Gilbert Strang: Linear Algebra & Learning from Data.

__/__/____

★ → PCA ⟷ Eigen value decomposition of A^TA.

★ $A^{m \times n}$

$\mathbb{R}^n \xrightarrow{A} \mathbb{R}^m$

Four fundamental subspaces

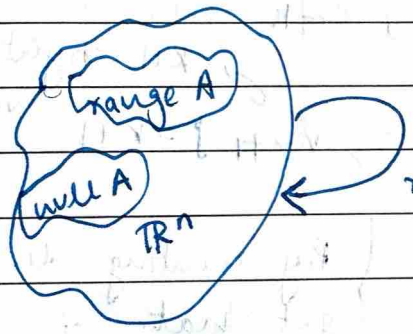
→ range A is subspace of $\mathbb{R}^m$ → (column space of A)

→ null space of A is subspace of $\mathbb{R}^n$ → $\{x \mid Ax = 0\}$

→ range A^T is subspace of $\mathbb{R}^n$ → (column space of A^T)

→ null space of A^T is subspace of $\mathbb{R}^m$ → $\{y \mid A^Ty = 0\}$.

col space of A^T = row space of A.

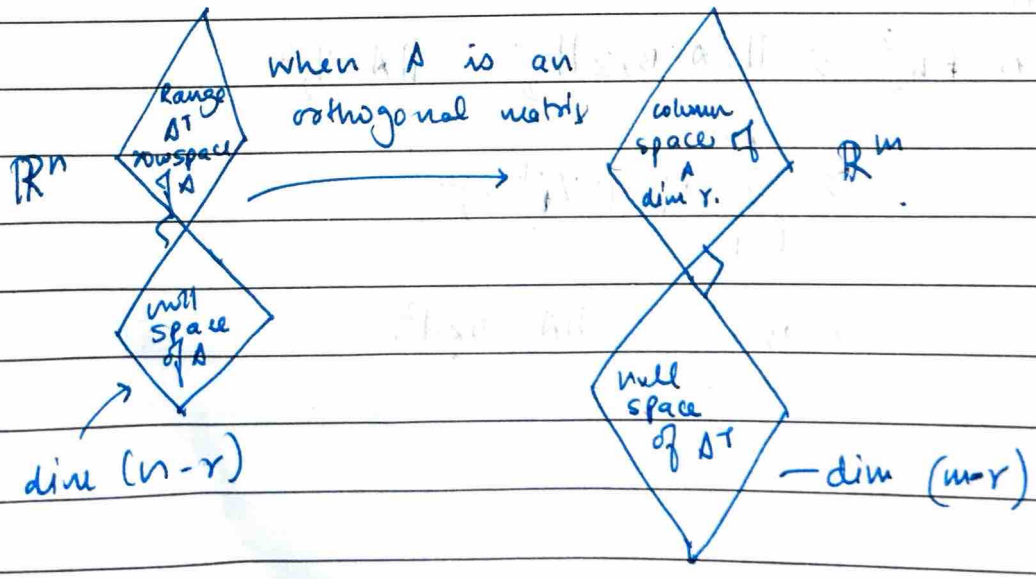


$A_{m \times n}$ $A_{n \times n}$.

rank $A = r$

dim (null (A)) + dim (range A) = n

$n - r + r = n$.



Projection matrices

$P^2 = P$
- range $P \cap$ null $P = \{0\}$
- range $P \oplus$ null $P = V$.

A projection P is an orthogonal projector $\iff P = P^*$

range $P \perp$ null P.

Suppose $P = P^*$, then let $x \in$ null P $= \text{range}(I-P)$, $y \in$ range P

$y = Pw$

$Py = P^2 w = y$

want $\langle x, y \rangle = 0$.

~~$\langle (I-P)x, Py \rangle = y^* P^* (I-P)x$~~

$\langle (I-P)x, Py \rangle = y^* P^* (I-P)x$

$= y^* (Px - P^2 x) = 0$.

WLOG: q_i's are orthonormal

$Pq_i = q_i \quad \forall \ 1 \le i \le n$

Conversely suppose range $P = \langle q_1, \dots q_n \rangle$ (say)

& null $P = \langle q_{n+1}, \dots, q_m \rangle$ say

$\to Pq_j = 0 \quad \forall \ n+1 \le j \le m$

Let

$$Q = \left[\begin{array}{c|c|c|c|c} q_1 & \cdots & q_n & \cdots & q_m \end{array}\right]$$

$$Q^* = \left[\begin{array}{c} q_1^* \\ \hline q_2^* \\ \hline \vdots \\ \hline q_m^* \end{array}\right]$$

$$PQ = \left[\begin{array}{c|c|c|c|c} q_1 & q_2 & \cdots & q_n & 0 \cdots 0 \end{array}\right]$$

$Q^* PQ =$

__/__/____

$Q^* P Q$

$$= \left[\begin{array}{c} q_1^* \\ \hline \vdots \\ \hline q_m^* \end{array}\right] \left[\begin{array}{c|c|c|c|c} q_1 & \cdots & q_n & 0 \cdots & 0 \end{array}\right] = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & 0 \\ & & & & & \ddots \\ & & & & & & 0 \end{bmatrix}$$

$Q^* P Q = \text{diag}(\underbrace{1, \ldots 1}_{n \text{ 1's.}}, 0, \ldots 0)$

$\boxed{P = Q \Sigma Q^*}$ → SVD of P.

$$Q = \overbrace{\left[\begin{array}{c|c|c|c|c} q_1 & \cdots & q_n & 0 \cdots & 0 \end{array}\right]}^{\hat{Q}}$$

$P = Q \Sigma Q^* = \hat{Q} \hat{Q}^*$

$\hat{Q} \to m \times n.$

$\boxed{P^* = (\hat{Q} \hat{Q}^*)^* = \hat{Q} \hat{Q}^* = P}$

for any orthogonal projection P

$\boxed{P = \hat{Q} \hat{Q}^*}$ → range of P is colspace of $\hat{Q}$ and action of P on V

projection of V on colspace of $\hat{Q}$

i.e.

$P : V \longrightarrow V$

m dim range $P = n \leq m$

$P^2 = P$

(if $n = m$, then $P = I$)

* $\therefore$ P is always rank deficient.

* **Orthogonal projection P is not an orthogonal matrix**

Warning: Orthogonal projections are not orthogonal matrices.

Special case: $\hat{q} = [q_*] \quad \|q\| = 1$

Then $P = q_* q_*^*$ → rank 1 matrix.

(denote by P_q)

$P_q(v)$ = projection of V in the direction of q (or on the subspace spanned by q)

$I - P = I - qq^*$

is orthogonal projection on the subspace $q^\perp$ (rank $(m-1)$ projection). It is denoted by $P_{\perp q}$

Summary: The orthogonal projection on a subspace $\{q_1, \dots q_n\}$ where q_i's are orthonormal is given by

$$P = \hat{q}\,\hat{q}^*$$

$$\hat{q} = \left[\, q_1 \,|\, \cdots \,|\, q_n \,\right]$$

* Orthogonal projection on any subspace A with basis

$S = \{a_1, \ldots, a_n\}$

let $A = [a_1 | \cdots | a_n]$

let y be the orthogonal projection of a vector x onto ~~vector~~ range (A).

$y - x \perp$ range (A).

let P_A denote the required orthogonal projection.

Then $y = P_A(x)$.

$y - x = P_A(x) - x \in \text{null } P_A \perp \text{range } P_A$

$y - x \perp$ range (A).

$y - x \perp$ range A

$y - x \perp a_j \quad \forall \ 1 \le j \le n$

$\therefore a_j^* (y - x) = 0. \quad \forall a_j$

$y \in$ range $A \Rightarrow y = Av$ for some v.

$\therefore a_j^* (Av - x) = 0 \quad \forall a_j$

$A^* (Av - x) = 0$

$A^* Av - A^* x = 0$

$A^* Av = A^* x.$ (A is full rank $\Rightarrow A^* A \neq$ singular)

$v = (A^* A)^{-1} A^* x$

$v = A^{-1} (A^{*-1}) A^* x.$

$v = A^{-1} x$

$$\boxed{v = (A^* A)^{-1} A^* x}$$

$$y = Av = \underbrace{A(A^*A)^{-1}A^*}_{P_A} x$$

Summary:

The orthogonal projection on a subspace with basis $\{a_1, \dots a_n\}$ which are not orthogonal is given by

$$P_A = A(A^*A)^{-1}A^* \quad \text{where } A = \begin{bmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{bmatrix}$$

$(I-P)$ is an orthogonal projection on a subspace $A^\perp$ (orthogonal to A).

★ **Last time**

- Projection $P: V \to V$ is a linear transformation satisfying $P^2 = P$.
- Orthogonal projection P : (1) range $P \perp$ null P ⇕ (2) $P^* = P$ (symmetric)

- Formula for orthogonal projection.

  (i) On a subspace W spanned by an orthonormal set of vectors $\{q_1, \dots, q_n\}$ $(n < m)$.

$$P_W = Q Q^*, \quad \text{where } Q = \begin{bmatrix} | & & | \\ q_1 & \cdots & q_n \\ | & & | \end{bmatrix}.$$

(ii) Orthogonal projection on 1-dimensional subspace spanned by $q \in V$, $\|q\| = 1$: $P_q = qq^*$ (rank 1 matrix).

(iii) On a subspace A ~~spanned by~~ with basis $\{a_1, \ldots, a_n\}$

$$P_A = A(A^*A)^{-1}A^*, \quad A = \left[\, a_1 \,|\, \cdots \,|\, a_n \,\right]$$

(iv) On a 1-dimensional subspace spanned by $a \in V$,

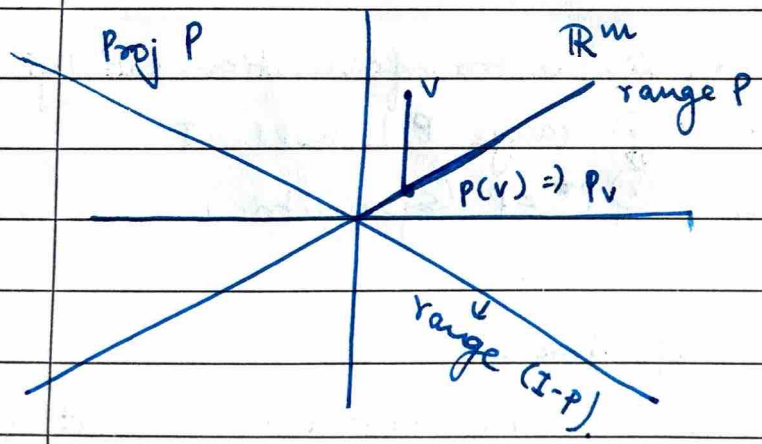
$$\boxed{P_a = \frac{aa^*}{\|a\|}} = \frac{aa^*}{a^*a}. \qquad \because \|a\| = a^*a.$$

(v) $I - P_W$, $I - P_V$, $I - P_A$, $I - P_a$

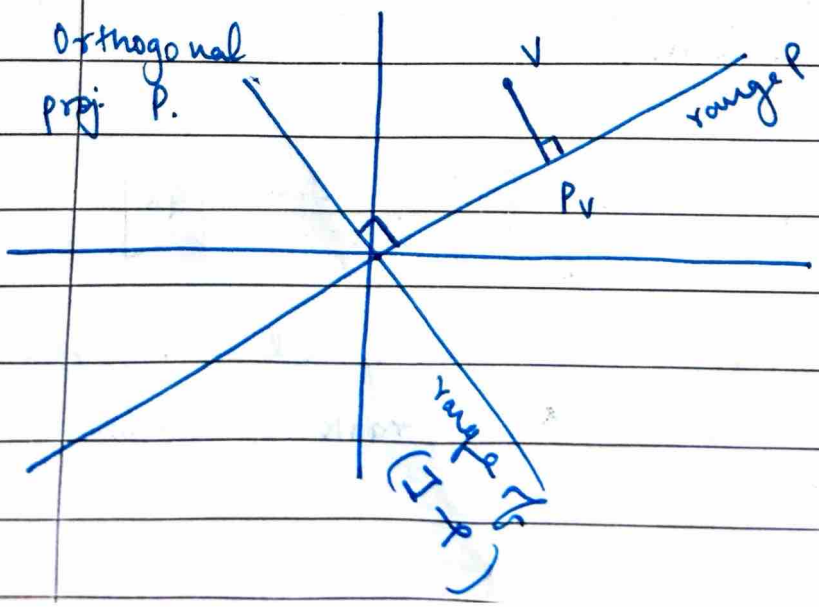
$\parallel$ $\parallel$ $\parallel$ $\parallel$

$P_{\perp W}$ $P_{\perp V}$ $P_{\perp A}$ $P_{\perp a}$.

★ **Schematic picture**



General projection.



Orthogonal projection.

★ QR factorization

Defn: Given $A \in \mathbb{R}^{m \times n}$ $(m \geq n)$, the QR factorization of A is a factorization of the form

$$\underset{m \times n}{[A]} = \underset{m \times n}{[Q]} \; \underset{n \times n}{[R]}$$

Q: orthogonal, R: upper triangular

$$\left[\begin{array}{c|c} * & * \\ \ddots & \vdots \\ \hline 0 & * \end{array}\right]$$

Full QR factorization.

When

$$[A]_{m \times n} = \underset{m \times n}{[Q]} \; \underset{n \times n}{[R]} \quad \text{(Reduced QR).}$$

Suppose

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} q_{11} & \cdots & q_{1n} \\ \vdots & & \\ q_{m1} & \cdots & q_{mn} \end{bmatrix} \begin{bmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \\ 0 & & r_{nn} \end{bmatrix}$$

$$= \begin{bmatrix} r_{11} q_{11} \\ r_{11} q_{21} \\ \vdots \\ r_{11} q_{n1} \end{bmatrix}$$

$a_{i1} = r_{11} q_{i1}$

~~[illegible]~~

$a_1 = r_{11} q_1$

$\Rightarrow \boxed{q_1 = a_1 / r_{11}}$

$a_2 = r_{12} q_1 + r_{22} q_2$

$\therefore q_2 = \dfrac{a_2 - r_{12} q_1}{r_{22}}$

~~[illegible]~~

$a_n = r_{1n} q_1 + \cdots + r_{nn} q_n$

So, $q_n = \dfrac{a_n - r_{1n} q_1 - r_{2n} q_2 \cdots - r_{(n-1),n} q_{n-1}}{r_{nn}}$

Since, columns of Q are orthonormal,

$\|q_1\| = 1 \Rightarrow r_{11} = \|a_1\|$

$\|q_j\| = 1 \Rightarrow r_{jj} = \| a_j - \sum_{i=1}^{j-1} r_{ij} q_i \|$ } Grahm Schmidt orthonormalization.

$q_i^* a_j$

Thus, Grahm Schmidt orthonormalization process gives rise to QR factorization

__/__/__

Theorem: Every $A \in \mathbb{C}^{m \times n}$ $(m \geq n)$ (WLOG) has a full QR factorization & hence also a reduced QR factorization.

{ If A is full rank, then G-S orthonormalization gives QR factorisation.

At each step we can construct q_j, since $a_j \neq 0 \ \forall j$

Recall: In G.S. orthonormalization given a set $A = \{a_1, \dots, a_n\}$ produce an orthonormal set $\{q_1, \dots, q_n\}$ satisfying

① $\|q_i\| = 1 \quad \forall i$

② $q_i^* q_j = 0 \quad \forall i \neq j$

③ $\text{span}\{a_1 \dots a_k\} = \text{span}\{q_1, \dots, q_k\} \quad \forall \ 1 \leq k \leq n.$

If A does not have full rank, then for some j,

$$v_j = a_j - \sum_{i=1}^{j-1} r_{ij} q_i$$

$$= a_j - \sum_{i=1}^{j-1} (q_i^* q_i) a_j = 0.$$

Then we can choose q_j arbitrarily satisfying

$$a_j \perp \{q_1, \dots, q_{j-1}\}$$

Then continue with next columns as usual.

$$a_j - \sum (q_i^* a_j) q_i$$

$$\downarrow$$

$$a_j - \sum_{i=1}^{j-1} (q_i q_i^*) a_j = v_j$$

* Uniqueness of QR factorization

⇒ Unique upto the sign of elements of diagonal of R. for full rank A.

⇒ If $A = QR$ is a reduced QR factorization of A, then for any $z \in \mathbb{C}$ s.t. $|z| = 1$, multiplying the i^{th} column of Q by z & i^{th} row of R by z^{-1}.

In particular the sign of r_{jj} can be +ve or -ve.

If A is full rank & $r_{jj} > 0 \ \forall j$, then the QR factorization of A is unique.

* Algorithm for Classical Gram Schmidt (CGS)

Only the inner loop.

```
for j = 1 to n
    v_j = a_j
    for i = 1 to j-1
        r_ij = q_i * a_j
        v_j = v_j - r_ij q_i
    r_jj = ||v_j||_2  (2-norm)
    q_j = v_j / r_jj
```

__/__/__

* **Modified Grahm Schmidt**
for ensuring Numerical stability.

MGS

for $i = 1$ to n:
$r_i = q_i$
$r_{ii} = \|v_i\|$
$q_i = v_i / r_{ii}$
for $j = i+1$ to n
$r_{ij} = q_i^* v_j$
$v_j = v_j - r_{ij} q_i$

Operation count $\approx 2mn^2$.

* **A high level view of Grahm Schmidt orthonormalization**

Given $A = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}$

G.S. $\leftrightarrow$ right multiplying A with suitable upper triangular matrices to obtain a matrix with orthogonal columns.

$(A R_1 R_2) = Q$
$A R^{-1} = Q$
$\boxed{A = QR}$

where $R_1 = \begin{bmatrix} \frac{1}{r_{11}} & -\frac{r_{12}}{r_{11}} & -\frac{r_{13}}{r_{11}} & \cdots \\ & 1 & & \\ & & & \\ & & & 1 \end{bmatrix}$

$R_2 = \begin{bmatrix} 1 & & & \\ & \frac{1}{r_{22}} & -\frac{r_{23}}{r_{22}} & \\ & & & \\ & & & 1 \end{bmatrix}$

$R_3 = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \frac{1}{r_{33}} & -\frac{r_{34}}{r_{33}} \\ & & & \ddots \\ & & & 1 \end{bmatrix}$ & so on.

* **Householder's method for QR factorization**

In contrast, Householder's method is to premultiply A with certain unitary matrices Q_k so that

$Q_n \cdots Q_1 A$ is an upper triangular matrix R

$$(\underbrace{Q_n \cdots Q_1}_{Q^{-1}}) A = R$$

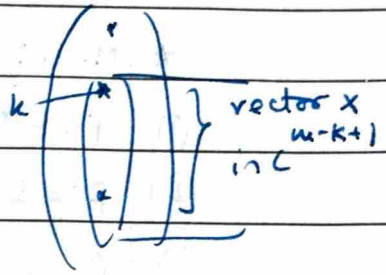
$$Q^{-1} A = R$$

$$\boxed{A = QR}$$

★ Householder constructed the matrix Q_k as follows:

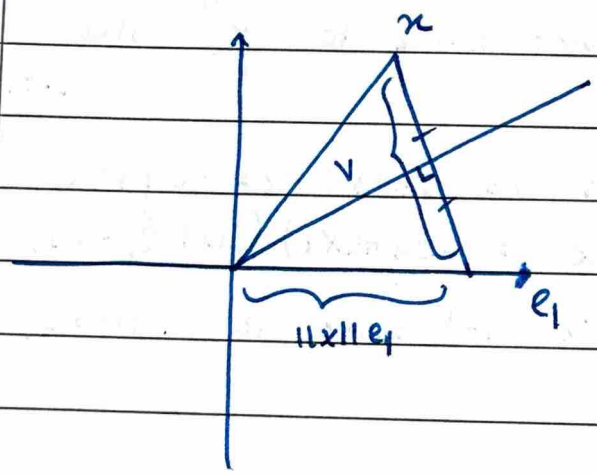
$$Q_k = \begin{bmatrix} I_{k-1} & 0 \\ 0 & F \end{bmatrix}$$ where F is a $(m-k+1) \times (m-k+1)$ matrix such that F introduces zeroes below the diagonal in the k^{th} column of $(Q_{k-1} \cdots Q_1) A$

Suppose the k^{th} column of $(Q_{k-i} \cdots Q_1 A)$ is



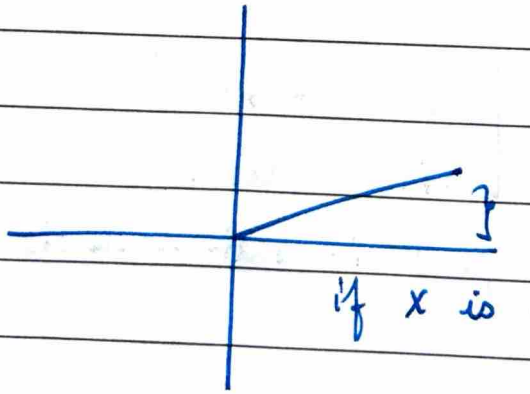
$$\begin{pmatrix} x_1 \\ \vdots \\ x_{m-k+1} \end{pmatrix} \quad x \longmapsto \|x\| e_1 \quad \begin{pmatrix} \|x\| \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (m-k+1).$$

x_1 is first component.



$$P_{\perp v} = I - \frac{vv^*}{v^*v} \qquad v = \|x\| e_1 - x$$

$$F = I - \frac{2vv^*}{v^*v}$$ Householder reflections.



$$v = -sg(x_1) \|x\| e_1 - x.$$

If x is close to e_1

Householder's idea was to choose a vector v such that the reflection across v maps x to $\|x\|e_1$

If v is taken to be $v = \pm \|x\|e_1 - x$ then $F = I - \frac{2vv^*}{v^*v}$

satisfies the purpose.

* **Note:**

① $F = F^*$

② $F^2 = I$

③ F is unitary (full rank)

④ For mathematical purposes we can choose any z s.t $|z| \cdot \|x\|e_1$ is not close to x, where $z \in \mathbb{C}$ s.t. $|z| = 1$.

But for numerical purposes we choose $z = -\operatorname{sgn}(x_1)$, so that

$$v = -\operatorname{sgn}(x_1)\|x\|e_1 - x = -\operatorname{sgn}(x_1)(\|x\|e_1 + x).$$

This makes sure that v is not close to 0. Hence, we avoid cancellation errors.

* **Givens' Rotations**

A Givens rotation $R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$

rotates any vector x through θ in anti-clockwise direction.

For a vector $x = \begin{pmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_j \\ \vdots \\ x_n \end{pmatrix}$ $\begin{matrix} \\ \\ \to x_i' \\ \\ \to x_j \\ \\ \end{matrix}$

For a vector x, the aim is to introduce '0' at the j^{th} spot affecting a change in the i^{th} spot, without affecting the rest of the entries.

let $R(i,j,\theta) =$ $\begin{matrix} & & i & & j \\ i & \cdots & \cos\theta & \cdots & -\sin\theta \\ j & \cdots & \sin\theta & \cdots & \cos\theta \end{matrix}$ s.t. $i < j$

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_i \\ x_j \end{bmatrix} = \begin{bmatrix} \sqrt{x_i^2 + x_j^2} \\ 0 \end{bmatrix}$$

$$\cos\theta = \frac{x_i}{\sqrt{x_i^2 + x_j^2}} \quad \& \quad \sin\theta = \frac{-x_j}{\sqrt{x_i^2 + x_j^2}}$$

Rotation matrix preserves length.

★ **Givens' rotations**

For a vector $x = \begin{pmatrix} x \\ x \\ x_i \\ \vdots \\ x_j \\ x \\ x \end{pmatrix} \begin{matrix} \\ \\ \leftarrow i \\ \\ \leftarrow j \\ \\ \\ \end{matrix}$

we can zero out x_j
by affecting a change in x_i
(while not changing the 2 norm of x)
using

$$R(i,j,\theta) = \begin{matrix} & & i & & j \\ & 1 & & & \\ i & \cdots & \cos\theta & \cdots & -\sin\theta \\ & & \vdots & & \\ j & \cdots & \sin\theta & \cdots & \cos\theta \end{matrix} \quad \text{s.t.}$$

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_i \\ x_j \end{bmatrix} = \begin{bmatrix} \sqrt{x_i^2 + x_j^2} \\ 0 \end{bmatrix}$$

$$\cos\theta = \frac{x_i}{\sqrt{x_i^2 + x_j^2}}$$

$$\sin\theta = \frac{-x_j}{\sqrt{x_i^2 + x_j^2}}$$

eg. to proceed from.

$$\begin{pmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & x \\ 0 & 0 & x \\ 0 & 0 & x \end{pmatrix} \text{ to } \begin{pmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & x \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

we proceed to 2-steps

$$\begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & c & -s \\ & & & s & c \end{pmatrix} \begin{pmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & x \\ 0 & 0 & x \\ 0 & 0 & x \end{pmatrix} = \begin{pmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & x \\ 0 & 0 & x \\ 0 & 0 & 0 \end{pmatrix}$$

Then apply,

$$\begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & c & -s & \\ & & s & c & \\ & & & & 1 \end{pmatrix} \begin{pmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & x \\ 0 & 0 & x \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & x \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

* **Iterative methods for solving systems of linear equations**

① Jacobi

② Gauss Siedel

③ Successive over relaxation (SOR).

To solve,

$Ax = b$.

$$+\frac{\sqrt{3}}{1} + \frac{7}{\sqrt{3}} = \frac{-3+7}{2\sqrt{3}} = \frac{4}{2\sqrt{3}}$$

__/__/__

Generalities: Assume $Ax=b$ has a unique solution.

Defn: A splitting of A is a decomposition $A = S-T$

$A = S-T$, where S is non-singular.

A splitting yields an iterative method as follows:

$$Ax=b.$$
$$(S-T)x=b.$$
$$Sx - Tx = b.$$
$$Sx = Tx + b.$$
$$x = S^{-1}Tx + S^{-1}b.$$

This suggest the following iterative method

$$x_{k+1} = Rx_k + c$$

where,
$R = S^{-1}T \rightarrow$ matrix of the iterative method.
& $c = S^{-1}b \rightarrow$ vector

Lemma 1: Let $\|.\|$ any induced matrix norm.
If $\|R\|<1$, then $x_{k+1} = Rx_k + c$ converges for any initial solution x_0

Proof:

$$x_{k+1} = Rx_k + c$$
$$- \quad x = Rx + c$$
$$(x_{k+1} - x) = R(x_k - x)$$

Now,

$$\|x_{k+1} - x\| = \|R(x_k - x)\| \leq \|R\| \, \|x_k - x\|$$
$$\leq \|R\| \, \|R\| \, \|x_{k-1} - x\|$$
$$\vdots$$
$$\leq \|R\|^{k+1} \, \|x_0 - x\|$$

$\|R\|<1 \Rightarrow \|R\|^{k+1} \rightarrow 0$ as $k \rightarrow \infty$

i.e. $\|x_{k+1} - x\| \rightarrow 0$ as $k \rightarrow \infty$ i.e. $\{x_{k+1}\} \rightarrow x$ as $k \rightarrow \infty$

__/__/____

Lemma 2: Let $\rho(R)$ denote the spectral radius of R

$\left(\rho(R) = \max\{|\lambda| / \lambda \text{ is an eigenvalue of } R\}\right)$

① $\rho(R) \le \|R\|$, for every induced matrix norm. (depends on R & ε).

② $\forall\ \varepsilon > 0$, $\exists\ \|\cdot\|_*$ such that

$$\|R\|_* = \rho(R) + \varepsilon.$$

Proof: ① Let λ be the e-value of R for which $|\lambda| = \rho(R)$

let x be the corresponding eigen vector i.e. $Rx = \lambda x$

$$\|R\| = \max_{g \neq 0} \frac{\|Rg\|}{\|g\|} \ge \frac{\|Rx\|}{\|x\|} = \frac{\|\lambda x\|}{\|x\|} = |\lambda| = \rho(R)$$

② (Lemma 6.5 in Demmel's book).

Theorem: The iterative method $x_{k+1} = Rx_k + c$.

converges to the solution of $Ax = b$ $\forall x_0$ & $\forall b$.

$\Leftrightarrow \rho(R) < 1$.

Proof: If $\rho(R) \ge 1$, let $|\lambda| = \rho(R)$ & let $Ry = \lambda y$.

Suppose $y + x = x_0$, so $y = x_0 - x$.

then

$$Ry = R(x_0 - x) = \lambda(x_0 - x).$$

$$x_{k+1} - x = R(x_k - x)$$

$$= R^2 (x_{k-1} - x)$$

$$\vdots$$

$$= R^{k+1}(x_0 - x) = \lambda^{k+1}(x_0 - x) \to 0 \quad \text{as } k \to \infty$$

Then in a contradiction to the assumption $\rho(R) < 1$.

Conversely, if $\rho(R) < 1$.
choose a norm $\|.\|_*$ such that $\|R\|_* < 1$, then by lemma 1 the iterative method converges.

Questions:-

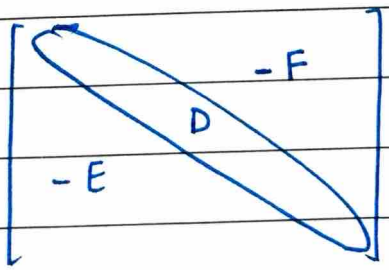
Questions to be considered

1. Is the given method convergent?
   - Does $\exists$ a matrix norm in which $\|R\| < 1$? or $\rho(R) < 1$?
2. Given 2 iterative methods which one converges faster ?
   Smaller spectral radius $\Rightarrow$ faster convergence.
   i.e. Which matrix gives smaller spectral radius ?

I Jacobi's Method

Assume, $a_{ii} \neq 0 \ \forall i$.

split A as $D - E - F$ where $D = \text{diag}(A)$
$-E$ = lower Δ of A
$-F$ = upper Δ of A.

$A =$ 

Now, let $D = S$.
and let $T = E + F$

__/__/____

$$Ax = b$$

$$(D - E - F)\,x = b.$$

$$Dx - (E+F)\,x = b.$$

$$x = \underbrace{\underbrace{D^{-1}}_{S}\underbrace{(E+F)}_{T}}_{J} x + \underbrace{D^{-1}b}_{C}$$

Jacobi's method

$$x_{k+1} = Jx_k + c.$$

where $J = D^{-1}(E+F)$

& $c = D^{-1}$

Notice that $J = D^{-1}(E+F)$

$$= D^{-1}E + D^{-1}F$$

$$= I - D^{-1}D + D^{-1}E + D^{-1}F$$

~~$= I + D^{-1}(-E-F+D)$~~

~~$= I - D^{-1}$~~

~~$= I - D^{-1}(-D+E+F)$~~

~~$= I - D^{-1}A.$~~

$$= I - D^{-1}(D - E - F)$$

$$= I - D^{-1}A.$$

$n \times n$ $\quad n \times 1$

$$Ax = b.$$

Initial soln $\quad x_0 = \begin{pmatrix} x_1^0 \\ x_2^0 \\ \vdots \\ x_n^0 \end{pmatrix}$

__/__/____

$n \times n \quad n \times 1$

$Ax = b$

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ \vdots \\ a_{j1}x_1 + a_{j2}x_2 + \cdots + a_{jn}x_n = b_j \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

Initial soln $D^{-1} = \begin{pmatrix} 1/a_{11} & & \\ & \ddots & \\ & & 1/a_{nn} \end{pmatrix}$

$$C = D^{-1}b = \begin{pmatrix} b_1/a_{11} \\ b_2/a_{22} \\ \vdots \\ b_n/a_{nn} \end{pmatrix}$$

Initial soln $x_0 = \begin{pmatrix} x_1^0 \\ x_2^0 \\ \vdots \\ x_n^0 \end{pmatrix}$

$$x_1 = (I - D^{-1}A)x_0 + C.$$

$$x_0 - D^{-1}Ax_0$$

$$x_1^1 = x_0^1 - \left(x_0^1 + \frac{a_{12}}{a_{11}}x_0^2 + \cdots + \frac{a_{1n}}{a_{11}}x_0^n\right) + \frac{b_1}{a_{11}}$$

$$= \frac{1}{a_{11}}\left(b_1 - a_{12}x_0^2 - \cdots - a_{1n}x_0^n\right)$$

$$x_1^1 = x_1^0 - \left(x_1^0 + \frac{a_{12}}{a_{11}}x_2^0 + \cdots + \frac{a_{1n}}{a_{11}}x_n^0\right) + \frac{b_1}{a_{11}}$$

$$= \frac{1}{a_{11}}\left(b_1 - a_{12}x_2^0 - \cdots - a_{1n}x_n^0\right)$$

$x_i^{j} = \frac{1}{a_{jj}} \left(b_j - \right.$ [illegible]

$x_j^{1} = \frac{1}{a_{jj}} \left(b_j - a_{j1} x_j^{0} - \overbrace{a_{jj} x_j}^{\text{(not present represented by a hat)}} - \cdots - a_{jn} x_n^{0} \right)$

$x_{k+1} = (I - D^{-1}A)\, x_k + c.$

$x_1^{k+1} = \frac{1}{a_{11}} \left(b_1 - a_{12} x_2^{k} - \cdots - a_{1n} x_n^{k} \right)$

$x_2^{k+1} = \frac{1}{a_{22}} \left(b_2 - a_{21} x_1^{k} - \overbrace{a_{22} x_2^{k}} - \cdots - a_{2n} x_n^{k} \right)$

$\vdots$

$x_i^{k+1} = \frac{1}{a_{ii}} \left(b_i - a_{i1} x_1^{k} - a_{i2} x_2^{k} - \cdots - \overbrace{a_{ii} x_i^{k}} - \cdots a_{in} x_n^{k} \right)$

$x_n^{k+1} = \frac{1}{a_{nn}} \left(b_n - a_{n1} x_1^{k} + a_{n2} x_2^{k} - \cdots - a_{n,n-1} x_{n-1}^{k}, \overbrace{a_{nn} x_n^{k}} \right)$

* Algorithm for one step of Jacobi's method

for $j = 1$ to n

$$x_j^{k+1} = \frac{1}{a_{ji}} \left(b_j - \sum_{\substack{i=1 \\ i \neq j}}^{n} a_{ji} x_i^k \right)$$

* Gauss-Siedel

$$x_1^{(k+1)} = \frac{1}{a_{11}} \left(b_1 - a_{12} x_2^{(k)} - a_{13} x_3^{(k)} - \dots - a_{1n} x_n^{(k)} \right)$$

$$x_2^{(k+1)} = \frac{1}{a_{22}} \left(b_2 - a_{21} x_1^{(k+1)} - a_{22} x_2^{(k)} - a_{23} x_3^{(k)} - a_{2n} x_n^{(k)} \right)$$

$$\vdots$$

$$x_j^{(k+1)} = \frac{1}{a_{jj}} \left(b_j - a_{ji} x_i^{(k+1)} - a_{jj} x_j^{(k)} \dots a_{jn} x_n^{(k)} \right)$$

$$x_n^{(k+1)} = \frac{1}{a_{nn}} \left(b_n - a_{n1} x_j^{(k+1)} - \dots - a_{n\,n-1} x_{n-1}^{(k+1)} \right)$$

The iterative method corresponds to

$$X^{k+1} = D^{-1} (E X^{k+1} + f X^k + b)$$

$$D X^{k+1} = E X^{k+1} + f X^k + b.$$

$$(D - E) X^{k+1} = f X^k + b.$$

$$X^{k+1} = \underbrace{(D-E)^{-1} f X^k}_{L_1} + \underbrace{(D-E)^{-1} b}_{C}$$

__/__/____

* **Algorithm for one step Gauss-Siedel method**

for $j = 1$ to n ← updated terms.

$$x_j^{k+1} = \frac{1}{a_{jj}} \left(b_j - \sum_{i=1}^{j-1} a_{ji}\, x_i^{(k+1)} - \sum_{i=j+1}^{n} a_{ji}\, x_i^{(k)} \right)$$

III) **Successive over-relaxation (SOR)**

ω: relaxation parameter ($\neq 0$)

Splitting of A.

$$A = \underbrace{\left(\frac{D}{\omega} - E \right)}_{S} - \underbrace{\left(\frac{1-\omega}{\omega} D + F \right)}_{T}$$

If	$\omega > 1$ over relaxing
	$\omega < 1$ under relaxing
	$\omega = 1$ Gauss Siedel.

The associated iterative method, is

$$X^{k+1} = L_\omega X^k + c,$$

where, $L_\omega = \left(\frac{D}{\omega} - E \right)^{-1} \left(\frac{1-\omega}{\omega} D + F \right)$

& $c = \left(\frac{D}{\omega} - E \right)^{-1} b.$

19/01/02

$$x_1^{(k+1)} = \frac{1}{a_{11}}\left(a_{11}x_1^{(k)} - \omega\left(a_{11}x_1^{(k)} + \cdots + a_{1n}x_n^{(k)} - b_1\right)\right)$$

$$= (1-\omega)\,x_1^{k} + \frac{\omega}{a_{11}}\left(b_1 - a_{12}x_2^{(k)} - \cdots - a_{1n}x_n^{(k)}\right)$$

$$x_2^{(k+1)} = \frac{1}{a_{22}}\left(a_{22}x_2^{(k)} - \omega\left(a_{21}x_1^{(k+1)} + \cdots a_{2n}x_n^{(k)} - b_2\right)\right)$$

$$= (1-\omega)\,x_2^{(k)} + \frac{\omega}{a_{22}}\left(b_2 - a_{21}x_1^{(k+1)} - \cdots - a_{2n}x_n^{(k)}\right)$$

$$x_n^{(k+1)} = (1-\omega)\,x_n^{(k)} + \frac{\omega}{a_{nn}}\left(b_n - a_{n1}x_1^{(k+1)} - \cdots a_{nn-1}x_{n-1}^{(k+1)}\right)$$

Defn: The rate of convergence of $x^{(k+1)} = Rx^{(k)} + c$ is $r(R) = -\log \rho(R)$ (smaller the value of $\rho(R)$ faster the convergence)

sections 6.5.3 & 6.5.4 of Demmel.

★ **Jacobi's method**

① # allocations at each step = $2n$.

② Can compute all components of $x^{(k+1)}$ in parallel.

③ Assumes that the diagonal entries are non-zero.

④ Does not always converge, it will converge for certain strictly diagonally dominant matrices.

★ **Strictly diagonally dominant matrices**

① a_{ii} > sum of rest of the row entries $\forall i$.

★ **Properties of Gauss Siedel method**

1) # allocations at each step = n.
2) Computation of components of $x^{(k+1)}$ is not parallelizable.
3) Converges if A is HPD.

★ Common requirement : non-zero diagonal entries (which can be attained by rearranging eqns if required).

★ **Least Squares Method** (for solving overdetermined $Ax=b$)

meq^{n}s n unknowns

$\therefore m>n$.

2 cases — A is full rank

A is rank deficient.

If $m>n$, then in general the system of eq^{n}s $Ax=b$ does not have a sol^n.

The aim is then to find x that minimizes the residual

$$r = Ax-b.$$

The problem is stated as follows:-

given $A \in \mathbb{C}^{m\times n}$, $m>n$, & $b \in \mathbb{C}^m$, find $x \in \mathbb{C}^n$ such that

$\|b - Ax\|_2$ is minimized.

__/__/____

* (motivation for a general idea)

* **Polynomial least squares**

Given m distinct pts. $x_1, \ldots, x_m$ & data $y_1, \ldots, y_m$ at these pts. essentially

$$((x_1, y_1) \ldots (x_m, y_m))$$

find a polynomial $p(x)$ which fits this information in a way so that

$$p(x_1) = y_1$$
$$p(x_2) = y_2$$
$$\vdots$$
$$p(x_m) = y_m$$

i.e. $\sum |p_i(x) - y_i|^2$ is minimized

Suppose, $p(x) = c_0 + c_1 x + \cdots c_{n-1} x^{n-1}$

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & & & & \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & & & & \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

★ **Least squares problems**

In general, we find a pt. $Ax \in \text{range } A$ s.t. $r = b - Ax$ is minimized.

Geometrically, the idea is to obtain Ax (or x)

~~$Ax = Pb$, (where P is the orthogonal projection on range A).~~

~~where b is orthogonal projection~~

~~where R is the orthogonal projection of b on range of A.~~

by orthogonal projection of b on range A.

P is orthogonal projection on range A.

Theorem: $A \in \mathbb{C}^{m \times n}$, $m \geq n$

A vector $x \in \mathbb{C}^n$ minimizes $\|r\|_2 = \|Ax - b\|_2 \Leftrightarrow r \perp \text{range}$

i.e. $A^* r = 0$.

Proof: Note that for $b \in \mathbb{C}^m$, we can write b as $b_1 - b_2$

where $b_1 \in \text{range } A$ $b_2 \in \text{null } A^*$

$Ax - b_1 \in \text{range } A$.

~~$r = Ax - b_1 + b_2$~~

$r = Ax - b = Ax - b_1 + b_2$ & $b_2 \perp Ax - b_1$

($Ax - b_1$: range A; b_2: null A^*)

$$\|Ax - b\|_2^2 = \|Ax - b_1\|_2^2 + \|b_2\|_2^2$$

Since b_2 is fixed, to minimize $\|Ax - b\|_2^2$, we must minimize $\|Ax - b_1\|_2^2$ $\|Ax - b_1\|_2^2$ is minimized $\Leftrightarrow Ax = b_1$

$Ax = b_1 \Leftrightarrow r = Ax - b = b_2 \in \text{null}(A^*) \Leftrightarrow r \perp \text{range } A$.

this is the content of the earlier theorem. __/__/____

Corollary $r \perp \text{range } A \iff x$ minimizing $r = b - Ax$

$\iff A^* Ax = A^* b.$

Proof: Suppose $r \perp \text{range } A$.

let x be s.t. $r = b - Ax$ is minimal.

$A^* r = A^*(b - Ax) = A^* b - A^* Ax = 0.$

Conversely if $A^* Ax = A^* b.$ $\rightarrow$ then $A^* \underbrace{(Ax - b)}_{r} = 0.$

$\Rightarrow r \perp \text{range } A.$

System of normal eq's for LSP $Ax = b$ (A is full rank).

Remarks: ① To see that Ax is indeed the orthogonal projection of b on range A,

Consider $P = A(A^*A)^{-1}A^*$

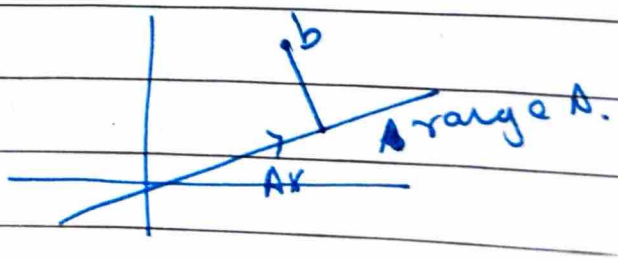
Then $Pb = A(A^*A)^{-1}A^* b = A(A^*A)^{-1}A^*Ax$

$= Ax.$

Theorem: The system of normal eq's $A^* Ax = A^* b$ has a unique solution $\Leftrightarrow A$ is full rank.

To summarize: If A has full rank, the soln to the LSP is unique & is given by

$x = \underbrace{(A^*A)^{-1}A^*}_{A^+} b = A^+ b$, $A^+ =$ psuedoinverse of A.



__/__/____

* Computing x_{LS}

There are 3 popular ways of computing x_{LS}:

I Solve the normal eq^{n}s

when A is full rank

A^*A is Hermitian Positive Definite

So, $A^*Ax = A^*b$

can be solved using **Cholesky** factorization.

1. form A^*A & A^*b.
2. find R (upper triangular) $A^*A = R^*R$
3. Solve $R^*w = A^*b$
4. then solve for $Rx = w$.

Operation count is dominated by the first 2 steps

$\approx O(mn^2 + n^3)$ $\approx O(mn^2 + \frac{n^3}{3})$

II **QR factorization**

If $A = QR$.

then $P = QQ^*$

so, $y = Pb = QQ^*b$.

Since, $y \in$ range A, $Ax = y$ for some x.

$QRx = QQ^*b$.

$\boxed{Rx = Q^*b}$

$y \in$ range A so we have unique solution.

III Using SVD.

If $A = U\Sigma V^*$

Then $P = UU^*$

$y = Pb = UU^*b.$

$\exists\ x$ s.t. $Ax = y$ ($\because y$ belongs to range A).

$U\Sigma V^* x = UU^*b.$

$$\boxed{\Sigma V^* x = U^* b}$$

★ **Rank deficient LSP's**

If A is rank deficient, then there are infinitely many solutions for the corresponding LSP.

Consider the system, $Ax = b$, where $A \in \mathbb{C}^{m \times n}$, $m \geq n$.

& rank $A = r < n$.

If x minimizes $\|Ax - b\|$ $x + z$, where $z \in$ null(A) also minimizes $Ax - b$.

To solve a rank deficient LSP, we consider a "complete orthogonal factorization" i.e. orthogonal matrices Q & Z s.t.

$$Q^T A Z = \begin{bmatrix} T_{11} & 0 \\ 0 & 0 \end{bmatrix} \begin{matrix} r \\ \} m-r \end{matrix}$$

$\underbrace{\qquad}_{r} \quad \underbrace{\qquad}_{n-r}$

If such a factorization is found then the

$$\|Ax - b\|_2^2 = \|AZZ^Tx - b\|_2^2$$

$$= \|Q^T AZZ^T x - Q^T b\|_2^2$$

$$= \quad \text{(where } c = Z^T x \text{ and } d = Q^T b\text{)}$$

$= \| Q^T Z Z^T A x - Q^T b \|_2^2$ (c, d)

$= \| T_{11} w - c \|_2^2 + \| d \|_2^2$

w is first r components of

$= \| Q^T A Z Z^T x - Q^T b \|_2^2$

$\| Ax - b \|_2^2 = \| T_{11} w - c \|_2^2 + \| d \|_2^2$ → This $Q^T b = \begin{pmatrix} c \\ d \end{pmatrix}$

$$\boxed{\| Ax - b \|_2^2 = \| T_{11} w - c \|_2^2 + \| d \|_2^2}$$

Intermediate Step $\Rightarrow \| Q^T A Z Z^T x - Q^T b \|_2^2 = \| T_{11} w - c + d \|_2^2$

$c = \begin{pmatrix} x \\ \vdots \\ x \\ 0 \\ \vdots \\ 0 \end{pmatrix}$ } first r components of $Q^T b$.

$d = \begin{pmatrix} 0 \\ 0 \\ x \\ \vdots \\ x \end{pmatrix}$ } r, } n-r last n-r components of $Q^T b$.

If x is to minimize $\| Ax - b \|_2^2$, then

w must minimize $\| T_{11} w - c \|_2^2$

$\| T_{11} w - c \|_2^2 = 0.$

$w = T_{11}^{-1} c$ → works.

We can choose y to be zero so that $Z^T x = \begin{pmatrix} T_{11}^{-1} c \\ 0 \end{pmatrix}$ } r, } n-r

So $$\boxed{x_{LS} = Z \begin{pmatrix} T_{11}^{-1} c \\ 0 \end{pmatrix}}$$

☆ Rank-deficient LSP's

Last time: if A has a "complete orthogonal factorisation"

$$A = Q \begin{bmatrix} T_{11} & 0 \\ 0 & 0 \end{bmatrix} Z^T \quad \text{i.e.} \quad Q^T A Z = \begin{bmatrix} T_{11} & 0 \\ 0 & 0 \end{bmatrix}$$

then the LS solution

$$x_{LS} = Z \begin{pmatrix} T_{11}^{-1} c \\ 0 \end{pmatrix} \begin{matrix} \} \text{rank } A = r \\ \} n-r \end{matrix}, \quad \&$$

$$Q^T b = \begin{pmatrix} c \\ d \end{pmatrix} \begin{matrix} \} r \\ \} n-r. \end{matrix}$$

Theorem: Suppose $A = U \Sigma V^*$ is the SVD of $A \in \mathbb{C}^{m \times n}$, $r = \text{rank } A$.

If $U = \begin{bmatrix} u_1 & \cdots & u_m \end{bmatrix}$, $V = \begin{bmatrix} v_1 & \cdots & v_n \end{bmatrix}$ & $b \in \mathbb{C}^m$,

then the j^{th} component of x_{LS} is given by,

$$(x_{LS})_j = \sum_{i=1}^{r} \frac{u_i^T b}{\sigma_i} v_{ji}.$$

x_{LS} minimizes $\|Ax - b\|_2$.

This x_{LS} has the smallest 2-norm among all minimizers.

Proof: This is the re-wording of the earlier proof in the case of SVD.

$$\|Ax - b\|_2^2 = \|U^T A V V^T x - U^T b\|_2^2$$

$$\|Ax - b\|_2^2 = \|U^T A x - U^T b\|_2^2 = \|U^T A V V^T x - U^T b\|_2^2$$

$$= \|\Sigma w - U^T b\|_2^2$$

$$= \left(\sum_{i=1}^{m} \sigma_i w_i - u_i^T b \right)^2$$

Since we have no control over

$$\sum_{i=1}^{m} (\sigma_i w_i - u_i^T b)^2 = \sum_{i=1}^{r} (\sigma_i w_i - u_i^T b)^2 + \sum_{i=r+1}^{m} (u_i^T b)^2$$

Since we have no control over $\sum_{i=r+1}^{m} (u_i^T b)^2$

To minimize $\|Ax-b\|_2^2$, we minimize $\sum_{i=1}^{r} (\sigma_i w_i - u_i^T b)^2$

Equating $\sum_{i=1}^{r} (\sigma_i w_i - u_i^T b)^2 = 0$ gives each $\sigma_i w_i = u_i^T b$ i.e.

$$w_i = \frac{u_i^T b}{\sigma_i}$$

let $w_i = \begin{cases} \frac{u_i^T b}{\sigma_i} & \text{for } 1 \le i \le r \\ 0 & \text{for } r+1 \le i \le m. \end{cases}$

So $w = \begin{pmatrix} \frac{u_1^T b}{\sigma_1} \\ \vdots \\ \frac{u_r^T b}{\sigma_r} \\ 0 \\ 0 \\ 0 \end{pmatrix}$

$$\therefore x_{LS} = Vw = V \begin{pmatrix} \frac{u_1^T b}{\sigma_1} \\ \vdots \\ \frac{u_r^T b}{\sigma_r} \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{r} \frac{u_i^T b}{\sigma_i} v_{1i} \\ \\ \sum_{i=1}^{r} \frac{u_i^T b}{\sigma_i} v_{mi} \end{pmatrix}$$

(Check) As in the case of full rank LSPs, this soln can also be expressed as

$$x_{LS} = A^+ b.$$

where $A^+ = V \Sigma^+ U^T$ $\quad \Sigma^+ = \begin{pmatrix} 1/\sigma_1 & & & \\ & \ddots & & \\ & & 1/\sigma_r & \\ & & & \ddots \\ & & & & 0 \end{pmatrix}$

II. QR with column pivoting

Usual $A = QR$

Why is column pivoting required?

$$A = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \end{bmatrix}_{m \times 4} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}_{m \times 4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{4 \times 4}$$

rank $a = 3$.

$$\begin{bmatrix} q_1 & q_1 & q_1 + q_2 & q_1 + q_2 + q_3 + q_4 \end{bmatrix}$$

a_1 a_2 a_3 a_4

range (A)

observe that range(A) is not in the span of any 3 columns of Q.

range $\{q_1, q_2, q_3\}$

The QR factorisation gives no info about rank A (i.e. about null(A)).

$A\Pi = QR$

↳ permutation matrix so that we have linearly independent columns to the left.

The algorithm produces the factorisation

$$Q^T A \Pi = \underset{\text{permutation.}}{\downarrow} \left[\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & R_{22} \end{array}\right] = \left[\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & 0 \end{array}\right] \begin{matrix} \}r \\ \}n-r. \end{matrix}$$

R is rank A

Q is orthogonal

R_{11} is upper triangular and non-singular.

Denote $A\Pi = \left[a_{c_1} \mid \cdots \mid a_{c_n}\right]$

& $Q = \left[q_1 \mid \cdots \mid q_n\right]$.

then $a_{c_k} = \sum_{i=1}^{\min\{r,k\}} r_{ik} q_i \in \text{span}\{q_1, \ldots, q_r\}$

Range of A = span $\{q_1, \ldots, q_r\}$.

Steps of the algorithm

1) Compute the $\|a_1\|_2 \cdots \|a_n\|_2$

let p be the smallest index for which

$$\|a_p\|_2 = \max_{1 \le i \le n} \{\|a_i\|_2\}$$

2) let $a_{c_1} = a_p$., i.e. apply permutation matrix P_1 to exchange columns 1 & p of A.

$$Q_1 \left(A P_1 \right) = \begin{bmatrix} * & | & | & | \\ 0 & | & | & | \\ \vdots & | & | & | \\ 0 & | & | & a_n \end{bmatrix}$$

$[a_1 \mid \cdots \mid a_n]$ ↘ a_p

3) Apply a suitable Householder matrix to annihilate subdiagonal entries of column 1.

$A \rightarrow AP_1 \rightarrow Q_1 A P_1 = A_1$

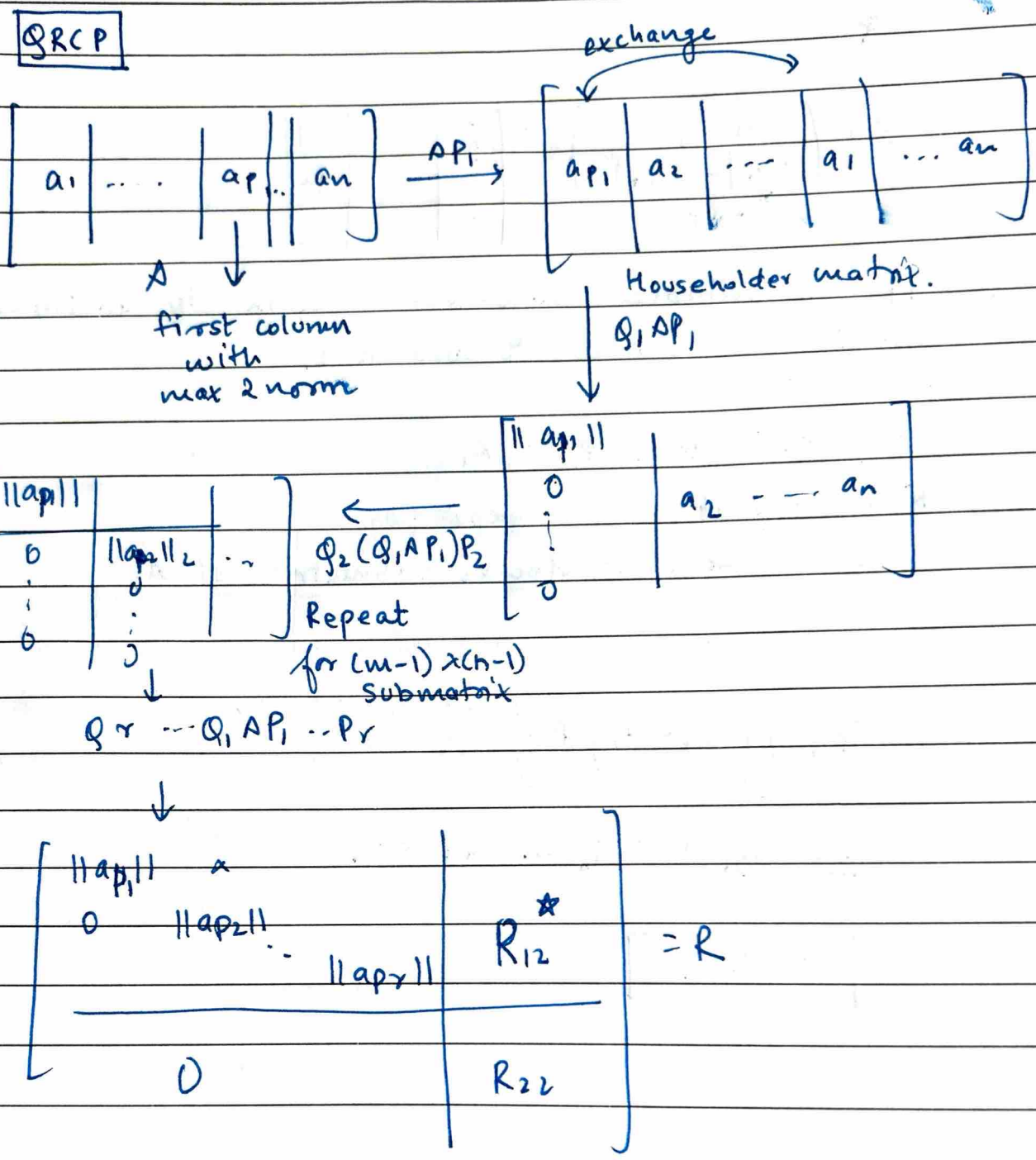
Now repeat these steps for the $m \times (n-1)$ lower submatrix of A_1

$$Q_2 \left(\left(Q_1 A P_1 \right) P_2 \right)$$

$$Q_{n-1} \cdots Q_1 A P_1 P_2 \cdots P_{n-1} = R.$$

$$\boxed{Q^T A \Pi = R}$$

★ QRCP



where $\|a_{p_1}\| \geq \|a_{p_2}\| \geq \cdots \geq \|a_{p_r}\|$

Remark Theoretically $R_{22} = 0$ (since $rk\,A = rk\,R = r$)
But in practice it rarely happens (unless some exceptional cancellation takes place) In any case $\|R_{22}\|$ is small

__/__/____

LSPs — full rank — $x_{LS} = A^+ b$
- using SVD
- using formulation
- using QR.

rank deficient — [if A has a complete orthogonal fact

$$Q^T A Z = \begin{bmatrix} T_{11} & 0 \\ 0 & 0 \end{bmatrix}$$

$\min \|e\|_2^2 = \|Ax - b\|_2^2$

(minimal) $x_{LS} = Q \begin{bmatrix} T_{11}^{-1} c \\ 0 \end{bmatrix}$

methods
- using SVD
- using QRCP (today).

$Q^T b = \begin{bmatrix} c \\ d \end{bmatrix} \} r$

$A : \mathbb{R}^n \to \mathbb{R}^m$

★ **Using QRCP for rank-deficient LSP** (suppose $A\Pi = QR$)

~~$A \in \mathbb{R}^{m \times n}$~~ ~~$A \in \mathbb{R}^{n \times m}$~~ $A \in \mathbb{R}^{m \times n}$.

$$\|Ax - b\|_2^2 = \| Q^T A \Pi \begin{bmatrix} y \\ z \end{bmatrix} - \begin{bmatrix} c \\ d \end{bmatrix} \|_2^2$$

$$= \|R_{11} y - (c - R_{12} z)\|_2^2 + \|d\|_2^2$$

where

$\Pi^T x = \begin{pmatrix} y \\ z \end{pmatrix} \} r \\ \} n-r$ $\quad Q^T b = \begin{bmatrix} c \\ d \end{bmatrix} \} r \\ \} n-r.$

$$x_{LS} = \Pi \begin{bmatrix} R_{11}^{-1} (c - R_{12} z) \\ z \end{bmatrix}$$

(basic sol^n x_B).

→ if x_{LS} with minimal norm is defined then z should be set to 0.

__/__/____

$$\therefore x_B = \Pi \begin{bmatrix} R_{11}^{-1} c \\ 0 \end{bmatrix} \begin{matrix} \} r \\ \} n-r \end{matrix}$$

★ **Numerical rank**

In theory, if rank $A = r$ & $A = U \Sigma V^*$

then $\Sigma = \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix}$

★ **Eigenvalue / Eigenvector methods**

For degree ≥ 5 there is no closed form solution use iterative methods instead.

The theoretical one method of finding roots of the characteristic polynomial doesn't have a closed form solution (or discriminant like solution) for polynomials of deg ≥ 5 (Abel's theorem).

Hence, the need for iterative methods.

E-value / E-vector methods

Direct method

Indirect method.

__/__/____

Direct methods

- Power iteration
- Block power iteration (simultaneous iteration)
- QR iteration
- Rayleigh quotient iteration.

Iterative methods

- Bisection method
- Divide & Conquer
- Jacobi's method
- Krylov subspace method
- Lanczos iterations.

- **Power iteration**

Idea $\{ b, Ab, A^2b, \dots \}$ → largest e-vector of A

↑ initial vector.

hm choose initial vector x_0

$i = 0$ to convergence

$y_{i+1} = Ax_i$

$x_{i+1} = y_{i+1} / \|y_{i+1}\|_2$ (approx e-vector)

$\lambda_{i+1} = x_{i+1}^T A x_{i+1}$ (approx-e-value).

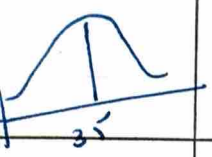
- Consider the simplest case $A = \text{diag}(\lambda_1 \dots \lambda_n)$

where $|\lambda_1| \geq |\lambda_2| \geq \dots |\lambda_n|$

16 10:30 - 1:00
17 9:30 - 1:00.

16
17
18
21

__/__/__



Note that in this case the e-vectors are e_i, i.e. columns of the identity matrix

let $x_i = \frac{A^i x_0}{\|A^i x_0\|}$

let $x_0 = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_n \end{pmatrix}$, $\xi_1 \neq 0$.

Then $A^i x_0 = A^i \begin{pmatrix} \xi_1 \\ \vdots \\ \xi_n \end{pmatrix} = \begin{pmatrix} \xi_1 \lambda_1^i \\ \xi_2 \lambda_2^i \\ \vdots \\ \xi_n \lambda_n^i \end{pmatrix}$

$\left(\frac{\lambda_n^i}{\lambda_n}\right)$

$= \xi_1 \lambda_1^i \begin{pmatrix} \frac{\xi_1}{\lambda_1}\left(\frac{\lambda_1^i}{\lambda_1}\right) \\ \vdots \\ \frac{\xi_n}{\lambda_n}\left(\frac{\lambda_n^i}{\lambda_n}\right) \end{pmatrix}$

Since each $|\lambda_i| \leq |\lambda_1|$,

$A^i x_0 \longmapsto \xi_1 \lambda_1^i e_1$

$\therefore x_i = \frac{A^i x_0}{\|A^i x_0\|} \longmapsto \pm e_1$ = e-vector corresponding to the largest e-value λ_1.

__/__/____

__/__/____

* Analysis of Orthogonal Iteration

1) Note that — A is diagonalizable

$\text{span}(Q_k) = \text{span}(Z_k)$

$= \text{span}(AQ_{k-1})$

$= \text{span}(A^2 Q_{k-2})$

$= \text{span}(A^k Z_0)$

$= \text{span}(SA^k S^{-1} Z_0)$.

(A is diagonalizable $\Rightarrow A = S \Lambda S^{-1}$, Λ ↑ matrix of e-value of A)

2) By assumption $|\lambda_1| > \dots > |\lambda_n| > |\lambda_{n+1}| \geq \dots \geq |\lambda_m|$

$\left|\frac{\lambda_i}{\lambda_n}\right| > 1$ for $i < n$.

& $\left|\frac{\lambda_i}{\lambda_n}\right| < 1$ for $i > n$.

So $\Lambda^k S^{-1} Z_0 = \lambda_n^k \begin{bmatrix} [V_k]_{n \times n} \\ \hline [W_k] \end{bmatrix}_{m \times n}$ (λ_n^k)

where $[W_k] \to 0$ as $k \to \infty$

& $[V_k] \to 0$.

Z_0 an orthogonal matrix

Set $Q_0 = Z_{0\ m \times n}$

for $i = 0$ to convergence

$Z_k = A Q_{k-1}$

$Q_k R_k = Z_k$

$\lambda_k (Q_k^* A Q_k) = \{ \lambda_1^{(k)} \dots \lambda_n^{(k)} \}$.

$$S = \underbrace{[s_1 | \cdots s_n}_{S_n} \underbrace{| s_{n+1} | \cdots | s_m]}_{S_{m-n}}$$

$\therefore$

$$S \Lambda^k S^{-1} z_0 = \lambda_n^k S \begin{bmatrix} [v_k] \\ \hline [w_k] \end{bmatrix}$$

$$S \Lambda^k S^{-1} z_0 = \lambda_n^k \left[S_n v_k + S_{m-n} w_k \right]$$

3) Finally, span Q_k = span$(S_n v_k + S_{m-n} w_k)$

$\rightarrow$ span $(S_n v_k)$ as $k \to \infty$

= span of first n e-vectors of A.

★ **QR iteration**

As before, we assume $|\lambda_1| > \cdots |\lambda_n|$

let

$$z_0 = \left[x_1^{(0)} \mid \cdots \mid x_m^{(0)} \right]$$ (initial matrix)

orthogonal columns.

Assume that all principal sub matrices of z_0 are non-singular.

Algorithm

$A \in \mathbb{C}^{m \times m}$

$T_0 = z_0^* A z_0$

$k = 1$ to convergence

$T_{k-1} = Q_k R_k$ (QR factorization)

$T_k = R_k Q_k$

end.

$T_0 = Z_0^* \Delta Z_0 = Q_1 R_1$

$T_1 = R_1 Q_1 = Q_2 R_2$

$T_2 = R_2 Q_2 = Q_3 R_3$.

On the one hand,

$T_{k-1} = Q_{k-1}^* \Delta Q_{k-1}$

$Q_0^* \Delta Q_0 = Q_1 R_1$

e-valve methods

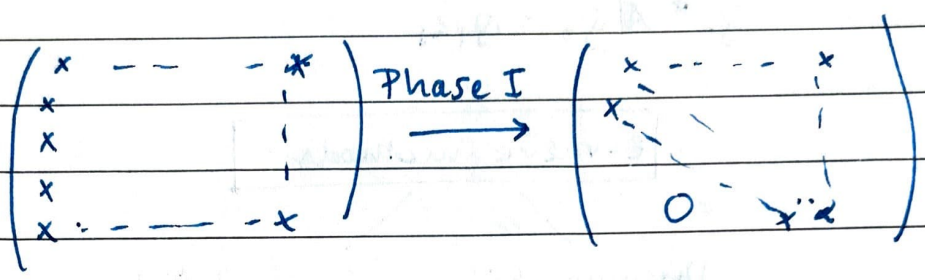
- Unsymmetric matrices
- Symmetric matrices
  - ↳ faster iterative methods.

__/__/____

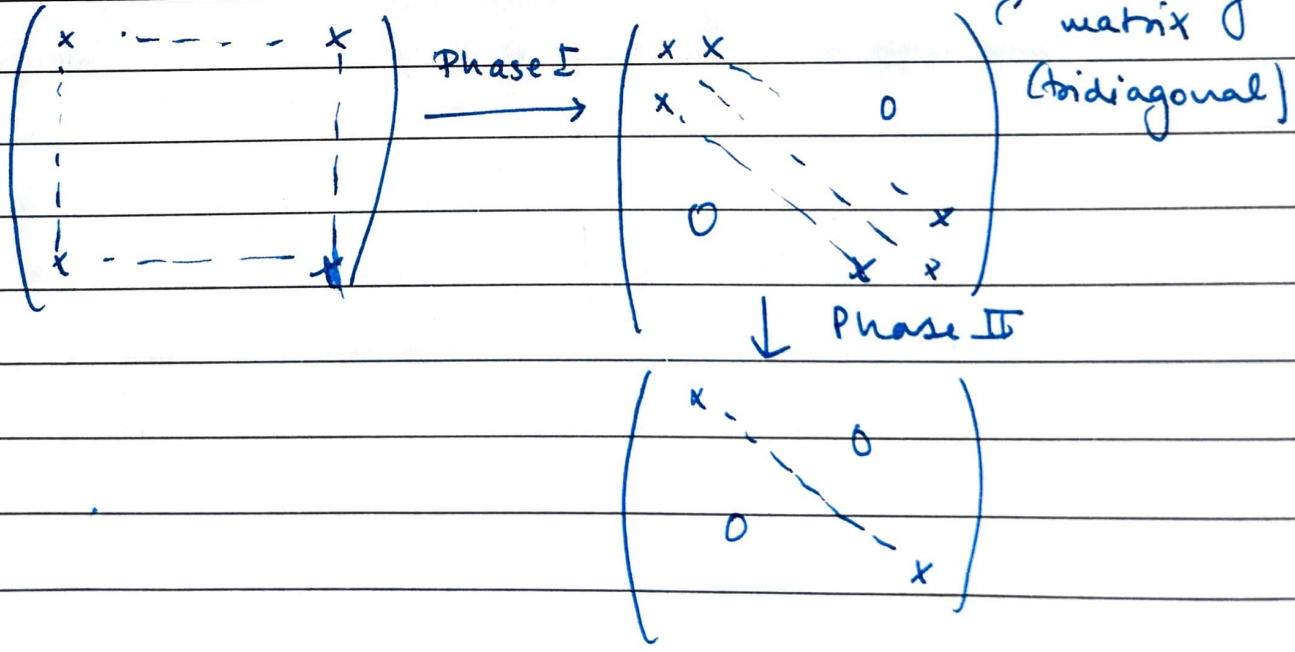
* A (original matrix) —reduction to a simpler form→ Hessenberg matrix —e-value method→ to obtain e-value & e-vector.

↓ has to be done through similarity transformation.

$A \neq A^*$



$A = A^*$



★ Conversion to Hessenberg form

Done using Householder matrices.

If we apply it as before

Suppose Q_1^* is a Householder matrix such that

$$Q_1^* A = \begin{pmatrix} x & \cdots & \cdots & x \\ 0 & & & x \\ \vdots & & & \vdots \\ 0 & \cdots & & x \end{pmatrix}$$

To make this a similarity transformation we need right multiplication with Q_1

$$(Q_1^* A) Q_1$$

But this step destroys the zeroes obtained earlier.

Better way

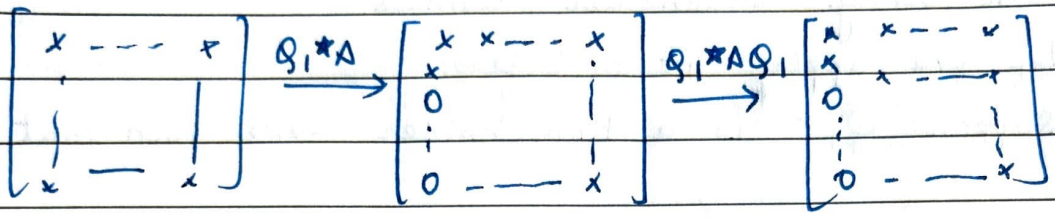
① At the first step, choose a Householder matrix that leaves the first row of A unchanged.

- when $Q_1^* A$ is computed, it forms linear combinations of rows $2, 3, \ldots, n$ (excludes row 1) & introduces zeroes in $(3,1), (4,1), \ldots (m,1)$
- when Q_1 is multiplied on the right of $Q_1^* A$, then it replaces columns of $2, 3, \ldots, n$ by their own linear combinations leaving the 1st column unchanged

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 5+14 \end{bmatrix}$

__/__/__

$Q_1^* A$



$Q_{m-2}^* \cdots Q_2^* Q_1^* A Q_1 Q_2 \cdots Q_{m-2} = Q^* A Q$ → Hessenberg matrix.

Q^* $\quad$ Q.

$I - \frac{2vv^*}{v^*v}$.

Algorithm $A \in \mathbb{R}^{m \times m}$

for $k = 1$ to $m-2$

$x = A_{k+1:m,k}$

$v_k = \text{sign}(x_1) \|x\|_2 e_1 + x$

$v_k = v_k / \|v_k\|$

$A_{k+1:m,k:m} = A_{k+1:m,k:m} - 2 v_k (v_k^* A_{k+1:m,k:m})$

~~$A_{1:m,k+1:m} = A_{1:m,k+1:m} - 2$~~

$A_{1:m,k+1:m} = A_{1:m,k+1:m} - 2 (A_{1:m,k+1:m} v_k) v_k^*$

end.

From orthogonal to QR iteration

The QR iteration is a rearrangement of the orthogonal iteration designed to form a sequence of matrices that converges to the Schur form of A.

(Digression)

Recall Schur form of A (decomposition)

Theorem: for every square $n \times n$ matrix A with entries in $\mathbb{C}$, $\exists$ unitary matrix Q & upper triangular matrix T such that $A = QTQ^*$

Observations: ① A & T are similar., diagonal entries of T are e-values of A.

② Columns of Q are the e-vectors of A.

Suppose $T_k = Q_k^* A Q_k$

i.e. $\{T_0, T_1, \dots T_k, \dots\} \to T = Q^* A Q$

core steps of

Recall the orthogonal iteration

$Q_0 = Z_0$

$Z_k = A Q_{k-1}$; $A Q_0 = Z_1 = Q_1 R_1$

$Q_k R_k = Z_k$; $A Q_1 = Z_2 = Q_2 R_2$

$Q_k R_k = Z_k$

$Q_k R_k = A Q_{k-1}$

$\boxed{R_k = Q_k^* A Q_{k-1}}$

Let's denote by T_k:

$T_{k-1} = Q_{k-1}^* A Q_{k-1}$ (by def^n)

$= Q_{k-1}^* Z_k$

$= (Q_{k-1}^* Q_k) R_k = \hat{Q}_k R_k$

$$T_k = Q_k^* A Q_k = \underbrace{(Q_k^* A Q_{k-1})}_{R_k}\underbrace{(Q_{k-1}^* Q_k)}_{\hat{Q}_k} = R_k \hat{Q}_k$$

$\therefore T_k$ can be obtained from T_{k-1} by computing QR factorization of T_{k-1} & reversing the order of QR.

QR iteration: $A \in \mathbb{C}^{n \times n}$

Z_0 unitary in $\mathbb{C}^{n \times n}$

$T_0 = Z_0^* A Z_0$ (principal submatrices of Z_0 are invertible).

$k=1$ to convergence:

$T_{k-1} = Q_k R_k$

$T_k = R_k Q_k$.

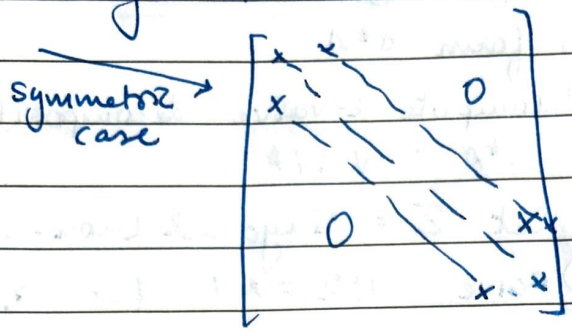
end.

Last time Any e-value computation is carried out in 2 stages.

A $\rightarrow$ $n \times n$ $\rightarrow$ Hessenberg form $\rightarrow$ apply e-value algo. $\rightarrow$ compute e-values & e-vectors.

Assymmetric (previous methods)

In the symmetric case the Hessenberg form is tridiagonal,



E-value / E-vector methods for symmetric matrices:

1) Bisection
2) Rayleigh quotient
3) Jacobi's
4) Divide & conquer.

* **Stability of Hessenberg matrices**

Let $\tilde{H}$ be the computed Hessenberg matrix & $\hat{Q}$ be the exact unitary matrix corresponding to the computed reflection vectors $\hat{v}_k$.

Then $\hat{Q}\tilde{H}\hat{Q}^* = A + \delta A$, where $\frac{\|\delta A\|}{\|A\|} = \Theta(\varepsilon_{mach})$.

$\therefore$ The computation of $\tilde{H}$ is backward stable.

* **Computing the SVD** (Given A)

Mathematically, ① form A^*A

The procedure is unstable. ② Compute e-value decomposition of $A^*A = V\Lambda V^*$

③ let Σ = diagonal (non-neg. sq. roots of Λ)

④ Solve $U\Sigma = AV$ for U.

* **A stable method is as follows**

Consider the $2m \times 2n$ Hermitian matrix

$$\begin{bmatrix} 0 & A^* \\ A & 0 \end{bmatrix}$$

If $A = U\Sigma V^*$, then $AV = U\Sigma$,

$A^*U = V\Sigma$.

$$\begin{bmatrix} 0 & A^* \\ A & 0 \end{bmatrix} \begin{bmatrix} V & V \\ U & -U \end{bmatrix} = \begin{bmatrix} A^*U & -A^*U \\ AV & AV \end{bmatrix}$$

$$= \begin{bmatrix} V & V \\ U & -U \end{bmatrix} \begin{bmatrix} \Sigma & 0 \\ 0 & -\Sigma \end{bmatrix} \rightarrow (*)$$

(*) is an e-value decomposition of H.

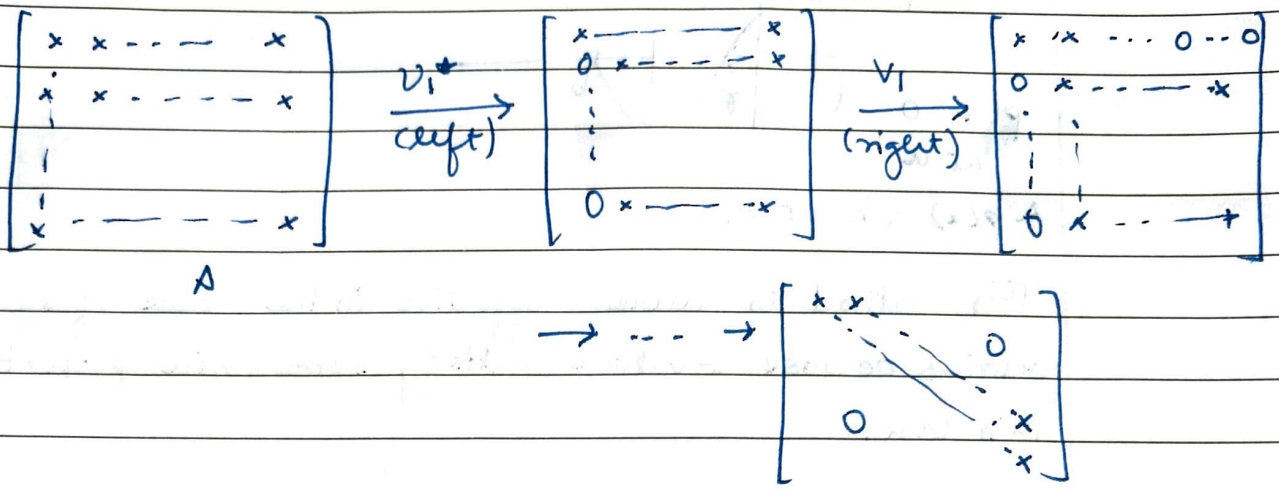
singular values of A are the abs. values of the e-values of H.

For non-sq. matrices a 2-phase method is used.

1st phase method - bidiagonalization.

2nd phase : QR method / divide & conquer.

★ Golub-Kahan bidiagonalization



$U_2^* (U_1^* A V_1 V_2) =$ bidiagonal matrix.

each U_i V_i is a Householder matrix.

Algorithm

1. form H
2. Compute its G-K bidiagonalization.
3. Apply e-value methods.

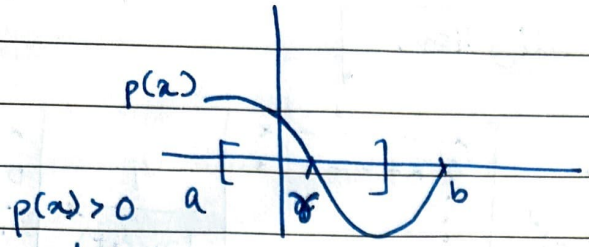
★ E-value and / E-vector algorithm for symmetric matrices

I] Bisection method - to locate roots of characteristic polynomial of a symmetric matrix A.

If $p(x)$ has a root at $r \in [a, b]$ then check $p(a)$ or $p(b)$ is zero. If yes, then root is formed.

If not, let $c = \frac{a+b}{2}$ & perform similar analysis for

$[a, c]$ & $[c, b]$. ... continue



$\Rightarrow p(x) < b$ at b.

This method is more effective in the case of symmetric matrices because e-values satisfy some nice properties in this case.

Let $A \in \mathbb{R}^{m \times m}$ be symmetric

Then A can be reduced to tridiagonal form -

$$\begin{pmatrix} a_1 & b_1 & \cdots & 0 \\ b_1 & a_2 & b_2 & \vdots \\ & b_2 & a_3 & \ddots & b_{m-1} \\ 0 & & \ddots & \ddots & \\ & & & b_{m-1} & a_m \end{pmatrix} \quad \begin{pmatrix} \text{suppose} \\ b_i \neq 0 \ \forall i \end{pmatrix}$$

① Let $A^{(1)}, \ldots A^{(m)}$ be the principal submatrices of the above tridiag-matrix.

Define $p_k(x) = \det(A^{(k)} - xI_{k \times k}) =$ char poly of $A^{(k)}$.

These characteristic polynomial satisfy,

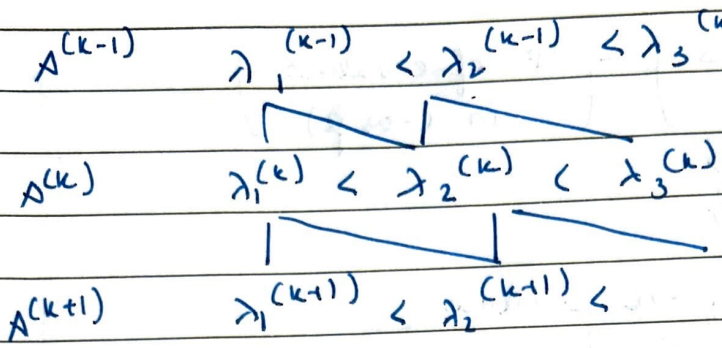
$$p_k(x) = (a_k - x)\, p_{k-1}(x) - b_{k-1}^2\, p_{k-2}(x)$$

② The e-values of A are distinct $(\lambda_1^{(k)} < \lambda_2^{(k)} < \ldots < \lambda_k^{(k)})$

(Sturm sequence property) (with the earlier assumptions)

The e-values of $A^{(k)}$ strictly interlace with e-values of $A^{(k+1)}$:

$$\lambda_j^{(k+1)} < \lambda_j^{(k)} < \lambda_{j+1}^{(k+1)}.$$



If $a(\lambda)$ = # of sign changes in the sequence

$p_0(\lambda), p_1(\lambda), \ldots, p_m(\lambda)$ (Sturm sequence)

then $a(\lambda)$ = # of e-values of A that are less than λ.

Using $a(\lambda)$ one can find the number of e-values of A in any interval.

$a(0)$ = # of sign changes in the seq.

$p_0(0), p_1(0), p_2(0), \ldots, p_m(0)$.

($p_0(0) = 1$)

= # of -ve e-values of A.

~~To find # of e-values in $(p, q]$ = $a(q) - a(p)$~~

To find # of e-values in $(p, q]$ = $a(q) - a(p)$.

Ex Find # of e-values of A

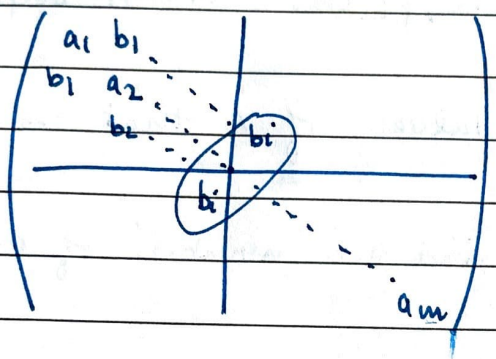
$$A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & 3 & 1 \\ 0 & 0 & 1 & -4 \end{pmatrix}$$

in the interval $[-1, 4]$.

$$\left(\begin{array}{c}\#\text{ of e-values} \\ \text{in } (-\infty, q)\end{array}\right) - \left(\begin{array}{c}\#\text{ of e-values} \\ \text{in } (-\infty, p)\end{array}\right)$$

$$\Downarrow$$

$$= \#\text{ of -ve e-values of } A + bI.$$

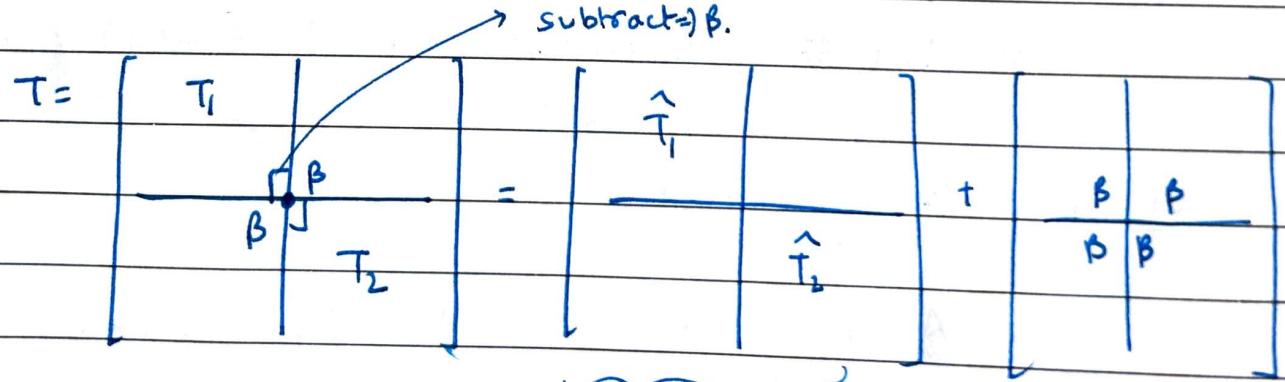


If same $b_i = 0$.
then "divide & conquer" (as a strategy).

★ **Divide and conquer method**

Let $T \in \mathbb{R}^{m \times m}$ be a symmetric tridiagonal matrix with non-zero off diagonal entries.
Let $n \approx \frac{m}{2}$, split T into submatrices

$[T_1]_{n \times n}$ & $[T_2]_{(m-n) \times (m-n)}$.



$\hat{T}_1$ & $\hat{T}_2$ are also tridiagonal symmetric.

Rank-1 correction.

__/__/____

If we have diagonalizations

$$\hat{T}_1 = Q_1 D_1 Q_1^T$$

&

$$\hat{T}_2 = Q_2 D_2 Q_2^T$$

then,

$$T = \begin{bmatrix} \hat{T}_1 & \\ & \hat{T}_2 \end{bmatrix} + \begin{bmatrix} \beta & \beta \\ \beta & \beta \end{bmatrix}$$

$$= \begin{bmatrix} Q_1 & \\ & Q_2 \end{bmatrix} \left(\begin{bmatrix} D_1 & \\ & D_2 \end{bmatrix} + \beta z z^T \right) \begin{bmatrix} Q_1^T & \\ & Q_2^T \end{bmatrix}$$

$\left(\begin{bmatrix} D_1 & \\ & D_2 \end{bmatrix} + \beta z z^T \right) = D$

where $z^T = (q_1^T, q_2^T)$

q_1^T ↓ last row of Q_1 ; q_2^T ↳ first row of Q_2.

★ What are the e-values of $D + ww^T$? (D diagonal, $w \neq 0$).

If λ is an e-value of $D + ww^T$, with e-vector q ($\neq 0$)

then $(D + ww^T)q = \lambda q$

i.e. $(D - \lambda I)q + w w^T q = 0.$

$$q + (D - \lambda I)^{-1} w w^T q = 0.$$

$$w^T q + w^T (D - \lambda I)^{-1} w w^T q = 0. \quad (w^T q \Rightarrow \text{scalar})$$

i.e.

$$w^T q \underbrace{\left(1 + w^T (D - \lambda I)^{-1} w\right)}_{= f(\lambda)} = 0$$

(scalar function related to this sptm

This is true if

$f(\lambda) = 0$.

$1 + \sum_{i=1}^{n} \frac{w_i^2}{d_i - \lambda} = 0.$

Roots of $f(\lambda)$ are eigen values of $D + w w^T$

III) ★ **Rayleigh quotient iteration**

A is symmetric $R_A : V \setminus \{0\} \to \mathbb{R}$

$$x \mapsto \frac{x^T A x}{x^T x}$$

★★★ If λ is an e-value of A with e-vector v, then

$R_A(v) = \lambda$.

The problem of estimating λ can be framed as a least squares problem as follows-

Given $x \neq 0$, find a scalar λ that minimizes

$\| Ax - \lambda x \|_2$

The normal eqns for the system,

$\therefore x^T x \lambda = x^* A x.$

$Ax = b$

$\Rightarrow A^T A x = A^T b$

It can be shown that the e-vectors of A are the stationary pts of $R_A(x)$.

As a consequence it can be shown that

$R_A(x) - R_A(q) = O(\|x - q\|^2)$ as $x \to q$

$\therefore R_A(x)$ is a "quadratically accurate" estimate of $R_A(q)$

★ Algorithm

1) ⇒ Choose $v^{(0)}$ with $\|v^{(0)}\|_2 = 1$
2) ⇒ $\lambda^{(0)} = v^{(0)T} A v^{(0)}$
3) for $k = 1$ to convergence
4) Solve $(A - \lambda^{(k-1)} I) w = v^{(k-1)}$ for w
5) $v^{(k)} = \frac{w}{\|w\|_2}$
6) $\lambda^{(k)} = R_A(v^{(k)})$
7) end.

Idea

① Start with a vector $v^{(0)}$
② Calculate $R_A(v^{(0)})$ - first approximation
③ Apply inverse iteration to $\lambda^{(0)}$ to get $v^{(1)}$, then apply R_A to $v^{(1)}$ & so on.

★ Jacobi's method

Idea: Try to eliminate / reduce the magnitude of off-diagonal entries so that eventually they become small enough to be declared as zero.

This is done with a sequence of orthogonal similarity transformation $A \mapsto Q^T A Q$.

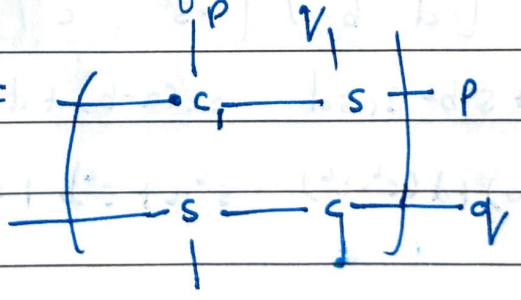
Define $\text{off}(A) = \sqrt{\sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij}^2}$ - Frobenius norm. of off diagonal entries of A.

The transformations

$A \mapsto J_1^T A J_1 \longrightarrow Q_2^T A_1 Q_2 \mapsto$ should result in a reduction of size of $\text{off}(A)$.

The matrix Q used for this is the Givens' rotation.

$J(p, q, \theta) =$



The basic step of the procedure is as follows -

$$\begin{pmatrix} b_{pp} & b_{pq} \\ b_{qp} & b_{qq} \end{pmatrix} = \underbrace{\begin{pmatrix} c & s \\ -s & c \end{pmatrix}}_{J^T} \underbrace{\begin{pmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} c & -s \\ s & c \end{pmatrix}}_{J}$$

Since Frobenius norm is preserved by orthogonal transformation

$$a_{pp}^2 + a_{qq}^2 + 2a_{pq}^2 = b_{pp}^2 + b_{qq}^2 + 2b_{pq} = b_{pp}^2 + b_{qq}^2$$

Consider $\text{off}(B)^2 = \|B\|_F^2 - \sum_{i=1}^{n} b_{ii}^2$

$$= \|A\|_F^2 - \sum_{\substack{i=1 \\ i \neq p,q}}^{n} b_{ii}^2 - b_{pp}^2 - b_{pq}^2 + (a_{pp}^2 + a_{qq}^2) - a_{pp}^2 - a_{qq}^2$$

$$= \|A\|_F^2 - \sum_{i=1}^{n} a_{ii}^2 + (a_{pp}^2 + a_{qq}^2 - b_{pp}^2 - b_{qq}^2)$$

$$= \text{off}(A)^2 - 2a_{pq}^2$$

$$\text{off}(B) < \text{off}(A).$$

$$J^T A J = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

$$\begin{pmatrix} c & -s \\ s & c \end{pmatrix} \begin{pmatrix} a & d \\ d & b \end{pmatrix} \begin{pmatrix} c & s \\ -s & c \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

$$\begin{pmatrix} c^2a + s^2b - 2csd & sc(a-b) + d(c^2-s^2) \\ sc(a-b) + d(c^2-s^2) & s^2a + c^2b + 2scd \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

$$sc(a-b) + d(c^2-s^2) = 0$$

$$\therefore \frac{d}{b-a} = \frac{sc}{c^2-s^2}$$

$$\tan 2\theta.$$

__/__/____

Algorithm

Given $n \times n$ symmetric matrix A & integers p, q $1 \le p < q \le n$.
Compute (c, s) s.t. if $B = J(p, q, \theta)^T A J(p, q, \theta)$
then $b_{pq} = b_{qp} = 0$.

if $a_{pq} \neq 0$ then

$\tau = (a_{qq} - a_{pp}) / 2a_{pq}$

if $\tau \geq 0$

$t = \frac{1}{(\tau + \sqrt{1+\tau^2})}$

else

$t = \frac{1}{(\sqrt{\tau - \sqrt{1+\tau^2}})}$

end

$c = \frac{1}{\sqrt{1+\tau^2}}$

$s = tc.$

★ **Classical Jacobi algorithm** → parameter that user sets (convergence criterial).

conv : off $(V^T A V) \leq$ tol. $(\|A\|_F)$

$V = I_n$, $\delta =$ tol. $(\|A\|_F)$.

while off $(A) > \delta$

choose (p, q) so that $|a_{pq}| = \max_{i \neq j} |a_{ij}|$

→ obtain (c, s) for this choice of p, q

$A = J(p, q, \theta)^T A J(p, q, \theta)$

$V = VJ(p, q, \theta)$.

* Iterative methods for e-valve/ e-vector computation

Krylov subspace methods

$Ax = b$.

$\{b, Ab, A^2b, \ldots\}$ Krylov sequence.

$K_1 = \langle b \rangle$

$K_2 = \langle b, Ab \rangle$

$K_p = \langle b, Ab, \ldots, A^{p-1}b \rangle$.

Given a matrix A and a vector b, let

$$K = \left[b \mid Ab \mid A^2b \mid \cdots \mid A^{n-1}b \right]$$

then $$AK = \left[Ab \mid A^2b \mid \cdots \mid A^n b \right]$$

If K is non-singular, let $c = -K^{-1}A^n b$.

then $$AK = K \left[e_2 \mid e_3 \mid \cdots \mid e_n \mid -c \right] = KC$$

where

$$C = \begin{bmatrix} 0 & 0 & \cdots & -c_1 \\ 1 & 0 & \cdots & -c_2 \\ 0 & 1 & & \\ & & \ddots & \\ 0 & \cdots & 1 & -c_n \end{bmatrix}$$

Note that C is a Hessenberg matrix

C is the ~~companion~~ companion matrix of

$$p(x) = x^n + \sum_{i=1}^{n} c_i x^{i-1}$$

$$C = K^{-1} A K$$

Here, K ~~maybe ill-co~~ may be ill-conditioned : A possible solution to replace K with an orthogonal matrix Q s.t. for all leading k columns of K & Q span the same subspace

$$K = QR$$

Then,

$$C = K^{-1} A K = R^{-1} Q^T A Q R = R^{-1} H R.$$

$$H = RCR^{-1}$$

upper triangular. $\Rightarrow$ H is upper Hessenberg.

To compute the columns of Q.

Let $Q = [q_1 | \cdots | q_n]$

$$Q^T A Q = H$$

$$\Rightarrow AQ = QH$$

Equating the j^{th} column on both sides -

$$Aq_j = \sum_{i=1}^{j+1} h_{ij} q_i$$

Since q_i are orthogonal $\forall\ 1 \le m \le j$,

$$\therefore q_m^T A q_j = \sum_{i=1}^{j+1} h_{ij} q_m^T q_i$$

$$= h_{mj}$$

$$\therefore Aq_j = \sum_{i=1}^{j} h_{ij} q_i + h_{j+1,j} q_{j+1}$$

$$h_{j+1,j} q_{j+1} = Aq_j - \sum_{i=1}^{j} h_{ij} q_i$$

★ Arnoldi's algorithm

$q_1 = \frac{b}{\|b\|_2}$ $\quad$ k = # of columns of Q (AH) to be computed.

for $j = 1$ to k
  $z = Aq_j$
  for $i = 1$ to j } like MGS.
    $h_{ij} = q_i^T z$
    $z = z - h_{ij} q_i$
  end
  $h_{j+1,j} = \|z\|_2$
  if $h_{j+1,j} = 0$; quit.
  $q_{j+1} = \frac{z}{h_{j+1,j}}$
end.

$$Q = \left[\begin{array}{c|c} Q_k & Q_u \end{array}\right]$$
(Q_k: known, Q_u: unknown.)

$$H = Q^T A Q$$
$$= [Q_k | Q_u]^T A [Q_k | Q_u]$$
$$= \begin{bmatrix} Q_k^T A Q_k & Q_k^T A Q_u \\ Q_u^T A Q_k & Q_u^T A Q_u \end{bmatrix}$$
$$= \left[\begin{array}{c|c} H_K & H_{uk} \\ \hline H_{ku} & H_{uu} \end{array}\right]$$
(H_K: known.)

__/__/____

Last time: Arnoldi's algorithm for generating an orthogonal basis for the Krylov subspace

$$K_k(A, b) = \{ b, Ab, A^2b, \ldots, A^{k-1}b \}$$

Why Krylov subspaces are useful?

- General dense matrix A: factorisation methods to solve $Ax=b$ & general e-value methods for A are effective
- If A is huge & sparse: need other methods - iterative methods like Krylov subspace methods.

Solving $Ax=b$ $\leftrightarrow$ optimisation problem.

Quadratic $f(x) = \frac{1}{2} x^T A x - b^T x + c.$

using matrix derivative $f'(x) = \frac{1}{2} A^T x + \frac{1}{2} Ax - b$

If A is symmetric p.d.

$$f'(x) = Ax - b. \qquad b - Ax_0 = r_0$$

$$K_k(A, r_0) = \{ r_0, Ar_0, \ldots, A^{k-1} r_0 \}$$

gradients are linear combinations of

When A is symmetric

$$R_A(x) = \frac{x^T A x}{x^T x}.$$

tri-diagonal matrix

(T) $\leftrightarrow$ optimizing $R_T(x)$

$\nabla R_T \in \langle A, Ar, A^2 r, \ldots A^{k-1} r \rangle.$

Krylov subspace methods

① Arnoldi's

$Ax=b$ — Conjugate Gradient, GMRES.

$Ax=\lambda x$ — Lanczos

* **Lanczos algorithm for extremal e-values of a symmetric p.d. A (large, sparse)**

Want: $Q^T A Q = T$ $\quad$ $T \Rightarrow$ tridiagonal

$AQ = QT$

$$T = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \ddots & \ddots & & \\ & \ddots & \ddots & \beta_{n-1} & \\ & & \beta_{n-1} & \alpha_n & \end{bmatrix}$$

$$Aq_j = \beta_{j-1} q_{j-1} + \alpha_j q_j + \beta_j q_{j+1}$$

$q_j^T A q_j = q_j^T q_{j-1}$ ← Q is orthogonal.

$q_j^T A q_j = \alpha_j$

$q_1 = b$, $\quad q_1 = \frac{a_1}{\|a_1\|}$, $\quad \beta = 0$

$q_1 = \frac{a_1}{\|a_1\|}$, $\beta = 0$

for $j = 1$ to k

$z = Aq_j$

$\alpha_j = q_j^T z$

$z = z - \alpha_j q_j - \beta_{j-1} q_{j-1}$

$\beta_j = \|z\|_2$

if $\beta_j = 0$, quit

$q_{j+1} = \frac{z}{\beta_j}$

$T = Q^T A Q$

$$= \begin{bmatrix} T_k & T_{UK} \\ k \times k & \\ \hline T_{kU} & T_U \end{bmatrix} \begin{matrix} \} k \\ \} n-k \end{matrix}$$

k $\quad$ $n-k$

$T_{kU}^* = T_{UK}$.