SI231b: Matrix Computations

Lecture 20: Computations of Singular Value Decomposition

Yue Qiu

qiuyue@shanghaitech.edu.cn

School of Information Science and Technology

ShanghaiTech University

Nov. 23, 2021

SVD and Four Fundamental Subspaces

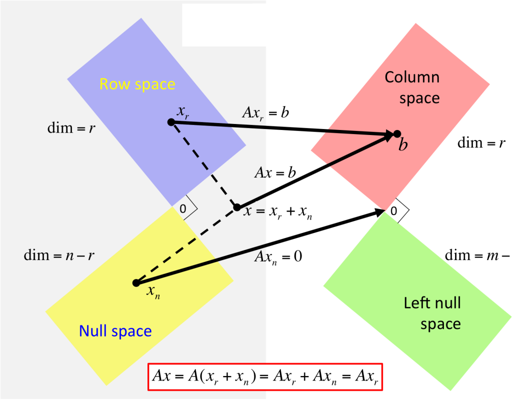


Figure 1: Four fundamental subspaces

In lecture 3, we have learnt that for $\mathrm{A} \in \mathbb{R}^{m \times n}$

- $\mathcal{R}(\mathrm{A}) \perp \mathcal{N}(\mathrm{A}^T)$, and $\mathcal{R}(\mathrm{A}) \oplus \mathcal{N}(\mathrm{A}^T) = \mathbb{R}^m$
- $\mathcal{R}(\mathrm{A}^T) \perp \mathcal{N}(\mathrm{A})$, and $\mathcal{R}(\mathrm{A}^T) \oplus \mathcal{N}(\mathrm{A}) = \mathbb{R}^n$

Property: The following properties hold:

(a) $\mathcal{R}(A) = \mathcal{R}(U_1)$, $\mathcal{R}(A)^\perp = \mathcal{N}(A^T) = \mathcal{R}(U_2)$;

(b) $\mathcal{R}(A^T) = \mathcal{R}(V_1)$, $\mathcal{R}(A^T)^\perp = \mathcal{N}(A) = \mathcal{R}(V_2)$;

(c) $\text{rank}(A) = r$ (the number of nonzero singular values).

Requires a proof.

Note:

- SVD can be used as a numerical tool to compute basis of $\mathcal{R}(A)$, $\mathcal{R}(A)^\perp$, $\mathcal{R}(A^T)$, $\mathcal{N}(A)$
- we have previously learnt the following properties
  - $\text{rank}(A^T) = \text{rank}(A)$
  - $\dim \mathcal{N}(A) = n - \text{rank}(A)$

  By SVD, the above properties are easily seen to be true.
- SVD is also used as a numerical tool to compute the rank of a matrix.

Matrix 2-Norm

Induced matrix p-norm from the vector p-norm

$$\|\mathsf{A}\|_p = \max_{\mathsf{x}\neq 0} \frac{\|\mathsf{A}\mathsf{x}\|_p}{\|\mathsf{x}\|_p} = \max_{\|\mathsf{x}\|_p=1} \|\mathsf{A}\mathsf{x}\|_p$$

$p = 2$: matrix 2-norm or spectral norm

$$\|\mathsf{A}\|_2 = \sigma_{\max}(\mathsf{A}).$$

Proof:

- for any x with $\|\mathsf{x}\|_2 \leq 1$,
$$\begin{aligned}\|\mathsf{A}\mathsf{x}\|_2^2 &= \|\mathsf{U}\boldsymbol{\Sigma}\mathsf{V}^T\mathsf{x}\|_2^2 = \|\boldsymbol{\Sigma}\mathsf{V}^T\mathsf{x}\|_2^2 \\ &\leq \sigma_1^2\|\mathsf{V}^T\mathsf{x}\|_2^2 = \sigma_1^2\|\mathsf{x}\|_2^2 \leq \sigma_1^2\end{aligned}$$
- $\|\mathsf{A}\mathsf{x}\|_2 = \sigma_1$ if we choose $\mathsf{x} = \mathsf{v}_1$

Implication to linear transformation: let $\mathsf{y} = \mathsf{A}\mathsf{x}$ be a linear transformation maps x to y. Under the constraint $\|\mathsf{x}\|_2 = 1$, the system output $\|\mathsf{y}\|_2^2$ is maximized when x is chosen as the 1st right singular vector.

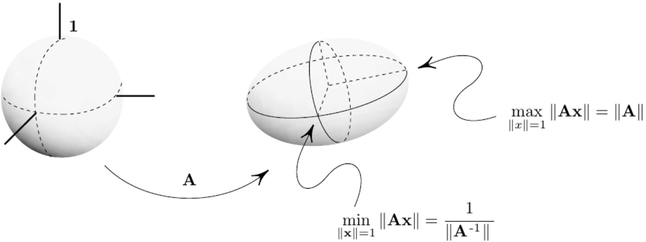


Figure 2: Linear transformation by nonsingular matrix A

When $A \in \mathbb{R}^{m \times n}$,

- $\|Ax\|_2 \geq \sigma_{\min}(A)\|x\|_2$ (hands-on exercise)

A Bad Idea for SVD Computations

For $A \in \mathbb{R}^{m \times n}$, from the SVD $A = U\boldsymbol{\Sigma}V^T$, we obtain that

$$A^T A = V\Lambda V^T, \qquad \Lambda = \boldsymbol{\Sigma}^T\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1^2, \dots, \sigma_p^2, \underbrace{0, \dots, 0}_{n-p \text{ zeros}}) \qquad (*)$$

Equation $(*)$ shows that the singular values of A are positive square roots of eigenvalues of $A^T A$. Thus, one might compute the SVD of A as follows:

1. Form $A^T A$;
2. Compute the eigenvalue decomposition $A^T A = V\Lambda V^T$;
3. Let $\boldsymbol{\Sigma}$ be an $m \times n$ diagonal matrix with diagonal entries being the nonnegative square root of diagonal entries of Λ;
4. Solve the system $U\boldsymbol{\Sigma} = AV$ for orthogonal matrix U (e.g., via QR factorization)

Remark: this approach is numerically unstable. Only used by people who rediscovered the SVD for themselves. [cf. Lecture 31 of Trefethen & Bau 97']

Two Stages of SVD Computations

Recall from previous lectures that in order to solve real symmetric/Hermitian eigenvalue problems,

1. Reduce the matrix to a tridiagonal form;
2. Perform QR iterations (with shift) to obtain a diagonal form.

Similarly, two stages of computations can be performed for the SVD

1. Applying orthogonal transformations to transform a matrix to bi-diagonal form;
2. Diagonalize the bi-diagonal matrix.

$$A = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \xrightarrow{\text{Stage 1}} \begin{bmatrix} \times & \times & & \\ & \times & \times & \\ & & \times & \times \\ & & & \times \\ & & & \end{bmatrix} \xrightarrow{\text{Stage 2}} \begin{bmatrix} \times & & & \\ & \times & & \\ & & \times & \\ & & & \times \\ & & & \end{bmatrix}$$

General Idea

Applying Householder reflectors alternatively on the left and right

- Each left reflection introduces zeros below the diagonal;
- Each right reflection introduces a row of zeros to the right of the first super-diagonal.

$$\mathrm{A} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \xrightarrow{\mathrm{U}_1} \underbrace{\begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \end{bmatrix}}_{\tilde{\mathrm{A}}_1 = \mathrm{U}_1 \mathrm{A}} \xrightarrow{\mathrm{V}_1^T} \underbrace{\begin{bmatrix} \times & \times & 0 & 0 \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \end{bmatrix}}_{\mathrm{A}_1 = \mathrm{U}_1 \mathrm{A} \mathrm{V}_1^T}$$

Here

- U_1 is the Householder reflector that reflects $\mathrm{A}(1:m,1)$
- $\tilde{\mathrm{V}}_1^T$ is the Householder reflector that reflects $\tilde{\mathrm{A}}_1(1,2:n)$ and $\mathrm{V}_1 = \mathrm{diag}(1,\ \tilde{\mathrm{V}}_1)$

Stage 1: Golub-Kahan Bidiagonalization

Following the above procedure, in the end we obtain (for $m > n$)

$$\mathsf{B} = \underbrace{\mathsf{U}_n \mathsf{U}_{n-1} \cdots \mathsf{U}_1}_{\mathsf{U}_a^T} \mathsf{A} \underbrace{\mathsf{V}_1^T \mathsf{V}_2^T \cdots \mathsf{V}_{n-1}^T}_{\mathsf{V}_a},$$

where B is a bi-diagonal matrix that has the form

$$\mathsf{B} = \begin{bmatrix} \alpha_1 & \beta_1 & & \\ & \alpha_2 & \ddots & \\ & & \ddots & \beta_{n-1} \\ & & & \alpha_n \\ & & & \end{bmatrix}.$$

It can be verified that $\alpha_i \geq 0$ and $\beta_i \geq 0$.

If only economic SVD is needed, the Golub-Kahan bidiagonalization procedure can be easily computed through two-term recursions, for details, cf.

https://www.netlib.org/utk/people/JackDongarra/etemplates/node198.html

Stage 2: SVD of Bi-diagonal Form

After stage 1, we obtain that $U_a^T A V_a = B$ where B is a bi-diagonal matrix. The goal of the next step is to compute the bi-diagonal SVD (bSVD) of B, i.e., $B = U_b \mathbf{\Sigma}_b V_b^T$.

Partition $U_b = \begin{bmatrix} U_b^{(1)} & U_b^{(2)} \end{bmatrix}$ where $U_b^{(1)}$ has n columns and let

$$Q = \frac{1}{\sqrt{2}} \begin{bmatrix} V_b & V_b & 0 \\ U_b^{(1)} & -U_b^{(1)} & \sqrt{2} U_b^{(2)} \end{bmatrix},$$

where $Q \in \mathbb{R}^{(m+n)\times(m+n)}$, then

$$Q^T \begin{bmatrix} 0 & B^T \\ B & 0 \end{bmatrix} Q = \mathrm{diag}(\sigma_1, \ \sigma_2, \ \cdots, \ \sigma_n, \ -\sigma_1, \ -\sigma_2, \ \cdots, \ -\sigma_n, \ \underbrace{0, \ 0, \ \cdots, \ 0}_{m-n}).$$

The task is to solve a real symmetric eigenvalue problem. After this stage, the SVD is given by $A = \underbrace{U_a U_b}_{U} \mathbf{\Sigma}_b \underbrace{V_b^T V_a^T}_{V^T}$

More on Bi-diagonal SVD

Permutations can be applied to the Golub-Kahan matrix $\begin{bmatrix} 0 & B^T \\ B & 0 \end{bmatrix}$ so that

$$\Pi \begin{bmatrix} 0 & B^T \\ B & 0 \end{bmatrix} \Pi^T$$

is a tridiagonal matrix.

Popular methods for eigenvalue computations of symmetric tridiagonal matrices such as divide-and-conquer, or relatively robust representations (RRR) can be applied. (cf. Chapter 8.4 of [Golub & van Loan13'] and [Großer & Lang03'] for your further interest)

- B. Großer and B. Lang. An $\mathcal{O}(n^2)$ algorithm for the bi-diagonal SVD. *Linear Algebra and Its Applications*, vol. 358, pp. 45–70, 2003.

One can compute the eigenvalue decomposition of $B^T B$ in a smart way to avoid numerical instability. (cf. Chapter 8.6 of [Golub & van Loan13'] for your further interest)

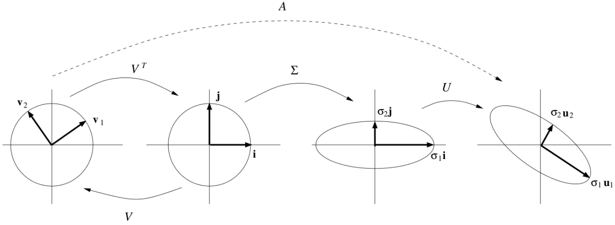
Geometric Interpretation of SVD

For any orthogonal matrix U, it can be verified that

- $\|Ux\|_2 = \|x\|_2$
- $< x,\ y > = < Ux,\ Uy >$

Applying the matrix A to a vector x can be interpreted by a three-stage procedure:

1. $\tilde{x} = V^T x$, i.e., $\tilde{x}$ is obtained by rotating x
2. $\hat{x} = \boldsymbol{\Sigma}\tilde{x}$, rescaling $\tilde{x}$ for $\hat{bx}$
3. $y = U\hat{x}$, rotating $\hat{x}$ for y.



Sensitivity of Linear Systems

For a nonsingular matrix A, we are concerned with the solution of the linear system $Ax = b$.

Question: if there is a small perturbation in A, what is the distance between the perturbed solution and exact solution x?

$$(A + \Delta A)(x + \Delta x) = b$$

From Lecture 6, we know that

$$\frac{\|\Delta x\|}{\|x\|} \leq \|A\|\|A^{-1}\| \frac{\|\Delta A\|}{\|A\|},$$

where $\|A\|\|A^{-1}\|$ is defined as the condition number of the matrix A and is denoted by $\kappa(A)$.

Note: $\kappa(A) \geq 1$ (how to prove?)

Sensitivity of Linear Systems

When the matrix 2-norm is used,

$$\kappa_2(A) = \|A\|_2\|A^{-1}\|_2 = \frac{\sigma_{\max}(A)}{\sigma_{\min}(A)}.$$

$\sigma_{\min}(A)$ measures the distance of A to singularity. For orthogonal matrix A, $\kappa_2(A) = 1$.

When $\sigma_{\min}(A)$ is close to zero,

- $\kappa_2(A)$ gets large $\rightsquigarrow$ small perturbation may lead to large solution error

$$x = A^{-1}b = V\Sigma^{-1}U^Tb = \sum_i \frac{u_i^T b}{\sigma_i} v_i$$

- inverting A gets more difficult and unstable

Note: $\kappa_2(A^TA) = \kappa_2(AA^T) = \kappa(A)^2$. (Can you prove this?)

This explains why forming problems with A^TA or AA^T is (almost) a bad idea.

Equivalence of Condition Number

The matrix A is said to be ill-conditioned if $\kappa(\mathrm{A})$ is large. This statement is a norm dependent property.

Any two condition numbers $\kappa_\alpha(\cdot)$ and $\kappa_\beta(\cdot)$ are equivalent on $\mathbb{R}^{m\times n}$, which means that constants c_1 and c_2 can be found so that

$$c_1\kappa_\alpha(\mathrm{A}) \leq \kappa_\beta(\mathrm{A}) \leq c_2\kappa_\alpha(\mathrm{A}), \quad \forall \mathrm{A} \in \mathbb{R}^{m\times n}.$$

For example, for $\mathrm{A} \in \mathbb{R}^{m\times n}$,

$$\frac{1}{n}\kappa_2(\mathrm{A}) \leq \kappa_1(\mathrm{A}) \leq n\kappa_2(\mathrm{A})$$

$$\frac{1}{n}\kappa_\infty(\mathrm{A}) \leq \kappa_2(\mathrm{A}) \leq n\kappa_\infty(\mathrm{A})$$

$$\frac{1}{n^2}\kappa_1(\mathrm{A}) \leq \kappa_\infty(\mathrm{A}) \leq n^2\kappa_1(\mathrm{A})$$

Therefore, if a matrix is ill-conditioned in the α-norm, it is also ill-conditioned in the β-norm.

Note: all vectors norms are equivalent and all matrix norms are also equivalent. (cf. Chapter 2.2 and 2.3 of [Golub & van Loan13'] for details.)

Recall from Lecture 10, for $\mathrm{A} \in \mathbb{R}^{m \times n}$, the pseudoinverse of A denoted by $\mathrm{A}^\dagger \in \mathbb{R}^{n \times m}$ satisfying the Moore–Penrose conditions.

1. $\mathrm{A}\mathrm{A}^\dagger\mathrm{A} = \mathrm{A}$
2. $\mathrm{A}^\dagger\mathrm{A}\mathrm{A}^\dagger = \mathrm{A}^\dagger$
3. $(\mathrm{A}\mathrm{A}^\dagger)^T = \mathrm{A}\mathrm{A}^\dagger$
4. $(\mathrm{A}^\dagger\mathrm{A})^T = \mathrm{A}^\dagger\mathrm{A}$

- R. Penrose. A Generalized Inverse for Matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406-413, 1955.

For a rank r matrix A, its SVD is given by

$$\mathrm{A} = \begin{bmatrix} \mathrm{U}_1 & \mathrm{U}_2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{\Sigma}} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathrm{V}_1^T \\ \mathrm{V}_2^T \end{bmatrix},$$

where $\tilde{\mathbf{\Sigma}} = \mathrm{diag}(\sigma_1, \dots, \sigma_r)$, $\mathrm{U}_1 \in \mathbb{R}^{m \times r}$, $\mathrm{V}_1 \in \mathbb{R}^{n \times r}$. Then we get $\mathrm{A}^\dagger = \mathrm{V}_1\tilde{\mathbf{\Sigma}}^{-1}\mathrm{U}_1^T$

Note: it is not necessary that $\mathrm{A}^\dagger\mathrm{A} = \mathrm{I}$ or $\mathrm{A}\mathrm{A}^\dagger = \mathrm{I}$

Readings

You are supposed to read

1. Lloyd N. Trefethen and David Bau III. Numerical Linear Algebra, *SIAM*, 1997.

   Lecture 31

For your own interest, if you want to dig into SVD computations, you are recommended to read

(a) Gene H. Golub and Charles F. Van Loan. Matrix Computations, *Johns Hopkins University Press*, 2013.

   Chapter 8.4, 8.6.