# Honors Numerical Analysis

## Next topic: Eigenvalue Problems

Recall: $\lambda, \vec{v}$ are an eigenvalue pair if $A\vec{v} = \lambda\vec{v}$.

Direct computation: form characteristic equation:

$$\underbrace{\det(A - \lambda I)}_{\substack{\text{polynomial in } \lambda \text{ of degree } n \text{ if } A \in \mathbb{R}^{n \times n}. \\ p(\lambda)}} = 0$$

The solution to $p(\lambda) = 0$ are the eigenvalues.

This is expensive for various reasons: forming $p(\lambda)$ cost $n!$ flops.

Then, a nonlinear root finding algorithm must be used to solve $p(\lambda) = 0$. (Bisection, Newton, etc.)

Application: Systems of linear Initial Value problems:

$\vec{y}' = A\vec{y}$ $\qquad$ $\vec{y}(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_n(t) \end{pmatrix}$ $\qquad$ $\vec{y}'(t) = \begin{pmatrix} y_1'(t) \\ y_2'(t) \\ \vdots \\ y_n'(t) \end{pmatrix}$

One solution method is to diagonalize A. (Investigate diagonalization of matrices.)

$$A = \underset{(\vec{v}_1\,\vec{v}_2 \cdots \vec{v}_n)}{P}\;\underset{\begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}}{D}\;P^{-1}$$

$$\vec{y}' = PDP^{-1}\vec{y}$$

$$\underbrace{P^{-1}\vec{y}'}_{\vec{u}'} = D\underbrace{P^{-1}\vec{y}}_{\vec{u}}$$

$\Rightarrow \quad \vec{u}' = D\vec{u}$

$$\Rightarrow \quad \begin{aligned} u_1' &= \lambda_1 u_1 \\ u_2' &= \lambda_2 u_2 \\ &\vdots \\ u_n' &= \lambda_n u_n \end{aligned} \qquad \Rightarrow \qquad \begin{aligned} u_i &= c_i e^{\lambda_i t} \\ u_i' &= \lambda_i c_i e^{\lambda_i t} \\ &= \lambda_i u_i \end{aligned}$$

$\Rightarrow$ Change variables back.

$$\vec{u} = \begin{pmatrix} c_1 u_1 \\ \vdots \\ c_n u_n \end{pmatrix} = P^{-1}\vec{y}$$

$\Rightarrow \quad \vec{y} = P\vec{u}.$

Solving $\vec{y}' = A\vec{y}$ was reduced to finding eigenvalues and eigenvectors of A.

Big Important Theorem:

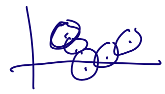
<u>Gerschgorin's Theorem</u> Let $A$ be an $n \times n$ matrix (real or complex). Then, all the eigenvalues of $A$ lie in

$$\bigcup_{i=1}^{n} D_i, \quad \text{where}$$

$i^{th}$ diagonal of $A$

$$D_i = \Big\{ z \in \mathbb{C} \text{ such that } |z - a_{ii}| \le \underbrace{\sum_{j \ne i} |a_{ij}|}_{\text{sum of off-diagonal entries in row } i.} \Big\}$$

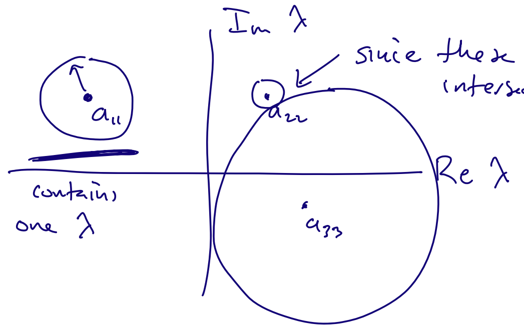
referred to as a "Gerschgorin Disk".



← 5 disks connected,

If $m$ disks are connected, then $m$ eigenvalues are located in this region.

<u>Graphically</u>




Simplest case:

$A$ is diagonal: $A = \begin{pmatrix} a_{11} & & 0 \\ & a_{22} & \\ 0 & & \ddots \\ & & & a_{nn} \end{pmatrix} \Rightarrow \lambda_i = a_{ii}$

<u>Proof:</u> Let $\lambda, \vec{v}$ be an eigenvalue/vector pair.

Then $A\vec{v} = \lambda \vec{v} \Rightarrow$ $i$th component

$$\sum a_{ij} v_j = \lambda v_i \quad \text{for all } i$$

$$\Rightarrow (\lambda - a_{ii}) v_i = \sum_{j \ne i} a_{ij} v_j$$

Pick largest element of $\vec{v}$, call it $v_k$. (in absolute value)

$$\Rightarrow \quad (\lambda - a_{kk}) v_k = \sum_{j \neq k} a_{kj} v_j$$

$$|\lambda - a_{kk}| \cancel{|v_k|} \leq \sum_{j \neq k} |a_{kj}| \underbrace{\frac{|v_j|}{|v_k|}}_{\leq 1}$$

$$\leq \sum_{j \neq k} |a_{kj}|$$

We will revisit this theorem in detail when discussing Jacobi's method.

## The Power Method

Goal Calculate the eigenvalue with largest magnitude and associated eigenvector. (Assume that A is diagonalizable.)

Start with a random vector $\vec{w}$.

If $\vec{w}$ is truly random, then it is a linear combination of every eigenvector of A.

$$\Rightarrow \quad \vec{w} = \sum_{j=1}^{n} c_j \vec{v}_j$$

eigenvectors.

Apply A to $\vec{w}$: $A\vec{w} = A\left(\sum_j c_j \vec{v}_j\right)$

$$= \sum_j c_j A \vec{v}_j$$

$$= \boxed{\sum_j c_j \lambda_j \vec{v}_j}$$

$$A^2 \vec{w} = A(A\vec{w})$$

$$= \sum_j c_j \lambda_j^2 \vec{v}_j$$

$$A^k \vec{w} = \sum_j c_j \lambda_j^k \vec{v}_j$$

For sufficiently large $k$, this is dominated by the largest $\lambda$, $\approx c_1 \lambda_1^k \vec{v}_1$.

(Assume that $|\lambda_1| > |\lambda_2| > |\lambda_3| \cdots$)

If $|\lambda_1| > |\lambda_2|$, sufficiently larger, then $\lambda_1$ dominates.

So eventually, if $\vec{y}^{(k)} = A^k \vec{w}$, then

$$\vec{y}^{(k+1)} = A^{k+1}\vec{w}$$
$$= A\vec{y}^{(k)} \approx \underline{\lambda_1}\,\vec{y}^{(k)}$$

Normalize these iterates on every step:

$$\vec{w}_0 = \vec{w}/\|\vec{w}\|$$

$$\vec{w}_1 = \frac{A\vec{w}_0}{\|A\vec{w}_0\|} \quad \cdots \quad \vec{w}_k = \frac{A\vec{w}_{k-1}}{\|A\vec{w}_{k-1}\|}.$$

Under this normalization, the eigenvector $\lambda_1$ is approximately equal to

(1) $W_{ki}/W_{(k-1)i} \approx \lambda_1$

$i^{th}$ component of $\vec{w}_k$

(2) Better option is to estimate $\lambda_1$ as

$$\lambda_1 \approx (A\vec{w}_k, \vec{w}_k)$$

Since $A\vec{w}_k \approx \lambda_1 \vec{w}_k$

$$\vec{w}_k^T A \vec{w}_k \approx \lambda_1 \underbrace{\vec{w}_k^T \vec{w}_k}_{=1}$$

$=1$ since $\vec{w}_k$ is a unit vector.

The $\vec{w}_k$'s approach $\vec{v}_1$ as $k \to \infty$.

How fast does the power method converge?

If $k$ is sufficiently large, then $A^k\vec{w} \approx c_1 \lambda_1^k \vec{v}_1$ (assume $c_1 > 0$)

$$A^k \vec{w} \approx c_1 \lambda_1^k \vec{v}_1$$

$$\Rightarrow \vec{v}_1 \approx \frac{1}{c_1 \lambda_1^k} A^k \vec{w}$$

$$= \frac{1}{c_1 \lambda_1^k}\left(c_1 \lambda_1^k \vec{v}_1 + c_2 \lambda_2^k \vec{v}_2 + \dots + c_n \lambda_n^k \vec{v}_n\right)$$

$$= \vec{v}_1 + \frac{c_2}{c_1}\left(\frac{\lambda_2}{\lambda_1}\right)^k \vec{v}_2 + \frac{c_3}{c_1}\left(\frac{\lambda_3}{\lambda_1}\right)^k \vec{v}_3 + \dots + \frac{c_n}{c_1}\left(\frac{\lambda_n}{\lambda_1}\right)^k \vec{v}_n .$$

If $|\lambda_j| < |\lambda_1|$ for $j > 1$, then $\left(\frac{\lambda_j}{\lambda_1}\right)^k \to 0$ as $k \to \infty$

$$\vec{w}_k = \frac{A^k \vec{w}}{\| A^k \vec{w} \|}$$

$$\| \vec{w}_k - \vec{v}_1 \| \approx \left|\frac{c_2}{c_1}\right| \left|\frac{\lambda_2}{\lambda_1}\right|^k$$

$$\sim O\left( \left|\frac{\lambda_2}{\lambda_1}\right|^k \right)$$

The convergence of the power method depends on the gap in the eigenvalues.

I.e. the relative size of $\lambda_2$ to $\lambda_1$.

This means that if $\left|\frac{\lambda_2}{\lambda_1}\right| \approx 1$, then convergence is very slow.

How do we accelerate this convergence?

Idea one: Power method with shift

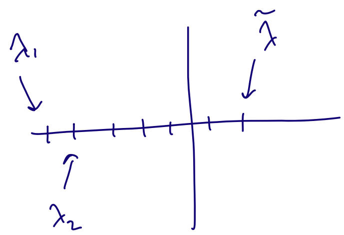
If a matrix $A$ has eigenvalues $\lambda_1 \ldots \lambda_n$, then $A - sI$ has eigenvalues $\lambda_i - s$.

Pf: $(A - sI)\vec{v} = A\vec{v} - s\vec{v}$

$$= \lambda\vec{v} - s\vec{v}$$

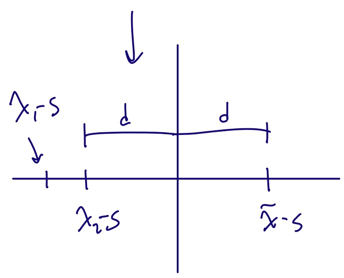
$$= (\lambda - s)\vec{v}$$

Choose $s$ to increase the convergence rate:




to minimize the ratio $\left|\frac{\lambda_2}{\lambda_1}\right|$,

choose shift such that $|\lambda_2| = |\tilde{\lambda}|$.




I.e. choose $s$ such that

$$\left|\frac{\lambda_2 - s}{\lambda_1 - s}\right| = \left|\frac{\tilde{\lambda} - s}{\lambda_1 - s}\right|.$$

Power method with shift allows for computing the most negative or the most positive eigenvalue.