SVD applications
Lecture 4
Warmup Problem

Someone hands you the SVD of a matrix $A$:

$$A = U S V^\top$$

1. What is the SVD of $A$ times its transpose?

$$AA^\top = ?$$

2. What is the SVD of $A$-transpose times $A$?

$$A^\top A = ?$$
Warmup Problem

Recall: \((AB)^\top = B^\top A^\top\)

Someone hands you the SVD of a matrix \(A\):

\[ A = U S V^\top \]
\[ A^\top = V S U^\top \]

1. What is the SVD of \(A\) times its transpose?

\[ AA^\top = \ ? \]

2. What is the SVD of \(A\)-transpose times \(A\)?

\[ A^\top A = \ ? \]
Answer:

Recall: \((AB)^\top = B^\top A^\top\)

Someone hands you the SVD of a matrix A:

\[ A = USV^\top \]
\[ A^\top = VSU^\top \]

1. What is the SVD of A times its transpose?

\[ AA^\top = (USV^\top)(VSU^\top) \]
\[ = US^2U^\top \]

2. What is the SVD of A-transpose times A?

\[ A^\top A = \ ? \]
Answer:

Recall: \((AB)^\top = B^\top A^\top\)

Someone hands you the SVD of a matrix \(A\):

\[ A = USV^\top \]

\[ A^\top = VSU^\top \]

1. What is the SVD of \(A\) times its transpose?

\[ AA^\top = (USV^\top)(VSU^\top) \]
\[ = US^2U^\top \]

2. What is the SVD of \(A^{-}\)transpose times \(A\)?

\[ A^\top A = (VSU^\top)(USV^\top) \]
\[ = VS^2V^\top \]
SVD review

\[ A = U S V^T \]

left singular vectors
(or orthogonal/unitary)

\[ U^T U = U U^T = I \]
SVD review

\[ A = U S V^T \]

\[ \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ u_1 & u_2 & \cdots & u_n \end{bmatrix} \]

left singular vectors
(or orthogonal/unitary)

\[ U^T U = U U^T = I \]

\[ \begin{bmatrix} s_1 & & & \\ & s_2 & & \\ & & \ddots & \\ & & & s_n \end{bmatrix} \]

singular values
(all \( \geq 0 \))

\[ s_1 \geq s_2 \geq \cdots \geq s_n \]
(by convention)
SVD review

$$A = USV^T$$

left singular vectors (orthogonal/unitary)

$$\tilde{U}^T U = UU^T = I$$

right singular vectors (orthogonal/unitary)

$$V^T V = VV^T = I$$

singular values

$$s_1 \geq s_2 \geq \ldots \geq s_n$$

(by convention)
SVD review

\[ A = USV^\top \]

inverse: \[ A^{-1} = V S^{-1} U^\top \]
SVD review

\[ A = U S V^\top \]

inverse:  
\[ A^{-1} = V S^{-1} U^\top \]

pseudo-inverse:  
\[ A^\dagger = V S^\dagger U^\top \]
Questions (group discussion)

How could you use SVD to:

1. determine whether a matrix is invertible?
2. find the rank of a matrix?
3. find an orthonormal basis for the row space?
4. find an orthonormal basis for the column space?
5. find an orthonormal basis for the *null space*?
Answers:

1. determine whether a matrix is invertible?
   Invertible if all singular values are > 0.

2. find the rank of a matrix?
   rank = # of non-zero singular values
3. Find an orthonormal basis for the row space?
4. Find an orthonormal basis for the column space?
5. Find an orthonormal basis for the null space?

- Note that any linear combination of \( (v_{k+1}, ..., v_n) \) has zero dot product with \( v_1...v_k \), hence gives zero when multiplied by \( A \) (and is thus in null space!)
The degree to which ill-conditioning prevents a matrix from being inverted accurately depends on the ratio of its largest to smallest singular value, a quantity known as the condition number:

\[ \text{condition number} = \frac{s_1}{s_n} \]

The larger the condition number, the more practically non-invertible it is. When using double floating point precision, matrices with condition numbers greater than \( \pi \times 10^{14} \) cannot be stably inverted.

You can compute the condition number yourself from the SVD, or using the built-in Matlab command `cond`, or the numpy command `numpy.linalg.cond`.

If \( A_{m \times n} \) is a non-square matrix, then \( U \) is \( m \times m \) and \( V \) is \( n \times n \), and \( S \) is non-square (and therefore has only \( \min(m, n) \) non-zero singular values. Such matrices are (obviously) non-invertible, though we can compute their pseudo-inverses using the formula above.

Figure 3: SVD of non-square matrices. The gray regions of the matrices are not needed, since they consist of all zeros in the \( S \) matrix, and are only hit by zeros in those portions of the \( U \) or \( V \) matrices. Dropping them results in the more compact "reduced" SVD for tall, skinny (above) or short, fat (below) matrices.
true vs. practical non-invertibility
true vs. practical non-invertibility

condition number: \( \frac{s_1}{s_n} \)

- matrix is not *practically* invertible if condition # too big (>10^{12})
- such a matrix called “ill-conditioned” or “singular”
- compute with: `numpy.linalg.cond`
eigenvectors

what is an eigenvector?

Q: when is an eigenvector equal to a singular vector?
eigenvectors

- for a (square) matrix $A$, a vector $\vec{x}$ such that

$$A\vec{x} = \lambda \vec{x}$$

that is, $Ax$ is a scaled version of $x$. 

\[ \begin{align*}
\text{eigenvalue} & \quad \text{eigenvector}
\end{align*} \]
positive semi-definite matrix

- matrix for which all eigenvalues are $\geq 0$

- equivalently definition:

  $$\bar{x}^\top A\bar{x} \geq 0 \text{ for any vector } \bar{x}$$
Spectral theorem

If a matrix $A$ is
• symmetric
• positive semi-definite

the singular value decomposition is also an eigen-decomposition:

$$A = U S U^\top$$

- matrix of (orthogonal) eigenvectors
- eigenvalues along diagonal
- singular vectors = eigenvectors
- singular values = eigenvalues
- Note that left and right singular vectors are the same!
spectral theorem

If a matrix $A$ is
• symmetric
• positive semi-definite

Then:

$$AA^\top = (VSU^\top)(USV^\top) = VS^2V^\top$$

• $V$ is matrix of orthogonal eigenvectors
• $s_i^2$ are eigenvalues
determinants

\[ \det |A| \]

- measure of the “change in volume” of a hypercube in \( \mathbb{R}^n \) upon multiplying by \( A \)

\[ A = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \]
determinants

\[ \det |A| \]  
- measure of the “change in volume” of a hypercube in \( \mathbb{R}^n \) upon multiplying by \( A \)

\[ A = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \]

\[ \det |A| = 3 \times 2 = 6 \]
determinants

\[ \det |A| \] - measure of the “change in volume” of a hypercube in \( \mathbb{R}^n \) upon multiplying by \( A \)

\[
\det |A| = \det |USU^\top| = \det |S| = \prod_{i=1}^{n} s_i
\]

product of singular values
SVD as a sum of outer-products

\[ A = U S V^\top \]
SVD as a sum of outer-products

$$A = USV^T$$

$$= S_1 u_1 v_1^T + S_2 u_2 v_2^T + \cdots + S_n u_n v_n^T$$
SVD as a sum of outer-products

\[ A = U S V^T \]

\[ = S_1 u_1 v_1^T + S_2 U_2 V_2^T + \ldots + S_n U_n V_n^T \]

(rank 1 matrix)
matrix approximation

- the best rank-$K$ approximation to $A$ (in terms of squared error) is given by truncating the SVD after $K$ terms.

$$= s_1 u_1 v_1^T + \ldots + s_K u_K v_K^T$$
matrix approximation

- the best rank-K approximation to $A$ (in terms of squared error) is given by truncating the SVD after $K$ terms.

\[
\begin{align*}
A & = S_i u_i v_i^T + \ldots + S_K u_K v_K^T \\
\text{Fraction of variance accounted for is given by:} \\
\frac{\sum_{i=1}^{K} S_i^2}{\sum_{j=1}^{N} S_j^2}
\end{align*}
\]
Frobenius norm
(the Euclidean norm for matrices)

\[ \| A \|_F = \sqrt{\sum_{ij} a_{ij}^2} \]

A

\begin{align*}
  a_{11} & \cdots & a_{1m} \\
  a_{21} & \cdots & a_{2m} \\
  \vdots & & \vdots \\
  a_{n1} & \cdots & a_{nm}
\end{align*}

sum of squared elements of A
Frobenius norm

(the Euclidean norm for matrices)

\[ \|A\|_F = \sqrt{\sum_{ij} a_{ij}^2} = \sqrt{\sum_i s_i^2} \]

(see notes for proof)
Thus we can write fraction of variance accounted for as:

\[
\sum_{i=1}^{K} \frac{s_i^2}{\|A\|_F^2} + \ldots + s_k \begin{pmatrix} u_k v_k^T \end{pmatrix}
\]

sum of squared first K singular values

sum of squares of all singular values
matrix approximation: applications to neural data

\[ A \approx s_1 \tilde{u}_1 + \tilde{v}_1^T \text{ timecourse } \pm 1 + \tilde{v}_2^T \text{ timecourse } \pm 2 \]

neurons (firing rate)

\[ s_1 \tilde{u}_1 \text{ neural weights } \pm 1 \]

\[ s_2 \tilde{u}_2 \text{ neural weights } \pm 2 \]
Summary

• inverse
• pseudoinverse
• rank
• condition number
• ill-conditioned / singular matrix
• eigenvectors & eigenvalues
• positive semi-definite matrices
• spectral theorem
• determinants
• low-rank matrix approximation
• Frobenius norm (Euclidean norm for matrices)