QR-method lecture 4

SF2524 - Matrix Computations for Large-scale Systems

Outline:

- Basic QR-method
- Improvement 1: Two-phase approach
  - Hessenberg reduction
  - Hessenberg QR-method
- **Improvement 2: Acceleration with shifts**
- Convergence theory

Improvement 2: Acceleration with shifts (Section 2.3)

Shifted QR-method

One step of shifted QR-method: Let $H_k = H$

$$\begin{aligned} H - \mu I &= QR \\ \bar{H} &= RQ + \mu I \end{aligned}$$

and $H_{k+1} := \bar{H}$.

Note:

$$H_{k+1} = \bar{H} = RQ + \mu I = Q^T(H - \mu I))Q + \mu I = Q^T H_k Q$$

$\Rightarrow$ One step of shifted QR-method is a similarity transformation, with a different Q matrix.

* matlab demo: `qr_shifted.m` *

Idealized situation: Let $\mu = \lambda(H)$

Suppose μ is an eigenvalue:
$\Rightarrow$ $H - \mu I$ is a singular Hessenberg matrix.

QR-factorization of singular Hessenberg matrices (Lemma 2.3.1)

The R-matrix in the QR-decomposition of a singular unreduced Hessenberg matrix has the structure

$$R = \begin{bmatrix} \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & \times & \times \\ & & & & 0 \end{bmatrix}.$$

* Matlab demo: Show QR-factorization of singular Hessenberg matrix in matlab *

Shifted QR for exact shift: $\mu = \lambda$

If $\mu = \lambda$ is an eigenvalue of H, then $H - \mu I$ is singular. Suppose Q, R a QR-factorization of a Hessenberg matrix and

$$R = \begin{bmatrix} \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & \times & \times \\ & & & & 0 \end{bmatrix}.$$

Then, * Prove on blackboard *

$$RQ = \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & & 0 \end{bmatrix}$$

and

$$\bar{H} = RQ + \lambda I = \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & & \lambda \end{bmatrix}.$$

$\Rightarrow \lambda$ is an eigenvalue of $\bar{H}$.

More precisely:

Lemma (Lemma 2.3.2)

Suppose λ is an eigenvalue of the Hessenberg matrix H. Let $\bar{H}$ be the result of one shifted QR-step. Then,

$$\begin{aligned} \bar{h}_{n,n-1} &= 0 \\ \bar{h}_{n,n} &= \lambda. \end{aligned}$$

Select the shift

How to select the shifts?

- Shifted QR-method with $\mu = \lambda$ computes an eigenvalue in one step.
- The exact eigenvalue not available. How to select the shift?

Rayleigh shifts

If we are close to convergence the diagonal element will be an approximate eigenvalue. Rayleigh shifts:

$$\mu := r_{m,m}.$$

Explanation of terminology

- The QR-method can be interpreted as equivalent to variant of Power Method applied to A. (Will be shown later)
- The QR-method can be interpreted as equivalent to variant of Power Method applied to A^{-1}. (Proof sketched in TB Chapter 29) $\Rightarrow$ Rayleigh shifts can be interpreted as Rayleigh quotient iteration.

One final trick: Deflation

Deflation: A technique to avoid computing for already converged eigenvalues

QR-step on reduced Hessenberg matrix

Suppose

$$H = \begin{pmatrix} H_0 & H_1 \\ 0 & H_3 \end{pmatrix},$$

where H_3 is upper triangularand let

$$\bar{H} = \begin{pmatrix} \bar{H}_0 & \bar{H}_1 \\ \bar{H}_2 & \bar{H}_3 \end{pmatrix},$$

be the result of one (shifted) QR-step. Then, $\bar{H}_2 = 0$, $\bar{H}_3 = H_3$ and $\bar{H}_0$ is the result of one (shifted) QR-step applied to H_0.

* show proof *

$\Rightarrow$ We can reduce the active matrix when an eigenvalue is converged.

This is called deflation.

Rayleigh shifts can be combined with deflation $\Rightarrow$

Algorithm 4 Hessenberg QR algorithm with Rayleigh quotient shift and deflation

Input: A Hessenberg matrix $A \in \mathbb{C}^{n\times n}$

Set $H^{(0)} := A$
for $m = n, \dots, 2$ **do**
  $k = 0$
  repeat
    $k = k + 1$
    $\sigma_k = h_{m,m}^{(k-1)}$
    $H_{k-1} - \sigma_k I =: Q_k R_k$
    $H_k := R_k Q_k + \sigma_k I$
  until $|h_{m,m-1}^{(k)}|$ is sufficiently small
  Save $h_{m,m}^{(k)}$ as a converged eigenvalue
  Set $H^{(0)} = H_{1:(m-1),1:(m-1)}^{(k)} \in \mathbb{C}^{(m-1)\times(m-1)}$
end for

* show Hessenberg qr with shifts in matlab *
Not proven: Hessenberg QR with givens can be combined with shifts
* `http://www.youtube.com/watch?v=qmgxzsWWsNc` *

Outline:

- Basic QR-method
- Improvement 1: Two-phase approach
  - Hessenberg reduction
  - Hessenberg QR-method
- Improvement 2: Acceleration with shifts
- **Convergence theory**

Outline:

- Basic QR-method
- Improvement 1: Two-phase approach
  - Hessenberg reduction
  - Hessenberg QR-method
- Improvement 2: Acceleration with shifts
- **Convergence theory**

Convergence theory - (Lecture notes PDF + TB Ch. 28)

Didactic simplification for convergence of QR-method: Assume $A = A^T$.

Convergence characterization

(1) Artificial algorithm: USI - Unnormalized Simultaneous Iteration
(2) Show convergence properties of USI
(3) Artificial algorithm: NSI - Normalized Simultaneous Iteration
(4) Show: USI $\Leftrightarrow$ NSI $\Leftrightarrow$ QR-method

Definition: Unnormalized simultaneous iteration (USI)

A generalization of power method with m vectors "simultaneously"

$$V^{(0)} = [v_1^{(0)}, \ldots, v_m^{(0)}] \in \mathbb{R}^{n\times m}.$$

Define

$$V^{(k)} := A^k V^{(0)}.$$

A QR-factorization generalizes the normalization step:

$$\hat{Q}^{(k)} \hat{R}^{(k)} = V^{(k)}.$$

* show in matlab usi *

Convergence of USI

Assumptions:

- **(ASM1)** Let eigenvalues ordered and distinct in modulus

$$|\lambda_1| > \cdots > |\lambda_n|.$$

- **(ASM2)** Assume leading principal submatrices of $X^T V^{(0)}$ are nonsingular, where X are the eigenvectors.

Theorem (PDF Lecture notes Thm 2.4.1 (essentially TB Thm 28.1))

Suppose ASM1 is satisfied for $A \in \mathbb{R}^{n\times n}$. Let the columns of X be eigenvectors of A Let $V^{(0)} \in \mathbb{R}^{n\times n}$ be ASM2 is satisfied Let $V^{(k)} := AV^{(k)}$, be the iterates of USI. Then, a QR-factorizations of $V^{(k)} = Q^{(k)}R^{(k)}$ satisfies

$$\|Q^{(k)} - X\| = \mathcal{O}(C^k)$$

where $C = \max_{\ell=1,\dots,n-1} |\lambda_\ell|/|\lambda_{\ell+1}|$.

* Show proof + matlab demo on USI *

Normalized Simultaneous Iteration (NSI)

Variants of the power method. Equivalent:

(i) $v_k = \frac{A^k v_0}{\|A^k v_0\|}$

(ii) $v_k = \frac{A v_{k-1}}{\|A v_{k-1}\|}$

USI is a generlization of (i).

NSI is a generalization of (ii).

Algorithm: (Normalized) Simultaneous Iteration

- Input $\hat{Q}^{(0)} \in \mathbb{R}^{n \times m}$
- For $k = 1, \ldots,$
  - Set $Z = A\hat{Q}^{k-1}$
  - Compute QR-factorization $\hat{Q}^{(k)}\hat{R}^{(k)} = Z$

USI and NSI are equivalent. More precisely:

Equivalence USI and NSI (TB Thm 28.2)

Suppose assumptions above are satisfied. If USI and NSI are started with the same vector they will generate the same sequence of matrices $\hat{Q}^k$ and $\hat{R}^k$.

* show usi_nsi_equiv.m *

Simultaneous iteration and QR-method

We will establish:

basic QR-method $\Leftrightarrow$ Simultaneous iteration with $\hat{Q}^{(0)} = I \in \mathbb{R}^{n\times n}$.

Simultaneous iteration satisfies

- $\underline{Q}^{(0)} = I$
- $Z_k = A\underline{Q}^{(k-1)}$
- $Z_k = \underline{Q}^{(k)} R^{(k)}$
- $A^{(k)} := (\underline{Q}^{(k)})^T A(\underline{Q}^{(k)})$

QR-method satisfies

- $A^{(0)} = A$
- $A^{(k-1)} = Q^{(k)} R^{(k)}$
- $A^{(k)} = R^{(k)} Q^{(k)}$
- $\underline{Q}^{(k)} := Q^{(1)} \cdots Q^{(k)}$

Define: $\underline{R}^{(k)} := R^{(k)} \cdots R^{(1)}$

Essentially: The above equations generate the same sequence of matrices

More precisely ...

TB Theorem 28.3:

Theorem (Equivalence simultaneous iteration and QR-method)

The above processes generate identical sequences of vectors. In particular,

$$A^k = \underline{Q}^{(k)}\underline{R}^{(k)}$$

and

$$A^{(k)} = (\underline{Q}^{(k)})^T A(\underline{Q}^{(k)}).$$

Beware: QR-factorization is not unique and equivalence only holds with one QR-factorization.

Important property:

$$A^{(k)} = (\underline{Q}^{(k)})^T A \underline{Q}^{(k)}$$

Consequence

$$\begin{aligned}
A^{(k)} &= Q^{(k)\,T} A Q^{(k)} \\
&= (X(I+\Delta_k))^T A X(I+\Delta_k) \\
&= (X(I+\Delta_k))^T X\Lambda(I+\Delta_k) \\
&= (I+\Delta_k)^T X^T X\Lambda(I+\Delta_k) \\
&= (I+\Delta_k)^T \Lambda(I+\Delta_k) \\
&= \Lambda + \Delta_k^T \Lambda + \Lambda\Delta_k + \Delta_k^2 \\
&= \Lambda + O(\|\Delta_k\|)
\end{aligned}$$

Hence, $A^{(k)}$ will approach a diagonal matrix at speed C^k.

* Matlab demos *

qr_nsi$_{e}quiv.m$