

计算方法

吉林大学计算机科学与技术学院
机器学习研究室
计算方法课程组

第4章 矩阵特征值与特征向量的计算

- 4.1 引入
 - 4.2 幂法及其变体
 - 4.3 Jacobi旋转法
 - 4.4 Householder变换*
 - 4.5 QR方法
 - 4.6 特征值问题的一些应用
-

第4章 矩阵特征值与特征向量的计算

- 4.1 引入
 - 4.2 幂法及其变体
 - 4.3 Jacobi旋转法
 - 4.4 Householder变换*
 - 4.5 QR方法
 - 4.6 特征值问题的一些应用
-

4.1 引入

设 A 为 n 阶方阵, $A = (a_{ij}) \in R^{n \times n}$

若

$$x \in R^n (x \neq 0)$$

有数 λ 使得

$$Ax = \lambda x \quad (4.1.1)$$

则称 λ 为 A 的特征值, x 为相应于 λ 的特征向量。

4.1 引入

特征问题的求解包括

1. 求特征值 λ , 满足

$$\varphi(\lambda) = \det(A - \lambda I) = 0 \quad (4.1.2)$$

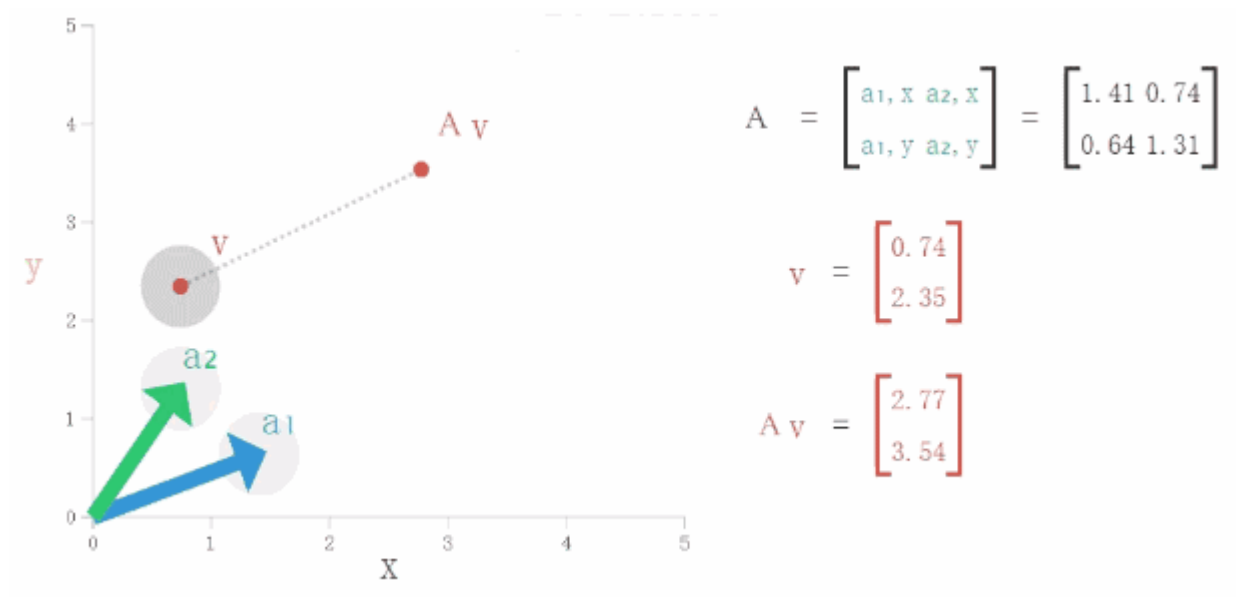
称为**A**的特征多项式

2. 求特征向量 $x \in R^n (x \neq 0)$

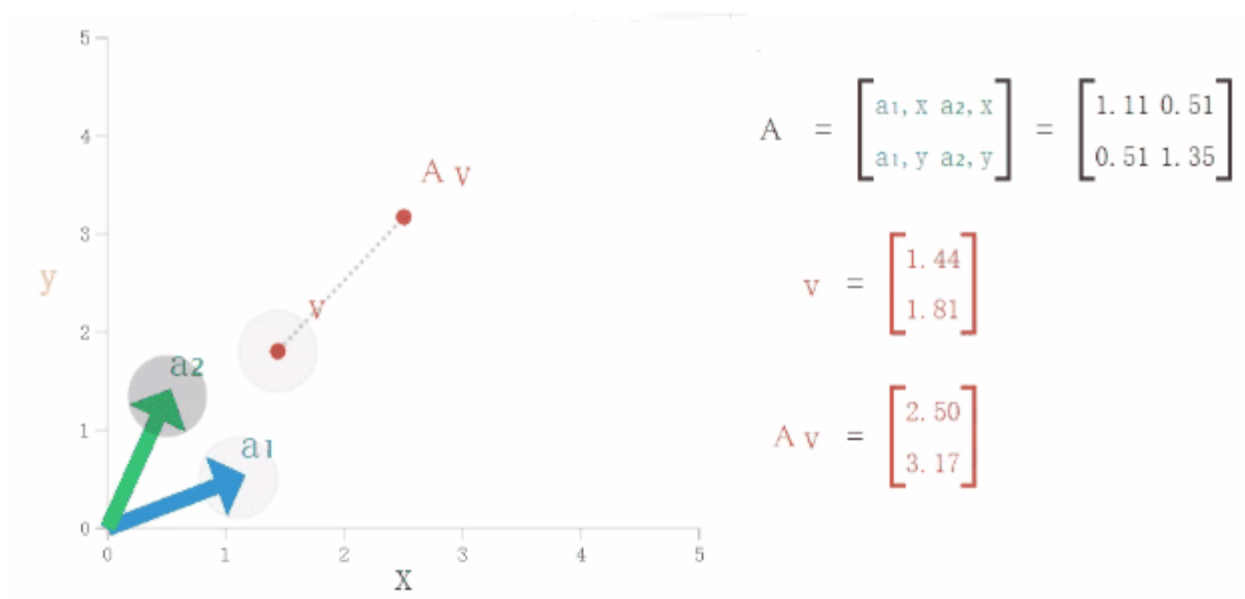
满足齐次方程组

$$(A - \lambda I)x = 0 \quad (4.1.3)$$

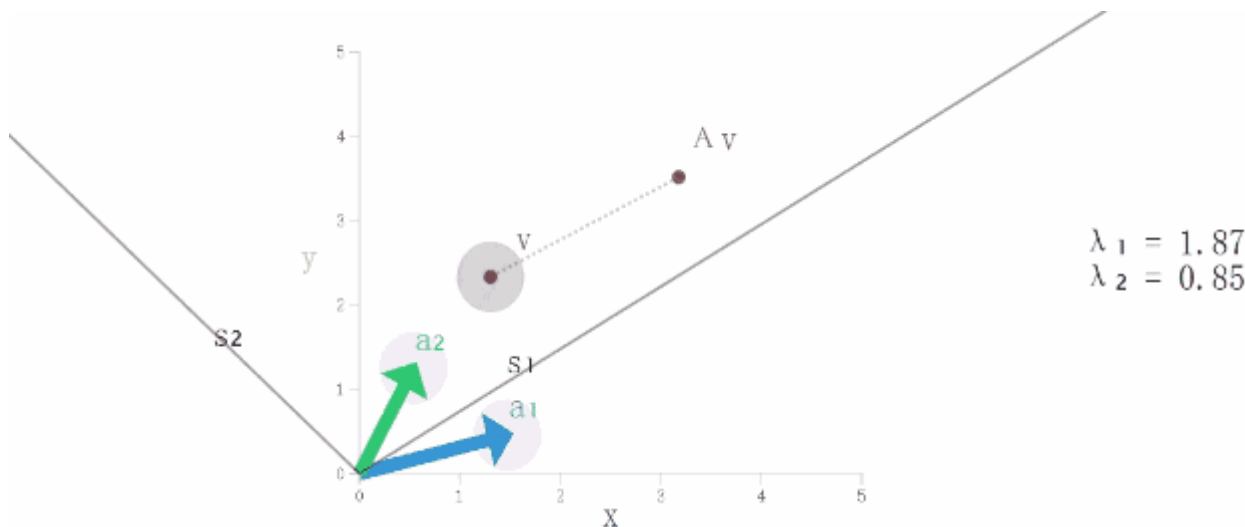
首先，我们通过改变向量 \mathbf{v} 的位置，看看向量 $\mathbf{A}\mathbf{v}$ 有什么变化（矩阵 \mathbf{A} 不动噢）



然后，我们不要动向量 \mathbf{v} ，改变矩阵 \mathbf{A} 每一列（通过移动 $\mathbf{a1}$ 和 $\mathbf{a2}$ ），再看看向量 $\mathbf{A}\mathbf{v}$ 有什么变化



接下来是见证奇迹的时刻！看看超模君的金手指怎么移动向量 \mathbf{v} 使它变成特征向量吧！



4.1 引入

矩阵特征值应用举例



4.1 引入

Google搜索引擎-PageRank

1998年，美国斯坦福大学的Larry Page和Sergey Brin创立了Google公司，他们的核心技术就是通过PageRank技术对海量的网页进行重要性分析。该技术利用网页的相互链接关系对网页进行组织，确定出每个网页的重要级别（PageRank）。当用户进行搜索时，Google找出符合搜索要求的网页，并按它们的PageRank大小依次列出。这样用户一般在显示结果的第一页或者前几页就能找到真正有用的结果。

。

4.1 引入

1998年，美国斯坦福大学的Larry Page和Sergey Brin创立了Google公司，他们的核心技术就是通过PageRank技术对海量的网页进行重要性分析。该技术利用网页的相互链接关系对网页进行组织，确定出每个网页的重要级别（PageRank）。当用户进行搜索时，Google找出符合搜索要求的网页，并按它们的PageRank大小依次列出。这样用户一般在显示结果的第一页或者前几页就能找到真正有用的结果。

形象的解释，PageRank技术的基本原理是：如果网页A链接到网页B，则认为“网页A投了网页B一票”，而且如果网页A是级别高的网页，则网页B的级别也相应地高。

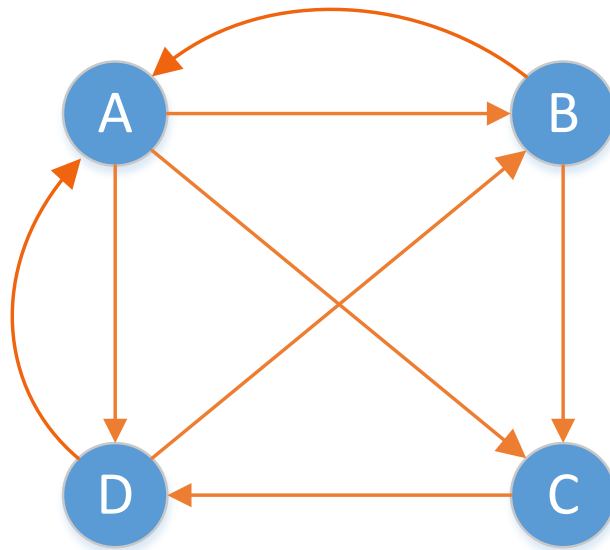
4.1 引入

- 设网页T的重要性为 $\text{Pr}(T)$ ，链出数为 $L(T)$ 。如果网页T存在一个指向网页A的连接，则把T的一部分重要性得分赋予A。值为： $\text{Pr}(T) / L(T)$ 。
- 则A的Pr值为一系列类似于T的页面重要性得分值的累加。于是一个页面的得票数由所有链向它的页面的重要性来决定。
- 一个页面的PageRank是由所有链向它的页面（链入页面）的重要性经过递归算法得到。



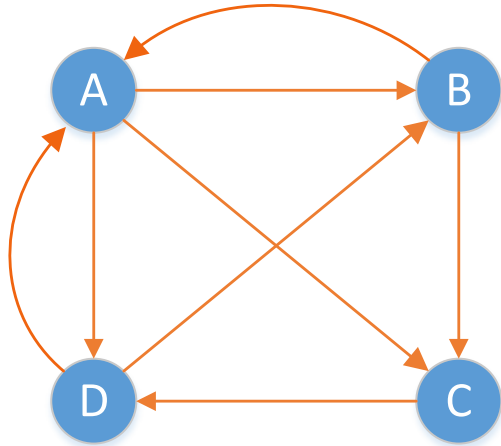
4.1 引入

假设世界上只有四张网页：A、B、C、D，其抽象结构如下图：



AB, AC, AD
BC, BA
CD
DA, DB

4.1 引入

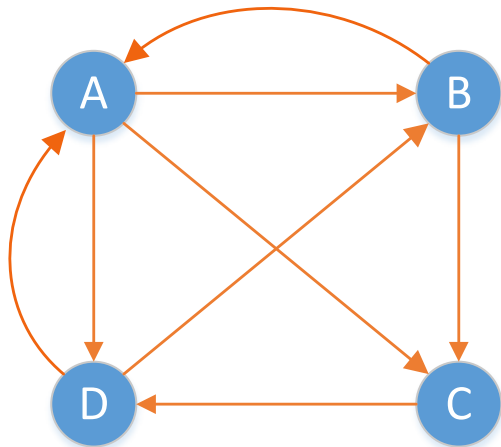


- M--网页间的链接矩阵

$$Pr = [Pr_1, Pr_2, Pr_3, Pr_4]^T$$

$$M = \begin{bmatrix} 0 & 1/2 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \\ 1/3 & 0 & 1 & 0 \end{bmatrix}$$

4.1 引入



$$Pr = [Pr_1, Pr_2, Pr_3, Pr_4]^T$$

$$M = \begin{bmatrix} 0 & 1/2 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \\ 1/3 & 0 & 1 & 0 \end{bmatrix}$$

- M--网页间的链接矩阵

其中，如果 m_{ij} 为0表示第 j 个网页没有到第 i 个网页的链接，否则 m_{ij} 为 $1/L_j$ 。由于所有网页的 \mathbf{Pr} 值可由所有链向它的页面的重要性加和得到，即有

4.1 引入

$$\begin{cases} \frac{1}{2}pr_2 + \frac{1}{2}pr_4 = pr_1 \\ \frac{1}{3}pr_1 + \frac{1}{2}pr_4 = pr_2 \\ \frac{1}{3}pr_1 + \frac{1}{2}pr_2 = pr_3 \\ \frac{1}{3}pr_1 + pr_3 = pr_4 \end{cases}$$

- 问题转换为求解一个4阶的线性方程组的问题
- 解 $\mathbf{Pr} = [0.26471, 0.23529, 0.20588, 0.29412]^T$

4.1 引入

设初始时每个页面的Pr值为 $1/N$ ，即 $1/4$

$$\text{Pr} = \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix}$$

4.1 引入

设初始时每个页面的Pr值为1/N，即1/4

$$\mathbf{Pr} = \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix}$$

注意， M 第一行分别是A、B、C和D转移到页面A的概率，而列向量 \mathbf{Pr} 的元素分别是A、B、C和D当前的Pr值，因此用 M 的第一行乘以 \mathbf{Pr} ，所得结果就是页面A最新Pr值的合理估计，同理， $M \cdot \mathbf{Pr}$ 的结果就分别代表A、B、C、D新Pr值。

$$M \cdot \mathbf{Pr} = \begin{bmatrix} 0.25000 \\ 0.20833 \\ 0.20833 \\ 0.33333 \end{bmatrix}$$

4.1 引入

然后用 M 再乘以这个新的向量，又会产生一个更新的 Pr 值向量。迭代这个过程，可以证明 Pr 最终会收敛，即

$$Pr = M \cdot Pr$$

对于这个例子，最终收敛的结果为

$$[0.26471, 0.23529, 0.20588, 0.29412]^T$$

上述过程即为PageRank算法的最基本思想。当然，真正使用的PageRank算法需要考虑孤立页面问题、作弊问题等复杂情况，感兴趣的同学可参考相关文献。

4.1 引入

上面的 $\mathbf{Pr} = \mathbf{M} \cdot \mathbf{Pr}$ 式表明 \mathbf{Pr} 为矩阵 \mathbf{M} 的特征向量，而前面的迭代求解过程即为第一节所要讲的乘幂法。

事实上，求解特征值问题在科学与工程领域中非常重要，如动力学系统和结构系统中的振动问题，需要求系统的频率与振幅，又如物理学中的某些临界值的确定等，都需要转化为特征值求解问题。对于一个 n 阶矩阵，求解其特征值问题相当于求解一个 n 次方程。我们知道，对于5次及以上的一元高次方程没有通用的代数解法和求根公式，因此必须给出求解特征值问题的数值解法。

第4章 矩阵特征值与特征向量的计算

- 4.1 引入
 - **4.2 幂法及其变体**
 - 4.3 Jacobi旋转法
 - 4.4 Householder变换*
 - 4.5 QR方法
 - 4.6 特征值问题的一些应用
-

4.2.1 乘幂法

乘幂法是用于求解大型稀疏矩阵的主特征值的迭代方法，其特点是公式简单，易于上机实现。

设 $A = (a_{ij}) \in R^{n \times n}$

取初始向量 $x^{(0)} \in R^n$

令 $x^{(1)} = Ax^{(0)}$

$$x^{(2)} = Ax^{(1)}$$

.....

$$x^{(k)} = Ax^{(k-1)} \quad (4.8)$$

由此可得向量序列 $\{x^{(k)}\}$

- 由(4.8), 有

$$x^{(k)} = A(Ax^{(k-2)}) = A^2 x^{(k-2)} = \dots = A^k x^{(0)} \quad (4.9)$$

- 由于 $\{x^{(k)}\}$ 是由 A 的 k 次幂左乘 $x^{(0)}$ 得到的, 称此方法为**乘幂法**
- (4.8)或(4.9)称为**乘幂公式**
- $\{x^{(k)}\}$ 称为**乘幂序列**

4.1.1 乘幂法

下面对乘幂过程进行分析，即讨论当 $k \rightarrow \infty$ 时， $\{x^{(k)}\}$ 与矩阵 A 的特征值及特征向量的关系。

设 A 有完全的特征向量系，且有**唯一**的主特征值（按模最大的特征值），即其特征值满足

$$|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n| \quad (4.10)$$

$\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n$ 为相应的特征向量且线性无关，从而构成 \mathbf{R}^n 上的一组基底。

对任取的初始向量 $\mathbf{x}^{(0)} \in \mathbf{R}^n$ ($\mathbf{x}^{(0)} \neq \mathbf{0}$)，它可按上述基底展开为

$$\mathbf{x}^{(0)} = \alpha_1 \mathbf{v}_1 + \alpha_2 \mathbf{v}_2 + \cdots + \alpha_n \mathbf{v}_n = \sum_{j=1}^n \alpha_j \mathbf{v}_j \quad (4.11)$$

其中 $\alpha_1, \alpha_2, \cdots, \alpha_n$ 为展开系数。

将上面 $x^{(0)}$ 的展开式带入(4.9)式, 得

$$x^{(k)} = A^k x^{(0)} = A^k \sum_{j=1}^n \alpha_j v_j = \sum_{j=1}^n \alpha_j (A^k v_j) = \sum_{j=1}^n \alpha_j \lambda_j^k v_j \quad (4.12)$$

上面的推导利用了 $A^k v_j = \lambda_j^k v_j$ 。由(4.10)式知 $\lambda_1 \neq 0$, 进而有

$$x^{(k)} = \lambda_1^k \left(\alpha_1 v_1 + \sum_{j=2}^n \alpha_j \left(\frac{\lambda_j}{\lambda_1} \right)^k v_j \right) = \lambda_1^k (\alpha_1 v_1 + \varepsilon_k) \quad (4.13)$$

其中

$$\varepsilon_k = \sum_{j=2}^n \alpha_j \left(\frac{\lambda_j}{\lambda_1} \right)^k v_j$$

由于 $\left| \frac{\lambda_j}{\lambda_1} \right| < 1, j = 2, 3, \dots, n$

故当 $k \rightarrow \infty$ 时, ε_k 为一无穷小量。此时

$$x^{(k)} \approx \lambda_1^k \alpha_1 v_1 \quad (4.14)$$

若 $\alpha_1 v_1 \neq 0$, 对 $i=1,2,\dots,n$, 计算相邻迭代向量对应分量比值

$$\frac{x_i^{(k+1)}}{x_i^{(k)}} \approx \frac{\lambda_1^{k+1} \alpha_1 v_1}{\lambda_1^k \alpha_1 v_1} = \lambda_1 \quad (4.15)$$

可见

$$\lim_{k \rightarrow \infty} \frac{x_i^{(k+1)}}{x_i^{(k)}} = \lambda_1 \quad (4.16)$$

这表明主特征值 λ_1 可由（4.15）式近似给出。另一方面，当 k 充分大时，由（4.14）可看出 $x^{(k)}$ 与 \mathbf{v}_1 只相差一常数因子，故可取其为主特征向量的近似。此时，迭代序列的收敛速度取决于 $|\lambda_2/\lambda_1|$ 的大小。

- 乘幂法可用于近似计算矩阵按模最大的一个（或几个）特征值以及相应的特征向量
- 当比值 $r = \left| \frac{\lambda_2}{\lambda_1} \right| \ll 1$ 时，收敛速度快
- 计算公式简便，便于在计算机上实现。

当 $|\lambda_1| > 1$ 时，当 k 充分大时，迭代向量会变得非常大，以致导致计算机上的上溢出，反之，当 $|\lambda_1| < 1$ 时，则会导致计算机上的下溢出。因此，在实际计算时，需要对乘幂法进行规范化。

规范化的乘幂法公式

令 $\max(x)$ 表示向量 x 个分量绝对值最大者，即如果有某个 i_0 ，使得

$$|x_{i_0}| = \max_{1 \leq i \leq n} |x_i|$$

则令 $\max(x) = x_{i_0}$

对任取的初始向量 $x^{(0)}$ ，记

$$y^{(0)} = x^{(0)} / \max(x^{(0)}) \quad (4.18)$$

并定义

$$x^{(1)} = Ay^{(0)} \quad (4.19)$$

规范化的乘幂法

- 取 n 维异于0的初始向量 $x^{(0)}$

—一般地令 $x^{(0)}$ 满足 $\|x^{(0)}\|_{\infty} = 1$

- 对于 $k=0,1,2,\dots$ 按下式迭代

$$\begin{cases} y^{(k)} = x^{(k)} / \max(x^{(k)}) \\ x^{(k+1)} = Ay^{(k)} \end{cases} \quad (k=0,1,\dots)$$

- 终止条件

$$\left| \|x^{(k)}\|_{\infty} - \|x^{(k-1)}\|_{\infty} \right| < \varepsilon$$

- 输出结果

$$\begin{cases} \lambda_1 = \max(x^{(k)}) \\ v_1 = y^{(k)} \end{cases}$$

规范化的乘幂法

- `function [maxEig,eigen_vec] = eigen_pow (A, u, ep)`
- `%To compute matrix A's biggest eigen value(maxEig) and appropriate`
- `%eigen vector(eigen_vec), u is the initial vector, ep is the error threshold`
- `err=100;`
- `u0=u;`
- `[ele,label]=max(abs(u0));`
- `v0 = u0/u0(label);`
- `while err>ep`
- `u1 = A*v0;`
- `[ele,label]=max(abs(u1));`
- `maxEig = u1(label);`
- `v1 = u1 / maxEig ;`
- `err = max (abs(v1-v0));`
- `v0 = v1;`
- `u0 = u1;`
- `end`
- `eigen_vec = v0;`
- `endfunction`

思考：



思考1：如果所取初始向量 $x^{(0)}$ 恰好使得 $\alpha_1=0$ 会怎样？

思考2：如果矩阵 A 的主特征值不唯一会怎样？

即分别当 $\lambda_1 = \lambda_2$, $\lambda_1 = -\lambda_2$, $\lambda_1 = \bar{\lambda}_2$ 时会怎样？

• 例题

- 例4.1 用规范化乘幂法计算矩阵A的主特征值及相应的特征向量

$$A = \begin{bmatrix} -4 & 14 & 0 \\ -5 & 13 & 0 \\ -1 & 0 & 2 \end{bmatrix}$$

$$\lambda_1 = 6, \lambda_2 = 3, \lambda_3 = 2$$

规范化幂法的步骤

- 初始化

$$x^{(0)} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

- 取最大值

$$1 = \max_{1 \leq i \leq n} \{(x^{(0)})_i\}$$

- 规范化

$$y^{(0)} = x^{(0)} / \max_{1 \leq i \leq n} \{(x^{(0)})_i\} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

- 迭代

$$x^{(1)} = Ay^{(0)} = \begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix}$$

规范化幂法的步骤

$$x^{(1)} = \begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix}$$

■ 取最大值

$$10 = \max_{1 \leq i \leq n} \{(x^{(1)})_i\}$$

■ 规范化

$$y^{(1)} = x^{(1)} / \max_{1 \leq i \leq n} \{(x^{(1)})_i\} = \begin{pmatrix} 1 \\ 0.8 \\ 0.1 \end{pmatrix}$$

■ 迭代

$$x^{(2)} = Ay^{(1)} = \begin{pmatrix} 7.2 \\ 5.4 \\ -0.8 \end{pmatrix}$$

规范化幂法计算结果

k	$\max(x^{(k)})$	$y^{(k)} = x^{(k)} / \max(x^{(k)})$	$x^{(k+1)} = Ay^{(k)}$
0	1	(1, 1, 1)	(10, 8, 1)
1	10	(1, 0.8, 0.1)	(7.2, 5.4, -0.8)
2	7.2	(1, 0.75, -0.111 111)	(6.5, 4.75, -1.222 222)
3	6.57	(1, 0.730 769, -0.203 704)	((6.230 766, 4.499 97, -1.407 408)
4	6.230 766	(1, 0.722 222, -0.225 880)	(6.111 108, 4.388 886, -1.145 176 7)
5	6.111 108	(1, 0.718 182, -0.237 561)	(6.054 548, 4.336 336, -1.475 122)
6	6.054 548	(1, 0.716 216, -0.243 639)	(6.027 024, 4.310 808, -1.487 278)
7	6.027 024	(1, 0.715 247, -0.246 768)	(6.013 458, 4.298 211, -1.493 536)
8	6.013 458	(1, 0.714 765, -0.248 366)	(6.006 71, 4.291 945, -1.496 732)
9	6.006 71	(1, 0.714 525, -0.249 177)	(6.003 35, 4.288 825, -1.496 354)
10	6.003 35	(1, 0.714 405, -0.249 586)	(6.001 67, 4.287 265, -1.499 172)
11	6.001 67	(1, 0.714 345, -0.239 792)	(6.000 83, 4.286 485, -1.499 584)
12	6.000 83	(1, 0.714 315, -0.249 896)	

取 $\max(x^{(12)}) = 6.00083$ 作为主特征值 λ_1 的近似值, 与精确值 $\lambda_1 = 6$ 相比, 此近似值的相对误差 ≤ 0.00014 , 相应于 λ_1 的特征向量的近似值取为 $y^{(12)} = (1, 0.714315, -0.249896)^T$.

再思考：

$$M = \begin{bmatrix} 0 & 1/2 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \\ 1/3 & 0 & 1 & 0 \end{bmatrix}$$



为什么一定有Pr，使得

$$M \cdot Pr = Pr$$

也就是说为什么M一定有特征值1？

再思考：

$$M = \begin{bmatrix} 0 & 1/2 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \\ 1/3 & 0 & 1 & 0 \end{bmatrix}$$



为什么一定有Pr，使得

$$M \cdot Pr = Pr$$

也就是说为什么M一定有特征值1？

为什么1是主特征值？

4.1.2 反幂法

4.1.2 反幂法

乘幂法是用于求解按模最大的特征值及其特征向量的一种方法，然而我们有时候要求的是按模最小的特征值及相应的特征向量。根据互逆矩阵之间的特征值互为倒数，可以很容易由乘幂法给出求解按模最小特征值及特征向量的反幂法。

3.1.2 反幂法

- 设 $A \in R^{n \times n}$ 可逆, 若有 $Ax = \lambda x$ ($x \neq \mathbf{0}$), 则 $A^{-1}x = \lambda^{-1}x$. 即 λ 若是 A 的特征值, 则 λ^{-1} 必为 A^{-1} 的特征值, 且特征向量相同。因此, 以 A^{-1} 为乘幂矩阵, 可用乘幂法求得 A^{-1} 的按模最大特征值 $1/\lambda_n$, 于是 λ_n 即为 A 的按模最小特征值, 特征向量相同。
- 对任取的初始向量 $x^{(0)}$, 可按公式

$$x^{(k)} = A^{-1}x^{(k-1)} \quad (4.21)$$

进行迭代, 即为求解矩阵 A 的按模最小特征值的反幂法。

然而公式（4.21）需要计算矩阵的逆，由第一章我们知道求矩阵的逆比较复杂、麻烦，而且，如果矩阵A是稀疏矩阵，求逆之后会破坏其稀疏性，因此将 (4.21)式改写为

$$Ax^{(k)} = x^{(k-1)} \quad (k = 0, 1, 2, \dots) \quad (4.22)$$

注意上式是求解系数矩阵为A的线性方程组。

和乘幂法类似，反幂法也需要进行规范化

$$\begin{cases} y^{(k)} = x^{(k)} / \max(x^{(k)}) \\ Ax^{(k+1)} = y^{(k)} \end{cases} \quad k = 0, 1, \dots \quad (4.23)$$

反幂法的每一步都要求解一个线性方程组，而系数矩阵 A 是保持不变的，因此可以先对矩阵 A 进行LU分解，即 $A=LU$ ，这样可以把每次迭代过程求解一个一般稀疏矩阵的方程组转化为求解两个三角形方程组的形式

$$\begin{cases} L\tilde{x} = y^{(k)} \\ Ux^{(k+1)} = \tilde{x} \end{cases} \quad k = 0, 1, \dots \quad (4.24)$$

当 $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_{n-1}| > |\lambda_n| > 0$ 时，有

$$\lim_{k \rightarrow \infty} \max(x^{(k)}) = 1/\lambda_n \quad (4.25)$$

而 $y^{(k)}$ 即为相应的近似特征向量，收敛速度取决于 $|\lambda_n/\lambda_{n-1}|$ 的大小。

-
- function [lamda,eigen_vec] = anti_eigen_pow (A, u, ep)
 - %To compute matrix A's smallest eigen value to lamda0(lamda) and appropriate
 - %eigen vector(eigen_vec), u is the initial vector, ep is the error threshold
 - err=100;
 - u0=u;
 - [ele,label]=max(abs(u0));
 - v0 = u0/u0(label);
 - while err>ep
 - u1 = A\v0;
 - [ele,label]=max(abs(u1));
 - lamda = u1(label)
 - v1 = u1 / lamda
 - err = max (abs(v1-v0));
 - v0 = v1;
 - u0 = u1;
 - end
 - eigen_vec = v0;
 - lamda = 1/lamda;
 - endfunction
-

例题

例4.2 计算矩阵A的绝对值最小特征值及特征向量，

$$A = \begin{bmatrix} 1 & 1 & 0.5 \\ 1 & 1 & 0.25 \\ 0.5 & 0.25 & 2 \end{bmatrix}$$

例题

例4.2 计算矩阵A的绝对值最小特征值及特征向量，

解：采用反幂法进行计算。

初始化, 令 $x^{(0)} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$

取最大值 $\max_{1 \leq i \leq n} \{(x^{(0)})_i\} = 1$,

规范化 $y^{(0)} = x^{(0)} / \max_{1 \leq i \leq n} \{(x^{(0)})_i\} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$,

迭代求解方程组 $Ax^{(1)} = y^{(0)}$, 可得 $x^{(1)} = \begin{pmatrix} 3 \\ -2 \\ 0 \end{pmatrix}$ 。

取最大值 $\max_{1 \leq i \leq n} \{(x^{(1)})_i\} = 3$,

规范化 $y^{(1)} = x^{(1)} / \max_{1 \leq i \leq n} \{(x^{(1)})_i\} = \begin{pmatrix} 1 \\ -2/3 \\ 0 \end{pmatrix}$,

$$A = \begin{bmatrix} 1 & 1 & 0.5 \\ 1 & 1 & 0.25 \\ 0.5 & 0.25 & 2 \end{bmatrix}$$

1 =

1.00000	0.00000	0.00000
1.00000	-0.00000	1.00000
0.50000	1.00000	0.00000

u =

1.00000	1.00000	0.50000
0.00000	-0.25000	1.75000
0.00000	0.00000	-0.25000

例题

例4.2 计算矩阵A的绝对值最小特征值及特征向量

迭代 $x^{(2)}=Ay^{(1)}=.....$ ，如此反复，直到相邻两步的最大值足够接近停止迭代。迭代结果见表4.1。

表4.1 迭代 k 次后的计算结果

k	$y^{(k)} = x^{(k)}/\max(x^{(k)})$	$\lambda \approx 1/\max(x^{(k)})$
0	$(1, 1, 1)^T$	
1	$(1, 0.6667, 0)^T$	0.333333
2	$(0.7482, 0.6497, 1)^T$	- 0.019608
3	$(1.00000, -0.95425, -0.13072)^T$	-0.016625
4	$(1.00000, -0.95165, -0.12996)^T$	-0.016648
5	$(1.00000, -0.95167, -0.12996)^T$	-0.016647

此时，用规范化反幂法计算绝对值最小特征值的近似值为

$$\lambda_1 \approx 1/\max(x^{(5)}) = -0.016647$$

相应的特征向量为 $v_1 \approx y^{(5)} = (1.00000, -0.95167, -0.12996)^T$ 。

思考：



思考 3： 利用**Gauss**消元法求解（4.22）式与
利用（4.24）式进行求解计算量有何区别？

4.1.3 幂法的平移

若已知 λ 和 x 为矩阵 A 的特征值及相应的特征向量，即有 $Ax=\lambda x$ ，则若令 $B=A-\lambda_0 I$ ，必有 $Bx=(\lambda-\lambda_0)x$ 。也就是说， $\lambda-\lambda_0$ 和 x 分别为矩阵 B 的特征值及相应的特征向量。我们将以矩阵 B 代替矩阵 A 进行乘幂迭代的方法称为乘幂法的原点平移。

4.1.3 幂法的平移

引进参数 λ_0 ，用矩阵 $B=A-\lambda_0I$ 来代替 A 进行乘幂迭代。
设 μ_i ($i=1,2,\dots,n$) 为矩阵 B 的特征值，则 B 与 A 特征值之间应有关系式

$$\mu_i = \lambda_i - \lambda_0 \quad (i=1,2,\dots,n)$$

设 v_i 为矩阵 A 相应于 λ_i 的特征向量，则 v_i 也是 μ_i 的特征向量。

对任何 $x^{(0)} \in R^n$, 关于矩阵B的乘幂公式(4.1.5)可表示为

$$x^{(k)} = B^k x^{(0)} = (A - \lambda_0 I)^k x^{(0)}$$

$$= \mu_1^k \left[\alpha_1 v_1 + \sum_{j=2}^n \alpha_j \left(\frac{\mu_j}{\mu_1} \right)^k v_j \right]$$

$$= (\lambda_1 - \lambda_0)^k \left[\alpha_1 v_1 + \sum_{j=2}^n \alpha_j \left(\frac{\lambda_j - \lambda_0}{\lambda_1 - \lambda_0} \right)^k v_j \right]$$

- 为加速收敛速度，应如此选择参数 λ_0 ，使

$$\omega(\lambda_0) = \max_{2 \leq j \leq n} \left| \frac{\lambda_j - \lambda_0}{\lambda_1 - \lambda_0} \right|^k \quad (4.1.29)$$

达到最小。

思考：



思考4： 是否 $|\lambda_2 - \lambda_0 / \lambda_1 - \lambda_0|$ 越小越好？

- 原点移位法是一个矩阵变换过程，变换简单且不破坏原矩阵的稀疏性。但由于预先不知道特征值的分布，应用有困难。
- 通常对特征值的分布有个大略估计 λ_0
- 设定一个参数值 $\lambda_0 \in (\lambda_n, \lambda_2)$ 进行试算，当所取 λ_0 对迭代有明显加速效应以后再进行计算。

例题

例4.3 采用原点位移的加速法求解矩阵A的最大特征值和特征向量（精确到 10^{-3} ）。

$$A = \begin{bmatrix} -3 & 1 & 0 \\ 1 & -3 & -3 \\ 0 & -3 & 4 \end{bmatrix}$$

例题

例4.3 采用原点位移的加速法求解矩阵A的最大特征值和特征向量（精确到 10^{-3} ）。

$$A = \begin{bmatrix} -3 & 1 & 0 \\ 1 & -3 & -3 \\ 0 & -3 & 4 \end{bmatrix}$$

解：取 $\lambda_0 = -4$ ，则矩阵 $B = A - \lambda_0 I$ 为

$$B = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & -3 \\ 0 & -3 & 8 \end{bmatrix}$$

对矩阵B应用规范化乘幂法公式，计算结果如下表第4、5列所示。 $k=7$ 时，停止迭代，从而可得

$$\lambda_1 = \mu_1 + \lambda_0 \approx 5.125$$

未加速的规范化乘幂法公式计算结果在第2，3列给出，需要迭代到第92步才能终止，可以看出原点位移法的加速效果是显著的。

例题

例4.3 采用原点位移的加速法求解矩阵A的最大特征值和特征向量（精确到 10^{-3} ）。

$$A = \begin{bmatrix} -3 & 1 & 0 \\ 1 & -3 & -3 \\ 0 & -3 & 4 \end{bmatrix}$$

表4.2 原点位移的加速法和规范化乘幂法的计算结果

k	$y_A^{(k)} = x_A^{(k)} / \max(x_A^{(k)})$	$\max(x_A^{(k)})$	$y_B^{(k)} = x_B^{(k)} / \max(x_B^{(k)})$	$\max(x_B^{(k)})$
0	$(1, 1, 1)^T$		$(1, 1, 1)^T$	
1	$(0.4000, 1.0000, -0.2000)^T$	-5.0000	$(0.7483, 0.6497, 1)^T$	1.7866587
...
6	$(-0.2874, 0.1573, 1.0000)^T$	-6.0584	$(-0.0461, -0.3749, 1.0000)^T$	9.1241
7	$(-0.2712, 1.0000, -0.9385)^T$	-3.7593	$(-0.0462, -0.3749, 1.0000)^T$	9.1247
...
91	$(-0.0461, -0.3751, 1.0000)^T$	5.1243		
92	$(-0.0462, -0.3748, 1.0000)^T$	5.1252		

在反幂法中也可以应用原点位移法加速迭代过程
如果矩阵 $(A-\lambda_0 I)^{-1}$ 存在，显然其特征值为

$$\frac{1}{\lambda_1 - \lambda_0}, \frac{1}{\lambda_2 - \lambda_0}, \dots, \frac{1}{\lambda_n - \lambda_0}$$

对应的特征向量为 v_1, v_2, \dots, v_n .

- 反幂法结合原点位移法可以用来求给定特征值的特征向量。
- 设 λ_0 为给定特征值，则应用反幂法， $(A-\lambda_0 I)^{-1}$ 作为乘幂矩阵。 $A-\lambda_0 I$ 的最小特征值

$$\mu = \frac{1}{\lambda_j - \lambda_0} \text{ 即 } \lambda_j = \lambda_0 + \frac{1}{\mu}。$$

当 λ_j 与 λ_0 很接近时， $1/\mu$ 会很小，此时特征向量为 $y^{(k)}$ ，它可以看作是 λ_0 对应的特征向量。只要选择的 λ_0 是 λ_j 一个较好的近似且特征值分离情况较好 $|\lambda_j - \lambda_0| \ll |\lambda_i - \lambda_0|$

一般的, 当 $r = \frac{|\lambda_j - \lambda_0|}{\min |\lambda_i - \lambda_0|}$

很小时，常常只要迭代1-2次就可以完成特征向量的计算。

- 例4.4 用反幂法求

$$A = \begin{pmatrix} 2 & -1 & 0 \\ 0 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}$$

接近2.93的特征值及特征向量。

取 $x^{(0)} = (0, 0, 1)^T$

• 例4.4 用反幂法求

$$A = \begin{pmatrix} 2 & -1 & 0 \\ 0 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}$$

接近2.93的特征值及特征向量。

取 $x^{(0)} = (0, 0, 1)^T$

解：对 $B = A - 2.93I$ 做三角分解得

$$\begin{aligned} A - 2.93I &= \begin{bmatrix} -0.93 & -1 & 0 \\ 0 & -0.93 & -1 \\ 0 & -1 & -0.93 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1/0.93 & 1 \end{bmatrix} \begin{bmatrix} -0.93 & -1 & 0 \\ 0 & -0.93 & -1 \\ 0 & 0 & -0.93 + 1/0.93 \end{bmatrix} \end{aligned}$$

- 例4.4 用反幂法求

$$A = \begin{pmatrix} 2 & -1 & 0 \\ 0 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}$$

接近2.93的特征值及特征向量。

取 $x^{(0)} = (0, 0, 1)^T$

解：

对B用反幂算法迭代，只需迭代3次，就可得
 $\lambda \approx 3.0000954$ ，与准确值3的误差小于 10^{-4} ， $\lambda \approx (1, -0.9992431, 0.9991478)^T$ 与准确值 $(1, -1, 1)^T$ 比较，残差
 $\|\gamma\|_{\infty} < 0.001$.

4.3 Jacobi旋转法

用来求实对称阵的全部特征值以及特征向量的方法。

设 $A = (a_{ij}) \in R^{n \times n}$ 为 n 阶实对称矩阵, 存在正交矩阵 P , 有

$$P^T A P = P^{-1} A P = D = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$$

其中 $\lambda_1, \lambda_2, \dots, \lambda_n$ 为 A 的 n 个特征值, 正交阵 P 的各列 P_1, P_2, \dots, P_n 恰为 A 的相应于 $\lambda_1, \lambda_2, \dots, \lambda_n$ 的特征向量。

• Jacobi旋转法的基本思想

寻找（或构造）一系列正交矩阵 P_1, P_2, \dots, P_n ，对A实施正交相似变换，将A逐渐约化为近似对角阵，从而得其全部特征值的近似值，把逐次得到的正交相似变换矩阵乘在一起，则积矩阵的各列便为所要求的特征向量的近似。

4.3.1 平面旋转阵

$$P = P_{ij} = \begin{bmatrix} 1 & & & & \\ & \dots & & & \\ & & \cos \theta & \dots & \sin \theta \\ & & \dots & \dots & \dots \\ & & -\sin \theta & \dots & \cos \theta \\ & & & \dots & & 1 \end{bmatrix}$$

Diagram labels:

- 第i列 (Column i)
- 第j列 (Column j)
- 第i行 (Row i)
- 第j行 (Row j)
- $(i \neq j)$

P 为正交矩阵 $P^T P = I$

- 用矩阵 \mathbf{P} 对 \mathbf{A} 进行正交相似变换，即令

$$\mathbf{C} = \mathbf{P}^T \mathbf{A} \mathbf{P}$$

则矩阵 \mathbf{A} 与 \mathbf{C} 的元素之间有如下关系式

- (1)第 i 行、 i 列及第 j 行、 j 列元素为

$$c_{ii} = a_{ii} \cos^2 \theta + a_{jj} \sin^2 \theta - a_{ij} \sin 2\theta \quad (4.3.1)$$

$$c_{jj} = a_{ii} \sin^2 \theta + a_{jj} \cos^2 \theta + a_{ij} \sin 2\theta \quad (4.3.2)$$

•(2)第*i*行第*j*列、第*j*行*i*列元素为

$$c_{ij} = c_{ji} = \frac{1}{2}(a_{ii} - a_{jj}) \sin 2\theta + a_{ij} \cos 2\theta \quad (4.3.3)$$

•(3)第*i*行、第*i*列其它元素为

$$c_{ik} = c_{ki} = a_{ik} \cos \theta - a_{jk} \sin \theta \quad (k \neq i, j) \quad (4.3.4)$$

•(4)第*j*行、第*j*列其它元素为

$$c_{jk} = c_{kj} = a_{jk} \cos \theta + a_{ik} \sin \theta \quad (k \neq i, j) \quad (4.3.5)$$

- (5)其它元素为

$$c_{lk} = a_{lk} \quad (l, k \neq i, j)$$

- 可以验证

$$c_{il}^2 + c_{jl}^2 = a_{il}^2 + a_{jl}^2 \quad (l \neq i, j) \quad \mathbf{(4.3.6)}$$

$$c_{ii}^2 + c_{jj}^2 + 2c_{ij}^2 = a_{ii}^2 + a_{jj}^2 + 2a_{ij}^2 \quad \mathbf{(4.3.7)}$$

- 引入记号

$$\sigma(A) = \sum_{i,j=1}^n a_{ij}^2$$

$$\tau(A) = \sum_{\substack{i,j=1 \\ i \neq j}}^n a_{ij}^2$$

则由**(4.3.6)** **(4.3.7)**式可知

$$\sigma(C) = \sigma(A) \quad \tau(C) = \tau(A) - 2a_{ij}^2 + 2C_{ij}^2 \quad \textbf{(4.3.8)}$$

- 若想经过一次正交相似变换，使得

$$c_{ij} = c_{ji} = 0$$

- 只需令(4.3.3)式的右端为零，即选角

$$\theta, \text{使得 } \frac{1}{2}(a_{ii} - a_{jj}) \sin 2\theta + a_{ij} \cos 2\theta = 0$$

$$\text{或 } \tan 2\theta = \frac{2a_{ij}}{a_{jj} - a_{ii}} \quad (|\theta| \leq \frac{\pi}{4}) \quad (4.3.9)$$

4.3.2 Jacobi旋转法

- 由于一次正交相似变换

$$A \rightarrow C = P^T A P$$

可将 A 的两个非对角元素化为零元素。因此可选一系列正交变换阵 P_K ，对 A 进行正交相似变换，直至将 A 化为近似对角阵。

• Jacobi旋转法的具体算法

- 在 $A = A_1 = (a_{ls}^{(1)})$ 的非对角元素中选取绝对值最大的元素，记为

$$\left| a_{i_1, j_1}^{(1)} \right| = \max_{l \neq s} \left| a_{ls}^{(1)} \right| (\neq 0)$$

- 对确定的 i_1, j_1 ，用(4.3.9)式确定出 θ ，从而产生平面旋转阵 $P_1 = P_{i_1, j_1}$ ，约化 A_1 为

$$A_2 = P_1^T A_1 P_1$$

$$\lambda = |a_{jj} - a_{ii}| \quad \mu = \text{sign}(a_{jj} - a_{ii}) \bullet 2a_{ij}$$

$$\tan 2\theta = \mu / \lambda$$

$$\cos 2\theta = \frac{\lambda}{\sqrt{\lambda^2 + \mu^2}} = \omega \quad \cos \theta = \sqrt{\frac{1 + \omega}{2}}$$

$$\sin 2\theta = \frac{\mu}{\sqrt{\lambda^2 + \mu^2}} \quad \sin \theta = \frac{\mu\omega}{\lambda\sqrt{2(1 + \omega)}}$$

且 $A_2 = (a_{ls}^{(2)})$ 的非对角元素 $a_{i_1, j_1}^{(2)} = a_{j_1, i_1}^{(2)} = 0$

- 再在 A_2 的非对角元素中选绝对值最大元素，确定出其位置，按 (3.3.9) 式定 θ 角，产生平面旋转阵 $P_2 = P_{i_2 j_2}$ ，并将 A_2 中的 (i_2, j_2) (j_2, i_2) 位置的元素约化为零，得到

$$A_3 = P_2^T A_2 P_2 = P_2^T P_1^T A_1 P_1 P_2$$

- 如此继续做下去，可得一系列平面旋转阵

$$P_k = P_{i_k, j_k} \quad k=1, 2, \dots$$

使得

$$A_{k+1} = P_k^T \dots P_2^T P_1^T A_1 P_1 P_2 \dots P_k \quad (4.3.10)$$

- 当 k 充分大时， A_k 的对角元素可作为矩阵 A 的特征值的近似值；

由于每一次变换都会将 \mathbf{A} 的两个非对角元素化
为零，由关系式（4.3.8）知

$$\tau(A_k) = \tau(A_{k-1}) - 2\left(a_{ij}^{(k-1)}\right)^2 + 2\left(a_{ij}^{(k)}\right)^2 = \tau(A_{k-1}) - 2\left(a_{ij}^{(k-1)}\right)^2$$

进一步，由于消掉的是非对角元素绝对值最大的
两个，知

$$\left(a_{ij}^{(k-1)}\right)^2 \geq \frac{1}{n(n-1)} \tau(A_{k-1})$$

即有

$$\begin{aligned}\tau(A_k) &\leq \tau(A_{k-1}) - \frac{2}{n(n-1)} \tau(A_{k-1}) \\ &= \left(1 - \frac{2}{n(n-1)}\right) \tau(A_{k-1}) \leq \dots \\ &\leq \left(1 - \frac{2}{n(n-1)}\right)^k \tau(A_1) \rightarrow 0 \quad (k \rightarrow \infty)\end{aligned}$$

• $S_k = P_1 P_2 \dots P_k$ 的各列为相应特征向量的近似值。

• $S_k = P_1 P_2 \dots P_k$ 的各列为相应特征向量的近似值。由 $S_k = S_{k-1} P_k$ 得

$$\begin{cases} S_{ip}^{(k)} = S_{ip}^{(k-1)} \cos \theta - S_{iq}^{(k-1)} \sin \theta, \\ S_{iq}^{(k)} = S_{ip}^{(k-1)} \sin \theta + S_{iq}^{(k-1)} \cos \theta, & i = 1, 2, \dots, n \\ S_{ij}^{(k)} = S_{ij}^{(k-1)}, & j \neq p, q, \end{cases}$$

• $S_k = P_1 P_2 \dots P_k$ 的各列为相应特征向量的近似值。由 $S_k = S_{k-1} P_k$ 得

$$\begin{cases} S_{ip}^{(k)} = S_{ip}^{(k-1)} \cos \theta - S_{iq}^{(k-1)} \sin \theta, \\ S_{iq}^{(k)} = S_{ip}^{(k-1)} \sin \theta + S_{iq}^{(k-1)} \cos \theta, & i = 1, 2, \dots, n \\ S_{ij}^{(k)} = S_{ij}^{(k-1)}, & j \neq p, q, \end{cases}$$

这样就不用保留每步的旋转矩阵 P_k ，而只需存储矩阵 S_k 。当 **Jacobi** 方法完成时， S_k 的列向量就是所求矩阵 **A** 的特征向量。

```

• function [lamda,eigen_vec] = jacobi (A)
• %To compute SYMMETRICAL matrix A's eigen values to lamda0(lamda)
• %eigen vectors(eigen_vec)
• ep=1e-6; %the error threshold and LOWER LIMIT of numbers
• UP=1e9; %The UPPER LIMIT of numbers
• B = A;
• eigen_vec = eye(size(B));
• while 1
•     [max_col,row_label]=max(abs(B-diag(diag(B))));
•     [max_B,col_label]=max(max_col);
•     if max_B<ep
•         break;
•     end
•     tan2theta=2*B(row_label(col_label),col_label)/(B(col_label,col_label) ...
• -B(row_label(col_label),row_label(col_label)));
•     if tan2theta>UP
•         sinTheta=sqrt(2)/2;
•         cosTheta=sinTheta;
•     elseif tan2theta<-UP
•         sinTheta=-sqrt(2)/2;
•         cosTheta=-sinTheta;
•     elseif abs(tan2theta)<ep
•         sinTheta=0;
•         cosTheta=1;
•     else
•         sinTheta=sin(atan(tan2theta)/2);
•         cosTheta=cos(atan(tan2theta)/2);
•     end
•     P=eye(size(B));
•     P(row_label(col_label),row_label(col_label))=cosTheta;
•     P(col_label,col_label)=cosTheta;
•     P(row_label(col_label),col_label)=sinTheta;
•     P(col_label,row_label(col_label))=-sinTheta;
•     eigen_vec = eigen_vec*P;
•     B = P'*B*P;
•     lamda = max(B);
• end
• endfunction

```

•例4.5 用Jacobi旋转法求特征值及特征向量

$$A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}$$

解：因为 $a_{12}^{(0)} = -1$ 是 A 中所有非主对角线元素中绝对值最大的元素，此时 $i=1, j=2$ ， $\cos\theta = \frac{1}{\sqrt{2}}$ ， $\sin\theta = -\frac{1}{\sqrt{2}}$ ，于是相应的平面旋转变换矩阵为

$$P_1 = \begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -1 \\ 0 & 0 & 1 \end{pmatrix}$$

进行旋转变换后可得

$$A_2 = P_1^T A P_1 = \begin{pmatrix} 1 & 0 & -\frac{1}{\sqrt{2}} \\ 0 & 3 & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 2 \end{pmatrix}$$

因为 $a_{13}^{(1)} = -\frac{1}{\sqrt{2}}$ 是 A_2 中所有非主对角线元素中绝对值最大的元素，此时 $i=1, j=3$ ，

$\cos\theta=0.8881$, $\sin\theta=-0.4597$ ，于是相应的平面旋转变换矩阵为 \leftarrow

$$P_2 = \begin{pmatrix} 0.8881 & 0 & -0.4597 \\ 0 & 1 & 0 \\ 0.4597 & 0 & 0.8881 \end{pmatrix} \leftarrow$$

进行旋转变换后可得 \leftarrow

$$A_3 = P_2^T A_2 P_2 = \begin{pmatrix} 0.6340 & -0.3251 & 0 \\ -0.3251 & 3 & -0.6280 \\ 0 & -0.6280 & 2.3660 \end{pmatrix} \leftarrow$$

如此经过 7 次旋转后，得 \leftarrow

$$A_8 = \begin{pmatrix} 0.5858 & 0 & 0 \\ 0 & 3.4142 & 0 \\ 0 & 0 & 2 \end{pmatrix} \leftarrow$$

由此可得矩阵 A 的特征根为

$$\lambda = (0.5858, 3.4142, 2)$$

相应的累次变换矩阵的乘积为

$$P^T = P_1^T P_2^T \cdots P_{10}^T = [p_1, p_2, p_3] = \begin{pmatrix} 0.5 & -0.5 & -0.7 \\ 0.7 & 0.7 & 0 \\ 0.5 & -0.5 & 0.7 \end{pmatrix}$$

即，对应的三个特征向量分别为

$$v_1 = \begin{pmatrix} 0.5 \\ 0.7 \\ 0.5 \end{pmatrix}, \quad v_2 = \begin{pmatrix} -0.5 \\ 0.7 \\ -0.5 \end{pmatrix}, \quad v_3 = \begin{pmatrix} -0.7 \\ 0 \\ 0.7 \end{pmatrix}。$$

- The iteration number is:0
- 0.707107 -0.707107 0.000000
- 0.707107 0.707107 0.000000
- 0.000000 0.000000 1.000000

- 1.000000 0.000000 -0.707107
- 0.000000 3.000000 -0.707107
- -0.707107 -0.707107 2.000000

- The iteration number is:1
- 0.888074 0.000000 -0.459701
- 0.000000 1.000000 0.000000
- 0.459701 0.000000 0.888074

- 0.633975 -0.325058 0.000000
- -0.325058 3.000000 -0.627963
- 0.000000 -0.627963 2.366025

- The iteration number is:2
- 1.000000 0.000000 0.000000
- 0.000000 0.851654 0.524105
- 0.000000 -0.524105 0.851654

- 0.633975 -0.276837 -0.170364
- -0.276837 3.386446 0.000000
- -0.170364 0.000000 1.979579

- The iteration number is:7
- 1.000000 0.000000 -0.000017
- 0.000000 1.000000 0.000000
- 0.000017 0.000000 1.000000

- 0.585786 -0.000000 0.000000
- -0.000000 3.414214 -0.000000
- 0.000000 -0.000000 2.000000
-

- The eigenvalues of A are:
0.585786 3.414214 2.000000
•

- and the corresponding
eigenvectors are:
- 0.500000 -0.500000 -0.707107
- 0.707107 0.707107 -0.000000
- 0.500000 -0.500000 0.707107

Jacobi过关法

- 计算矩阵A的非对角元素平方和，预取阈值

$$\gamma_0 = \left(\sum_{l \neq s}^n a_{ls}^2 \right)^{1/2} = [\tau(A)]^{1/2}$$

$$\gamma_1 = \frac{\gamma_0}{n}$$

- 对A的非对角元素进行扫描，对 $|a_{ij}| \geq \gamma_1$ 的元素进行旋转变换，将 a_{ij} 化为零，其余元素视为“过关”，不作相应的变换

Jacobi过关法

- 当所有非对角元素绝对值都小于 γ_1 后，缩小阈值，重复前面的步骤
- 直到满足 $\gamma_n = \frac{\gamma_0}{n^r} \leq \varepsilon$ ，停止计算

4.4豪斯霍尔德(Householder) 变换*

4.5 QR 方法

4.6 特征值问题的一些应用

END
