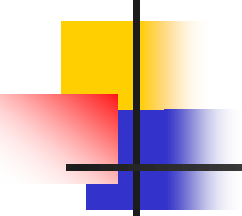


第八章

矩阵特征值和特征向量的计算



- § 8.1 引言
- § 8.2 乘幂法与反幂法
- § 8.3 Jacobi方法



§ 8.1 引言

设 \mathbf{A} 为 $n \times n$ 矩阵，所谓 \mathbf{A} 的特征问题是求数 λ 和非零向量 \mathbf{x} ，使

$$\mathbf{Ax} = \lambda \mathbf{x}$$

成立。数 λ 叫做 \mathbf{A} 的一个特征值，非零向量 \mathbf{x} 叫做与特征值 λ 对应的特征向量。这个问题等价于求使方程组 $(\mathbf{A} - \lambda \mathbf{I})\mathbf{x} = \mathbf{0}$ 有非零解的数 λ 和相应的非零向量 \mathbf{x} 。

线性代数理论中是通过求解特征多项式 $\det(\mathbf{A} - \lambda \mathbf{I}) = 0$ 的零点而得到 λ ，然后通过求解退化的方程组 $(\mathbf{A} - \lambda \mathbf{I})\mathbf{x} = \mathbf{0}$ 而得到非零向量 \mathbf{x} 。当矩阵阶数很高时，这种方法极为困难。目前用数值方法计算矩阵的特征值以及特征向量比较有效的方法是迭代法和变换法。

§ 8.2 乘幂法与反幂法

重点精讲8.1 乘幂法

一、乘幂法

通过求矩阵特征向量求出特征值的一种迭代方法，它用以求按模最大的特征值和相应的特征向量。

设实矩阵 \mathbf{A} 的特征值为 $\lambda_1, \lambda_2, \dots, \lambda_n$ ，相应的特征向量 $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ 线性无关。设 \mathbf{A} 的特征值按模排序为：

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$$

则对任一非零向量 $\mathbf{V}^{(0)} \in \mathbf{R}^n$ ，可以得到：

$$\mathbf{V}^{(0)} = c_1 \mathbf{x}_1 + c_2 \mathbf{x}_2 + \dots + c_n \mathbf{x}_n = \sum_{j=1}^n c_j \mathbf{x}_j$$

令 $\mathbf{V}^{(k)} = \mathbf{A} \mathbf{V}^{(k-1)}, k = 1, 2, \dots$ ，可以构造一个向量序列，

$$\mathbf{V}^{(1)} = \mathbf{A} \mathbf{V}^{(0)} = c_1 \lambda_1 \mathbf{x}_1 + c_2 \lambda_2 \mathbf{x}_2 + \dots + c_n \lambda_n \mathbf{x}_n = \sum_{j=1}^n c_j \lambda_j \mathbf{x}_j$$



乘幂法（续）

$$\mathbf{V}^{(2)} = \mathbf{A}\mathbf{V}^{(1)} = c_1\lambda_1^2\mathbf{x}_1 + c_2\lambda_2^2\mathbf{x}_2 + \cdots + c_n\lambda_n^2\mathbf{x}_n = \sum_{j=1}^n c_j\lambda_j^2\mathbf{x}_j$$

$$\mathbf{V}^{(k)} = \mathbf{A}\mathbf{V}^{(k-1)} = c_1\lambda_1^k\mathbf{x}_1 + c_2\lambda_2^k\mathbf{x}_2 + \cdots + c_n\lambda_n^k\mathbf{x}_n = \sum_{j=1}^n c_j\lambda_j^k\mathbf{x}_j$$

若 $|\lambda_1| > |\lambda_j| (j = 2, 3, \cdots, n)$

$$\mathbf{V}^{(k)} = \mathbf{A}^k\mathbf{V}^{(0)} = \lambda_1^k \left[c_1\mathbf{x}_1 + \sum_{j=2}^n c_j \left(\frac{\lambda_j}{\lambda_1} \right)^k \mathbf{x}_j \right]$$

若 $c_1 \neq 0$, 由于 $\left| \frac{\lambda_j}{\lambda_1} \right| < 1 (j \geq 2)$, 故 k 充分大时,

$$\mathbf{V}^{(k)} \approx \lambda_1^k c_1 \mathbf{x}_1$$

乘幂法（续）

$\mathbf{V}^{(k)}$ 是相应于 λ_1 的近似特征向量

设 $V_l^{(k)}$ 表示 $\mathbf{V}^{(k)}$ 的第 l 个分量。

$$\frac{V_l^{(k+1)}}{V_l^{(k)}} = \frac{\lambda_1^{k+1} c_1(\mathbf{x}_1)_l + (\boldsymbol{\varepsilon}_{k+1})_l}{\lambda_1^k c_1(\mathbf{x}_1)_l + (\boldsymbol{\varepsilon}_k)_l} \approx \lambda_1, l = 1, 2, \dots, n$$

Remark: 具体计算时， $\mathbf{V}^{(0)}$ 的选取很难保证一定有 $c_1 \neq 0$ 。但是，由于舍入误差的影响，只要迭代次数足够多，如 $\mathbf{V}^{(m)} = c'_1 \mathbf{x}_1 + c'_2 \mathbf{x}_2 + \dots + c'_n \mathbf{x}_n$ ，就会有 $c'_1 \neq 0$ ，因而最后结论是成立的。对于 $V_l^{(k)} = 0$ 的情形，由于对任意 l 均有上面的结论，故只要取另外的 l 使 $V_l^{(k)} \neq 0$ 即可。

乘幂法（续）

若主特征值是实数，但不是单根，如 $\lambda_1 = \lambda_2 = \cdots = \lambda_r$ 而 $|\lambda_1| > |\lambda_{r+1}| \geq \cdots \geq |\lambda_n|$ ，则

$$\mathbf{V}^{(k)} = \lambda_1^k \left[\sum_{j=1}^r c_j \mathbf{x}_j + \sum_{j=r+1}^n c_j \left(\frac{\lambda_j}{\lambda_1} \right)^k \mathbf{x}_j \right]$$

$\mathbf{V}^{(k)}$ 仍然是相对于 λ_1 的近似特征向量。

$$\lambda_1 \approx \frac{V_l^{(k+1)}}{V_l^{(k)}}, l = 1, 2, \dots, n$$

Remark: 由于相应于 λ_1 的特征向量空间可能不是一维的，由上式得到的 $\mathbf{V}^{(k)}$ 只是该子空间的一个特征向量。而且不同的 $\mathbf{V}^{(0)}$ 可能得到线性无关的 $\mathbf{V}^{(k)}$ 。

乘幂法（续）

Remark: 以上讨论说明了乘幂法的基本原理。通过上述对乘幂法过程的分析可知，乘幂法本质上是一种迭代法，公式计算简单，便于上机实践，可以方便地用于求矩阵按模最大的一个(或几个)特征值及相应的特征向量。需要注意的是：

- (1) 从理论上讲，对于任意给定的初始向量 $\mathbf{V}^{(0)}$ ，有可能使得组合式中的 $c_1=0$ 或其中前面几个组合系数为零。此时可通过算法的迭代误差自行调整。
- (2) 若特征值计算式中的第 l 个分量为零，此时计算其它的分量比值即可。

乘幂法（续）

(3) 在用乘幂法求矩阵的主特征值及对应的特征向量时，迭代向量的分量 $V_j^{(k)}$ 可能会出现绝对值非常大 (当 $|\lambda_1| > 1$) 或者绝对值非常小 (当 $|\lambda_1| < 1$) 的现象，从而造成计算中溢出的可能。为此，实用中常需要每进行 m 步就对迭代向量 $\mathbf{V}^{(k)}$ 进行一次规范化，即用

$$\tilde{\mathbf{V}}^{(k)} = \frac{\mathbf{V}^{(k)}}{\max \mathbf{V}^{(k)}} \quad (\text{其中 } \max \mathbf{V}^{(k)} \text{ 表示向量 } \mathbf{V}^{(k)} \text{ 的按模最大的分量})$$

代替 $\mathbf{V}^{(k)}$ 继续迭代。由于特征向量允许相差一个常数因子，因此从 $\tilde{\mathbf{V}}^{(k)}$ 开始往后继续迭代与从 $\mathbf{V}^{(k)}$ 继续迭代得到的特征值及特征向量是相同的，但这样规范化后却避免了溢出的可能性。至于 m 取多少，取决于实际情况，可以取 $m=5$ 或 $m=1$ 等等。

乘幂法（续）

(4) 乘幂法的收敛速度取决于 $r = \left| \frac{\lambda_2}{\lambda_1} \right| < 1$ 或 $r = \left| \frac{\lambda_{r+1}}{\lambda_1} \right| < 1$ 或 $r = \left| \frac{\lambda_3}{\lambda_1} \right| < 1$ 的程度。 $r \ll 1$ 时收敛速度较快， $r \approx 1$ 时收敛速度较慢。对收敛较慢的乘幂法，可以采用原点平移法或Rayleigh商加速法来加快其收敛速度，相关内容读者可参阅有关文献。

例题

用乘幂法求如下矩阵的主特征值及其对应的特征向量，要求 $|\lambda_1^{(k+1)} - \lambda_1^{(k)}| \leq 10^{-3}$ 。

$$\mathbf{A} = \begin{pmatrix} 3 & -4 & 3 \\ -4 & 6 & 3 \\ 3 & 3 & 1 \end{pmatrix}$$

解：取初始向量 $\mathbf{V}^{(0)} = (1 \ 1 \ 1)^T$ ，用乘幂法公式进行计算，且取 $\lambda_1^{(k)} = \frac{V_1^{(k)}}{V_1^{(k-1)}}$ ，每迭代5步进行一次规

范化，计算结果见下表。

例题

k	$(V^{(k)})^T$	λ_1
0	(1.00000 1.00000 1.00000)	
1	(2.00000 5.00000 7.00000)	2.0000000
2	(7.00000 43.00000 28.00000)	3.5000000
3	(-67.00000 314.00000 178.00000)	-9.5714286
4	(-923.00000 2686.00000 919.00000)	13.7761194
5	(-10756.00000 22565.00000 6208.00000)	11.6533044
6	(-4.6046532 8.7320186 1.84511411)	9.6600967
7	(-43.2066918 76.3460669 14.2272103)	9.3832672
⋮	⋮	⋮
16	(-5.3591669 8.8695349 1.3387438)	8.8713202
17	(-47.5394092 78.6701089 11.8698478)	8.8706714

故取8.8706714作为主特征值 λ_1 的近似值，相应于 λ_1 的特征向量的近似值取为 $(-47.5394092, 78.6701089, 11.8698478)^T$ 。

二、原点平移法

对收敛较慢的乘幂法，可以采用原点平移法加快其收敛速度。

设矩阵 $\mathbf{B} = \mathbf{A} - p\mathbf{I}$ ，这里 p 是可以选取的参数。当 \mathbf{A} 有特征值 λ_i 及相应特征向量 \mathbf{x}_i 时， \mathbf{B} 有特征值

$\mu_i = \lambda_i - p$ 及其对应的特征向量 \mathbf{x}_i ($i = 1, 2, \dots, n$)。

若 \mathbf{A} 的主特征值为 λ_1 ，则要选择适当的参数 p ，使其满足 $\lambda_1 - p$ 是 \mathbf{B} 的主特征值，即

$$|\lambda_1 - p| > |\lambda_j - p|, j = 2, 3, \dots, n$$

且

$$\max_{2 \leq j \leq n} \left| \frac{\lambda_j - p}{\lambda_1 - p} \right| < \left| \frac{\lambda_2}{\lambda_1} \right|$$



原点平移法（续）

对矩阵 \mathbf{B} 应用乘幂法，可以使得计算的主特征值的过程得到加速。这种方法通常称为**原点平移法**。对于参数 p 的选择依赖于对矩阵 \mathbf{A} 的特征值分布的大致了解。通常可以用Gerschgorin(盖尔)圆盘定理得到矩阵 \mathbf{A} 的特征值分布情况。

原点平移法（续）

定理(Gerschgorin圆盘定理) 设 \mathbf{A} 为 n 阶实矩阵，则
(1) \mathbf{A} 的每一个特征值必定属于下述 n 个闭圆盘(称为Gerschgorin圆)

$$|\lambda - a_{ii}| \leq r_i = \sum_{j=1, j \neq i}^n |a_{ij}| \quad (i = 1, 2, \dots, n)$$

的并集之中；

(2) 由矩阵 \mathbf{A} 的所有Gerschgorin圆组成的连通部分中任取一个，如果它是由 k 个Gerschgorin圆构成，则在这个连通部分中有且仅有 \mathbf{A} 的 k 个特征值(Gerschgorin圆相重时重复计数，特征值相同时也重复计算)。



原点平移法（续）

求得矩阵 \mathbf{B} 的主特征值 $\mu_1 = \lambda_1 - p$ 后，可得矩阵 \mathbf{A} 的主特征值 $\lambda_1 = \mu_1 + p$ ，同时得到对应的特征向量的近似 \mathbf{x}_1 。

Remark: 事实上，如果对于矩阵的特征值能够分离的很清楚，可以利用原点平移法求得矩阵的所有特征值及其相应的特征向量。但需要说明的是，虽然常常能够选择合适的 p 值使乘幂法得到加速，但设计一个自动选择适当参数的过程非常困难。原点平移法的价值不在于直接使用它使迭代过程加速，而在于把原点平移法与其它方法(如反幂法等)结合使用以获得更好的效果。

三、反幂法



计算矩阵按模最小的特征值及相应的特征向量。

设矩阵 \mathbf{A} 非奇异，用 \mathbf{A}^{-1} 代替 \mathbf{A} 作幂的方法就称为反幂法。当 \mathbf{A} 的特征值满足 $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$ 时， \mathbf{A}^{-1} 的特征值满足 $\left|\frac{1}{\lambda_1}\right| \leq \left|\frac{1}{\lambda_2}\right| \leq \dots \leq \left|\frac{1}{\lambda_n}\right|$ ，并且 \mathbf{A}^{-1}

对应于 \mathbf{A}^{-1} 的相应的特征向量是相同的。

对 \mathbf{A}^{-1} 用乘幂法求解时，由乘幂迭代格式

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

便可求出 \mathbf{A}^{-1} 的按模最大特征值 $\frac{1}{\lambda_n}$ ，取倒数，即为矩阵 \mathbf{A} 的按模最小的特征值。

反幂法（续）

对于求逆运算，一方面计算复杂、麻烦，另一方面，有时会破坏的稀疏性，为避免求逆，可将上面的序列改变为

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

利用矩阵的三角分解 $\mathbf{A}=\mathbf{L}\mathbf{U}$ ，则每次迭代只需解二个三角形方程组

$$\begin{cases} \mathbf{L}\mathbf{y} = \mathbf{V}^{(k)} \\ \mathbf{U}\mathbf{V}^{(k+1)} = \mathbf{y} \end{cases}$$

当 $|\lambda_1| \geq |\lambda_2| \geq \cdots > |\lambda_n| > 0$ 时，

反幂法（续）

$$\lim_{k \rightarrow \infty} \frac{V_l^{(k+1)}}{V_l^{(k)}} = \frac{1}{\lambda_n}$$

同时 $\mathbf{V}^{(k+1)}$ 便为所求的近似特征向量，收敛速度依赖于 $\left| \frac{\lambda_n}{\lambda_{n-1}} \right|$ 。

Reamrk: 根据反幂法与乘幂法的上述关系，若已知矩阵的某个近似特征值后，可以结合原点平移法与反幂法来求得矩阵的更精确的特征值与相应的特征向量，且收敛速度快、精度高。

例题

用反幂法计算矩阵A的最接近-13的特征值和对应的特征向量。

$$\mathbf{A} = \begin{bmatrix} -12 & 3 & 3 \\ 3 & 1 & -2 \\ 3 & -2 & 7 \end{bmatrix}$$

解： 反幂法和原点平移法结合计算！

$$\mathbf{B} = \mathbf{A} - (-13)\mathbf{I} = \begin{bmatrix} 1 & 3 & 3 \\ 3 & 14 & -2 \\ 3 & -2 & 20 \end{bmatrix}$$

对矩阵直接三角分解得

$$\mathbf{B} = \mathbf{LU} = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 3 & -11/5 & 1 \end{bmatrix} \begin{bmatrix} 1 & 3 & 3 \\ 0 & 5 & -11 \\ 0 & 0 & -66/5 \end{bmatrix}$$

取初始向量 $V^{(0)}=(1 \ 1 \ 1)^T$ ，用反幂法时需求解线性方程组 $BV^{(1)}=V^{(0)}$ 得到向量 $V^{(1)}$ ，具体求解过程列表如下（仅计算完成表中前三步，小数点后保留四位）：

k	$(V^{(k)})^T$	B^{-1} 矩阵的最大特征值	A 的特征值
0	(1.0000 1.0000 1.0000)		
1	(-2.4545 0.6667 0.4848)	-2.4545	-13.4074
2	(11.2836 -2.6464 -1.9329)	-4.5971	-13.2175
3	(-51.2388 12.0470 8.7939)	-4.5409	-13.2202

从而矩阵 A 的接近于-13的特征值的近似值为-13.2202，相应近似特征向量为 $(-51.2388 \quad 12.0470 \quad 8.7939)^T$
 $=-51.2388*(1 \ -0.2351 \ -0.1716)^T$ 。

§ 8.3 Jacobi方法

用以求实对称矩阵全部特征值和特征向量的一个变换方法。其基本思想是通过一系列正交变换，化实对称矩阵为近似对角阵，这个对角阵的对角元素就是该矩阵的特征值，这些正交矩阵的乘积矩阵的各个列就是相应的特征向量。

有关代数知识：

1. 矩阵 \mathbf{A} 和相似矩阵 $\mathbf{B} = \mathbf{P} \mathbf{A} \mathbf{P}^{-1}$ 的特征值相同。
2. 若矩阵 \mathbf{Q} 满足 $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}$ ，则称 \mathbf{Q} 为正交阵，当然 $\mathbf{Q}^{-1} = \mathbf{Q}^T$ 。正交阵 $\mathbf{Q}_1, \mathbf{Q}_2, \dots, \mathbf{Q}_k$ 的乘积 $\mathbf{Q} = \mathbf{Q}_1 \mathbf{Q}_2 \cdots \mathbf{Q}_k$ 仍为正交阵。

有关代数知识（续）

3. 实对称矩阵的特征值均为实数，且存在标准正交的特征向量系。

4. 若 \mathbf{A} 是上(或下)三角阵或对角阵，则 \mathbf{A} 的主对角元素即为其特征值。

5. 若 \mathbf{A} 为实对称矩阵，则必存在正交阵 \mathbf{Q} ，使得 $\mathbf{Q}\mathbf{A}\mathbf{Q}^T = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ ，其中 $\lambda_i (i = 1, 2, \dots, n)$ 为 \mathbf{A} 的特征值， \mathbf{Q}^T 的各列为相应的特征向量。

6. 设 \mathbf{A} 为实对称矩阵， \mathbf{Q} 为任意正交矩阵，则

$$\|\mathbf{A}\|_F^2 = \|\mathbf{Q}\mathbf{A}\|_F^2 = \|\mathbf{A}\mathbf{Q}\|_F^2 = \|\mathbf{Q}\mathbf{A}\mathbf{Q}^T\|_F^2 = \sum_{j=1}^n \lambda_j^2(\mathbf{A})$$

有关代数知识（续）

7. Givens旋转矩阵

$$\mathbf{R}(p, q, \theta) = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & \cos \theta & \cdots & \sin \theta & \\ & & & \vdots & & \vdots & \\ & & & -\sin \theta & \cdots & \cos \theta & \\ & & & & & & 1 & \\ & & & & & & & \ddots & \\ & & & & & & & & 1 \end{pmatrix} \begin{matrix} \\ \\ \\ \leftarrow p\text{行} \\ \\ \leftarrow q\text{行} \\ \\ \\ \end{matrix}$$

$\begin{matrix} \uparrow & \uparrow \\ p\text{列} & q\text{列} \end{matrix}$

称为Givens旋转矩阵。

有关代数知识（续）

Givens旋转矩阵常简记为 \mathbf{R}_{pq} 。
该矩阵是一个正交阵，即

$$\mathbf{R}_{pq}^T = \mathbf{R}_{pq}^{-1}$$

用旋转矩阵作相似变换阵时十分方便。Jacobi方法就是用这种旋转矩阵对实对称阵作一系列的旋转相似变换，从而将实对称阵约化为近似对角阵的。用旋转矩阵作旋转变换的几何意义是，在 n 维空间中，以 p 、 q 轴形成的超平面上，把 p 、 q 轴旋转一个角度 θ 。

一、古典Jacobi方法

考虑实矩阵 $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ ($a_{12} = a_{21} \neq 0$)

令 $\mathbf{R} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix}$

易知，对任何 θ ， \mathbf{R} 为正交矩阵。称 \mathbf{R} 为平面旋转矩阵。

令 $\mathbf{A}^{(1)} = \mathbf{R} \mathbf{A} \mathbf{R}^T = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} = (a_{ij}^{(1)})_{2 \times 2}$

由矩阵乘法可得

$$\begin{cases} a_{11}^{(1)} = a_{11} \cos^2 \theta + 2a_{12} \sin \theta \cos \theta + a_{22} \sin^2 \theta \\ a_{22}^{(1)} = a_{11} \sin^2 \theta - 2a_{12} \sin \theta \cos \theta + a_{22} \cos^2 \theta \\ a_{12}^{(1)} = a_{21}^{(1)} = (a_{22} - a_{11}) \sin \theta \cos \theta + a_{12} (\cos^2 \theta - \sin^2 \theta) \end{cases}$$



古典Jacobi方法（续）

由上式最后一式知，要使 \mathbf{A} 的相似矩阵 $\mathbf{A}^{(1)}$ 成为对角阵，只须适当选取 θ ，使

$$a_{12}^{(1)} = a_{21}^{(1)} = (a_{22} - a_{11}) \sin \theta \cos \theta + a_{12} \cos 2\theta = 0$$

即取
$$\tan 2\theta = \frac{2a_{21}}{a_{11} - a_{22}}$$

其中 $|\theta| \leq \frac{\pi}{4}$ ，当 $a_{11} = a_{22}$ 时，可选 $\theta = \frac{\pi}{4}$ 。

确定 θ 后，旋转矩阵 \mathbf{R} 则随之确定。故 \mathbf{A} 的特征值为

$$\lambda_1 = a_{11}^{(1)}, \lambda_2 = a_{22}^{(1)}$$

古典Jacobi方法（续）

特征向量即是 \mathbf{R}^T 的各列，即对应于 λ_1, λ_2 的特征向量是

$$\mathbf{x}_1 = (\cos \theta, \sin \theta)^T, \quad \mathbf{x}_2 = (-\sin \theta, \cos \theta)^T$$

可以看出，当 \mathbf{A} 为实二阶对称阵时，用适当的正交相似变换一次即可把 \mathbf{A} 化为对角阵。当 \mathbf{A} 为 n 阶实对称阵时，要用到 n 阶旋转矩阵（Givens旋转矩阵），它与单位矩阵的区别仅在于 $(p,p), (p,q), (q,p), (q,q)$ 四个位置的元素不一样。容易验证， $\mathbf{R}(p,q, \theta)$ 具有如下性质：

- （1） $\mathbf{R}(p,q, \theta)$ 为正交阵；
- （2） \mathbf{RA} 只改变 \mathbf{A} 的第 p 行与第 q 行元素， \mathbf{AR}^T 只改变 \mathbf{A} 的第 p 列与第 q 列元素， \mathbf{RAR}^T 只改变 \mathbf{A} 的第 p, q 行与 p, q 列的元素。

古典Jacobi方法（续）

Jacobi方法就是用一系列的旋转相似变换逐渐将 A 化为对角阵的过程：

$$\begin{cases} \mathbf{A}_0 = \mathbf{A} \\ \mathbf{A}_{k+1} = \mathbf{R}_{k+1} \mathbf{A}_k \mathbf{R}_{k+1}^T, k = 0, 1, 2, \dots \end{cases}$$

恰当地选取每个旋转阵 \mathbf{R}_{k+1} ，就可以使 \mathbf{A}_k 趋于对角化。

设 $\mathbf{R}_{k+1} = \mathbf{R}(p, q, \theta)$ 。由 \mathbf{R}_{k+1} 正交可知， \mathbf{A}_{k+1} 与 \mathbf{A}_k 相似，且 \mathbf{A}_k 都为是对称阵。 \mathbf{A}_{k+1} 与 \mathbf{A}_k 的差别仅在于 p 、 q 行与 p 、 q 列的元素。由矩阵乘法可得：

$$\begin{cases} a_{pj}^{(k+1)} = a_{pj}^{(k)} \cos \theta + a_{qj}^{(k)} \sin \theta = a_{jp}^{(k+1)} \\ a_{qj}^{(k+1)} = -a_{pj}^{(k)} \sin \theta + a_{qj}^{(k)} \cos \theta = a_{jq}^{(k+1)}, j \neq p, q \end{cases}$$

古典Jacobi方法（续）

$$\begin{cases} a_{pp}^{(k+1)} = a_{pp}^{(k)} \cos^2 \theta + 2a_{pq}^{(k)} \sin \theta \cos \theta + a_{qq}^{(k)} \sin^2 \theta \\ a_{qq}^{(k+1)} = a_{pp}^{(k)} \sin^2 \theta - 2a_{pq}^{(k)} \sin \theta \cos \theta + a_{qq}^{(k)} \cos^2 \theta \\ a_{pq}^{(k+1)} = (a_{qq}^{(k)} - a_{pp}^{(k)}) \sin \theta \cos \theta + a_{pq}^{(k)} (\cos^2 \theta - \sin^2 \theta) = a_{qp}^{(k+1)} \end{cases}$$

$$a_{ij}^{(k+1)} = a_{ij}^{(k)}, i, j \neq p, q$$

由上面几式容易知道：

$$(a_{pj}^{(k+1)})^2 + (a_{qj}^{(k+1)})^2 = (a_{pj}^{(k)})^2 + (a_{qj}^{(k)})^2, j \neq p, q$$

$$(a_{ij}^{(k+1)})^2 = (a_{ij}^{(k)})^2, i, j \neq p, q$$

再由正交矩阵的性质（代数知识6）有：

$$(a_{pp}^{(k+1)})^2 + (a_{qq}^{(k+1)})^2 + 2(a_{pq}^{(k+1)})^2 = (a_{pp}^{(k)})^2 + (a_{qq}^{(k)})^2 + 2(a_{pq}^{(k)})^2$$

古典Jacobi方法（续）

若 $a_{pq}^{(k)} \neq 0$ ，选取 θ ，使 $a_{pq}^{(k+1)} = 0$ ，只需 θ 满足：

$$\tan(2\theta) = \frac{2a_{pq}^{(k)}}{a_{pp}^{(k)} - a_{qq}^{(k)}}$$

这里假定 $|\theta| \leq \frac{\pi}{4}$ 。若 $a_{pp}^{(k)} = a_{qq}^{(k)}$ ，则取 $\theta = \text{sign}(a_{pq}^{(k)}) \cdot \frac{\pi}{4}$ ，故有

$$\begin{cases} \cos \theta = \frac{\sqrt{2}}{2} \\ \sin \theta = \text{sign}(a_{pq}^{(k)}) \cos \theta \end{cases}$$

当 $a_{pp}^{(k)} \neq a_{qq}^{(k)}$ 时，令

$$\tan(2\theta) = \frac{1}{d} = \frac{2a_{pq}^{(k)}}{a_{pp}^{(k)} - a_{qq}^{(k)}}$$

古典Jacobi方法（续）

则 $\tan^2 \theta + 2d \tan \theta - 1 = 0$, $\tan \theta = -d \pm \sqrt{d^2 + 1}$
为避免相近的数相减, 取

$$t = \tan \theta = \frac{\text{sign}(d)}{|d| + \sqrt{d^2 + 1}}$$

最后得 $\cos \theta = \frac{1}{\sqrt{t^2 + 1}}$, $\sin \theta = t \cos \theta$

此时有 $(a_{pp}^{(k+1)})^2 + (a_{qq}^{(k+1)})^2 = (a_{pp}^{(k)})^2 + (a_{qq}^{(k)})^2 + 2(a_{pq}^{(k)})^2$

引入记号 $D(\mathbf{A}) = \sum_{i=1}^n a_{ii}^2$ 与 $S(\mathbf{A}) = \sum_{\substack{i,j=1 \\ i \neq j}}^n a_{ij}^2$

其中 $D(\mathbf{A})$ 表示对角元的平方和, $S(\mathbf{A})$ 表示非对角元的平方和。

古典Jacobi方法（续）

则

$$\begin{cases} D(\mathbf{A}_{k+1}) = D(\mathbf{A}_k) + 2(a_{pq}^{(k)})^2 \\ S(\mathbf{A}_{k+1}) = S(\mathbf{A}_k) - 2(a_{pq}^{(k)})^2 \end{cases}$$

这说明，只要 $a_{pq}^{(k)} \neq 0$ ，则按上述方法构造的旋转矩阵 $\mathbf{R}(p, q, \theta)$ 对 \mathbf{A}_k 变换后，就会使对角线元素平方和增加，非对角线元素平方和减少。

Remark: 使用旋转矩阵变换后，虽然 $a_{pq}^{(k+1)} = 0$ ，但 $a_{pq}^{(m)}$ ($m > k + 1$) 可能又会变成非零。因此，并不能通过有限次旋转变换就可以将矩阵 \mathbf{A} 化为对角阵。但是，Jacobi方法确实是收敛的，我们可以通过反复变换将 \mathbf{A} 化为近似的对角阵。

下面我们来证明古典Jacobi方法的收敛性。

古典Jacobi方法（续）

定理： 设 \mathbf{A} 为实对称矩阵，则由 $\mathbf{A}_0=\mathbf{A}$ 出发，经古典Jacobi方法计算所产生的矩阵序列 $\{\mathbf{A}_k\}$ 收敛于对角阵 $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ ，且 $\lambda_1, \lambda_2, \dots, \lambda_n$ 就是 \mathbf{A} 的全部特征值。

证明： 由前面的推导过程知， $S(\mathbf{A}_{k+1}) = S(\mathbf{A}_k) - 2(a_{pq}^{(k)})^2$

我们在计算时是选取 $|a_{pq}^{(k)}| = \max_{i < j} |a_{ij}^{(k)}|$ ，故

$$S(\mathbf{A}_k) = \sum_{\substack{i,j=1 \\ i \neq j}}^n (a_{ij}^{(k)})^2 \leq \sum_{i,j=1}^n (a_{pq}^{(k)})^2 = (n^2 - n)(a_{pq}^{(k)})^2 \quad (a_{pq}^{(k)})^2 \geq \frac{1}{n(n-1)} S(\mathbf{A}_k)$$
$$S(\mathbf{A}_{k+1}) = S(\mathbf{A}_k) - 2(a_{pq}^{(k)})^2 \leq S(\mathbf{A}_k) - \frac{2}{n(n-1)} S(\mathbf{A}_k) = (1 - \frac{2}{n(n-1)}) S(\mathbf{A}_k)$$

从而
$$S(\mathbf{A}_{k+1}) \leq (1 - \frac{2}{n(n-1)})^{k+1} S(\mathbf{A}_0)$$

当 $n > 2$ 时，显然有 $\lim_{k \rightarrow \infty} S(\mathbf{A}_k) = 0$ ，即非对角元的平方和趋于零， \mathbf{A}_k 趋于对角阵，Jacobi方法收敛。

证毕

特征向量的计算

由

$$\mathbf{A}_1 = \mathbf{R}_1 \mathbf{A} \mathbf{R}_1^T, \quad \mathbf{A}_2 = \mathbf{R}_2 \mathbf{A}_1 \mathbf{R}_2^T, \dots, \quad \mathbf{A}_k = \mathbf{R}_k \mathbf{A}_{k-1} \mathbf{R}_k^T$$

$$\begin{aligned} \mathbf{A}_k &= \mathbf{R}_k \mathbf{A}_{k-1} \mathbf{R}_k^T = \mathbf{R}_k \mathbf{R}_{k-1} \mathbf{A}_{k-2} \mathbf{R}_{k-1}^T \mathbf{R}_k^T \\ &= \mathbf{R}_k \mathbf{R}_{k-1} \cdots \mathbf{R}_2 \mathbf{R}_1 \mathbf{A} \mathbf{R}_1^T \mathbf{R}_2^T \cdots \mathbf{R}_{k-1}^T \mathbf{R}_k^T \end{aligned}$$

$$\text{记} \quad \mathbf{H}_0^T = \mathbf{I} \quad \mathbf{H}_{k-1}^T = \mathbf{R}_1^T \mathbf{R}_2^T \cdots \mathbf{R}_{k-1}^T$$

$$\text{则有} \quad \begin{cases} \mathbf{H}_k^T = \mathbf{H}_{k-1}^T \mathbf{R}_k^T \\ \mathbf{A}_k = \mathbf{H}_k \mathbf{A}_0 \mathbf{H}_k^T \end{cases}, k = 1, 2, \dots$$

特征向量的计算（续）

若矩阵序列 \mathbf{A}_k 收敛于对角阵 $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ ，
则 \mathbf{H}_k^T 的第 j 列就是 \mathbf{A} 对应于 λ_j 的标准正交特征向量的
近似值。

\mathbf{H}_k^T 的计算可以使用上述的递推关系式的第一式，
其元素间的关系为：

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

Jacobi方法的计算步骤

- 1.选主元，即确定 $p, q(p < q)$ ，使 $|a_{pq}^{(k)}| = \max_{i < j} |a_{ij}^{(k)}|$ ；
- 2.计算 $\cos \theta, \sin \theta$ ；
- 3.按照递推公式计算新的正交矩阵 \mathbf{H}_{k+1}^T 的元素；
- 4.计算 \mathbf{A}_{k+1} 的元素，其中 $a_{pq}^{(k+1)} = 0$ 。

反复执行以上各步，直至 $S(\mathbf{A}_{k+1}) < \varepsilon$ 为止。这里的 ε 为给定的误差界。

二、Jacobi过关法

使用古典Jacobi方法时，每次先要在非对角元中挑选主元，这要花费很多机时。实用中常采用Jacobi过关法，方法如下：

1. 给定控制误差限 ε ；

2. 计算非对角元素的平方和
$$v_0 = 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^n a_{ij}^2 ;$$

3. 设置一个阈值 $v_1 > 0$ ，如 $v_1 = \frac{v_0}{n}$ ；

4. 对A的非对角元 $a_{ij}(i < j)$ 逐个扫描，若某个 $|a_{ij}| > v_1$ ，则立即对A做一次旋转相似变换，之后对所得新矩阵继续扫描，只要有非对角元的绝对值大于 v_1 ，就用旋转相似变换将其变为零。如此多次扫描和变换，直到每个非对角元 $|a_{ij}| \leq v_1$ ；

Jacobi过关法（续）

5. 若 $v_1 \leq \varepsilon$ ，则结束计算，得到特征值与特征向量；否则转向6；

6. 缩小阈值，比如用 $\frac{v_1}{n}$ 代替 v_1 ，重复4~5。

Remark: Jacobi算法的**优点**是数值稳定性比较好，精度高，求得的特征向量正交性好。Jacobi算法的**缺点**是当 A 为稀疏矩阵时，Jacobi变换将破坏其稀疏性质，且只能适用于实对称矩阵。