9. QR algorithm

- tridiagonal symmetric matrices
- basic QR algorithm
- QR algorithm with shifts

QR algorithm

- the standard method for computing eigenvalues and eigenvectors
- we discuss the algorithm for symmetric eigendecomposition

$$A = Q\Lambda Q^T = \sum_{i=1}^{n} \lambda_i q_i q_i^T$$

there are two stages

1. reduce A to tridiagonal form by an orthogonal similarity transformation

$$Q_1^T A Q_1 = T, \qquad T \text{ tridiagonal, } Q_1 \text{ orthogonal}$$

2. compute eigendecomposition $T = Q_2 \Lambda Q_2^T$ by a fast iterative algorithm

Outline

- **tridiagonal symmetric matrices**
- basic QR algorithm
- QR algorithm with shifts

Reflector

$$Q = I - vv^T$$

- v is a vector with norm $\|v\| = \sqrt{2}$
- Q is symmetric and orthogonal
- product $Qx = x - (v^T x)v$ requires $4m$ flops if x and v are m-vectors

Reflection to multiple of first unit vector (133A lecture 6)

- an easily constructed reflector maps a given y to a multiple of $e_1 = (1, 0, \ldots, 0)$
- if $y \neq 0$, choose the reflector defined by

$$v = \frac{\sqrt{2}}{\|w\|} w, \qquad w = y + \operatorname{sign}(y_1)\|y\| e_1$$

this reflector maps y to $Qy = -\operatorname{sign}(y_1)\|y\| e_1$

Reduction to tridiagonal form

given an $n \times n$ symmetric matrix A, find orthogonal Q such that

$$Q^T A Q = \begin{bmatrix} a_1 & b_1 & 0 & \cdots & 0 & 0 & 0 \\ b_1 & a_2 & b_2 & \cdots & 0 & 0 & 0 \\ 0 & b_2 & a_3 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a_{n-2} & b_{n-2} & 0 \\ 0 & 0 & 0 & \cdots & b_{n-2} & a_{n-1} & b_{n-1} \\ 0 & 0 & 0 & \cdots & 0 & b_{n-1} & a_n \end{bmatrix}$$

- this can be achieved by a product of reflectors $Q = Q_1 Q_2 \cdots Q_{n-2}$
- complexity is order n^3

First step

partition A as

$$A = \begin{bmatrix} a_1 & c_1^T \\ c_1 & B_1 \end{bmatrix} \qquad c_1 \text{ is an } (n-1)\text{-vector, } B_1 \text{ is } (n-1)\times(n-1)$$

- find a reflector $I - v_1 v_1^T$ that maps c_1 to $b_1 e_1$ and define

$$Q_1 = \begin{bmatrix} 1 & 0 \\ 0 & I - v_1 v_1^T \end{bmatrix}$$

- multiply A with Q_1 to introduce zeros in positions $3, \ldots, n$ of 1st column and row

$$\begin{aligned} Q_1 A Q_1 &= \begin{bmatrix} a_1 & c_1^T(I - v_1 v_1^T) \\ (I - v_1 v_1^T)c_1 & (I - v_1 v_1^T)B_1(I - v_1 v_1^T) \end{bmatrix} \\ &= \begin{bmatrix} a_1 & b_1 e_1^T \\ b_1 e_1 & B_1 - v_1 w_1^T - w_1 v_1^T \end{bmatrix} \qquad \text{where } w_1 = B_1 v_1 - \frac{v_1^T B_1 v_1}{2} v_1 \end{aligned}$$

- computation of $2,2$ block requires order $4n^2$ flops

General step

after $k-1$ steps,

$$
Q_{k-1}\cdots Q_1 A Q_1 \cdots Q_{k-1} = \left[\begin{array}{cccccc|cc}
a_1 & b_1 & 0 & \cdots & 0 & 0 & 0 & 0 \\
b_1 & a_2 & b_2 & \cdots & 0 & 0 & 0 & 0 \\
0 & b_2 & a_3 & \cdots & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & a_{k-2} & b_{k-2} & 0 & 0 \\
0 & 0 & 0 & \cdots & b_{k-2} & a_{k-1} & b_{k-1} & 0 \\
\hline
0 & 0 & 0 & \cdots & 0 & b_{k-1} & a_k & c_k^T \\
0 & 0 & 0 & \cdots & 0 & 0 & c_k & B_k
\end{array}\right]
$$

- find a reflector $I - v_k v_k^T$ that maps the $(n-k)$-vector c_k to $b_k e_1$ and define

$$
Q_k = \begin{bmatrix} I_{k-1} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & I - v_k v_k^T \end{bmatrix}, \qquad (I - v_k v_k^T) B_k (I - v_k v_k^T) = \begin{bmatrix} a_{k+1} & c_{k+1}^T \\ c_{k+1} & B_{k+1} \end{bmatrix}
$$

- complexity of step k is $4(n-k)^2$ plus lower order terms

Complexity

- complexity for complete algorithm is dominated by

$$\sum_{k=1}^{n-2} 4(n-k)^2 \approx \frac{4}{3}n^3$$

- Q is stored in factored form (the vectors v_k are stored)
- if needed, assembling the matrix Q adds another order n^3 term

QR factorization of tridiagonal matrix

suppose A is $n \times n$ and tridiagonal, with QR factorization

$$A = QR$$

Q and R have a special structure (dots indicate possible nonzero elements):

$$\begin{bmatrix} \bullet & \bullet & & & & \\ \bullet & \bullet & \bullet & & & \\ & \bullet & \bullet & \bullet & & \\ & & \bullet & \bullet & \bullet & \\ & & & \bullet & \bullet & \bullet \\ & & & & \bullet & \bullet \end{bmatrix} = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ & \bullet & \bullet & \bullet & \bullet & \bullet \\ & & \bullet & \bullet & \bullet & \bullet \\ & & & \bullet & \bullet & \bullet \\ & & & & \bullet & \bullet \end{bmatrix} \begin{bmatrix} \bullet & \bullet & \bullet & & & \\ & \bullet & \bullet & \bullet & & \\ & & \bullet & \bullet & \bullet & \\ & & & \bullet & \bullet & \bullet \\ & & & & \bullet & \bullet \\ & & & & & \bullet \end{bmatrix}$$

- Q is zero below the first subdiagonal ($Q_{ij} = 0$ if $i > j + 1$)

  column k is column k of A orthogonalized with respect to previous columns

- R is zero above second superdiagonal ($R_{ij} = 0$ if $j > i + 2$)

  follows from considering $R = Q^T A$ and the property of Q

Computing tridiagonal QR factorization

QR factorization of $n \times n$ tridiagonal A takes order n operations

$$Q^T A = R$$

for example, in the Householder algorithm (133A lecture 6)

- Q^T is a product of reflectors $H_k = I - v_k v_k^T$ that make A upper triangular

$$H_{n-1} \cdots H_1 \begin{bmatrix} A_{11} & A_{12} & 0 & \cdots & 0 & 0 \\ A_{21} & A_{22} & A_{23} & \cdots & 0 & 0 \\ 0 & A_{32} & A_{33} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & A_{n-1,n-1} & A_{n-1,n} \\ 0 & 0 & 0 & \cdots & A_{n,n-1} & A_{nn} \end{bmatrix} = R$$

  if A is tridiagonal, each vector v_k has only two nonzero elements

- Q is stored in factored form (as the reflectors v_k)
- we can allow zeros on diagonal of R, to extend QR factorization to singular A

Outline

- tridiagonal symmetric matrices
- **basic QR algorithm**
- QR algorithm with shifts

QR algorithm

suppose A is a symmetric $n \times n$ matrix

Basic QR iteration: start at $A_1 = A$ and repeat for $k = 1, 2, \ldots,$

- compute QR factorization $A_k = Q_k R_k$
- define $A_{k+1} = R_k Q_k$

for most matrices,

- A_k converges to a diagonal matrix of eigenvalues of A
- $U_k = Q_1 Q_2 \cdots Q_k$ converges to matrix of eigenvectors

Some immediate properties

$$A_1 = A, \qquad A_k = Q_k R_k, \quad A_{k+1} = R_k Q_k \quad (k \geq 1)$$

- the matrices A_k are symmetric: the first matrix $A_1 = A$ is symmetric and

$$A_{k+1} = R_k Q_k = Q_k^T A_k Q_k$$

- continuing recursively, we see that an orthogonal similarity relates A_k and A:

$$\begin{aligned} A_{k+1} &= (Q_1 Q_2 \cdots Q_k)^T A (Q_1 Q_2 \cdots Q_k) \\ &= U_k^T A U_k \end{aligned}$$

  therefore the matrices A_k all have the same eigenvalues as A

- the orthogonal matrices $U_k = Q_1 Q_2 \cdots Q_k$ and R_k satisfy

$$A U_{k-1} = U_{k-1} A_k = U_{k-1} Q_k R_k = U_k R_k$$

Equivalent form

a related algorithm ("simultaneous iteration") uses the last property to generate U_k:

$$AU_{k-1} = U_k R_k$$

we note that the right-hand side is a QR factorization

Simultaneous iteration: start at $U_0 = I$ and repeat for $k = 1, 2, \ldots,$

- multiply with A: compute $V_k = AU_{k-1}$
- compute QR factorization $V_k = U_k R_k$

if the matrices U_k converge to U, then R_k converges to a diagonal matrix, since

$$R_k = U_k^T V_k = U_k^T A U_{k-1}$$

so the limit of R_k is both symmetric ($U^T AU$) and triangular, hence diagonal

Interpretation

simultaneous iteration is a matrix extension of the *power iteration*

Power iteration: start at n-vector u_0 with $\|u_0\| = 1$, and repeat for $k = 1, 2, \ldots,$

- multiply with A: compute $v_k = Au_{k-1}$
- normalize: $u_k = v_k / \|v_k\|$

this is a simple iteration for computing an eigenvector with the largest eigenvalue

- suppose the eigenvalues of A satisfy $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$
- suppose $u_0 = \alpha_1 q_1 + \cdots + \alpha_n q_n$ where q_i is a normalized eigenvector for λ_i
- after k power iterations, u_k is the normalized vector

$$A^k u_0 = \lambda_1^k (\alpha_1 q_1 + \alpha_2 (\lambda_2/\lambda_1)^k q_2 + \cdots + \alpha_n (\lambda_n/\lambda_1)^k q_n)$$

- if $\alpha_1 \neq 0$, the vector $\pm u_k$ converges to q_1, and $u_k^T A u_k$ converges to λ_1

QR iteration with tridiagonal A

now suppose A in the basic QR iteration on page 9.10 is tridiagonal and symmetric

- we already noted that the matrices A_k are symmetric if A is symmetric (p. 9.11):

$$A_{k+1} = R_k Q_k = Q_k^T A_k Q_k$$

- Q-factor of a tridiagonal matrix is zero below the first subdiagonal (page 9.8)
- this implies that the product $A_{k+1} = R_k Q_k$ is zero below the first subdiagonal:

$$\begin{bmatrix} \bullet & \bullet & \bullet & & & \\ & \bullet & \bullet & \bullet & & \\ & & \bullet & \bullet & \bullet & \\ & & & \bullet & \bullet & \bullet \\ & & & & \bullet & \bullet \\ & & & & & \bullet \end{bmatrix} \begin{bmatrix} \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ & \bullet & \bullet & \bullet & \bullet & \bullet \\ & & \bullet & \bullet & \bullet & \bullet \\ & & & \bullet & \bullet & \bullet \\ & & & & \bullet & \bullet \end{bmatrix} = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ & \bullet & \bullet & \bullet & \bullet & \bullet \\ & & \bullet & \bullet & \bullet & \bullet \\ & & & \bullet & \bullet & \bullet \\ & & & & \bullet & \bullet \end{bmatrix}$$

- since A_{k+1} is also symmetric, it is tridiagonal

hence, for tridiagonal symmetric A, the complexity of one QR iteration is order n

Outline

- tridiagonal symmetric matrices
- basic QR algorithm
- **QR algorithm with shifts**

QR algorithm with shifts

in practice, a multiple of the identity is subtracted from A_k before factoring

QR iteration with shifts: start at $A_1 = A$ and repeat for $k = 1, 2, \ldots,$

- choose a shift μ_k
- compute QR factorization $A_k - \mu_k I = Q_k R_k$
- define $A_{k+1} = R_k Q_k + \mu_k I$

- iteration still preserves symmetry and tridiagonal structure in A_k
- with properly chosen shifts, the iteration always converges
- with properly chosen shifts, convergence is fast (usually cubic)

Complexity

overall complexity of QR method for symmetric eigendecomposition $A = Q\Lambda Q^T$

Eigenvalues: if eigenvectors are not needed, we can leave Q in factored form

- reduction of A to tridiagonal form costs $(4/3)n^3$
- for tridiagonal matrix, complexity of one QR iteration is linear in n
- on average, number of QR iterations is a small multiple of n

hence, cost is dominated by $(4/3)n^3$ for initial reduction to tridiagonal form

Eigenvalues and eigenvectors: if Q is needed, order n^3 terms are added

- reduction to tridiagonal form and accumulating orthogonal matrix costs $(8/3)n^3$
- finding eigenvalues and eigenvectors of tridiagonal matrix costs $6n^3$

hence, total cost is $(26/3)n^3$ plus lower order terms

References

- Lloyd N. Trefethen and David Bau, III, *Numerical Linear Algebra* (1997).

  lectures 26–29 in this book discuss the QR iteration

- James W. Demmel, *Applied Numerical Linear Algebra* (1997).

  page 213 of this book gives details for the complexity figures on page 9.16