# CS 450: Numerical Anlaysis[1]

## Eigenvalue Problems

University of Illinois at Urbana-Champaign

[1] *These slides have been drafted by Edgar Solomonik as lecture templates and supplementary material for the book "Scientific Computing: An Introductory Survey" by Michael T. Heath (slides).*

# Eigenvalues and Eigenvectors

- A matrix $\boldsymbol{A}$ has eigenvector-eigenvalue pair (eigenpair) $(\lambda, \boldsymbol{x})$ if

- Each $n \times n$ matrix has up to $n$ eigenvalues, which are either real or complex

# Eigenvalue Decomposition

- If a matrix $\boldsymbol{A}$ is diagonalizable, it has an *eigenvalue decomposition*

- $\boldsymbol{A}$ and $\boldsymbol{B}$ are *similar*, if there exist $\boldsymbol{Z}$ such that $\boldsymbol{A} = \boldsymbol{Z}\boldsymbol{B}\boldsymbol{Z}^{-1}$

# Similarity of Matrices

| *matrix* | *similarity* | *reduced form* |
|---|---|---|
| SPD | | |
| real symmetric | | |
| Hermitian | | |
| normal | | |
| real | | |
| diagonalizable | | |
| arbitrary | | |

# Canonical Forms

- Any matrix is *similar* to a bidiagonal matrix, giving its *Jordan form*:

- Any diagonalizable matrix is *unitarily similar* to a triangular matrix, giving its *Schur form*:

# Eigenvectors from Schur Form

- Given the eigenvectors of one matrix, we seek those of a similar matrix:

- Its easy to obtain eigenvectors of triangular matrix $\boldsymbol{T}$:

# Rayleigh Quotient

- For any vector $\boldsymbol{x}$, the *Rayleigh quotient* provides an estimate for some eigenvalue of $\boldsymbol{A}$:

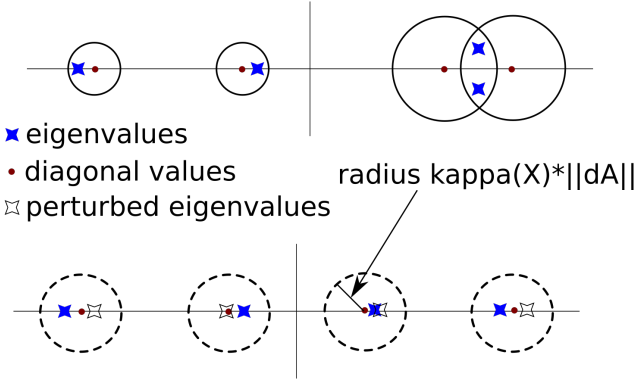
# Perturbation Analysis of Eigenvalue Problems

- Suppose we seek eigenvalues $\boldsymbol{D} = \boldsymbol{X}^{-1}\boldsymbol{A}\boldsymbol{X}$, but find those of a slightly perturbed matrix $\boldsymbol{D} + \boldsymbol{\delta D} = \hat{\boldsymbol{X}}^{-1}(\boldsymbol{A} + \boldsymbol{\delta A})\hat{\boldsymbol{X}}$:

- Gershgorin's theorem allows us to bound the effect of the perturbation on the eigenvalues of a (diagonal) matrix:
  *Given a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, let $r_i = \sum_{j\neq i} |a_{ij}|$, define the Gershgorin disks as*

$$D_i = \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i\}.$$

# Gershgorin Theorem Perturbation Visualization




- Top corresponds to Gershgorin disks on complex plane of 4-by-4 real matrix.
- Bottom part corresponds to bounds on Gershgorin disks of $\boldsymbol{X}^{-1}(\boldsymbol{A}+\boldsymbol{\delta A})\boldsymbol{X}$, which contain the eigenvalues $\boldsymbol{D}$ of $\boldsymbol{A}$ and the perturbed eigenvalues $\boldsymbol{D}+\boldsymbol{\delta D}$ of $\boldsymbol{A}+\boldsymbol{\delta A}$ provided that $||\boldsymbol{\delta A}||$ is sufficiently small.

# Conditioning of Particular Eigenpairs

- Consider the effect of a matrix perturbation on an eigenvalue $\lambda$ associated with a right eigenvector $\boldsymbol{x}$ and a left eigenvector $\boldsymbol{y}$, $\lambda = \boldsymbol{y}^H \boldsymbol{A}\boldsymbol{x} / \boldsymbol{y}^H \boldsymbol{x}$

- A more accurate eigenvalue approximation than Rayleigh quotient for a normalized perturbed eigenvector (e.g., iterative guess) $\hat{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{\delta x}$, can be obtained with an estimate of both eigenvectors (also $\hat{\boldsymbol{y}} = \boldsymbol{y} + \boldsymbol{\delta y}$),

# Power Iteration

- *Power iteration* can be used to compute the largest eigenvalue of a real symmetric matrix $\boldsymbol{A}$:

- The error of power iteration decreases at each step by the ratio of the largest eigenvalues:

# Inverse and Rayleigh Quotient Iteration

***Activity:*** *Inverse Iteration with a Shift*
***Activity:*** *Rayleigh Quotient Iteration*

- *Inverse iteration* uses LU/QR/SVD of $\boldsymbol{A}$ to run power iteration on $\boldsymbol{A}^{-1}$

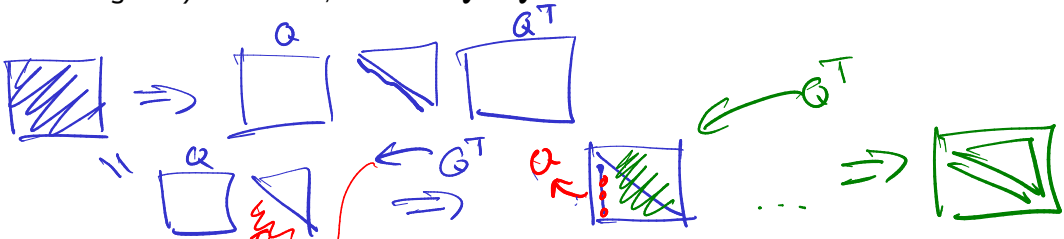
- *Rayleigh quotient iteration* provides rapid convergence to an eigenpair

# Deflation

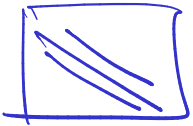
- Power, inverse, and Rayleigh-quotient iteration compute a single eigenpair, to obtain further eigenpairs, can perform *deflation*

# Direct Matrix Reductions

- We can always compute an orthogonal similarity transformation to reduce a general matrix to *upper-Hessenberg* (upper-triangular plus the first subdiagonal) matrix $\boldsymbol{H}$, i.e. $\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{H}\boldsymbol{Q}^T$:



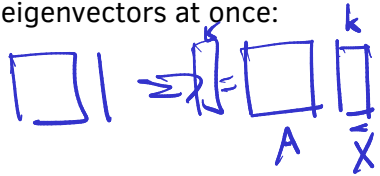

- In the symmetric case, Hessenberg form implies tridiagonal:



# Simultaneous and Orthogonal Iteration

***Demo:*** *Orthogonal Iteration*
***Activity:*** *Orthogonal Iteration*

- *Simultaneous iteration* provides the main idea for computing many eigenvectors at once:




$$X^{(i+1)} = A X^{(i)}$$

$$\lim_{i \to \infty} \operatorname{span}(X^{(i)}) = \operatorname{span}(\overbrace{\{q_1 \cdots q_k\}}^{\text{leading } k \text{ evecs of } A})$$

- Orthogonal iteration performs QR at each step to ensure stability

$O(nk^2)$ $X^{(i)} = Q^{(i)} R^{(i)}$

$O(n^2k)$ $X^{(i+1)} = \underbrace{A}_{A^TA} Q^{(i)}$

$\operatorname{span}(X^{(i)}) = \operatorname{span}(Q^{(i)})$

Ortho-iter$(\underbrace{A^TA}_{})$ = "Randomized SVD"

evecs of $A^TA$ are singular vectors of $A$

# QR Iteration

- QR iteration reformulates orthogonal iteration for $n = k$ to reduce cost/step,

Orth-iter computes

$\hat{Q}_{i+1}\hat{R}_{i+1} = A\hat{Q}_i$

$A_{i+1} = R_i Q_i$

$Q_i R_i = A_i$

$R_i = Q_i^T A_i$

$A_{i+1} = R_i Q_i = Q_i^T A_i Q_i$

- Using induction, we assume $\boldsymbol{A}_i = \hat{Q}_i^T \boldsymbol{A} \hat{Q}_i$ and show that QR iteration obtains $\boldsymbol{A}_{i+1} = \hat{Q}_{i+1}^T \boldsymbol{A} \hat{Q}_{i+1}$

$\hat{Q}_i^T = Q_i^T \cdots Q_1^T$

$\hat{Q}_i^T A \hat{Q}_i = A_i$

inductive assumption

$Q_i R_i = A_i$

$\hat{Q}_{i+1}\hat{R}_{i+1} = A\hat{Q}_i = \hat{Q}_i A_i = \hat{Q}_i Q_i R_i$

$R_i = Q_i^T \hat{Q}_i^T A \hat{Q}_i$

$A_{i+1} = R_i Q_i = \quad Q_i$

QR iteration converges to Schur form

$$A = Q^T \triangledown Q$$

$$\hat{Q}_i = Q$$

$$A \hat{Q}_i = \underbrace{Q^T \triangledown}_{Q \quad R}$$

# QR Iteration with Shift

- QR iteration can be accelerated using shifting:

$$Q_i R_i = A_i - \sigma_i I$$

$$A_{i+1} = R_i Q_i + \sigma_i I$$

$A_{i+1}$ is similar to $A_i$,

$$R_i Q_i = Q_i^T (A_i - \sigma_i I) Q_i$$

$$Q_i (A_{i+1} - \sigma_i I) Q_i^T = Q_i R_i$$

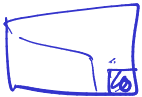
$Q^T Q$

$Q Q^T$

- The shift is typically selected to accelerate convergence with respect to a particular eigenvalue:

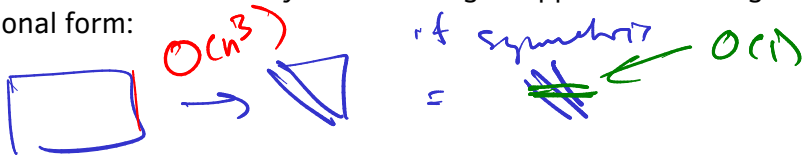
QR iteration on $A$ is also performing inverse iteration on $A_i$ at each step

$$A_i - \sigma_i I$$

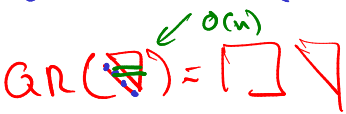
$A_i =$ 

# QR Iteration Complexity

- QR iteration is accelerated by first reducing to upper-Hessenberg or tridiagonal form:




QR-iteration




need $n-1$ Givens rotations instead of $n^2/2$

each Givens rotation acts on 2 rows of the matrix

QR iteration cost / iteration is $O(n^2)$ for nonsym

$O(n)$ for sym

# Solving Tridiagonal Symmetric Eigenproblems

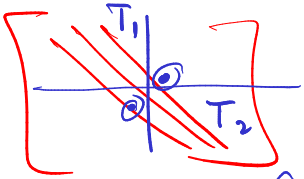
A variety of methods exists for the tridiagonal eigenproblem:

- QR iteration

  $O(1)$ QR factorizations / eigenvalue $\Rightarrow$ $O(n^2)$ cost to get eigenvalues

  $O(n^3)$ eigenvectors

- Divide and conquer




$\Rightarrow$ nonlinear equation to combine eigendecompositions of $T_1$ and $T_2$

MRRR eigvecs in $O(n^2)$ cost

# Solving the Secular Equation for Divide and Conquer

To solve the eigenproblem at each step, the divide and conquer method needs to diagonalize a rank-1 perturbation of a diagonal matrix

$$\boldsymbol{A} = \boldsymbol{D} + \alpha \boldsymbol{u}\boldsymbol{u}^T.$$

# Introduction to Krylov Subspace Methods

- *Krylov subspace methods* work with information contained in the $n \times k$ matrix

starting vector

$$\boldsymbol{K}_k = \underbrace{[\boldsymbol{x}_0 \quad \boldsymbol{A}\boldsymbol{x}_0 \quad \cdots \quad \boldsymbol{A}^{k-1}\boldsymbol{x}_0]}_{\text{extract maximal amount of info from these products}}$$

note

(Assume $K_n$ is invertible)

- The matrix $\boldsymbol{K}_n^{-1}\boldsymbol{A}\boldsymbol{K}_n$ is a *companion matrix* $\boldsymbol{C}$:

$$k_n^{(i)} = A^{i-1} x_0$$

$$AK_n = [Ak_n^{(1)} \cdots Ak_n^{(n)}] = [k_n^{(2)} \cdots k_n^{(n)} \; Ak_n^{(n)}]$$

$$K_n^{-1} A K_n = \begin{array}{c} k_n^{-1} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \\ 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \end{bmatrix} \end{array}$$

$\longleftarrow$ companion matrix

# Krylov Subspaces

$\kappa(K_k)$ is bad for large $k$ $\quad p(A) = X\, p(D)\, X^{-1}$

- Given $\boldsymbol{Q}_k \boldsymbol{R}_k = \boldsymbol{K}_k$, we obtain an orthonormal basis for the Krylov subspace,

span$(Q_k)$ $\quad (A + 2A^2 + 3I + \frac{1}{2}A^3)$

$$\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{x}_0) = span(\boldsymbol{Q}_k) = \{p(\boldsymbol{A})\boldsymbol{x}_0 : deg(p) < k\},$$

  where $p$ is any polynomial of degree less than $k$.

- The Krylov subspace includes the $k-1$ approximate dominant eigenvectors generated by $k-1$ steps of power iteration:

$$\max_{x \in span(Q_k)} \rho_A(x) \geq \rho_A(A^{k-1}x)$$

# Krylov Subspace Methods


- The $k \times k$ matrix $\boldsymbol{H}_k = \boldsymbol{Q}_k^T \boldsymbol{A} \boldsymbol{Q}_k$ minimizes $||\boldsymbol{A}\boldsymbol{Q}_k - \boldsymbol{Q}_k \boldsymbol{H}_k||_2$:

$$Q_k \overset{?}{X} \cong A Q_k$$

$$\cancel{Q_k^+ Q_k} X = \underbrace{Q_k^+ A Q_k}_{H_k}$$

- $\boldsymbol{H}_k$ is Hessenberg, because the companion matrix $\boldsymbol{C}_k$ is Hessenberg:

$$Q_n R_n = C_n$$


$$K_n^{-1} A U_n = C_n$$

$$R_n^{-1} Q_n^+ A Q_n R = C_n$$

$$H_n = Q_n^+ A Q_n = R_n C_n R_n^{-1} =$$

# Rayleigh-Ritz Procedure



- The eigenvalues/eigenvectors of $\boldsymbol{H}_k$ are the *Ritz values/vectors*:

will approximate those of $A$

$H_k x = \lambda x$

$A(Q_k x) \approx \lambda Q_k x$

- The Ritz vectors and values are the *ideal approximations* of the actual eigenvalues and eigenvectors based on only $\boldsymbol{H}_k$ and $\boldsymbol{Q}_k$:

$$\max_{\underbrace{x \in \mathrm{span}(Q_k)}_{x = Q_k y}} \rho_A(x) = \max_{\dots} \frac{x^T A x}{x^T x} = \max_{\substack{y \in \mathbb{R}^k \\ y \neq 0}} \frac{y^T \overbrace{Q_k^T A Q_k}^{H_k} y}{y^T y}$$

# Arnoldi Iteration

***Demo:*** *Arnoldi Iteration*
***Demo:*** *Arnoldi Iteration with Complex Eigenvalues*

- Arnoldi iteration computes $\boldsymbol{H} = \boldsymbol{H}_n$ directly using the recurrence $\boldsymbol{q}_i^T \boldsymbol{A} \boldsymbol{q}_j = h_{ij}$, where $\boldsymbol{q}_l$ is the $l$th column of $\boldsymbol{Q}_n$:

- After each matrix-vector product, orthogonalization is done with respect to each previous vector:

$u_j = A q_j$, $h_{ij} = q_i^T u_j$ for each $i < j$

$q_{j+1} = \left( u_j - \sum_{j<i} h_{ij} q_i \right)$

normalized

# Lanczos Iteration

- Lanczos iteration provides a method to reduce a symmetric matrix to a tridiagonal matrix:

$H_k = Q_k^T A Q_k$ is upper Hessenberg

is symmetric if $A$ is symmetric

$\hookrightarrow$ tridiagonal

- After each matrix-vector product, it suffices to orthogonalize with respect to two previous vectors:

need to compute only $h_{j-1,j} = h_{j,j-1} = q_{j-1}^T u_j$

# Cost Krylov Subspace Methods

- The cost of matrix-vector multiplication when the matrix has $m$ nonzeros

$$\begin{bmatrix} 4 & & & \\ & 0 & 3 & 0 \\ & 2 & 0 & 6 \end{bmatrix}$$

$v_i = \sum_j a_{ij} u_j$

$v_i = 0 \; \forall i$

for each nonzero $a_{ij}$ in $A$

$v_i \mathrel{+}= a_{ij} u_j$

2m cost in total

- The cost of orthogonalization at the $k$th iteration of a Krylov subspace method is

| per iteration cost | over all iterations ($k$) |
|---|---|
| for Arnoldi $O(nk)$ | $O(nk^2)$ |
| Lanczos $O(n)$ | $O(nk)$ |

$nk \le m$

# Restarting Krylov Subspace Methods

- In finite precision, Lanczos generally loses orthogonality, while orthogonalization in Arnoldi can become prohibitively expensive:

- Consequently, in practice, low-dimensional Krylov subspace methods are constructed repeatedly using carefully selected new starting vectors:

# Generalized Eigenvalue Problem

- A generalized eigenvalue problem has the form $\boldsymbol{A}\boldsymbol{x} = \lambda \boldsymbol{B}\boldsymbol{x}$,

- When $\boldsymbol{A}$ and $\boldsymbol{B}$ are symmetric and $\boldsymbol{B}$ is SPD, we can perform Cholesky on $\boldsymbol{B}$, multiply $\boldsymbol{A}$ by the inverted factors, and diagonalize it:

- Alternative canonical forms and methods exist that are specialized to the generalized eigenproblem.