文章编号:1006-0464(2016)04-0399-04

计算实对称矩阵特征值特征向量的幂法

曾 莉,肖 明

(西南民族大学计算机科学与技术学院,四川 成都 610041)

摘 要:幂法是一种计算实矩阵主特征值的一种迭代方法,在幂法的基础上进行了扩展,提出了一种能计算实对称矩阵所有特征向量和特征值的迭代方法,并对该方法的收敛性进行了证明,最后通过数值实验验证了该方法的有效性。

关键词:对称矩阵;特征向量;特征值;幂法

中图分类号:O241.6 **文献标志码**:A

An algorithm for computing eigenvalues and eigenvectors of real and symmetric matrix

ZENG Li,XIAO Ming

(College of Computer Science and Technology, Southwest University for Nationalities, Chengdu 610014,China)

Abstract:The power method was an iterative algorithm of calculating the main characteristic value of a real matrix. By extending this method,this paper proposed an approach to calculate all the eigenvectors and eigenvalues of a real and symmetric matrix and proved its convergence. Furthermore,the effectiveness of the algorithm is illustrated by numerical simulations.

Key words:Symmetric matrix;Eigenvector;Eigenvalue;Power method

在物理学、量子力学和自动控制等领域中都涉及到求矩阵特征值和特征向量的问题,例如在对图像处理和数据做主成分分析时就需要求解协方差矩阵的特征值和特征向量。文献[1-2]中介绍了很多常用求解特征值问题的方法,如:LR 方法、QR 方法、幂法和反幂法、雅可比法,文献[3-4]提出了对幂法的改进,文献[5-6]讨论了非负不可约矩阵最大特征值 Perron 根的计算方法,文献[7]给出了针对 Pascal 矩阵谱半径和相应特征向量的一个快速算法。本文将讨论求解针对实对称矩阵所有特征值和特征向量问题,对此,文献[8]采用投影幂法处理,而文献[9-10]用神经网络的方法来求解并证明了收敛性。本文从幂法出发,给出了基于幂法扩展的子空间迭代算法,并给出了其收敛性的详细证明。

1 幂法思想

设实矩阵 $A=(a_{ij})\in R^{n\times n}(i,j=1,2,\cdots,n)$,有一个完全的特征向量组,其特征值为 $\lambda_1,\lambda_2,\cdots,\lambda_n$,对应的特征向量为 $x_1,x_2,\cdots,x_n$。已知 A 的主特征值是实根,且满足条件:$|\lambda_1|>|\lambda_2|\geqslant|\lambda_3|\geqslant\cdots\geqslant|\lambda_n|$,幂法主要用于计算绝对值最大的特征值,其基本思想是对任意的非零初始向量 $v_0\in R^n$,v_0 在 $x_i(i=1,2,\cdots,n)$ 下分解为,

$$v_0=\sum_{i=1}^{n}a_ix_i(i=1,2,\cdots,n) \tag{1}$$

由矩阵 A 构造一向量序列

收稿日期:2015-12-26。
基金项目:西南民族大学中央高校基本科研业务费专项资金项目资助(3182014NZYQN33)。
作者简介:曾 莉(1974-),女,副教授,硕士。E-mail:Lzeng@126.com。

$$\begin{cases} v_1 = Av_0, \\ v_2 = Av_1 = A^2 v_0, \\ \quad \vdots \\ v_{k+1} = Av_k = A^{k+1} v_0 \\ \quad \vdots \end{cases}$$

于是 $v_k = Av_{k-1} = A^k v_0 = \sum_{i=1}^{n} a_i \lambda_i^k x_i$，当 $a_1 \neq 0$ 时，

$$v_k = \sum_{i=1}^{n} a_i \lambda_i^k x_i = a_1 \lambda_1^k x_1 + a_2 \lambda_2^k x_2 + \cdots + a_n \lambda_n^k x_n$$

$$= \lambda_1^k [a_1 x_1 + \sum_{i=2}^{n} a_i (\lambda_i / \lambda_1)^k x_i] = \lambda_1^k (a_1 x_1 + \varepsilon_k)$$

当 k 充分大时有 $v_k \approx \lambda_1^k a_1 x_1$，$v_{k+1} \approx v_{k+1} \approx \lambda_1^{k+1} a_1 x_1 \approx \lambda_1 v_k$。所以，$\lambda_1 \approx \frac{v_{k+1}}{v_k}$。又 $v_{k+1} = Av_k \approx \lambda_1 v_k$ 所以，v_k 为的 λ_1 近似特征向量。

2 扩展幂法求实对称矩阵的所有特征值和特征向量

实对称矩阵的不同特征值对应的特征向量一定是正交的，因此，可以通过幂法迭代得到矩阵的主特征值 λ_1 和主特征向量 x_1，再重新给出一个新的与 v_0 线性无关的初始迭代向量 v_1，使 v_1 和 x_1 正交化，则可以由幂法迭代出矩阵的特征值 λ_2 和特征向量 x_2。同样使新给出的初始迭代向量 v_3 和 x_1、x_2 正交化，则可以由幂法得到 λ_3 和 x_3，以此类推，可以计算出矩阵的全部特征值和特征向量。

在幂法中，若初始迭代向量 v_0 中 a_1 等于0，则不会迭代出矩阵的主特征值 λ_1 和主特征向量 x_1，但会收敛到其他的特征值和特征向量，因此我们对幂法进行扩展，选用一组线性无关的向量，将它们正交规范化后作为初始迭代向量，以它们为列组成矩阵 $B_0 = (v_1, \cdots, v_n) \in R^{n\times n}$，对矩阵 B_0 使用幂法迭代，使其依次转换为 $B_1, B_2, \cdots, B_k, \cdots$。显然，经过多次迭代后，$B_k$ 的部分列向量会逐渐收敛到特征向量，再以这些列向量为基础，对 B_k 的其他列向量做正交规范化，然后继续使用幂法迭代，从而使 B_k 中更多的列向量收敛到对应特征向量，如此不断重复，直到各列向量趋于稳定停止。幂法基本算法中是用 B_k 的各个列向量的按模最大分量，作为矩阵 A 的近似特征值，为了加快收敛速度，本文选用Rayleigh商来近似特征值，具体扩展算法如下所示：

扩展算法：

(1) 输入矩阵 $A = (a_{ij} \in R^{n\times n}$，给定误差界 $\varepsilon > 0$，置 $k = 0$ 和最大迭代次数 MAX；

(2) 给出一组线性无关且已正交规范化的初始向量 $v_1(0), v_2(0), \cdots, v_n(0)$；

(3) 分别将 $v_1(0), v_2(0), \cdots, v_n(0)$ 做为列向量，构成矩阵 B；

(4) 重复以下操作，逐步确定各近似特征向量，设 $i = 0$，内部迭代次数 iter；

(ⅰ) 重复做 iter 次操作：

$k = k + 1, u_j(k) = Av_j(k-1), v_j(k) = \frac{u_j(k)}{\max(u_j(k))} (j = 1, 2, \cdots, n)$，即 $B = AB$；

(ⅱ) 计算 $\lambda_j(k) = \frac{(v_j(k), v_j(k-1))}{(v_j(k-1), v_j(k-1))}, (j = 1, 2, \cdots, n)$；

(ⅲ) 对所有 $j = i, i+1, \cdots, n$ 将 $\lambda_j(k)$ 按从大到小排列，同时调整对应 $v_j(k)$ 的在 B 中的位置；

(ⅳ) 对 B 做正交规范化；

(ⅴ) 对所有 $j = i, i+1, \cdots, n$ 验证 $|\lambda_j(k) - \lambda_j(k-1)| < \varepsilon$ 是否成立，若成立，则同时交换 $\lambda_j(k)$、$v_j(k)$ 交换与 $\lambda_i(k)$、$v_i(k)$ 的位置，$i = i + 1$；

(ⅵ) 若 $i = n$ 则转(5)步，否则，若 $k <$ MAX 则继续重复(4)，否则输出溢出，算法结束。

(5) 输出各 $\lambda_j(k)$ 为矩阵 A 的特征值，$v_j(k)$ 为对应特征向量，算法结束。

3 算法的收敛性讨论

设实对称矩阵 A 的全部特征值为 $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n$，对应的特征向量为 $x_1, x_2, \cdots, x_n$，它们组成 R^n 中的一组标准正交基，对应特征子空间为 $V_1, V_2, \cdots, V_n$，用 V 表示 $V_1 \oplus V_2 \oplus \cdots \oplus V_n$，并记 $V_i^{\perp} (i = 1, 2, \cdots, n)$ 表示 R^n 中 V_i 的正交补空间，设 $V_0^{\perp} = V$。

定理 1 若非零初始向量 v_0 属于 $\bigcap_{i=0}^{j-1} V_i^{\perp} - \bigcap_{i=0}^{j} V_i^{\perp}, (j = 1, \cdots, n)$，则 $\lim_{k\to\infty} v_k \in V_j, j = 1, \cdots, n$。

证明 (Ⅰ) 当 $j = 1$ 时，非零初始向量 v_0 属于 $\bigcap V_0^{\perp} - \bigcap V_1^{\perp}$，即(1)式中 $a_1 \neq 0$，所以 $\lim_{k\to\infty} v_k = \lim_{k\to\infty} \sum_{i=1}^{n} a_i \lambda_i^k x_i = \lim_{k\to\infty} \lambda_1^k [a_1 x_1 + \sum_{i=2}^{n} a_i (\lambda_i / \lambda_1)^k x_i] = \lambda_1^k a_1 x_1 \in V_1$ 得证。

(Ⅱ) 当 $j = 2, \cdots, n$ 时，非零初始向量 v_0 属于 $\bigcap_{i=0}^{j-1} V_i^{\perp} - \bigcap_{i=0}^{j} V_i^{\perp}$，由于 $v_0 \in V_i^{\perp} (i = 1, \cdots, j-1)$，所以

(1) 式中 $a_i=0$，$(i=1,2,\cdots,j-1)$，又由于 $v_0\in\bigcap_{i=0}^{j-1}V_i^{\perp}-\bigcap_{i=0}^{j}V_i^{\perp}$，所以 (1) 式中 $a_j\neq 0$。即 $\lim\limits_{k\to\infty}v_k=\lim\limits_{k\to\infty}\sum\limits_{i=j}^{n}a_i\lambda_i^k x_i=\lim\limits_{k\to\infty}\lambda_j^k[a_j x_j+\sum\limits_{i=j+1}^{n}a_i(\lambda_i/\lambda_j)^k x_i]=\lambda_j^k a_j x_j\in V_j$。

综合 Ⅰ 和 Ⅱ 可知结论成立。

定理 2　若要求 $\lim\limits_{k\to\infty}v_k\in V_j(j=1,\cdots,n)$，则非零初始向量 v_0 应分别属于 $\bigcap_{i=0}^{j-1}V_i^{\perp}-\bigcap_{i=0}^{j}V_i^{\perp}$，$(j=1,\cdots,n)$。

证明　反证法，假设 $v_0\notin\bigcap_{i=0}^{j-1}V_i^{\perp}-\bigcap_{i=0}^{j}V_i^{\perp}$，而 $\lim\limits_{k\to\infty}v_k\in V_j(j=1,\cdots,n)$ 下面证明这是矛盾的。

因为 $v_0\notin\bigcap_{i=0}^{j-1}V_i^{\perp}-\bigcap_{i=0}^{j}V_i^{\perp}$，所以 $v_0\in V-(\bigcap_{i=0}^{j-1}V_i^{\perp}-\bigcap_{i=0}^{j}V_i^{\perp})$

即 $v_0\in V-\bigcap_{i=0}^{j-1}V_i^{\perp}+\bigcap_{i=0}^{j}V_i^{\perp}=(V_0^{\perp}-V_1^{\perp})+(V_1^{\perp}-V_1^{\perp}\cap V_2^{\perp})+\cdots+(\bigcap_{i=0}^{j-2}V_i^{\perp}-\bigcap_{i=0}^{j-1}V_i^{\perp})+\bigcap_{i=0}^{j}V_i^{\perp}$

根据抽屉原理，v_0 必属于 $\bigcap_{i=0}^{j}V_i^{\perp}$，$(\bigcap_{i=0}^{t-1}V_i^{\perp}-\bigcap_{i=0}^{t}V_i^{\perp})(t=1,2,\cdots,j-1)$ 之一。

若 $v_0\in\bigcap_{i=0}^{j}V_i^{\perp}$，则 (1) 式中 $a_i=0$，$(i=1,2,\cdots,j)$，因 v_0 为非零向量，故必存在一个最小的 l，有 $a_l\neq 0(j<l\leqslant n)$，因此 $\lim\limits_{k\to\infty}v_k=\lim\limits_{k\to\infty}\sum\limits_{i=l}^{n}a_i\lambda_i^k x_i=\lim\limits_{k\to\infty}\lambda_l^k[a_l x_l+\sum\limits_{i=l+1}^{n}a_i(\lambda_i/\lambda_l)^k x_i]=\lambda_l^k a_l x_l\in V_l$ 与假设矛盾。

若 $v_0\in(\bigcap_{i=0}^{t-1}V_i^{\perp}-\bigcap_{i=0}^{t}V_i^{\perp})(t=1,2,\cdots,j-1$，可得 $\lim\limits_{k\to\infty}(v_k\in V_t(1\leqslant t<j)$ 与假设矛盾。

定理 3　任给 R^n 中的一组线性无关且已正交规范化的 n 个向量 $v_1(0),v_2(0),\cdots,v_n(0)$，则必定存在一个向量，不妨设为 $v_j(0)$，有 $v_j(0)\in(V-V_1^{\perp})$。

证明　反证法，假设 $v_j(0)\emptyset(V-V_1^{\perp})$，则 $v_j(0)\in V_1^{\perp}(1\leqslant j\leqslant n)$。而 $V_1^{\perp}$ 的秩为 $n-1$，这与 $v_1(0),v_2(0),\cdots,v_n(0)$ 是 n 个线性无关向量相矛盾。

由定理 3 和定理 1 可知，本文算法中一定会有向量收敛到子空间 V_1，通过正交规范化，则其它 $n-1$ 个向量均属于 $V_1^{\perp}$，按定理 3 的证明方法类推到，可知必定有一个向量属于 $(\bigcap_{i=0}^{1}V_i^{\perp}-\bigcap_{i=0}^{2}V_i^{\perp})$，再由定理 2 知其最终收敛到子空间 V_2，如此重复，最终可得到全部特征向量和特征值。

4　数值实验

例 1　求矩阵 $A=\begin{bmatrix}2&1&0\\1&3&1\\0&1&4\end{bmatrix}$ 的全部特征值和特征向量。

本例选择矩阵的各列正交规范化后做初始向量，并设定 $\varepsilon=10^{-6}$，iter $=2$。通过 C++ 编程实现算法，程序运行结果如表 1 所示：

表 1　求解例 1 矩阵特征值和特征向量的运行情况

	Matlab 解			本文算法运行情况			
迭代次数				8		12	
锁定个数				1		3	
特征值	4.732 1	3	1.267 9	4.732 051	4.732 050 8	3	1.267 949
特征向量	0.211 3	−0.577 4	0.788 7	0.211 303 9	0.211 324 8	0.577 305 9	0.788 707 7
	0.577 4	−0.577 4	0.577 4	0.577 365 6	0.577 350 3	0.577 382 8	−0.577 317 7
	0.788 7	0.577 4	0.211 3	0.788 669 5	0.788 675 1	−0.577 362 2	0.211 292 5

例 2　求 8 阶实对矩阵

$$A=\begin{pmatrix}1.2&-0.7&0&\cdots&0&0\\-0.7&1.2&-0.7&\cdots&0&0\\0&-0.7&1.2&\cdots&0&0\\\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\0&0&0&\cdots&1.2&-0.7\\0&0&0&\cdots&-0.7&1.2\end{pmatrix}$$

的全部特征值和特征向量。

本例选择矩阵 A 的各列正交规范化后做初始向量，并设定 $\varepsilon=10^{-6}$，$iter=5$。程序运行情况如表 2 所示：

表 2　求解例 2 矩阵特征值和特征向量的运行情况

Matlab 解								
特征值	2.515 6	2.272 5	1.900 0	1.443 1	0.956 9	0.500 0	0.127 5	−0.115 6
特征向量	−0.161 2	−0.303 0	0.408 2	−0.464 2	0.464 2	−0.408 2	−0.303 0	0.161 2
	0.303 0	0.464 2	−0.408 2	0.161 2	0.161 2	−0.408 2	−0.464 2	0.303 0
	−0.408 2	−0.408 2	0.000 0	0.408 2	−0.408 2	−0.000 0	−0.408 2	0.408 2
	0.464 2	0.161 2	0.408 2	−0.303 0	−0.303 0	0.408 2	−0.161 2	0.464 2
	−0.464 2	0.161 2	−0.408 2	−0.303 0	0.303 0	0.408 2	0.161 2	0.464 2
	0.408 2	−0.408 2	−0.000 0	0.408 2	0.408 2	0.000 0	0.408 2	0.408 2
	−0.303 0	0.464 2	0.408 2	0.161 2	−0.161 2	−0.408 2	0.464 2	0.303 0
	0.161 2	−0.303 0	0.408 2	−0.464 2	−0.464 2	−0.408 2	0.303 0	0.161 2
本文算法运行情况								
迭代次数	20	25	35	55	65	70		
确定个数	1	2	4	5	6	8		
近似特征值	2.5132	2.282 1	1.900 0	1.443 1	0.956 9	0.500 0	0.127 5	−0.115 6
对应特征向量	−0.187 5	−0.264 4	−0.423 6	0.464 2	0.408 2	−0.302 7	0.464 2	0.161 8
	0.343 0	0.412 2	0.431 7	−0.161 2	0.408 2	−0.463 7	0.161 2	0.303 8
	−0.442 7	−0.370 0	−0.020 5	−0.408 3	0.000 0	−0.407 5	−0.408 2	0.408 9
	0.476 5	0.142 1	−0.401 2	0.303 0	−0.408 2	−0.160 4	−0.303 0	0.464 5
	−0.448 0	0.178 5	0.419 0	0.303 0	−0.408 3	0.162 0	0.303 0	0.464 0
	0.370 6	−0.442 0	−0.024 8	−0.408 3	−0.000 0	0.408 9	0.408 2	0.407 5
	−0.261 0	0.511 2	−0.380 3	−0.161 2	0.408 3	0.464 8	−0.161 2	0.302 2
	0.134 0	−0.338 3	0.390 0	0.464 2	0.408 3	0.303 3	−0.464 2	0.160 7

5 结束语

本文在幂法上的基础上，给出了一个能计算实对称矩阵所有特征值和特征向量的扩展算法。用严格的理论证明了，从一组线性无关且已正交规范化的初始向量出发，一定能迭代出实对称矩阵的全部特征向量和特征值，并用两个例子验证了算法的有效性。

参考文献：

[1] 徐树方. 矩阵计算的理论与方法[M]. 北京：北京大学出版社，1995：229-269.

[2] 李庆扬，王超能，易大义. 数值分析[M]. 第 4 版. 武汉：华中科技大学出版社，2006：219-245.

[3] SALKUYEH D K，TOUTOUNIAN F. Optimal Iterate of the Power and Inverse Iteration Methods[J]. Applied Numerical Mathematics，2009，59(59)：1537-1548.

[4] GUBEMATISs J E，BOOTH T E. Multiple Extremal Eigenpairs by the Power Method[J]. Journal of Computational Physics，2008，227(19)：8508-8522.

[5] 曾莉，肖明，杨军，等. 非负不可约矩阵 Perron 根的一种迭代算法[J]. 安徽大学学报：自然科学版，2013，37(3)：31-37.

[6] 曾莉，肖明. 非负不可约矩阵 Perron 根的迭代算法研究[J]. 科技通报，2013，29(9)：4-6.

[7] 汪祥，吴武华，廖川荣. 计算 Pascal 矩阵谱半径和相应特征向量的一个快速算法[J]. 南昌大学学报：理科版，2010，34(1)：16-18.

[8] 杨淑娥，陈立萍. 求实矩阵全部特征值的投影幂法[J]. 北京工业大学学报，1994，20(2)：77-82.

[9] 章毅，王平安，周明天. 用神经网络计算矩阵的特征值和特征向量[J]. 计算机学报，2000，23(1)：71-76.

[10] 刘怡光，游志胜，曹丽萍，等. 一种计算矩阵特征值特征向量的神经网络方法[J]. 软件学报，2005，16(06)：1064-1071.