Dr C.-B. Schönlieb

Mathematical Tripos Part II: Michaelmas Term 2015

Numerical Analysis – Lecture 22

5 Eigenvalues and eigenvectors

Remark 5.1 (Introduction to matrix eigenvalue calculations) Let A be a real $n \times n$ matrix. The eigenvalue equation is $A\boldsymbol{w} = \lambda \boldsymbol{w}$, where λ is a scalar, which may be complex if A is not symmetric. There exists a nonzero vector $\boldsymbol{w} \in \mathbb{R}^n$ satisfying this equation if and only if $\det(A - \lambda I) = 0$. The function $p(\lambda) = \det(A - \lambda I)$, $\lambda \in \mathbb{C}$, is a polynomial of degree n, but calculating the eigenvalues by finding the roots of p is a disaster area because of loss of accuracy due to rounding errors.

If the polynomial has some multiple roots and if A is not symmetric, then the number of linearly independent eigenvectors may be fewer than n, but there are always n mutually orthogonal real eigenvectors in the symmetric case. We assume in all cases, however, that the eigenvalue equations $A\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i$, $i = 1..n$, are satisfied by eigenvectors $\boldsymbol{w}_i$ that are linearly independent, which can be achieved by making an arbitrarily small change to A if necessary.

Method 5.2 (The power method) The iterative algorithms that will be studied for the calculation of eigenvalues and eigenvectors are all closely related to the power method, which has the following basic form for generating a single eigenvalue and eigenvector of A.

We pick a nonzero vector $\boldsymbol{x}^{(0)}$ in $\mathbb{R}^n$. Then, for $k = 0, 1, 2, \ldots$, we let $\boldsymbol{x}^{(k+1)}$ be a nonzero multiple of $A\boldsymbol{x}^{(k)}$, typically to satisfy $\|\boldsymbol{x}^{(k+1)}\| = 1$ so that

$$\boldsymbol{x}^{(k+1)} = A\boldsymbol{x}^{(k)} / \|A\boldsymbol{x}^{(k)}\|, \qquad k = 0, 1, 2, \ldots$$

This method is oriented on finding an eigenvector corresponding to the largest eigenvalue as the the following theorem shows.

Theorem 5.3 *Let $A\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i$, where the eigenvalues of A satisfy $|\lambda_1| \leq \cdots \leq |\lambda_{n-1}| < |\lambda_n|$ and the eigenvectors are of the unit length $\|\boldsymbol{w}_i\| = 1$. Assume $\boldsymbol{x}^{(0)} = \sum_{i=1}^n c_i \boldsymbol{w}_i$ with $c_n \neq 0$. Then $\boldsymbol{x}^{(k)} \to \pm \boldsymbol{w}_n$ as $k \to \infty$.*

Proof. Given $x^{(0)}$ as in the assumption, $\boldsymbol{x}^{(k)}$ is a multiple of

$$A^k \boldsymbol{x}^{(0)} = \textstyle\sum_{i=1}^n c_i \lambda_i^k \boldsymbol{w}_i = c_n \lambda_n^k \left(\boldsymbol{w}_n + \sum_{i=1}^{n-1} \frac{c_i}{c_n} \left(\frac{\lambda_i}{\lambda_n} \right)^k \boldsymbol{w}_i \right).$$

Since $\|\boldsymbol{x}^{(k)}\| = \|\boldsymbol{w}_n\| = 1$, we conclude that $\boldsymbol{x}^{(k)} = \pm \boldsymbol{w}_n + \mathcal{O}(\rho^k)$, where the sign is that of $c_n \lambda_n^k$ and the ratio $\rho = \frac{|\lambda_{n-1}|}{|\lambda_n|} < 1$ characterizes the rate of convergence. □

Here are the details of an implementation of the procedure.

0. Pick $\boldsymbol{x}^{(0)} \in \mathbb{R}^n$ satisfying $\|\boldsymbol{x}^{(0)}\| = 1$. Let ε be a small positive tolerance. Set $k = 0$.

1. Calculate $\widetilde{\boldsymbol{x}}^{(k+1)} = A\boldsymbol{x}^{(k)}$ and set $\lambda = \frac{\boldsymbol{x}^{(k)T} A \boldsymbol{x}^{(k)}}{\boldsymbol{x}^{(k)T} \boldsymbol{x}^{(k)}}$.
(This λ is called the *Raleigh quotient* and it minimizes $f(\mu) = \|\widetilde{\boldsymbol{x}}^{(k+1)} - \mu \boldsymbol{x}^{(k)}\|$ over μ.

2. If $f(\lambda) \leq \varepsilon$, accept λ as an eigenvalue and $\boldsymbol{x}^{(k)}$ as the corresponding eigenvector.

3. Otherwise, let $\boldsymbol{x}^{(k+1)} = \widetilde{\boldsymbol{x}}^{(k+1)} / \|\widetilde{\boldsymbol{x}}^{(k+1)}\|$, increase k by one and go back to **1.**

The termination occurs because, by the previous theorem, we have

$$\begin{aligned} \|\widetilde{\boldsymbol{x}}^{(k+1)} - \lambda \boldsymbol{x}^{(k)}\| &= \min_{\mu} \|\widetilde{\boldsymbol{x}}^{(k+1)} - \mu \boldsymbol{x}^{(k)}\| \leq \|\widetilde{\boldsymbol{x}}^{(k+1)} - \lambda_n \boldsymbol{x}^{(k)}\| \\ &= \|A\boldsymbol{x}^{(k)} - \lambda_n \boldsymbol{x}^{(k)}\| = \|A\boldsymbol{w}_n - \lambda_n \boldsymbol{w}_n\| + \mathcal{O}(\rho^k) = \mathcal{O}(\rho^k) \to 0. \end{aligned}$$

Discussion 5.4 (Deficiencies of the power method) The power method may perform adequately if $c_n \neq 0$ and $|\lambda_{n-1}| < |\lambda_n|$, where we are using the notation of Theorem 5.3, but often it is unacceptably slow. The difficulty of $c_n = 0$ is that, theoretically, in this case the method should find an

eigenvector $\boldsymbol{w}_m$ with the largest m such that $c_m \neq 0$, but practically computer rounding errors can introduce a small nonzero component of $\boldsymbol{w}_n$ into the sequence $\boldsymbol{x}^{(k)}$, and then $\boldsymbol{w}_n$ may be found eventually, but one has to wait for the small component to grow. Moreover, $|\lambda_{n-1}| = |\lambda_n|$ is not uncommon when A is real and nonsymmetric, because the spectral radius of A may be due to a complex conjugate pair of eigenvalues. The technique of the next paragraph is designed for that case. Then the use of *shifts* will be studied, because they can be highly useful, particularly in the more efficient methods for eigenvalue calculations that will be considered later.

Method 5.5 (Inverse iteration) This method is highly useful in practice. It is similar to the power method 5.2, except that, instead of $\boldsymbol{x}^{(k+1)}$ being a multiple of $A\boldsymbol{x}^{(k)}$, we make the choice

$$(A - sI)\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)}, \qquad k = 0, 1, \ldots, \tag{5.1}$$

where s is a scalar that may depend on k and $\|x^{(k)}\| = 1$. Therefore the calculation of $\boldsymbol{x}^{(k+1)}$ from $\boldsymbol{x}^{(k)}$ requires the solution of an $n \times n$ system of linear equations whose matrix is $(A - sI)$. Further, if s is a constant and if $A - sI$ is nonsingular, we deduce from (5.1) that $\boldsymbol{x}^{(k)}$ is a multiple of $(A - sI)^{-k}\boldsymbol{x}^{(0)}$.

We again let $\boldsymbol{x}^{(0)} = \sum_{i=1}^{n} c_i \boldsymbol{w}_i$, as in the proof of Theorem 5.3, assuming that $\boldsymbol{w}_i$, $i = 1..n$, are linearly independent eigenvectors of A that satisfy $A\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i$. Therefore we note that the eigenvalue equation implies $(A - sI)\boldsymbol{w}_i = (\lambda_i - s)\boldsymbol{w}_i$, which in turn implies $(A - sI)^{-1}\boldsymbol{w}_i = (\lambda_i - s)^{-1}\boldsymbol{w}_i$. It follows that $\boldsymbol{x}^{(k)}$ is a multiple of

$$(A - sI)^{-k}\boldsymbol{x}^{(0)} = \sum_{i=1}^{n} c_i (A - sI)^{-k}\boldsymbol{w}_i = \sum_{i=1}^{n} c_i (\lambda_i - s)^{-k}\boldsymbol{w}_i.$$

Thus, if the m-th number in the set $\{|\lambda_i - s|\}$ is the smallest and if c_m is nonzero, then $\boldsymbol{x}^{(k)}$ tends to be a multiple of $\boldsymbol{w}_m$ as $k \to \infty$. We see that the speed of convergence can be excellent if s is very close to λ_m. Further, it can be made even faster by adjusting s during the calculation. Typical details are given in the following implementation.

Algorithm 5.6 (Typical implementation of inverse iteration)

0. Set s to an estimate of an eigenvalue of A. Prescribe $\boldsymbol{x}^{(0)} \neq 0$, let $0 < \varepsilon \ll 1$ and set $k = 0$.

1. Calculate (with pivoting if necessary) the LU factorization of $A - sI$.

2. Stop if U is singular because then s is an eigenvalue of A, while its eigenvector is any vector in the null space of U: it can be found easily, U being upper triangular.

3. Calculate $\boldsymbol{x}^{(k+1)}$ by solving $(A - sI)\boldsymbol{x}^{(k+1)} = LU\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)}$ using the LU factorization from **1.**

4. Set η to the number that minimizes $f(\mu) = \|\boldsymbol{x}^{(k)} - \mu\boldsymbol{x}^{(k+1)}\|$.

5. Stop if $f(\eta) \leq \varepsilon\|\boldsymbol{x}^{(k+1)}\|$. Since $f(\eta) = \|A\boldsymbol{x}^{(k+1)} - (s + \eta)\boldsymbol{x}^{(k+1)}\|$, we let $s + \eta$ be the calculated eigenvalue of A and $\boldsymbol{x}^{(k+1)}/\|\boldsymbol{x}^{(k+1)}\|$ be its eigenvector.

6. Otherwise, replace $\boldsymbol{x}^{(k+1)}$ by $\boldsymbol{x}^{(k+1)}/\|\boldsymbol{x}^{(k+1)}\|$, increase k by one, and either return to **3** without changing s or to **1** after replacing s by $s + \eta$.

Remark 5.7 (Further on inverse iteration) Algorithm 5.6 is very efficient if A is an *upper Hessenberg matrix:* every element of A under the first subdiagonal is zero (i.e. $a_{ij} = 0$ if $j < i - 1$). In this case the LU factorization in **1** requires just $\mathcal{O}(n^2)$ or $\mathcal{O}(n)$ operations when A is nonsymmetric or symmetric, respectively. Thus the replacement of s by $s + \eta$ in **6** need not be expensive, so fast convergence can often be achieved easily. There are standard ways of giving A this convenient form which will be considered later. This and our next topic, *deflation* depend on the following basic result.

Matlab demo: Download the Matlab GUI for *Computing eigenvalues and eigenvectors* from `http://www.maths.cam.ac.uk/undergrad/course/na/ii/eigenstuff/eigenstuff.php`. Compare the power method with inverse iteration for different types of matrices, e.g. symmetric/nonsymmetric, upper Hessenberg, etc.