha_1 \lambda_1^k v_1 + \alpha_2 \lambda_2^k v_2 + \cdots + \alpha_n \lambda_n^k v_n \\&= \lambda_1^k \left[\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right]\end{aligned}$$

下面讨论两种比较最简单的情况.

① 按模最大的特征值只有一个(即为单实根), $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$. 则:

$$\begin{aligned}x^{(k)} &= \alpha_1 \lambda_1^k v_1 + \alpha_2 \lambda_2^k v_2 + \cdots + \alpha_n \lambda_n^k v_n \\&= \lambda_1^k \left[\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right]\end{aligned}$$

设 $\alpha_1 \neq 0$. 由于 $\left| \frac{\lambda_i}{\lambda_1} \right| < 1 (i \geq 2)$, 对充分大的 k 有 $\left| \left(\frac{\lambda_i}{\lambda_1} \right)^k \right| \ll 1$, 故

$$\begin{cases} x^{(k)} \approx \lambda_1^k \alpha_1 v_1 \\ x^{(k+1)} \approx \lambda_1^{k+1} \alpha_1 v_1 \approx \lambda_1 x^{(k)} \end{cases}$$

故 $\lambda_1 \approx \frac{x_i^{(k+1)}}{x_i^{(k)}} (1 \leq i \leq n)$, 相应的特征向量近似为 $x^{(k)}$.

下面讨论两种比较最简单的情况.

① 按模最大的特征值只有一个(即为单实根), $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$. 则:

$$\begin{aligned}x^{(k)} &= \alpha_1 \lambda_1^k v_1 + \alpha_2 \lambda_2^k v_2 + \cdots + \alpha_n \lambda_n^k v_n \\&= \lambda_1^k \left[\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right]\end{aligned}$$

设 $\alpha_1 \neq 0$. 由于 $\left| \frac{\lambda_i}{\lambda_1} \right| < 1 (i \geq 2)$, 对充分大的 k 有 $\left| \left(\frac{\lambda_i}{\lambda_1} \right)^k \right| \ll 1$, 故

$$\begin{cases} x^{(k)} \approx \lambda_1^k \alpha_1 v_1 \\ x^{(k+1)} \approx \lambda_1^{k+1} \alpha_1 v_1 \approx \lambda_1 x^{(k)} \end{cases}$$

故 $\lambda_1 \approx \frac{x_i^{(k+1)}}{x_i^{(k)}} (1 \leq i \leq n)$, 相应的特征向量近似为 $x^{(k)}$.

注: $\frac{x_i^{(k+1)}}{x_i^{(k)}}$ 收敛于 λ_1 的速度取决于 $\left| \frac{\lambda_2}{\lambda_1} \right|$ 的大小.

② 按模最大的实特征值有两个，且互为相反数.

② 按模最大的实特征值有两个，且互为相反数. 即

$|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \cdots |\lambda_n|$, $\lambda_1 = -\lambda_2 > 0$. 由

$$x^{(k)} = \lambda_1^k \left[\alpha_1 v_1 + (-1)^k \alpha_2 v_2 + \alpha_3 \left(\frac{\lambda_3}{\lambda_1} \right)^k v_3 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right]$$

② 按模最大的实特征值有两个，且互为相反数. 即

$$|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \cdots |\lambda_n|, \quad \lambda_1 = -\lambda_2 > 0. \text{ 由}$$

$$x^{(k)} = \lambda_1^k \left[\alpha_1 v_1 + (-1)^k \alpha_2 v_2 + \alpha_3 \left(\frac{\lambda_3}{\lambda_1} \right)^k v_3 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right]$$

当 k 充分大时有：（不妨设 $\alpha_1 \neq 0 \neq \alpha_2$ ）

$$x^{(k)} \approx \lambda_1^k \left(\alpha_1 v_1 + \alpha_2 (-1)^k v_2 \right)$$

$x^{(k)}$ 与 $x^{(k+2)}$ 几乎相差一个常数因子 λ_1^2 . 故此时模最大特征值为：

$$x^{(k+2)} \approx \lambda_1^2 x^{(k)} \quad \lambda_1 = \sqrt{x_i^{(k+2)} / x_i^{(k)}}$$

幂法迭代格式 $x^{(k)} = Ax^{(k-1)}$

又由 (利用 $x^{(k+2)} \approx \lambda_1^2 x^{(k)}$)

$$\begin{aligned} A(x^{(k+1)} + \lambda_1 x^{(k)}) &= x^{(k+2)} + \lambda_1 x^{(k+1)} \\ &\approx \lambda_1 (x^{(k+1)} + \lambda_1 x^{(k)}) \end{aligned}$$

$$\begin{aligned} A(x^{(k+1)} - \lambda_1 x^{(k)}) &= x^{(k+2)} - \lambda_1 x^{(k+1)} \\ &\approx -\lambda_1 (x^{(k+1)} - \lambda_1 x^{(k)}) \end{aligned}$$

又由 (利用 $x^{(k+2)} \approx \lambda_1^2 x^{(k)}$)

$$\begin{aligned} A(x^{(k+1)} + \lambda_1 x^{(k)}) &= x^{(k+2)} + \lambda_1 x^{(k+1)} \\ &\approx \lambda_1 (x^{(k+1)} + \lambda_1 x^{(k)}) \end{aligned}$$

$$\begin{aligned} A(x^{(k+1)} - \lambda_1 x^{(k)}) &= x^{(k+2)} - \lambda_1 x^{(k+1)} \\ &\approx -\lambda_1 (x^{(k+1)} - \lambda_1 x^{(k)}) \end{aligned}$$

知, 与 $\lambda_1, \lambda_2 = -\lambda_1$ 对应的特征向量分别为

$$\begin{aligned} v_1 &\approx x^{(k+1)} + \lambda_1 x^{(k)} \\ v_2 &\approx x^{(k+1)} - \lambda_1 x^{(k)} \end{aligned}$$

算法描述

向量序列还有很多更复杂的情况，可参考有关书籍或选用其他方法. 本章只讨论幂法计算的两种最简单和基本情况. 具体算法描述如下：

算法描述

向量序列还有很多更复杂的情况，可参考有关书籍或选用其他方法。本章只讨论幂法计算的两种最简单和基本情况。算法描述如下：

1. 选取初值 $x^{(0)}$ ，构造向量序列 $x^{(k)} = Ax^{(k-1)} = A^k x^{(0)}$ 。

算法描述

向量序列还有很多更复杂的情况，可参考有关书籍或选用其他方法。本章只讨论幂法计算的两种最简单和基本情况。算法描述如下：

1. 选取初值 $x^{(0)}$ ，构造向量序列 $x^{(k)} = Ax^{(k-1)} = A^k x^{(0)}$ 。
2. 若序列表现为，相邻两个向量各个分量比趋于同一常数，则

$$\begin{cases} \lambda_1 \approx x_i^{(k+1)} / x_i^{(k)} \\ v_1 \approx x^{(k)} \end{cases}$$

算法描述

向量序列还有很多更复杂的情况，可参考有关书籍或选用其他方法。本章只讨论幂法计算的两种最简单和基本情况。算法描述如下：

1. 选取初值 $x^{(0)}$ ，构造向量序列 $x^{(k)} = Ax^{(k-1)} = A^k x^{(0)}$ 。
2. 若序列表现为，相邻两个向量各个分量比趋于同一常数，则

$$\begin{cases} \lambda_1 \approx x_i^{(k+1)} / x_i^{(k)} \\ v_1 \approx x^{(k)} \end{cases}$$

3. 若序列表现为，奇偶序列各个分量比分别趋向于常数，则

$$\begin{cases} \lambda_1 \approx \sqrt{x_i^{(k+1)} / x_i^{(k-1)}} \\ v_1 \approx x^{(k+1)} + \lambda_1 x^{(k)} \\ v_2 \approx x^{(k+1)} - \lambda_1 x^{(k)} \end{cases}$$

算法描述

向量序列还有很多更复杂的情况，可参考有关书籍或选用其他方法。本章只讨论幂法计算的两种最简单和基本情况。算法描述如下：

1. 选取初值 $x^{(0)}$ ，构造向量序列 $x^{(k)} = Ax^{(k-1)} = A^k x^{(0)}$ 。
2. 若序列表现为，相邻两个向量各个分量比趋于同一常数，则

$$\begin{cases} \lambda_1 \approx x_i^{(k+1)} / x_i^{(k)} \\ v_1 \approx x^{(k)} \end{cases}$$

3. 若序列表现为，奇偶序列各个分量比分别趋向于常数，则

$$\begin{cases} \lambda_1 \approx \sqrt{x_i^{(k+1)} / x_i^{(k-1)}} \\ v_1 \approx x^{(k+1)} + \lambda_1 x^{(k)} \\ v_2 \approx x^{(k+1)} - \lambda_1 x^{(k)} \end{cases}$$

4. 若序列表现为其他，退出；需采用其他方法

例：计算矩阵 $A = \begin{pmatrix} 1/4 & 1/5 \\ 1/5 & 1/6 \end{pmatrix}$ 的按模最大的特征值和它的特征向量.

例：计算矩阵 $A = \begin{pmatrix} 1/4 & 1/5 \\ 1/5 & 1/6 \end{pmatrix}$ 的按模最大的特征值和它的特征向量.

解：取 $x^{(0)} = (1, 0)^T$ ，计算 $x^{(k)} = Ax^{(k-1)}$ ，结果如下

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_1^{(k)} / x_1^{(k-1)}$	$x_2^{(k)} / x_2^{(k-1)}$
0	1	0		
1	0.25	0.2	0.25	—
2	0.10250	0.083333	0.41	0.41665
3	0.042292	0.034389	0.41260	0.41267
4	0.017451	0.014190	0.41263	0.41263

例：计算矩阵 $A = \begin{pmatrix} 1/4 & 1/5 \\ 1/5 & 1/6 \end{pmatrix}$ 的按模最大的特征值和它的特征向量.

解：取 $x^{(0)} = (1, 0)^T$ ，计算 $x^{(k)} = Ax^{(k-1)}$ ，结果如下

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_1^{(k)} / x_1^{(k-1)}$	$x_2^{(k)} / x_2^{(k-1)}$
0	1	0		
1	0.25	0.2	0.25	—
2	0.10250	0.083333	0.41	0.41665
3	0.042292	0.034389	0.41260	0.41267
4	0.017451	0.014190	0.41263	0.41263

由表知 A 的模最大特征值只有一个，且可取 $\lambda_1 \approx 0.41263$ ，对应特征向量为 $v_1 \approx x^{(4)} = (0.017451, 0.014190)^T$.

幂法的规范运算

幂法的规范运算

为什么要进行**规范**运算？

幂法的规范运算

为什么要进行**规范**运算？前述算法有致命漏洞(BUG)！

幂法的规范运算

为什么要进行规范运算？前述算法有致命漏洞(BUG)！

在幂法中，构造的序列为(这里假设模最大特征值唯一， $\alpha_1 \neq 0$)

$$x^{(k)} = \lambda_1^k \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right)$$

幂法的规范运算

为什么要进行规范运算？前述算法有致命漏洞(BUG)！

在幂法中，构造的序列为(这里假设模最大特征值唯一， $\alpha_1 \neq 0$)

$$x^{(k)} = \lambda_1^k \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right)$$

可以看出，当 $k \rightarrow \infty$ 时， $x_i^{(k)} \rightarrow \begin{cases} 0, & |\lambda_1| < 1 \\ \infty, & |\lambda_1| > 1 \end{cases}$

幂法的规范运算

为什么要进行规范运算？前述算法有致命漏洞(BUG)！

在幂法中，构造的序列为(这里假设模最大特征值唯一， $\alpha_1 \neq 0$)

$$x^{(k)} = \lambda_1^k \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right)$$

可以看出，当 $k \rightarrow \infty$ 时， $x_i^{(k)} \rightarrow \begin{cases} 0, & |\lambda_1| < 1 \\ \infty, & |\lambda_1| > 1 \end{cases}$

因此，若序列收敛较慢的话，很可能造成计算的**溢出**或**归0**。为此，引入**幂法的规范化**。

幂法的规范运算

将迭代格式 $x^{(k+1)} = Ax^{(k)}$ 改为
$$\begin{cases} y^{(0)} = x^{(0)} \neq 0 \\ x^{(k+1)} = Ay^{(k)} \\ y^{(k+1)} = \frac{x^{(k+1)}}{\|x^{(k+1)}\|_\infty} \end{cases},$$

通过观察 $\{y^{(k)}\}$ 的变化规律来确定 A 的模最大特征值和相应的特征向量，称为幂法的规范化.

幂法的规范运算

将迭代格式 $x^{(k+1)} = Ax^{(k)}$ 改为
$$\begin{cases} y^{(0)} = x^{(0)} \neq 0 \\ x^{(k+1)} = Ay^{(k)} \\ y^{(k+1)} = \frac{x^{(k+1)}}{\|x^{(k+1)}\|_\infty} \end{cases}$$

通过观察 $\{y^{(k)}\}$ 的变化规律来确定 A 的模最大特征值和相应的特征向量，称为幂法的规范化.

幂法的规范运算

将迭代格式 $x^{(k+1)} = Ax^{(k)}$ 改为
$$\begin{cases} y^{(0)} = x^{(0)} \neq 0 \\ x^{(k+1)} = Ay^{(k)} \\ y^{(k+1)} = \frac{x^{(k+1)}}{\|x^{(k+1)}\|_\infty} \end{cases}$$

通过观察 $\{y^{(k)}\}$ 的变化规律来确定 A 的模最大特征值和相应的特征向量，称为**幂法的规范化**。

由迭代格式有

$$\begin{cases} y^{(k)} = \frac{Ay^{(k-1)}}{\|x^{(k)}\|_\infty} = \cdots = \frac{A^k y^{(0)}}{\|x^{(k)}\|_\infty \cdots \|x^{(1)}\|_\infty} \\ \|y^{(k)}\|_\infty = 1 \end{cases}$$

$$\Rightarrow y^{(k)} = \frac{A^k y^{(0)}}{\|A^k y^{(0)}\|_\infty}$$

设 $x^{(0)} = y^{(0)} = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n$ (不妨设 $\alpha_1 \neq 0$), 则:

$$y^{(k)} = \frac{\lambda_1^k \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right)}{|\lambda_1^k| \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty}$$

设 $x^{(0)} = y^{(0)} = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n$ (不妨设 $\alpha_1 \neq 0$), 则:

$$y^{(k)} = \frac{\lambda_1^k \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right)}{|\lambda_1|^k \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty}$$

① 若 $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$ (即为单实根), 则:

$$y^{(k)} \rightarrow \begin{cases} (-1)^k \frac{\alpha_1}{|\alpha_1|} \frac{v_1}{\|v_1\|_\infty}, & \lambda_1 < 0 \\ \frac{\alpha_1}{|\alpha_1|} \frac{v_1}{\|v_1\|_\infty}, & \lambda_1 > 0 \end{cases}$$

当序列 $\{y^{(k)}\}$ 有极限或奇偶项分别有符号相反的极限时, 为此种情况。

$$\begin{aligned}
x^{(k+1)} &= Ay^{(k)} = \frac{A^{k+1}y^{(0)}}{\|A^k y^{(0)}\|_\infty} \\
&= \frac{\lambda_1^{k+1} \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^{k+1} v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^{k+1} v_n \right)}{\left| \lambda_1^k \right| \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty} \\
&\Rightarrow \|x^{(k+1)}\|_\infty \approx \frac{\left| \lambda_1^{k+1} \right| \cdot \|\alpha_1 v_1\|_\infty}{\left| \lambda_1^k \right| \cdot \|\alpha_1 v_1\|_\infty} \approx |\lambda_1|
\end{aligned}$$

$$\begin{aligned}
x^{(k+1)} &= Ay^{(k)} = \frac{A^{k+1}y^{(0)}}{\|A^k y^{(0)}\|_\infty} \\
&= \frac{\lambda_1^{k+1} \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^{k+1} v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^{k+1} v_n \right)}{\left| \lambda_1^k \right| \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty} \\
&\Rightarrow \|x^{(k+1)}\|_\infty \approx \frac{\left| \lambda_1^{k+1} \right| \cdot \|\alpha_1 v_1\|_\infty}{\left| \lambda_1^k \right| \cdot \|\alpha_1 v_1\|_\infty} \approx |\lambda_1|
\end{aligned}$$

- $\{y^{(k)}\}$ 收敛时,

$$\begin{aligned}
 x^{(k+1)} &= Ay^{(k)} = \frac{A^{k+1}y^{(0)}}{\|A^k y^{(0)}\|_\infty} \\
 &= \frac{\lambda_1^{k+1} \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^{k+1} v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^{k+1} v_n \right)}{\left| \lambda_1^k \right| \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty} \\
 &\Rightarrow \|x^{(k+1)}\|_\infty \approx \frac{\left| \lambda_1^{k+1} \right| \cdot \|\alpha_1 v_1\|_\infty}{\left| \lambda_1^k \right| \cdot \|\alpha_1 v_1\|_\infty} \approx |\lambda_1|
 \end{aligned}$$

- $\{y^{(k)}\}$ 收敛时, 则有 $\lambda_1 > 0$, 且 $\begin{cases} \lambda_1 \approx \|x^{(k+1)}\|_\infty \\ v_1 \approx y^{(k)} \end{cases}$
- $\{y^{(2k)}\}, \{y^{(2k-1)}\}$ 分别收敛到互为反号的两个向量时,

$$\begin{aligned}
x^{(k+1)} &= Ay^{(k)} = \frac{A^{k+1}y^{(0)}}{\|A^k y^{(0)}\|_\infty} \\
&= \frac{\lambda_1^{k+1} \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^{k+1} v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^{k+1} v_n \right)}{\left| \lambda_1^k \right| \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty} \\
&\Rightarrow \|x^{(k+1)}\|_\infty \approx \frac{\left| \lambda_1^{k+1} \right| \cdot \|\alpha_1 v_1\|_\infty}{\left| \lambda_1^k \right| \cdot \|\alpha_1 v_1\|_\infty} \approx |\lambda_1|
\end{aligned}$$

- $\{y^{(k)}\}$ 收敛时, 则有 $\lambda_1 > 0$, 且 $\begin{cases} \lambda_1 \approx \|x^{(k+1)}\|_\infty \\ v_1 \approx y^{(k)} \end{cases}$
- $\{y^{(2k)}\}, \{y^{(2k-1)}\}$ 分别收敛到互为反号的两个向量时, 则有 $\lambda_1 < 0$, 且 $\begin{cases} \lambda_1 \approx -\|x^{(k+1)}\|_\infty \\ v_1 \approx y^{(k)} \end{cases}$

② 若 $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \cdots \geq |\lambda_n|, \lambda_1 = -\lambda_2$ (设 $\alpha_1 \neq 0 \neq \alpha_2$)

② 若 $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \cdots \geq |\lambda_n|$, $\lambda_1 = -\lambda_2$ (设 $\alpha_1 \neq 0 \neq \alpha_2$)

$$y^{(k)} = \frac{\lambda_1^k \left(\alpha_1 v_1 + \alpha_2 (-1)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right)}{|\lambda_1^k| \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 (-1)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty}$$

则 $y^{(2k)}, y^{(2k-1)}$ 分别收敛到两个向量且这两个向量之间没有反号关系.

② 若 $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \cdots \geq |\lambda_n|$, $\lambda_1 = -\lambda_2$ (设 $\alpha_1 \neq 0 \neq \alpha_2$)

$$y^{(k)} = \frac{\lambda_1^k \left(\alpha_1 v_1 + \alpha_2 (-1)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right)}{|\lambda_1^k| \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 (-1)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty}$$

则 $y^{(2k)}, y^{(2k-1)}$ 分别收敛到两个向量且这两个向量之间没有反号关系.

$$\text{计算} \begin{cases} \tilde{x}^{(k+1)} = Ay^{(k)} \\ \tilde{x}^{(k+2)} = A\tilde{x}^{(k+1)} \end{cases} \quad \tilde{X}^{(k+2)} \approx \lambda_1^2 y^{(k)}$$

② 若 $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \cdots \geq |\lambda_n|$, $\lambda_1 = -\lambda_2$ (设 $\alpha_1 \neq 0 \neq \alpha_2$)

$$y^{(k)} = \frac{\lambda_1^k \left(\alpha_1 v_1 + \alpha_2 (-1)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right)}{|\lambda_1^k| \cdot \left\| \left(\alpha_1 v_1 + \alpha_2 (-1)^k v_2 + \cdots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \right\|_\infty}$$

则 $y^{(2k)}, y^{(2k-1)}$ 分别收敛到两个向量且这两个向量之间没有反号关系.

$$\text{计算 } \begin{cases} \tilde{x}^{(k+1)} = Ay^{(k)} \\ \tilde{x}^{(k+2)} = A\tilde{x}^{(k+1)} \end{cases} \Rightarrow \begin{cases} \lambda_1 \approx \sqrt{\tilde{x}_i^{(k+2)} / y_i^{(k)}} \\ v_1 \approx \tilde{x}^{(k+2)} + \lambda_1 \tilde{x}^{(k+1)} \\ v_2 \approx \tilde{x}^{(k+2)} - \lambda_1 \tilde{x}^{(k+1)} \end{cases}$$

$$\tilde{X}^{(k+2)} \approx \lambda_1^2 y^{(k)}$$

规范化幂法的算法描述

规范化幂法的算法描述

1. 给出初值, 计算序列 $\{y^{(k)}\}$.
2. 若序列收敛, 则 $\lambda_1 \approx \|x^{(k+1)}\|_\infty$, $v_1 \approx y^{(k)}$.
3. 若序列的奇偶序列分别收敛于反号的两个向量, 则 $\lambda_1 \approx -\|x^{(k+1)}\|_\infty$, $v_1 \approx y^{(k)}$.
4. 若序列的奇偶序列分别收敛, 但不收敛于符号相反的向量, 则
$$\begin{cases} \lambda_1 \approx \sqrt{\tilde{x}_i^{(k+2)} / y_i^{(k)}} \\ v_1 \approx \tilde{x}^{(k+2)} + \lambda_1 \tilde{x}^{(k+1)} \\ v_2 \approx \tilde{x}^{(k+2)} - \lambda_1 \tilde{x}^{(k+1)} \end{cases}, \text{ 其中 } \begin{cases} \tilde{x}^{(k+1)} = Ay^{(k)} \\ \tilde{x}^{(k+2)} = A\tilde{x}^{(k+1)} \end{cases}$$
5. 其他情况需另行考虑.

规范化幂法的算法描述

1. 给出初值, 计算序列 $\{y^{(k)}\}$.
2. 若序列收敛, 则 $\lambda_1 \approx \|x^{(k+1)}\|_\infty$, $v_1 \approx y^{(k)}$.
3. 若序列的奇偶序列分别收敛于反号的两个向量, 则 $\lambda_1 \approx -\|x^{(k+1)}\|_\infty$, $v_1 \approx y^{(k)}$.
4. 若序列的奇偶序列分别收敛, 但不收敛于符号相反的向量, 则
$$\begin{cases} \lambda_1 \approx \sqrt{\tilde{x}_i^{(k+2)} / y_i^{(k)}} \\ v_1 \approx \tilde{x}^{(k+2)} + \lambda_1 \tilde{x}^{(k+1)} \\ v_2 \approx \tilde{x}^{(k+2)} - \lambda_1 \tilde{x}^{(k+1)} \end{cases}, \text{ 其中 } \begin{cases} \tilde{x}^{(k+1)} = Ay^{(k)} \\ \tilde{x}^{(k+2)} = A\tilde{x}^{(k+1)} \end{cases}$$
5. 其他情况需另行考虑. **注:** 算法中的序列产生顺序为

$$\begin{aligned} x^{(0)} \rightarrow y^{(0)} = x^{(0)} \rightarrow x^{(1)} = Ay^{(0)} \rightarrow y^{(1)} &= \frac{x^{(1)}}{\|x^{(1)}\|_\infty} \\ \rightarrow x^{(2)} = Ay^{(1)} \rightarrow y^{(2)} &= \frac{x^{(2)}}{\|x^{(2)}\|_\infty} \rightarrow \dots \end{aligned}$$

例 8.4 用规范运算计算矩阵 A 的按模最大的特征值和它的特征向量.

$$A = \begin{pmatrix} 4 & -1 & 1 \\ 16 & -2 & -2 \\ 16 & -3 & -1 \end{pmatrix}$$

$$\begin{aligned}\lambda_1 &\approx \sqrt{\tilde{x}_i^{(k+2)} / y_i^{(k)}} \\ v_1 &\approx \tilde{x}^{(k+2)} + \lambda_1 \tilde{x}^{(k+1)} \\ v_2 &\approx \tilde{x}^{(k+2)} - \lambda_1 \tilde{x}^{(k+1)}\end{aligned}$$

表 8.4

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$y_1^{(k)}$	$y_2^{(k)}$	$y_3^{(k)}$
0	0.5	0.5	1	0.5	0.5	1
1	2.5	5	5.5	0.454545	0.909091	1
2	1.909089	3.454538	3.553628	0.537222	0.972116	1
3	2.176772	4.65132	4.679104	0.465201	0.994091	1
4	2.176772	3.455134	3.461124	0.539392	0.998269	1
5	2.159299	4.633734	4.635485	0.465721	0.999627	1
6	1.862661	3.452282	3.452655	0.539487	0.999892	1
7	2.158056	4.632000	4.632116	0.465890	0.999975	1
8	1.863585	3.454290	3.454315	0.534950	0.999993	1
9	2.157985	4.631926	4.631926	0.465893	0.999999	1
10	1.863573	3.454290	3.454291	0.539495	1	1
11	2.157980	4.631920	4.631920	0.465893	1	1
12	1.863572	3.454288	3.454288			
13	7.454288	16	16			

表 8.4

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$y_1^{(k)}$	$y_2^{(k)}$	$y_3^{(k)}$
0	0.5	0.5	1	0.5	0.5	1
1	2.5	5	5.5	0.454545	0.909091	1
2	1.909089	3.454538	3.553628	0.537222	0.972116	1
3	2.176772	4.65132	4.679104	0.465201	0.994091	1
4	2.176772	3.455134	3.461124	0.539392	0.998269	1
5	2.159299	4.633734	4.635485	0.465721	0.999627	1
6	1.862661	3.452282	3.452655	0.539487	0.999892	1
7	2.158056	4.632000	4.632116	0.465890	0.999975	1
8	1.863585	3.454290	3.454315	0.534950	0.999993	1
9	2.157985	4.631926	4.631926	0.465893	0.999999	1
10	1.863573	3.454290	3.454291	0.539495	1	1
11	2.157980	4.631920	4.631920	0.465893	1	1
12	1.863572	3.454288	3.454288			
13	7.454288	16	16			

$$\lambda_1 = \sqrt{x_2^{(13)} / y_2^{(11)}} = 4, \quad \lambda_2 = -4$$

$$v_1 = X^{(13)} + \lambda_1 X^{(12)} = (14.908576, 29.817152, 29.817152)$$

$$v_2 = X^{(13)} - \lambda_1 X^{(12)} = (0, 2.182848, 2.182848)$$

Remark

1. 实用的是规范化的幂法.
2. 规范化的幂法不需考虑模最大特征值是否为重根.
3. 算法的收敛速度主要由 $|\frac{\lambda_2}{\lambda_1}|$ 决定, 比值越小收敛越快; 由于 $A - p\mathbb{I}_n$ 的所有特征值为 $\{\lambda_i - p\}_{i=1}^n$, 当算法收敛很慢的时候, 可以尝试计算 $A - p\mathbb{I}_n$ 的特征值. (参见课本P174的**原点位移法**)
4. 当 $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \cdots \geq |\lambda_n|$ 且 λ_1, λ_2 共轭时, 也可求出 λ_1, λ_2 和它们的特征向量.
5. 若没有其它更多关于模最大特征值的信息, 只能试着做, 由 $y^{(k)}$ 的规律判断属于哪种情况, 有可能失败, 需另行处理.

Shifted Power Method

原点位移法.

矩阵 $B = A - pI$ 特征值与矩阵 A 的特征值分别为 λ 与 $\lambda - p$. 通过计算矩阵 $A - pI$ 的特征值得到矩阵 A 的特征值的方法称为原点位移法.

若取得的 p 满足

$$\left| \frac{\lambda_2 - p}{\lambda_1 - p} \right| < \left| \frac{\lambda_2}{\lambda_1} \right|$$

则对矩阵 B 的幂法, 收敛会比矩阵 A 的幂法要快. 只有对矩阵特征值的分布有个大致了解, 才能有效选取 p .

例 8.5 设 $A = \begin{pmatrix} -4 & 14 & 0 \\ -5 & 13 & 0 \\ -1 & 0 & 2 \end{pmatrix}$, A 的特征值为 6, 3 和 2, 故取位移量 p

为 2.5, 则 $B = \begin{pmatrix} -6.5 & 14 & 0 \\ -5 & 10.5 & 0 \\ -1 & 0 & -0.5 \end{pmatrix}$, B 的特征值为 3.5, 0.5 和 -0.5 , 则取初

始向量为 $X^{(0)} = (1, 1, 1)^T$, 用规范的幂法运算可得表 8.5.

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$y_1^{(k)}$	$y_2^{(k)}$	$y_3^{(k)}$
0	1	1	1	1	1	1
1	10.	8.	1.	1	0.8	0.1
2	7.2	5.4	-0.8	1	0.75	-0.11111111
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
16	6.00005232	4.28576287	-1.49997383	1	0.71428758	-0.24999346
17	6.00002616	4.28573858	-1.49998692	1	0.71428665	-0.24999673
18	6.00001308	4.28572643	-1.49999346	1	0.71428618	-0.24999836

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$y_1^{(k)}$	$y_2^{(k)}$	$y_3^{(k)}$
0	1	1	1	1	1	1
1	10.	8.	1.	1	0.8	0.1
2	7.2	5.4	-0.8	1	0.75	-0.11111111
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
16	6.00005232	4.28576287	-1.49997383	1	0.71428758	-0.24999346
17	6.00002616	4.28573858	-1.49998692	1	0.71428665	-0.24999673
18	6.00001308	4.28572643	-1.49999346	1	0.71428618	-0.24999836

用原点位移后的规范的幂法运算的结果为表 8.6.

表 1: 表8.6

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$y_1^{(k)}$	$y_2^{(k)}$	$y_3^{(k)}$
0	1	1	1	1	1	1
1	7.5	5.5	-1.5	1	0.73333333	-0.2
2	3.76666667	2.7	-0.9	1	0.71681416	-0.23893805
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
5	3.50071416	2.50053562	-0.87511159	1	0.714293	-0.24998087
6	3.500102	2.5000765	-0.87500956	1	0.71428676	-0.24999545
7	3.50001457	2.50001093	-0.87500228	1	0.71428586	-0.24999961

反幂法

反幂法

用途：计算 A 的模最小特征值与相应的特征向量.

Theorem (on Eigenvalues of Matrix Inverse)

If λ is an eigenvalue of A and if A is nonsingular, then λ^{-1} is an eigenvalue of A^{-1} .

Theorem (on Eigenvalues of Matrix Inverse)

If λ is an eigenvalue of A and if A is nonsingular, then λ^{-1} is an eigenvalue of A^{-1} .

Proof.

Let $Ax = \lambda x$ with $x \neq 0$. Then

$$x = A^{-1}(\lambda x) = \lambda A^{-1}x$$

Theorem (on Eigenvalues of Matrix Inverse)

If λ is an eigenvalue of A and if A is nonsingular, then λ^{-1} is an eigenvalue of A^{-1} .

Proof.

Let $Ax = \lambda x$ with $x \neq 0$. Then

$$x = A^{-1}(\lambda x) = \lambda A^{-1}x \iff A^{-1}x = \lambda^{-1}x$$



反幂法

用途：计算A的模最小特征值与相应的特征向量.

由 $Av = \lambda v \implies A^{-1}v = \frac{1}{\lambda}v$ 知， A 和 A^{-1} 的特征值互为倒数. 于是对 A^{-1} 使用幂法求其模最大特征值 μ ，则 $\frac{1}{\mu}$ 即为 A 的模最小特征值.

反幂法

用途：计算A的模最小特征值与相应的特征向量.

由 $Av = \lambda v \implies A^{-1}v = \frac{1}{\lambda}v$ 知, A 和 A^{-1} 的特征值互为倒数. 于是对 A^{-1} 使用幂法求其模最大特征值 μ , 则 $\frac{1}{\mu}$ 即为 A 的模最小特征值.

规范化的反幂法迭代格式为:

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

反幂法

用途：计算A的模最小特征值与相应的特征向量.

由 $Av = \lambda v \implies A^{-1}v = \frac{1}{\lambda}v$ 知, A 和 A^{-1} 的特征值互为倒数. 于是对 A^{-1} 使用幂法求其模最大特征值 μ , 则 $\frac{1}{\mu}$ 即为 A 的模最小特征值.

规范化的反幂法迭代格式为:

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

为避免求逆运算, 可利用前面的 LU 分解(只需作一次)由

$$Ax^{(k+1)} = LUx^{(k+1)} = y^{(k)} \Rightarrow \text{解出 } x^{(k+1)}$$

Shifted (Inverse) Power Method

Shifted (Inverse) Power Method

- Shifted Power Method

For a given value $\mu \in \mathbb{C}$, we can find/compute the eigenvalue, say λ_k , **farthest from μ by applying the power method** to $A - \mu I$. Here, we suppose that $\exists \epsilon > 0$ s.t. $|\lambda_k - \mu| > \epsilon$ for some eigenvalue λ_k of A , and for all other eigenvalues λ_j of A , we have $0 < |\lambda_j - \mu| < \epsilon$. Note that the power method when applied to $A - \mu I$ finds its eigenvalue $z = \lambda_k - \mu$. So, $\lambda_k = z + \mu$.

Shifted (Inverse) Power Method

- Shifted Power Method

For a given value $\mu \in \mathbb{C}$, we can find/compute the eigenvalue, say λ_k , **farthest from μ by applying the power method** to $A - \mu I$. Here, we suppose that $\exists \epsilon > 0$ s.t. $|\lambda_k - \mu| > \epsilon$ for some eigenvalue λ_k of A , and for all other eigenvalues λ_j of A , we have $0 < |\lambda_j - \mu| < \epsilon$. Note that the power method when applied to $A - \mu I$ finds its eigenvalue $z = \lambda_k - \mu$. So, $\lambda_k = z + \mu$.

- Shifted Inverse Power Method

Similarly, we can find the eigenvalue of A **closest to a given value μ by applying the inverse power method** to $A - \mu I$.

A Summary for Power Method

Method	Equation	Find
Power	$x^{(k+1)} = Ax^{(k)}$	largest eigenvalue $ \lambda_1 $
Inverse Power	$Ax^{(k+1)} = x^{(k)}$	smallest eigenvalue $ \lambda_n $
Shifted Power	$x^{(k+1)} = (A - \mu I)x^{(k)}$	eigenvalue farthest from μ
Shifted Inverse Power	$(A - \mu I)x^{(k+1)} = x^{(k)}$	eigenvalue closest from μ

实对称矩阵的Jacobi方法

问题：求实对称矩阵 A 的所有特征值.

实对称矩阵的Jacobi方法

问题：求实对称矩阵 A 的所有特征值.

由线性代数知：若矩阵 A 实对称，则存在正交阵 Q ，使得

$$Q^T A Q = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix}$$

对角线上的元素即为 A 的所有特征值（ Q 为正交阵 $\implies Q^T = Q^{-1}$ ）.

实对称矩阵的Jacobi方法

问题：求实对称矩阵A的所有特征值.

由线性代数知：若矩阵A实对称，则存在正交阵Q，使得

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix}$$

对角线上的元素即为A的所有特征值（Q为正交阵 $\implies \mathbf{Q}^T = \mathbf{Q}^{-1}$ ）.

直接找Q不大容易. Jacobi方法通过构造一系列Givens正交阵 $\mathbf{Q}_1, \dots, \mathbf{Q}_m$ 对A作正交变换，使得非对角元比重逐次变小，当足够小时，可以近似认为对角元就是A的所有特征值.

不妨设 $p < q$, 则 **Givens** 矩阵 $\mathbf{Q}(p, q, \theta)$ 具有如下形式: (易知它必是一个正交矩阵)

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

p q
 $\downarrow$ $\downarrow$

即 $\mathbf{Q}_{pq} = \sin \theta, \mathbf{Q}_{qp} = -\sin \theta, \mathbf{Q}_{pp} = \mathbf{Q}_{qq} = \cos \theta$

记 $A = (a_{ij})$, $B = \mathbf{Q}^T(p, q, \theta) A \mathbf{Q}(p, q, \theta) = (b_{ij})$. 则:

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

记 $A = (a_{ij})$, $B = \mathbf{Q}^T(p, q, \theta) A \mathbf{Q}(p, q, \theta) = (b_{ij})$. 则:

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

选取 θ 满足:

$$b_{pq} = b_{qp} = a_{pq} \cos 2\theta + \frac{a_{pp} - a_{qq}}{2} \sin 2\theta = 0, \text{ 即 } \cot 2\theta = \frac{a_{qq} - a_{pp}}{2a_{pq}} \triangleq s$$

记 $A = (a_{ij})$, $B = Q^T(p, q, \theta) A Q(p, q, \theta) = (b_{ij})$. 则:

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

选取 θ 满足:

$$b_{pq} = b_{qp} = a_{pq} \cos 2\theta + \frac{a_{pp} - a_{qq}}{2} \sin 2\theta = 0, \text{ 即 } \cot 2\theta = \frac{a_{qq} - a_{pp}}{2a_{pq}} \triangleq s$$

令 $t = \tan \theta$ 并取三角恒等方程(式) $t^2 + 2s \cdot t - 1 = 0$ 的按模较小根

记 $A = (a_{ij})$, $B = Q^T(p, q, \theta) A Q(p, q, \theta) = (b_{ij})$. 则:

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

选取 θ 满足:

$$b_{pq} = b_{qp} = a_{pq} \cos 2\theta + \frac{a_{pp} - a_{qq}}{2} \sin 2\theta = 0, \text{ 即 } \cot 2\theta = \frac{a_{qq} - a_{pp}}{2a_{pq}} \triangleq s$$

令 $t = \tan \theta$ 并取三角恒等方程(式) $t^2 + 2s \cdot t - 1 = 0$ 的按模较小根 (易知, $t_s = -s \pm \sqrt{s^2 + 1}$, 特别地, 若 $s = 0$, 取 $t = 1$),

记 $A = (a_{ij})$, $B = \mathbf{Q}^T(p, q, \theta) A \mathbf{Q}(p, q, \theta) = (b_{ij})$. 则:

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

选取 θ 满足:

$$b_{pq} = b_{qp} = a_{pq} \cos 2\theta + \frac{a_{pp} - a_{qq}}{2} \sin 2\theta = 0, \text{ 即 } \cot 2\theta = \frac{a_{qq} - a_{pp}}{2a_{pq}} \triangleq s$$

令 $t = \tan \theta$ 并取三角恒等方程(式) $t^2 + 2s \cdot t - 1 = 0$ 的按模较小根 (易知, $t_s = -s \pm \sqrt{s^2 + 1}$, 特别地, 若 $s = 0$, 取 $t = 1$), 可得

$$\begin{cases} \cos \theta = \frac{1}{\sqrt{1+t^2}} \\ \sin \theta = \frac{t}{\sqrt{1+t^2}} \end{cases}, \text{ 以及Givens矩阵 } \mathbf{Q}, \text{ 且有 } b_{pq} = b_{qp} = 0;$$

此时, $\begin{cases} b_{pp} = a_{pp} - ta_{pq} \\ b_{qq} = a_{qq} + ta_{pq}. \end{cases}$ 记 $\sigma_1(A) = \sum_{i \neq j} a_{ij}^2$, $\sigma_2(A) = \sum_{i=1}^n a_{ii}^2$, 则可

证: (参见, 第4版, 180页)

$$\begin{cases} \sigma_1(B) \equiv \sum_{i \neq j} b_{ij}^2 = \sigma_1(A) - 2a_{pq}^2 \\ \sigma_2(B) \equiv \sum_{i=1}^n b_{ii}^2 = \sigma_2(A) + 2a_{pq}^2 \end{cases}$$

此时, $\begin{cases} b_{pp} = a_{pp} - ta_{pq} \\ b_{qq} = a_{qq} + ta_{pq} \end{cases}$ 记 $\sigma_1(A) = \sum_{i \neq j} a_{ij}^2$, $\sigma_2(A) = \sum_{i=1}^n a_{ii}^2$, 则可

证: (参见, 第4版, 180页)

$$\begin{cases} \sigma_1(B) \equiv \sum_{i \neq j} b_{ij}^2 = \sigma_1(A) - 2a_{pq}^2 \\ \sigma_2(B) \equiv \sum_{i=1}^n b_{ii}^2 = \sigma_2(A) + 2a_{pq}^2 \end{cases}$$

注: 由 A 变到 $B = Q^T A Q$, 特征值不变, 但对角元所占比重一般会增大 ($a_{pq} = 0$ 时, 比重不变) .

此时, $\begin{cases} b_{pp} = a_{pp} - ta_{pq} \\ b_{qq} = a_{qq} + ta_{pq} \end{cases}$ 记 $\sigma_1(A) = \sum_{i \neq j} a_{ij}^2$, $\sigma_2(A) = \sum_{i=1}^n a_{ii}^2$, 则可

证: (参见, 第4版, 180页)

$$\begin{cases} \sigma_1(B) \equiv \sum_{i \neq j} b_{ij}^2 = \sigma_1(A) - 2a_{pq}^2 \\ \sigma_2(B) \equiv \sum_{i=1}^n b_{ii}^2 = \sigma_2(A) + 2a_{pq}^2 \end{cases}$$

注: 由 A 变到 $B = Q^T A Q$, 特征值不变, 但对角元所占比重一般会增大 ($a_{pq} = 0$ 时, 比重不变).

记 $A^{(0)} = (a_{ij})$, $A^{(k)}$ 为对 $A^{(k-1)}$ 进行一次如前的Givens变换后所得矩阵. 则:

此时, $\begin{cases} b_{pp} = a_{pp} - ta_{pq} \\ b_{qq} = a_{qq} + ta_{pq} \end{cases}$ 记 $\sigma_1(A) = \sum_{i \neq j} a_{ij}^2$, $\sigma_2(A) = \sum_{i=1}^n a_{ii}^2$, 则可

证: (参见, 第4版, 180页)

$$\begin{cases} \sigma_1(B) \equiv \sum_{i \neq j} b_{ij}^2 = \sigma_1(A) - 2a_{pq}^2 \\ \sigma_2(B) \equiv \sum_{i=1}^n b_{ii}^2 = \sigma_2(A) + 2a_{pq}^2 \end{cases}$$

注: 由 A 变到 $B = Q^T A Q$, 特征值不变, 但对角元所占比重一般会增大 ($a_{pq} = 0$ 时, 比重不变).

记 $A^{(0)} = (a_{ij})$, $A^{(k)}$ 为对 $A^{(k-1)}$ 进行一次如前的Givens变换后所得矩阵. 则:

定理: 若 A 实对称, 则 $A^{(k)}$ 收敛于 $\text{diag}\{\lambda_1, \dots, \lambda_n\}$.

此时, $\begin{cases} b_{pp} = a_{pp} - ta_{pq} \\ b_{qq} = a_{qq} + ta_{pq} \end{cases}$ 记 $\sigma_1(A) = \sum_{i \neq j} a_{ij}^2$, $\sigma_2(A) = \sum_{i=1}^n a_{ii}^2$, 则可

证: (参见, 第4版, 180页)

$$\begin{cases} \sigma_1(B) \equiv \sum_{i \neq j} b_{ij}^2 = \sigma_1(A) - 2a_{pq}^2 \\ \sigma_2(B) \equiv \sum_{i=1}^n b_{ii}^2 = \sigma_2(A) + 2a_{pq}^2 \end{cases}$$

注: 由 A 变到 $B = Q^T A Q$, 特征值不变, 但对角元所占比重一般会增大 ($a_{pq} = 0$ 时, 比重不变).

记 $A^{(0)} = (a_{ij})$, $A^{(k)}$ 为对 $A^{(k-1)}$ 进行一次如前的Givens变换后所得矩阵. 则:

定理: 若 A 实对称, 则 $A^{(k)}$ 收敛于 $\text{diag}\{\lambda_1, \dots, \lambda_n\}$.

注: 为了让 $A^{(k)}$ 收敛得更快, 一般选取 p, q ($p \neq q$) 使得对应位置的元素的模最大.

例：用Jacobi方法计算对称阵 $\mathbf{A} = \begin{pmatrix} 4 & 2 & 2 \\ 2 & 5 & 1 \\ 2 & 1 & 6 \end{pmatrix}$ 的全部特征值.

例：用Jacobi方法计算对称阵 $\mathbf{A} = \begin{pmatrix} 4 & 2 & 2 \\ 2 & 5 & 1 \\ 2 & 1 & 6 \end{pmatrix}$ 的全部特征值.

解：记 $A^{(0)} = A$ ，选取 $p = 1, q = 2$. $a_{pq}^{(0)} = a_{12}^{(0)} = 2$,

例：用Jacobi方法计算对称阵 $\mathbf{A} = \begin{pmatrix} 4 & 2 & 2 \\ 2 & 5 & 1 \\ 2 & 1 & 6 \end{pmatrix}$ 的全部特征值.

解：记 $A^{(0)} = A$ ，选取 $p = 1, q = 2$. $a_{pq}^{(0)} = a_{12}^{(0)} = 2$ ，于是有
 $s = \frac{a_{22}^{(0)} - a_{11}^{(0)}}{2a_{12}^{(0)}} = 0.25$ ， t 取为 $t^2 + 2st - 1 = 0$ 的按模较小根，
故 $t = 0.780776$. 进而得到：

$$\cos \theta = (1 + t^2)^{-\frac{1}{2}} = 0.788206, \sin \theta = t \cos \theta = 0.615412$$

此即为Givens变换矩阵所需元素.

$$\mathbf{Q}_1 = \mathbf{Q}(\mathbf{p}, \mathbf{q}, \theta) = \begin{pmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0.788206 & 0.615412 & 0 \\ -0.615412 & 0.788206 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

故有：

$$\mathbf{A}^{(1)} = \mathbf{Q}_1^T \mathbf{A}^{(0)} \mathbf{Q}_1 = \begin{pmatrix} 2.438448 & 0 & 0.961 \\ 0 & 6.561552 & 2.01903 \\ 0.961 & 2.01903 & 6 \end{pmatrix}$$

$$\mathbf{Q}_1 = \mathbf{Q}(\mathbf{p}, \mathbf{q}, \theta) = \begin{pmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0.788206 & 0.615412 & 0 \\ -0.615412 & 0.788206 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

故有：

$$\mathbf{A}^{(1)} = \mathbf{Q}_1^T \mathbf{A}^{(0)} \mathbf{Q}_1 = \begin{pmatrix} 2.438448 & 0 & 0.961 \\ 0 & 6.561552 & 2.01903 \\ 0.961 & 2.01903 & 6 \end{pmatrix}$$

再选取 $p = 2, q = 3$, $a_{pq}^{(1)} = a_{23}^{(1)} = 2.01903$, 类似地可得

$$\mathbf{A}^{(2)} = \begin{pmatrix} 2.438448 & 0.631026 & 0.724794 \\ 0.631026 & 8.320386 & 0 \\ 0.724794 & 0 & 4.241166 \end{pmatrix}$$

$$\mathbf{A}^{(3)} = \begin{pmatrix} 2.183185 & 0.595192 & 0 \\ 0.595192 & 8.320386 & 0.209614 \\ 0 & 0.209614 & 4.496424 \end{pmatrix}$$

$$\mathbf{A}^{(4)} = \begin{pmatrix} 2.125995 & 0 & -0.020048 \\ 0 & 8.377576 & 0.208653 \\ -0.020048 & 0.208653 & 4.496424 \end{pmatrix}$$

$$\mathbf{A}^{(5)} = \begin{pmatrix} 2.125995 & -0.001073 & -0.020019 \\ -0.001073 & 8.388761 & 0 \\ -0.020019 & 0 & 4.485239 \end{pmatrix}$$

$$\mathbf{A}^{(6)} = \begin{pmatrix} 2.125825 & 0 & -0.001072 \\ 0 & 8.388761 & 0.000009 \\ -0.001072 & 0.000009 & 4.485401 \end{pmatrix}$$

$$\mathbf{A}^{(7)} = \begin{pmatrix} 2.125825 & 0 & 0 \\ 0 & 8.388761 & 0.000009 \\ 0 & 0.000009 & 4.485401 \end{pmatrix}$$

$$\mathbf{A}^{(6)} = \begin{pmatrix} 2.125825 & 0 & -0.001072 \\ 0 & 8.388761 & 0.000009 \\ -0.001072 & 0.000009 & 4.485401 \end{pmatrix}$$

$$\mathbf{A}^{(7)} = \begin{pmatrix} 2.125825 & 0 & 0 \\ 0 & 8.388761 & 0.000009 \\ 0 & 0.000009 & 4.485401 \end{pmatrix}$$

从而 A 的特征值可取为

$$\lambda_1 \approx 2.125825, \lambda_2 \approx 8.388761, \lambda_3 \approx 4.485401$$

QR分解 / 正交分解

计算矩阵的全部特征值和特征向量的方法。

基本思想: 任意矩阵 A 与矩阵 $P^{-1}AP$ 的特征值相同

矩阵 A 的正交三角化: 用正交矩阵左乘 A 实现消元,
从而将 A 化为上三角矩阵

正交三角化的主要手段: Householder 变换、Givens 变换,
以及 Gram-Schmidt 正交化过程。

QR方法简介

Theorem (QR分解定理)

若 A 为 n 阶实方阵，则存在正交阵 Q 和上三角阵 R 使得 $A = QR$ 。

QR方法简介

Theorem (QR分解定理)

若 A 为 n 阶实方阵，则存在正交阵 Q 和上三角阵 R 使得 $A = QR$ 。

注：定理中的 Q, R 均有构造性算法求出。建立在QR分解上的QR方法是求矩阵全部特征值的有效方法，在 A 为对称阵时效率至少可与Jacobi 方法相当。

QR方法简介

Theorem (QR分解定理)

若 A 为 n 阶实方阵，则存在正交阵 Q 和上三角阵 R 使得 $A = QR$ 。

注：定理中的 Q, R 均有构造性算法求出. 建立在QR分解上的QR方法是求矩阵全部特征值的有效方法，在 A 为对称阵时效率至少可与Jacobi 方法相当. 基本思路为：令 $A_1 \triangleq A$.

作 A_1 的QR分解 $A_1 = Q_1 \cdot R_1$. $A_2 \triangleq R_1 \cdot Q_1$

→ 作 A_2 的QR分解 $A_2 = Q_2 \cdot R_2$. $A_3 \triangleq R_2 \cdot Q_2$

→ 作 A_3 的QR分解 $A_3 = Q_3 \cdot R_3$. $A_4 \triangleq R_3 \cdot Q_3$

→ ...

QR方法简介

Theorem (QR分解定理)

若 A 为 n 阶实方阵, 则存在正交阵 Q 和上三角阵 R 使得 $A = QR$ 。

注: 定理中的 Q, R 均有构造性算法求出. 建立在QR分解上的QR方法是求矩阵全部特征值的有效方法, 在 A 为对称阵时效率至少可与Jacobi 方法相当. 基本思路为: 令 $A_1 \triangleq A$.

$$\begin{aligned} & \text{作 } A_1 \text{ 的QR分解 } A_1 = Q_1 \cdot R_1. \quad A_2 \triangleq R_1 \cdot Q_1 \\ \longrightarrow & \text{作 } A_2 \text{ 的QR分解 } A_2 = Q_2 \cdot R_2. \quad A_3 \triangleq R_2 \cdot Q_2 \\ \longrightarrow & \text{作 } A_3 \text{ 的QR分解 } A_3 = Q_3 \cdot R_3. \quad A_4 \triangleq R_3 \cdot Q_3 \\ \longrightarrow & \dots \end{aligned}$$

由以上过程构造的矩阵序列 $\{A_k\}$ 全部(正交)相似, 且在 A 满足一定条件时, A_k 收敛到拟上三角阵或对角阵。

定理 设 $A = (a_{ij})_{n \times n} \in \mathbb{R}^{n \times n}$ 满足以下两个条件:

- A 的特征值 λ_j ($j = 1, \dots, n$) 的模互不相等;
- 存在可逆矩阵 P , 使得 $A = P^{-1}DP$, $D = \text{diag}(\lambda_1, \dots, \lambda_n)$; 且 P 可作 LU 分解, 则由 QR 算法产生的矩阵序列 $\{A_k\}_{k=1}^{\infty}$ 收敛于对角矩阵 D .

矩阵的QR分解(正交三角分解)

注：矩阵的QR分解算法实际上得到

$$Q^T A = R$$

矩阵的QR分解(正交三角分解)

注：矩阵的QR分解算法实际上得到

$$Q^T A = R \Leftrightarrow A = QR.$$

矩阵的QR分解(正交三角分解)

注：矩阵的QR分解算法实际上得到

$$Q^T A = R \Leftrightarrow A = QR.$$

Theorem

Let $x, y \in \mathbb{R}^n$ s.t. $\|x\|_2 = \|y\|_2$. Then there exists an orthogonal matrix (正交矩阵) U of the form $I - 2vv^T$ s.t. $Ux = y$, where $v = \frac{x-y}{\|x-y\|_2}$.

矩阵的QR分解(正交三角分解)

注: 矩阵的QR分解算法实际上得到

$$Q^T A = R \Leftrightarrow A = QR.$$

Theorem

Let $x, y \in \mathbb{R}^n$ s.t. $\|x\|_2 = \|y\|_2$. Then there exists an orthogonal matrix (正交矩阵) U of the form $I - 2vv^T$ s.t. $Ux = y$, where $v = \frac{x-y}{\|x-y\|_2}$.

- **Remark:** an orthogonal matrix U of the form $I - 2vv^T$ is called a **Householder matrix**. It's easy to see that $U = U^T$ and $\det u = -1$. $Ux = y$ is a **Householder transformation** (变换) or mirror mapping (镜像映射)

Householder变换

Householder变换也称为初等反射变换

Def. 8.4.1 设向量 $w \in \mathbb{R}^n$, 且 $\|w\|_2^2 = w^T w = 1$, 则称 $H(w) = I - 2ww^T$ 为

*Householder*矩阵, 或初等反射矩阵.

Householder变换

Householder变换也称为初等反射变换

Def. 8.4.1 设向量 $w \in \mathbb{R}^n$, 且 $\|w\|_2^2 = w^T w = 1$, 则称 $H(w) = I - 2ww^T$ 为

*Householder*矩阵, 或初等反射矩阵.

性质:

- $H(w) = H(-w)$
- 对称性: $H^T = H$
- 正交性: $H^T H = I \Rightarrow H^2 = I$
- $\det(H) = -1$
- H 仅有2个不等的特征值 ± 1 ; 其中 $+1$ 是 $n - 1$ 重特征值, -1 是其单特征值, w 为其对应的特征向量

定理 设任意向量 $x \neq 0 \in \mathbb{R}^n$, e 为任意 n 维单位向量, 则存在 Householder 矩阵 H 使得 $Hx = \pm \|x\|_2 e$.

定理 设任意向量 $x \neq 0 \in \mathbb{R}^n$, e 为任意 n 维单位向量, 则存在 Householder 矩阵 H 使得 $Hx = \pm \|x\|_2 e$.

证明: 取 $w = \frac{x - \|x\|_2 e}{\|x - \|x\|_2 e\|_2} \Rightarrow \|w\|_2^2 = w^T w = 1$,

令 $H(w) = I - 2ww^T$, 则有

定理 设任意向量 $x \neq 0 \in \mathbb{R}^n$, e 为任意 n 维单位向量, 则存在 Householder 矩阵 H 使得 $Hx = \pm \|x\|_2 e$.

证明: 取 $w = \frac{x - \|x\|_2 e}{\|x - \|x\|_2 e\|_2} \Rightarrow \|w\|_2^2 = w^T w = 1$,

令 $H(w) = I - 2ww^T$, 则有

$$Hx = x - 2 \frac{x - \|x\|_2 e}{\|x - \|x\|_2 e\|_2^2} (x - \|x\|_2 e)^T x$$

$$\|x - \|x\|_2 e\|_2^2 = (x - \|x\|_2 e)^T (x - \|x\|_2 e)$$

$$= x^T x - 2\|x\|_2 e^T x + \|x\|_2^2 = 2(x^T - \|x\|_2 e^T)x$$

$$\Rightarrow Hx = \|x\|_2 e;$$

取 $e = e_i$, $\alpha = \|x\|_2$ (在实际计算中为了减小舍入误差,
常常取 $\alpha = \text{sign}(x_i)\|x\|_2$) ; 则:

$$w = \frac{x - \alpha e_i}{\|x - \alpha e_i\|_2}, H(w) = I - 2ww^T \Rightarrow: Hx = \alpha e_i$$

若取 $e = e_i$ ($i = 1, \dots, n$), 可将 x 转换为只有第 i 个分量不为0 的向量。

取 $e = e_i$, $\alpha = \|x\|_2$ (在实际计算中为了减小舍入误差,
常常取 $\alpha = \text{sign}(x_i)\|x\|_2$) ; 则:

$$w = \frac{x - \alpha e_i}{\|x - \alpha e_i\|_2}, H(w) = I - 2ww^T \Rightarrow: Hx = \alpha e_i$$

若取 $e = e_i$ ($i = 1, \dots, n$), 可将 x 转换为只有第 i 个分量不为0 的向量。
几何意义: 取 $e = (1, 0, \dots, 0)^T$, 令 $\alpha = \|x\|_2$, 则有:

$$Hx = \alpha e = \alpha(1, 0, \dots, 0)^T$$

如图: Householder矩阵将任意非0向量 x 映射为沿虚线对称的另一个向量
 $\Rightarrow$ Householder变换又称为**初等反射变换** (镜像变换)

如图 8.1 所示, Householder 矩阵把向量 x 映射为沿虚线对称的另一个向量. 因此, Householder 变换又被称为镜像映射.

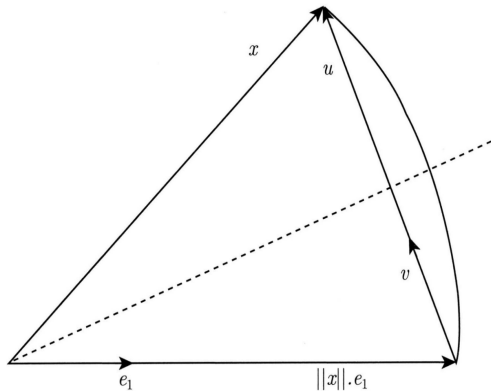


图 8.1 镜像映射

在QR分解的第一步, let

$$v_1 = (A_1 - \beta_1 e^{(1)}) / \|A_1 - \beta_1 e^{(1)}\|_2 \quad \text{where}$$

$$A = (A_1, A_2, \dots, A_n), \quad e^{(1)} = (1, 0, \dots, 0)^T,$$

$$\beta_1 = \|A_1\|_2$$

在QR分解的第一步, let

$$v_1 = (A_1 - \beta_1 e^{(1)}) / \|A_1 - \beta_1 e^{(1)}\|_2 \quad \text{where}$$

$$A = (A_1, A_2, \dots, A_n), \quad e^{(1)} = (1, 0, \dots, 0)^T,$$

$$\beta_1 = \|A_1\|_2 \quad \Rightarrow \quad \|A_1\|_2 = \|\beta_1 e^{(1)}\|_2,$$

在QR分解的第一步, let

$$v_1 = (A_1 - \beta_1 e^{(1)}) / \|A_1 - \beta_1 e^{(1)}\|_2 \quad \text{where}$$

$$A = (A_1, A_2, \dots, A_n), \quad e^{(1)} = (1, 0, \dots, 0)^T,$$

$$\beta_1 = \|A_1\|_2 \quad \Rightarrow \quad \|A_1\|_2 = \|\beta_1 e^{(1)}\|_2,$$

By the previous Theorem, (**Householder 变换**)

$$U_1 A_1 \equiv (I - 2v_1 v_1^T) A_1 = \beta_1 e^{(1)}$$

在QR分解的第一步, let

$$v_1 = (A_1 - \beta_1 e^{(1)}) / \|A_1 - \beta_1 e^{(1)}\|_2 \quad \text{where}$$

$$A = (A_1, A_2, \dots, A_n), \quad e^{(1)} = (1, 0, \dots, 0)^T,$$

$$\beta_1 = \|A_1\|_2 \quad \Rightarrow \quad \|A_1\|_2 = \|\beta_1 e^{(1)}\|_2,$$

By the previous Theorem, (**Householder 变换**)

$$U_1 A_1 \equiv (I - 2v_1 v_1^T) A_1 = \beta_1 e^{(1)}$$

Denote

$$U_1 A = \begin{pmatrix} \beta_1 & * \\ \mathbf{0} & B_{(n-1) \times (n-1)} \end{pmatrix}_{n \times n}$$

and $B_{(n-1) \times (n-1)} = (B_1, B_2, \dots, B_{n-1})$.

and $B_{(n-1) \times (n-1)} = (B_1, B_2, \dots, B_{n-1})$.

Next, construct another orthogonal matrix U_2 s.t.

$$U_2 U_1 A_1 = \beta_1 \mathbf{e}^{(1)}, U_2 U_1 A_2 = (*, \beta_2, 0, \dots, 0)^T$$

and $B_{(n-1) \times (n-1)} = (B_1, B_2, \dots, B_{n-1})$.

Next, construct another orthogonal matrix U_2 s.t.

$$U_2 U_1 A_1 = \beta_1 \mathbf{e}^{(1)}, U_2 U_1 A_2 = (*, \beta_2, 0, \dots, 0)^T$$

$$U_2 = \begin{pmatrix} 1 & \mathbf{0} \\ \mathbf{0} & I_{(n-1) \times (n-1)} - 2\mathbf{v}_2 \mathbf{v}_2^T \end{pmatrix}_{n \times n}, \quad \text{where}$$

$$\mathbf{v}_2 \in \mathbb{R}^{n-1}, \quad \|\mathbf{v}\|_2 = (B_1 - \beta_2 \bar{\mathbf{e}}^{(1)}) / \|B_1 - \beta_2 \bar{\mathbf{e}}^{(1)}\|_2,$$

$$\text{and } \beta_2 = \|B_1\|_2, \bar{\mathbf{e}}^{(1)} = (1, 0, \dots, 0)^T \in \mathbf{R}^{n-1}.$$

In general, the orthogonal matrix Q^T is build up step-by-step as

$$Q^T = U_{n-1} U_{n-2} \cdots U_1$$

where

$$U_k = \begin{pmatrix} I_{(k-1) \times (k-1)} & \mathbf{0} \\ \mathbf{0} & I_{(n-k+1) \times (n-k+1)} - 2\mathbf{v}\mathbf{v}^T \end{pmatrix}_{n \times n}$$

with $\mathbf{v} \in \mathbb{R}^{n-k+1}$ ($k = 1, \dots, n-1$), $\|\mathbf{v}\|_2 = 1$.

In general, the orthogonal matrix Q^T is build up step-by-step as

$$Q^T = U_{n-1} U_{n-2} \cdots U_1$$

where

$$U_k = \begin{pmatrix} I_{(k-1) \times (k-1)} & \mathbf{0} \\ \mathbf{0} & I_{(n-k+1) \times (n-k+1)} - 2\mathbf{v}\mathbf{v}^T \end{pmatrix}_{n \times n}$$

with $\mathbf{v} \in \mathbb{R}^{n-k+1}$ ($k = 1, \dots, n-1$), $\|\mathbf{v}\|_2 = 1$.
Finally, we have

$$Q^T A \equiv U_{n-1} U_{n-2} \cdots U_1 A = R_{n \times n}$$

In general, the orthogonal matrix Q^T is build up step-by-step as

$$Q^T = U_{n-1} U_{n-2} \cdots U_1$$

where

$$U_k = \begin{pmatrix} I_{(k-1) \times (k-1)} & \mathbf{0} \\ \mathbf{0} & I_{(n-k+1) \times (n-k+1)} - 2\mathbf{v}\mathbf{v}^T \end{pmatrix}_{n \times n}$$

with $\mathbf{v} \in \mathbb{R}^{n-k+1}$ ($k = 1, \dots, n-1$), $\|\mathbf{v}\|_2 = 1$.
Finally, we have

$$Q^T A \equiv U_{n-1} U_{n-2} \cdots U_1 A = R_{n \times n} \Leftrightarrow A = QR$$

where

$$R_{n \times n} = \begin{pmatrix} \beta_1 & * & \dots & * & * \\ 0 & \beta_2 & \dots & * & * \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \beta_{n-1} & * \\ 0 & 0 & \dots & 0 & * \end{pmatrix}$$

Example

Use the Householder transformation to find the QR-factorization of the matrix

$$A = \begin{pmatrix} \mathbf{12} & -51 & 4 \\ 6 & \mathbf{167} & -68 \\ -4 & 24 & -\mathbf{41} \end{pmatrix}$$

Solution. Let

$$x = A_1 = (12, 6, -4)^T, \quad \beta_1 = \|A_1\|_2 = 14,$$

$$y = \beta_1 e_1 = (14, 0, 0)^T, \text{ and}$$

$$v = \frac{x-y}{\|x-y\|} = \frac{1}{2\sqrt{14}}(-2, 6, -4)^T = \frac{1}{\sqrt{14}}(-1, 3, -2)^T.$$

So

$$U_1 = I - 2vv^T = \begin{pmatrix} 6/7 & 3/7 & -2/7 \\ 3/7 & -2/7 & 6/7 \\ -2/7 & 6/7 & 3/7 \end{pmatrix}$$

$$U_1 A = \begin{pmatrix} 14 & 21 & -14 \\ 0 & -49 & -14 \\ 0 & 168 & 77 \end{pmatrix}$$

Similarly, one can construct the 2nd orthogonal matrix

$$U_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -7/25 & 24/25 \\ 0 & 24/25 & 7/25 \end{pmatrix}$$

and get

Similarly, one can construct the 2nd orthogonal matrix

$$U_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -7/25 & 24/25 \\ 0 & 24/25 & 7/25 \end{pmatrix}$$

and get

$$U_2 U_1 A = \begin{pmatrix} 14 & 21 & -14 \\ 0 & -175 & -70 \\ 0 & 0 & -35 \end{pmatrix}$$

Similarly, one can construct the 2nd orthogonal matrix

$$U_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -7/25 & 24/25 \\ 0 & 24/25 & 7/25 \end{pmatrix}$$

and get

$$U_2 U_1 A = \begin{pmatrix} 14 & 21 & -14 \\ 0 & -175 & -70 \\ 0 & 0 & -35 \end{pmatrix} = R \Leftrightarrow$$

$$Q^T A = R \Leftrightarrow A = QR$$

- Householder's QR-Factorization

One of the most useful orthogonal factorizations is called **QR-Factorization**. The objective is to factor an $m \times n$ matrix A into a product

$$A_{m \times n} = Q_{m \times m} R_{m \times n}$$

where Q is a **unitary matrix** and R is an $m \times n$ **upper triangular matrix**. The factorization algorithm actually produces

$$Q^* A_{m \times n} = R_{m \times n}$$

