

# Numerical Methods for Computational Science and Engineering

Autumn Semester 2018

Prof. Rima Alaifari, SAM, ETH Zurich

QR decomposition  
(Find one that is numerically stable?)

Gram-Schmidt revisited:

$Q = [q^1 \dots q^n] \in \mathbb{R}^{m,n}$  with orthonormal columns such that:

$$q^1 = t_{11}a^1$$

$$q^2 = t_{12}a^1 + t_{22}a^2$$

$$q^3 = t_{13}a^1 + t_{23}a^2 + t_{33}a^3$$

:

$$q^k = t_{1n}a^1 + t_{2n}a^2 + \dots + t_{nn}a^n.$$

For a matrix  $A = [a^1, a^2, \dots, a^n] \in \mathbb{R}^{m,n}$ :

Step 1:

$$\left[ \begin{array}{c|c|c|c} & a^1 & a^2 & \dots & a^n \\ \hline a^1 & | & | & \dots & | \\ a^2 & | & | & \dots & | \\ \vdots & | & | & \dots & | \\ a^n & | & | & \dots & | \end{array} \right] \left[ \begin{array}{ccccc} t_{11} & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & 0 & 1 & & \ddots \\ \vdots & \vdots & & & 1 \\ 0 & 0 & & & 1 \end{array} \right] = \left[ \begin{array}{c|c|c|c} & t_{11}a^1 & a^2 & \dots & a^n \\ \hline t_{11}a^1 & | & | & \dots & | \\ a^2 & | & | & \dots & | \\ \vdots & | & | & \dots & | \\ a^n & | & | & \dots & | \end{array} \right]$$

Step 2:

$$\left[ \begin{array}{c|c|c|c} & t_{11}a^1 & a^2 & \dots & a^n \\ \hline t_{11}a^1 & | & | & \dots & | \\ a^2 & | & | & \dots & | \\ \vdots & | & | & \dots & | \\ a^n & | & | & \dots & | \end{array} \right] \left[ \begin{array}{ccccc} 1 & \tilde{t}_{12} & 0 & \dots & 0 \\ 0 & t_{22} & 1 & 0 & \dots & 0 \\ \vdots & 0 & 0 & 1 & & \ddots \\ \vdots & \vdots & & & & 1 \\ 0 & 0 & 0 & & & 1 \end{array} \right] \xrightarrow[t_{12}=t_{11}\tilde{t}_{12}]{} \left[ \begin{array}{c|c|c|c} & t_{11}a^1 & t_{12}a^1 + t_{22}a^2 & \dots & a^n \\ \hline t_{11}a^1 & | & | & \dots & | \\ t_{12}a^1 + t_{22}a^2 & | & | & \dots & | \\ a^3 & | & | & \dots & | \\ \vdots & | & | & \dots & | \\ a^n & | & | & \dots & | \end{array} \right]$$

At each step: multiplication from the right by  
upper triangular matrix

$$Q = A \underbrace{T_1 \dots T_n}_{=: T}$$

$R = T^{-1}$  is also upper triangular  
 $\Rightarrow A = QR$

QR decomposition via orthogonal transformations

$$Q \underbrace{\dots Q}_T \cdot A = R \quad \leftarrow \text{upper triangular}$$

$$A = QR$$

Orthogonal transformations on vectors:  $\langle u, v \rangle = \langle Qu, Qv \rangle$

- preserve their length
- preserve angles between vectors

are only combinations of rotations & reflections

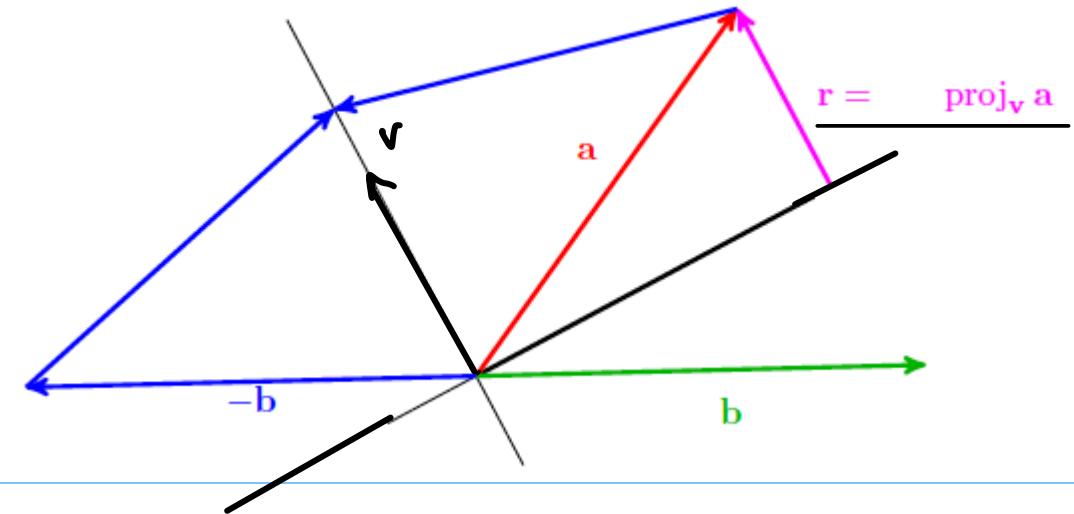
rotation only:  $\det Q_{\text{rot}} = +1$

reflection only:  $\det Q_{\text{ref}} = -1$

Householder reflections

Idea: use reflections only

Consider reflection of vector  $a$  wrt. vector  $v$



Vector  $a$  reflected at  $v$  gives  $b$

$$b = a - 2r = a + 2 \underbrace{\text{proj}_v a}_{\text{proj}_v a}$$

$$= a - 2 \frac{v^T a}{\|v\|_2^2} v$$

$$= a - 2 \frac{vv^T}{\|v\|_2^2} a = \left( I_m - 2 \frac{vv^T}{v^T v} \right) a$$

↑  
m × n identity matrix

$$\underbrace{(v^T a)}_{\text{scalar}} v = v (v^T a)$$

scalar

Now we can define the Householder matrix  $H_v$  as

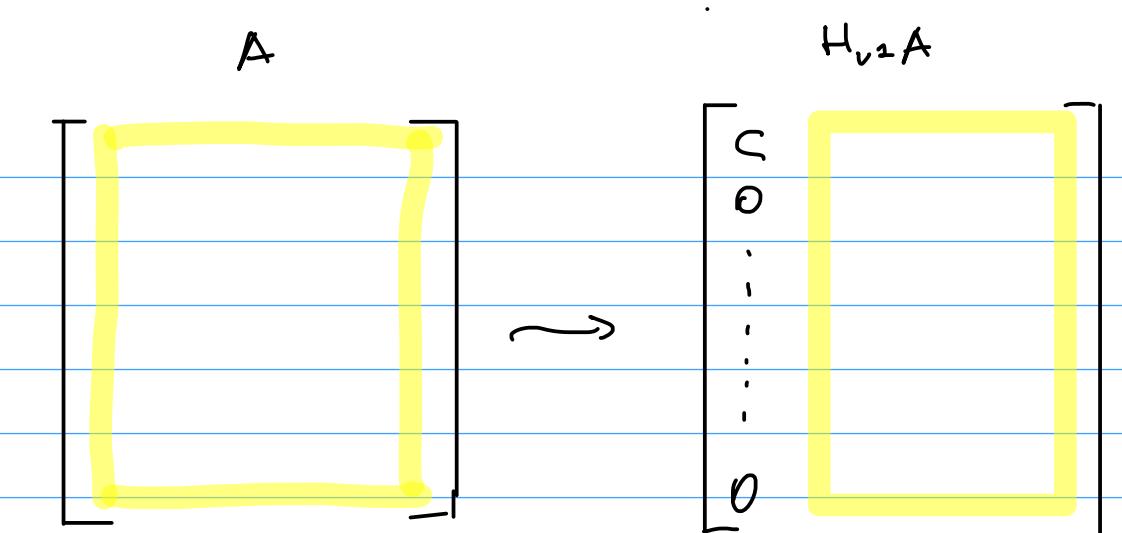
$$H_v := I_m - \frac{2vv^T}{v^T v}$$

so that  $H_v a = b$ .

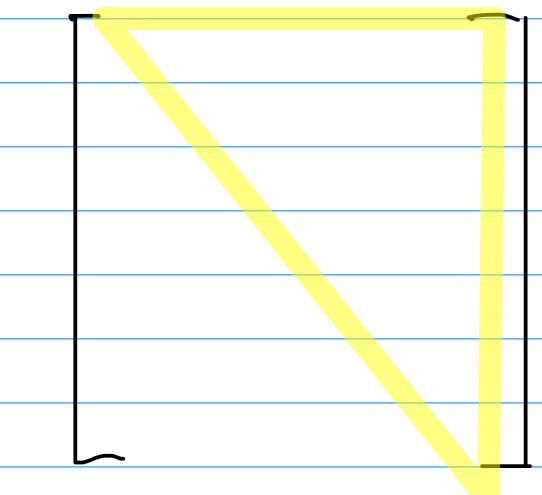
Goal: Find vector  $v^1, \dots, v^n$  such that

$$\underbrace{H_{v^n} \dots H_{v^1} A}_{= Q^T} = R$$

↑ upper triangular



$$H_{v_n} H_{v_{n-1}} \dots H_{v^1} A$$



1<sup>st</sup> step :

$$H_{v^1} a^1 = \begin{pmatrix} c \\ 0 \\ \vdots \\ 0 \end{pmatrix} = c e^1$$

$\nearrow e^i$   $i$ -th unit vector

Find  $v^1$  and  $c$  s.t.

$$H_{v^1} a = a - 2 \frac{(v^1)^\top a}{\|v^1\|_2^2} v^1 = ce^{-1}$$

$$\Rightarrow v^1 = (a - ce^{-1}) \frac{\|v^1\|_2^2}{2(v^1)^\top a}$$

$\Rightarrow v^1$  must parallel to  $a - ce^{-1}$

Note: Rescaling of  $v^1$  doesn't effect the above eqn:

$$\textcolor{red}{C} v^1 = (a - ce^{-1}) \frac{\|v^1\|_2^2}{2(v^1)^\top a} \quad \checkmark$$

We set  $v^1 = a - ce^{-1}$

This implies:

$$\frac{\|v^1\|_2^2}{2(v^1)^\top a} = 1$$

Use this to determine  $c$ :

$$\|a - ce^{-1}\|_2^2 = \|v^1\|_2^2 = 2(v^1)^\top a = 2(a - ce^{-1})^\top a$$

$$\|a\|_2^2 - 2a^\top (ce^{-1}) + c^2 = 2\|a\|_2^2 - 2a^\top (ce^{-1})$$

$$\Rightarrow c^2 = \|a\|_2^2$$

$$\Rightarrow c = \pm \|a\|$$

Note: Cancellation can occur in  $v^1 = a - ce^{-1}$  + amplification because we divide by  $\|v^1\|_2^2$

Good news: can happen only for one of the possible choices

$$c = \|a\|_2 \quad \text{or} \quad c = -\|a\|_2$$

Choose depending on first entry of  $a^1$ :

$$v^1 = \begin{cases} \frac{1}{2}(a^1 - \|a^1\|_2 e^1) & \text{if } (a^1)_1 < 0 \\ \frac{1}{2}(a^1 + \|a^1\|_2 e^1) & \text{if } (a^1)_1 > 0 \end{cases}$$

How does a general HH step look like?

Consider j-th step:

Goal: create  $m-j$  zeros in a vector  $\tilde{a}^j$

$$\tilde{a}^j = \begin{bmatrix} \tilde{a}_1^j \\ \tilde{a}_2^j \end{bmatrix} \in \mathbb{R}^{m-j+1}$$

Can we find  $v^j$  s.t.

$$H_{v^j} \tilde{a}^j = \begin{bmatrix} r_1^j \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^m$$

?  $\{ m-j \text{ zeros} \}$

Choose  $v^j = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \tilde{a}_2^j \end{bmatrix} \in \mathbb{R}^{m-j+1}$

$\{ j-1 \text{ zeros} \}$

+  $c_j e^j$

with  $c_j^{\pm} = \pm \|\tilde{a}_2^j\|$

choose sign dep. on sign  $(\tilde{a}_2^j)_1$

Compute

$$H_{v^j} \tilde{a}^j = \tilde{a}^j - 2 \underbrace{\frac{(\tilde{a}^j)^T v^j}{\|v^j\|^2} v^j}_{= 1} = \tilde{a}^j - v^j$$

$$(\tilde{a}^j)^T v^j = (\tilde{a}_2^j)^T (\tilde{a}_2^j + c_j \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix})$$

$$= \|\tilde{a}_2^j\|_2^2 + (\tilde{a}_2^j)_1 \cdot c_j$$

$$(v^j)^T v^j = (\tilde{a}_2^j + \begin{pmatrix} c_j \\ \vdots \\ 0 \end{pmatrix})^T (\tilde{a}_2^j + \begin{pmatrix} c_j \\ \vdots \\ 0 \end{pmatrix})$$

$$= \|\tilde{a}_2^j\|_2^2 + 2 (\tilde{a}_2^j)_1 \cdot c_j + c_j^2$$

$$\Rightarrow (\tilde{a}_2^j)^T v^j = 2 \left\{ \|\tilde{a}_2^j\|_2^2 + (\tilde{a}_2^j)_1 \cdot c_j \right\}$$

$$\frac{(\tilde{a}_2^j)^T v^j}{(v^j)^T v^j} = \frac{1}{2}$$

$$H_{v^j} \tilde{a}^j = \tilde{a}^j - v^j = \begin{bmatrix} \tilde{a}_1^j \\ \vdots \\ \tilde{a}_2^j \end{bmatrix} - \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} - c_j \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$\in \mathbb{R}^{m-j+1}$

entry

coincide  
on indices  
 $j+1$  to  $m$

$$= \begin{bmatrix} r^j \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \begin{array}{l} \text{---} \\ \text{---} \\ \text{---} \end{array} \quad \begin{array}{l} \text{---} \\ \text{---} \\ \text{---} \end{array}$$

$\in \mathbb{R}^j$

$m-j$  zeros

What happens to previous columns  $r^k$   $k < j$   
 ||

as we apply  $H_{v^j}$

$$H_{v^j} r^k = r^k - 2 \frac{(v^j)^T r^k}{\|v^j\|_2^2} v^j = r^k$$

$$(v^j)^T r^k = \left\langle \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \tilde{a}_2^j \end{pmatrix}, \begin{pmatrix} (r^k)_1 \\ \vdots \\ (r^k)_k \\ 0 \\ \vdots \\ 0 \end{pmatrix} \right\rangle = 0$$

Altogether:  $H_{v^n} \dots H_{v^1} A = R$

$$Q = H_{v^1}^T \dots H_{v^n}^T$$

Instead of forming & storing  $Q$ : in practice  
 only the vectors  $v^1, \dots, v^n$  are stored as one

triangular matrix ( $v^j$ : first  $j-1$  entries are zero)

Remark:

For a "tall" matrix  $A \in \mathbb{R}^{m,n}$  ( $m > n$ )

one can either compute the "full" QR decomposition

$$A = \underbrace{\tilde{Q}}_{\text{columns}} \begin{bmatrix} Q \\ R \end{bmatrix}$$

The matrix  $\tilde{Q}$  has  $m$  columns and  $n$  rows. The matrix  $R$  is  $n \times n$  and triangular.

or a "reduced / thin" QR decomposition:

$$A = \tilde{Q} \begin{bmatrix} \tilde{R} \\ 0 \end{bmatrix}$$

The matrix  $\tilde{Q}$  has  $n$  columns and  $n$  rows. The matrix  $\tilde{R}$  is  $n \times n$  and triangular.

In Gram-Schmidt: starting from  $a^1, \dots, a^n$  columns  
we compute an orthogonal set of vectors  $q^1, \dots, q^n$   
 $\rightarrow$  reduced QR

How to obtain a full QR?

augment  $\tilde{R}$  by zero rows

Find columns  $q^{n+1}, \dots, q^m$  such that

$\{q^1, \dots, q^m\}$  ONB for  $\mathbb{R}^m$

With Householder reflections

either compute full QR decomposition

$$Q = H_{v_1}^T \cdots H_{v_n}^T$$

(recall:  $H_{v_i}$  are  $m \times m$  matrices)

or: take "full" "identity" matrix  $I_{m,n} = \begin{pmatrix} 1 & & & \\ & \ddots & & \\ & & \ddots & 1 \\ & & & 0 \end{pmatrix}$

and compute reflections successively

Example: Instead of constructing

$$H_{v^n}^T = I_m - 2 \frac{v^n(v^n)^T}{\|v^n\|_2^2}$$

we simply compute the reflection for each column

$e^\pm$  of  $I_{m,n}$ :

$$\tilde{q}^\pm := e^\pm - 2 \frac{(v^n)^T e^\pm}{\|v^n\|_2^2} v^n$$

Next, apply w.r.t.  $v^{n-1}$  on  $\tilde{q}^1, \dots, \tilde{q}^n$

This way: thin QR  $\tilde{Q} = [\tilde{q}^1, \dots, \tilde{q}^n]$

In EIGEN:

MatrixXd Q = qr.householderQ();

(full Q)

MatrixXd Qthin = (qr.householderQ()) \* MatrixXd::Identity(m,n);

↑  
apply all HH reflections n times

In general: HH reflections are directly applied to A

without constructing Q

Complexity of  $HH^T$  for  $m > n$ ?

1 reflection applied to 1 vector :  $\Theta(m)$

1 reflection applied to  $A$  :  $\Theta(mn)$

$n$  reflections applied to  $A$  :  $\Theta(mn^2)$

Solving least-squares problem with  
QR decomposition

Suppose we have computed decomposition  $A = QR$ .  
(recall  $Q^H Q = I$ )

$$\text{Then } \|Ax - b\|_2 = \|QRx - QQ^H b\|_2$$

$$= \|Q(Rx - Q^H b)\|_2 \underset{\substack{\uparrow \\ Q \text{ preserves length}}}{=} \|Rx - \underbrace{Q^H b}_2\|_2$$

Thus solving  $Ax = b$  in least-squares sense

$\Leftrightarrow$  to solving  $Rx = \tilde{b}$  in least-sq. sense

$$\left\| \begin{bmatrix} R \\ \vdots \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} - \begin{bmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_m \end{bmatrix} \right\|_2 \rightarrow \min$$

The zero rows can never be fulfilled:

$$x = R_0^{-1} \begin{bmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_n \end{bmatrix}$$

$$R_0 = (R)_{1:n, 1:n}$$

We can also compute the residual:

$$r = Q \begin{bmatrix} 0 \\ \vdots \\ \tilde{b}_{n+1} \\ \vdots \\ \tilde{b}_m \end{bmatrix} \Rightarrow \|r\|_2 = \left( \sum_{i=n+1}^m \tilde{b}_i^2 \right)^{1/2}$$

Alternatively to QR reflections: Givens rotations  
are based on rotations only.

### 3.4. The Singular Value Decomposition

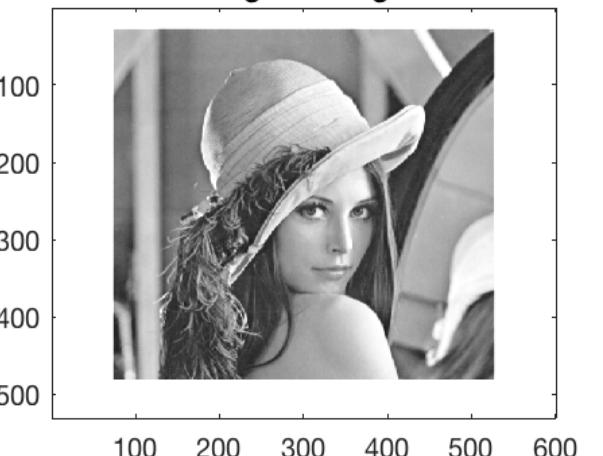
A different orthogonal decomposition

- Example:
- How to compress an image?
  - How to extract common features

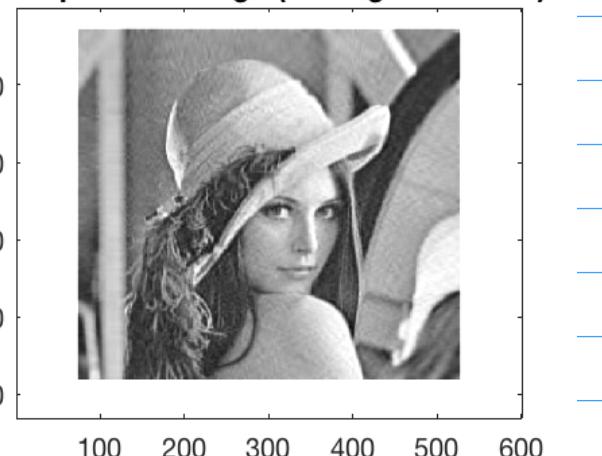
from a data set?

- How to solve  $Ax=b$  approximately?
  - 1.) Construction of Moore-Penrose inverse  $A^+$
  - 2.) What can be done if  $\text{cond}(A)$  is large?

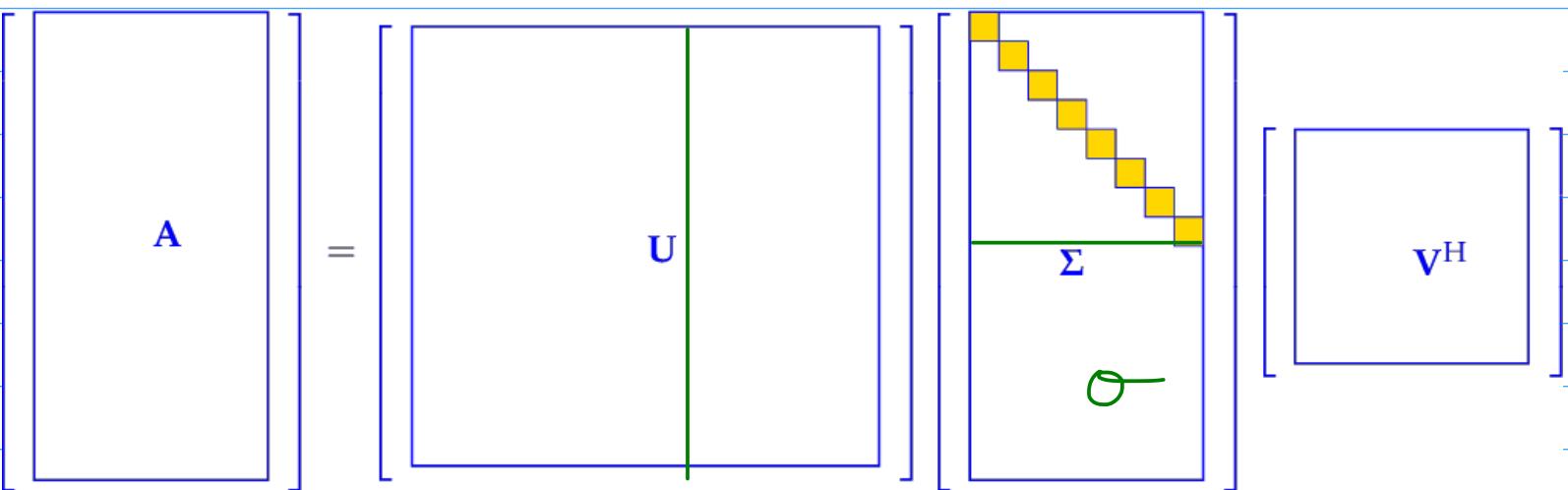
Original image



Compressed image (50 singular values)



Scheme of SVD for "tall"  $A$ : ( $m > n$ )

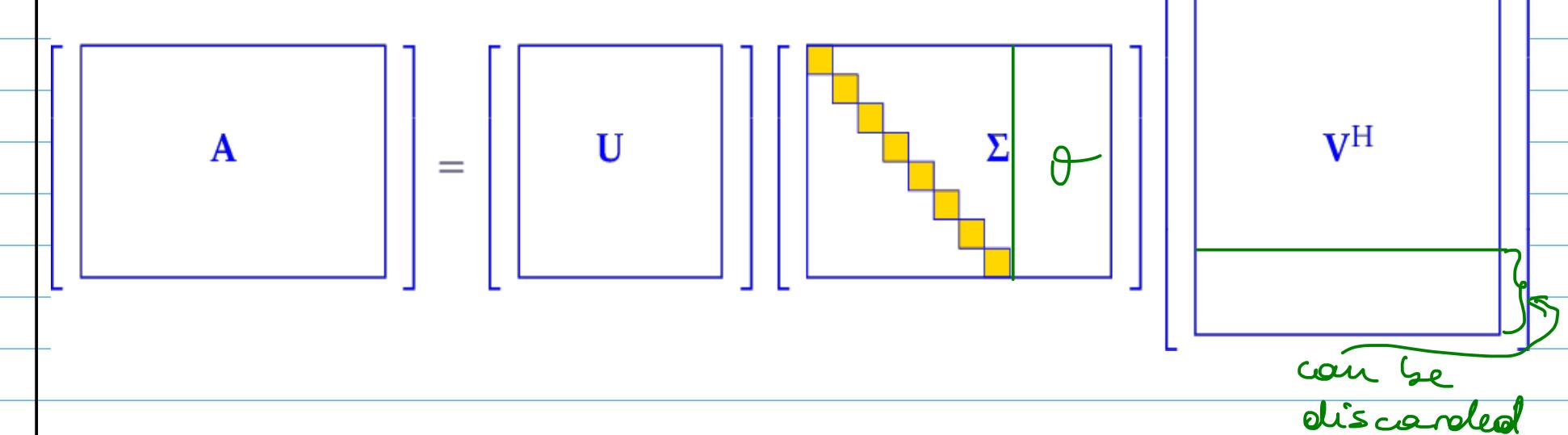


### Theorem 3.4.1 (Singular value decomposition)

For any  $A \in \mathbb{K}^{m,n}$ , there are unitary matrices  $U \in \mathbb{K}^{m,m}$ ,  $V \in \mathbb{K}^{n,n}$  and a (generalized) diagonal (\*) matrix  $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_p) \in \mathbb{R}^{m,n}$ ,  $p := \min\{m, n\}$ ,  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$ , such that

$$A = U \Sigma V^H.$$

Scheme of SVD for "fat"  $A$ : ( $m < n$ )



Reduced SVD for  $m > n$ :

$$\mathbf{A} = \mathbf{U} \begin{bmatrix} \Sigma \\ \vdots \end{bmatrix} \mathbf{V}^H$$

$\Sigma \in \mathbb{K}^{n, n}$   
 $\mathbf{U} \in \mathbb{K}^{m, n}$

Case  $m < n$ : reduced SVD:  $\Sigma \in \mathbb{K}^{m, m}$

$$\mathbf{V} \in \mathbb{K}^{n, m}$$

Facts from linear algebra:

**Lemma 3.4.1.** The squares  $\sigma_i^2$  of the non-zero singular values of  $\mathbf{A}$  are the non-zero eigenvalues of  $\mathbf{A}^H \mathbf{A}$  and  $\mathbf{A} \mathbf{A}^H$  with associated eigenvectors  $(\mathbf{V})_{:,1}, \dots, (\mathbf{V})_{:,p}$  and  $(\mathbf{U})_{:,1}, \dots, (\mathbf{U})_{:,p}$ , respectively.

**Lemma 3.4.2** (SVD and rank of a matrix). If, for some  $1 \leq r \leq p := \min\{m, n\}$ , the singular values of  $\mathbf{A} \in \mathbb{K}^{m,n}$  satisfy  $\sigma_1 \geq \dots \geq \sigma_r > \sigma_{r+1} = \dots = \sigma_p = 0$ , then

- $\text{rank}(\mathbf{A}) = r$  (no. of non-zero singular values),
- $\mathcal{N}(\mathbf{A}) = \text{Span}\{(\mathbf{V})_{:,r+1}, \dots, (\mathbf{V})_{:,n}\}$ ,
- $\mathcal{R}(\mathbf{A}) = \text{Span}\{(\mathbf{U})_{:,1}, \dots, (\mathbf{U})_{:,r}\}$ .

Implication: • SVD determines  $\text{rank}(\mathbf{A})$

• with  $\mathbf{U}$  &  $\mathbf{V}$  we obtain ONB

for  $\mathcal{R}(\mathbf{A})$  &  $\mathcal{N}(\mathbf{A})$ !

↳ useful for construction of Moore-Penrose inverse

In EIGEN: Jacobi SVD is numerically stable

either compute sing. values only or

add flags ComputeThinU, ComputeThinV

ComputeFullU, ComputeFullV

cost of thin SVD:  $\mathcal{O}(mn^2)$   $m \geq n$ .

Generalized solutions by SVD

Assume  $A \in \mathbb{K}^{m,n}$   $m \geq n$   $\text{rank}(A) = r \leq n$

$$A = [U_1 \ U_2] \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix}$$

$$\Sigma_r = \text{diag}(\sigma_1, \dots, \sigma_r)$$

least squares:  $\min \|Ax - b\|_2$

$$\|Ax - b\|_2^2 = \|U \Sigma V^H x - b\|_2^2 = \|U^H (U \Sigma V^H x - b)\|_2^2$$

$$= \|\sum V_i^H x - U^H b\|_2^2$$

$$= \left\| \begin{bmatrix} \sum_r V_i^H x \\ \vdots \\ 0 \end{bmatrix} - \begin{bmatrix} U_1^H b \\ U_2^H b \end{bmatrix} \right\|_2^2$$

$$= \left\| \begin{bmatrix} \sum_r V_i^H x - U_1^H b \\ -U_2^H b \end{bmatrix} \right\|_2^2$$

$$= \left\| \sum_r V_i^H x - U_1^H b \right\|_2^2 + \underbrace{\|U_2^H b\|_2^2}_{\substack{\uparrow \\ \|\cdot\|_{K^r}^2}} \underbrace{\quad}_{\text{fixed}}$$

equivalently:  $\min_x \|\sum_r V_i^H x - U_1^H b\|_2^2$

$$\sum_r V_1^H x = U_1^H b \quad \text{LSE} \quad r \times n$$

If  $r < n$ : non-uniqueness from  $\mathcal{N}(V_1^H)$

$$\mathcal{N}(V_1^H) = \mathbb{R}(V_1)^{\perp}$$

Choose  $x \in \mathbb{R}(V_1)$

$$x = V_1 z \quad \text{for some } z \in \mathbb{K}^r$$

$$\sum_r V_1^H \underbrace{V_1 z}_{=I} = U_1^H b$$

$$\begin{aligned} \sum_r z &= U_1^H b \\ \Rightarrow z &= \sum_r^{-1} U_1^H b \end{aligned}$$

$$\text{Generalized } x : V_1 z = V_1 \sum_r^{-1} U_1^H b$$

Theorem (Pseudoinverse and SVD):

If  $A \in \mathbb{K}^{m,n}$  has the SVD  $A = U \Sigma V^H$ , then its Moore-Penrose Pseudoinverse is given by

$$A^+ = V \sum_r^{-1} U^H$$

SVD-based optimization & approximation

Norm-based Extremes

Consider the following minimization problem:

Given  $A \in \mathbb{K}^{m,n} \quad m \geq n$

find  $x \in \mathbb{K}^n$  s.t.  $\|Ax\|_2 \rightarrow \min, \|x\|_2 = 1$

Use SVD:  $A = U\Sigma V^H$

$$\min_{\|x\|_2=1} \|Ax\|_2^2 = \min_{\|x\|_2=1} \|U\Sigma V^H x\|_2^2 = \min_{\|x\|_2=1} \|\underbrace{\Sigma(V^H x)}_y\|_2^2$$

$$\begin{aligned} &= \min_{\|y\|_2=1} \|\Sigma y\|_2^2 = \min_{\|y\|_2=1} (\tilde{\sigma}_1^2 y_1^2 + \dots + \tilde{\sigma}_n^2 y_n^2) \\ &\quad = \tilde{\sigma}_n^2 \end{aligned}$$

$$\tilde{\sigma}_1 \geq \dots \geq \tilde{\sigma}_n$$

minima value is attained by  $y = e_n$

$$\Rightarrow V^H x = y = e_n$$

$$\Rightarrow x = Ve_n = (V)_{1:n}$$

## Best low-rank approximation

For a given matrix  $A$ , find a low-rank matrix  $\tilde{A}$  that is the closest to  $A$  and has a desired rank  $k$ .

Find a matrix  $\tilde{A} \in \mathbb{K}^{m,n}$ ,  $\text{rank}(\tilde{A}) \leq k$  s.t.

s.t.  $\|A - \tilde{A}\|_{\text{F}}^2 \rightarrow \min$  over all rank- $k$  matrices

$\|\cdot\|_{\text{F}}$ : Frobenius norm for  $A \in \mathbb{K}^{m,n}$

$$\|A\|_{\text{F}}^2 := \sum_{i=1}^m \sum_{j=1}^n |\alpha_{ij}|^2$$

$\|\cdot\|_{\text{F}}$  can be computed from the sing. values

$$\|A\|_{\text{F}}^2 = \sum_{j=1}^r \tilde{\sigma}_j^2 \quad (\text{rank}(A) = r)$$

Desirable:  $k \ll \min\{m, n\}$

- Applications:
- data compression
  - modelling
  - PCA

Use SVD:  $A = U\Sigma V^H$

$$A = \sum_{i=1}^r \tilde{\sigma}_i (U)_{:,i} (V)_{:,i}^H$$

$\underbrace{\qquad\qquad\qquad}_{\text{outer product}}$

Take  $\tilde{A} = \sum_{i=1}^k \tilde{\sigma}_i (U)_{:,i} (V)_{:,i}^H$

$\tilde{\sigma}_1 \geq \tilde{\sigma}_2 \geq \dots \geq \tilde{\sigma}_r$

Turns out: This is the best rank- $k$  approximation to  $A$ .

$$\mathcal{R}_k(m, n) := \{ F \in \mathbb{K}^{m,n} : \text{rank}(F) \leq k \}$$

$\tilde{A}$  is the closest element in  $\mathcal{R}_k(m, n)$  to  $A$ .



in 2-norm & in Frobenius norm  
 $\uparrow$   
 induced op. norm

Theorem:

Let  $A = U\Sigma V^H$  be the SVD of  $A \in \mathbb{K}^{m,n}$ .

For  $k \in \{1, \dots, \text{rank}(A)\}$ , set

$$U_k := [(U)_{:,1}, \dots, (U)_{:,k}] \in \mathbb{K}^{m,k}$$

$$V_k := [(V)_{:,1}, \dots, (V)_{:,k}] \in \mathbb{K}^{n,k}$$

$$\Sigma_k := \text{diag}(\tilde{\sigma}_1, \dots, \tilde{\sigma}_k)$$

Then:

$$\|A - U_k \sum_k (V_k)^H\| \leq \|A - F\| \quad \text{if } F \in \mathbb{R}_k(m, n)$$

↑  
↑  
this holds for  
 $\|\cdot\| = \|\cdot\|_F$ ,  $\|\cdot\| = \|\cdot\|_2$

Best low-rank approximation is useful e.g. for compression

For best  $k$ -rank approximation: reduce required memory from  $m \cdot n$  to  $k(m+n-1)$ .

How good is the approximation:

$$A - \tilde{A} = \sum_{i=k+1}^r \sigma_i(U)_{:,i} (V)_{:,i}^H$$

Sing. values of  $A - \tilde{A}$ :  $\sigma_{k+1} \geq \dots \geq \sigma_r$

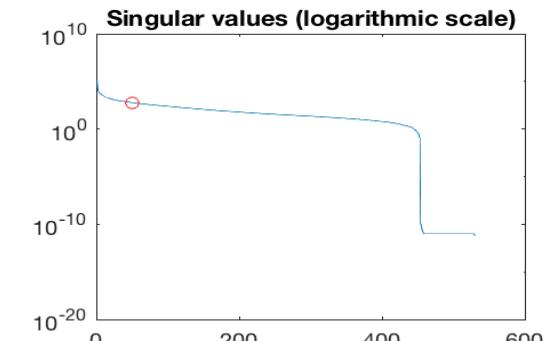
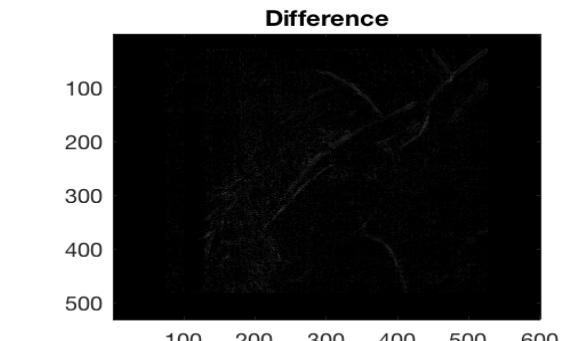
$$\|A - \tilde{A}\|_2 = \sigma_{k+1}$$

$$\|A - \tilde{A}\|_F^2 = \sum_{i=k+1}^r \sigma_i^2$$

Example: Image compression



dim:  $530 \times 600$



Original image:  $530 \cdot 600 \approx 3 \cdot 10^5$

Compressed image:  $50 \cdot \underbrace{1129}_{m+n-1} \approx 5 \cdot 10^4$

Note: In practice, other methods are used

(e.g. ridgelets / curvelets)

## Principle Component Analysis (PCA)





