Welcome to Ma221! Lecture 26, Oct 23

Eigenvalue algorithms:

Power method

→ Inverse iteration

→ Othogonal Iteration

→ QR Iteration

Power Method

$i=0$, given x_0 (random ok)

repeat

$y_{i+1} = A \cdot x_i$

$x_{i+1} = y_{i+1} / \|y_{i+1}\|_2$ ... approx evec

$\lambda_{i+1}' = x_{i+1}^T A \cdot x_{i+1}$ ... approx eval

$i = i+1$

until convergence

Convergence:

$A = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$

$|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$

$$x_i = A^i x_0 / \|A^i x_0\|_2$$

$$= [\lambda_1^i x_0(1), \lambda_2^i x_0(2) \ldots \lambda_n^i x_0(n)]^T / \| " \|_2$$

$$= \lambda_1^i \left[x_0(1), \left(\frac{\lambda_2}{\lambda_1}\right)^i x_0(2), \ldots, \left(\frac{\lambda_n}{\lambda_1}\right)^i x_0(n)\right]^T / \| " \|_2$$

$$= \Big[x_0(1), \underbrace{\left(\frac{\lambda_2}{\lambda_1}\right)}_{<1}^i x_0(2), \ldots\Big] / \| " \|_2$$

so $\left(\frac{\lambda_2}{\lambda_1}\right)^i \to 0$

$\rightarrow [1, 0 \ldots\ldots 0]$ if $x_0(1) \neq 0$

rate of convergence is $\left|\frac{\lambda_2}{\lambda_1}\right|^i$

Suppose A diagonalizable: $A = S \Lambda S^{-1}$

$A^i = (S\Lambda S^{-1})^i = S \Lambda^i S^{-1} = S \operatorname{diag}(\lambda_1^i, \ldots, \lambda_n^i) S^{-1}$

$A^i x_0 = S \Lambda^i S^{-1} x_0$

$= S \Lambda^i z \qquad z = S^{-1} x_0$

$= S [\lambda_1^i z(1), \lambda_2^i z(2), \ldots, \lambda_n^i z(n)]^T$

$= \lambda_1^i S [z(1), \underbrace{\left(\frac{\lambda_2}{\lambda_1}\right)}_{<1}^i z(2), \ldots]^T$

$\rightarrow \lambda_1^i S [z(1), 0, \text{———} 0]$

$= \lambda_1^i S[:,1] \cdot z(1)$

= first column of S (times a constant)

= evec for λ_1

To converge at a good rate, need

(1) $\left|\frac{\lambda_2}{\lambda_1}\right| < 1$, smaller is faster

can't count on this, eg A orthog

(2) z_1 nonzero, ok if x_0 chosen randomly

Inverse Iteration:

fix case $|\lambda_1| \approx |\lambda_2|$, or want evec for any λ_i

power method on $B = (A - \sigma I)^{-1}$

σ called "shift"

largest eval of $B = \dfrac{1}{(\text{closest eval of } A \text{ to } \sigma) - \sigma}$

$i = 0$, x_0 given

repeat

$\quad y_{i+1} = (A - \sigma I)^{-1} x_i$

$\quad x_{i+1} = y_{i+1} / \|y_{i+1}\|_2$

$\quad \lambda'_{i+1} = x_{i+1}^T A x_{i+1}$

$\quad i = i+1$

until convergence

evecs of B same as A

evals of B are $\dfrac{1}{\lambda_i(A) - \sigma}$

suppose σ closest to λ_k

Same as analysis as before $\rightarrow$

$$\begin{array}{rl} & \left[\begin{array}{c} ((\lambda_k - \sigma)/(\lambda_1 - \sigma))^i \frac{z_1}{z_k} \\ ((\lambda_k - \sigma)/(\lambda_2 - \sigma))^i z_2/z_k \\ \vdots \\ 1 \\ \vdots \\ ((\lambda_i - \sigma)/(\lambda_n - \sigma))^i z_n/z_k \end{array}\right] \\ k^{th}\text{ component} \rightarrow & \end{array}$$

if $z_k \neq 0$, $\rightarrow \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$ k^{th} component

where do we get σ?

update σ based on $\lambda'_{i+1} \rightarrow$

Quadratic convergence for general A
Cubic " for $A = A^T$

Next Algorithm: converge to whole invariant subspace of any dimension, not just 1

Orthogonal Iteration
given Z_0, $n \times p$ orthog matrix
$i = 0$
repeat
$Y_{i+1} = A Z_i$
factor $Y_{i+1} = Z_{i+1} \cdot R_{i+1}$ ... QR decomp
... Z_{i+1} spans approximate invariant subspace
$i = i+1$
until convergence
$p = 1$: same as power method

Convergence: $A = S \Lambda S^{-1}$
$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_p| > |\lambda_{p+1}| \geq \dots \geq |\lambda_n|$$
gap

$$\mathrm{span}(Z_{i+1}) = \mathrm{span}(Y_{i+1}) = \mathrm{span}(A Z_i)$$
$= \mathrm{span}(A^i Z_0)$ ... induction
$$= \mathrm{span}(S \Lambda^i S^{-1} Z_0)$$
$$A^i Z_0 = S \Lambda^i S^{-1} Z_0$$

$$= S\lambda_p^i \operatorname{diag}\Big(\underbrace{\big(\tfrac{\lambda_1}{\lambda_p}\big)^i, \dots, \big(\tfrac{\lambda_{p-1}}{\lambda_p}\big)^i, 1}_{\geq 1}, \underbrace{\big(\tfrac{\lambda_{p+1}}{\lambda_p}\big)^i \dots \big(\tfrac{\lambda_n}{\lambda_p}\big)^i}_{<1}\Big) \cdot S^{-1} Z_0$$

$$= S\lambda_p^i \begin{array}{cc} p & \\ \begin{bmatrix} V_i \\ W_i \end{bmatrix} & \begin{array}{l} p \\ n-p \end{array} \end{array}$$

V_i multiplied by $\left(\frac{\lambda_k}{\lambda_p}\right)^i \geq 1$

W_i multiplied by $\left(\frac{\lambda_k}{\lambda_p}\right)^i < 1$

$W_i \to 0$, V_i grows, stays full rank

$A^i Z_0 \to \lambda_p^i S \begin{bmatrix} V_i \\ 0 \end{bmatrix}$ = linear comb of leading p cols of S

= first p evecs for $\lambda_1, \dots, \lambda_p$

= invariant subspace for $\lambda_1, \dots, \lambda_p$

First col of Z_i same as from power method on one vector

First s cols of Z_i same as running with $p = s$
$<p$

$\Rightarrow$ Orthogonal Iter. computes p invariant subspaces at same time assuming $|\lambda_1| > |\lambda_2| > \dots$

$\Rightarrow$ Why not let $p = n$, $Z_0 = I$ compute n invariant subspaces?
(obstacle: real matrix with $|\lambda| = |\bar{\lambda}|$)

Thm: Run Orthog Iter on A with

$Z_0 = I$, $|\lambda_1| > |\lambda_2| > \cdots$

and all submatrices $S(1:k, 1:k)$ have full rank

then $A_i = Z_i^T A Z_i$ (Z_i orthog, so similar to A)

converges to Schur form

$A_i \to$ upper triangular with

$A_i(j,j) \to \lambda_j$

proof: for each k, span of first k columns of Z_i converge to invariant subspace spanned by first k evecs, i.e. for $\lambda_1, \ldots, \lambda_k$

$$Z_i = \begin{array}{cc} & \begin{array}{cc} k & n-k \end{array} \\ n & \begin{bmatrix} Z_{i1}, & Z_{i2} \end{bmatrix} \end{array}$$

$$Z_i^H A Z_i = \begin{bmatrix} Z_{i1}^H \\ Z_{i2}^H \end{bmatrix} A \left[Z_{i1}, Z_{i2} \right]$$

$$= \begin{array}{cc} & \begin{array}{cc} k & n-k \end{array} \\ \begin{array}{c} k \\ n-k \end{array} & \left[\begin{array}{c|c} Z_{i1}^H A Z_{i1} & Z_{i1}^H A Z_{i2} \\ \hline Z_{i2}^H A Z_{i1} & Z_{i2}^H A Z_{i2} \end{array}\right] \end{array}$$

$\uparrow$ if $(2,1)$ corner $\to 0$ for all $k \Rightarrow$ upper triangular

$Z_{i1} \to$ invariant subspace

$A Z_{i1} \to Z_{i1} \cdot B^{k \times k}$

$$Z_{i2}^{*} A Z_{i1} \to \underbrace{Z_{i2}^{*} Z_{i1}}_{0} \cdot B$$

by orthogonality of Z_i

(for Matlab code used in demo, see typed notes)