Lect 1

NLA

Yuji Nakatsukasa
nakatsukasa@maths.ox.ac.uk

prereq.
$A = A^T = V\Lambda V^T$

notes
slides

Goals: solve/understand problems involving matrices $A \in \mathbb{R}^{m\times n}$

1. Understand the SVD (Sing val Decomp) $A = U\Sigma V^T$ $\boxed{A}$
   proof, properties applications
   computation

A_{ij} : (i,j) entry

2. Solve $\boxed{A}x = b$ or $\min \|\boxed{A}x - b\|_2$

   linear system — Least-squares problem

   given $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$, find x

   given $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^n$, $m \geq n$, find x

3. solve $Ax = \lambda x$ (or $A = U\Sigma V^T$)

   given $A \in \mathbb{R}^{n\times n}$, find $\lambda \in \mathbb{C}$, $x \in \mathbb{C}^n$

Actual coverage: 1, then
algorithms for 2+3.

- Direct (classical) methods. $(n \lesssim 10^4)$
- Iterative (Krylov) methods. $(n \lesssim 10^6$ ish$)$
- Randomised methods. (“beyond”) “ ∠ books
  ∠

Why $Ax = b$ and $Ax = \lambda x$?

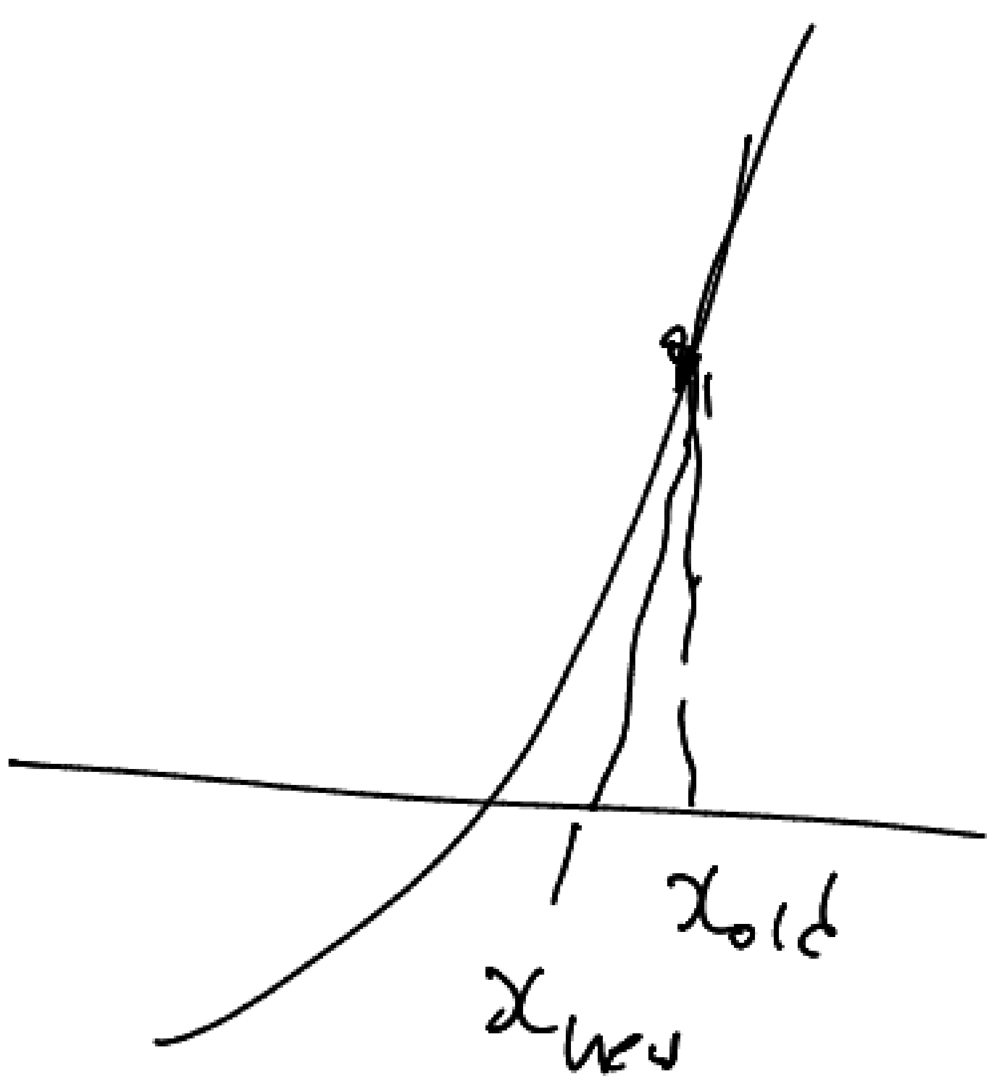
- So many problems boil down to them!

e.g. Newton's method for {high-dim v of}

$\min_x f(x)$ $\quad f : \mathbb{R}^n \to \mathbb{R}$.

idea: solve $F := \nabla f = 0$.

$$= \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix} \in \mathbb{R}^n$$



$x := x - J^{-1} F(x)$

$(= x - \delta x)$

J: gradient of F = Hessian of f

$$J_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}$$

$\delta x = J^{-1} F$

$\Leftrightarrow \underbrace{J \delta x = F}_{\text{lin. sys!}}$

most computational work in solving

- Eigenvalue problems: Google Pagerank (Gleich 2020 SIAM Review)

Schrodinger eqn

finding roots (poly) (revisit)

hence min. $p(x)$ via finding $p'(x) = 0$

close link to SVD

Basic LA review

A nonsingular $\Leftrightarrow$ - - - - -

Structured matrices (slide)

Symmetric $A = A^T \Rightarrow A = V \Lambda V^T$, V orth, $\Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$

$\Leftarrow$ if Λ real.

Vector norms

for $x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \in \mathbb{R}^n$, $\|x\|_2 = \sqrt{x^T x} = \sqrt{x_1^2 + \cdots + x_n^2}$

$\|x\|_p = \left(|x_1|^p + \cdots + |x_n|^p\right)^{\frac{1}{p}}$

most often: $p = 1, 2, \infty$

$\|x\|_\infty = \max_i |x_i|$

axioms - -

inequalities. (exercise)

$$\underbrace{\frac{1}{\sqrt{n}} \|x\|_2 \le \underbrace{\|x\|_\infty}_{} \le \|x\|_2}_{\text{trivial}}$$

$$\|x\|_2 = \sqrt{|x_1|^2 + \cdots + |x_n|^2} \le \sqrt{\|x\|_\infty^2 + \cdots + \|x\|_\infty^2} = \sqrt{n} \|x\|_\infty .$$

$(\|x\|_p \le \|x\|_q$ if $p > q)$

$$\underbrace{\frac{1}{\sqrt{n}} \|x\|_1 \le \|x\|_2}_{\text{Cauchy-Schwarz}} \underbrace{\le \|x\|_1}_{\text{easy.}}$$

C-S

$|x^T y| \le \|x\|_2 \|y\|_2$

proof: $0 \le \|x - cy\|_2^2 = c^2 y^T y - 2c\, y^T x + x^T x$.

min. when $= y^T y \underbrace{\left(c - \frac{y^T x}{y^T y} \right)}_{=0}^2 - \frac{(y^T x)^2}{y^T y} + x^T x \ge 0$

so $x^T x \ge (y^T x)^2 y^T y$.

Matrix norms

$$\|A\|_p = \max_x \frac{\|Ax\|_p}{\|x\|_p}$$

$p = 2$: spectral norm, $\|A\|_2 = \sigma_{max}(A)$

(To come!)

$p = 1$: $\|A\|_1 =$ (max col sum)

$\|A\|_\infty =$ (max row sum)

Frobenius norm $\|A\|_F = \sqrt{\sum_{i,j} |a_{ij}|^2}$

$= \sqrt{\text{Trace}(A^T A)}$ $\left(= \sqrt{\vec{A}^T \vec{A}}\right)$

$$\vec{A} = \begin{bmatrix} a_{11} \\ \vdots \\ a_{n1} \\ a_{21} \\ \vdots \\ a_{nn} \end{bmatrix} \in \mathbb{R}^{n^2}$$

$\frac{1}{\sqrt{n}} \|A\|_\infty \le \|A\|_2 \le \sqrt{m} \|A\|_\infty$

via vector ineq.

subordinate property $\|AB\|_p \le \|A\|_p \|B\|_p$.

not for all norms!

(another proof for C-S:

$|x^T y| = \|x^T y\|_2 \le \|x^T\|_2 \|y\|_2 = \|x\|_2 \|y\|_2$)

J 2022-1

Subspaces.

$$S = \text{span}\{\underset{\substack{\uparrow \\ \text{scalar}}}{c_i}\,\underset{\substack{\uparrow \\ \text{vector}}}{v_i}\}$$

$v_1 \cdots v_d$: lin. indep.
↳ not unique!

$$x \in S \iff x = \sum_{i=1}^{d} c_i v_i$$

VM works for any nonsingular M.

convenient representation; via $\left[\, V \,\right] = \left[v_1 \cdots v_d \right]$

s.t.

$$x \in S \iff x = Vc$$

often V orthonormal.

Thm: Subspaces $\underset{d_1}{n\left[\, V_1 \,\right]}$ and $\underset{d_2}{n\left[\, V_2 \,\right]}$ have nontrivial intersection

if $d_1 + d_2 > n$. (i.e., $\exists\, x \neq 0$ s.t. $x \in V_1$ and $x \in V_2$.)

proof: " $n\underset{d_1+d_2>n}{\left[V_1 \;\; V_2 \right]}$ has null vector".

via SVD or QR

$$\left[V_1 \;\; V_2 \right]\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = 0 \iff V_1 c_1 = -V_2 c_2 = x$$

is in both.

$\boxed{M}$

if $d_1 + d_2 = n$ not true

$<$

SVD.

Recall if $A = A^T$ (symmetric),

$A = V \Lambda V^T$ $\quad V^T V = I = V V^T$. mat of eigvecs.

$\Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$. mat of eigvals.

Thm: SVD. Let $A \in \mathbb{R}^{m \times n}$, $m \geq n$. Then $\exists$ U orthonormal, V orthogonal

s.t.

$\Sigma = \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \end{bmatrix}$, $\sigma_i \geq 0$. $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$. sing vals.

$\boxed{A} = \boxed{U} \boxed{\Sigma} \boxed{V^T}$.

$[u_1 \cdots u_n] \leftarrow$ Left sing vecs

v_i: right sing vecs $\begin{bmatrix} v_1^T \\ \vdots \\ v_n^T \end{bmatrix}$

(when $m < n$, $\boxed{A^T} = \boxed{V} \Sigma \boxed{U^T} \Leftrightarrow A = \boxed{U} \Sigma \boxed{V^T}$)

"rotate - strech/shrink - rotate".

proof: treat $m \geq n$ case.

$A^T A$ is symmetric and positive semidef $\leftarrow$ $A^T A x = \lambda x$, $x^T A^T A x = \lambda x^T x$, $\lambda = \frac{\|Ax\|^2}{\|x\|^2}$

$= V \Lambda V^T$, $\Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$ $\quad \lambda_i \geq 0$

so consider $\boxed{B} = \boxed{A} V$. $\quad B^T B = \Lambda$

so if $\lambda_n > 0$, $\quad B \Lambda^{-\frac{1}{2}}$ is orthonormal. $=: U$. $\quad \Lambda^{\frac{1}{2}} =: \Sigma$

and $A = B V^T = U \Sigma V^T$.

(if $\lambda_n = 0$, $\quad B \begin{bmatrix} \lambda_1^{-\frac{1}{2}} & & & \\ & \ddots & & \\ & & \lambda_r^{-\frac{1}{2}} & \\ & & & 1 \\ & & & & \ddots \\ & & & & & 1 \end{bmatrix} = [U_1 \ 0]$ and

$A = B V^T = [U_1 \ 0] \begin{bmatrix} \Sigma_1 & \\ & I \end{bmatrix} V^T$

$= [U_1] \Sigma_1 V_1^T \quad = [U_1 \ U_\perp] \begin{bmatrix} \Sigma_1 & \\ & 0 \end{bmatrix} V^T$ last page

Full SVD $A = U \Sigma V^T = [U \ U_\perp] \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} V^T$, $\boxed{A} = \boxed{U} \Sigma \boxed{V^T} = U [\Sigma \ 0] \begin{bmatrix} V^T \\ V_\perp^T \end{bmatrix}$

row, colums space

rank.

two representations

$$A = U\Sigma V^T$$

$$= [U_1\ U_2]\begin{bmatrix}\Sigma_r & \\ & 0\end{bmatrix}\begin{bmatrix}V_1^T \\ V_2^T\end{bmatrix} = \sum_{i=1}^{\textcircled{r}} \sigma_i u_i v_i^T$$

column space: span$\{Ax\}$

$$Ax = \sum_{i=1}^{r} \sigma_i u_i (v_i^T x) \in \text{span}\{u_1 \dots u_r\}.$$

etc.

J2022-2

$$\left(\begin{array}{l} Av_i = \sigma_i u_i, \quad u_i^T A = \sigma_i v_i \\ [U\ U_\perp] \text{ orthogonal.} \end{array}\right)$$

SVD and eigval decomp

$$A = U\Sigma V^T \Longrightarrow A^T A = V\Sigma^2 V^T. \quad (1)$$

$$AA^T = U\Sigma^2 U^T \quad (2)$$

(warning: $(1)(2) \Longrightarrow A \cancel{=} U\Sigma V^T$ not always (need care in sign of eigvecs V, U's)

— $\sigma_i^2(A) = \lambda_i(A^T A)$ (so σ_i unique)

— σ_i of structured matrices. (sheet)

— $A = U\Sigma V^T \Longleftrightarrow \underset{\text{symmetric.}}{\begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}} = \underbrace{\begin{bmatrix} U & U & U_\perp \\ V & -V & 0 \end{bmatrix}}_{W} \begin{bmatrix} \Sigma & & \\ & -\Sigma & \\ & & 0 \end{bmatrix} W^T$

(proof: $\begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}\begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} U\Sigma \\ V\Sigma \end{bmatrix} = \begin{bmatrix} U \\ V \end{bmatrix}\Sigma$.)

$A^T U_\perp = 0.$

$$A = [U\ U_\perp]\begin{bmatrix}\Sigma \\ 0\end{bmatrix} V^T$$

Uniqueness.

$$A = U \Sigma V^T$$

$$\Rightarrow A = U S S^{-1} \Sigma S^{-1} \underset{\uparrow}{S} V^T$$

$$S = \begin{bmatrix} \pm 1 & & \\ & \ddots & \\ & & \pm 1 \end{bmatrix}$$

when $\sigma_i = \sigma_j$ (e.g. $\sigma_1 = \sigma_2$)

$$A = U \begin{bmatrix} \sigma_1 I_2 & & & \\ & \sigma_3 & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} V^T$$

$$= U \begin{bmatrix} \sigma_1 \underset{\uparrow}{Q} Q^T & & & \\ & \sigma_3 & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} V^T$$

$Q \in \mathbb{R}^{2 \times 2}$ orthogonal.

$$= \underbrace{U \begin{bmatrix} Q & \\ & I_{n-2} \end{bmatrix}}_{\tilde{U}} \begin{bmatrix} \sigma_1 I_2 & & & \\ & \sigma_3 & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} \underbrace{\begin{bmatrix} Q^T & \\ & I_{n-2} \end{bmatrix} V^T}_{\tilde{V}^T}$$

this is only nonuniqueness; essentially as V is eigvec matrix.

$$\|A\|_2 = \max \frac{\|Ax\|_2}{\|x\|_2} \underbrace{= \sigma_1}_{\text{Claim.}}$$

$$= \max_{\|x\|_2=1} \|Ax\|_2$$

$$= \max_{''} \| U \Sigma \underbrace{V^T x}_{y} \|_2$$

orthonormal

$$\|Uz\|_2^2 = \|z\|_2^2$$

$$(Uz)^T(Uz) = z^T U^T U z = z^T z$$

$$= \max_{\|y\|_2=1} \|\Sigma y\|_2$$

$$= \max_{''} \left\| \begin{bmatrix} \sigma_1 y_1 \\ \sigma_2 y_2 \\ \vdots \\ \sigma_n y_n \end{bmatrix} \right\|_2 = \max_{''} \sqrt{\sum_{i=1}^{n} (\sigma_i y_i)^2}$$

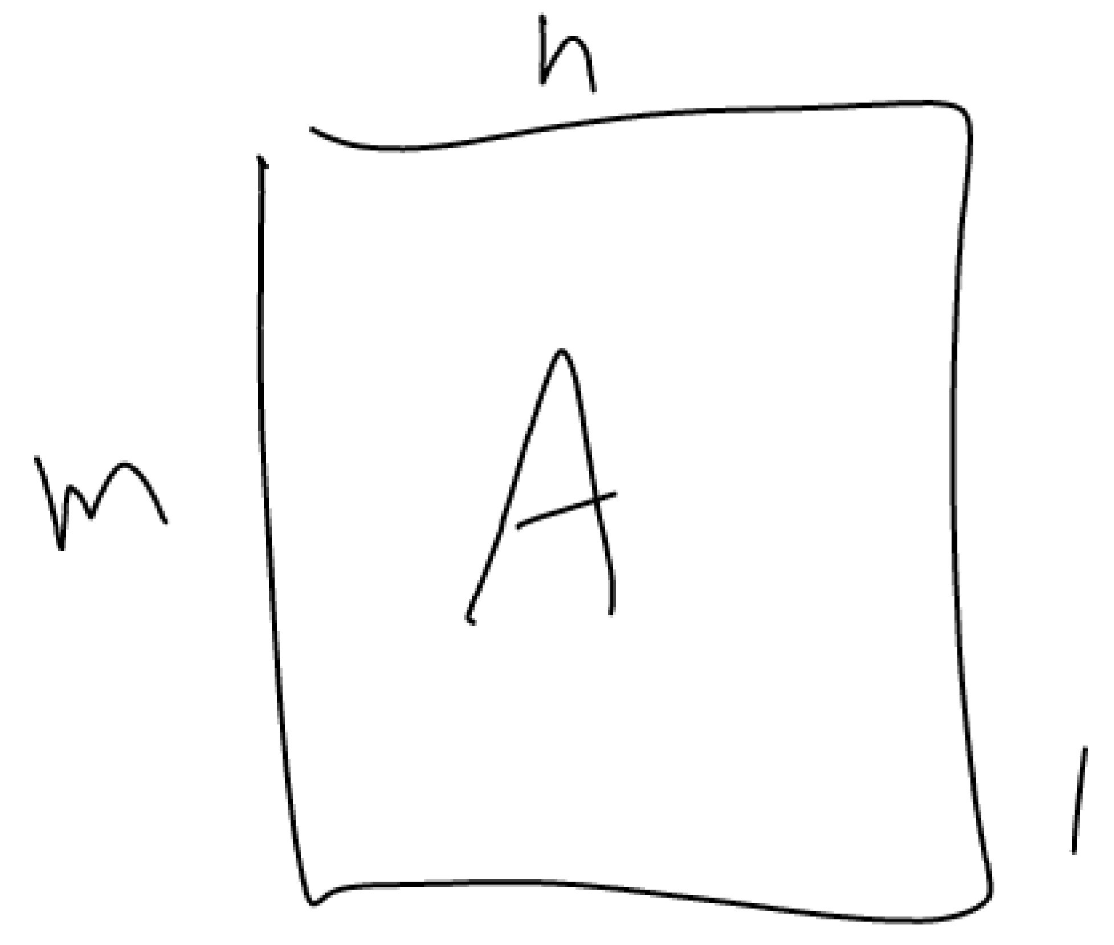
$$\le \max_{\|y\|_2=1} \sqrt{\sigma_1^2 \sum_{i=1}^{n} y_i^2} = \sigma_1 .$$

$$\left(\begin{aligned} \|A\|_F^2 &= \mathrm{Tr}(A^T A) \\ &= \mathrm{Tr}((U\Sigma V^T)^T U \Sigma V^T) \\ &= \mathrm{Tr}(V \Sigma^2 V^T) \end{aligned} \right)$$

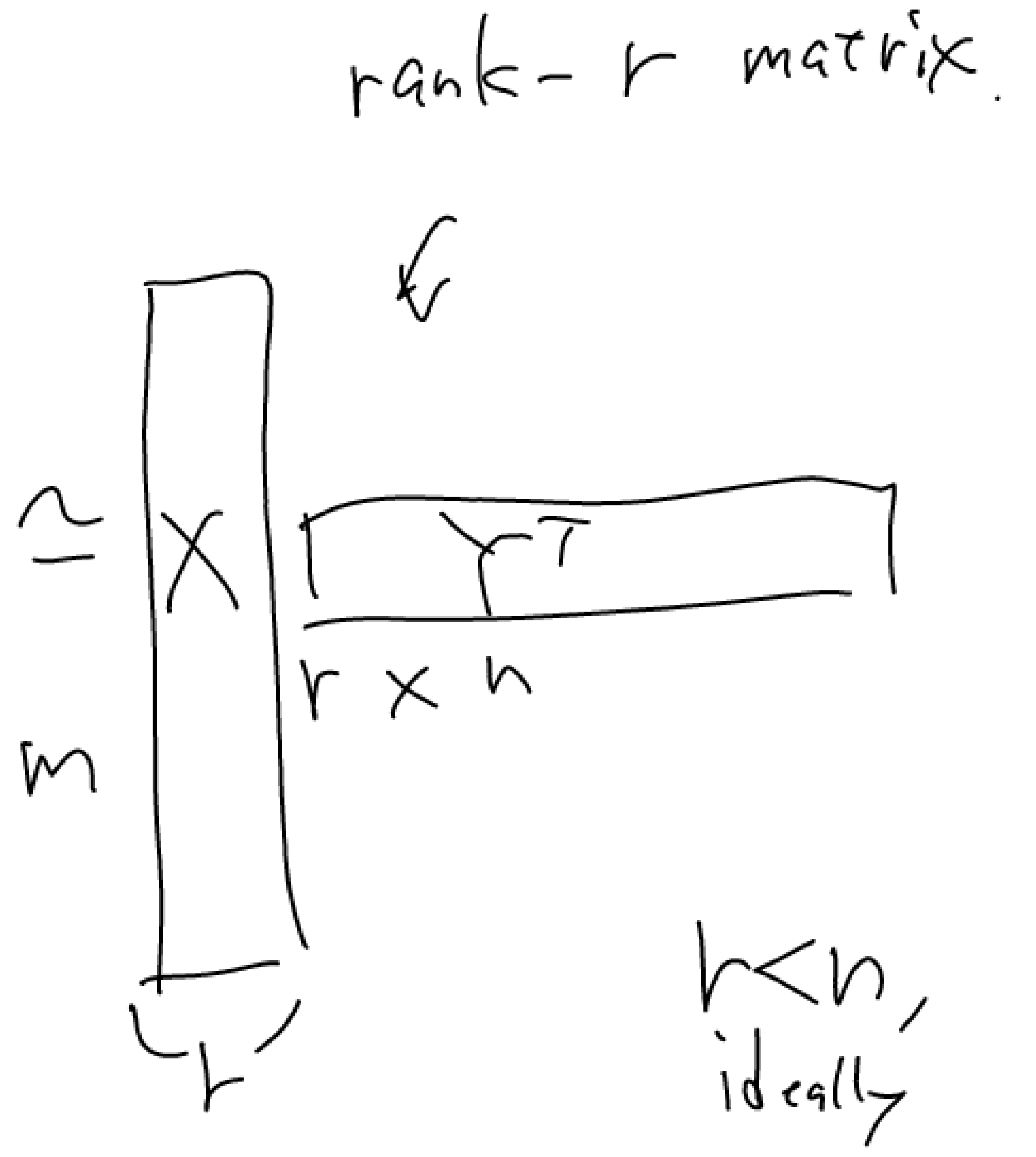
$$= \mathrm{Tr}(\Sigma^2)$$

Low-rank approx (Application of SVD)

Given



approx. by



$r<n$, ideally $r \ll n$.

why?
- Storage
  $mn \to (m+n)r$
- "interpretable"
- computational efficiency
  $Ax : 2mn$, vs $XY^Tx : 2(m+n)r$

Low-rank matrix

$\mathrm{rank}(B) \overset{\leq}{=} r$

$$\Leftrightarrow B = XY^T = \begin{bmatrix} x_1 \cdots x_r \end{bmatrix} \begin{bmatrix} y_1^T \\ \vdots \\ y_r^T \end{bmatrix} = x_1y_1^T + x_2y_2^T + \cdots + x_ry_r^T$$

$$= \sum_{i=1}^{r} x_i y_i$$

"$\Rightarrow$": via SVD.

for "$\Leftarrow$" i.e. $\mathrm{rank}(B) \leq r$, let

$$X = U_X \Sigma_X V_X^T$$
$$Y = U_Y \Sigma_Y V_Y^T$$

and note

$$XY^T = U_X (\underset{\substack{\uparrow \\ r\times r.}}{\Sigma_X V_X^T} V_Y \Sigma_Y) U_Y^T$$

$$= \underset{n\times r}{\boxed{U_X U_r}} \underset{\substack{\uparrow \\ r\times r}}{\Sigma_r} \underset{\substack{\uparrow \\ r\times n}}{V_r^T U_Y^T}$$

economical SVD.

Q: find "best" rank-r approximation $A_r = \sum_{i=1}^{r} x_i y_i$, i.e.

$$\|A - A_r\|_2 \leq \|A - B\|_2, \quad \forall B \ \text{rank}(B) \leq r.$$

and $\|A - A_r\|_2 = \sigma_{r+1}$.

Solution: Truncated SVD A_r

$$A = U \Sigma V^T = \begin{bmatrix} u_1 \cdots u_n \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} \begin{bmatrix} v_1^T \\ \vdots \\ v_n^T \end{bmatrix}$$

$$= u_1 \sigma_1 v_1^T + u_2 \sigma_2 v_2^T + \cdots + u_n \sigma_n u_n^T$$

$$= \sum_{i=1}^{n} u_i \sigma_i v_i^T$$

Then

$$A_r = \sum_{i=1}^{\textcircled{r}} u_i \sigma_i v_i^T .$$

Note: great approx when $\sigma_{r+1} \approx 0$, otherwise not.
(not all matrices are approx. low-rank)

proof: first,

$$A - A_r = \sum_{i=r+1}^{n} u_i \sigma_i v_i^T$$

$$= \begin{bmatrix} u_{r+1} \cdots u_n \end{bmatrix} \begin{bmatrix} \sigma_{r+1} & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} \begin{bmatrix} v_{r+1}^T \\ \vdots \\ v_n^T \end{bmatrix} \quad \text{SVD!}$$

so $\|A - A_r\|_2 = \sigma_{r+1}$.

For the main part: WTS

$$\underset{\text{rank} \leq r.}{\|A - \overset{\uparrow}{B}\|_2} \geq \underset{\sim}{\|A - A_r\|_2} = \sigma_{r+1}$$

1. Write $B = \underbrace{\boxed{B_1}}_{r.}\boxed{B_2^T}$

2. B has null space $\underbrace{\boxed{W}}_{n-r}$ ↖ orthonormal. s.t. $BW = 0.$

Then

$$\|A - B\|_2 \geq \|(A-B)W\|_2 = \|AW\|_2$$

subordinate $\|XY\|_2 \leq \|X\|_2 \|Y\|_2$.

3. WTS $\|AW\|_2 \geq \sigma_{r+1}$.

Let $A = \sum_{i=1}^{n} u_i \sigma_i v_i^T$. let $V_1 = [v_1 \cdots v_{r+1}]$, consider.

$[W, V_1] = [\underbrace{\overset{W}{w_1 \cdots w_n},}_{n-r} \underbrace{\overset{V_1}{v_1 \cdots v_{r+1}}}_{r+1}] \in \mathbb{R}^{n \times (n+1)}$ has a null vector, $\begin{bmatrix} x_w \\ x_v \end{bmatrix} \in W \cap V_1$

$\exists$ $Wx_w = -V_1 x_v$ let $\|Wx_w\|_2 = 1$.

so $\|x_v\|_2 = 1$.

4. Then $\|AW\|_2 \geq \|AWx_w\|_2 = \|AV_1 x_v\|_2$

$$= \|U\begin{bmatrix}\Sigma_1 & \\ & \Sigma_2\end{bmatrix}[V_1\ V_2]^T V_1 x_v\|_2 = \|\begin{bmatrix}\Sigma_1 & \\ & \Sigma_2\end{bmatrix}\overset{(r+1)\times(r+1)\ \swarrow}{\begin{bmatrix} I_{r+1} \\ 0 \end{bmatrix}} x_v\|_2$$

$$\geq \sigma_{r+1}.$$

(demo: Image → matrix $\boxed{A}$ → low-rank approx $\boxed{A} \simeq \boxed{U_r}\boxed{\Sigma_r}\boxed{V_r^T}$ → Image

Courant - Fischer therome. (check spelly.) (properties of SVD)

$$\sigma_i(A) = \max_{\dim(S)=i} \min_{\substack{x\in S\\ \|x\|_2=1}} \|Ax\|_2$$

i.e.,

$$\left(= \max_{Q,\ Q^TQ=I_i} \underbrace{\min_{\|y\|_2=1} \|AQy\|_2}_{\sigma_{\min}(AQ) = \sigma_i(AQ)} \right)$$

proof: Fix $S = \text{span}(\underset{i}{Q})$

$$\min_{x\in S} \|Ax\|_2 = \min_y \|\Sigma V^T Q y\|_2$$

now let $V_i = [v_i \cdots v_n]$ ← R singvecs of A.

then $n\,[\underbrace{V_i}_{n-i+1}\ \underbrace{Q}_{i}]$ has null vector, $\Rightarrow$ $w \in \text{span}(V_i)$, $w \in \text{span}(Q)$

i.e. $w = Qy_*$

so $V^T w = \begin{bmatrix} v_1^T \\ \vdots \\ v_{i-1}^T \\ v_i^T \\ \vdots \\ v_n^T \end{bmatrix} w = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ * \end{bmatrix}$

then

$$\|Aw\|_2 = \left\| \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} V^T w \right\|_2$$

$$= \left\| \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \begin{bmatrix} \sigma_i & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} V_i^T w \end{bmatrix} \right\|_2 \le \underbrace{\left\| \begin{bmatrix} \sigma_i & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} \right\|_2}_{\le \sigma_i} \underbrace{\|V_i^T w\|_2}_{\le 1} \le \sigma_i$$

(holds for any $S = \text{span}(Q)$), so RHS $\le \sigma_i$.

now, take $Q = [v_1 \cdots v_i]$. Then ($w = v_i$ (ith singvec))

$$\min_{\substack{x\in S\\ \|x\|=1}} \|Ax\|_2 = \min_y \|AQy\|_2 = \min_y \left\| \begin{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_i \end{bmatrix} \underbrace{V_i^T y}_{\text{norm}=1} \\ 0 \end{bmatrix} \right\| \ge \sigma_i.$$

Corollaries of C-F.

- Weyl's Thm. $\sigma_i(A+E) \in \sigma_i(A) + [-\|E\|_2, \|E\|_2]$.

$$\sigma_i(A+E) = \max_{Q^TQ=I_i} \min_{y} \underbrace{\|(A+E)Qy\|_2}$$

$$\le \|AQy\|_2 + \underbrace{\|EQy\|_2}_{\le \|E\|_2.}$$

$$\ge \|AQy\|_2 - \|EQy\|_2$$

$$\ge \|AQy\|_2 - \|E\|_2.$$

- Lots of inequalities on σ_i via C-F

e.g.

$$\sigma_i\left(\begin{bmatrix} A_1 \\ A_2 \end{bmatrix}\right) \ge \sigma_i(A_1), \sigma_i(A_2)$$

proof: $= \max \min \underbrace{\left\| \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} x \right\|_2}_{\ge \|A_1x\|_2, \ \|A_2x\|_2.}$

$$\sigma_i\left(\begin{bmatrix} A_1 & A_2 \end{bmatrix}\right) \ge \sigma_i(A_1), \sigma_i(A_2)$$

$$= \max_{Q^TQ=I_i} \min_{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in Q.} \left\| \begin{bmatrix} A_1 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right\|_2 \ge \max_{Q=\begin{bmatrix} Q_1 \\ 0 \end{bmatrix}} \min \underbrace{\left\| \begin{bmatrix} A_1 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right\|_2}_{\| [A_1x_1] \|_2} = \sigma_i(A_1).$$

(for many more, see "Topics in Matrix Analysis Horn + Johnson" Ch.3)

Weyl's Thm for $\underset{\sim}{A}x = \lambda x$

$A = A^T$.

(note $|\lambda_i| = \sigma_i$ for $A = A^T$.)

$$\lambda_i(A) - \|E\|_2 \le \lambda_i(A+E) \le \lambda_i(A) + \|E\|_2$$

Singvals are $\underset{\sim\sim\sim}{\text{well-conditioned}}$. (always)

(small perturbations don't change σ_i too much.)

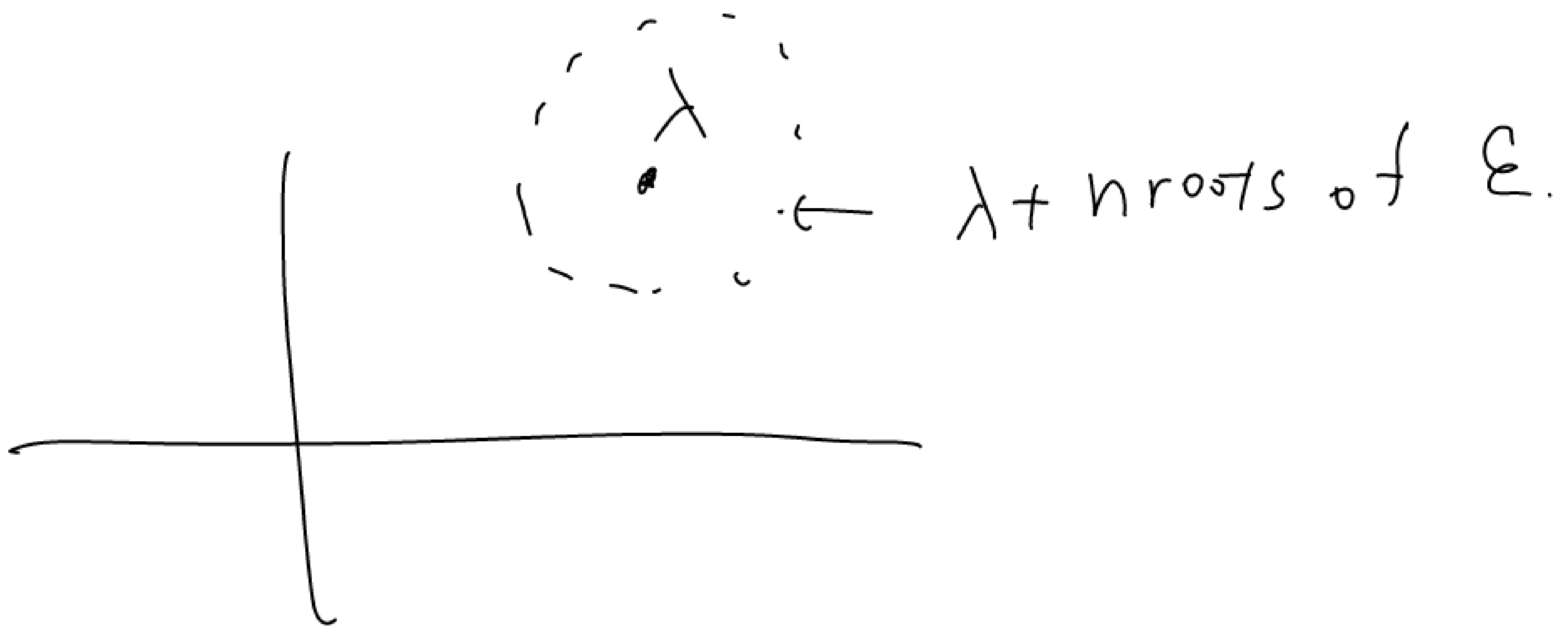
eigvals of $A = A^T$ // .

NOT true for $A \ne A^T$!

$$\begin{bmatrix} \lambda & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ \varepsilon & & & \lambda \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \lambda & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda \end{bmatrix}$$

$\uparrow$ eig $\simeq \lambda + \varepsilon^{\frac{1}{n}}$

$\uparrow$ eig $= \lambda$.



Matrix Decomps.

(Slide)

(Start discussing $Ax=b$.)

$Ax=b$

1. Find L, U s.t. $A = LU$

2. Solve $Ly=b$ for y $(Ux=y)$

3. solve $Ux=y$ for x.

2, 3 are <u>triangular</u> systems. $\Rightarrow$ $O(n^2)$ flops to solve.

e.g.
$$\begin{bmatrix} U_{11} & \cdots & \\ & U_{ij} & \\ & & \ddots \end{bmatrix}\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y \\ \vdots \\ y_n \end{bmatrix}$$
via "back substitution"

$$U_{nn} x_n = y_n \iff x_n = \frac{y_n}{U_{nn}}$$

$$U_{n-1,n-1} x_{n-1} + U_{n-1,n} x_n = y_{n-1} \iff x_{n-1} = \frac{1}{U_{n-1,n-1}}(y_{n-1} - U_{n-1,n} x_n)$$

generally, at i'th step

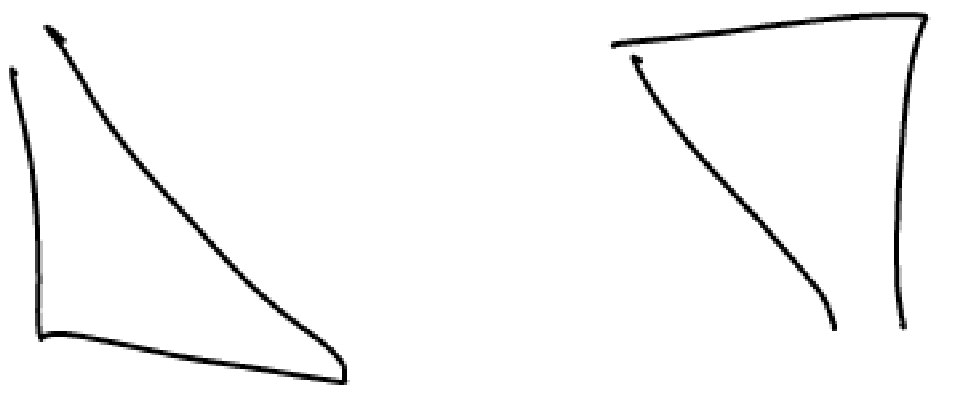
$$x_{n-i} = \underbrace{\frac{1}{U_{n-i,n-i}}\Big(y_{n-i} - \underbrace{\sum_{j=n-i+1}^{n} U_{n-i,j}\, x_j}_{\text{known.}}\Big)}_{O(i) \text{ flops.}}$$
so

total: $\sum_{i=1}^{n} i = O(n^2)$ flops.

LU decomposition.

$$\boxed{A} = \boxed{L}\,\boxed{U}$$

(usually) n×n



lower upper triangular.

for 2023-, Questionable if $\sum L_i U_i$ notation helpful,

useful to think wrt

$$LU = [l_1 l_2 \cdots l_n]\begin{bmatrix} u_1^T \\ \vdots \\ u_n^T\end{bmatrix} = \overbrace{L_1U_1 + L_2U_2 + \cdots + L_nU_n}^{\text{each rank-1}}$$

$$[l_1\, 0 \cdots 0]\begin{bmatrix}u_1^T\\0\\\vdots\\0\end{bmatrix} \qquad \underbrace{[0 \cdots 0\, l_i\, 0 \cdots 0]\begin{bmatrix}0\\\vdots\\0\\u_i^T\\0\\\vdots\\0\end{bmatrix}}$$

$$\begin{bmatrix}0 & 0\\ & \boxed{*}\end{bmatrix}$$

$l_i u_i^T$

$$\begin{bmatrix}0\\0\\ *\end{bmatrix}[0 \cdots 0\ *]$$

to compute,

note $L_1U_1 = A$ in 1st $\begin{cases}\text{row}\\\text{column.}\end{cases}$

so

$$\begin{bmatrix}A_{11}&\cdots&A_{1n}\\\vdots&\ddots&\\A_{n1}&&A_{nn}\end{bmatrix} = \underbrace{L_1U_1}_{l_1u_1^T} + \begin{bmatrix}0&\cdots&0\\\vdots&\underset{\text{nonzero.}}{*}&\\0&&\end{bmatrix}$$

essentially uniquely determines L_1, U_1 (if exist).

i.e. $l_1 = c\cdot\begin{bmatrix}A_{11}\\\vdots\\A_{n1}\end{bmatrix}$,

$u_1^T = \hat{c}\cdot[A_{11}\cdots A_{1n}]$, (if $A_{11} \neq 0$!)

convention: $l_{11} = 1$. $l_1 = \begin{bmatrix}A_{21}/A_{11}\\\vdots\\A_{n1}/A_{11}\end{bmatrix}$, $u_1 = [A_{11}\cdots A_{1n}]$.

$$A = l_1 u_1^T + \begin{bmatrix} 0 & \cdots & 0 \\ 0 & A_1 & \end{bmatrix}$$

$$\uparrow \; A - L_1 U_1$$

Carry on.

$$A_1 = l_2 u_2^T + \begin{bmatrix} 0 & \cdots & 0 \\ 0 & A_3 & \end{bmatrix}$$

then

$$A = l_1 u_1^T + \begin{bmatrix} 0 \\ l_2 \end{bmatrix} [0 \; u_2^T]^T + \cdots + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ l_n \end{bmatrix} [0 \cdots 0 \; u_n^T]$$

$$= \begin{bmatrix} & 0 & 0 \; 0 & & 0 \\ l_1 \; l_2 & 0 & 0 \; 0 & \cdots & \vdots \\ & l_3 & l_4 & & 0 \\ & & & & l_n \end{bmatrix} \begin{bmatrix} u_1^T \\ 0 \; u_2^T \\ 0 \; 0 \; u_3^T \\ \vdots \\ 0 \cdots 0 \; u_n^T \end{bmatrix} = L U.$$

$(2(n-i)^2$ at ith step$)$

Cost: $\frac{2}{3} n^3$ flops (sheet).

Triangular matrices (☆) notes?

$L_1 + L_2 \in L$ ((lower) triangular

$L_1 L_2 \in L$

$L_1^{-1} \in L$

solving $\triangle x = b$. via forward substitution;

$L_{11} x_1 = b_1$,

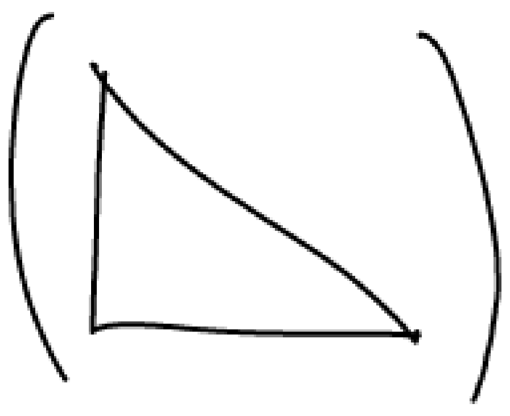
$L_{21} x_1 + L_{22} x_2 = b_2 \Rightarrow x_2 = \square$.

$L_{31} x_1 + L_{32} x_2 + L_{33} x_3 = b_3 \Rightarrow x_3 = \square$

$\vdots$

$\Big\} O(n^2)$ flops.

Standard method for $Ax=b$:
(classical, Dense)

1. $A = PLU$. $\frac{2}{3}n^3$
2. solve $Ly = Pb$. $y=b$. $O(n^2)$
3. $Ux = y$. $O(n^2)$

Then $PAx = LUx = Ly = Pb$.
so $Ax=b$

overall $\frac{2}{3}n^3$ flops.

Caveat: we assumed ((1,1) entry of A_i was $\neq 0$!)

if not, pivot (permute rows s.t. (1,1) entry $\neq 0$)

instead of

$$A_i = l_i u_i^T + \boxed{\begin{matrix} & 0 \\ 0 & * \end{matrix}},$$

$$P_i A_i = l_i u_i^T + \begin{bmatrix} & 0 \\ 0 & * \end{bmatrix}.$$

gives

P_i permutation (orthogonal) & 0,1. $\begin{bmatrix} & 1 & \\ 1 & & 1 \\ & 1 & \end{bmatrix}$

(not so immediate gives $\overset{\wedge}{LU}$ of A or similar)

$$\underbrace{P_{n-1}\cdots P_1}_{}A = LU$$
$= PA$

after k steps:

$$P_k \cdots P_1 A = L_1U_1 + \cdots L_kU_k + \boxed{\quad}$$

pivot at (k+1)st step

$$P_{k+1}(\qquad) = \underbrace{P_{k+1}(\qquad)}_{} + \boxed{\begin{matrix} & \\ 0 & * \end{matrix}}$$

$(P_{k+1}L_i =: \tilde{L}_i$ still triangular!

$= \hat{L}_1U_1 + \cdots + \hat{L}_kU_k$

Cholesky when $A \succeq 0$.

If $A \succ 0$, simplifications + convenience

$\uparrow$ sym + pos def (eigvals > 0)

$\iff$ $x^T A x \geq 0$, $= 0$ iff $x = 0$

so $A_{1,1} > 0$ ($\because$ take $x = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$)

hence pivoting unnecessary in LU.

furthermore, can take $u_1 = l_1$ by symmetry. AND

$$A - l_1 l_1^T = \begin{bmatrix} 0 & 0 \\ 0 & \boxed{A_1} \end{bmatrix}$$

$\uparrow$ also $A_1 = A_1^T$ and $\underline{A_1 \succ 0}$. $\leftarrow$ not trivial.

(revisit; or consider

$y^T A_1 y = z^T A z$ for some $z = \begin{bmatrix} \alpha \\ y \end{bmatrix}$

+: cost is $\frac{1}{3} n^3$ instead of $\frac{2}{3} n^3$ for $A \neq A^T$.

+: Always stable (revisit)

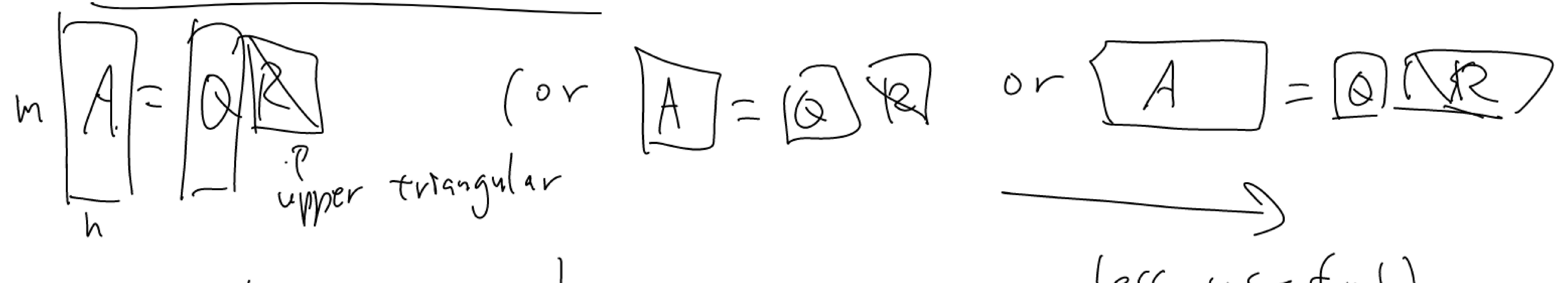
+: no need for pivot.

Recap:

LU: solves $\boxed{A}x=b$

QR: solves $\min_x \|Ax-b\|$ $A\in\mathbb{R}^{m\times n}$, $m>n$
(often $m\gg n$)

QR factorisation



less useful)

Always exists!
&
Always stable

Gram-Schmidt: gives QR!

input: $a_1,\dots a_n$ (colums of A) $A=\begin{bmatrix}a_1\cdots a_n\end{bmatrix}$

output: $q_1,\dots q_n$ (orthonormal)

(and $\begin{bmatrix} r_{11} & \cdots & r_{1n}\\ & \ddots & \vdots\\ & & r_{nn}\end{bmatrix}$)

$q_1 = \frac{a_1}{\|a_1\|}$ (normalise)

$\tilde q_2 = a_2 - q_1 \underbrace{q_1^T a_2}_{r_{12}}$ ($\in \text{span}(a_1,a_2)$, $\tilde q_2 \perp q_1$)

$q_2 = \frac{\tilde q_2}{\|\tilde q_2\|_2}$

$\tilde q_j = a_j - \sum_{i=1}^{j-1} q_i \underbrace{q_i^T a_j}_{r_{ij}}$ ($\tilde q_j \perp q_i$, $i<j$)

$q_j = \frac{\tilde q_j}{\|\tilde q_j\|_2 \leftarrow r_{jj}}$

Then eqn wrt a_j: $a_j = r_{jj} q_j + \sum r_{ij} q_i \iff \begin{bmatrix} A \end{bmatrix} = \begin{bmatrix} q_1 \cdots q_n \end{bmatrix} R.$

Numerically (on computer), better algorm is

Householder QR

Householder reflectors

$H = I - 2vv^T$, $\|v\|_2 = 1$.

- symmetric (obvious)
- orthogonal.

$H_i^{-1} = H_i^T = H_i$

eigendecomposition

$$H = \begin{bmatrix} v & V_\perp \end{bmatrix}\begin{bmatrix} v^T \\ V_\perp^T \end{bmatrix} - 2\begin{bmatrix} v & V_\perp \end{bmatrix}\begin{bmatrix} 1 & & \\ & 0 & \\ & & \ddots & \\ & & & 0 \end{bmatrix}\begin{bmatrix} v^T \\ V_\perp \end{bmatrix}$$

$$= \begin{bmatrix} v & V_\perp \end{bmatrix}\begin{bmatrix} -1 & & \\ & 1 & \\ & & \ddots & \\ & & & 1 \end{bmatrix}\begin{bmatrix} v^T \\ V_\perp^T \end{bmatrix}$$

Lemma: given $u, w \in \mathbb{R}^n$ with $\|u\|_2 = \|w\|_2$,

$\exists$ v s.t. $\|v\|_2 = 1$,

$$Hu = (I - 2vv^T)u = w$$

proof: Let $v = \dfrac{u-w}{\|u-w\|_2}$

Then

$$(I - 2vv^T)u = u - \underbrace{2v(v^Tu)}$$

$$= 2(u-w)\frac{(u-w)^Tu}{\|u-w\|_2^2} = \underbrace{\left(\frac{\|u\|_2^2 - w^Tu}{2\|u\|_2^2 - 2w^Tu}\right)}_{\frac{1}{2}} \cdot 2(u-w)$$

$$= u - (u-w) = w.$$

(explain picture)

Main usage: $Hu = \begin{bmatrix} \|u\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

map u to

(explain Householder QR) "Orthogonal Triangularization"

$$H_1 A = \begin{bmatrix} \|a_1\|_2 & \\ 0 & \\ \vdots & * \\ 0 & \end{bmatrix}$$

$$\underset{\uparrow}{H_2} H_1 A = \begin{bmatrix} * & * \\ 0 & \begin{matrix} * \\ 0 \\ \vdots \\ 0 \end{matrix} \; * \end{bmatrix}$$

$= I - 2 v_2 v_2^T$, $\quad v_2 = \begin{bmatrix} 0 \\ * \\ \vdots \\ * \end{bmatrix}$ $\qquad v_k = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ * \\ \vdots \\ * \end{bmatrix}$) k-1 0's.

Continue: $H_n \cdots H_1 A = \begin{bmatrix} x & \cdots & x \\ 0 & \ddots & x \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix}$

$(H_i^{-1} = H_i^T = H_i)$

$$A = \underbrace{H_1 \cdots H_n}_{Q_F = [Q \;\; Q_\perp]} \begin{bmatrix} R \\ 0 \end{bmatrix}$$

$$= Q_F \begin{bmatrix} R \\ 0 \end{bmatrix} = \underset{\uparrow}{Q} R$$

leading n cols of Q_F.

(Consider Householder QR of $\boxed{A}$ orthonormal.
We get $A = Q\begin{bmatrix} R \\ 0 \end{bmatrix} = [Q \;\; Q_\perp]\begin{bmatrix} I \\ 0 \end{bmatrix} = [A \;\; \underset{\uparrow}{Q_\perp}]\begin{bmatrix} I \\ 0 \end{bmatrix}$

orth complement, proof of existence!

Also useful: Givens rotation (operate locally on matrix)

$$G_1 = \begin{array}{c} \\ i \rightarrow \\ \\ j \rightarrow \\ \\ \end{array} \begin{bmatrix} I & & & & \\ & \overset{i}{\cos\theta} & & \overset{j}{\sin\theta} & \\ & & I & & \\ & -\sin\theta & & \cos\theta & \\ & & & & I \end{bmatrix}$$

GA: rotate (i,j) rows

AG: " cols.

Solving Least-squares problem

$$\min \|\boxed{A}x - b\|_2 \qquad (\text{assume } \operatorname{rank}(A) = n)$$

(– LU isn't useful!)

soln: Take $\boxed{A} = \boxed{Q_F}\begin{bmatrix} R \\ 0 \end{bmatrix}$

$$\|Ax - b\|_2 = \left\|\boxed{Q_F}\begin{bmatrix} R \\ 0 \end{bmatrix}x - b\right\|_2$$

$$= \left\|\begin{bmatrix} R \\ 0 \end{bmatrix}x - Q_F^T b\right\|_2$$

$$= \left\|\begin{bmatrix} R \\ 0 \end{bmatrix}x - \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}\right\|_2$$

minimised when $Rx = b_1$

$x = R^{-1} b_1 = R^{-1} Q^T b$ where

$Q_F = [Q \; Q_\perp]$

So alg:

1. Compute thin QR $\boxed{A} = \boxed{Q}R$

2. Solve $Rx = Q^T b$

(note: Q_F not needed!)

DEMO

Numerical Stability

Often overlooked but v important

Consider

$$Ax = b.$$

Recall example

$$\left(\begin{aligned} A &= U \Sigma V^T \\ &= U \cdot \begin{bmatrix} 1 & \\ & 10^{-15} \end{bmatrix}, \end{aligned} \quad b = A \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \left(x = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right) \right.$$

$$\left. \text{but computed: } \hat{x} = A \backslash b = \begin{bmatrix} 1 \\ 0.9.. \end{bmatrix} \right)$$

with computed solution $\hat{x}$,

$A\hat{x} \neq b.$, but $A\hat{x} \approx b$

Does this imply $x \approx \hat{x}$? Often, NO.

But we Can say quite a bit about $\hat{x}$.

Key notion: $\underbrace{\text{(ill-)conditioning}}_{\text{property of problem}}$ vs $\underbrace{\text{(in)stability}}_{\text{prop. of algorithm}}$

Conditioning: "Derivative"; sensitivity to perturbation.

given task $Y = f(\overset{\uparrow \text{input}}{X})$,

how much does $\tilde{Y} = f(X + \delta X)$ (perturbed input) differ relative to δX?

formally "condition number"

$$\kappa = \lim_{\|\delta x\|_2 \to 0} \sup_{\delta x} \frac{\| f(x) - f(x+\delta x) \|}{\| \delta x \|}$$

A problem is called ill-conditioned if κ is "large".

(usually $\kappa \gg 1$)

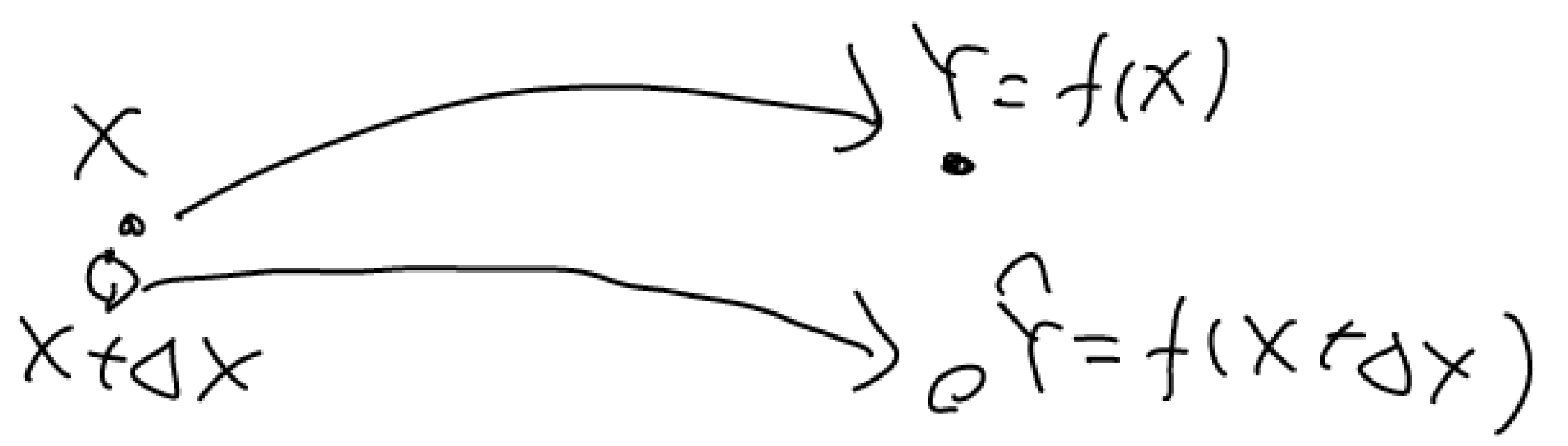
(well-cond if κ "small")

Stability (of Algorithm)

An alg for computing $y = f(x)$ is called <u>backward stable</u> if its output satisfies

$$\hat{y} = f(x+\Delta x)$$

with Δx small. $\left(\dfrac{\|\Delta x\|}{\|x\|} = O(\underset{\substack{\uparrow \\ \text{machine} \\ \text{precision}}}{\varepsilon}) \right)$



"exact soln of a slightly wrong problem"

(e.g. for $Ax = b$, b. stable if with computed $\hat{x}$, $(A+\Delta A)\hat{x} = b + \Delta b$, $\dfrac{\|\Delta A\|}{\|A\|}, \dfrac{\|\Delta b\|}{\|b\|} = O(\varepsilon)$)

(forward stable: $y - \hat{y}$ is "small")

not get in details

NOTE: back stable does NOT mean $y \approx \hat{y}$.

But Back stable $\underset{\sim}{+}$ well-conditioned

$$\Rightarrow y \approx \hat{y}.$$

"argument":

$$\|\hat{y} - y\| = \|f(x+\Delta x) - f(x)\| \lesssim \underbrace{\kappa}_{\substack{\uparrow \\ \text{small} \\ (\text{well-cond}) \\ \text{usually } O(1)}} \cdot \underbrace{\|\Delta x\|}_{\substack{\uparrow \\ \text{small} \\ (\text{back-stable}) \\ O(\varepsilon \cdot \|x\|)}}.$$

κ for $Ax=b$.

slides

Suppose $(A+\Delta A)\hat{x} = b$ (backward stable).

$\|\Delta A\| \le \epsilon \|A\|$

$$\hat{x} = (A+\Delta A)^{-1} b = (A(I+A^{-1}\Delta A))^{-1} b$$
$$= \underline{(I+A^{-1}\Delta A)^{-1}} A^{-1} b$$
$$= I - A^{-1}\Delta A + (A^{-1}\Delta A)^2 + \cdots$$
$$= A^{-1}b - A^{-1}\Delta A A^{-1} b + O((\Delta A)^2)$$
$$= x - A^{-1}\Delta A x + O((\Delta A)^2)$$

$$\frac{\|x-\hat{x}\|}{\|x\|} \lesssim \|A^{-1}\Delta A\|$$
$$\le \|A^{-1}\| \cdot \epsilon \|A\|$$
$$= \kappa_2(A)\cdot \epsilon$$

opening problem: $\kappa_2(A) = 10^{15}$,
highly ill-conditioned.
backward stable, but $\frac{\|x-\hat{x}\|}{\|x\|} \gg \epsilon$.
$A = LU + E$ then $LUx=b$

Remainder: Discuss examples.

Separate Table

		Back stable?	proof?	
	triangular $Lx=b$	✓	✓	
	LU no pivot for $Ax=b$	✗	(✓)	
LU	with pivot	✓ in practice	✗	← missing!
	Cholesky $A \succ 0$ $A=R^TR$	✓	✓	
	Gram-Schmidt	✗	✓	
QR	Householder QR	✓	✓	← please know
	$\min\|Ax-b\|_2$ via House-QR	✓	✓	
	via $A^TAx=A^Tb$	✗	✓	
	Matrix mult.	✗	(✗)	
	$Ax=\lambda x$ via QR alg	✓	✓	← coming

Examples

(i) triangular systems : stable.

$Lx = b$ via (back) substitution.

computed $\hat{x}$ satisfies $(L+\delta L)\hat{x} = b$ $\|\delta L\| \leq \|L\|$.

(again, not accurate soln.)

(ii) LU (w/o) pivots for $Ax = b$.

Suppose $LU = A + \delta A$

but $\|L\| \, \|U\| \gg \|A\|$. Then by (i),

$$(L+\delta L)(U+\delta U)\hat{x} = b.$$

$$\iff (A+\widetilde{\Delta A})\hat{x} = b.$$

$$\widetilde{\Delta A} = \delta L U + L \delta U + \delta L \delta U,$$

$$\frac{\|\delta L\|}{\|L\|}, \frac{\|\delta U\|}{\|U\|} = O(\varepsilon).$$

(nice if each term $O(\varepsilon\|A\|)$, but

$$\|\delta L U\| \leq \|\delta L\| \cdot \|U\| = \varepsilon \|L\| \cdot \|U\| \gg \varepsilon \|A\|.$$

so NOT stable (e.g. when $\|U\| \gg 1$)

But stable if $\|L\| \cdot \|U\| \simeq \|A\|$ (note $\|A\| \simeq \|LU\| \leq \|L\| \cdot \|U\|$)

Depending on time, skip writing rest

(iii) Cholesky $A = R^T R \succeq 0$.

In this case, $R^T R = A + \Delta A$ implies $\|R^T\| \cdot \|R\| \simeq \|A\|$ immediately,

hence stable.

(iv) Gram-Schmidt: Unstable

(computed $\hat{Q}$: $\kappa_2(\hat{Q}) \simeq 1+\epsilon\kappa_2(A)^2$)

(modified G-S better, not perfect

(v) Matmul <u>NOT</u> stable.

Need $fl\left(\boxed{A}\,\boxed{B}\right) = (A+\Delta A)(B+\Delta B)$ (consider $\boxed{A}\,\boxed{B}$)

exceptions when A or B is <u>orthogonal</u>.

what IS true:

$fl(AB) = AB + \epsilon\cdot\|A\|\cdot\|B\|$

if $A = Q$ orthogonal,

$$fl(QB) = QB + \epsilon\cdot\|B\|$$
$$= Q(B+\Delta B), \quad \Delta B = Q^T\cdot\epsilon\|B\|$$

(vi) Householder QR.

recall $A = QR$ via $\underbrace{H_n \cdots H_1}_{\text{orthogonal}} A = R$.

perfect QR fact: $A \simeq \hat{Q}\hat{R}$,

$\|A-\hat{Q}\hat{R}\|/\|A\| = O(\epsilon)$,

$\|\hat{Q}^T\hat{Q} - I\| = O(\epsilon)$.

NOT
$\|Q-\hat{Q}\| = O(\epsilon)$
$\|R-\hat{R}\| = O(\epsilon)$

"orthogonal linear Algebra is stable"

Next: QR algorithm for $Ax = \lambda x$

(culmination of orthogonal LA)

Lect 9. Eigenvalue Problems.

$$n\underset{n}{\boxed{A}}x = \lambda x$$

- given A, find λ, $x \neq 0$.
- impossible to solve exactly if $n \geq 5$

proof/appl. of $Ax = \lambda x$: Companion matrix

for $p(x) = \sum_{k=0}^{n} c_k x^k$,

$$p(\lambda) = 0 \iff \lambda \in \overset{\text{set of eigvals}}{\text{eig}(C)}\text{, where}$$

$$C = \begin{bmatrix} -\frac{c_{k-1}}{c_k} & \cdots & \cdots & -\frac{c_0}{c_k} \\ 1 & & & 0 \\ & \ddots & 0 & \vdots \\ 0 & & \ddots & \vdots \\ & & 1 & 0 \end{bmatrix}$$

(proof: verify $\begin{bmatrix} \lambda^{k-1} \\ \vdots \\ \lambda \\ 1 \end{bmatrix}$ is eigvec)

(poly roots cannot be computed if deg ≥ 5)

hence by Abel's Thm, impossible. ▨

Hence, algs are iterative/ approximate.

NOTE: eigvals(A) via roots$(\det(\lambda I - A))$ is bad idea!

much better is converse: roots$(p(x))$ via eigvals(C)

Usual goal of eig(A): <u>Schur form</u>

Thm: $A \in \mathbb{C}^{n\times n}$, $\exists U$ (unitary) s.t.

$$A = U \underset{\sim}{T} U^*$$

◁ upper triang.

here, $\mathbb{C}^{n\times n}$ rather than $\mathbb{R}^{n\times n}$. "Real Schur" where U real, T 2×2 up-tri

NOTE: 1. $\text{eig}(A) = \text{eig}(XAX^{-1})$ $\forall X$ nonsingular.

so $= \text{eig}(T) = \text{diag}(T)$.

similarity transformation

Orthogonal Linear algebra

2. If $AA^* = A^*A$ (normal), then
$TT^* = T^*T \Rightarrow T$ diagonal. (via equating diag. entries)

proof: Recall $\det(\lambda I - A) = 0 \Rightarrow \exists v_{\neq 0}$ s.t. $(\lambda I - A)v = 0$.

Suppose $Av = \lambda_1 v$. Take $U_1 = \underset{\sim}{[v \ V_\perp]}$ $n\times n$ unitary

Then
$$A\overbrace{[v \ V_\perp]}^{U_1} = [\lambda_1 v \ \ *],$$

$$[v \ V_\perp]^* \ \ // \ = \begin{bmatrix} \lambda_1 & * \\ 0 & \\ \vdots & A_2 \\ 0 & \end{bmatrix} =: \tilde{A}_2 \quad \text{(as } U_1 \text{ unitary)}$$

Next find $A_2 v_2 = \lambda_2 v_2$ and $\tilde{U}_2 = [v_2 \ V_{2\perp}]$

$(n-1)\times(n-1)$ unitary.

$$\begin{bmatrix} 1 & \\ & \tilde{U}_2 \end{bmatrix}^* \tilde{A}_2 \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & \tilde{U}_2 \end{bmatrix}}_{U_2} = \begin{bmatrix} \lambda_1 & \times & * \\ & \lambda_2 & * \\ 0 & 0 & A_3 \end{bmatrix}$$

continue $n-1$ times to get

$$\underbrace{U_{n-1}^* \cdots U_1^*}_{U^*} A \underbrace{U_1 \cdots U_{n-1}}_{U} = \begin{bmatrix} \lambda_1 & & * \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} =: T.$$

Similar to Householder QR,
- keep multiply orth.
- needs to be both sides.

(Recap: Mat decomp.)

Power method (V important!)

V simple

(for computing one eigenpair)

Let x: arbitrary vector.

Algorithm: $x := \frac{Ax}{\|Ax\|_2}$, repeat.

note! (after k iters, $x_k = \frac{A^k x_0}{\|A^k x_0\|_2}$)

$(\lambda = x^* A x)$

Claim: $(\lambda, x) \to$ (dominant) eigenpair under mild conditions.

"proof": suppose $|\underbrace{\lambda_1}_{\text{dominant eigval}}| > |\lambda_i| \ \forall i \neq 1$

$$A = V \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} V^{-1}$$

V ← not orthogonal.

(diagonalisable).

More generally, extend to when $A = X \begin{bmatrix} J_1 & & \\ & \ddots & \\ & & J_n \end{bmatrix} X^{-1}$

Then $x = \sum_{i=1}^{n} c_i v_i$ (via $\begin{bmatrix} v_1 \cdots v_n \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix} = x$

$\Leftrightarrow \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix} = V^{-1} x$.

$$A^k x_0 = \sum_{i=1}^{n} c_i A v_i$$

$$= \sum_{i=1}^{n} (\lambda_i^k c_i) v_i$$

so

$$x_k = \left((\lambda_1^k c_1) v_1 + (\lambda_2^k c_2) v_2 + \cdots + (\lambda_n^k c_n) v_n \right) / \|A^k x_0\|_2$$

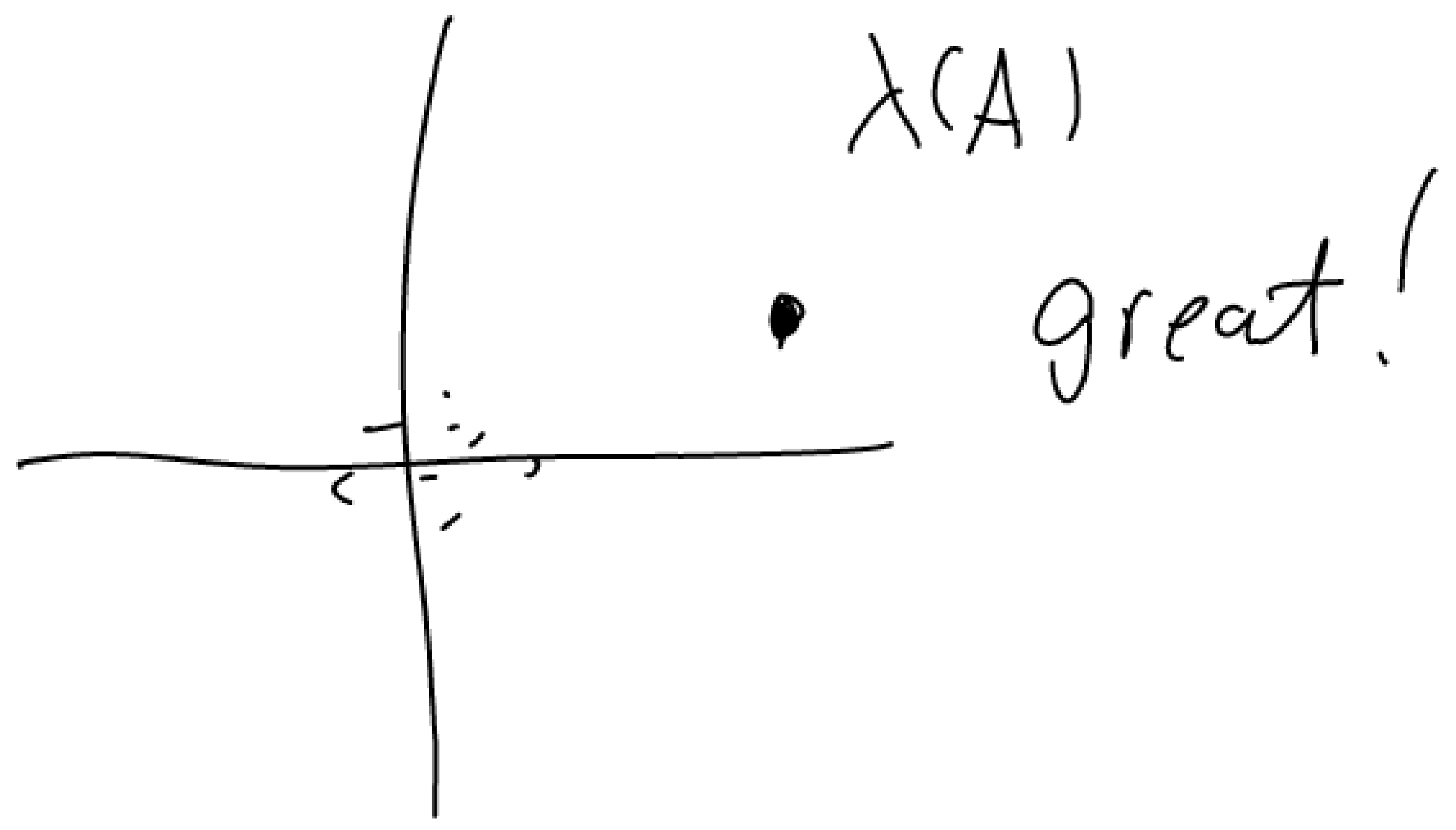
$$= C \left(v_1 + \frac{c_2}{c_1} \cdot \underbrace{\left(\frac{\lambda_2}{\lambda_1}\right)^k}_{\downarrow \atop 0} v_2 + \cdots + \frac{c_k}{c_1} \underbrace{\left(\frac{\lambda_k}{\lambda_1}\right)^k}_{\downarrow \atop 0} v_k + \cdots \right)$$

so $x_k \to (\pm) v_1$, assuming $c_1 \neq 0$, (generic assumption)

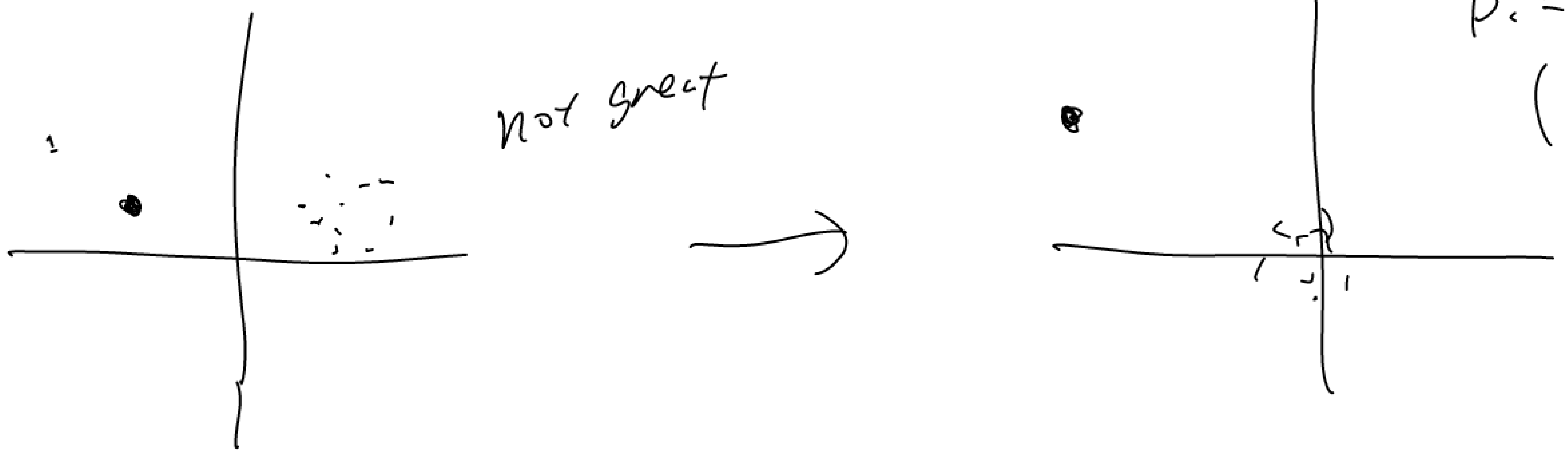
dominant eigvec. Then $\lambda = x_k^* A x_k \to \lambda_1$

Convergence speed $\sim \left| \frac{\lambda_2}{\lambda_1} \right|^k$

we need $|\lambda_1| > |\lambda_2|$, but not diagonalisable power method converges like $\left|\frac{\lambda_2}{\lambda_1}\right|^k$.

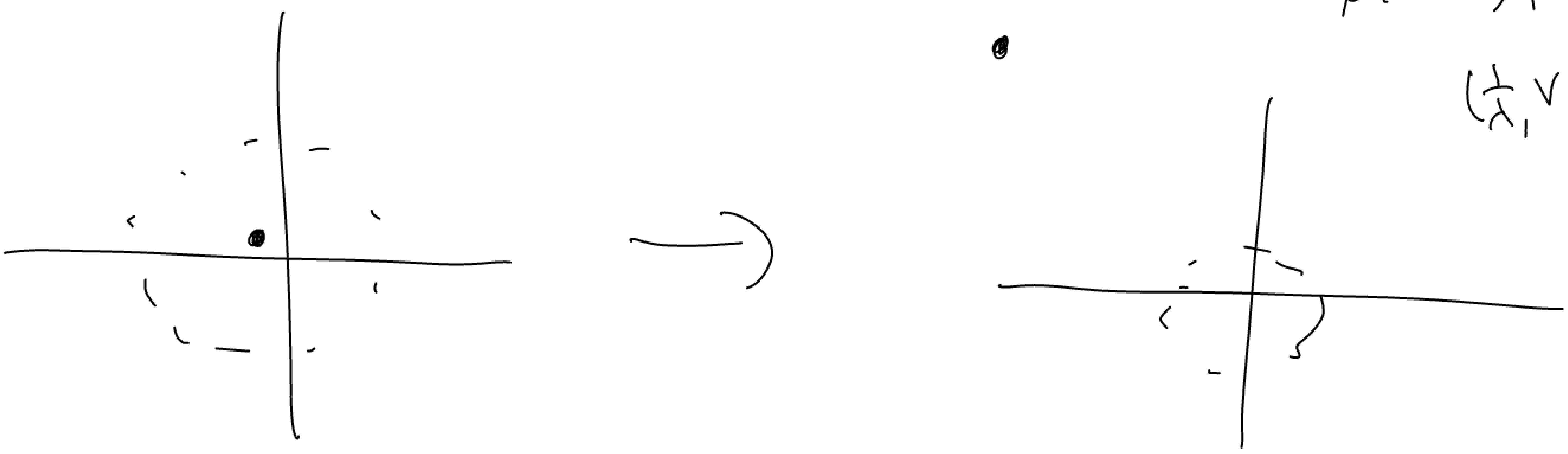


$A: (\lambda, v)$



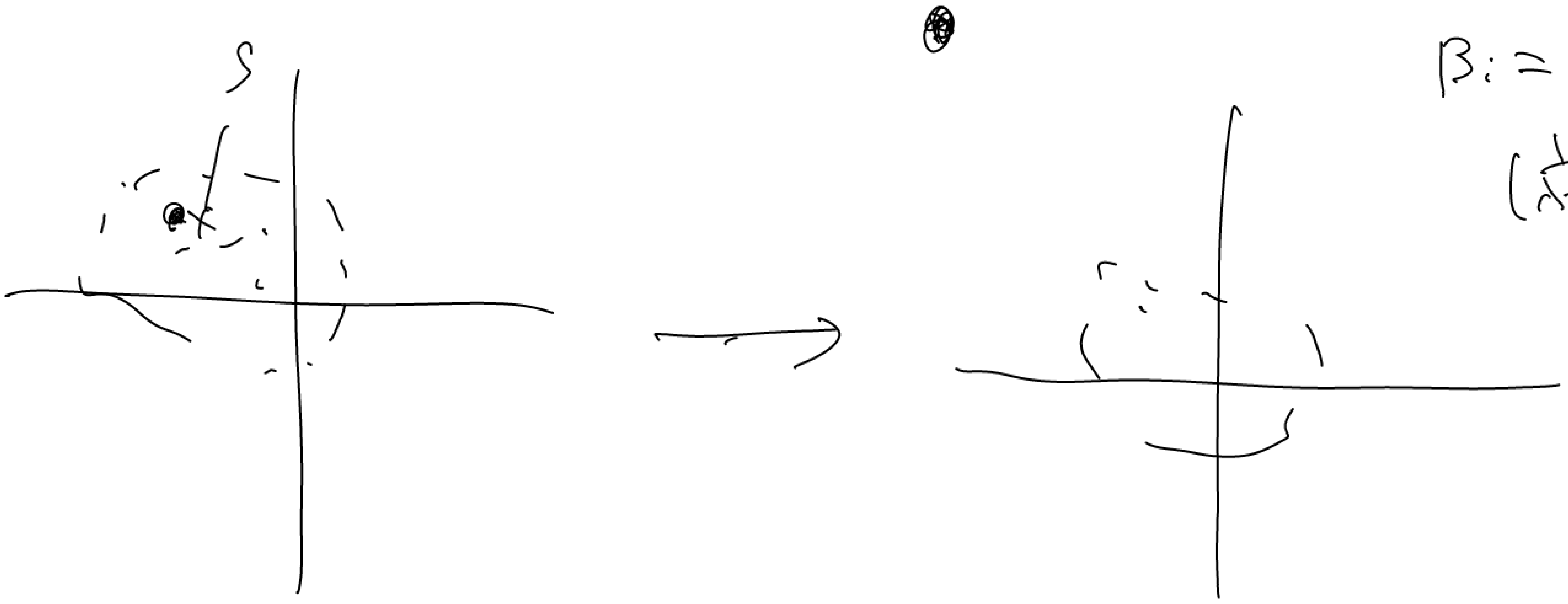
$B := A - sI$

$(\lambda - s, v)$



$B := A^{-1}$

$(\frac{1}{\lambda}, v)$



$B := (A - sI)^{-1}$

$(\frac{1}{\lambda - s}, v)$

(Shifted) Inverse power method

power method applied to $B := (A - \underset{\underset{\in \mathbb{C}}{\uparrow}}{\mu} I)^{-1}$

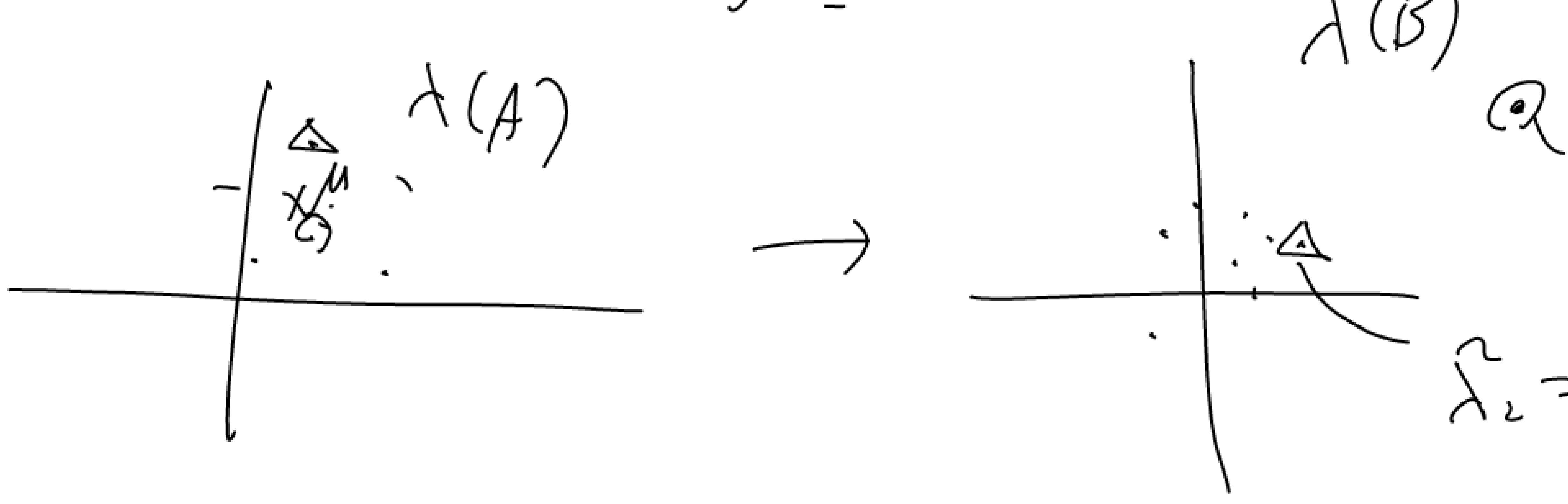
Q: eigvals $\tilde{\lambda}$ of B?

Suppose $Av = \lambda v$

$$(A - \mu I)v = (\lambda - \mu)v$$

$$\frac{1}{\lambda - \mu} v = (A - \mu I)^{-1} v = Bv$$

so $\tilde{\lambda} = \frac{1}{\lambda - \mu}$.



$\tilde{\lambda}_1 = \frac{1}{\lambda_{\sigma(1)} - \mu}$

$\tilde{\lambda}_2 = \frac{1}{\lambda_{\sigma(2)} - \mu}$

convergence speed is

$$\left|\frac{\tilde{\lambda}_2}{\tilde{\lambda}_1}\right| = \left|\frac{\lambda_{\sigma(1)} - \mu}{\lambda_{\underset{\underset{\text{permutation}}{\uparrow}}{\sigma(2)}} - \mu}\right|, \quad \approx 0 \text{ if } \lambda_{\sigma(1)} - \mu \approx 0 \text{ (!)}$$

Taking $\mu = \underbrace{\lambda_k = x_k^T A x_k}_{\text{change every iteration}} \to \lambda$ with $\underline{\text{quadratic accuracy}}$

$(\text{err}_k = O(\text{err}_{k-1}^2))$

Now, QR algorithm for $Ax = \lambda x$

(find All eigenvalues).
(+ eigvecs)

(slide)

Recall needs be iterative

Basic alg:

$$A = A_1 = Q_1 R_1, \quad \text{(QR fact)}$$

$$A_2 = R_1 Q_1 \quad \text{(mat-mul)}$$

$$= Q_2 R_2 \quad \text{(QR fact)}$$

Repeat: $A_k = R_{k-1} Q_{k-1}$ $\left(= Q_{k-1}^T Q_{k-1} R_{k-1} Q_{k-1} = Q_{k-1}^T A_{k-1} Q_{k-1}\right)$

$$= Q_k R_k$$

- orthogonal LA. ⇒ stable

(proof)
(Banks)

properties

1. $A_{k+1} = Q^{(k)T} A Q^{(k)}$
2. $A^k = (Q_1 \cdots Q_k)(R_k \cdots R_1) =: Q^{(k)} R^{(k)}$,

proof: 1: $A_{k+1} = Q_k^T A_k Q_k$, induction.

2: suppose $A^{(k-1)} = Q^{(k-1)} R^{(k-1)}$,

$$Q^{(k-1)T} A Q^{(k-1)} = A_k = Q_k R_k$$

$$\Longleftrightarrow A = Q^{(k-1)} Q_k R_k Q^{(k-1)T}$$

so $A \cdot A^{(k-1)} = Q^{(k-1)} Q_k \cdot R_k \cdot R^{(k-1)}$ $\square$

Understanding QR alg:

$A^k = Q^{(k)}R^{(k)}$ by property 2.

consider $A^k \begin{bmatrix}1\\0\\\vdots\\0\end{bmatrix} = Q^{(k)} \underbrace{R^{(k)} \begin{bmatrix}1\\0\\\vdots\\0\end{bmatrix}} = \underset{\sim}{c} \cdot q_1^{(k)}$

scalar, value does not matter

1st col of $Q^{(k)}$.

$$\nabla \begin{bmatrix}1\\0\\0\\\vdots\\0\end{bmatrix} = \begin{bmatrix}*\\0\\\vdots\\0\end{bmatrix}$$

now, by theory of power method

$c \cdot q_1^{(k)} = A^k \begin{bmatrix}1\\0\\\vdots\\0\end{bmatrix} \to \underset{\sim}{v_1}$ (dominant eigvec) as $k \to \infty$.

so $A_k = Q^{(k)T} A Q^{(k)}$

$\to \begin{bmatrix}\lambda_1 & * \\ 0 & *\end{bmatrix}$ as $k \to \infty$. (conv. speed $\sim \left|\frac{\lambda_2}{\lambda_1}\right|$)

More good news:

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

invert: $A^{-k} = R^{-(k)} Q^{(k)T}$

transpose: $(A^{-k})^T = Q^{(k)} \underset{\Delta}{R^{-(k)T}}$

$$(A^{-k})^T \begin{bmatrix}0\\\vdots\\0\\1\end{bmatrix} = Q^{(k)} \underbrace{\Delta \begin{bmatrix}0\\\vdots\\0\\1\end{bmatrix}}_{= \begin{bmatrix}0\\\vdots\\0\\*\end{bmatrix}} = \tilde{c} \cdot \underset{\sim}{q_n^{(k)}}$$

last col of $Q^{(k)}$

so $q_n^{(k)} \to v_n$ w/ conv speed $\sim \left|\frac{\lambda_n}{\lambda_{n-1}}\right|$.

so

$A_k = Q^{(k)T} A Q^{(k)}$

$\to \begin{bmatrix}\lambda_1 & * & \\ 0 & & \\ & 0 & \lambda_n\end{bmatrix}$ as $k \to \infty$.

This can be made even better by shifting

$$A_k - s_k I = Q_k R_k \qquad (s_k \in \mathbb{C})$$

$$A_{k+1} = R_k Q_k + s_k I .$$

If $s_k \sim \lambda$, then $q_n^{(k)} \to x_n$ w/ speed $\left| \frac{\lambda_{\sigma(1)} - s}{\lambda_{\sigma(2)} - s} \right| \approx 0$

QR alg performs shift-invert power method without inverting matrices.

amazing!

	# iter	flops / $\overset{QR}{\text{iter}}$	overall
QR alg	$O(10000n)$	$O(n^3)$	$O(10000n^3)$
QRalg + shift	$O(n)$	$O(n^3)$	$O(n^4)$
QRalg + shift + Hessenberg reduc.	$O(n)$	$O(n^2)$	$O(n^3)$ $\sim 20n^3$ ($\frac{4}{3}n^3$ if $A=A^T$)

preprocessing for practical speed (slides)

not as beautiful

$$A = \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{bmatrix}$$

$$H_1 A = \begin{bmatrix} x & \\ x & \\ 0 & X \\ 0 & \end{bmatrix}$$

$= I - 2v_1v_1^T$

$v_1 = \begin{bmatrix} 0 \\ x \end{bmatrix}$

← diff from Householder QR fact (need to keep mat. similar)

$$H_1 A H_1 = \begin{bmatrix} x & x & \\ x & x & X \\ 0 & x & \\ 0 & x & \end{bmatrix}$$

$$H_2 H_1 A H_1 = \begin{bmatrix} x & x & \\ x & x & X \\ & x & \\ & 0 & x\ x \end{bmatrix}$$

$I - 2v_2v_2^T$, $v_2 = \begin{bmatrix} 0 \\ 0 \\ x \\ x \end{bmatrix}$

$$H_2 H_1 A H_1 H_2 = \begin{bmatrix} x & x & x & \\ x & x & x & \\ & x & x & \\ 0 & & x & \end{bmatrix}$$ (upper Hessenberg).

so overall, each QRiter is $O(n^2)$ flops!

Lemma. if A_1 Hessenberg, $A_1 = QR$ $A_2 = RQ$ is also Hessenberg.

"proof":

$$A_1 = \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ & x & x & x \\ & & x & x \end{bmatrix}$$

$$G_1 A_1 = \begin{bmatrix} x & x & x & x \\ 0 & x & x & x \\ & x & x & x \\ & & x & x \end{bmatrix}$$

$$G_i = \begin{bmatrix} I & & \\ & c_i \ s_i & \\ & -s_i \ c_i & \\ & & I \end{bmatrix}$$

$$\underbrace{G_3 G_2 G_1}_{Q^T} A_1 = \begin{bmatrix} x & x & x & x \\ & x & x & x \\ & & x & x \\ & & & x \end{bmatrix} = R.$$

so $A_2 = RQ = R\,G_1^T G_2^T G_3^T$

$$= \begin{bmatrix} x & x & x & x \\ & x & x & x \\ & & x & x \\ & & & x \end{bmatrix} \begin{bmatrix} c & -s & & \\ s & c & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & c & -s & \\ & s & c & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \hat{c} & -\hat{s} \\ & & \hat{s} & \hat{c} \end{bmatrix}$$

$$= \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ & & x & x \\ & & & x \end{bmatrix} G_2^T G_3^T = \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ & x & x & x \\ & & & x \end{bmatrix} G_3^T = \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ & x & x & x \\ & & x & x \end{bmatrix}.$$

Computing SVD of $A \in \mathbb{R}^{m \times n}$ $(m \geq n)$

Goal: find U, V (orthogonal) s.t.

$$U^* A V = \begin{bmatrix} \Sigma \\ 0 \end{bmatrix}$$

(note U, V not need correlated)

Alg: first (anal. to Hessenberg reduction)

Householder refl.

$$H_1 A = \begin{bmatrix} x & x \\ 0 & x \\ \vdots & \\ 0 & \end{bmatrix}$$

(as in QR)

$$H_1 A \tilde{H}_1 = \begin{bmatrix} x & x & 0 \cdots 0 \\ 0 & & \\ \vdots & & X \\ 0 & & \end{bmatrix}$$

$$= (I - 2\tilde{v}_1 \tilde{v}_1^T), \quad \tilde{v}_1 = \begin{bmatrix} 0 \\ x \end{bmatrix}$$

$$\vdots$$

$$H_n \cdots H_1 A \hat{H}_1 \cdots \hat{H}_{n-1} = \begin{bmatrix} x & x & & & \\ & x & x & & \\ & & x & x & \\ & & & \ddots & x \\ & & & & x \\ & 0 & & & \end{bmatrix} =: B$$

Then "find eigdecomp of $B^T B$ via QR alg"

(D-C if interested).

QZ, other eigen problems

Iterative Methods

1. SVD
2. $Ax=b$ < (stability)
3. $Ax=\lambda x$
4. Iterative
5. Randomised

(slides x 2)

Goal: gradually improve solution, each iteration requiring $\ll n^3$ flops.

Key idea: work with subspaces

1. find a "good" subspace $S=\mathrm{span}(Q)$
2. find solution $x \in S$, e.g. for $Ax=b$, $\mathrm{span}(\boxed{Q})=\mathrm{span}(S)$ and solve $\min \|A\underbrace{Qy}_{x_k}-b\|_2$

How to do 1? (e.g. for $Ax=b$). $b, Ab, A^2b, \dots$ "only way"

Generate subspace from ${}_n\underset{n}{\boxed{A}}$, b

(i) random (?)

(ii) $[b, Ab, A^2b, \dots]$

(iii) $A=L+D+U$, $[b, Lb, \dots]$??

We'll focus on Krylov subspace (ii)

$$\underbrace{S}_{K_k(A,b):} = \mathrm{span}(b, Ab, A^2b, \dots, A^{k-1}b)$$

(usually) k-dimensional subspace

Note: $x_k \in S \iff x_k = \sum_{i=0}^{k-1} c_i A^i b$

$= P_{k-1}(A)b$, P_{k-1}: polynomial degree $\leq k-1$.

That is: Krylov subspace methods: find approx. solution (for e.g. $Ax=b$, $Ax=\lambda x$) of form $x \approx x_k = P_{k-1}(A)b$

Analysis reduces to polynomial approximation theory

e.g. $x=A^{-1}b \simeq P_{k-1}(A)b$, i.e. $z^{-1} \simeq p_k(z)$ on $z \in \mathrm{eig}(A)$

$\Longleftrightarrow$ find approximate soln

$$x \in \mathcal{K}_k(A,b) = \text{span}\{\underbrace{b, Ab, \dots, A^{k-1}b}_{\text{ill-conditioned matrix,}}\}$$

(note: $A^{k-1}b \to x_{\max}$), by power method

We want a good (well-conditioned) basis for $K_k(A,b)$.

Use $\boxed{\text{Arnoldi}}$ process

"naive method for Q_k s.t. span$[b, Ab, \cdots A^{k-1}b]$ = span(Q_k)

1. Compute "

2. Compute QR. ← But V ill-conditioned! ($A^k b \to$ dom. eigvec)

Arnoldi: orthogonalize each step, A-mult to final vector.

mult - mult - ⋯ - mult → orth
vs
mult - orth - mult - orth - ⋯

(i) $q_1 = \frac{b}{\|b\|_2}$ (same as naive)

(ii) $\tilde{q}_2 = Aq_1 - q_1 \underbrace{(q_1^T A q_1)}_{h_{1,1}}$,

$q_2 = \frac{\tilde{q}_2}{\underbrace{\|q_2\|_2}_{= h_{2,1}}}$ (more or less same)

note $q_2 \in$ span(b, Ab)

(iii) $\tilde{q}_3 = \underbrace{Aq_2} - q_1(q_1^T A q_2) - q_2(q_2^T A q_2)$

(Aq_2 not $A^2 b$!!)

$q_3 = \frac{\tilde{q}_3}{\|\tilde{q}_3\|_2}$ $\in$ Span(b, Ab, A^2b)

but span$[q_1, q_2, Aq_2]$ = span$[q_1, q_2, q_3)$ = span$[b, Ab, A^2b]$

(k) $\tilde{q}_k = Aq_{k-1} - \sum_{j=1}^{k-1} q_j \underbrace{(q_j^T A q_{k-1})}_{h_{j,k-1}}$

$q_k = \frac{\tilde{q}_k}{\|q_k\|_2 - h_{k+1,k}}$

This yields Arnoldi decomposition $AQ_k = Q_k H_k + q_{k+1} \hat{e}_k$
$= Q_{k+1} H_{k+1}$

$$A[q_1 \cdots q_k] = AQ_k = [q_1 \cdots q_k] \underbrace{\begin{bmatrix} h_{11} & & \\ h_{21} & H_{ij} & \\ 0 & \ddots & \\ \vdots & & \\ 0 & 0 & \times \end{bmatrix}}_{(k+1) \times k} + q_{k+1}[0 \cdots 0 \; h_{k+1,k}]$$

Thm: span$(q_1 \cdots q_k)$ = span$(b, Ab, \cdots A^{k-1}b)$. $\forall k$

(assumption: $h_{k+1,k} \neq 0$) ← if not, we're in fact happy proof: Induction on k. $k=1$ trivial, and

$q_{k-1} = p_{k-1}(A)b$

$\exists$ poly. degree exactly $k-1$.

($p_{k-1}(z) = \sum_{i=0}^{k-1} c_i z^i$, $c_{k-1} \neq 0$.

Now

$$q_k = \frac{1}{h_{k+1,k}}\Big(\underbrace{\underbrace{Aq_{k-1}}_{A\cdot p_{k-1}(A)b}}_{\text{degree exactly } k} - \underbrace{\underbrace{\sum_{i=1}^{k} h_{i,k}\, q_k}_{\tilde p_{k-1}(A)\cdot b}}_{\text{degree} \le k-1}\Big)$$

so $q_k = \underbrace{p_k(A)b}_{\deg = k}$ and $\mathrm{span}(q_1 \cdots q_k) = \mathrm{span}(p_1(A)b \cdots p_{k-1}(A)b)$

$= \mathrm{span}(b, Ab, \dots A^{k-1}b)$.

The end product is <u>Arnoldi decomposition</u>

$$\underset{n\times k}{A\,Q_k} = \underset{n\times(k+1)}{Q_{k+1}}\,\underset{\substack{(k+1)\times k\\ \text{Hesseberg.}}}{\tilde H_k} \qquad \left(= Q_k H_k + q_{k+1} h_{k+1,k} [0, \cdots 0, 1]\right)$$

even "better" than $[b, Ab, \cdots, A^{k-1}b] = Q_k \hat R_k$ (next)

NOTE:

– HMT, discuss intuition.

$$\begin{bmatrix}\sigma_1 & & \\ & \sigma_2 & \\ & & \ddots\end{bmatrix}\begin{bmatrix} Q \end{bmatrix} = \begin{bmatrix}\sigma_1 q_1^T \\ \sigma_2 q_2^T \\ \vdots \end{bmatrix}$$

Q: $\begin{bmatrix} q_1^T \\ \vdots \\ q_n^T\end{bmatrix}$ or Gaussian

(if $\sigma_1 \approx \sigma_2 \approx \cdots \approx \sigma_r \gg \sigma_{r+1} \dots$

$\approx \begin{bmatrix}\boxed{h_m} \\ 0\end{bmatrix} \to |Q|^2$

GMRES for $Ax=b$

massively successful alg, most cited in 90s

find "best" soln in $x_k \in \text{span}(Q_k)$
$= \mathcal{K}_k(A,b)$

i.e. $x_k = \underset{x \in Q_k}{\operatorname{argmin}} \|Ax-b\|_2$

now

$$\min_{x \in Q_k} \|Ax-b\|_2 = \min_y \|AQ_k y - b\|_2$$

$$= \min_y \|Q_{k+1}\tilde{H}_k y - b\|_2$$

Left-mult $\begin{bmatrix} Q_{k+1}^T \\ Q_\perp^T \end{bmatrix}$ orthogonal.

$$= \min_y \left\| \begin{bmatrix} \tilde{H}_k \\ 0 \end{bmatrix} y - \underbrace{\begin{bmatrix} Q_{k+1}^T b \\ Q_{k\perp}^T \end{bmatrix}}_{= \begin{bmatrix} \|b\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}} \right\|_2$$

soln: $\min_y \left\| \tilde{H}_k y - \begin{bmatrix} \|b\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \right\|_2$

$\uparrow$ $(k+1)\times k$ Hessenberg L-S.

Recall

QR of Hess: via k Givens rotations

e.g. $$G_2\left(G_1 \underset{\overset{\parallel}{\tilde{H}_k}}{\begin{bmatrix} x & x & x \\ x & x & x \\ & x & x \\ & & x \\ & & x \end{bmatrix}}\right) = G_3\left(G_2 \begin{bmatrix} x & x & x \\ 0 & x & x \\ & x & x \\ & & x \end{bmatrix}\right) = G_3 \begin{bmatrix} x & x & x \\ & x & x \\ & & x \\ & & x \end{bmatrix} = \begin{bmatrix} x & x & x \\ & x & x \\ & & x \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix}$$

so $\tilde{H}_k = G_1^T G_2^T G_3^T \begin{bmatrix} R \\ 0 \end{bmatrix}$.

For 2022 lect 13

Recall GMRES

$$x_k = \underset{x \in K_k(A,b)}{\arg\min} \|Ax - b\|_2$$

Arnoldi

$$AQ_k = Q_{k+1}\tilde{H}_k$$

$$\min_{x \in Q_k} \|Ax - b\|_2 = \min_y \|AQ_k y - b\|_2$$

$$= \min_y \|Q_{k+1}\tilde{H}_k y - b\|_2$$

Left-mult $\begin{bmatrix} Q_{k+1}^T \\ Q_\perp^T \end{bmatrix}$ orthogonal.

$$= \min_y \left\| \begin{bmatrix} \tilde{H}_k \\ 0 \end{bmatrix} y - \underbrace{\begin{bmatrix} Q_{k+1}^T b \\ Q_{k\perp}^T \end{bmatrix}} \right\|_2$$

$$\begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \end{bmatrix} = \begin{bmatrix} \|b\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \leftarrow \text{as } q_1 = \frac{b}{\|b\|_2}$$

(forgot last time!)

$$= \min_y \left\| \tilde{H}_k y - \begin{bmatrix} \|b\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \right\|_2$$

note: $= 0$ if $h_{k+1,k} = 0$!

$\min \|Ax - b\|_2 = 0$, exact soln

GMRES convergence

$x_k \in K_k(A,b) \iff x_k = P_{k-1}(A)b.$ $\quad P_{k-1} \in \mathcal{P}_{k-1}$ (poly degree $\le k-1$)

$$\min_{x_k = P_{k-1}(A)b} \|Ax_k - b\|_2 = \min_{\deg(P_{k-1}) \le k-1} \|A\cdot P_{k-1}(A)b - b\|_2$$

$$= \min_{\substack{\tilde{p}_k(0)=0 \\ \deg(\tilde{p}_k) \le k}} \|(\tilde{p}_k(A) - I)b\|_2 \qquad (\tilde{p}(z) = z \cdot p_{k-1}(z))$$

$$= \min_{\substack{\deg(P) \le k \\ P(0)=1}} \|P(A)b\|_2 \qquad (P(z) = 1 - \tilde{p}_k(z))$$

polynomial optimisation

Now assume A diagonalisable $A = X\Lambda X^{-1}$

then $P(A) = X p(\Lambda) X^{-1}$

$$= X \begin{pmatrix} p(\lambda_1) & & \\ & \ddots & \\ & & p(\lambda_n) \end{pmatrix} X^{-1}$$

$$\|P(A)\|_2 = \|X p(\Lambda) X^{-1}\|_2$$

$$\le \|X\|_2 \cdot \|X^{-1}\|_2 \cdot \|p(\Lambda)\|_2$$

$$= \kappa_2(X) \cdot \underline{\max_{z \in \lambda(A)} |p(z)|}$$

i.e. GMRES finds p s.t. (up to $\kappa(X)$)

reduced to scalar problem.

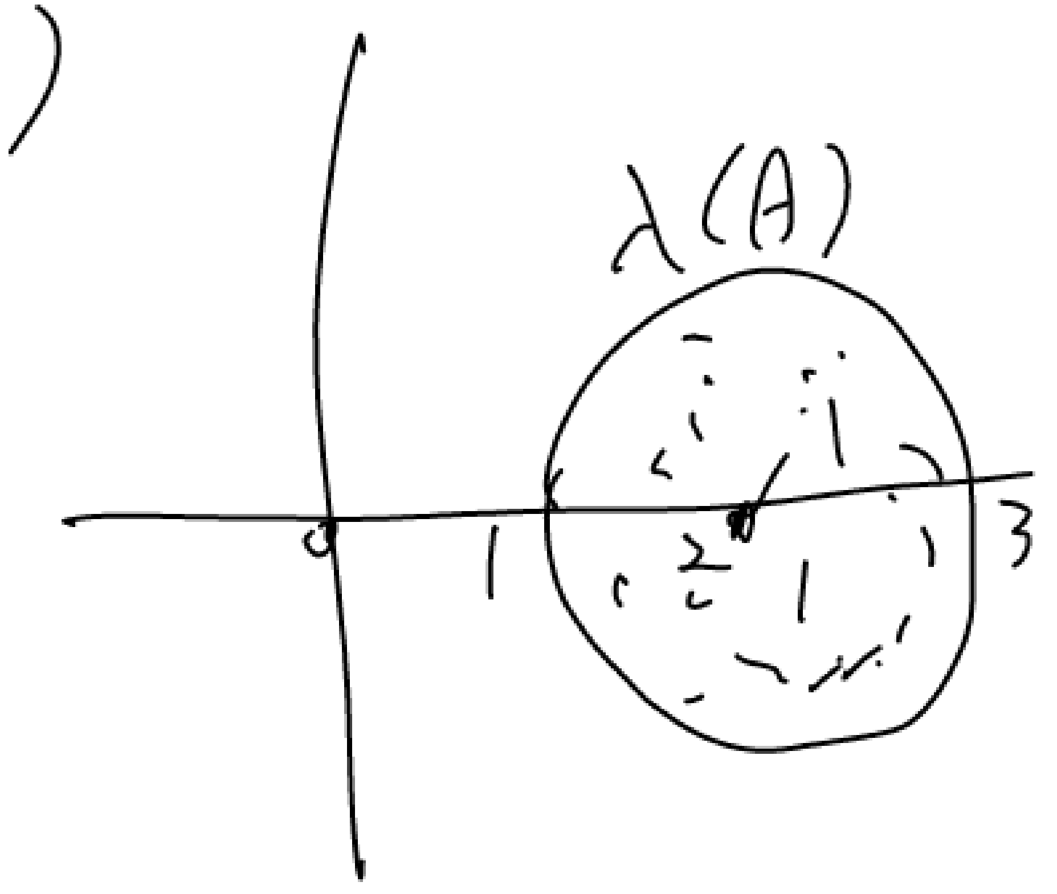
$$\min_{\substack{\deg(p)=k \\ p(0)=1}} \max_i |p(\lambda_i)| :$$

$|p(0)| = 1$ and $|p(\lambda_i)|$ small.

"small on eigvals"

when does GMRES converge rapidly?

(i)

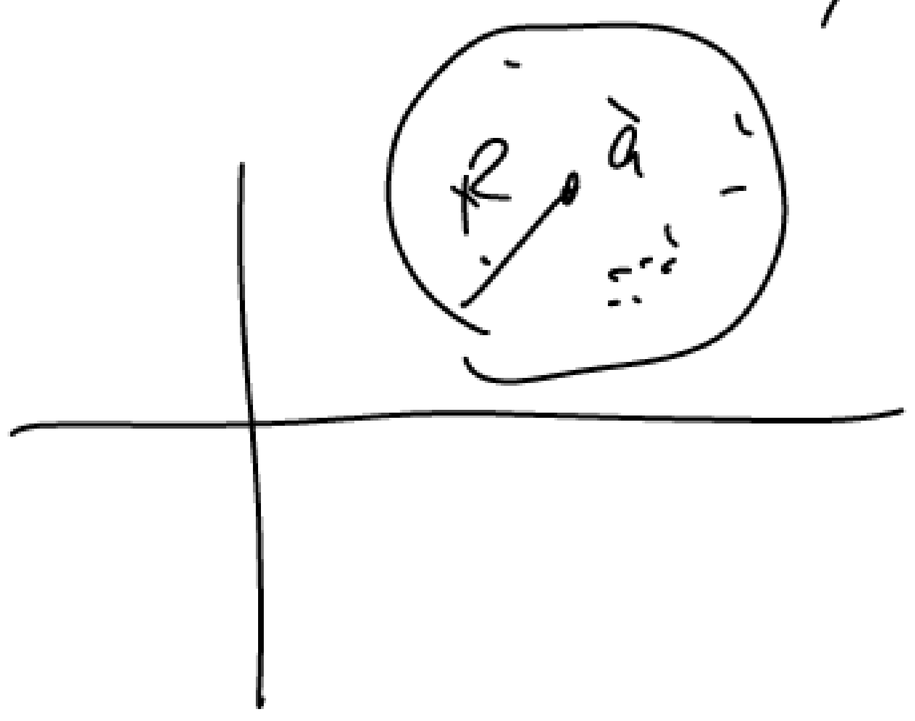


$\begin{pmatrix} \deg(P)=k, \\ |P(0)|=1. \end{pmatrix}$

try $P(z) = \frac{(z-2)^k}{2^k}$

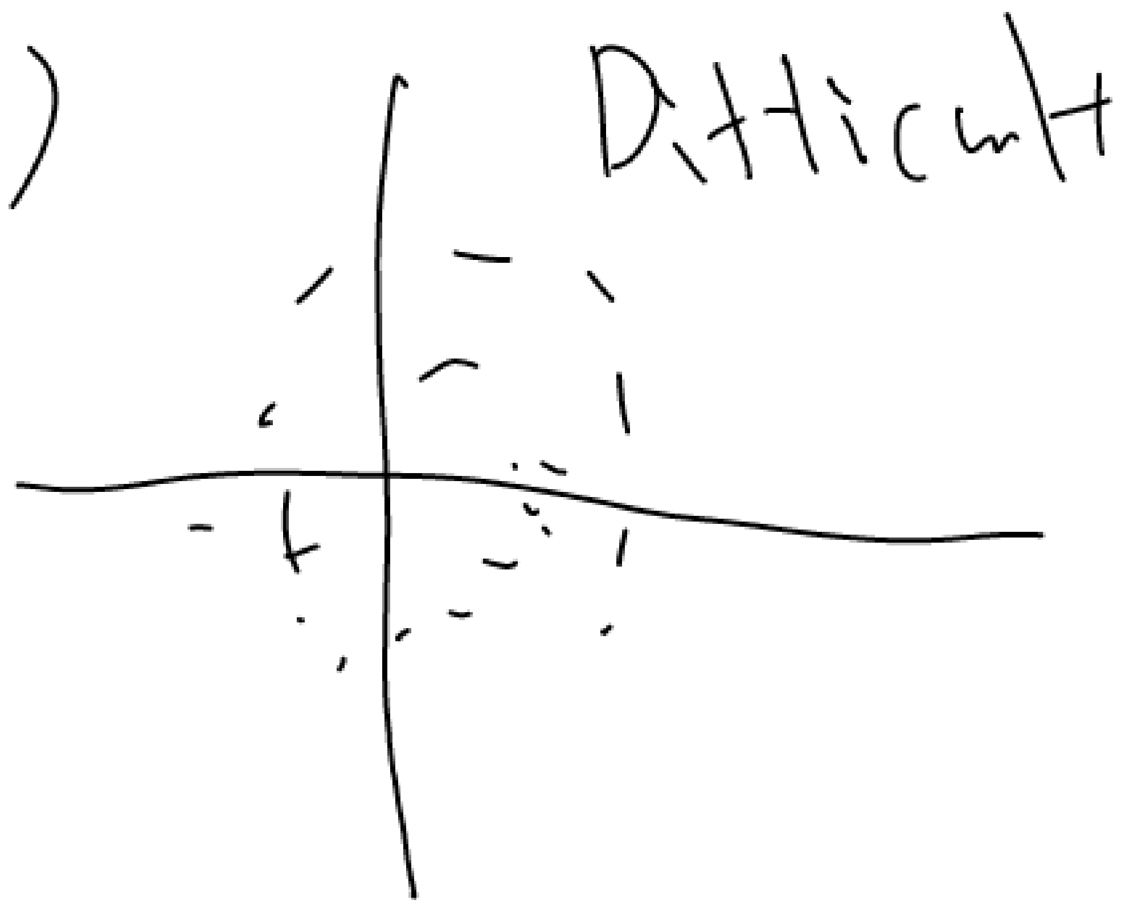
Then $|P(\lambda_i)| \le |\frac{1}{2}|^k$ exponential convergence

(ii) more generally



$p(z) = \frac{(z-a)^k}{|a|^k} \Rightarrow |p(\lambda)| \le \frac{R^k}{|a|^k}$

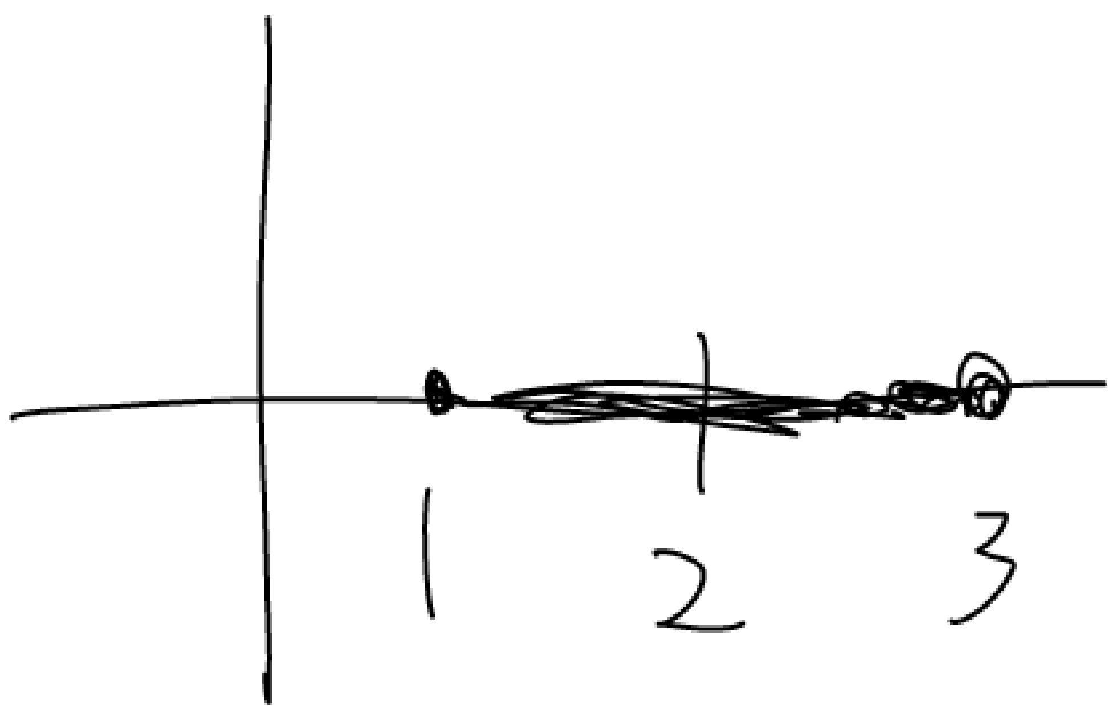
(iii)



Difficult! GMRES may not work well.

What if $\lambda(A) \in \mathbb{R}$?

(iv)



can do $p(z) = \frac{(z-a)^k}{a^k}$ again, but can do much better

(coming up)

e.g. when $A = A^T$

Most problems are like (iii)!

remedy: preconditioning $MAx = Mb$
more like (i) or (iv)

when $A=A^T$

Lanczos iteration/process

Arnoldi $AQ_k = Q_kH_k + q_{k+1}[0,\dots,0,h_{k+1,k}]$, $Q_k^TQ_k = I_k$

$\downarrow$ $H_k = Q_k^TAQ_k$, so when $A=A^T$, $H_k=H_k^T$

$Q_k\underset{\sim}{T_k}$

Hess + symmetric $\Rightarrow$ tridiagonal. $T_k = \begin{bmatrix}\alpha_1 & \beta_1 & & 0\\ \beta_1 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_{k-1}\\ 0 & & \beta_{k-1} & \alpha_k\end{bmatrix}$

general

kth column gives

$$Aq_k = \underbrace{Q_kT_k\begin{bmatrix}0\\ \vdots\\ 0\\ 1\end{bmatrix}}_{\begin{bmatrix}0\\ \vdots\\ 0\\ \alpha\\ \beta\end{bmatrix}} + h_{k+1,k}\,q_{k+1}$$

so

$$q_{k+1} = \frac{1}{h_{k+1,k}}\left(Aq_k - \alpha q_{k-1} - \beta q_k\right)$$

$$= \frac{1}{h_{k+1,k}}\left((A-\beta I)\underset{\sim}{q_k} - \alpha \underset{\sim}{q_{k-1}}\right)$$

also $\beta = (q_k^TAq_k)$, $\alpha = q_{k-1}^TAq_k$

So to get q_{k+1} via Aq_{k+1}, only need to orthogonalise against q_k and q_{k-1}.

(Aq_k automatically orthogonal to $q_1, \dots q_{k-2}$)

$\Rightarrow$ massive speedup over Arnoldi ($A \neq A^T$ case)

$O(nk)$ instead of $O(nk^2)$

+k A-mult   +k A-mult

CG for $Ax=b$, $A=A^T>0$.

start with Lanczos $AQ_k = Q_kT_k + q_{k+1}[0,\dots,0,h_{k+1,k}]$

CG idea: $x_k \in Q_k$ s.t. $\Leftrightarrow x_k = Q_k y$.

$Q_k^T(Ax_k-b)=0$ "Galerkin orthogonality" i.e. residual orth. to Q_k.

$$\Leftrightarrow \underbrace{Q_k^TAQ_k}_{} y = Q_k^Tb = \begin{bmatrix}\|b\|\\0\\\vdots\\0\end{bmatrix}$$

$$T_ky = \begin{bmatrix}\|b\|\\0\\\vdots\\0\end{bmatrix} \quad - Ⓐ$$

tridiagonal lin. system, $O(k)$ flops.

exact soln $Ax_*=b$.

$$\boxed{\text{Thm: } x_k = Q_ky \text{ minimises } \|x_k-x_*\|_A \text{ over } x_k\in Q_k \quad =: \sqrt{(x_k-x_*)A(x_k-x_*)},\ \text{norm}}$$

proof:
$$\|x_k-x_*\|_A^2 = (x_k-x_*)^TA(x_k-x_*)$$
$$= (Q_ky-x_*)^TA(Q_ky-x_*)$$
$$= y^TT_ky - 2\underbrace{y^TQ_k^TAx_*}_{} + x_*^TAx_*$$
$$\tilde b = \begin{bmatrix}\|b\|\\0\\\vdots\\0\end{bmatrix}$$

"complete squares"
$$y^TTy - 2\tilde b^Ty$$
$$=(y-T^{-1}\tilde b)^TT(y-T^{-1}\tilde b) - \tilde b^TT^{-1}\tilde b$$
min. when $y=T^{-1}\tilde b$. same as Ⓐ

□ < (slide)

fast to compute via T_k structure

only b.c. $A>0$

(or:)
min. wrt y (e.g. via gradient $=0$) noting convexity.

$$2T_ky - 2\underbrace{Q_k^T\tilde b}_{\begin{bmatrix}\|b\|\\0\\\vdots\\0\end{bmatrix}} = 0$$

CG convergence

Define $e_k = x_* - x_k$; error at kth iter
($x_0 = 0, Ax_* = b$)

$$\frac{\|e_k\|_A}{\|e_0\|_A} = \min_{x \in K_k(A,b)} \|x_k - x_*\|_A / \|e_0\|_A$$

$$= \min_{\deg(p_{k-1}) \le k-1} \|p_{k-1}(A)\, b - \underbrace{A^{-1} b}_{= x_*}\|_A / \|e_0\|_A$$

$= e_0$. and $b = Ax_*$.

$$= \quad '' \quad \|(p_{k-1}(A)\,A - I)\, e_0\|_A / \|e_0\|_A$$

$$= \min_{\substack{\deg(p) \le k \\ p(0) = 1}} \|p(A)\, e_0\|_A / \|e_0\|_A \qquad A = V \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} V^T$$

$$= \min_{''} \underbrace{\left\| V \begin{bmatrix} p(\lambda_1) & & \\ & \ddots & \\ & & p(\lambda_n) \end{bmatrix} V^T e_0 \right\|_A}_{X} / \|e_0\|_A$$

$$X^2 = \underbrace{e_0^T V}_{y} \begin{bmatrix} \ddots \end{bmatrix} V^T A V \begin{bmatrix} \ddots \end{bmatrix} V^T e_0 \qquad V^T e_0 = y$$

$$= y^T \begin{bmatrix} p(\lambda_1)^2 \cdot \lambda_1 & & \\ & \ddots & \\ & & p(\lambda_n)^2 \cdot \lambda_n \end{bmatrix} y$$

$$\le \max_i |p(\lambda_i)^2| \cdot y^T \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} y$$

$$= \max |p(\lambda_i)|^2 \cdot \|e_0\|_A^2$$

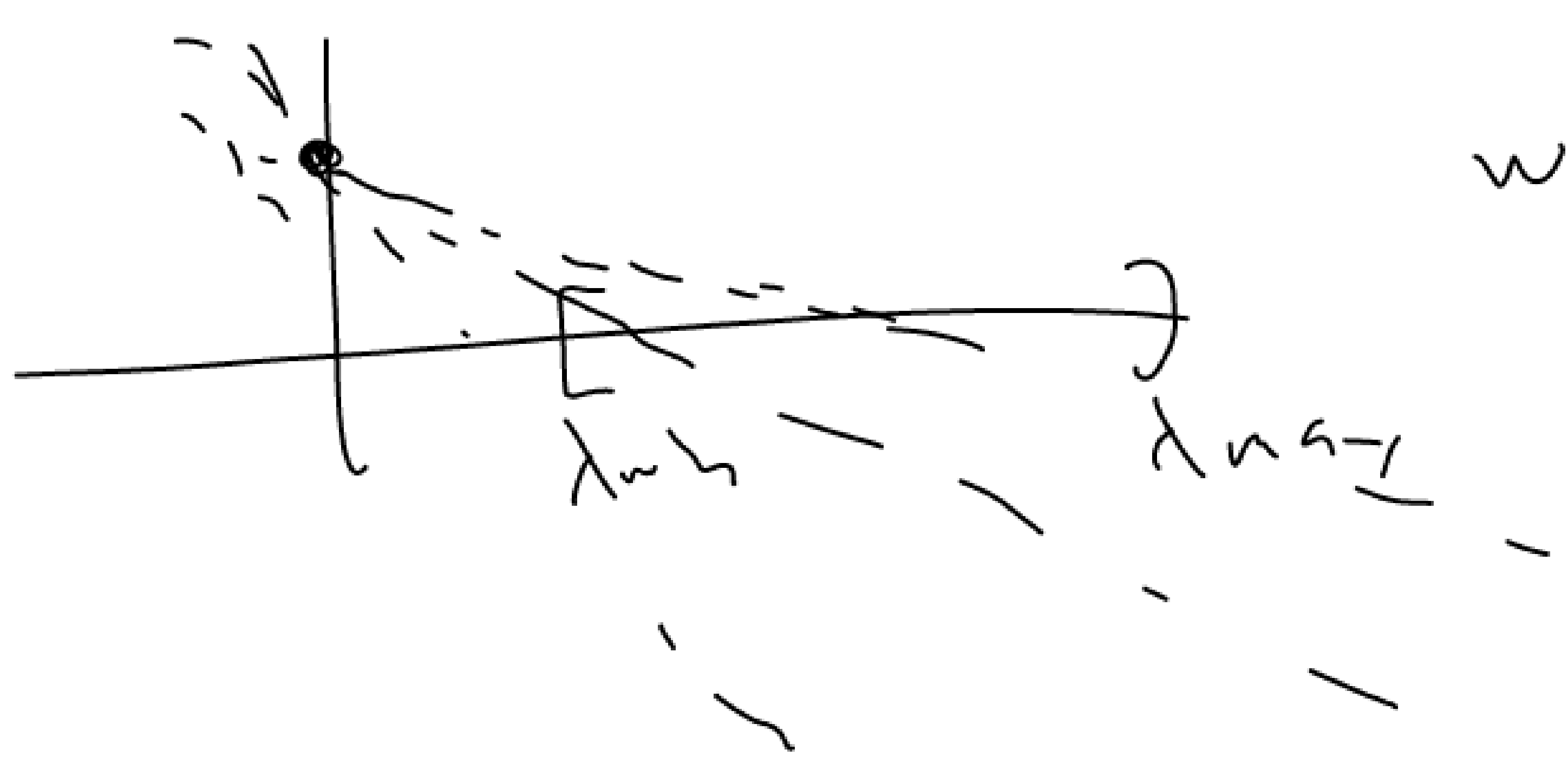
note

$$y^T \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} y = e_0^T \underbrace{V \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} V^T}_{A} e_0 = \|e_0\|_A^2$$

again (as in GMRES), minimise $|p(z)|$ on $z \in \lambda(A)$, $p(0) = 1$

(Chebyshev polynomials)

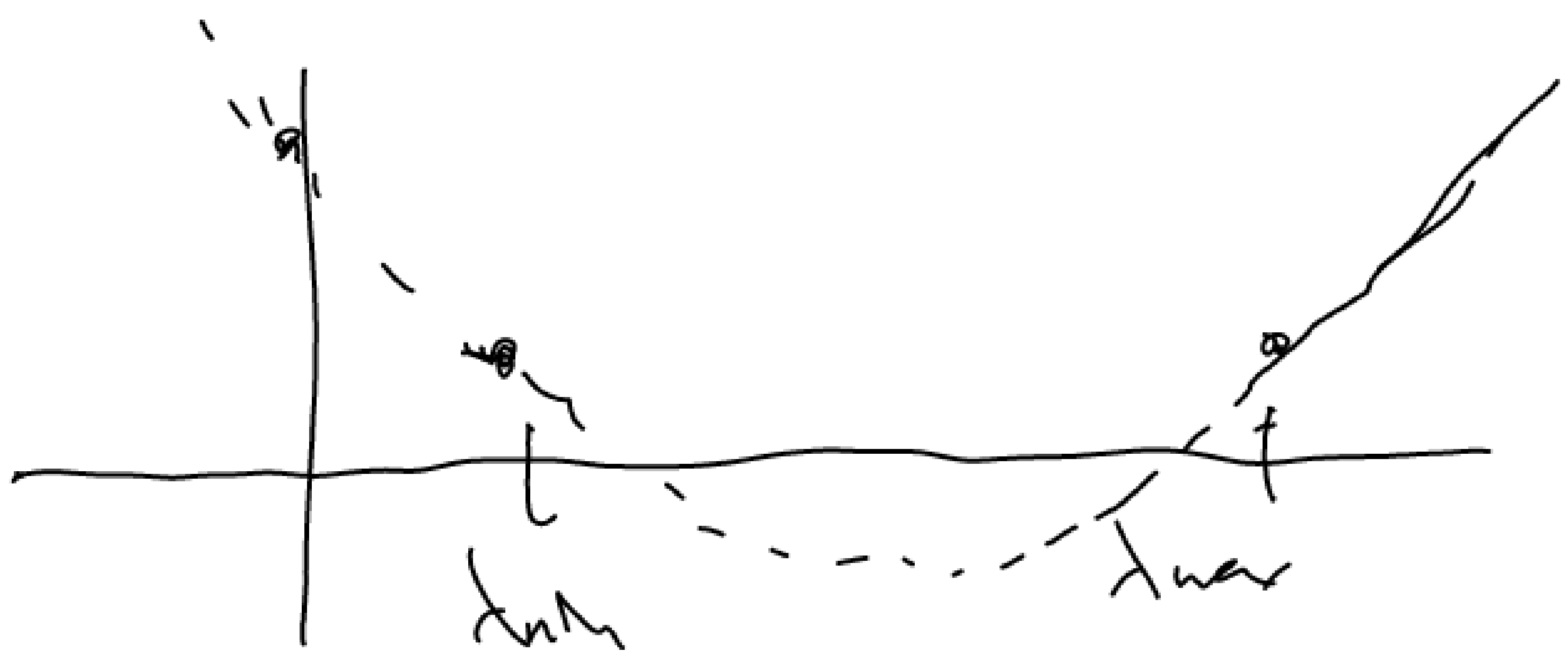
1. when $k = 1$



which line?

ans: s.t. $p(\lambda_{max}) = -p(\lambda_{min})$

2. when $k = 2$



3 general k

Chebyshev poly: $\cos(k\theta) = T_k(\cos\theta)$

oscillates k+1 times on $\theta \in [0,\pi]$

$\cos(k\theta) = T_k(\cos\theta)$ $\quad (= T_k(x)$ via $x = \cos\theta$,

$x \in [-1,1]$

$z = e^{i\theta}$ $\quad = \frac{1}{2}(z^k + z^{-k})$ $\quad x = \frac{1}{2}(z + \frac{1}{z})$

$$2 \cdot \frac{1}{2}(z^k + z^{-k}) \cdot \frac{1}{2}(z + \frac{1}{z})$$

$$= \frac{1}{2}(z^{k+1} + z^{-(k+1)}) + \frac{1}{2}(z^{k-1} + z^{-(k-1)})$$

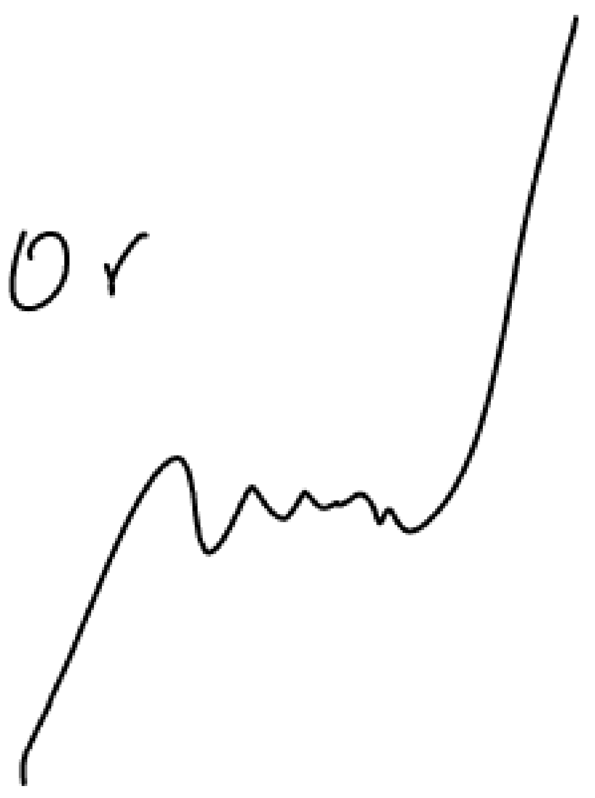
so $2\,T_k(x) \cdot x = T_{k+1}(x) + T_{k-1}(x)$

(3-term) recurrence.

$|T_k(x)|$ grows rapidly outside $[-1,1]$.

(fastest)

or

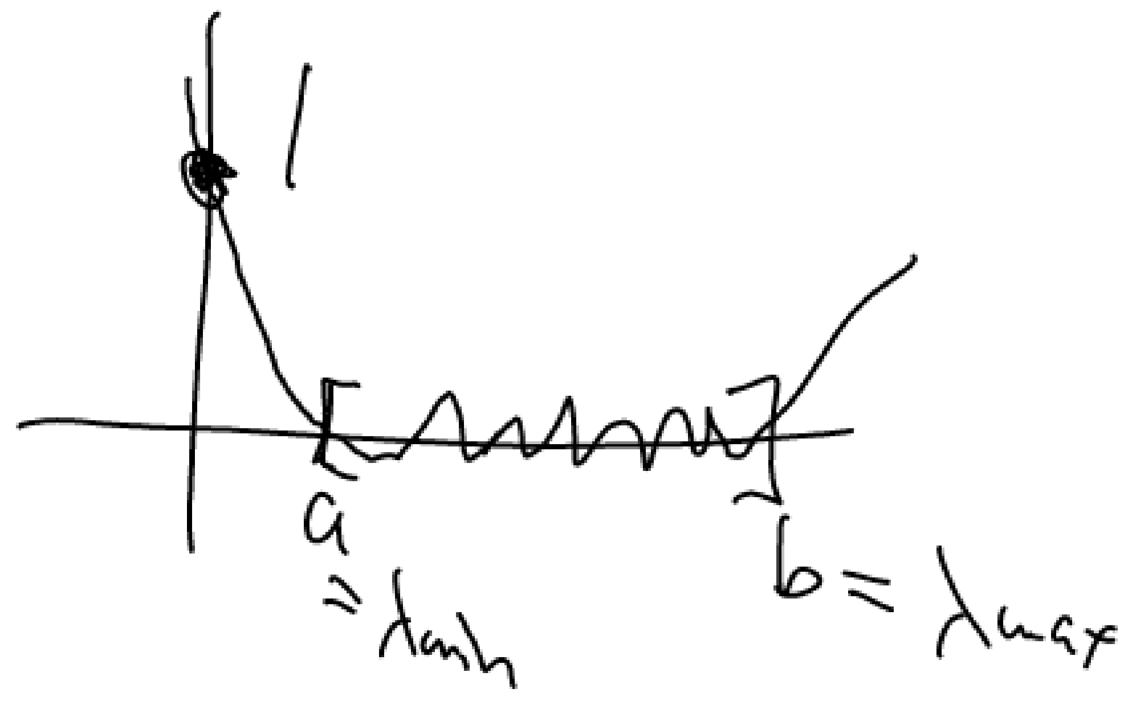


Roughly,

$T_k(x) = 2^k x^k + (\deg \le k-1)$ so $T_k(x) = O(2^k x^k)$

in example, $\sim 4^{-k}$ instead of 2^{-k}

Usage here:



consider $P_k(x) =: \underbrace{T_k\left(\frac{2(x-a)}{b-a} - 1\right)}_{\substack{x=a:\,-1 \\ x=b:\,1}} \cdot \frac{1}{T_k\left(\frac{-2a}{b-a} - 1\right)}$

$\uparrow$ $x = 0$

Then on $[a,b]$,

$$|P_k(x)| \le \frac{1}{T_k\left(\frac{-2a}{b-a} - 1\right)} = \frac{1}{\underbrace{T_k\left(\frac{b+a}{b-a}\right)}}$$ as $|T_k(x)| = |T_k(-x)|$

---

if time:

$\left(x_* =\right) \frac{b+a}{b-a} = \frac{1}{2}(z + \frac{1}{z})$; $\left(z = \left(\frac{b+a}{b-a}\right) + \sqrt{\left(\frac{b+a}{b-a}\right)^2 - 1} \right.$

$$\left(\frac{b+a}{b-a}\right)^2 - 1 = \frac{(b+a)^2 - (b-a)^2}{(b-a)^2} = \frac{4ab}{(b-a)^2}$$

$$\left. = \left(\frac{b+a}{b-a}\right) + \frac{\sqrt{4ab}}{b-a} = \frac{(\sqrt{b}+\sqrt{a})^2}{b-a} = \frac{\sqrt{b}+\sqrt{a}}{\sqrt{b}-\sqrt{a}} = \frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1} \right)$$

so $|T_k(x_*)| = \left|\frac{z^k + z^{-k}}{2}\right|$

$$\ge \frac{1}{2} \cdot \left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^k$$

Hence after k CG iterations, $|p_k(\lambda_i)| \le 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k$

so $$\frac{\|e_k\|_A}{\|e_0\|_A} \le \max_i |p_k(\lambda_i)| \cdot \|e_0\|_A$$
$$\le 2\cdot\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \cdot \|e_0\|_A^2$$

$$\kappa_2(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} \quad \text{as } A>0$$

(cond. num)

(NOTE: $Ax=b$, even $\|Ax-b\|_2$ can be reduced to posdef system via normal eqn
$A^TAx = A^Tb$
works in $\text{span}(b, A^TAb, (A^TA)^2b, \ldots)$
Great if $\kappa(A)=O(1)$, otherwise not.)

MINRES: symmetric variant of GMRES. (skip)

(separate board) (skip if no time)

	matrix	minimise	how	cost k iter
CG	$A=A^T>0$	$\|x_k-x_*\|_A$	$Q^T(Ax-b)=0$.	$Mk+O(nk)$
MINRES	$A=A^T$	$\|Ax-b\|_2$	$\min_y \left\|Ty-\begin{bmatrix}\|b\|\\0\\\vdots\\0\end{bmatrix}\right\|_2$ $(x=Qy)$	$Mk+O(nk^2)$
GMRES	general A	$\|Ax-b\|_2$	$\min \left\|Hy-\begin{bmatrix}\|b\|\\0\\\vdots\\0\end{bmatrix}\right\|_2$	$Mk+O(nk)$

M: cost for A-times vector.

(nonex, skip)

Lanczos for $Ax=\lambda x$, $A\overset{T}{=}A$ (always krylov, Q_k)

Idea: find eigpair $(\hat\lambda,\hat x)$ s.t. $\hat x\in\text{span}(Q_k)$

$\iff \hat x = Q_k y$.

$$AQ_ky \simeq \lambda Q_ky$$

"rectangular eigenproblem"

(usually no soln!)

one approach!

Impose $Q_k^T(AQ_ky-\lambda Q_ky)=0$. "Galerkin orthogonality"

$\iff T_ky=\lambda y$ (k solns)

then $(\hat\lambda,\hat x)=(\lambda,Q_ky)$ is approximate eigenpair.

Convergence: Very complicated!

But usually rapid conv to extremal eigval's (largest/smallest)

By C-F,

$$-\ \lambda_{\max}(A) = \max_x \frac{x^TAx}{x^Tx} \geq \max_{x\in Q_k}\frac{x^TAx}{x^Tx} \geq \frac{v^TAv}{v^Tv} \leftarrow \lambda \text{ w/ power method } k \text{ iter.}$$

↑ Subsp.

↑ $v := p(A)b = A^{k-1}b$

↓ $\lambda_{\max}$ (as we know)

"somewhat similar" for

(demo)

Randomised Alg in NLA

Recap:

- Direct / Classical methods
    - LU for $Ax = b$.
    - QR alg for $Ax = \lambda x$
  - \+ pro: great reliability
  - \- con: $O(n^3)$ cost!

- Iterative methods
  - \+ : really fast when it works
  - \- : not always work

- Randomised algs
  - \+ : sometimes dramatic speedup. with high probability.
    "allow for some error in soln".
  - \- : "lack of reproducibility".

We'll use random matrices in algs; so study them first.

Gaussian matrices

$\underset{n}{m\,\boxed{G}}$ $\quad G_{ij} \sim N(0,1)$ indep. standard normal / Gaussian

Key properties:

1. orthogonal invariance. Let $\underset{m}{m\,\boxed{Q}}$, $\underset{n}{n\,\boxed{V}}$ orthogonal. (indep of G)

$$G \text{ Gaussian} \Leftrightarrow QGV \text{ Gaussian.} \qquad \text{"}G \overset{\text{dist}}{=} QGV\text{"}$$

$$\Rightarrow G\boxed{V_1} \text{ - orthonormal "}.$$

(sketch proof:) Consider QG. or Qg. (each col.)

$$\mathbb{E}(Qg) = Q\mathbb{E}[g] = 0.$$

$$\mathrm{Var}[Qg] = \mathbb{E}[(Qg)(Qg)^T] = Q\mathbb{E}[gg^T]Q^T = QQ^T = I.$$

and distribution of mult.var. Gaussian determined completely by mean + covariance matrix.

(or: joint pdf is $\left(\frac{1}{\sqrt{2\pi}}\right)^n \cdot \exp\left(\frac{-1}{2}(g_1^2+\dots+g_n^2)\right)$, invariant under orth. transform

2. Marchenko-Pastur rule.

$\underset{\sqrt{n}}{\sqrt{m}\,\boxed{G}}$ $\quad$ "$\sigma_i(G) \in [\sqrt{m}-\sqrt{n}, \sqrt{m}+\sqrt{n}]$"

$$\left.\begin{matrix}\mathbb{P}[\sigma_1(G) > \sqrt{m}+\sqrt{n}+t] \\ \mathbb{P}[\sigma_n(G) < \sqrt{m}-\sqrt{n}-t]\end{matrix}\right. \leq \exp(-t^2)$$

Hence $\kappa_2(G) \lesssim \dfrac{\sqrt{m}+\sqrt{n}}{\sqrt{m}-\sqrt{n}} = O(1)$ $\qquad$ e.g. if $m \geq 4n$, $\kappa_2(G) \lesssim 3$

proof: omitted/nonexaminable.

demo

Randomised algorithm for $\min \|\underset{m\ n}{\boxed{A}}x - b\|_2$

Suppose $Ax = b$, $m >> n$.

quite natural — e.g. $\min_{x \in \text{span}(\underset{m\times n}{Q})} \|Ax - b\|$

Quick alg? "row subselection"

$$\begin{bmatrix} \\ A \\ \\ \end{bmatrix} x = \begin{bmatrix} \\ b \\ \\ \end{bmatrix} \Rightarrow \boxed{A_1}x = b_1$$

when $Ax \approx b$, usually works.

But fails when e.g. $A = \begin{bmatrix} \varepsilon I \\ A_2 \\ \vdots \\ A_k \end{bmatrix}$

Remedy: randomisation!

Idea: "Randomise mixing to avoid pathologies."

(i) Consider $\min_x \|\underbrace{\boxed{G_Q}}_{m\times m}(Ax - b)\|_2$

(ii) subselection $\min_x \|\underset{s\times m}{\boxed{G}}(Ax - b)\|_2$ sketch matrix

$$\underbrace{\boxed{n < s << m}}_{\text{sketch size}}$$

"Sketch and solve" for LS

$x_* = \text{argmin} \| Ax - b\|_2$

$\hat{x} = \text{argmin} \| G(Ax-b)\|_2$ $\qquad s\,\underset{m}{\boxed{\quad G \quad}}$ $\qquad m >> \underset{\sim\sim}{s > n}$

Let ${}_m\underset{n+1}{[A,b]} = QR.$

$$\|Ax-b\|_2 = \left\|[A,b]\begin{bmatrix}x\\-1\end{bmatrix}\right\|_2$$
$$= \left\|QR\begin{bmatrix}x\\-1\end{bmatrix}\right\|_2 = \left\|R\begin{bmatrix}x\\-1\end{bmatrix}\right\|_2$$

now

$$\|G(Ax-b)\|_2 = \left\|GQR\begin{bmatrix}x\\-1\end{bmatrix}\right\|_2, \quad \text{so}$$

$$\sigma_{min}(GQ)\cdot\left\|R\begin{bmatrix}x\\-1\end{bmatrix}\right\|_2 \quad \le \quad \le \|GQ\|_2\cdot\left\|R\begin{bmatrix}x\\-1\end{bmatrix}\right\|_2$$

Now, (i) $s\,\underset{n}{\boxed{GQ}}$ is Gaussian,

(ii) By M-P, $\sigma_{min}(GQ), \|GQ\|_2 \in [\sqrt{s}-\sqrt{n}, \sqrt{s}+\sqrt{n}]$

so $\forall x$,

$$\left(1-\sqrt{\tfrac{n}{s}}\right)\|Ax-b\|_2 \le \frac{1}{\sqrt{s}}\|G(Ax-b)\|_2 \le \left(1+\sqrt{\tfrac{n}{s}}\right)\|Ax-b\|_2 \quad - ✪$$

by minimising $\|G(Ax-b)\|_2$,

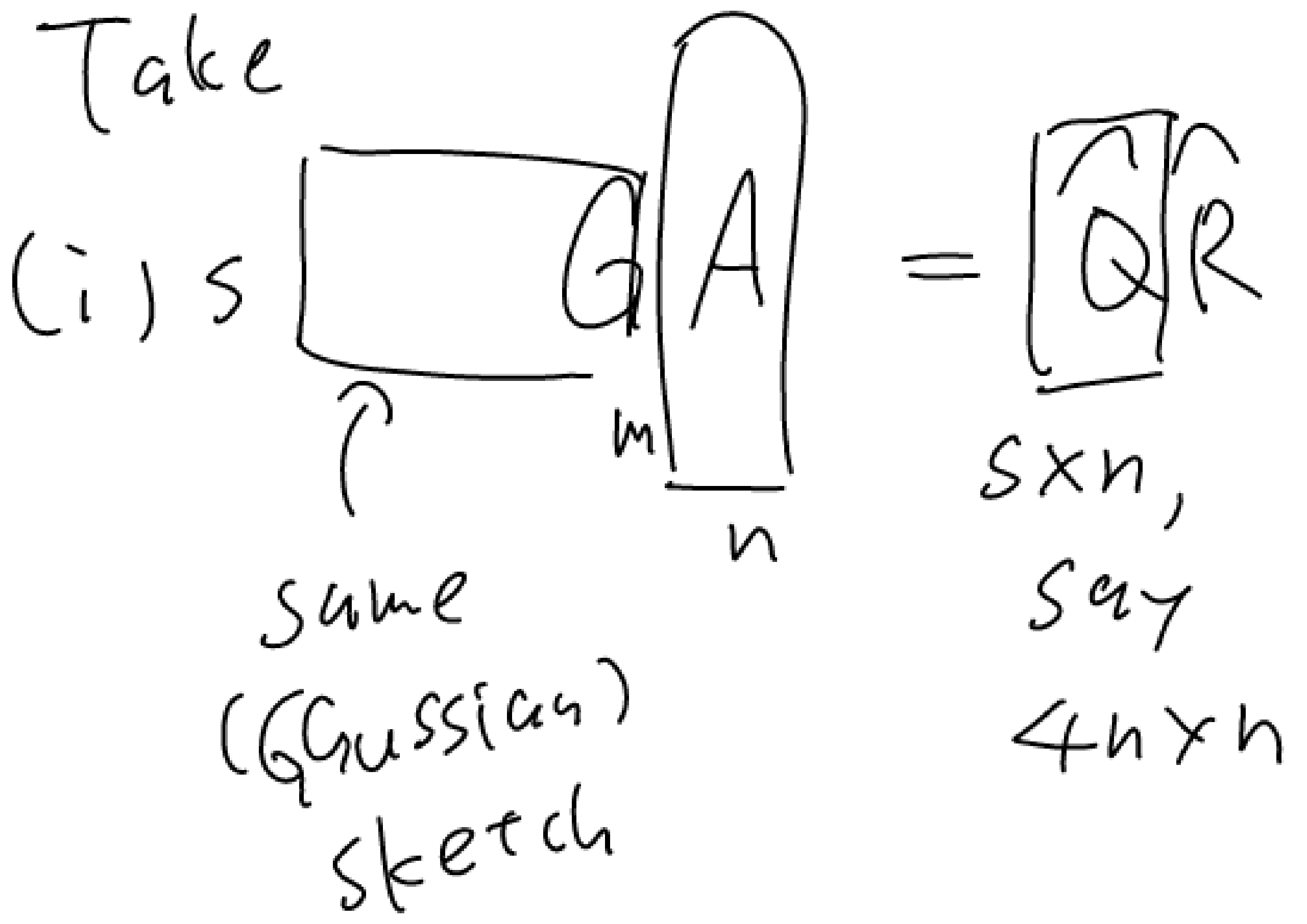
$$\|A\hat{x}-b\|_2 \le \frac{1}{\sqrt{s}-\sqrt{n}}\|G(A\hat{x}-b)\|_2$$
$$\le \frac{1}{\sqrt{s}-\sqrt{n}}\|G(Ax_*-b)\|_2 \quad \text{(by optimality of } \hat{x} \text{ for } \|G(Ax-b)\|_2)$$
$$\le \frac{\sqrt{s}+\sqrt{n}}{\sqrt{s}-\sqrt{n}}\|(Ax_*-b)\|_2. \quad \text{by ✪ w/ } x = x_*.$$

(think $\|Ax_*-b\|_2 = 10^{-10}$, then $\|A\hat{x}-b\|_2 \lesssim 3\cdot10^{-10}$)

Better news: Sketch to precondition.
(for full accuracy)

Take

(i)



(ii) $\min_x \|Ax-b\|_2 = \min_y \|A\hat{R}^{-1}y - b\|_2$, solve

via normal equation using CG.

$$(A^TA)x = A^Tb \iff (\hat{R}^{-T}A^TA\hat{R}^{-1})y = \hat{R}^{-T}A^Tb$$

(then $x = \hat{R}^{-1}y$)

Recall CG fast if $\kappa_2(\text{Mat}) = O(1)$

comment on $\text{span}(A^Tb, (A^TA)A^Tb, (A^TA)^2A^Tb, \dots)$

Thm: $\kappa_2(A\hat{R}^{-1}) = O(1)$

proof: $$\boxed{G}\boxed{A} = (\underset{\text{Gaussian, } 4n\times n}{GQ})R = \tilde{Q}\underbrace{\tilde{R}R}_{\hat{R}} = \hat{Q}\hat{R} \quad (\hat{Q} = \tilde{Q})$$

So $$A\hat{R}^{-1} = A\cdot(R^{-1}\tilde{R}^{-1}) = QR\cdot R^{-1}\tilde{R}^{-1} = Q\tilde{R}^{-1}$$

now $\kappa(\tilde{R}^{-1}) = \kappa(\tilde{R}) = \kappa(\underbrace{GQ}_{=\tilde{Q}\tilde{R}}) = O(1)$ by M-P. ∎

So CG converges like $\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^\ell$ after ℓ iter.

NOTE:

1. No need to compute $A\hat{R}^{-1}$, in CG do $(A(\hat{R}^{-1}x))$
2. In practice, use fast sketch for G

(Recap of SVD)

Recall

- SVD computation $O(mn^2)$

- with best rank-r approximation,

$$\|A - A_r\|_2 = \sigma_{r+1}$$

pseudo inverse
(p-inv)

For ${}_m\boxed{M}_n = \boxed{U}\boxed{\Sigma}\boxed{V}^T = U_r \Sigma_r V_r^T$,

p-inv ${}_n\boxed{M^\dagger}_m = V_r \Sigma_r^{-1} U_r^T$ "$(ABC)^\dagger = C^\dagger B^\dagger A^\dagger$"

(invert only singvals > 0)

properties:

- $MM^\dagger M = M$, $M^\dagger M M^\dagger = M^\dagger$, $MM^\dagger = (MM^\dagger)^T$, $M^\dagger M = (M^\dagger M)^T$

often def.

- if $m = n$ and $\exists M^{-1}$, then $M^{-1} = M^\dagger$.

@ - for $\boxed{A}x = b$, $x = A^\dagger b$ gives minimum-norm soln.
(under det.)

Randomized SVD, rank-r case

suppose rank(A) = r. Then

$$A = \boxed{U_1}\,\Sigma_1\,\boxed{V_1^T}$$

now consider

$$A\underset{\substack{\uparrow \\ \text{Gaussian, } n \times r}}{X} = U_1 \Sigma_1 (V_1^T X).$$

Claim: Letting $AX = QR$, $\quad QQ^TA = A$.

"proof": $\underset{\substack{\square \\ r\times r}}{\Sigma_1}\underset{\substack{\square \\ r\times r}}{(V_1^T X)} = \boxed{\tilde{Q}}\boxed{R}$ $\quad \tilde{Q}$ orthogonal.

↖ full row-rank

so $$AX = \underbrace{U_1 \cdot \tilde{Q}}\, R$$
$$= Q \; R.$$ i.e. $\operatorname{span}(U_1) = \operatorname{span}(Q)$

$\iff U_1 W = Q$ for some orthogonal $\underset{r\times r}{W}$.

$\iff QQ^T = U_1 U_1^T$

$\Rightarrow QQ^TA = U_1U_1^TA = A$.

HMT algorithm for low-rank approx.

$\underset{m\times n}{A} \approx \boxed{}\,\boxed{}$

1. Draw random (Gaussian) mat $n\,\underset{r}{\boxed{X}}$

   $\tilde{r}$ assume given

2. Compute $m\,\underset{r}{\boxed{AX}}$

3. Compute $AX =_m \underset{r}{\boxed{Q}}\boxed{R}$ (QR fact.)

4. Output $\hat{A} =_m \boxed{Q}\,\boxed{Q^T A}$

NOTES:

- $O(mnr)$ cost.

$$\boxed{\begin{aligned}\text{Thm: } \mathbb{E}[\|A - A\|_2] &\le O(1)\,\|A - A_{\hat{r}}\|_2 \\ \text{for } \hat{r} &< r. \quad (\text{say } \hat{r} = 0.9\,r)\end{aligned}}$$

- "nearly optimal" in two ways:
  - (i) $O(1)$ constant.
  - (ii) not $\|A - A_r\|_F$ but $\|A - A_{\hat{r}}\|_F$.

HMT analysis

$(AX = QR)$
$\quad\llcorner$ random $n\times r$

(Error=) $A - QQ^TA = (I - QQ^T)A$

$$= (I - QQ^T)A(I - XM^T) \quad \text{for any} \quad r\,\underset{n}{\boxed{M^T}}$$

(set cleverly!)

(reason: $AX = QR$, so
$(I - QQ^T)AX = (I - QQ^T)QR = 0$)

(idea:) want
$A(I - XM^T)$ small.

$\Sigma_1: \hat{r}\times\hat{r}$

Write $A = [U_1\ U_2]\begin{bmatrix}\Sigma_1 & \\ & \Sigma_2\end{bmatrix}\begin{bmatrix}V_1^T\\ V_2^T\end{bmatrix}$.

want "$V_1^T(I - XM^T)$ to disappear"

$$V_1^T = (V_1^TX)M^T$$

$$\hat{r}\,\underset{n}{\boxed{\qquad\qquad}} = \underset{\hat{r}\times r}{\boxed{\ }}\ \underset{r\times n}{\boxed{\qquad\qquad}}$$

$(\hat{r} < r << n)$

ith col: $v_i = \underset{\hat{r}\times r}{(V_1^TX)}m_i$ underdet system

so set $M^T = (V_1^TX)^\dagger V_1^T$.

Then,

$$\text{Error} = (I - QQ^T)[U_1\ U_2]\begin{bmatrix}\Sigma_1 & \\ & \Sigma_2\end{bmatrix}\begin{bmatrix}V_1^T\\ V_2^T\end{bmatrix}(I - X(V_1^TX)^\dagger V_1^T)$$

$$= (I - QQ^T)U_2\Sigma_2V_2^T(I - X(V_1^TX)^\dagger V_1^T)$$

so $\|\text{Error}\|_2 \le \underbrace{\|I - QQ^T\|_2}_{=1}\cdot\underbrace{\|\Sigma_2\|_2}_{\sigma_{\hat{r}+1}}\underbrace{\|I - X(V_1^TX)^\dagger V_1^T\|_2}_{Z:\ \text{need}\ O(1)}$

$Z \le 1 + \|X\|_2\cdot\|\underbrace{(V_1^TX)^\dagger}_{(\text{Gaussian})^\dagger}\|_2$ $\quad \hat{r}\times r$ so $\sigma\sim[\sqrt{r}-\sqrt{\hat{r}}, \sqrt{r}+\sqrt{\hat{r}}]$, by M-P

$\le 1 + (\sqrt{n}+\sqrt{r})\cdot\frac{1}{\sqrt{r}-\sqrt{\hat{r}}} = O(1)$

Key concepts for revision

- Norms
- SVD
- low-rank approx
- C-F
- $Ax=b$ via LU
- $\min\|Ax-b\|_2$ via QR
- QR alg for $Ax=\lambda x$
- Arnoldi, Lanczos decomp for Krylov subspace.
- GMRES for $Ax=b$, $A\neq A^T$
- CG for $Ax=b$, $A=A^T\succ 0$
- $\min\|Ax-b\|_2$ via sketch-to-solve, sketch-to-precondition
- HMT: randomised low-rank approx

Randomised algorithm for $\min \|Ax - b\|_2$

Observation 1.

suppose $Ax = b$, $A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}$. $b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$

Then $A_1 x = b_1$

Lin sys or LS gives same solution x! ("subselection of rows")

2. if $Ax \simeq b$,

$A_1 x \simeq b$. so "expect" solution

$\min \|A_1 x - b_1\|_2$ would give good soln. $\hat{x}$

3. But easy counter example! e.g.

$$A = \begin{bmatrix} \varepsilon I \\ I \\ I \\ I \\ \vdots \\ I \end{bmatrix} \leftarrow A_1 \qquad b = \begin{bmatrix} b_1 \\ b_2 \\ b_2 \\ \vdots \\ b_2 \end{bmatrix}$$

then $x \simeq b_1$

but

$\hat{x} = \frac{b_1}{\varepsilon}$.

"pathological choice of subselection."

Idea: "Randomise mixing to avoid pathologies."

(i) Consider $\min_x \| \underbrace{\boxed{G_Q}}_{m \times m} (Ax - b) \|_2$

sketch matrix

(ii) subselection $\min_x \| \underbrace{\boxed{G_2}}_{s \times m} (Ax - b) \|_2$

$\boxed{n < s \ll m}$.

sketch size