Welcome to Ma221! Lecture 18, Fall 24

QR Iteration with Shift

$A_0 = A$

$i = 0$

repeat

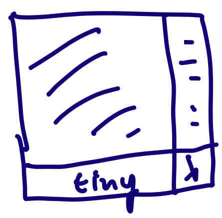
  choose σ_i near eval, $\sigma_i = A_i(n,n)$

  factor $A - \sigma_i I = Q_i R_i$

  $A_{i+1} = R_i Q_i + \sigma_i I \quad \ldots \quad = Q_i^T A_i Q_i$

  $i = i+1$

until convergence



Matlab demo, code on webpage { show why quadratic convergence

Making QR iteration practical

1) each iteration costs 1 QR decomp + 1 matmul $= O(n^3) \Rightarrow$ if just one iteration per eval $\Rightarrow O(n^4)$ cost, want $O(n^3)$

2) How to shift to converge to real Schur form?

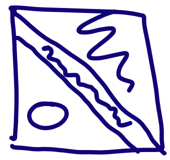
3) How to decide convergence?

4) How to minimize communication?

5) How to get down to cost of Strassen-like algs?

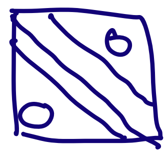
Answers:

(1) Preprocess $A = QHQ^T$ where Q orthog, H upper Hessenberg

$H =$ 

QR iteration on H keeps it upper Hess, lowers cost to $O(n^2)$

$\Rightarrow$ total cost reduced to $n \cdot O(n^2) = O(n^3)$

$A = A^T \Rightarrow H = Q^T A Q = H^T =$  tridiagonal

$\Rightarrow$ cost of one step $= O(n)$

(Chap5) + cubic convergence

(2) Converge to real Schur form:

pairs of complex evals of real A appear as $\lambda, \bar{\lambda}$

take one QR iteration starting with A_i, using λ

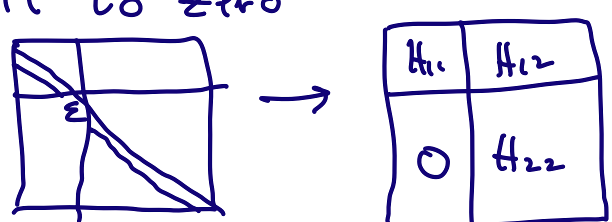
another " " with A_{i+1} using $\bar{\lambda}$

$\Rightarrow A_{i+2}$ real, don't bother computing imaginary parts, take 2 shifts at a time

(3) Detecting convergence:

If any $H(i+1,i)$ small enough, $O(\varepsilon) \cdot \|H\|$

set it to zero



splits problem into 2 smaller, independent subproblems

Eg H_{22} could be 2x2 with $\lambda, \bar{\lambda}$ evals
$\Rightarrow$ real Schur form

see posted papers by Srisvastara et al

(4) Reducing communication:
no known way to attain $O\left(\frac{n^3}{\sqrt{\text{cache size}}}\right)$
using deterministic alg,
need randomization, with divide + conquer
see other posted paper by Srivastava

(5) Strassen? same idea

More detail on Hessenberg QR

How to reduce $A = Q H Q^T$
$H = Q^T A Q$

Analogous to QR with Householder transforms:

$$\begin{bmatrix} I_3 & 0 \\ 0 & H^{2\times2} \end{bmatrix} \begin{bmatrix} I_2 & 0 \\ 0 & H^{3\times3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & H^{4\times4} \end{bmatrix} \begin{bmatrix} x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & H^{4\times4} \end{bmatrix}^T \begin{bmatrix} I_2 & 0 \\ 0 & H^{3\times3} \end{bmatrix}^T \begin{bmatrix} I_3 & 0 \\ 0 & H^{2\times2} \end{bmatrix}^T$$

upper Hessenberg

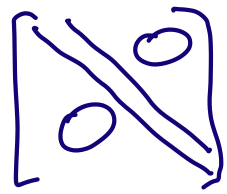
Cost: $\frac{10}{3} n^3 + O(n^2)$ just for H

or $\frac{14}{3} n^3 + O(n^2)$ for Q too

much more than QR decomp or LU, still not bottleneck

SVD similar: reduce to bidiagonal form

$$\begin{bmatrix} I_3 & 0 \\ 0 & H^{2\times 2}\end{bmatrix}\begin{bmatrix} I_2 & 0 \\ 0 & H^{3\times 3}\end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & H^{4\times 4}\end{bmatrix} H^{5\times 5} \begin{bmatrix} x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & H^{4\times 4}\end{bmatrix}\begin{bmatrix} I_2 & 0 \\ 0 & H^{3\times 3}\end{bmatrix}\begin{bmatrix} I_3 & 0 \\ 0 & H^{2\times 2}\end{bmatrix}$$



bidiagonal

compute SVD of this

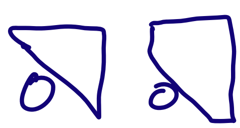
QR Iteration on upper Hessenberg matrix in $O(n^2)$ flops

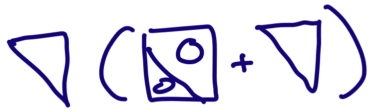
Lemma: upper Hess. preserved by QR iteration

pf: A upper Hess $\Rightarrow A - \sigma I$ upper Hess

$\Rightarrow A - \sigma I = QR$, Q upper Hess

col i Q = linear comb of cols $1:i$ of $A - \sigma I$

then $RQ =$  also upper Hess

$=$ 

How to do one step of QR iteration in $O(n^2)$ flops

Def: H upper Hess is unreduced if all $H(i+1,i) \neq 0$ (else split)

Implicit Q Theorem: suppose $Q^T A Q$ upper Hess and unreduced. Then columns 2 through n of Q are uniquely determined by col 1 (up to scaling by ± 1)

1 Step of QR in $O(n^2)$ flops

$A - \sigma I = QR$, first col is $\begin{bmatrix} A(1,1)-\sigma \\ A(2,1) \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

Let Q_1 be Givens rotation such that first col of $Q \Rightarrow Q_1^T \begin{bmatrix} A(1,1)-\sigma \\ A(2,1) \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}$

bulge

$$Q_4^T \left[Q_3^T \left[Q_2^T \left[Q_1^T \begin{bmatrix} x & x & x & x & x \\ x & x & x & x & x \\ + & x & x & x & x \\ & + & x & x & x \\ & & + & x & x \end{bmatrix} Q_1 \right. \right. \right. Q_2 \; Q_3 \; Q_4$$

Upper Hess! first col of $Q_1 Q_2 Q_3 Q_4$ same as Q_1

bulge chasing

Cost = cost of $2n$ Givens rotations $= O(n^2)$

Proof of Implicit Q Thm

Let q_i be col i of Q, use induction on i

$$Q^T A Q = H \Rightarrow AQ = QH$$

Col 1: $Aq_1 = H(1,1) \cdot q_1 + H(2,1) q_2$

$\Rightarrow$ determines $H(1,1)$, $H(2,1)$, q_2 via QR:

$$[q_1, Aq_1] = [q_1, q_2] \cdot \begin{bmatrix} 1 & H(1,1) \\ 0 & H(2,1) \end{bmatrix}$$

More generally:

suppose we have $q_1, q_2, \ldots q_i$

Get next column $(AQ)_i = (QH)_i$

$$q_j^T \left[Aq_i = \sum_{j=1}^{i+1} q_j H(j,i) \right]$$

$$q_j^T A q_i = H(j,i) \quad \text{for } j = 1, \ldots, i$$

$$Aq_i - \sum_{j=1}^{i} q_j H(j,i) = q_{i+1} \cdot H(i+1,i)$$

gives us q_{i+1} and $H(i+1,i)$

used in LAPACK xGEES (Schur)

or xGEEV (evals + eves)

eig(A) in Matlab

Chap 5: Symmetric Eigenproblem + SVD

Goals: Perturbation Theory
Algorithms (depend on pert. theory)

Real symmetric $A = A^T = Q \Lambda Q^T$, $QQ^T = I$
Complex Hermitian $A = A^H = Q \Lambda Q^H$, $QQ^H = I$

can reduce $A = A^H$ to real symm. tridiagonal

$\Lambda = \mathrm{diag}(\lambda_1, \dots, \lambda_n)$ $\lambda_1 \ge \dots \ge \lambda_n$

$Q = [q_1, \dots, q_n]$

Complex symmetric different

$\begin{bmatrix} 1 & i \\ i & -1 \end{bmatrix}$ $i = \sqrt{-1}$ has two evals $= 0$
" one evec

Most results will apply to SVD
(Thm 3.3 part 4)

$$B = \left[\begin{array}{c|c} 0 & A \\ \hline A^T & 0 \end{array}\right] = B^T$$

eigendecomp of B related to SVD of A
evals of B = $\pm$ singular values of A
(+ zeros if A rectangular)

evecs of B closely related to
sing. vecs of A

$\Rightarrow$ reuse algs for sym eig for SVD
(some open problems)

extends perturbation theory from sym B to SVD(A)
changing A to $A+E$ changes
B to $B + \left[\begin{array}{c|c} 0 & E \\ \hline E^T & 0 \end{array}\right]$

Def: Rayleigh Quotient $\rho(u, A) = \frac{u^T A u}{u^T u}$
$u \neq 0$

Properties: if $Au = \lambda u \Rightarrow \rho(u,A) = \lambda$

$$u = \sum_{i=1}^{n} b_i q_i = Qb \quad b = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$$

$$\rho(u,A) = \frac{(Qb)^T A (Qb)}{(Qb)^T (Qb)} = \frac{b^T \overbrace{Q^T A Q}^{\Lambda} b}{b^T \underbrace{Q^T Q}_{I} b}$$

$$= \frac{b^T \Lambda b}{b^T b}$$

$$= \sum_{i=1}^{n} \lambda_i b_i^2 \Big/ \sum_{i=1}^{n} b_i^2$$

= convex combination of all evals

$$\lambda_1 \geq \rho(u,A) \geq \lambda_n$$

$$\lambda_1 = \max_{u \neq 0} \rho(u,A) \quad , \quad \lambda_n = \min_{u \neq 0} \rho(u,A)$$

choose $u = q_1$ choose $u = q_n$

all evals expressible using $\rho(u, A)$

Courant-Fischer Minimax Thm

R^j = j-dimensional subspace of $\mathbb{R}^n$

S^{n-j+1} = n-j+1 " " " "

$$\max_{R^j} \min_{\substack{r \in R^j \\ r \neq 0}} \rho(r, A) = \lambda_j = \min_{S^{n-j+1}} \max_{\substack{s \in S^{n-j+1} \\ s \neq 0}} \rho(s, A)$$

max over R^j attained by span$(q_1, \dots, q_j)$

min over $r \in R^j$ " " $r = q_j$

min over S^{n-j+1} " " span$(q_j, \dots q_n)$

max over $s \in S^{n-j+1}$ " " $s = q_j$

Proof! Given any R^j and S^{n-j+1}

their dimensions add up to

$j + n - j + 1 = n + 1$

$\Rightarrow R^j$ and S^{n-j+1} intersect in $x_{RS} \neq 0$

$$\min_{\substack{r \in R^j \\ r \neq 0}} \rho(r, A) \leq \rho(x_{RS}, A) \leq \max_{\substack{s \in S^{n-j+1} \\ s \neq 0}} \rho(s, A)$$

Let R' maximize $\min_{\substack{r \in R' \\ r \neq 0}} \rho(r, A)$

$\dim R' = j$

Let S' minimize $\max_{\substack{s \in S' \\ s \neq 0}} \rho(s, A)$

$\dim S' = n - j + 1$

$$\lambda_j \leq \max_{R^j} \min_{\substack{r \in R^j \\ r \neq 0}} \rho(r, A) = \min_{\substack{r \in R' \\ r \neq 0}} \rho(r, A) \leq \rho(x_{R'S'})$$

$$\leq \max_{\substack{s \in S' \\ s \neq 0}} \rho(s, A) = \min_{S^{n-j+1}} \max_{\substack{s \in S^{n-j+1} \\ s \neq 0}} \rho(s, A) \leq \lambda_j$$

If we choose $R^j = \mathrm{span}(q_1, \ldots, q_j)$

$$r = q_j \Rightarrow \min_{r \in R^j} \rho(r, A) = \rho(q_j, A) = \lambda_j$$

If we choose $S^{n-j+1} = \mathrm{span}(q_j, \ldots q_n)$

and $s = q_j$

$$\max_{\substack{s \in S^{n-j+1} \\ s \neq 0}} \rho(s, A) = \rho(q_j, A) = \lambda_j$$

all inequalities are equalities to λ_j QED