Numerical Analysis

EE, NCKU

Tien-Hao Chang (Darby Chang)

In the previous slide

- Eigenvalues and eigenvectors
- The power method
  - locate the dominant eigenvalue
- Inverse power method
- Deflation

In this slide

- Find all eigenvalues of a symmetric matrix
  - reducing a symmetric matrix to tridiagonal form
  - eigenvalues of symmetric tridiagonal matrices (QR algorithm)

Obtain all eigenvalues

- To compute all of the eigenvalues of a symmetric matrix, we will proceed in two stages
  - transform to symmetric tridiagonal form
    - this step requires a fixed, finite number of operations (not iterative)
  - an iterative procedure on the symmetric tridiagonal matrix that generates a sequence of matrices converged to a diagonal matrix

Why two stages?

- Why not apply the iterative technique directly on the original matrix?
  - transforming an $n \times n$ symmetric matrix to symmetric tridiagonal form requires on the order of $\frac{4}{3}n^3$ arithmetic operations
  - the iterative reduction of the symmetric tridiagonal matrix to diagonal form then requires $O(n^2)$ arithmetic operations
  - on the other hand, applying the iterative technique directly to the original matrix requires on the order of $\frac{4}{3}n^3$ arithmetic operations per iteration

4.4

Reduction to symmetric tridiagonal form

Before going into 4.4

- Similarity transformations
- Orthogonal matrices

Definition. Let A be an $n \times n$ matrix and let M be any nonsingular $n \times n$ matrix. The matrix $B = M^{-1}AM$ is said to be SIMILAR to A. The process of converting A to B is called a SIMILARITY TRANSFORMATION.

To establish that a similarity transformation does not affect any of the eigenvalues of A, we proceed as follows. The eigenvalues of B are solutions of the equation $\det(B - \lambda I) = 0$; but

$$\begin{aligned}
\det(B - \lambda I) &= \det(M^{-1}AM - \lambda I) \\
&= \det\left[M^{-1}(A - \lambda I)M\right] \\
&= \det(M^{-1})\det(A - \lambda I)\det(M) \\
&= (\det(M))^{-1}\det(A - \lambda I)\det(M) \\
&= \det(A - \lambda I).
\end{aligned}$$

Thus, $\det(B - \lambda I) = 0$ if and only if $\det(A - \lambda I) = 0$, which implies that A and B have exactly the same eigenvalues.

Similarity transformation

Theorem. Let A be an $n \times n$ matrix.

1. If A has a row or column consisting only of zero entries, then $\det(A) = 0$;
2. If A has two rows the same or two columns the same, then $\det(A) = 0$;
3. $\det(A^T) = \det(A)$;
4. If A is nonsingular, then $\det(A^{-1}) = (\det(A))^{-1}$;
5. If B is an $n \times n$ matrix, then $\det(AB) = \det(A)\det(B)$.

Recall that

http://www.dianadepasquale.com/ThinkingMonkey.jpg

Orthogonal matrix

Although any nonsingular matrix can be used to generate a similarity transformation, we would like to use matrices whose inverses are easy to compute. The class of *orthogonal matrices* will suit our needs nicely.

Definition. The $n \times n$ matrix Q is called an ORTHOGONAL MATRIX if $Q^{-1} = Q^T$.

- Similarity transformation with an orthogonal matrix maintains symmetry
  - $\boldsymbol{A}$ is symmetric and $\boldsymbol{B} = \boldsymbol{Q}^{-1}\boldsymbol{A}\boldsymbol{Q}$
  - $\boldsymbol{B}^T = (\boldsymbol{Q}^{-1}\boldsymbol{A}\boldsymbol{Q})^T = (\boldsymbol{Q}^T\boldsymbol{A}\boldsymbol{Q})^T = \boldsymbol{Q}^T\boldsymbol{A}\boldsymbol{Q} = \boldsymbol{B}$, that is, $\boldsymbol{B}$ is also symmetric
- Multiplication by an orthogonal matrix does not change the Euclidean norm
  - $(\boldsymbol{Q}\mathbf{x})^T\boldsymbol{Q}\mathbf{x} = \mathbf{x}^T\boldsymbol{Q}^T\boldsymbol{Q}\mathbf{x} = \mathbf{x}^T\boldsymbol{Q}^{-1}\boldsymbol{Q}\mathbf{x} = \mathbf{x}^T\mathbf{x}$

Householder matrix

Reducing a Symmetric Matrix to Tridiagonal Form

There are several different algorithms available for reducing a symmetric matrix to tridiagonal form. Most work in a sequential manner, applying a succession of similarity transformations which gradually produce the desired form. These techniques differ only in the family of orthogonal matrices used to generate the similarity transformations. There are also direct "tridiagonalization" techniques that determine the single orthogonal matrix needed to reduce the original matrix to tridiagonal form. These techniques work in much the same way that the direct factorization techniques we developed in Section 3.5 compute the LU decomposition of a matrix. One such technique will be treated in the exercises.

Here, we will restrict our attention to a reduction algorithm based on the use of Householder matrices.

Definition. A HOUSEHOLDER MATRIX is any matrix of the form

$$H = I - 2\mathbf{w}\mathbf{w}^T,$$

where $\mathbf{w}$ is a column vector with $\mathbf{w}^T\mathbf{w} = 1$.

It is quite easy to show that Householder matrices are both symmetric and orthogonal; that is, $H^{-1} = H$ (Exercise 2). Geometrically, multiplication of a vector $\mathbf{x}$ by the Householder matrix H results in the reflection of $\mathbf{x}$ across the hyperplane whose normal vector is $\mathbf{w}$ (see Figure 4.7).

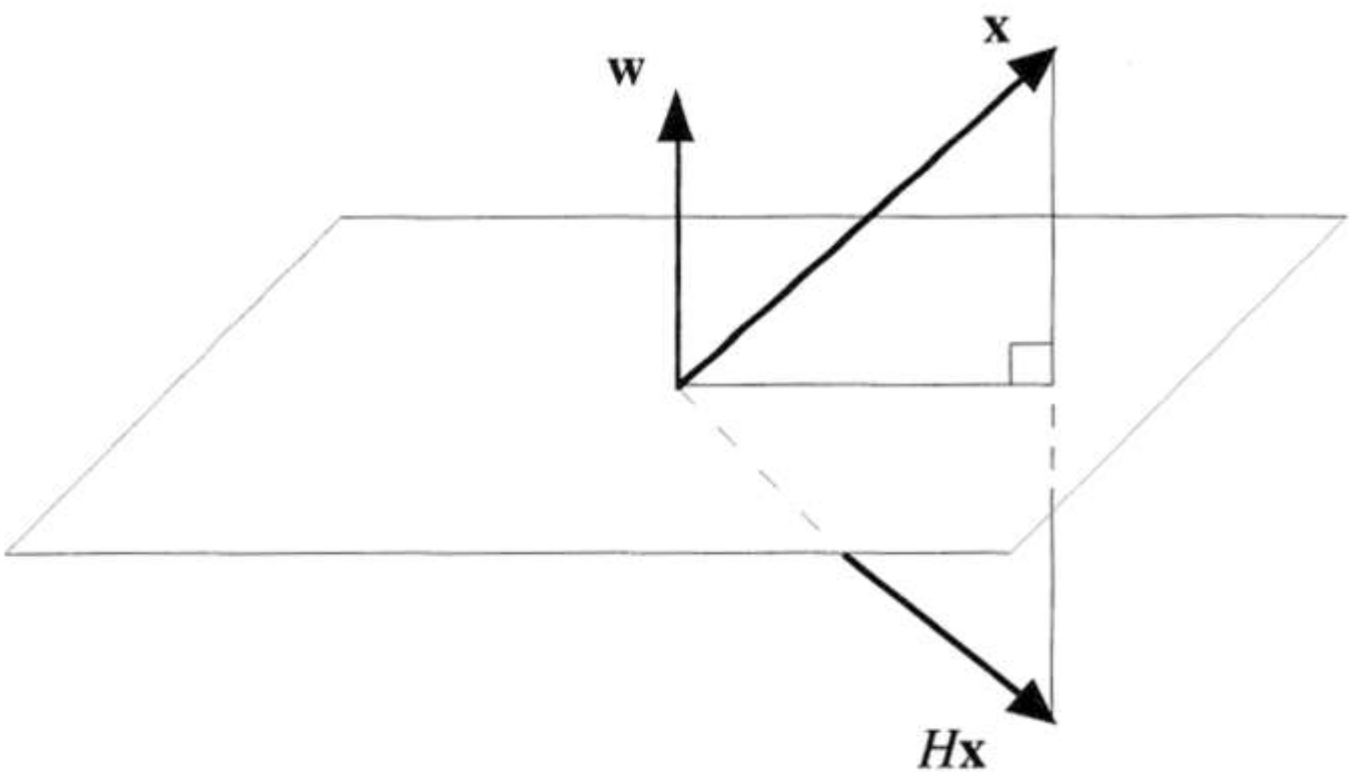

Figure 4.7 Reflection of a vector $\mathbf{x}$ across the hyperplane whose normal vector is $\mathbf{w}$.

In practice

In practice, the Householder matrices are not computed explicitly, only the vector $\mathbf{w}$ is computed. For, once the vector $\mathbf{w}$ is known, the similarity transformation HAH is given by

$$(I - 2\mathbf{w}\mathbf{w}^T)A(I - 2\mathbf{w}\mathbf{w}^T) = A - 2\mathbf{w}\mathbf{w}^T A - 2A\mathbf{w}\mathbf{w}^T + 4\mathbf{w}\mathbf{w}^T A\mathbf{w}\mathbf{w}^T,$$

which is completely determined by $\mathbf{w}$. The computation of HAH can be simplified tremendously if we define $\mathbf{u} = A\mathbf{w}$ and $K = \mathbf{w}^T\mathbf{u} = \mathbf{w}^T A\mathbf{w}$. Then

$$\begin{aligned} HAH &= A - 2\mathbf{w}\mathbf{w}^T A - 2A\mathbf{w}\mathbf{w}^T + 4\mathbf{w}\mathbf{w}^T A\mathbf{w}\mathbf{w}^T \\ &= A - 2\mathbf{w}\mathbf{u}^T - 2\mathbf{u}\mathbf{w}^T + 4K\mathbf{w}\mathbf{w}^T \\ &= A - 2\mathbf{w}(\mathbf{u}^T - K\mathbf{w}^T) - 2(\mathbf{u} - K\mathbf{w})\mathbf{w}^T. \end{aligned}$$

If we now let $\mathbf{q} = \mathbf{u} - K\mathbf{w}$, then $HAH = A - 2\mathbf{w}\mathbf{q}^T - 2\mathbf{q}\mathbf{w}^T$.

$n-2$ similarity transformations

The algorithm to reduce a symmetric matrix to tridiagonal form using Householder matrices involves a sequence of $n-2$ similarity transformations, as illustrated in Figure 4.8 for the case $n = 5$. The first Householder matrix, H_1, is selected so that $H_1 A$ will have zeros in the first $n-2$ rows of the nth column and the nth row of A will not be affected. By symmetry, when $H_1 A H_1$ is computed to complete the transformation, the zeros in the nth column will not be changed, but zeros will appear in the first $n-2$ columns of the nth row. Each subsequent Householder matrix, H_i ($i = 2, 3, 4, \ldots, n-2$), is then selected so that

$$H_i H_{i-1} \cdots H_2 H_1 A H_1 H_2 \cdots H_{i-1}$$

will have zeros in the first $n-i-1$ rows of the $(n-i+1)$st column but will not affect the bottom i rows. Completing the ith transformation will place zeros in the first $n-i-1$ columns of the $(n-i+1)$st row.

$$
\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{bmatrix}
\xrightarrow{H_1 A H_1}
\begin{bmatrix} \times & \times & \times & \times & 0 \\ \times & \times & \times & \times & 0 \\ \times & \times & \times & \times & 0 \\ \times & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{bmatrix}
$$

$$
\xrightarrow{H_2 H_1 A H_1 H_2}
\begin{bmatrix} \times & \times & \times & 0 & 0 \\ \times & \times & \times & 0 & 0 \\ \times & \times & \times & \times & 0 \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{bmatrix}
$$

$$
\xrightarrow{H_3 H_2 H_1 A H_1 H_2 H_3}
\begin{bmatrix} \times & \times & 0 & 0 & 0 \\ \times & \times & \times & 0 & 0 \\ 0 & \times & \times & \times & 0 \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{bmatrix}
$$

Figure 4.8 Illustration of Householder reduction to symmetric tridiagonal form for a 5×5 matrix. Each $\times$ denotes an element that is not necessarily zero.

Generating appropriate Householder matrices

Determining the appropriate Householder matrix for use in each step of the above algorithm requires the solution of the following fundamental problem:

> Given an integer k and an n-dimensional column vector $\mathbf{x}$, select $\mathbf{w}$ so that $H\mathbf{x} = (I - 2\mathbf{w}\mathbf{w}^T)\mathbf{x}$ has zeros in the first $n - k - 1$ rows but leaves the last k elements in $\mathbf{x}$ unchanged.

Note that this problem specification contains only $n - 1$ conditions on the vector $\mathbf{w}$. The last condition comes from the requirement that $\mathbf{w}^T\mathbf{w} = 1$, or, equivalently, that the vector $(I - 2\mathbf{w}\mathbf{w}^T)\mathbf{x}$ have the same Euclidean norm as the vector $\mathbf{x}$.

To solve this problem, first note that in order for the last k elements in $\mathbf{x}$ to be unchanged, the last k elements in $\mathbf{w}$ must be zero. This guarantees that the last k rows and columns of H are identical to the identity matrix. Thus $\mathbf{w}$ must be of the form

$$\mathbf{w} = \begin{bmatrix} w_1 & w_2 & w_3 & \cdots & w_{n-k} & 0 & \cdots & 0 \end{bmatrix}^T.$$

Let $\mathbf{b} = (I - 2\mathbf{w}\mathbf{w}^T)\mathbf{x}$, where by construction $\mathbf{b}$ will have the form

$$\mathbf{b} = \begin{bmatrix} 0 & \cdots & 0 & \alpha & x_{n-k+1} & \cdots & x_n \end{bmatrix}^T,$$

with $n - k - 1$ zeros at the beginning of the vector. Since multiplication by the Householder matrix must preserve the Euclidean norm, we must have $\mathbf{b}^T\mathbf{b} = \mathbf{x}^T\mathbf{x}$, which implies that

$$\alpha^2 = x_1^2 + x_2^2 + x_3^2 + \cdots + x_{n-k}^2.$$

To proceed further, let's rearrange the equation defining the vector $\mathbf{b}$ as

$$\mathbf{x} - 2\mathbf{w}\mathbf{w}^T\mathbf{x} = \mathbf{b}. \tag{1}$$

Premultiplying equation (1) by $\mathbf{w}^T$ yields

$$\mathbf{w}^T\mathbf{x} - 2\mathbf{w}^T\mathbf{w}\mathbf{w}^T\mathbf{x} = \mathbf{w}^T\mathbf{b},$$

which simplifies to

$$-\mathbf{w}^T\mathbf{x} = \alpha w_{n-k} \tag{2}$$

upon taking into account the form of both $\mathbf{w}$ and $\mathbf{b}$ and using the fact that $\mathbf{w}^T\mathbf{w} = 1$. Substituting equation (2) into equation (1) produces

$$\mathbf{x} + 2\alpha w_{n-k}\mathbf{w} = \mathbf{b},$$

or, in component form,

$$\begin{aligned} x_i + 2\alpha w_{n-k} w_i &= 0 \qquad (i = 1, 2, 3, \ldots, n-k-1) \\ x_{n-k} + 2\alpha w_{n-k}^2 &= \alpha. \end{aligned}$$

From the last of these equations we see that

$$w_{n-k} = \sqrt{\frac{1}{2}\left(1 - \frac{x_{n-k}}{\alpha}\right)}.$$

To avoid cancellation error, we will choose $\operatorname{sgn}(\alpha) = -\operatorname{sgn}(x_{n-k})$. With w_{n-k} determined, the remaining nonzero entries in $\mathbf{w}$ are given by

later

$$w_i = -\frac{1}{2}\frac{x_i}{\alpha w_{n-k}} \qquad (i = 1, 2, 3, \ldots, n-k-1).$$

Any Questions?

About generating Householder matrices

To proceed further, let's rearrange the equation defining the vector $\mathbf{b}$ as

$$\mathbf{x} - 2\mathbf{w}\mathbf{w}^T\mathbf{x} = \mathbf{b}. \tag{1}$$

Premultiplying equation (1) by $\mathbf{w}^T$ yields

$$\mathbf{w}^T\mathbf{x} - 2\mathbf{w}^T\mathbf{w}\mathbf{w}^T\mathbf{x} = \mathbf{w}^T\mathbf{b},$$

which simplifies to

$$-\mathbf{w}^T\mathbf{x} = \alpha w_{n-k} \tag{2}$$

upon taking into account the form of both $\mathbf{w}$ and $\mathbf{b}$ and using the fact that $\mathbf{w}^T\mathbf{w} = 1$. Substituting equation (2) into equation (1) produces

$$\mathbf{x} + 2\alpha w_{n-k}\mathbf{w} = \mathbf{b},$$

or, in component form,

$$\begin{aligned} x_i + 2\alpha w_{n-k} w_i &= 0 \qquad (i = 1, 2, 3, \ldots, n-k-1) \\ x_{n-k} + 2\alpha w_{n-k}^2 &= \alpha. \end{aligned}$$

From the last of these equations we see that

$$w_{n-k} = \sqrt{\frac{1}{2}\left(1 - \frac{x_{n-k}}{\alpha}\right)}.$$

To avoid cancellation error, we will choose $\operatorname{sgn}(\alpha) = -\operatorname{sgn}(x_{n-k})$. With w_{n-k} determined, the remaining nonzero entries in $\mathbf{w}$ are given by

Is there any possible cancellation errors?

$$w_i = -\frac{1}{2}\frac{x_i}{\alpha w_{n-k}} \qquad (i = 1, 2, 3, \ldots, n-k-1).$$

To proceed further, let's rearrange the equation defining the vector $\mathbf{b}$ as

$$\mathbf{x} - 2\mathbf{w}\mathbf{w}^T\mathbf{x} = \mathbf{b}. \tag{1}$$

Premultiplying equation (1) by $\mathbf{w}^T$ yields

$$\mathbf{w}^T\mathbf{x} - 2\mathbf{w}^T\mathbf{w}\mathbf{w}^T\mathbf{x} = \mathbf{w}^T\mathbf{b},$$

which simplifies to

$$-\mathbf{w}^T\mathbf{x} = \alpha w_{n-k} \tag{2}$$

upon taking into account the form of both $\mathbf{w}$ and $\mathbf{b}$ and using the fact that $\mathbf{w}^T\mathbf{w} = 1$. Substituting equation (2) into equation (1) produces

$$\mathbf{x} + 2\alpha w_{n-k}\mathbf{w} = \mathbf{b},$$

or, in component form,

$$\begin{aligned} &x_i + 2\alpha w_{n-k} w_i = 0 \qquad (i = 1, 2, 3, \ldots, n-k-1) \\ &x_{n-k} + 2\alpha w_{n-k}^2 = \alpha. \end{aligned}$$

From the last of these equations we see that

$$w_{n-k} = \sqrt{\frac{1}{2}\left(1 - \frac{x_{n-k}}{\alpha}\right)}.$$

To avoid cancellation error, we will choose $\mathrm{sgn}(\alpha) = -\mathrm{sgn}(x_{n-k})$. With w_{n-k} determined, the remaining nonzero entries in $\mathbf{w}$ are given by

$$w_i = -\frac{1}{2}\frac{x_i}{\alpha w_{n-k}} \quad (i = 1, 2, 3, \ldots, n-k-1).$$

http://thomashawk.com/hello/209/1017/1024/Jackson%20Running.jpg

In action

EXAMPLE 4.8 Reduction to Tridiagonal Form

Consider the symmetric 4×4 matrix

$$A = \begin{bmatrix} -1 & -2 & 1 & 2 \\ -2 & 3 & 0 & -2 \\ 1 & 0 & 2 & 1 \\ 2 & -2 & 1 & 4 \end{bmatrix}.$$

For the first step of the reduction to tridiagonal form, we want to produce zeros in the first two rows of the last column of A and leave the last element in that column alone. Therefore, we are working with $k = 1$ and the vector $\mathbf{x} = \begin{bmatrix} 2 & -2 & 1 & 4 \end{bmatrix}^T$. With this vector, we compute $\alpha^2 = 2^2 + (-2)^2 + 1^2 = 9$ and since $\operatorname{sgn}(x_3) = +1$, we choose $\alpha = -3$. It then follows that

$$w_3 = \sqrt{\frac{1}{2}\left(1 - \frac{1}{-3}\right)} = \frac{\sqrt{6}}{3};$$

$$w_2 = -\frac{1}{2}\frac{-2}{-3(\sqrt{6}/3)} = -\frac{\sqrt{6}}{6}; \text{ and}$$

$$w_1 = -\frac{1}{2}\frac{2}{-3(\sqrt{6}/3)} = \frac{\sqrt{6}}{6}.$$

Hence, $\mathbf{w} = (\sqrt{6}/6)\begin{bmatrix} 1 & -1 & 2 & 0 \end{bmatrix}^T$. Next, we compute

$$\mathbf{u} = A\mathbf{w} = (\sqrt{6}/6)\begin{bmatrix} 3 & -5 & 5 & 6 \end{bmatrix}^T;$$

$$K = \mathbf{w}^T\mathbf{u} = 3; \text{and}$$

$$\mathbf{q} = \mathbf{u} - K\mathbf{w} = (\sqrt{6}/6)\begin{bmatrix} 0 & -2 & -1 & 6 \end{bmatrix}^T.$$

Therefore,

$$H_1AH_1 = A - \frac{1}{3}\begin{bmatrix} 1 \\ -1 \\ 2 \\ 0 \end{bmatrix}\begin{bmatrix} 0 & -2 & -1 & 6 \end{bmatrix} - \frac{1}{3}\begin{bmatrix} 0 \\ -2 \\ -1 \\ 6 \end{bmatrix}\begin{bmatrix} 1 & -1 & 2 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} -1 & -4/3 & 4/3 & 0 \\ -4/3 & 5/3 & 1 & 0 \\ 4/3 & 1 & 10/3 & -3 \\ 0 & 0 & -3 & 4 \end{bmatrix}.$$

For the second (and final) step of the reduction, we want to produce a zero in the first row of the third column of H_1AH_1 and leave the last two elements in that column alone. Therefore, we are working with $k = 2$ and the vector $\mathbf{x} = \begin{bmatrix} 4/3 & 1 & 10/3 & -3 \end{bmatrix}^T$. With this vector, we compute $\alpha^2 = 25/9$ and since $\mathrm{sgn}(x_2) = +1$, we choose $\alpha = -5/3$. It then follows that

$$w_2 = \sqrt{\frac{1}{2}\left(1 - \frac{1}{-5/3}\right)} = \frac{2\sqrt{5}}{5} \quad \text{and}$$

$$w_1 = -\frac{1}{2}\frac{4/3}{(-5/3)(2\sqrt{5}/5)} = \frac{\sqrt{5}}{5}.$$

Hence, $\mathbf{w} = (\sqrt{5}/5)\begin{bmatrix} 1 & 2 & 0 & 0 \end{bmatrix}^T$. Next, we compute

$$\mathbf{u} = A\mathbf{w} = (\sqrt{5}/5)\begin{bmatrix} -11/3 & 2 & 10/3 & 0 \end{bmatrix}^T;$$
$$K = \mathbf{w}^T\mathbf{u} = 1/15; \quad \text{and}$$
$$\mathbf{q} = \mathbf{u} - K\mathbf{w} = (\sqrt{5}/5)\begin{bmatrix} -56/15 & 28/15 & 10/3 & 0 \end{bmatrix}^T.$$

Therefore,

$$\begin{aligned} H_2H_1AH_1H_2 &= H_1AH_1 - \frac{2}{5}\begin{bmatrix} 1 \\ 2 \\ 0 \\ 0 \end{bmatrix}\begin{bmatrix} -\frac{56}{15} & \frac{28}{15} & \frac{10}{3} & 0 \end{bmatrix} \\ &\quad - \frac{2}{5}\begin{bmatrix} -56/15 \\ 28/15 \\ 10/3 \\ 0 \end{bmatrix}\begin{bmatrix} 1 & 2 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 149/75 & 68/75 & 0 & 0 \\ 68/75 & -33/25 & -5/3 & 0 \\ 0 & -5/3 & 10/3 & -3 \\ 0 & 0 & -3 & 4 \end{bmatrix}. \end{aligned}$$

Any Questions?

4.4 Reduction to symmetric tridiagonal form

4.5

Eigenvalues of symmetric tridiagonal matrices

The Very Basics of the QR Algorithm

We will start with a basic description of the QR algorithm and gradually develop the details. Let $A = A^{(0)}$ be a given matrix. The QR algorithm constructs the sequence of matrices $\{A^{(i)}\}$ as follows: for $i = 0, 1, 2, \ldots,$

- factor $A^{(i)}$ into the product $Q^{(i)}R^{(i)}$, where $Q^{(i)}$ is an orthogonal matrix (i.e., $\left[Q^{(i)}\right]^{-1} = \left[Q^{(i)}\right]^{T}$) and $R^{(i)}$ is an upper triangular matrix; and
- compute $A^{(i+1)} = R^{(i)}Q^{(i)}$.

From the relation $A^{(i)} = Q^{(i)}R^{(i)}$, it follows that ${Q^{(i)}}^{T}A^{(i)} = R^{(i)}$, since $Q^{(i)}$ is an orthogonal matrix. The calculation in the second step is then equivalent to $A^{(i+1)} = R^{(i)}Q^{(i)} = {Q^{(i)}}^{T}A^{(i)}Q^{(i)}$. Hence, each iteration performs a similarity transformation with an orthogonal matrix, which implies that the eigenvalues of $A^{(i+1)}$ are identical to those of $A^{(i)}$.

As just described, the QR algorithm can be applied to any matrix. We will, however, discuss the implementation of the QR algorithm for symmetric tridiagonal matrices only. For details of the algorithm applied to more general matrices, consult Wilkinson [1], Golub and van Loan [2], or Press, et. al. [3].

What is the effect of performing the iterations of the QR algorithm? Consider the symmetric tridiagonal matrix

$$A^{(0)} = \begin{bmatrix} 4 & 3 & 0 \\ 3 & 1 & -1 \\ 0 & -1 & 3 \end{bmatrix}.$$

A portion of the sequence $\{A^{(i)}\}$ is

$$A^{(2)} = \begin{bmatrix} 5.923 & -0.276 & 0 \\ -0.276 & 2.227 & -1.692 \\ 0 & -1.692 & -0.155 \end{bmatrix}$$

$$A^{(4)} = \begin{bmatrix} 5.950 & -0.0664 & 0 \\ -0.0664 & 3.071 & -0.241 \\ 0 & -0.241 & -1.021 \end{bmatrix}$$

$$A^{(6)} = \begin{bmatrix} 5.951 & -0.0178 & 0 \\ -0.0178 & 3.084 & -0.0272 \\ 0 & -0.0272 & -1.035 \end{bmatrix}$$

$$A^{(8)} = \begin{bmatrix} 5.951 & -0.00478 & 0 \\ -0.00478 & 3.084 & -0.00306 \\ 0 & -0.00306 & -1.035 \end{bmatrix}$$

$$A^{(10)} = \begin{bmatrix} 5.951 & -0.00128 & 0 \\ -0.00128 & 3.084 & -0.000345 \\ 0 & -0.000345 & -1.035 \end{bmatrix}.$$

The off-diagonal elements → zero while the diagonal elements → the eigenvalues (in decreasing order)

QR factorization

The heart of the QR algorithm

Rotation matrix

Definition. Let $i < j$. The orthogonal matrix, $P_{(i,j)}$, which is identical to the identity matrix with the exception that

$$p_{i,i} = p_{j,j} = \cos\theta \quad \text{and} \quad p_{i,j} = -p_{j,i} = \sin\theta,$$

for some angle θ, is called a ROTATION MATRIX.

The name *rotation matrix* arises from the geometric fact that $P_{(i,j)}$ represents the rotation of the ith and jth axes about the origin of the coordinate system by an angle of θ. For later use, it is important to note that premultiplication of an arbitrary matrix, M, by $P_{(i,j)}$ affects only the ith and jth rows. In particular,

$$\begin{matrix} i\text{th row of} \\ P_{(i,j)}M \end{matrix} = \cos\theta \cdot \begin{matrix} i\text{th row} \\ \text{of } M \end{matrix} + \sin\theta \cdot \begin{matrix} j\text{th row} \\ \text{of } M \end{matrix}$$

and

$$\begin{matrix} j\text{th row of} \\ P_{(i,j)}M \end{matrix} = -\sin\theta \cdot \begin{matrix} i\text{th row} \\ \text{of } M \end{matrix} + \cos\theta \cdot \begin{matrix} j\text{th row} \\ \text{of } M \end{matrix} .$$

QR factorization

The factorization of the symmetric tridiagonal matrix $A^{(i)}$ now proceeds in exactly the same manner as the matrix factorization algorithms we developed in Chapter 3. For an $n \times n$ matrix, we make $n-1$ passes through the matrix, with each pass "zeroing" out a specific element below the main diagonal. Thus, in the first pass, $P_{(1,2)}$ is chosen so that $P_{(1,2)}A^{(i)}$ has a zero in row 2, column 1. Next, $P_{(2,3)}$ is chosen so that $P_{(2,3)}P_{(1,2)}A^{(i)}$ has a zero in the third row of the second column, $P_{(3,4)}$ is chosen so that $P_{(3,4)}P_{(2,3)}P_{(1,2)}A^{(i)}$ has a zero in the fourth row of the third column, and so on. Finally, $P_{(n-1,n)}$ is chosen so that $P_{(n-1,n)} \cdots P_{(3,4)}P_{(2,3)}P_{(1,2)}A^{(i)}$ is an upper triangular matrix. Hence, $R^{(i)} = P_{(n-1,n)} \cdots P_{(3,4)}P_{(2,3)}P_{(1,2)}A^{(i)}$.

To examine the details of this factorization scheme more closely, let

$$A^{(i)} = \begin{bmatrix} a_1 & b_1 & & & & & \\ b_1 & a_2 & b_2 & & & & \\ & b_2 & a_3 & b_3 & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & \cdot \\ & & & & b_{n-2} & a_{n-1} & b_{n-1} \\ & & & & & b_{n-1} & a_n \end{bmatrix}.$$

For notational convenience, let the cosine and sine values associated with the rotation matrix $P_{(j,j+1)}$ be denoted by c_j and s_j, respectively. Carrying out the multiplication $P_{(1,2)}A^{(i)}$, we find

$$\begin{bmatrix} c_1 & s_1 & & \\ -s_1 & c_1 & & \\ & & 1 & \\ & & & \ddots \end{bmatrix} \begin{bmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & b_3 & \\ & & \ddots & \ddots & \ddots \end{bmatrix}$$

$$= \begin{bmatrix} a_1c_1 + b_1s_1 & b_1c_1 + a_2s_1 & b_2s_1 & & \\ -a_1s_1 + b_1c_1 & -b_1s_1 + a_2c_1 & b_2c_1 & & \\ & b_2 & a_3 & b_3 & \\ & & \ddots & \ddots & \ddots \end{bmatrix}. \quad (1)$$

We now want to choose c_1 and s_1 so that $-a_1 s_1 + b_1 c_1 = 0$. One solution of this equation, which also satisfies the fundamental trig identity $c_1^2 + s_1^2 = 1$, is

$$c_1 = \frac{a_1}{\sqrt{a_1^2 + b_1^2}} \quad \text{and} \quad s_1 = \frac{b_1}{\sqrt{a_1^2 + b_1^2}}.$$

With c_1 and s_1 selected so that $-a_1 s_1 + b_1 c_1 = 0$, note that a_1 appears on the right-hand side of (1) in the first row, first column only—the precise location of a_1 in the matrix $A^{(i)}$. We may therefore overwrite a_1 with the expression

$$a_1 c_1 + b_1 s_1 = a_1 \frac{a_1}{\sqrt{a_1^2 + b_1^2}} + b_1 \frac{b_1}{\sqrt{a_1^2 + b_1^2}} = \sqrt{a_1^2 + b_1^2}.$$

In a similar manner, we would like to save the first two elements in the second column of $P_{(1,2)} A^{(i)}$ in place of b_1 and a_2; unfortunately, to calculate these elements, both b_1 and a_2 are required. However, if we save the current value of b_1 in a temporary variable, say t, we may then overwrite b_1 with the expression $tc_1 + a_2 s_1$ and a_2 with $-ts_1 + a_2 c_1$. Finally, we save the value of b_2 in the variable t and then overwrite b_2 with the quantity $b_2 c_1 = tc_1$. We need to save b_2 in order to calculate the sine and cosine values associated with the next rotation matrix.

It turns out that the element in the third column of the first row of $P_{(1,2)}A^{(i)}$, b_2s_1, does not need to be saved. The remaining passes in the factorization step do not involve the first row, so the indicated element will not be needed for any later calculations. Further, as we will see shortly, the calculation of the product $R^{(i)}Q^{(i)}$ can be carried out without knowing this element. Technically, by not saving the value b_2s_1, we are not obtaining the true QR factorization of the matrix $A^{(i)}$. We are, however, maintaining all the information we will need to calculate $R^{(i)}Q^{(i)} = A^{(i+1)}$, which, in our present circumstances, is the real objective.

The calculations required by all subsequent passes in the factorization step are identical to those indicated for the first pass, with two exceptions. First, we of course need to increment the subscripts for each new pass. Second, for the jth pass, with $j = 2, 3, 4, ..., n-1$, c_j and s_j are given by

$$c_j = \frac{a_j}{\sqrt{a_j^2 + t^2}} \qquad \text{and} \qquad s_j = \frac{t}{\sqrt{a_j^2 + t^2}}$$

since we've used t to save the old value of b_j. We can therefore implement the entire factorization process as follows.

```
save b_1 in the temporary variable t
for j = 1, 2, 3, ..., n − 1
        let r = sqrt(a_j^2 + t^2)
        compute c_j = a_j/r and s_j = t/r
        overwrite a_j with r
        save b_j in t
        overwrite b_j with t c_j + a_{j+1} s_j
        overwrite a_{j+1} with −t s_j + a_{j+1} c_j
        if ( j ≠ n − 1 )
                save b_{j+1} in t
                overwrite b_{j+1} with t c_j
        end
end
```

The first line in this pseudocode has been included so that the first pass can be handled in the same manner as all of the later passes. The final two statements have been placed inside a conditional statement since, during the last pass through the matrix, there is no element $b_{j+1} = b_n$ to overwrite.

http://thomashawk.com/hello/209/1017/1024/Jackson%20Running.jpg

In action

EXAMPLE 4.10 The QR Factorization of a Symmetric Tridiagonal Matrix

Consider again the symmetric tridiagonal matrix

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

For this example, we have

$$a_1 = 4, \quad a_2 = 1, \quad a_3 = 3, \quad b_1 = 3, \quad \text{and} \quad b_2 = -1.$$

To prepare for the first pass, we set $t = b_1 = 3$. We then calculate

$$r = \sqrt{a_1^2 + t^2} = 5, \quad c_1 = \frac{a_1}{r} = \frac{4}{5} \quad \text{and} \quad s_1 = \frac{t}{r} = \frac{3}{5},$$

and set $a_1 = r = 5$. Next, we set $t = b_1 = 3$ and then calculate

$$\begin{aligned} b_1 &= tc_1 + a_2 s_1 = 3; \quad \text{and} \\ a_2 &= -ts_1 + a_2 c_1 = -1. \end{aligned}$$

Finally, set $t = b_2 = -1$ and calculate $b_2 = tc_1 = -\frac{4}{5}$.

The second pass starts with the calculations

$$r = \sqrt{a_1^2 + t^2} = \sqrt{2}, \quad c_2 = \frac{a_2}{r} = -\frac{1}{\sqrt{2}}, \quad \text{and} \quad s_2 = \frac{t}{r} = -\frac{1}{\sqrt{2}}.$$

After setting $a_2 = r = \sqrt{2}$ and $t = b_2 = -\frac{4}{5}$, we then calculate

$$\begin{aligned} b_2 &= tc_2 + a_3 s_2 = -\tfrac{11}{5\sqrt{2}}; \quad \text{and} \\ a_3 &= -ts_2 + a_3 c_2 = -\tfrac{19}{5\sqrt{2}}. \end{aligned}$$

The results of our factorization of A are therefore

$$\begin{array}{lll} a_1 = 5, & a_2 = \sqrt{2}, & a_3 = -\frac{19}{5\sqrt{2}}, \\ b_1 = 3, & b_2 = -\frac{11}{5\sqrt{2}}, & \\ c_1 = \frac{4}{5}, & s_1 = \frac{3}{5}, & \\ c_2 = -\frac{1}{\sqrt{2}}, & s_2 = -\frac{1}{\sqrt{2}}. & \end{array}$$

We will now examine how to use these values to compute the product $R^{(0)}Q^{(0)}$.

The product $\boldsymbol{R}^{(i)}\boldsymbol{Q}^{(i)}$

The Very Basics of the QR Algorithm

We will start with a basic description of the QR algorithm and gradually develop the details. Let $A = A^{(0)}$ be a given matrix. The QR algorithm constructs the sequence of matrices $\{A^{(i)}\}$ as follows: for $i = 0, 1, 2, \ldots$,

- factor $A^{(i)}$ into the product $Q^{(i)}R^{(i)}$, where $Q^{(i)}$ is an orthogonal matrix (i.e., $\left[Q^{(i)}\right]^{-1} = \left[Q^{(i)}\right]^{T}$) and $R^{(i)}$ is an upper triangular matrix; and
- compute $A^{(i+1)} = R^{(i)}Q^{(i)}$.

From the relation $A^{(i)} = Q^{(i)}R^{(i)}$, it follows that $Q^{(i)^T}A^{(i)} = R^{(i)}$, since $Q^{(i)}$ is an orthogonal matrix. The calculation in the second step is then equivalent to $A^{(i+1)} = R^{(i)}Q^{(i)} = Q^{(i)^T}A^{(i)}Q^{(i)}$. Hence, each iteration performs a similarity transformation with an orthogonal matrix, which implies that the eigenvalues of $A^{(i+1)}$ are identical to those of $A^{(i)}$.

As just described, the QR algorithm can be applied to any matrix. We will, however, discuss the implementation of the QR algorithm for symmetric tridiagonal matrices only. For details of the algorithm applied to more general matrices, consult Wilkinson [1], Golub and van Loan [2], or Press, et. al. [3].

Recall that

http://www.dianadepasquale.com/ThinkingMonkey.jpg

The Product $R^{(i)}Q^{(i)}$

Earlier, we established that the upper triangular matrix in the QR factorization of the matrix $A^{(i)}$ is given by $R^{(i)} = P_{(n-1,n)} \cdots P_{(3,4)}P_{(2,3)}P_{(1,2)}A^{(i)}$. Combining this expression with the equation ${Q^{(i)}}^T A^{(i)} = R^{(i)}$, we see that ${Q^{(i)}}^T = P_{(n-1,n)} \cdots P_{(3,4)}P_{(2,3)}P_{(1,2)}$. This, in turn, implies that

$$Q^{(i)} = P^T_{(1,2)}P^T_{(2,3)}P^T_{(3,4)} \cdots P^T_{(n-1,n)}.$$

To form the product $R^{(i)}Q^{(i)}$, however, there is no need to compute the matrix $Q^{(i)}$ explicitly. Instead, we can save the s_j and c_j values associated with each rotation matrix, $P_{(j,j+1)}$, and then postmultiply $R^{(i)}$ by the transpose of each rotation matrix in succession. To carry out each multiplication we make use of the relations

$$\begin{array}{c} i\text{th column of} \\ MP^T_{(i,j)} \end{array} = \cos\theta \cdot \begin{array}{c} i\text{th column} \\ \text{of } M \end{array} + \sin\theta \cdot \begin{array}{c} j\text{th column} \\ \text{of } M \end{array} \tag{2}$$

and

$$\begin{array}{c} j\text{th column of} \\ MP^T_{(i,j)} \end{array} = -\sin\theta \cdot \begin{array}{c} i\text{th column} \\ \text{of } M \end{array} + \cos\theta \cdot \begin{array}{c} j\text{th column} \\ \text{of } M \end{array} . \tag{3}$$

As with the factorization process, we can deduce the complete sequence of calculations for obtaining the product $R^{(i)}Q^{(i)}$ by examining just the first multiplication, $R^{(i)}P^T_{(1,2)}$. We find

$$\begin{bmatrix} a_1 & b_1 & e_1 & & & \\ & a_2 & b_2 & e_2 & & \\ & & a_3 & b_3 & e_3 & \\ & & & \ddots & \ddots & \ddots \end{bmatrix} \begin{bmatrix} c_1 & -s_1 & & \\ s_1 & c_1 & & \\ & & 1 & \\ & & & \ddots \end{bmatrix}$$

$$= \begin{bmatrix} a_1c_1 + b_1s_1 & -a_1s_1 + b_1c_1 & e_1 & & & \\ a_2s_1 & a_2c_1 & b_2 & e_2 & & \\ & & a_3 & b_3 & e_3 & \\ & & & \ddots & \ddots & \ddots \end{bmatrix}. \quad (4)$$

Here, the e_j denote the values that we know are present in $R^{(i)}$ but that we did not save during the factorization step.

We now make two very important observations. First, based on equations (2) and (3), we know that postmultiplication by $P^T_{(2,3)}, P^T_{(3,4)}, \ldots, P^T_{(n-1,n)}$ will have no effect on the first column of $R^{(i)}P^T_{(1,2)}$. Therefore, the first column of $R^{(i)}P^T_{(1,2)}$, as shown on the right-hand side of (4), is the first column of $R^{(i)}Q^{(i)}$. Second, since $A^{(0)}$ is symmetric and each $Q^{(i)}$ is orthogonal, it follows that $A^{(i+1)} = R^{(i)}Q^{(i)}$ must also be symmetric (see Exercise 1). Consequently, not only do we know the first column of $R^{(i)}Q^{(i)}$ after this first multiplication, we know the first row as well. Thus, calculation of the values along the main diagonal and below must be carried out, but calculations above the main diagonal are unnecessary.

The algorithm for $\boldsymbol{R}^{(i)}\boldsymbol{Q}^{(i)}$

Bringing all this information together, it follows that to obtain the product $R^{(i)}Q^{(i)}$, we need to perform the operations

overwrite a_j with $a_jc_j + b_js_j$;
overwrite b_j with $a_{j+1}s_j$; and
overwrite a_{j+1} with $a_{j+1}c_j$,

for $j = 1, 2, 3, ..., n-1$. Note that the e_j do not play a role in any of these computations.

http://thomashawk.com/hello/209/1017/1024/Jackson%20Running.jpg

In action

EXAMPLE 4.11 The Product $R^{(0)}Q^{(0)}$ from the Previous Example

The results of our factorization of

$$A = \begin{bmatrix} 3 & 2 & 0 \\ 2 & -5 & -1 \\ 0 & -1 & 4 \end{bmatrix}$$

were

$$\begin{array}{lll} a_1 = 5, & a_2 = \sqrt{2}, & a_3 = -\frac{19}{5\sqrt{2}}, \\ b_1 = 3, & b_2 = -\frac{11}{5\sqrt{2}}, & \\ c_1 = \frac{4}{5}, & s_1 = \frac{3}{5}, & \\ c_2 = -\frac{1}{\sqrt{2}}, & s_2 = -\frac{1}{\sqrt{2}}. & \end{array}$$

The first set of calculations leading to the product $R^{(0)}Q^{(0)}$ yields

$$\begin{aligned} a_1 &= a_1c_1 + b_1s_1 = \frac{29}{5} = 5.8; \\ b_1 &= a_2s_1 = \frac{3\sqrt{2}}{5}; \quad \text{and} \\ a_2 &= a_2c_1 = \frac{4\sqrt{2}}{5}. \end{aligned}$$

The second set of calculations then gives

$$\begin{aligned} a_2 &= a_2c_2 + b_2s_2 = -\frac{4}{5} + \frac{11}{10} = 0.3; \\ b_2 &= a_3s_2 = 1.9; \quad \text{and} \\ a_3 &= a_3c_2 = 1.9. \end{aligned}$$

Hence,

$$A^{(1)} = R^{(0)}Q^{(0)} = \begin{bmatrix} 5.8 & 0.848528 & 0 \\ 0.848528 & 0.3 & 1.9 \\ 0 & 1.9 & 1.9 \end{bmatrix}.$$

Any Questions?

Eigenvalues and eigenvectors

Exercise 4

2010/5/30 2:00pm

Email to darby@ee.ncku.edu.tw or hand over in class. Note that the fourth problem is a programming work.

In Exercises 1–7, a matrix A and a vector $\mathbf{x}^{(0)}$ are given. Perform five iterations of the appropriate version of the power method.

1. $A = \begin{bmatrix} 3 & 2 & -2 \\ -3 & -1 & 3 \\ 1 & 2 & 0 \end{bmatrix}$ and $\mathbf{x}^{(0)} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$

In Exercises 1–4, approximate the eigenvalue of the given matrix that is nearest to the indicated value, and determine its associated eigenvector. In each case use a convergence tolerance of 5×10^{-5}.

1. $A = \begin{bmatrix} 1 & 4 & 5 \\ 4 & -3 & 0 \\ 5 & 0 & 7 \end{bmatrix} \quad q = 1$

1. For each of the following matrices, an eigenvalue-eigenvector pair is given. Determine the deflation vector $\mathbf{x}$ and the deflated matrix B corresponding to Wielandt deflation.

(a) $A = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 2 \\ 0 & 1 & 3 \end{bmatrix}, \qquad \lambda_1 = 4, \qquad \mathbf{v}_1 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$

(b) $A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 4 & -17 & 8 \end{bmatrix}, \qquad \lambda_1 = 4, \qquad \mathbf{v}_1 = \begin{bmatrix} 1/16 \\ 1/4 \\ 1 \end{bmatrix}$

(c) $A = \begin{bmatrix} 19 & -9 & -6 \\ 25 & -11 & -9 \\ 17 & -9 & -4 \end{bmatrix}, \qquad \lambda_1 = 2, \qquad \mathbf{v}_1 = \begin{bmatrix} 3/4 \\ 3/4 \\ 1 \end{bmatrix}$

(d) $A = \begin{bmatrix} 4 & 2 & -2 & 2 \\ 1 & 3 & 1 & -1 \\ 0 & 0 & 2 & 0 \\ 1 & 1 & -3 & 5 \end{bmatrix}, \qquad \lambda_1 = 6, \qquad \mathbf{v}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$

Write a program to obtain R given a symmetric tridiagonal matrix A

1. Let $A^{(0)}$ be a symmetric matrix. Prove that the matrices $A^{(i)}$ produced by the QR algorithm are symmetric for all i.