7

Evaluation of Eigenvalues and Eigenvectors

Before we discuss methods for computing eigenvalues, we recall a remark made in Chap. 5. A given n^{th}-degree polynomial $p(c)$ is the characteristic polynomial of some matrix. The companion matrix of equation (3.225) is one such matrix. Thus, given a general polynomial p, we can form a matrix A whose eigenvalues are the roots of the polynomial; and likewise, given a square matrix, we can write a polynomial in its eigenvalues. It is a well-known fact in the theory of equations that there is no general formula for the roots of a polynomial of degree greater than 4. This means that we cannot expect to have a direct method for calculating eigenvalues of any given matrix.

The eigenvalues of some matrices, of course, can be evaluated directly. The eigenvalues of a diagonal matrix, for example, are merely the diagonal elements. In that case, the characteristic polynomial is of the factored form $\prod(a_{ii} - c)$, whose roots are immediately obtainable. For general eigenvalue computations, however, we must use an iterative method.

In statistical applications, the matrices whose eigenvalues are of interest are often symmetric. Symmetric matrices are diagonalizable and have only real eigenvalues. (As usual, we will assume the matrices themselves are real.) The problem of determining the eigenvalues of a symmetric matrix therefore is simpler than the corresponding problem for a general matrix. In many statistical applications, the symmetric matrices of interest are nonnegative definite, and this can allow use of simpler methods for computing eigenvalues and eigenvectors. In addition, nonsymmetric matrices of interest in statistical applications are often irreducible nonnegative matrices, and computations for eigenvalues and eigenvectors for matrices of this type are also often simpler. (We will discuss such matrices in Sect. 8.7.3.)

In this chapter, we describe various methods for computing eigenvalues. A given method may have some desirable property for particular applications, and in some cases, the methods may be used in combination. Some of the methods rely on sequences that converge to a particular eigenvalue or eigenvector. The power method, discussed in Sect. 7.2, is of this type; one

J.E. Gentle, *Matrix Algebra*, Springer Texts in Statistics,
DOI 10.1007/978-3-319-64867-5_7

eigenpair at a time is computed. Other methods are based on sequences of orthogonally similar matrices that converge to a diagonal matrix. An example of such a method is called the *LR method*. This method, which we will not consider in detail, is based on a factorization of A into left and right factors, F_{L} and F_{R}, and the fact that if c is an eigenvalue of $F_{\mathrm{L}}F_{\mathrm{R}}$, then it is also an eigenvalue of $F_{\mathrm{R}}F_{\mathrm{L}}$ (property 8, page 136). If $A = L^{(0)}U^{(0)}$ is an LU decomposition of A with 1s on the diagonal of either $L^{(0)}$ or $U^{(0)}$, iterations of LU decompositions of the similar matrices

$$L^{(k+1)}U^{(k+1)} = U^{(k)}L^{(k)},$$

under some conditions, will converge to a similar diagonal matrix. The sufficient conditions for convergence include nonnegative definiteness.

7.1 General Computational Methods

For whatever approach is taken for finding eigenpairs, there are some general methods that may speed up the process or that may help in achieving higher numerical accuracy. Before describing some of the techniques, we consider a bound on the sensitivity of eigenvalues to perturbations of the matrix.

7.1.1 Numerical Condition of an Eigenvalue Problem

The upper bounds on the largest eigenvalue, given in inequalities (3.235) and (3.236) on page 142, provide a simple indication of the region in the complex plane in which the eigenvalues lie.

The Gershgorin disks (inequalities (3.239) and (3.240), page 145) provide additional information about the regions of the complex plane in which the eigenvalues lie. The Gershgorin disks can be extended to define separate regions that contain eigenvalues, but we will not consider those refinements here. The spectral radius and/or Gershgorin disks can be used to obtain approximate values to use in some iterative approximation methods; see equation (7.13) below, for example.

In any computational problem, it is of interest to know what is the effect on the solution when there are small changes in the problem itself. This leads to the concept of a condition number, as we discussed in Sect. 6.1.1 beginning on page 267. The objective is to quantify or at least determine bounds on the rate of change in the "output" relative to changes in the "input".

In the eigenvalue problem, we begin with a square matrix A. We assume that A is diagonalizable. (All symmetric matrices are diagonalizable, and equation (3.248) on page 149 gives necessary and sufficient conditions which many other matrices encountered in statistical applications also satisfy.) We form $V^{-1}AV = C = \mathrm{diag}((c_1, \ldots, c_n))$, where the c_i are the eigenvalues of A.

The approach, as in Sect. 6.1.1, is to perturb the problem slightly by adding a small amount δA to A. Let $\widetilde{A} = A + \delta A$. (Notice that δA does not necessarily represent a scalar multiple of the matrix.)

If A is well-conditioned for the eigenvalue problem, then if $\|\delta A\|$ is small relative to $\|A\|$, the differences in the eigenvalues of A and of $\widetilde{A}$ are likewise small. Let d be any eigenvalue of $\widetilde{A}$ that is not an eigenvalue of A. (If all eigenvalues of $\widetilde{A}$ are eigenvalues of A, then the perturbation has had no effect, and the question we are addressing is not of interest.) Our interest will be in

$$\min_{c \in \sigma(A)} |c - d|.$$

If d is an eigenvalue of $\widetilde{A}$, then $A + \delta A - dI$ is singular and so $V^{-1}(A + \delta A - dI)V$ is also singular. Simplifying this latter expression, we have that $C - dI + V^{-1}\delta AV$ is singular. Since d is not an eigenvalue of A, however, $C - dI$ must be nonsingular, and so $(C - dI)^{-1}$ exists. Multiplying the two expressions we have that $I + (C - dI)^{-1}V^{-1}\delta AV$ is also singular; hence -1 is an eigenvalue of $(C - dI)^{-1}V^{-1}\delta AV$, and so by property 16 on page 140, we have

$$1 \leq \|(C - dI)^{-1}V^{-1}\delta AV\|,$$

for any consistent norm. (Recall that all matrix norms are consistent in my definition.) Furthermore, again using the consistency property multiple times,

$$\|(C - dI)^{-1}V^{-1}\delta AV\| \leq \|(C - dI)^{-1}\| \, \|V^{-1}\| \, \|\delta A\| \, \|V\|.$$

In equation (6.7) on page 269, we defined "the" condition number for a nonsingular matrix V as $\kappa(V) = \|V\| \, \|V^{-1}\|$. Now, since $C - dI$ is a diagonal matrix, we can rewrite the two inequalities above as

$$\min_{c \in \sigma(A)} |c - d| \leq \kappa(V)\|\delta A\|; \tag{7.1}$$

that is, the eigenvalues of the perturbed matrix are within given bounds from the eigenvalues of the original matrix.

This fact is called the Bauer-Fike theorem, and it has several variations and ramifications. It is closely related to Gershgorin disks. Our interest here is just to provide a perturbation bound that conveniently relates to the condition number of the diagonalizing matrix.

If A is symmetric, it is orthogonally diagonalizable, and the V above is an orthogonal matrix. Hence, if A is a symmetric matrix, $\widetilde{A} = A + \delta A$, and d is an eigenvalue of $\widetilde{A} = A + \delta A$, then

$$\min_{c \in \sigma(A)} |c - d| \leq \|\delta A\|.$$

7.1.2 Eigenvalues from Eigenvectors and Vice Versa

Some methods for eigenanalysis yield the eigenvalues, and other methods yield the eigenvectors. Given one member of an eigenpair, we usually want to find the other member.

If we are given an eigenvector v of the matrix A, there must be some element v_j that is not zero. For any nonzero element of the eigenvector, the eigenvalue corresponding to v is

$$(Av)_j/v_j. \tag{7.2}$$

Likewise, if the eigenvalue c is known, a corresponding eigenvector is any solution to the singular system

$$(A - cI)v = 0. \tag{7.3}$$

(It is relevant to note that the system is singular because many standard software packages will refuse to solve singular systems whether or not they are consistent!)

An eigenvector associated with the eigenvalue c can be found using equation (7.3) if we know the position of any nonzero element in the vector. Suppose, for example, it is known that $v_1 \neq 0$. We can set $v_1 = 1$ and form another system to solve for the remaining elements of v by writing

$$\begin{bmatrix} a_{11} - 1 & a_1^{\mathrm{T}} \\ a_2 & A_{22} - cI_{n-1} \end{bmatrix} \begin{bmatrix} 1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{7.4}$$

where v_2 is an $(n-1)$-vector and a_1^{T} and a_2 are the remaining elements in the first row and first column, respectively, of A. Rearranging this, we get the $(n-1) \times (n-1)$ system

$$(A_{22} - cI_{n-1})v_2 = -a_2. \tag{7.5}$$

The locations of any zero elements in the eigenvector are critical for using this method. To form a system as in equation (7.4), the position of some nonzero element must be known. Another problem in using this method arises when the geometric multiplicity of the eigenvalue is greater than 1. In that case, the system in equation (7.5) is also singular, and the process must be repeated to form an $(n-2)\times(n-2)$ system. If the multiplicity of the eigenvalue is k, the first full rank system encountered while continuing in this way is the one that is $(n-k) \times (n-k)$.

7.1.3 Deflation

Whenever an eigenvalue together with its associated left and right eigenvectors for a real matrix A are available, another matrix can be formed for which all the other nonzero eigenvalues and corresponding eigenvectors are the same

as for A. (Of course the left and right eigenvalues for many matrices are the same.)

Suppose c_i is an eigenvalue of A with associated right and left eigenvectors v_i and w_i, respectively. Now, suppose that c_j is a nonzero eigenvalue of A such that $c_j \neq c_i$. Let v_j and w_j be, respectively, right and left eigenvectors associated with c_j. Now,

$$\langle Av_i, w_j\rangle = \langle c_i v_i, w_j\rangle = c_i\langle v_i, w_j\rangle,$$

but also

$$\langle Av_i, w_j\rangle = \langle v_i, A^{\mathrm{T}}w_j\rangle = \langle v_i, c_j w_j\rangle = c_j\langle v_i, w_j\rangle.$$

But if

$$c_i\langle v_i, w_j\rangle = c_j\langle v_i, w_j\rangle$$

and $c_j \neq c_i$, then $\langle v_i, w_j\rangle = 0$. Consider the matrix

$$B = A - c_i v_i w_i^{\mathrm{H}}. \tag{7.6}$$

We see that

$$\begin{aligned} Bw_j &= Aw_j - c_i v_i w_i^{\mathrm{H}} w_j \\ &= Aw_j \\ &= c_j w_j, \end{aligned}$$

so c_j and w_j are, respectively, an eigenvalue and an eigenvector of B.

The matrix B has some of the flavor of the sum of some terms in a spectral decomposition of A. (Recall that the spectral decomposition is guaranteed to exist only for matrices with certain properties. In Chap. 3, we stated the existence for diagonalizable matrices but derived it only for symmetric matrices.)

The ideas above lead to a useful method for finding eigenpairs of a diagonalizable matrix. (The method also works if we begin with a simple eigenvalue.) We will show the details only for a real symmetric matrix.

7.1.3.1 Deflation of Symmetric Matrices

Let A be an $n \times n$ symmetric matrix. A therefore is diagonalizable, its eigenvalues and eigenvectors are real, and the left and right eigenvalues are the same.

Let (c, v), with $v^{\mathrm{T}}v = 1$, be an eigenpair of A. Now let X be an $n \times n-1$ matrix whose columns form an orthogonal basis for $\mathcal{V}(A - vv^{\mathrm{T}})$. One easy way of doing this is to choose $n-1$ of the n unit vectors of order n such that none are equal to v and then, beginning with v, use Gram-Schmidt transformations to orthogonalize the vectors, using Algorithm 2.1 on page 39. (Assuming v is not a unit vector, we merely choose $e_1, \ldots, e_{n-1}$ together with v as the starting set of linearly independent vectors.) Now let $P = [v|X]$. We have

$$P^{-1} = \begin{bmatrix} v^{\mathrm{T}} \\ X^{\mathrm{T}}(I - vv^{\mathrm{T}}) \end{bmatrix},$$

as we see by direct multiplication, and

$$P^{-1}AP = \begin{bmatrix} c & 0 \\ 0 & B \end{bmatrix}, \tag{7.7}$$

where B is the $(n-1) \times (n-1)$ matrix $X^{\mathrm{T}}AX$.

Clearly, B is symmetric and the eigenvalues of B are the same as the other $n-1$ eigenvalues of A. The important point is that B is $(n-1) \times (n-1)$.

7.1.4 Preconditioning

The convergence of iterative methods applied to a linear system $Ax = b$ can often be speeded up by replacing the system by an equivalent system $M^{-1}Ax = M^{-1}b$. The iterations then depend on the properties, such as the relative magnitudes of the eigenvalues, of $M^{-1}A$ rather than A. The replacement of the system $Ax = b$ by $M^{-1}Ax = M^{-1}b$ is called *preconditioning*. (It is also sometimes called *left preconditioning*, and the use of the system $AM^{-1}y = b$ with $y = Mx$ is called *right preconditioning*. Either or both kinds of preconditioning may be used in a given iterative algorithm.) The matrix M is called a *preconditioner*.

Determining an effective preconditioner matrix M^{-1} for eigenvalue computations is not straightforward. In general, the objective would be to determine $M^{-1}A$ so that it is "close" to I, because then the eigenvalues might be easier to obtain by whatever method we may use. The salient properties of I are that it is normal (see Sect. 8.2.3 beginning on page 345) and its eigenvalues are clustered.

There are various kinds of preconditioning. We have considered preconditioning in the context of an iterative algorithm for solving linear systems on page 284. Some preconditioning methods work better as an adjunct to one algorithm, and others work better in conjunction with some other algorithm. Obviously, the efficacy depends on the nature of the data input to the problem. In the case of a sparse matrix A, for example an incomplete factorization $A \approx \widetilde{L}\widetilde{U}$ where both $\widetilde{L}$ and $\widetilde{U}$ are sparse, $M = \widetilde{L}\widetilde{U}$ may be a good preconditioner. We will not consider any of the details here. Benzi (2002) provides a good survey of techniques, but it is difficult to identify general methods that work well.

7.1.5 Shifting

If c is an eigenvalue of A, then $c - d$ is an eigenvalue of $A - dI$, and the associated eigenvectors are the same. (This is property 7 on page 136.) Hence, instead of seeking an eigenvalue of A, we might compute (or approximate) an

eigenvalue of $A - dI$. (We recall also, from equation (6.11) on page 272, that, for appropriate signs of d and the eigenvalues, the condition number of $A - dI$ is better than the condition number of A.)

Use of $A - dI$ amounts to a "shift" in the eigenvalue. This can often improve the convergence rate in an algorithm to compute an eigenvalue. (Remember that all general algorithms to compute eigenvalues are iterative.)

The best value of d in the shift depends on both the algorithm and the characteristics of the matrix. Various shifts have been suggested. One common value of the shift is based on the Rayleigh quotient shift; another common value is called the "Wilkinson shift", after James Wilkinson. We will not discuss any of the particular shift values here.

7.2 Power Method

The power method is a straightforward method that can be used for a real diagonalizable matrix with a simple dominant eigenvalue. A symmetric matrix is diagonalizable, of course, but it may not have a simple dominant eigenvalue.

The power method finds the dominant eigenvalue. In some applications, only the dominant eigenvalue is of interest. If other eigenvalues are needed, however, we can find them one at a time by deflation.

Let A be a real $n \times n$ diagonalizable matrix with a simple dominant eigenvalue. Index the eigenvalues c_i so that $|c_1| > |c_2| \geq \cdots |c_n|$, with corresponding normalized eigenvectors v_i. Note that the requirement for the dominant eigenvalue that $c_1 > c_2$ implies that c_1 and the dominant eigenvector v_1 are unique and that c_1 is real (because otherwise $\bar{c}_1$ would also be an eigenvalue, and that would violate the requirement).

Now let x be an n-vector that is not orthogonal to v_1. Because A is assumed to be diagonalizable, the eigenvectors are linearly independent and so x can be represented as a linear combination of the eigenvectors,

$$x = b_1 v_1 + \cdots + b_n v_n. \tag{7.8}$$

Because x is not orthogonal to v_1, $b_1 \neq 0$. The power method is based on a sequence

$$x,\ Ax,\ A^2x, \ldots.$$

(This sequence is a finite Krylov space generating set; see equation (6.26).) From the relationships above and the definition of eigenvalues and eigenvectors, we have

$$\begin{aligned} Ax &= b_1 Av_1 + \cdots + b_n Av_n \\ &= b_1 c_1 v_1 + \cdots + b_n c_n v_n \\ A^2x &= b_1 c_1^2 v_1 + \cdots + b_n c_n^2 v_n \\ \cdots &= \cdots \end{aligned}$$

$$\begin{aligned} A^j x &= b_1 c_1^j v_1 + \cdots + b_n c_n^j v_n \\ &= c_1^j \left(b_1 v_1 + \cdots + b_n \left(\frac{c_n}{c_1} \right)^j v_n \right). \end{aligned} \tag{7.9}$$

To simplify the notation, let

$$u^{(j)} = A^j x / c_1^j \tag{7.10}$$

(or, equivalently, $u^{(j)} = Au^{(j-1)}/c_1$). From equations (7.9) and the fact that $|c_1| > |c_i|$ for $i > 1$, we see that $u^{(j)} \to b_1 v_1$, which is the nonnormalized dominant eigenvector.

We have the bound

$$\begin{aligned} \left\| u^{(j)} - b_1 v_1 \right\| &= \left\| b_2 \left(\frac{c_2}{c_1} \right)^j v_2 + \cdots \right. \\ &\qquad \left. \cdots + b_n \left(\frac{c_n}{c_1} \right)^j v_n \right\| \\ &\leq |b_2| \left| \frac{c_2}{c_1} \right|^j \|v_2\| + \cdots \\ &\qquad \cdots + |b_n| \left| \frac{c_n}{c_1} \right|^j \|v_n\| \\ &\leq (|b_2| + \cdots + |b_n|) \left| \frac{c_2}{c_1} \right|^j. \end{aligned} \tag{7.11}$$

The last expression results from the fact that $|c_2| \geq |c_i|$ for $i > 2$ and that the v_i are unit vectors.

From equation (7.11), we see that the norm of the difference of $u^{(j)}$ and $b_1 v_1$ decreases by a factor of approximately $|c_2/c_1|$ with each iteration; hence, this ratio is an important indicator of the rate of convergence of $u^{(j)}$ to the dominant eigenvector.

If $|c_1| > |c_2| > |c_3|$, $b_2 \neq 0$, and $b_1 \neq 0$, the power method converges linearly (see page 511); that is,

$$0 < \lim_{j \to \infty} \frac{\|u^{(j+1)} - b_1 v_1\|}{\|u^{(j)} - b_1 v_1\|} < 1 \tag{7.12}$$

(see Exercise 7.1c, page 324). Shifting the matrix to form $A - dI$ results in a matrix with eigenvalues with different relative sizes, and may be useful in speeding up the convergence.

If an approximate value of the eigenvector v_1 is available and x is taken to be that approximate value, the convergence will be faster. If an approximate value of the dominant eigenvalue, $\widehat{c}_1$, is available, starting with any $y^{(0)}$, a few iterations on

$$(A - \widehat{c}_1 I)y^{(k)} = y^{(k-1)} \tag{7.13}$$

may yield a better starting value for x. Once the eigenvector associated with the dominant eigenvalue is determined, the eigenvalue c_1 can easily be determined, as described above.

7.2.1 Inverse Power Method

If A is nonsingular, we can also use the power method on A^{-1} to determine the smallest eigenvalue of A. This is called the "inverse power method".

The rate of convergence may be very different from that of the power method applied to A. Shifting is also generally important in the inverse power method. Of course this method only determines the eigenvalue with the smallest absolute value. If other eigenvalues are needed, we can find them one at a time by deflation.

7.3 Jacobi Method

The Jacobi method for determining the eigenvalues of a simple symmetric matrix A uses a sequence of orthogonal similarity transformations that eventually results in the transformation

$$A = PCP^{-1}$$

(see equation (3.247) on page 149) or

$$C = P^{-1}AP,$$

where C is diagonal. Recall that similar matrices have the same eigenvalues.

The matrices for the similarity transforms are the Givens rotation or Jacobi rotation matrices discussed on page 238. The general form of one of these orthogonal matrices, $G_{pq}(\theta)$, given in equation (5.12) on page 239, is the identity matrix with $\cos\theta$ in the $(p,p)^{\text{th}}$ and $(q,q)^{\text{th}}$ positions, $\sin\theta$ in the $(p,q)^{\text{th}}$ position, and $-\sin\theta$ in the $(q,p)^{\text{th}}$ position:

$$G_{pq}(\theta) = \begin{array}{c} \\ p \\ \\ q \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccc} \quad & p & \quad & q & \quad \end{array} \\ \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & \cos\theta & 0 & \sin\theta & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix} \end{array}.$$

The Jacobi iteration is

$$A^{(k)} = G^{\mathrm{T}}_{p_k q_k}(\theta_k) A^{(k-1)} G_{p_k q_k}(\theta_k),$$

where p_k, q_k, and θ_k are chosen so that the $A^{(k)}$ is "more diagonal" than $A^{(k-1)}$. Specifically, the iterations will be chosen so as to reduce the sum of the squares of the off-diagonal elements, which for any square matrix A is

$$\|A\|_{\mathrm{F}}^2 - \sum_i a_{ii}^2.$$

The orthogonal similarity transformations preserve the Frobenius norm

$$\left\|A^{(k)}\right\|_{\mathrm{F}} = \left\|A^{(k-1)}\right\|_{\mathrm{F}}.$$

Because the rotation matrices change only the elements in the $(p,p)^{\text{th}}$, $(q,q)^{\text{th}}$, and $(p,q)^{\text{th}}$ positions (and also the $(q,p)^{\text{th}}$ position since both matrices are symmetric), we have

$$\left(a_{pp}^{(k)}\right)^2 + \left(a_{qq}^{(k)}\right)^2 + 2\left(a_{pq}^{(k)}\right)^2 \;=\; \left(a_{pp}^{(k-1)}\right)^2 + \left(a_{qq}^{(k-1)}\right)^2 + 2\left(a_{pq}^{(k-1)}\right)^2.$$

The off-diagonal sum of squares at the k^{th} stage in terms of that at the $(k-1)^{\text{th}}$ stage is

$$\begin{aligned}\left\|A^{(k)}\right\|_{\mathrm{F}}^2 - \sum_i \left(a_{ii}^{(k)}\right)^2 &= \left\|A^{(k)}\right\|_{\mathrm{F}}^2 - \sum_{i\neq p,q} \left(a_{ii}^{(k)}\right)^2 - \left(\left(a_{pp}^{(k)}\right)^2 + \left(a_{qq}^{(k)}\right)^2\right) \\ &= \left\|A^{(k-1)}\right\|_{\mathrm{F}}^2 - \sum_i \left(a_{ii}^{(k-1)}\right)^2 - 2\left(a_{pq}^{(k-1)}\right)^2 + 2\left(a_{pq}^{(k)}\right)^2.\end{aligned} \tag{7.14}$$

Hence, for a given index pair, (p,q), at the k^{th} iteration, the sum of the squares of the off-diagonal elements is minimized by choosing the rotation matrix so that

$$a_{pq}^{(k)} = 0. \tag{7.15}$$

As we saw on page 239, it is easy to determine the angle θ so as to introduce a zero in a single Givens rotation. Here, we are using the rotations in a similarity transformation, so it is a little more complicated.

The requirement that $a_{pq}^{(k)} = 0$ implies

$$a_{pq}^{(k-1)}\left(\cos^2\theta - \sin^2\theta\right) + \left(a_{pp}^{(k-1)} - a_{qq}^{(k-1)}\right)\cos\theta\sin\theta = 0. \tag{7.16}$$

Using the trigonometric identities

$$\begin{aligned}\cos(2\theta) &= \cos^2\theta - \sin^2\theta \\ \sin(2\theta) &= 2\cos\theta\sin\theta,\end{aligned}$$

in equation (7.16), we have

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

which yields a unique angle in $[-\pi/4,\ \pi/4]$. Of course, the quantities we need are $\cos\theta$ and $\sin\theta$, not the angle itself. First, using the identity

$$\tan\theta = \frac{\tan(2\theta)}{1 + \sqrt{1 + \tan^2(2\theta)}},$$

we get $\tan\theta$ from $\tan(2\theta)$; and then from $\tan\theta$ we can compute the quantities required for the rotation matrix $G_{pq}(\theta)$:

$$\begin{aligned} \cos\theta &= \frac{1}{\sqrt{1 + \tan^2\theta}}, \\ \sin\theta &= \cos\theta \tan\theta. \end{aligned}$$

Convergence occurs when the off-diagonal elements are sufficiently small. The quantity (7.14) using the Frobenius norm is the usual value to compare with a convergence criterion, ϵ.

From equation (7.15), we see that the best index pair, (p, q), is such that

$$\left|a_{pq}^{(k-1)}\right| = \max_{i<j} \left|a_{ij}^{(k-1)}\right|.$$

If this choice is made, the Jacobi method can be shown to converge (see Watkins 2002). The method with this choice is called the *classical Jacobi* method.

For an $n \times n$ matrix, the number of operations to identify the maximum off-diagonal is $O(n^2)$. The computations for the similarity transform itself are only $O(n)$ because of the sparsity of the rotators. Of course, the computations for the similarity transformations are more involved than those to identify the maximum off-diagonal, so, for small n, the classical Jacobi method should be used. If n is large, however, it may be better not to spend time looking for the maximum off-diagonal. Various *cyclic Jacobi* methods have been proposed in which the pairs (p, q) are chosen systematically without regard to the magnitude of the off-diagonal being zeroed. Depending on the nature of the cyclic Jacobi method, it may or may not be guaranteed to converge. For certain schemes, quadratic convergence has been proven; for at least one other scheme, an example showing failure of convergence has been given. See Watkins (2002) for a discussion of the convergence issues.

The Jacobi method is one of the oldest algorithms for computing eigenvalues, and has recently become important again because it lends itself to easy implementation on parallel processors (see Zhou and Brent 2003).

Notice that at the k^{th} iteration, only two rows and two columns of $A^{(k)}$ are modified. This is what allows the Jacobi method to be performed in parallel.

We can form $\lfloor n/2 \rfloor$ pairs and do $\lfloor n/2 \rfloor$ rotations simultaneously. Thus, each parallel iteration consists of a choice of a set of index pairs and then a batch of rotations. Although, as we have indicated, the convergence may depend on which rows are chosen for the rotations, if we are to achieve much efficiency by performing the operations in parallel, we cannot spend much time in deciding how to form the pairs for the rotations. Various schemes have been suggested for forming the pairs for a parallel iteration. A simple scheme, called "mobile Jacobi" (see Watkins 2002), is:

1. Perform $\lfloor n/2 \rfloor$ rotations using the pairs
$$(1,2),\ (3,4),\ (5,6),\ldots.$$
2. Interchange all rows and columns that were rotated.
3. Perform $\lfloor (n-1)/2 \rfloor$ rotations using the pairs
$$(2,3),\ (4,5),\ (6,7),\ldots.$$
4. Interchange all rows and columns that were rotated.
5. If convergence has not been achieved, go to 1.

The notation above that specifies the pairs refers to the rows and columns at the current state; that is, after the interchanges up to that point. The interchange operation is a similarity transformation using an elementary permutation matrix (see page 81), and hence the eigenvalues are left unchanged by this operation. The method described above is a good one, but there are other ways of forming pairs. Some of the issues to consider are discussed by Luk and Park (1989), who analyzed and compared some proposed schemes.

7.4 QR Method

The most common algorithm for extracting eigenvalues is the QR method. While the power method and the Jacobi method require diagonalizable matrices, which restricts their practical use to symmetric matrices, the QR method can be used for nonsymmetric matrices. It is simpler for symmetric matrices, of course, because the eigenvalues are real. Also, for symmetric matrices the computer storage is less, the computations are fewer, and some transformations are particularly simple. In the following description, we will assume that the matrix is symmetric.

The basic idea behind the use of the QR method is that for a symmetric matrix A, the simple iterations beginning with $A^{(0)} = A$, for $k = 1, 2, \ldots$,

$$\begin{aligned} Q^{(k)}R^{(k)} &= A^{(k-1)} \\ A^{(k)} &= R^{(k)}Q^{(k)} \end{aligned}$$

lead to an orthogonal triangularization of A.

These iterations by themselves would be slow and would only work for certain matrices, so the QR method requires that the matrix first be transformed into upper Hessenberg form (see page 59). A matrix can be reduced to Hessenberg form in a finite number of similarity transformations using either Householder reflections or Givens rotations.

The Hessenberg form for a symmetric matrix is tridiagonal. The Hessenberg form allows a large savings in the subsequent computations, even for nonsymmetric matrices.

Even in the Hessenberg form, the matrices $A^{(k)}$ are shifted by $c^{(k)}I$, where $c^{(k)}I$ is an approximation of an eigenvalue, which can be obtained in various ways (see Trefethen and Bau 1997; pages 219 and following).

After the matrix has been transformed into a similar Hessenberg matrix, a sequence of similar Hessenberg matrices that converge to triangular matrix is formed. The QR method for determining the eigenvalues is iterative and produces a sequence of Hessenberg matrices that converge to a triangular matrix. An upper Hessenberg matrix is formed and its eigenvalues are extracted by a process called "chasing", which consists of steps that alternate between creating nonzero entries in positions $(i+2, i)$, $(i+3, i)$, and $(i+3, i+1)$ and restoring these entries to zero, as the nonzero entries are moved farther down the matrix. For example,

$$\begin{bmatrix} \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{Y} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{Y} & \mathtt{Y} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{0} & \mathtt{0} & \mathtt{0} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{0} & \mathtt{0} & \mathtt{0} & \mathtt{0} & \mathtt{X} & \mathtt{X} \end{bmatrix} \rightarrow \begin{bmatrix} \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{0} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{0} & \mathtt{Y} & \mathtt{X} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{0} & \mathtt{Y} & \mathtt{Y} & \mathtt{X} & \mathtt{X} & \mathtt{X} \\ \mathtt{0} & \mathtt{0} & \mathtt{0} & \mathtt{0} & \mathtt{0} & \mathtt{X} & \mathtt{X} \end{bmatrix}.$$

In the j^{th} step of the QR method, a bulge is created and is chased down the matrix by similarity transformations, usually Givens transformations,

$$G_k^{-1} A^{(j-1,k)} G_k.$$

The transformations are based on the eigenvalues of 2×2 matrices in the lower right-hand part of the matrix.

There are some variations on the way the chasing occurs. Haag and Watkins (1993) describe an efficient modified QR algorithm that uses both Givens transformations and Gaussian elimination transformations, with or without pivoting. For the $n \times n$ Hessenberg matrix $A^{(0,0)}$, the first step of the Haag-Watkins procedure begins with a 3×3 Householder reflection matrix, $\widetilde{G}_0$, whose first column is

$$(A^{(0,0)} - \sigma_1 I)(A^{(0,0)} - \sigma_2 I)e_1,$$

where σ_1 and σ_2 are the eigenvalues of the 2×2 matrix

$$\begin{bmatrix} a_{n-1,n-1} & a_{n-1,n} \\ a_{n-1,n} & a_{n,n} \end{bmatrix},$$

and e_1 is the first unit vector of length n. The $n \times n$ matrix G_0 is $\text{diag}(\widetilde{G}_0, I)$. The initial transformation $G_0^{-1}A^{(0,0)}G_0$ creates a bulge with nonzero elements $a_{31}^{(0,1)}$, $a_{41}^{(0,1)}$, and $a_{42}^{(0,1)}$.

After the initial transformation, the Haag-Watkins procedure makes $n-3$ transformations

$$A^{(0,k+1)} = G_k^{-1}A^{(0,k)}G_k,$$

for $k = 1, 2, \ldots, n-3$, that chase the bulge diagonally down the matrix, so that $A^{(0,k+1)}$ differs from Hessenberg form only by the nonzero elements $a_{k+3,k+1}^{(0,k+1)}$, $a_{k+4,k+1}^{(0,k+1)}$, and $a_{k+4,k+2}^{(0,k+1)}$. To accomplish this, the matrix G_k differs from the identity only in rows and columns $k+1$, $k+2$, and $k+3$. The transformation

$$G_k^{-1}A^{(0,k)}$$

annihilates the entries $a_{k+2,k}^{(0,k)}$ and $a_{k+3,k}^{(0,k)}$, and the transformation

$$(G_k^{-1}A^{(0,k)})G_k$$

produces $A^{(0,k+1)}$ with two new nonzero elements, $a_{k+4,k+1}^{(0,k+1)}$ and $a_{k+4,k+2}^{(0,k+1)}$. The final transformation in the first step, for $k = n-2$, annihilates $a_{n,n-2}^{(0,k)}$. The transformation matrix G_{n-2} differs from the identity only in rows and columns $n-1$ and n. These steps are iterated until the matrix becomes triangular. As the subdiagonal elements converge to zero, the shifts for use in the first transformation of a step (corresponding to σ_1 and σ_2) are determined by 2×2 submatrices higher on the diagonal. Special consideration must be given to situations in which these submatrices contain zero elements. For this, the reader is referred to Watkins (2002) or Golub and Van Loan (1996).

This description has just indicated the general flavor of the QR method. There are different variations on the overall procedure and then many computational details that must be observed. In the Haag-Watkins procedure, for example, the G_ks are not unique, and their form can affect the efficiency and the stability of the algorithm. Haag and Watkins (1993) describe criteria for the selection of the G_ks. They also discuss some of the details of programming the algorithm. A very careful description of the basic algorithm and various modifications is provided in Trefethen and Bau (1997), pages 196 through 224.

7.5 Krylov Methods

In the power method, we encountered the sequence

$$x,\ Ax,\ A^2x, \ldots .$$

This sequence is a finite Krylov space generating set. As we mentioned on page 284, several methods for computing eigenvalues are often based on a Krylov space,

$$\mathcal{K}_k = \mathcal{V}(\{v, Av, A^2v, \ldots, A^{k-1}v\}).$$

(Aleksei Krylov used these vectors to construct the characteristic polynomial.)

The two most important Krylov methods are the Lanczos tridiagonalization algorithm and the Arnoldi orthogonalization algorithm. We will not discuss these methods here but rather refer the interested reader to Golub and Van Loan (1996).

7.6 Generalized Eigenvalues

In Sect. 3.8.12, we defined the generalized eigenvalues and eigenvectors by replacing the identity in the definition of ordinary eigenvalues and eigenvectors by a general (square) matrix B:

$$|A - cB| = 0. \tag{7.17}$$

If there exists a finite c such that this determinant is zero, then there is some nonzero, finite vector v such that

$$Av = cBv. \tag{7.18}$$

As we have seen in the case of ordinary eigenvalues, symmetry of the matrix, because of diagonalizability, allows for simpler methods to evaluate the eigenvalues. In the case of generalized eigenvalues, symmetry together with positive definiteness allows us to reformulate the problem to be much simpler. If A and B are symmetric and B is positive definite, we refer to the pair (A, B) as *symmetric*.

If A and B are a symmetric pair, B has a Cholesky decomposition, $B = T^{\mathrm{T}}T$, where T is an upper triangular matrix with positive diagonal elements. We can therefore rewrite equation (7.18) as

$$T^{-\mathrm{T}}AT^{-1}u = cu, \tag{7.19}$$

where $u = Tv$. Note that because A is symmetric, $T^{-\mathrm{T}}AT^{-1}$ is symmetric, and since c is an eigenvalue of this matrix, it is real. Its associated eigenvector (with respect to $T^{-\mathrm{T}}AT^{-1}$) is likewise real, and therefore so is the generalized eigenvector v. Because $T^{-\mathrm{T}}AT^{-1}$ is symmetric, the ordinary eigenvectors can

be chosen to be orthogonal. (Recall from page 153 that eigenvectors corresponding to distinct eigenvalues *are* orthogonal, and those corresponding to a multiple eigenvalue can be chosen to be orthogonal.) This implies that the generalized eigenvectors of the symmetric pair (A, B) can be chosen to be B-conjugate.

Because of the equivalence of a generalized eigenproblem for a symmetric pair to an ordinary eigenproblem for a symmetric matrix, any of the methods discussed in this chapter can be used to evaluate the generalized eigenpairs of a symmetric pair. The matrices in statistical applications for which the generalized eigenvalues are required are often symmetric pairs. For example, Roy's maximum root statistic, which is used in multivariate analysis, is a generalized eigenvalue of two Wishart matrices.

The generalized eigenvalues of a pair that is not symmetric are more difficult to evaluate. The approach of forming upper Hessenberg matrices, as in the QR method, is also used for generalized eigenvalues. We will not discuss this method here but instead refer the reader to Watkins (2002) for a description of the method, which is called the QZ algorithm.

7.7 Singular Value Decomposition

The standard algorithm for computing the singular value decomposition

$$A = UDV^{\mathrm{T}}$$

is due to Golub and Reinsch (1970) and is built on ideas of Golub and Kahan (1965). The first step in the Golub-Reinsch algorithm for the singular value decomposition of the $n \times m$ matrix A is to reduce A to upper bidiagonal form:

$$A^{(0)} = \begin{bmatrix} \mathtt{X} & \mathtt{X} & 0 & \cdots & 0 & 0 \\ 0 & \mathtt{X} & \mathtt{X} & \cdots & 0 & 0 \\ 0 & 0 & \mathtt{X} & \cdots & 0 & 0 \\ & & & \ddots & \ddots & \\ 0 & 0 & 0 & \cdots & \mathtt{X} & \mathtt{X} \\ 0 & 0 & 0 & \cdots & 0 & \mathtt{X} \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}.$$

We assume $n \geq m$. (If this is not the case, we merely use A^{T}.) This algorithm is basically a factored form of the QR algorithm for the eigenvalues of $A^{(0)\mathrm{T}}A^{(0)}$, which would be symmetric and tridiagonal.

The Golub-Reinsch method produces a sequence of upper bidiagonal matrices, $A^{(0)}, A^{(1)}, A^{(2)}, \ldots$, which converges to the diagonal matrix D. (Each of these has a zero submatrix below the square submatrix.) Similar to the QR method for eigenvalues, the transformation from $A^{(j)}$ to $A^{(j+1)}$ is effected by a sequence of orthogonal transformations,

$$\begin{aligned} A^{(j+1)} &= R_{m-2}^{\mathrm{T}} R_{m-3}^{\mathrm{T}} \cdots R_0^{\mathrm{T}} A^{(j)} T_0 T_1 \cdots T_{m-2} \\ &= R^{\mathrm{T}} A^{(j)} T, \end{aligned}$$

which first introduces a nonzero entry below the diagonal (T_0 does this) and then chases it down the diagonal. After T_0 introduces a nonzero entry in the $(2,1)$ position, R_0^{T} annihilates it and produces a nonzero entry in the $(1,3)$ position; T_1 annihilates the $(1,3)$ entry and produces a nonzero entry in the $(3,2)$ position, which R_1^{T} annihilates, and so on. Each of the R_ks and T_ks are Givens transformations, and, except for T_0, it should be clear how to form them.

If none of the elements along the main diagonal or the diagonal above the main diagonal is zero, then T_0 is chosen as the Givens transformation such that T_0^{T} will annihilate the second element in the vector

$$(a_{11}^2 - \sigma_1,\ a_{11}a_{12},\ 0,\ \cdots,\ 0),$$

where σ_1 is the eigenvalue of the lower right-hand 2×2 submatrix of $A^{(0)\mathrm{T}} A^{(0)}$ that is closest in value to the (m,m) element of $A^{(0)\mathrm{T}} A^{(0)}$. This is easy to compute (see Exercise 7.6).

If an element along the main diagonal or the diagonal above the main diagonal is zero, we must proceed slightly differently. (Remember that for purposes of computations "zero" generally means "near zero"; that is, to within some set tolerance.)

If an element above the main diagonal is zero, the bidiagonal matrix is separated at that value into a block diagonal matrix, and each block (which is bidiagonal) is treated separately.

If an element on the main diagonal, say a_{kk}, is zero, then a singular value is zero. In this case, we apply a set of Givens transformations from the left. We first use G_1, which differs from the identity only in rows and columns k and $k+1$, to annihilate the $(k, k+1)$ entry and introduce a nonzero in the $(k, k+2)$ position. We then use G_2, which differs from the identity only in rows and columns k and $k+2$, to annihilate the $(k, k+2)$ entry and introduce a nonzero in the $(k, k+3)$ position. Continuing this process, we form a matrix of the form

$$\begin{bmatrix} \mathtt{X} & \mathtt{X} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathtt{X} & \mathtt{X} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathtt{X} & \mathtt{Y} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathtt{X} & \mathtt{X} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathtt{X} & \mathtt{X} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \mathtt{X} & \mathtt{X} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathtt{X} \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The Y in this matrix (in position $(k-1, k)$) is then chased up the upper block consisting of the first k rows and columns of the original matrix by using Givens transformations applied from the right. This then yields two block bidiagonal matrices (and a 1×1 0 matrix). We operate on the individual blocks as before.

After the steps have converged to yield a diagonal matrix, $\widetilde{D}$, all of the Givens matrices applied from the left are accumulated into a single matrix and all from the right are accumulated into a single matrix to yield a decomposition

$$A = \widetilde{U}\widetilde{D}\widetilde{V}^{\mathrm{T}}.$$

There is one last thing to do. The elements of $\widetilde{D}$ may not be nonnegative. This is easily remedied by postmultiplying by a diagonal matrix G that is the same as the identity except for having a -1 in any position corresponding to a negative value in $\widetilde{D}$. In addition, we generally form the singular value decomposition is such a way that the elements in D are nonincreasing. The entries in $\widetilde{D}$ can be rearranged by a permutation matrix $E_{(\pi)}$ so they are in nonincreasing order. So we have

$$D = E_{(\pi)}^{\mathrm{T}}\widetilde{D}GE_{(\pi)},$$

and the final decomposition is

$$\begin{aligned} A &= \widetilde{U}E_{(\pi)}GDE_{(\pi)}^{\mathrm{T}}\widetilde{V}^{\mathrm{T}} \\ &= UDV^{\mathrm{T}}. \end{aligned}$$

If $n \geq \frac{5}{3}m$, a modification of this algorithm by Chan (1982a,b) is more efficient than the standard Golub-Reinsch method.

Exercises

7.1. Simple matrices and the power method.

a) Let A be an $n \times n$ matrix whose elements are generated independently (but not necessarily identically) from real-valued continuous distributions. What is the probability that A is simple?

b) Under the same conditions as in Exercise 7.1a, and with $n \geq 3$, what is the probability that $|c_{n-2}| < |c_{n-1}| < |c_n|$, where c_{n-2}, c_{n-1}, and c_n are the three eigenvalues with the largest absolute values?

c) Prove that the power method converges linearly if $|c_{n-2}| < |c_{n-1}| < |c_n|$, $b_{n-1} \neq 0$, and $b_n \neq 0$. (The bs are the coefficients in the expansion of $x^{(0)}$.)
Hint: Substitute the expansion in equation (7.11) on page 314 into the expression for the convergence ratio in equation (7.12).

d) Suppose A is simple and the elements of $x^{(0)}$ are generated independently (but not necessarily identically) from continuous distributions. What is the probability that the power method will converge linearly?

7.2. Consider the matrix

$$\begin{bmatrix} 4 & 1 & 2 & 3 \\ 1 & 5 & 3 & 2 \\ 2 & 3 & 6 & 1 \\ 3 & 2 & 1 & 7 \end{bmatrix}.$$

a) Use the power method to determine the largest eigenvalue and an associated eigenvector of this matrix.
b) Find a 3×3 matrix, as in equation (7.7), that has the same eigenvalues as the remaining eigenvalues of the matrix above.
c) Using Givens transformations, reduce the matrix to upper Hessenberg form.

7.3. In the matrix

$$\begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 5 & 2 & 0 \\ 3 & 2 & 6 & 1 \\ 0 & 0 & 1 & 8 \end{bmatrix},$$

determine the Givens transformations to chase the 3 in the $(3, 1)$ position out of the matrix.

7.4. In the matrix

$$\begin{bmatrix} 2 & 1 & 0 & 0 \\ 3 & 5 & 2 & 0 \\ 0 & 0 & 6 & 1 \\ 0 & 0 & 0 & 8 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

determine the Givens transformations to chase the 3 in the $(2, 1)$ position out of the matrix.

7.5. In the QR methods for eigenvectors and singular values, why can we not just use additional orthogonal transformations to triangularize the given matrix (instead of just forming a similar Hessenberg matrix, as in Sect. 7.4) or to diagonalize the given matrix (instead of just forming the bidiagonal matrix, as in Sect. 7.7)?

7.6. Determine the eigenvalue σ_1 (on page 323) used in forming the matrix T_0 for initiating the chase in the algorithm for the singular value decomposition. Express it in terms of $a_{m,m}$, $a_{m-1,m-1}$, $a_{m-1,m}$, and $a_{m-1,m-2}$.