Singular Value Decomposition

Singular Value Decomposition

$$A \text{ is } m \times n$$

A = (orthogonal) (diagonal) (orthogonal)

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

$$\begin{bmatrix} x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \end{bmatrix} = \begin{bmatrix} \\ \\ & U & \\ \\ \\ \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \square & \\ & & \sigma_r \\ \\ \\ \end{bmatrix} \begin{bmatrix} \\ & V^T & \\ \\ \end{bmatrix}$$

$$A = U\Sigma V^T \longrightarrow A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \square + \sigma_r u_r v_r^T$$

Applications of the SVD

- **Image processing**
  - **Suppose a satellite takes a picture, and wants to send it to earth. The picture may contain 1000 by 1000 "pixels"—little squares each with a definite color. We can code the colors, in a range between black and white, and send back 1,000,000 numbers**

picture

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

matrix

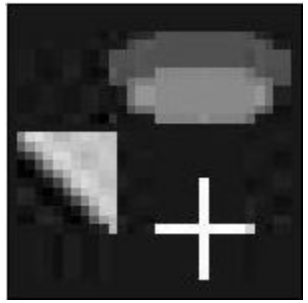
Applications of the SVD: Image processing

- It is better to find the essential information in the 1000 by 1000 matrix, and send only that.
- Typically, some are significant and others are extremely small.
- If we keep 60 and throw away 940, then we send only the corresponding 60 columns of U, and V.
- If only 60 terms are kept, we send 60 times 2000 numbers instead of a million.
- The other 940 columns are multiplied by small singular values that are being ignored. In fact, we can do the matrix multiplication as columns times rows:

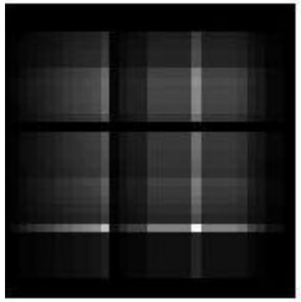
$$A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_r u_r v_r^T$$

Applications of the SVD: Image processing

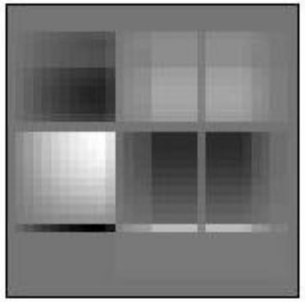
- **The SVD of a 32-times-32 digital image A is computed**



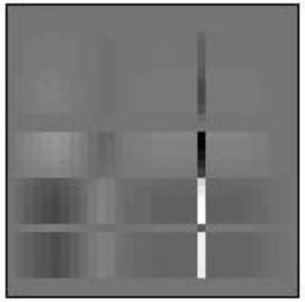
$\sigma_1 u_1 v_1^T$



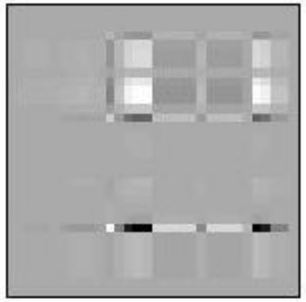
$\sigma_2 u_2 v_2^T$

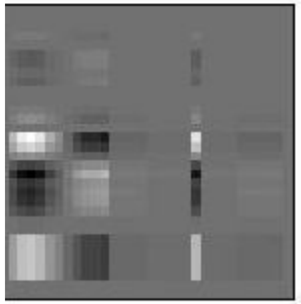


$\sigma_3 u_3 v_3^T$

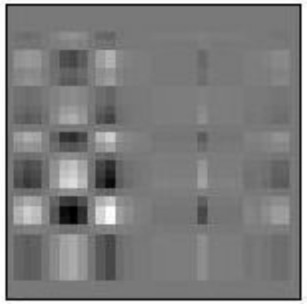


$\sigma_4 u_4 v_4^T$

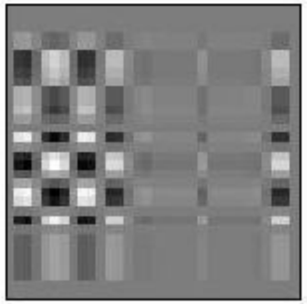




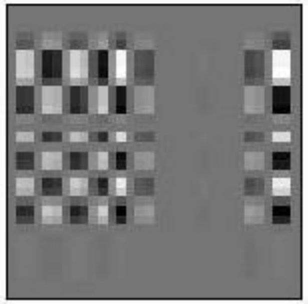
$\sigma_5 u_5 v_5^T$



$\sigma_6 u_6 v_6^T$



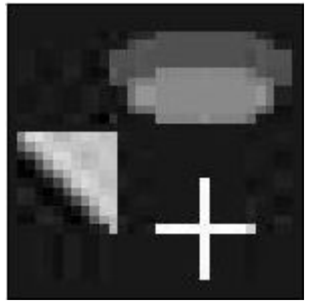
$\sigma_7 u_7 v_7^T$



$\sigma_8 u_8 v_8^T$

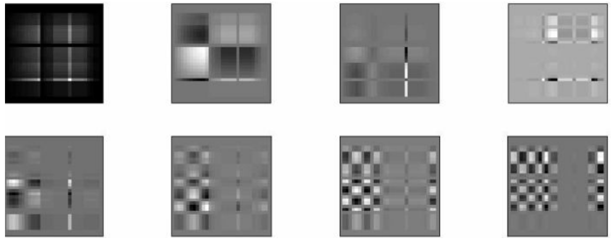
Slide taken from https://faculty.kfupm.edu.sa/MATH/ffairag/math590_111/notes/svd2.ppt

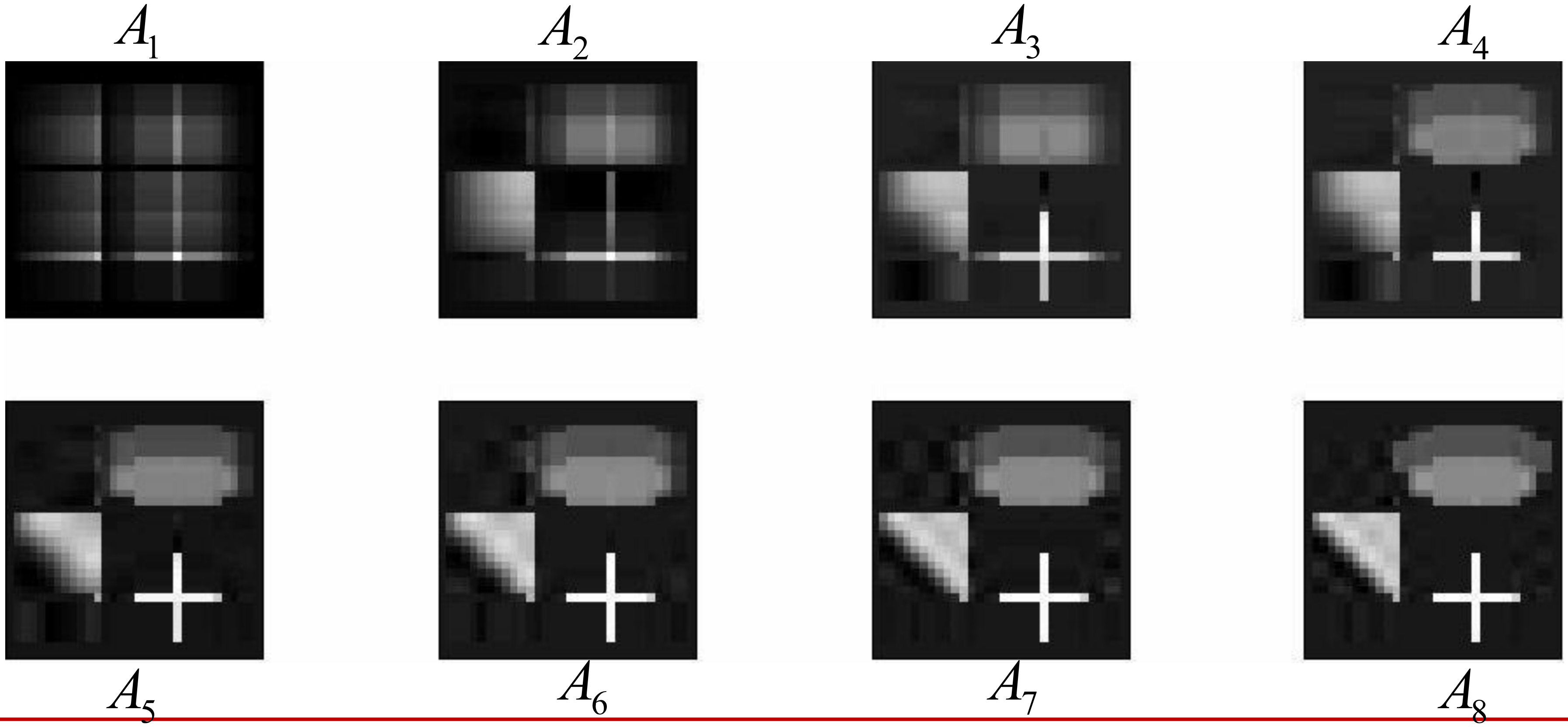
Applications of the SVD: Image processing



$$A_s = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_s u_s v_s^T$$

$$s \le r$$





Slide taken from https://faculty.kfupm.edu.sa/MATH/ffairag/math590_111/notes/svd2.ppt

Inverses

- The SVD makes it easy to compute (and understand) the inverse of a matrix. We exploit the fact that U and V are orthogonal, meaning their transposes are their inverses, i.e., $U^TU = UU^T = I$ and $V^TV = VV^T = I$.
- The inverse of A (if it exists) can be determined easily from the SVD, namely:

$$A^{-1} = \left(V^T\right)^{-1}\Sigma^{-1}(U)^{-1} = V\Sigma^{-1}U^T,$$

where, $\Sigma^{-1} = \begin{pmatrix} 1/\Sigma_{11} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & 1/\Sigma_{\mathrm{nn}} \end{pmatrix}$

For orthogonal matrix X: $(X)^1=(X)^T$

Pseudo-inverse

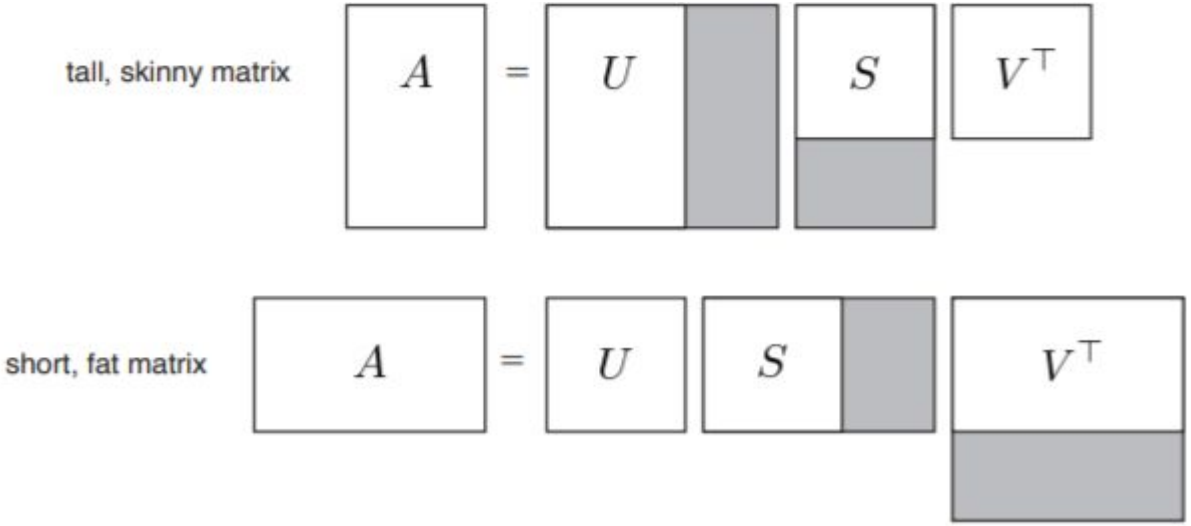
- The SVD also makes it easy to see when the inverse of a matrix doesn't exist. Namely, if any of the singular values $\Sigma_{ii} = 0$, then the Σ^{-1} doesn't exist, because the corresponding diagonal entry would be $1/\Sigma_{ii} = 1/0$.
- If a matrix A has any zero singular values (let's say $\Sigma_{jj} = 0$), then multiplying by A effectively destroys information because it takes the component of the vector along the right singular vector $v$and multiplies it by zero.
  - We can't recover this information, so there's no way to "invert" the mapping Ax to recover the original x that came in.

Pseudo-inverse

- If a matrix A has any zero singular values (let's say $\Sigma_{jj} = 0$), then multiplying by A effectively destroys information because it takes the component of the vector along the right singular vector $v$and multiplies it by zero.
    - We can't recover this information, so there's no way to "invert" the mapping Ax to recover the original x that came in.
- The best we can do is to recover the components of x that weren't destroyed via multiplication with zero.
    - The matrix that recovers all recoverable information is called the pseudo-inverse, and is often denoted $A^{\dagger}$

SVD of non-square matrix

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



Pseudo-inverse

- Suppose we have an $n \times n$ matrix A, which has only k non-zero singular values.
- The pseudoinverse of A can then be written similarly to the inverse: $A^{\dagger} = U\Sigma^{\dagger}\mathbf{V}^{\mathbf{T}}$

$$\boldsymbol{\Sigma}^{\dagger} = \begin{pmatrix} \mathbf{1/\Sigma_{11}} & \cdots & & & & \\ & \ddots & & & & \\ & & \mathbf{1/\Sigma_{kk}} & & & \\ & & & \mathbf{0} & & \\ & & & & \ddots & \\ & & & & & \mathbf{0} \end{pmatrix}$$

Inverses: Condition number

- In practical situations, a matrix may have singular values that are not exactly equal to zero, but are so close to zero that it is not possible to accurately compute them.
    - In such cases, the matrix is what we call ill-conditioned, because dividing by the singular values (values that are arbitrarily close to zero) will result in numerical errors.
- 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.

$$condition\ number\ = \mathbf{\Sigma_{11}/\Sigma_{nn}}$$

- The larger the condition number, the more practically non-invertible it is.

Document matrices

$$\begin{array}{cc} & \begin{array}{c}\textbf{d}\text{ terms}\\ \text{(e.g., theorem, proof, etc.)}\end{array} \\ \begin{array}{l}\textbf{n}\\ \text{documents}\end{array} & \begin{pmatrix} A \\ \\ \begin{array}{l} A_{ij} = \text{frequency of the } j\text{-th} \\ \text{term in the } i\text{-th document} \end{array} \end{pmatrix} \end{array}$$

Find a subset of the terms that accurately clusters the documents

SVD Example

$$
\begin{array}{c|ccccc}
 & \text{data} & \text{inf.} & \text{retrieval} & \text{brain} & \text{lung} \\
\hline
\text{CS} & 1 & 1 & 1 & 0 & 0 \\
 & 2 & 2 & 2 & 0 & 0 \\
 & 1 & 1 & 1 & 0 & 0 \\
 & 5 & 5 & 5 & 0 & 0 \\
\text{MD} & 0 & 0 & 0 & 2 & 2 \\
 & 0 & 0 & 0 & 3 & 3 \\
 & 0 & 0 & 0 & 1 & 1
\end{array}
=
\begin{array}{cc}
\text{CS Concept} & \text{MD-Concept} \\
0.18 & 0 \\
0.36 & 0 \\
0.18 & 0 \\
0.90 & 0 \\
0 & 0.53 \\
0 & 0.80 \\
0 & 0.27
\end{array}
\times
\begin{bmatrix}
9.64 & 0 \\
0 & 5.29
\end{bmatrix}
\times
\begin{array}{cccccl}
0.58 & 0.58 & 0.58 & 0 & 0 & \text{CS Concept} \\
0 & 0 & 0 & 0.71 & 0.71 & \text{MD-Concept}
\end{array}
$$

doc-to-concept similarity matrix

'strength' of CS-concept

term-to-concept similarity matrix

SVD - Interpretation #1

-

SVD - Interpretation #2

- **Q: how exactly is dim. reduction done?**
  - **set the smallest eigenvalues to zero:**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0.18 & 0 \\ 0.36 & 0 \\ 0.18 & 0 \\ 0.90 & 0 \\ 0 & 0.53 \\ 0 & 0.80 \\ 0 & 0.27 \end{bmatrix} \text{x} \begin{bmatrix} 9.64 & 0 \\ 0 & \xcancel{5.29} \end{bmatrix} \text{x}$$

$$\begin{bmatrix} 0.58 & 0.58 & 0.58 & 0 & 0 \\ 0 & 0 & 0 & 0.71 & 0.71 \end{bmatrix}$$

SVD - Interpretation #2

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \sim \begin{bmatrix} 0.18 & 0 \\ 0.36 & 0 \\ 0.18 & 0 \\ 0.90 & 0 \\ 0 & 0.53 \\ 0 & 0.80 \\ 0 & 0.27 \end{bmatrix} \text{x} \begin{bmatrix} 9.64 & 0 \\ 0 & \cancel{0} \end{bmatrix} \text{x} \begin{bmatrix} 0.58 & 0.58 & 0.58 & 0 & 0 \\ 0 & 0 & 0 & 0.71 & 0.71 \end{bmatrix}$$

SVD - Interpretation #2

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \sim \begin{bmatrix} 0.18 & \cancel{0} \\ 0.36 & \cancel{0} \\ 0.18 & \cancel{0} \\ 0.90 & \cancel{0} \\ 0 & \cancel{0.53} \\ 0 & \cancel{0.80} \\ 0 & \cancel{0.27} \end{bmatrix} \text{x} \begin{bmatrix} 9.64 & 0 \\ 0 & \cancel{0} \end{bmatrix} \text{x} \begin{bmatrix} 0.58 & 0.58 & 0.58 & 0 & 0 \\ \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0.71} & \cancel{0.71} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \sim \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Social-network matrices

$$\begin{array}{cc} & \textbf{d}\text{ groups} \\ & \text{(e.g., BU group, opera, etc.)} \\ \textbf{n}\text{ users} & \begin{pmatrix} \\ A \\ \text{A_{ij} = partitiipation of the} \\ \text{i-th user in the j-th group} \\ \end{pmatrix} \end{array}$$

Find a subset of the groups that accurately clusters social-network users

Recommendation systems

d products

n customers

$$A$$

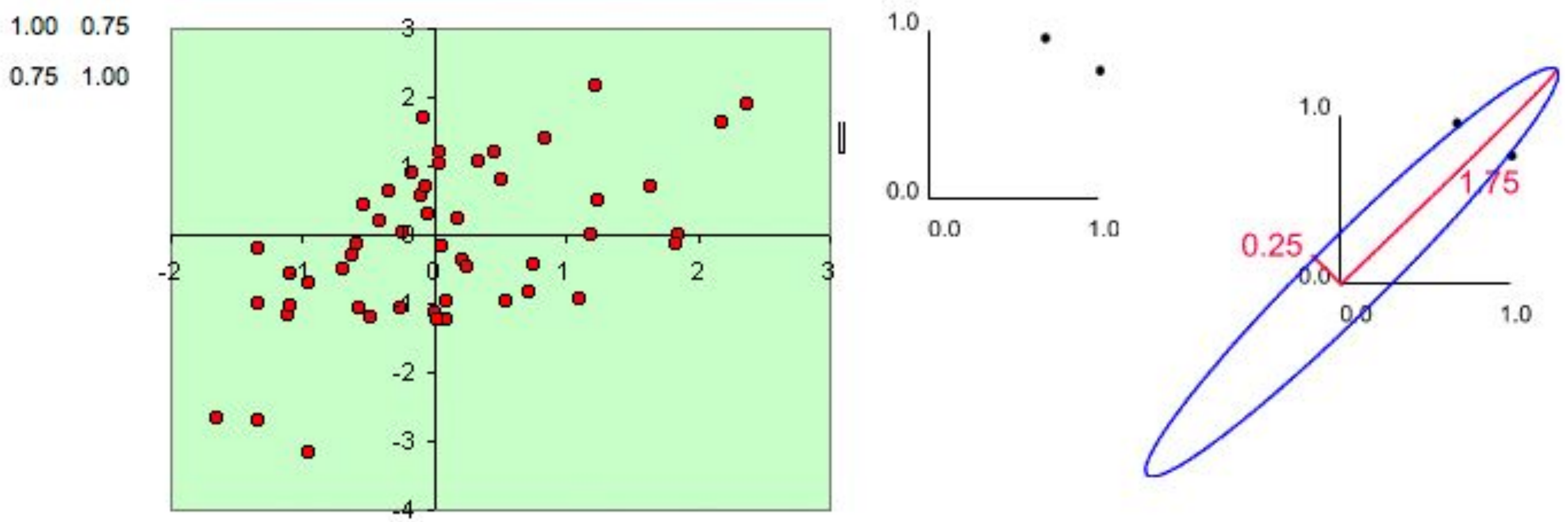
A_{ij} = frequency of the j-th product is bought by the i-th customer

Find a subset of the products that accurately describe the behavior or the customers

Geometrical Interpretation: Eigenvalues and Eigenvectors

- Consider a covariance matrix, A, i.e., A = 1/n S S^T for some S

$$\mathbf{A} = \begin{bmatrix} 1 & .75 \\ .75 & 1 \end{bmatrix} \Rightarrow \lambda_1 = 1.75, \lambda_2 = 0.25$$

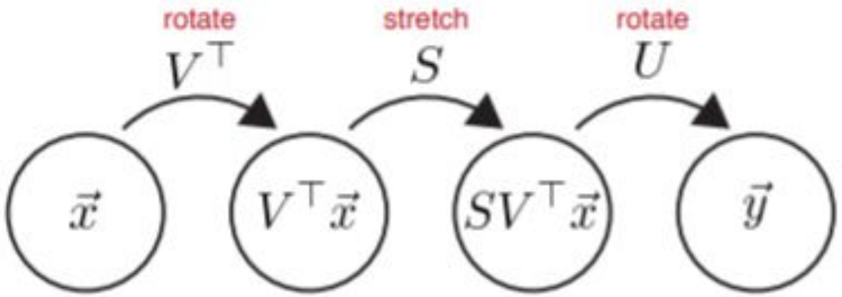


- Ellipse: major axis as the larger eigenvalue and the minor axis as the smaller eigenvalue

SVD - Interpretation #3

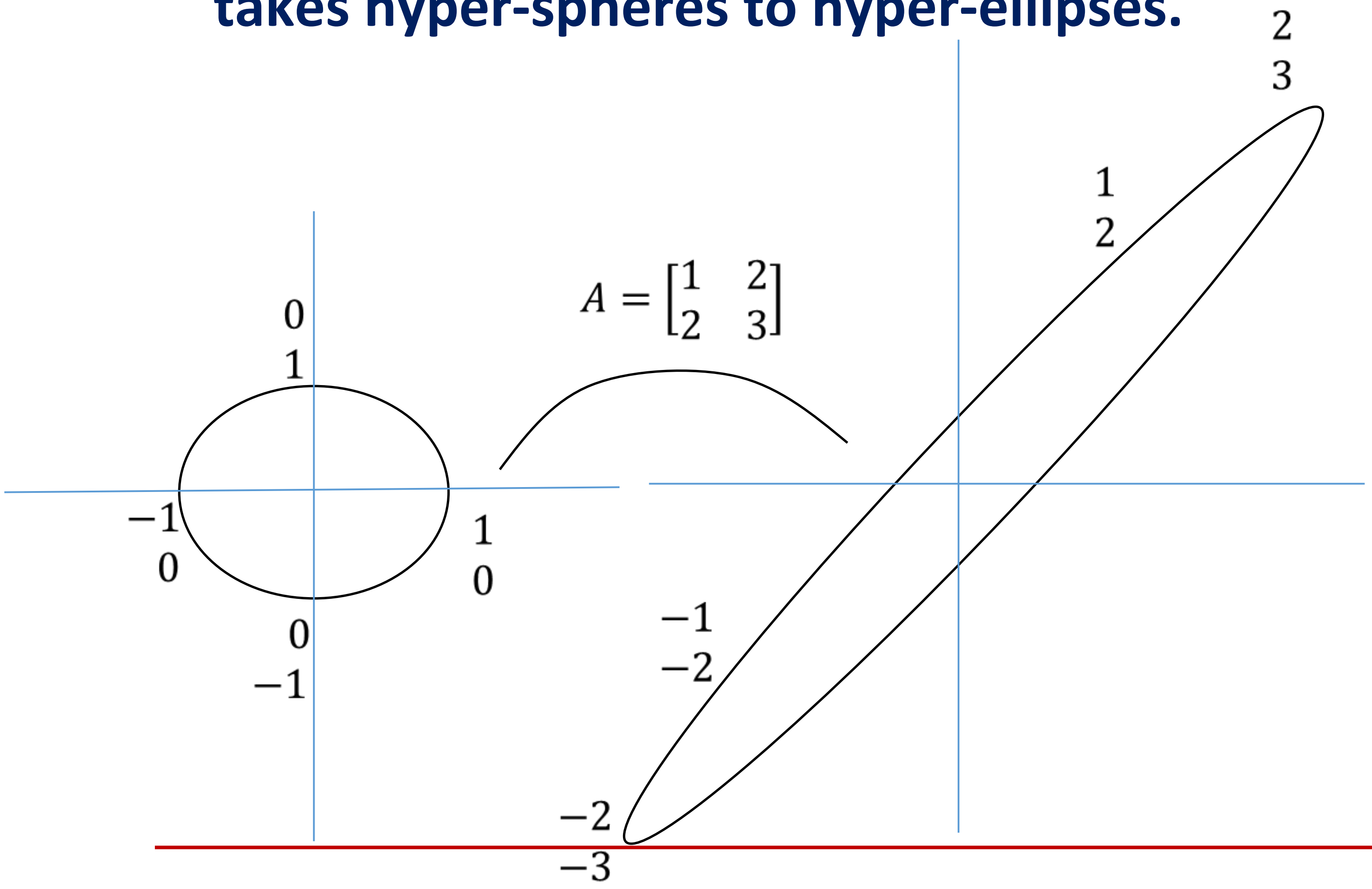
- The SVD tells us that we can think of the action of A upon any vector x in terms of three steps
  1. rotation (multiplication by V which doesn’t change vector length of x).
  2. stretching along the cardinal axes (where the I’th component is stretched by s_i).
  3. another rotation (multipication by U).

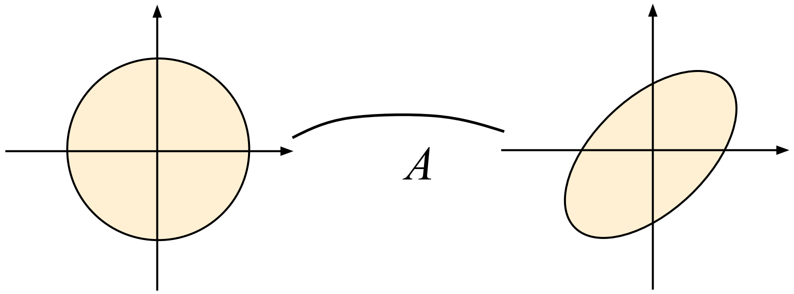
$$\vec{y} = A\vec{x} = (USV^\top)\,\vec{x}$$

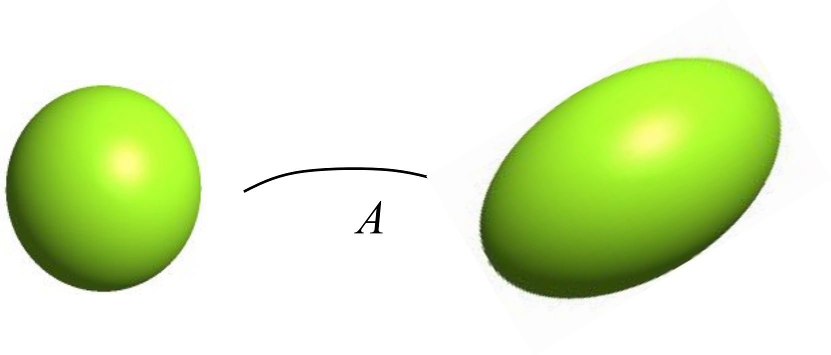


SVD - Interpretation #3

- A linear (non-singular) transform A always takes hyper-spheres to hyper-ellipses.

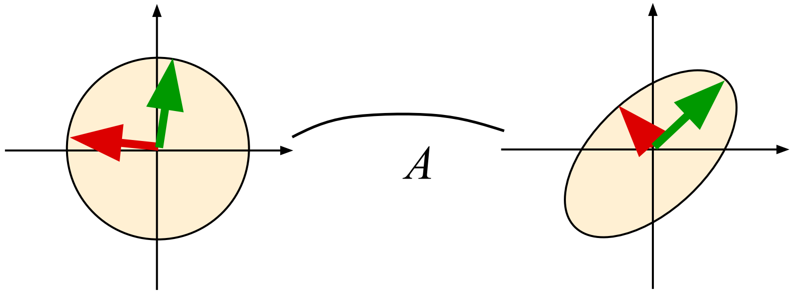

$A = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$
$\begin{matrix} 0 \\ 1 \end{matrix}$
$\begin{matrix} -1 \\ 0 \end{matrix}$
$\begin{matrix} 1 \\ 0 \end{matrix}$
$\begin{matrix} 0 \\ -1 \end{matrix}$
$\begin{matrix} 2 \\ 3 \end{matrix}$
$\begin{matrix} 1 \\ 2 \end{matrix}$
$\begin{matrix} -1 \\ -2 \end{matrix}$
$\begin{matrix} -2 \\ -3 \end{matrix}$

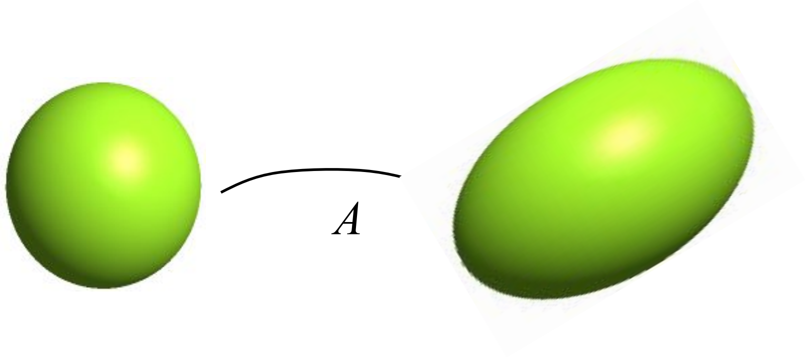




SVD - Interpretation #3

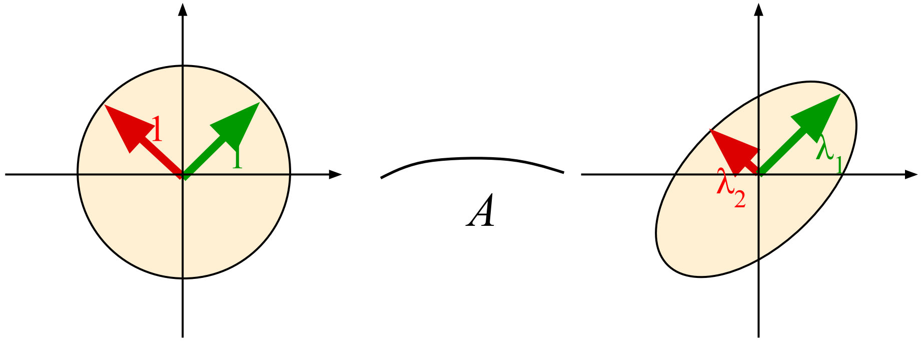
Thus, one good way to understand what A does is to find which vectors are mapped to the “main axes” of the ellipsoid.





SVD - Interpretation #3

- In this case $\mathbf{A}$ is just a scaling matrix.
- The eigen decomposition of $\mathbf{A}$ tells us which orthogonal axes it scales, and by how much:



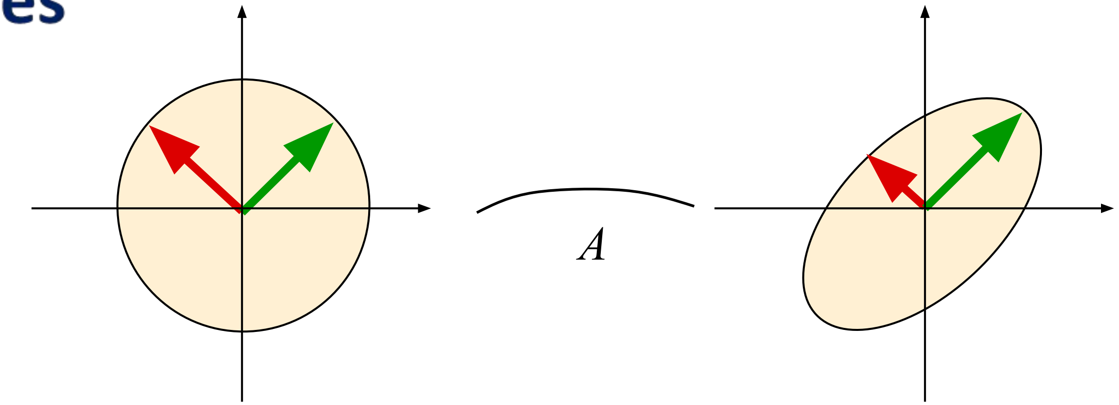
$$\boxed{A\mathbf{v}_i = \lambda_i \mathbf{v}_i}$$

$$A = \left[\mathbf{v}_1\ \mathbf{v}_2 \ldots \mathbf{v}_n\right] \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix} \left[\mathbf{v}_1\ \mathbf{v}_2 \ldots \mathbf{v}_n\right]^T$$

Geometric analysis of linear transformations

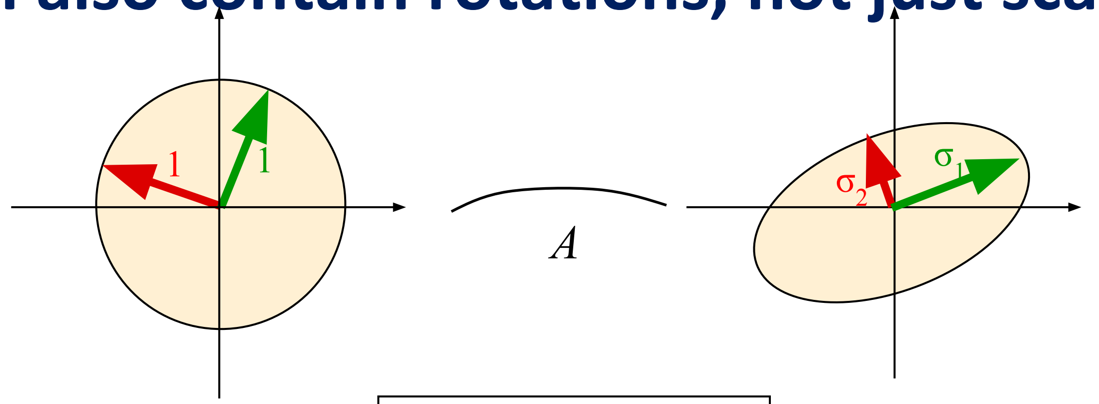
- If we are lucky: $A = V \Sigma V^T$, V orthogonal (true if A is symmetric)
- The eigenvectors of A are the axes of the ellipse

$$\boxed{\begin{aligned} AV &= V\Sigma \\ AVV^T &= V\Sigma V^T \\ A &= V\Sigma V^T \end{aligned}}$$



SVD - Interpretation #3

- **In general A will also contain rotations, not just scales:**



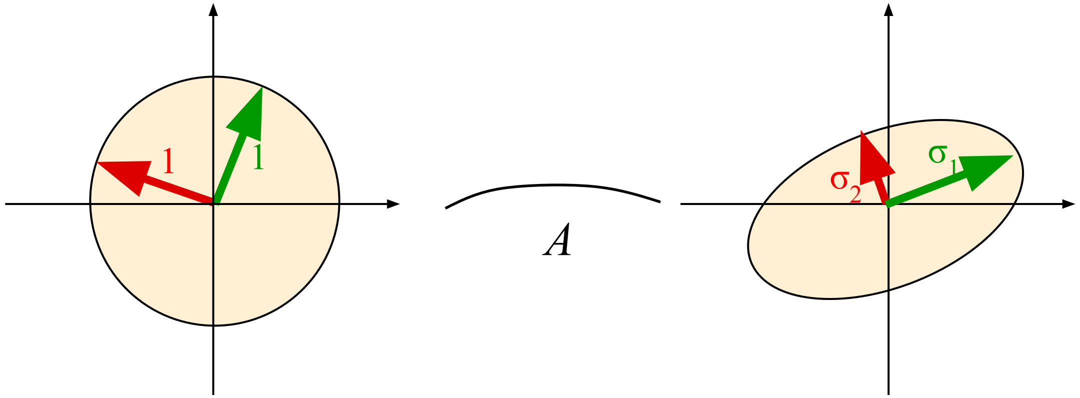
$$\boxed{AV = U\Sigma}$$

$$\underset{}{A}\overset{\textit{orthonormal}}{\begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_n\end{bmatrix}} = \overset{\textit{orthonormal}}{\begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_n\end{bmatrix}}\begin{bmatrix}\sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n\end{bmatrix}$$

$$\boxed{A\mathbf{v_i} = \sigma_i \mathbf{u_i}, \quad \sigma_i \geq 0}$$

SVD - Interpretation #3

- **In general A will also contain rotations, not just scales:**



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

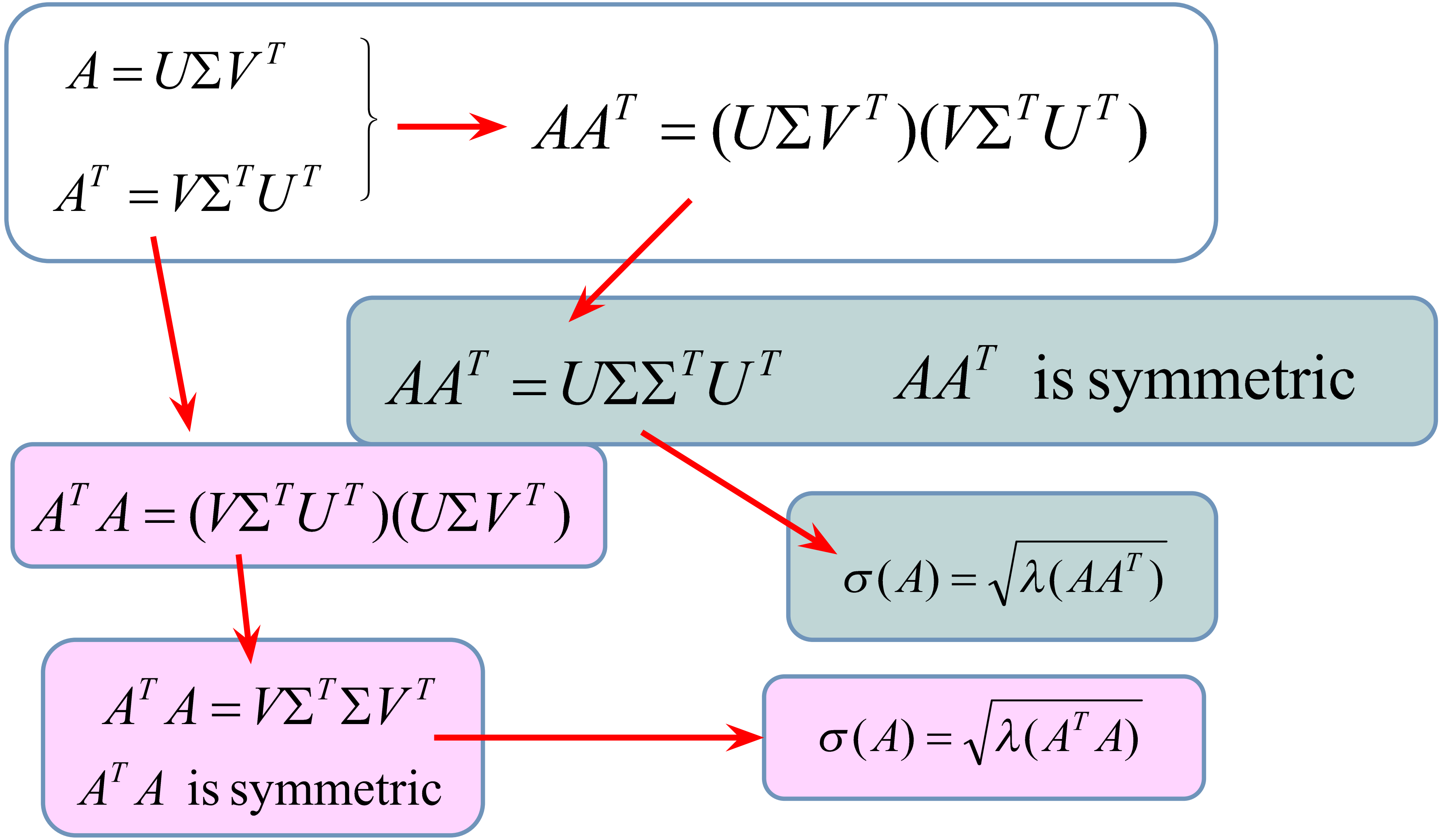
$$\boxed{A\mathbf{v}_i = \lambda_i u_i}$$

$$A = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \ldots & \mathbf{u}_n\end{bmatrix}\begin{bmatrix}\sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n\end{bmatrix}\begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2 & \ldots & \mathbf{v}_n\end{bmatrix}^T$$

Singular value decomposition (SVD)

-

Singular value decomposition (Computation)



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

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

$$AA^T = (U\Sigma V^T)(V\Sigma^T U^T)$$

$$AA^T = U\Sigma\Sigma^T U^T$$

AA^T is symmetric

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

$$\sigma(A) = \sqrt{\lambda(AA^T)}$$

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

$A^T A$ is symmetric

$$\sigma(A) = \sqrt{\lambda(A^T A)}$$

Singular value decomposition (SVD)

-

SVD Example

- $$\boldsymbol{A} = \begin{bmatrix} \mathbf{3} & \mathbf{1} & \mathbf{1} \\ \mathbf{-1} & \mathbf{3} & \mathbf{1} \end{bmatrix}$$

First, we compute the singular values σ_i by finding the eigenvalues of AA^T

$$AA^T = \begin{bmatrix} \mathbf{11} & \mathbf{1} \\ \mathbf{1} & \mathbf{11} \end{bmatrix}$$

- **Setting determinant** $\det(AA^T - \lambda) = 0$
  - $\lambda^2 - \mathbf{22}\lambda + \mathbf{120} = (\lambda - \mathbf{10})(\lambda - \mathbf{12})$ **, so the singular values are** $\sigma_1 = \sqrt{\mathbf{10}}$ **and** $\sigma_2 = \sqrt{\mathbf{12}}$**.**

SVD Example

- To calculate the eigen vector

$$Av = \lambda v \Rightarrow \begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$(11 - \lambda)x_1 + x_2 = 0$$
$$x_1 + (11 - \lambda)x_2 = 0$$

- For $\lambda = 10$

$$x_1 + x_2 = 0 \Longrightarrow x_1 = -x_2$$

- which is true for lots of values, so we'll pick $\mathrm{x}_1 = 1$ and $x_2 = -1$
- Similarly, for $\lambda = 12$, x1 = 1 and x2 = 1

SVD Example

- $$\boldsymbol{U} = \begin{bmatrix} \mathbf{1} & \mathbf{1} \\ \mathbf{1} & -\mathbf{1} \end{bmatrix}$$

- **Finally, we have to convert this matrix into an orthogonal matrix which we do by applying the** *Gram-Schmidt orthonormalization* **process to the column vectors.**

$$\boldsymbol{U} = \begin{bmatrix} \mathbf{1/\sqrt{2}} & \mathbf{1/\sqrt{2}} \\ \mathbf{1/\sqrt{2}} & -\mathbf{1/\sqrt{2}} \end{bmatrix}$$

- **Similarly,** V **can be calculated by following the same process by calculating the eigenvectors of** $A^T A$

SVD Example

- $$\boldsymbol{U} = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$
- Similarly, V can be calculated by following the same process by calculating the eigenvectors of $A^T A$

$$\boldsymbol{V} = \begin{bmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ \frac{2}{\sqrt{2}} & \frac{-1}{\sqrt{5}} & 0 \\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{30}} & \frac{-5}{\sqrt{30}} \end{bmatrix}$$

SVD Example

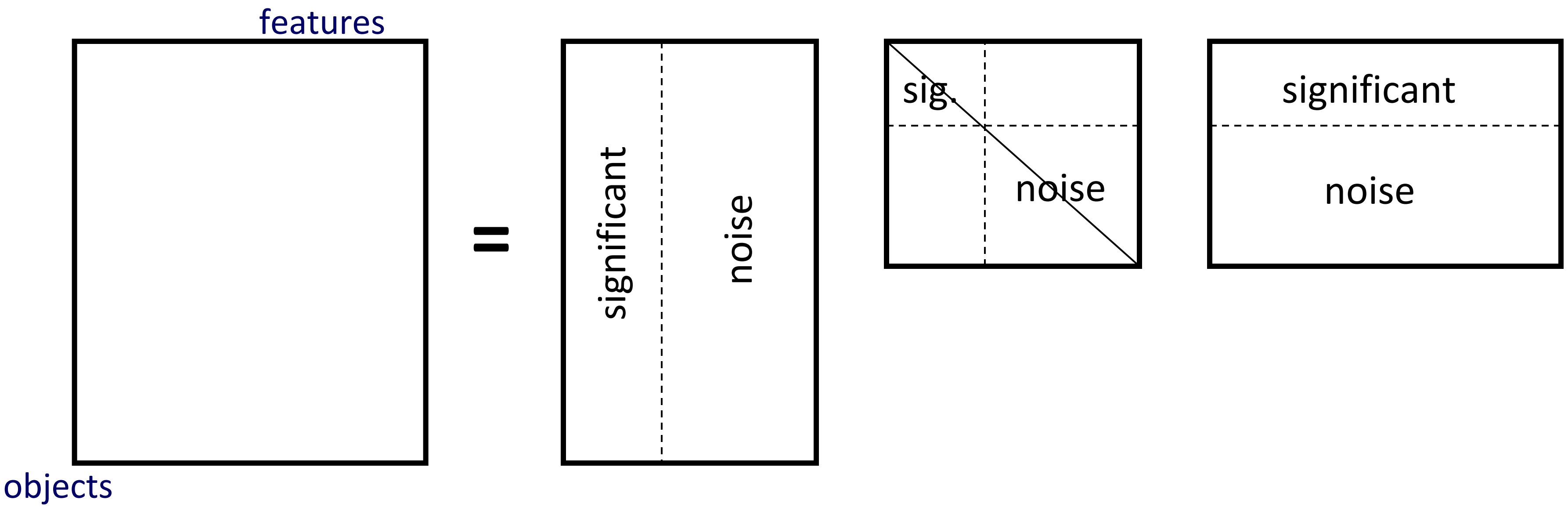
-

$$U = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} \; V = \begin{bmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ \frac{2}{\sqrt{2}} & \frac{-1}{\sqrt{5}} & 0 \\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{30}} & \frac{-5}{\sqrt{30}} \end{bmatrix} \; S = \begin{bmatrix} \sqrt{12} & 0 & 0 \\ 0 & \sqrt{10} & 0 \end{bmatrix}$$

$$A_{mn} = USV^T = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix}$$

SVD and Rank-k approximations

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



$$\underset{\textbf{n x d}}{\left(\quad A_k \quad \right)} = \underset{\textbf{n x k}}{\left(\quad U_k \quad \right)} \cdot \underset{\textbf{k x k}}{\left(\quad \Sigma_k \quad \right)} \cdot \underset{\textbf{k x d}}{\left(\quad V_k^T \quad \right)}$$

Best Rank-k approximations (A_k):

Eckart-Young theorem

- **Let A and B be any $m \times n$ matrices, with B having rank k. Then**

$$||A - A_k||_F \leq ||A - B||_F$$

- Frobenius Norm

$$||A||_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}(a_{ij})^2}$$

$$||A||_F = \sqrt{Tr(A^T A)}$$

$$||A||_F = \sqrt{\sum_{i=1}^{n}\lambda_i(A)}$$

$\lambda_i$is the ith non-zero eigenvalues of $A^*A, A^* conjugate\ transpose$

RANK 3

$$\begin{matrix}1 & 2 & 3\\ 5 & 6 & 7\\ 3 & 5 & 4\end{matrix}$$

A

RANK 3

$$\begin{matrix}0 & 2 & 1\\ 5 & 4 & 7\\ 1 & 5 & 2\end{matrix}$$

A_k

RANK 3

$$\begin{matrix}3 & 0 & 3\\ 5 & 9 & 7\\ 3 & 1 & 4\end{matrix}$$

B

$$||A - A_k||_F = \sqrt{\sum_{ij}(A_{ij} - (A_k)_{ij})^2} = \sqrt{1^2 + 0^2 + 2^2 + 0^2 + 2^2 + 0^2 + 2^2 + 0^2 + 2^2} = \sqrt{17} = 4.1231$$

$$||A - B||_F = \sqrt{\sum(A_{ij} - B_{ij})^2} = \sqrt{(-2)^2 + 2^2 + 0^2 + 0^2 + (-3)^2 + 0^2 + 0^2 + 4^2 + 0^2} = \sqrt{33} = 5.7446$$

- **Let A and B be any $m \times n$ matrices, with B having rank k. Then**

$$||A - A_k||_F \leq ||A - B||_F$$

RANK 3

$$\begin{matrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 3 & 5 & 4 \end{matrix}$$

A

RANK 3

$$\begin{matrix} 0 & 2 & 1 \\ 5 & 4 & 7 \\ 1 & 5 & 2 \end{matrix}$$

A_k

RANK 3

$$\begin{matrix} 3 & 0 & 3 \\ 5 & 9 & 7 \\ 3 & 1 & 4 \end{matrix}$$

B

$$\mathrm{A} - A_k = \begin{matrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 2 & 0 & 2 \end{matrix}$$

$$\sqrt{Tr(\,(\mathrm{A} - A_k)^T(A - A_k))} = \sqrt{5+4+8} = \sqrt{17}$$

$$= ||A - A_k||_F$$

- Frobenius Norm

$$||A||_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}(a_{ij})^2}$$

$$||A||_F = \sqrt{Tr(A^T A)}$$

$$||A||_F = \sqrt{\sum^{n} \lambda_i(A)}$$

$\lambda_i$is the ith non-zero eigenvalues of $A^*A, A^*conjugate\ transpose$

Best Rank-k approximations (A_k):
Eckart-Young theorem

- **Let A and B be any $m \times n$ matrices, with B having rank k. Then**

$$||A - A_k||_F \leq ||A - B||_F$$

RANK 3

$$\begin{matrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 3 & 5 & 4 \end{matrix}$$

A

RANK 3

$$\begin{matrix} 0 & 2 & 1 \\ 5 & 4 & 7 \\ 1 & 5 & 2 \end{matrix}$$

A_k

RANK 3

$$\begin{matrix} 3 & 0 & 3 \\ 5 & 9 & 7 \\ 3 & 1 & 4 \end{matrix}$$

B

$$\mathrm{A} - A_k = \begin{matrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 2 & 0 & 2 \end{matrix}$$

$$\sqrt{\sum_{i=1}^{n} \lambda_i((A - A_k)^*.(A - A_k))}$$

$$= \sqrt{0.3153 + 4 + 12.6847} \quad = \sqrt{17} \quad = ||A - A_k||_F$$

- Frobenius Norm

$$||A||_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{m} (a_{ij})^2}$$

$$||A||_F = \sqrt{Tr(A^T A)}$$

$$||A||_F = \sqrt{\sum_{i=1}^{n} \lambda_i(A)}$$

$\lambda_i$is the ith non-zero eigenvalues of $A^*A, A^*conjugate\ transpose$

- **Frobenius Norm**

$$||A||_F^2 = Tr(A^T A)$$

- **Let A_k is the best rank-k approximation of A**

$$\begin{aligned}
||A - A_k||_F^2 &= Tr\big((A - A_k)^T (A - A_k)\big) \\
&= Tr\big((U\Sigma V^T - U\Sigma_k V^T)^T (U\Sigma V^T - U\Sigma_k V^T)\big) \\
&= Tr\big((V\Sigma U^T - V\Sigma_k U^T)(U\Sigma V^T - U\Sigma_k V^T)\big) \\
&= Tr\big((V\Sigma - V\Sigma_k) U^T U (\Sigma V^T - \Sigma_k V^T)\big) \\
&= Tr(V(\Sigma - \Sigma_k)(\Sigma - \Sigma_k) V^T) \\
&= Tr\big(V V^T (\Sigma - \Sigma_k)(\Sigma - \Sigma_k)\big) \\
&= \sum_{i=k+1}^{n} \lambda_i(A)
\end{aligned}$$

Power method for computing the SVD

-

Power method for computing the SVD

- Let $A = \begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix}$ and $\mathrm{x}_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$
- To compute x_1, we multiply Ax_0 to get
- The Frobenius norm of the Ax_0 is $\sqrt{5^2 + 8^2} = 9.434$, then
- For the next iteration, we compute
- The Frobenius norm of the result is 6.971, so we divide to obtain
- For the next iteration, we compute
- The Frobenius norm of the result is 6.997, so we divide to obtain
- The x after convergence

Power method for computing the SVD

- Hence,

$$\lambda_1 = x^T A x = [0.447 \quad 0.894] \begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 0.447 \\ 0.894 \end{bmatrix} = 6.993$$

Determining λ_2

- To find the second eigenpair we create a new matrix

$$A^* = A \; - \; \lambda_1 x x^T$$

- Then, use power iteration on $A^*$to compute its largest eigenvalue.
- The obtained x^* and λ^*correspond to the second largest eigenvalue and the corresponding eigenvector of matrix A.

Reference

- **Feature Extraction, K. Ramachandra Murthy**
- **Singular Value Decomposition Tutorial Kirk Baker**
- **Eigen Decomposition and Singular Value Decomposition, Based on the slides by Mani Thomas Modified and extended by Longin Jan Latecki**
- **Statistical Modeling and Analysis of Neural Data (NEU 560) Princeton University, Spring 2018 Jonathan Pillow**