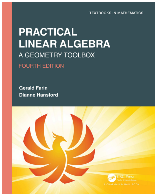
Practical Linear Algebra: A Geometry Toolbox
Fourth Edition

Chapter 16: The Singular Value Decomposition

Gerald Farin & Dianne Hansford

A K Peters/CRC Press
www.farinhansford.com/books/pla



Outline

Matrix decomposition: fundamental tool for
— understanding the action of a matrix
— establishing its suitability to solve a problem
— solving linear systems more efficiently and effectively

Symmetric matrices: *eigendecomposition*
More general matrices: the *singular value decomposition*

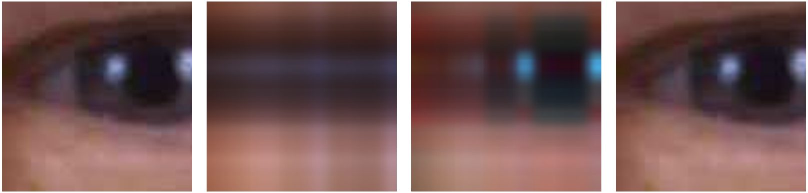


Image compression and the SVD
Original image→Highest compression→One additional term added→14 of 105 matrix terms

The Geometry of the 2×2 Case

Orthonormal vectors $\mathbf{v}_1$ and $\mathbf{v}_2 \Rightarrow$ *orthogonal matrix* $V = [\mathbf{v}_1\ \mathbf{v}_2]$
Orthonormal vectors $\mathbf{u}_1$ and $\mathbf{u}_2 \Rightarrow$ *orthogonal matrix* $U = [\mathbf{u}_1\ \mathbf{u}_2]$

Want $\mathbf{v}_i$ and $\mathbf{u}_i$ such that $A\mathbf{v}_1 = \sigma_1\mathbf{u}_1$ and $A\mathbf{v}_2 = \sigma_2\mathbf{u}_2$:

$$AV = U\Sigma \qquad \text{where} \qquad \Sigma = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}$$

The singular value decomposition (SVD) of A:

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

σ_i called the singular values of A

The Geometry of the 2×2 Case

Properties of *symmetric positive definite matrices* such as $A^{\mathrm{T}}A$:
— Real and positive eigenvalues
— Eigenvectors are orthogonal

$$\begin{aligned} A^{\mathrm{T}}A &= (U\Sigma V^{\mathrm{T}})^{\mathrm{T}}(U\Sigma V^{\mathrm{T}}) \\ &= V\Sigma^{\mathrm{T}}U^{\mathrm{T}}U\Sigma V^{\mathrm{T}} \\ &= V\Sigma^{\mathrm{T}}\Sigma V^{\mathrm{T}} \\ &= V\Lambda' V^{\mathrm{T}} \end{aligned}$$

$$\text{where} \quad \Lambda' = \begin{bmatrix} \lambda'_1 & 0 \\ 0 & \lambda'_2 \end{bmatrix} = \Sigma^{\mathrm{T}}\Sigma = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$$

This is the eigendecomposition of $A^{\mathrm{T}}A$

Columns of V called the right singular vectors of A

The Geometry of the 2 × 2 Case

Eigendecomposition of symmetric positive definite AA^{T}

$$\begin{aligned} AA^{\mathrm{T}} &= (U\Sigma V^{\mathrm{T}})(U\Sigma V^{\mathrm{T}})^{\mathrm{T}} \\ &= U\Sigma V^{\mathrm{T}} V \Sigma^{\mathrm{T}} U^{\mathrm{T}} \\ &= U\Sigma\Sigma^{\mathrm{T}} U^{\mathrm{T}} \\ &= U\Lambda' U^{\mathrm{T}} \end{aligned}$$

$$\text{where} \quad \Lambda' = \Sigma\Sigma^{\mathrm{T}} = \Sigma^{\mathrm{T}}\Sigma$$

Eigenvalues of AA^{T} are diagonal entries of Λ'

Eigenvectors of AA^{T} are columns of U

— Called the left singular vectors of A

The Geometry of the 2×2 Case

Elements of the SVD of A:

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

— The singular values

$$\sigma_i = \sqrt{\lambda_i'}$$

where λ_i' are the eigenvalues of $A^{\mathrm{T}}A$ and AA^{T}

— The columns of V are the eigenvectors of $A^{\mathrm{T}}A$

— The columns of U are the eigenvectors of AA^{T}

Once $\mathbf{v}_i$ known, compute $\mathbf{u}_i = A\mathbf{v}_i/\|\cdot\|$ since $AV = U\Sigma$

The Geometry of the 2 × 2 Case

Example: symmetric positive definite matrix that scales in $\mathbf{e}_1$-direction

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

$$AA^{\mathrm{T}} = A^{\mathrm{T}}A = \begin{bmatrix} 9 & 0 \\ 0 & 1 \end{bmatrix} \qquad \text{eigenvalues: } \lambda'_1 = 9 \quad \lambda'_2 = 1$$

$$\Rightarrow \quad \sigma_1 = 3 \quad \text{and} \quad \sigma_2 = 1$$

$$U = V = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

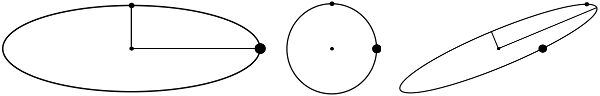
$$\text{SVD of } A : \Sigma V^{\mathrm{T}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Positive definite matrix ⇒ SVD identical to eigendecomposition

The Geometry of the 2×2 Case

Action: unit circle $\Rightarrow$ *action ellipse*

— Semi-major axis length σ_1 — Semi-minor axis length σ_2



$$\begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \qquad \text{circle} \qquad \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}$$

(previous example) (next example)

Unit circle to action ellipse mapping tracked with

Thick point: $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ Thin point: $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$

The Geometry of the 2×2 Case

Example: shear $A = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} \quad \Rightarrow \quad A^{\mathrm{T}}A = \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} \quad AA^{\mathrm{T}} = \begin{bmatrix} 5 & 2 \\ 2 & 1 \end{bmatrix}$

Eigenvalues: $\lambda_1' = 5.82$ and $\lambda_2' = 0.17 \quad \Rightarrow \quad \sigma_1 = 2.41$ and $\sigma_2 = 0.41$

$$\text{Eigenvectors of } A^{\mathrm{T}}A: \quad V = \begin{bmatrix} 0.38 & -0.92 \\ 0.92 & 0.38 \end{bmatrix}$$

$$\text{Eigenvectors of } AA^{\mathrm{T}}: \quad U = \begin{bmatrix} 0.92 & -0.38 \\ 0.38 & 0.92 \end{bmatrix}$$

$$\text{SVD of } A: \quad \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.92 & -0.38 \\ 0.38 & 0.92 \end{bmatrix} \begin{bmatrix} 2.41 & 0 \\ 0 & 0.41 \end{bmatrix} \begin{bmatrix} 0.38 & 0.92 \\ -0.92 & 0.38 \end{bmatrix}$$

The Geometry of the 2×2 Case

Break down the action of A in terms of the SVD

$$\begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.92 & -0.38 \\ 0.38 & 0.92 \end{bmatrix} \begin{bmatrix} 2.41 & 0 \\ 0 & 0.41 \end{bmatrix} \begin{bmatrix} 0.38 & 0.92 \\ -0.92 & 0.38 \end{bmatrix}$$

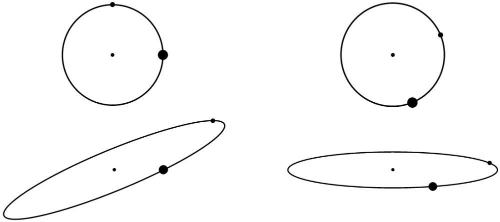
Clockwise from top left:

— Initial point set forming a circle with two reference points

— $V^{\mathrm{T}}\mathbf{x}$ rotates clockwise 67.5°

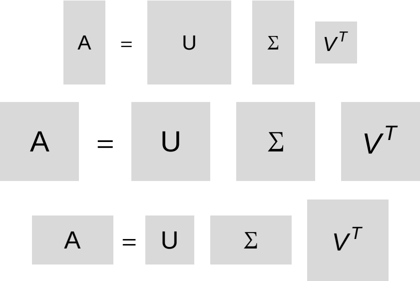
— $\Sigma V^{\mathrm{T}}\mathbf{x}$ stretches in $\mathbf{e}_1$ and shrinks in $\mathbf{e}_2$

— $U\Sigma V^{\mathrm{T}}\mathbf{x}$ rotates counterclockwise 22.5°



The General Case

Now: $m \times n$ matrix A — not necessarily square nor invertible



Top: $m > n$ Middle: $m = n$ Bottom: $m < n$

U is $m \times m$ Σ is $m \times n$ V^{T} is $n \times n$

$A^{\mathrm{T}}A = V\Lambda' V^{\mathrm{T}} \Rightarrow \Lambda'$ is $n \times n$ $AA^{\mathrm{T}} = U\Lambda' U^{\mathrm{T}} \Rightarrow \Lambda'$ is $m \times m$

Both Λ' hold the same non-zero eigenvalues $\Rightarrow$ rank $\leq \min\{m, n\}$

The General Case

Want $\mathbf{v}_i$ and $\mathbf{u}_i$ such that $A\mathbf{v}_i = \sigma_i \mathbf{u}_i$

$$AV = U\Sigma$$

Rank r of A plays a role in the SVD

Main properties:

1. Σ has non-zero singular values $\sigma_1, \ldots, \sigma_r$ and all other entries zero
2. First r columns of U form an orthonormal basis for column space of A
3. Last $m - r$ columns of U form an orthonormal basis for null space of A^{T}
4. First r columns of V form an orthonormal basis for row space of A (column space of A^{T})
5. Last $n - r$ columns of V form an orthonormal basis for null space of A

Items 2 – 5 identify the four fundamental subspaces

The General Case

Example: Rank 2 matrix: $A = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{bmatrix}$

$$A^{\mathrm{T}}A = \begin{bmatrix} 1 & 0 \\ 0 & 5 \end{bmatrix} \quad \begin{matrix} \lambda'_1 = 5 \\ \lambda'_2 = 1 \end{matrix} \quad V = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

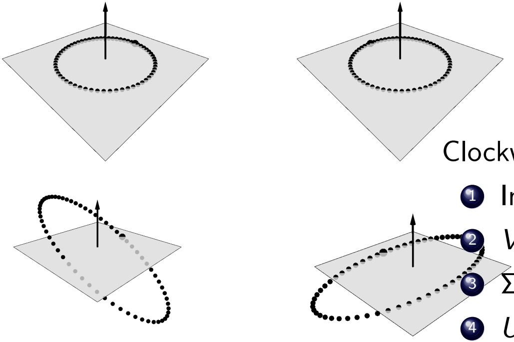
$$AA^{\mathrm{T}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 2 \\ 0 & 2 & 1 \end{bmatrix} \quad \begin{matrix} \lambda'_1 = 5 \\ \lambda'_2 = 1 \\ \lambda'_3 = 0 \end{matrix} \quad U = \begin{bmatrix} 0 & 1 & 0 \\ 0.89 & 0 & -0.44 \\ 0.44 & 0 & 0.89 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 2.23 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$A = U\Sigma V^{\mathrm{T}}: \quad \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0.89 & 0 & -0.44 \\ 0.44 & 0 & 0.89 \end{bmatrix} \begin{bmatrix} 2.23 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$m - r = 1 \Rightarrow \mathbf{u}_3$ is in the null space of $A^{\mathrm{T}} \Rightarrow A^{\mathrm{T}}\mathbf{u}_3 = \mathbf{0}$

The General Case

SVD and action of a matrix



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

Clockwise from top left:

1. Initial circle point set
2. $V^{\mathrm{T}}\mathbf{x}$ reflects
3. $\Sigma V^{\mathrm{T}}\mathbf{x}$ stretches in $\mathbf{e}_1$
4. $U\Sigma V^{\mathrm{T}}\mathbf{x}$ rotates

The General Case

$$\textbf{Example:} A = \begin{bmatrix} -0.8 & 0 & 0.8 \\ 1 & 1.5 & -0.3 \end{bmatrix}$$

$$A^{\mathrm{T}}A = \begin{bmatrix} 1.64 & 1.5 & -0.94 \\ 1.5 & 2.25 & -0.45 \\ -0.94 & -0.45 & 0.73 \end{bmatrix} \quad \begin{matrix} \lambda_1' = 3.77 \\ \lambda_2' = 0.84 \\ \lambda_3' = 0 \end{matrix} \quad V = \begin{bmatrix} -0.63 & 0.38 & 0.67 \\ -0.71 & -0.62 & -0.31 \\ 0.30 & -0.68 & 0.67 \end{bmatrix}$$

$$AA^{\mathrm{T}} = \begin{bmatrix} 1.28 & -1.04 \\ -1.04 & 3.34 \end{bmatrix} \quad \begin{matrix} \lambda_1' = 3.77 \\ \lambda_2' = 0.84 \end{matrix} \quad U = \begin{bmatrix} 0.39 & -0.92 \\ -0.92 & -0.39 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 1.94 & 0 & 0 \\ 0 & 0.92 & 0 \end{bmatrix}$$

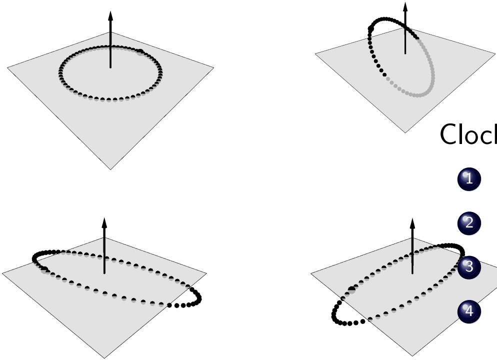
SVD: $A = U\Sigma V^{\mathrm{T}}$

$$\begin{bmatrix} -0.8 & 0 & 0.8 \\ 1 & 1.5 & -0.3. \end{bmatrix} = \begin{bmatrix} 0.39 & -0.92 \\ -0.92 & -0.39 \end{bmatrix} \begin{bmatrix} 1.94 & 0 & 0 \\ 0 & 0.92 & 0 \end{bmatrix} \begin{bmatrix} -0.63 & -0.71 & 0.3 \\ 0.38 & -0.62 & -0.68 \\ 0.67 & -0.31 & 0.67 \end{bmatrix}$$

$n - r = 1 \;\Rightarrow\; \mathbf{v}_3$ in null space of $A \;\Rightarrow\; A\mathbf{v}_3 = \mathbf{0}$

The General Case

SVD and action of a matrix



$$A = \begin{bmatrix} -0.8 & 0 & 0.8 \\ 1 & 1.5 & -0.3 \end{bmatrix}$$

Clockwise from top left:

1. Initial circle point set
2. $V^{\mathrm{T}}\mathbf{x}$
3. $\Sigma V^{\mathrm{T}}\mathbf{x}$
4. $U\Sigma V^{\mathrm{T}}\mathbf{x}$

The General Case

Example: a projection into the $[\mathbf{e}_1, \mathbf{e}_2]$-plane — a rank deficient matrix

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

A is symmetric and idempotent $\quad \Rightarrow \quad A = A^{\mathrm{T}}A = AA^{\mathrm{T}}$

$$A = U\Sigma V^{\mathrm{T}}: \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rank $= 2$
$\Rightarrow$ first 2 columns of U form orthonormal basis for column space of A
$\Rightarrow$ first 2 columns of V form orthonormal basis for row space of A
$\mathbf{e}_3$ vector projected to the zero vector $\Rightarrow$ spans the null space of A and A^{T}

SVD Steps

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

Here: review steps — for a robust algorithm $\Rightarrow$ advanced numerical methods

Input: $m \times n$ matrix A
Output: U, V, Σ such that $A = U\Sigma V^{\mathrm{T}}$

1. Find the *eigenvalues* $\lambda'_1, \ldots, \lambda'_n$ of $A^{\mathrm{T}}A$
   - Order the λ'_i so that $\lambda'_1 \geq \lambda'_2 \geq \ldots \geq \lambda'_n$
   - Suppose $\lambda'_1, \ldots, \lambda'_r > 0$, then the *rank* of A is r
2. Create an $m \times n$ diagonal matrix Σ with $\sigma_{i,i} = \sqrt{\lambda'_i}, i = 1, \ldots, r$
3. Find the corresponding (normalized) eigenvectors $\mathbf{v}_i$ of $A^{\mathrm{T}}A$
4. Create an $n \times n$ matrix V with column vectors $\mathbf{v}_i$
5. Find the (normalized) eigenvectors $\mathbf{u}_i$ of AA^{T}
6. Create an $m \times m$ matrix U with column vectors $\mathbf{u}_i$

SVD Steps

Notes on steps:

- Can compute $\mathbf{u}_i, i = 1, r$ as $\mathbf{u}_i = A\mathbf{v}_i / \| \cdot \|$

  If $m > n$ then the remaining $\mathbf{u}_i$ are found from the null space of A^{T}

- The only "hard" task is finding the λ_i'

  Since $A^{\mathrm{T}}A$ is symmetric $\Rightarrow$ Can choose a highly efficient algorithm

- Forming $A^{\mathrm{T}}A$ can result in an ill-posed problem: $\kappa(A^{\mathrm{T}}A) = \kappa(A)^2$

  Avoid direct computation of this matrix

  — employ the Householder method

Singular Values and Volumes

Determinant:

$$\det U = \pm 1 \quad \text{and} \quad \det V = \pm 1 \quad \Rightarrow \quad |\det A| = \det \Sigma = \sigma_1 \cdot \ldots \cdot \sigma_n$$

Application: given a 2D triangle T with area φ
Transform $T \to T'$ with 2D linear map with singular values σ_1, σ_2
Area of $T' = \pm\sigma_1\sigma_2\varphi$

Application: given a 3D object O with volume φ
Transform $O \to O'$ with 3D linear map with singular values $\sigma_1, \sigma_2, \sigma_3$
Volume of $O' = \pm\sigma_1\sigma_2\sigma_3\varphi$

Recall determinants without using singular values

$$\det A = \lambda_1 \cdot \ldots \cdot \lambda_n$$

The Pseudoinverse

The inverse of a matrix:
— Limited to square, nonsingular matrices
— Mainly a theoretical tool for analyzing the solution to a linear system

The generalized inverse or psuedoinverse $A^\dagger$
— For general matrices
— Suited for practical use
— Can be computed with the SVD

Special case: $m \times n$ diagonal matrix Σ with diagonal elements σ_i

$$\text{Pseudoinverse: } n \times m \;\; \Sigma^\dagger \;\; \text{with} \sigma_i^\dagger = \begin{Bmatrix} 1/\sigma_i & \text{if} \;\; \sigma_i > 0 \\ 0 & \text{else} \end{Bmatrix}$$

If $\text{rank}(\Sigma) = r$ then
— $\Sigma^\dagger\Sigma$ holds the $r \times r$ identity matrix
— All other elements are zero

The Pseudoinverse

Leads to the pseudoinverse for a general $m \times n$ matrix A

$$A^\dagger = (U\Sigma V^{\mathrm{T}})^{-1} = V\Sigma^\dagger U^{\mathrm{T}}$$

If A is square and invertible then $A^\dagger = A^{-1}$

Properties:

$$A^\dagger A A^\dagger = A^\dagger \qquad \text{and} \qquad A A^\dagger A = A$$

Often times called the Moore-Penrose generalized inverse

Primary application: *least squares approximation*

Example: Find the pseudoinverse of

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0.89 & 0 & -0.44 \\ 0.44 & 0 & 0.89 \end{bmatrix} \begin{bmatrix} 2.23 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$\Sigma^{\dagger} = \begin{bmatrix} 1/2.23 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$A^{\dagger} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2/5 & 1/5 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1/2.23 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0.89 & 0.44 \\ 1 & 0 & 0 \\ 0 & -0.44 & 0.89 \end{bmatrix}$$

Example: square and nonsingular A

$$A = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \qquad \text{and} \qquad A^{-1} = \begin{bmatrix} 1/3 & 0 \\ 0 & 1 \end{bmatrix}$$

The pseudoinverse is equal to the inverse:

$$A^{\dagger} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1/3 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1/3 & 0 \\ 0 & 1 \end{bmatrix}$$

Least Squares

Overdetermined linear system: m equations in n unknowns where $m \geq n$

$$A\mathbf{x} = \mathbf{b}$$

Linear system is inconsistent
— unlikely that $\mathbf{b}$ lives in subspace $\mathcal{V}$ defined by columns of A

The least squares solution finds the orthogonal projection of $\mathbf{b}$ into $\mathcal{V}$
— Call this projection $\mathbf{b}'$
$\Rightarrow$ Solution to $A\mathbf{x} = \mathbf{b}'$ produces vector closest to $\mathbf{b}$ that lives in $\mathcal{V}$

Normal equations

$$A^{\mathrm{T}}A\mathbf{x} = A^{\mathrm{T}}\mathbf{b} \quad \text{solution minimizes} \quad \|A\mathbf{x} - \mathbf{b}\|$$

This system can be ill-posed $\quad\Rightarrow\quad$ use *pseudoinverse*

$$\mathbf{x} = A^{\dagger}\mathbf{b}$$

Least Squares

Why is $\mathbf{x} = A^{\dagger}\mathbf{b}$ the least squares solution?

Find $\mathbf{x}$ to minimize $\|A\mathbf{x} - \mathbf{b}\|$

$$\begin{aligned} A\mathbf{x} - \mathbf{b} &= U\Sigma V^{\mathrm{T}}\mathbf{x} - \mathbf{b} \\ &= U\Sigma V^{\mathrm{T}}\mathbf{x} - UU^{\mathrm{T}}\mathbf{b} \\ &= U(\Sigma\mathbf{y} - \mathbf{z}) \end{aligned}$$

This new framing of the problem exposes that

$$\|A\mathbf{x} - \mathbf{b}\| = \|\Sigma\mathbf{y} - \mathbf{z}\|$$

$\Rightarrow$ an easier diagonal least squares problem to solve

Least Squares

Steps:

1. Compute the SVD $A = U\Sigma V^{\mathrm{T}}$
2. Compute the $m \times 1$ vector $\mathbf{z} = U^{\mathrm{T}}\mathbf{b}$
3. Compute the $n \times 1$ vector $\mathbf{y} = \Sigma^{\dagger}\mathbf{z}$
   — Least squares solution to $m \times n$ problem $\Sigma\mathbf{y} = \mathbf{z}$

requires minimizing
$\mathbf{v} = \Sigma\mathbf{y} - \mathbf{z}$

$\mathrm{rank}(\Sigma) = r$

$$\mathbf{v} = \begin{bmatrix} \sigma_1 y_1 - z_1 \\ \sigma_2 y_2 - z_2 \\ \vdots \\ \sigma_r y_r - z_r \\ -z_{r+1} \\ \vdots \\ -z_m \end{bmatrix}$$

$\mathbf{y}$ minimizing $\mathbf{v}$: $y_i = z_i/\sigma_i \quad i = 1, \ldots, r \quad \Rightarrow \quad \mathbf{y} = \Sigma^{\dagger}\mathbf{z}$

4. Compute the $n \times 1$ solution vector $\mathbf{x} = V\mathbf{y}$

Summarize — The calculations in reverse order include

$$\mathbf{x} = V\mathbf{y}$$
$$\mathbf{x} = V(\Sigma^{\dagger}\mathbf{z})$$
$$\mathbf{x} = V\Sigma^{\dagger}(U^{\mathrm{T}}\mathbf{b})$$

Example: Revisit temperature-time data: find the best fit line coefficients — Chapter 12 (normal equations) and Chapter 13 (Householder)

$$\begin{bmatrix} 0 & 1 \\ 10 & 1 \\ 20 & 1 \\ 30 & 1 \\ 40 & 1 \\ 50 & 1 \\ 60 & 1 \end{bmatrix} \mathbf{x} = \begin{bmatrix} 30 \\ 25 \\ 40 \\ 40 \\ 30 \\ 5 \\ 25 \end{bmatrix}$$

Step 1) Compute the SVD $A = U\Sigma V^{\mathrm{T}}$

$$\Sigma = \begin{bmatrix} 95.42 & 0 \\ 0 & 1.47 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad \Sigma^{\dagger} = \begin{bmatrix} 0.01 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.68 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

U 7×7
Σ 7×2
V 2×2

Step 2) $\mathbf{z} = U^{\mathrm{T}}\mathbf{b} = \begin{bmatrix} 54.5 \\ 51.1 \\ 3.2 \\ -15.6 \\ 9.6 \\ 15.2 \\ 10.8 \end{bmatrix}$

Step 3) $\mathbf{y} = \Sigma^{\dagger}\mathbf{z} = \begin{bmatrix} 0.57 \\ 34.8 \end{bmatrix}$

Step 4) $\mathbf{x} = V\mathbf{y} = \begin{bmatrix} -0.23 \\ 34.8 \end{bmatrix}$ $\Rightarrow$ best fit line: $x_2 = -0.23x_1 + 34.8$

Least Squares

The normal equations give a best approximation

$$\mathbf{x} = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}\mathbf{b} \quad \text{to the original problem} \quad A\mathbf{x} = \mathbf{b}$$

by considering $\mathbf{b}'$ in the subspace of A called $\mathcal{V}$

Substitute this expression for $\mathbf{x}$ into $A\mathbf{x} = \mathbf{b}'$:

$$\mathbf{b}' = A(A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}\mathbf{b} = AA^{\dagger}\mathbf{b} = \mathrm{proj}_{\mathcal{V}}\mathbf{b}$$

Goal is to project $\mathbf{b}$ into $\mathcal{V}$ $\Rightarrow$ $AA^{\dagger}$ is a projection

Property $A^{\dagger}AA^{\dagger} = A^{\dagger}$ ensures necessary idempotent property

Application: Image Compression

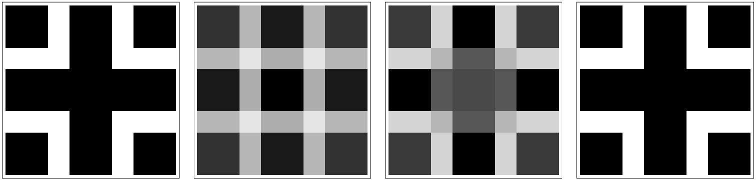
Given: $m \times n$ matrix A

with $k = \min(m, n)$ singular values σ_i where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k$

Find: the SVD of A and write as a sum of k rank one matrices:

$$A = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^{\mathrm{T}} + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^{\mathrm{T}} + \ldots + \sigma_k \mathbf{u}_k \mathbf{v}_k^{\mathrm{T}}$$

— An image has the same structure as a matrix
— Interpret matrix elements brightness values (gray scales)
— Figure (left): input image with 8×8 pixels



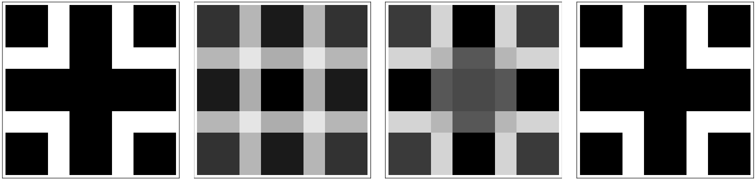
Application: Image Compression

Singular values for this matrix are $\sigma_i = 6.2, \quad 1.7, \quad 1.49, \quad 0, \ldots, 0$
Images from left to right $A, \quad A_1, \quad A_2, \quad A_3$ (original image is A)

$$A_1 = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^{\mathrm{T}} \quad \Rightarrow \quad \text{best rank one approximation to } A$$

$$A_2 = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^{\mathrm{T}} + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^{\mathrm{T}} \quad \text{best rank two approximation to } A$$

Original image is nearly replicated already
— Due to the fact that $\sigma_4 = \ldots = \sigma_8 = 0$
—Last image is $A_3 = A_2 + \sigma_3 \mathbf{u}_3 \mathbf{v}_3^{\mathrm{T}} = A$ since the remaining singular values are zero



Application: Image Compression

The change in an image by adding the smallest σ_i can be hardly noticeable
$\Rightarrow$ Omitting images corresponding to small σ_k amounts to compressing the original image

Chapter introduction Figure created from a 105×105 color image
— 3 matrices (red, green, blue)

$$\begin{aligned}
\text{red:} \quad & \sigma_i = 47.9, 5.9, 3.9, 1.7, \ldots \\
\text{green:} \quad & \sigma_i = 33.9, 4.8, 3.6, 1.6, \ldots \\
\text{blue:} \quad & \sigma_i = 32.7, 6.1, 3.2, 2.0, \ldots .
\end{aligned}$$

In Figure: $A, \quad A_1, \quad A_2, \quad A_{14}$

Application: Image Compression

Quantify each matrix term's contribution using the *Frobenius norm*

Define the *energy* or *fraction of power* of each matrix term as

$$E_i = \frac{||\sigma_i \mathbf{u}_i \mathbf{v}_i^{\mathrm{T}}||_F^2}{||A||_F^2} = \frac{\sigma_i^2}{\sigma_1^2 + \ldots + \sigma_k^2} \qquad \text{for } i = 1, \ldots, k$$

Cumulative energy for an image can be defined as

$$C_i = E_1 + \ldots + E_i$$

For *red component* in the introduction to chapter Figure:

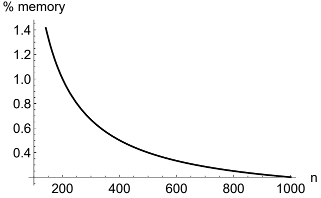
$$C_1 = 0.9752 \quad C_2 = 0.9901 \quad C_{14} = 0.9999$$

$\Rightarrow$ first two matrix terms constitute a good approximation to the original image

Application: Image Compression

Progressively sending matrix term data
— Send sets of $\sigma_i, \mathbf{u}_i, \mathbf{v}_i$, one at a time then progressively rebuilding the image on the client side

Example: 1000×1000 image
— A rank 1 approximation is represented by 2001 values
— 0.2% of the 1 million brightness values for one color component



SVD matrix term memory: for an $n \times n$ image, the percentage of memory that one matrix term requires is plotted

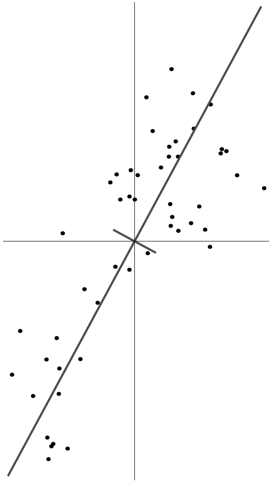
Scatter plot: data pairs recorded in Cartesian coordinates

Each circle represents a coordinate pair (point) in the $[\mathbf{e}_1, \mathbf{e}_2]$-system

Example: Gross domestic product and poverty rate pairs

How might we reveal trends in this data set?

Principal Component Analysis



Given: 2D data set $\mathbf{x}_1, \ldots, \mathbf{x}_n$
such that $\mathbf{x}_1 + \ldots + \mathbf{x}_n = \mathbf{0}$

Let $\mathbf{d}$ be a unit vector
Project $\mathbf{x}_i$ onto line containing $\mathbf{d}$
— Squared length $(\mathbf{x}_i \cdot \mathbf{d})^2$

$$I(\mathbf{d}) = (\mathbf{x}_1 \cdot \mathbf{d})^2 + \ldots + (\mathbf{x}_n \cdot \mathbf{d})^2$$

Rotate $\mathbf{d}$ around the origin
— Compute $I(\mathbf{d})$

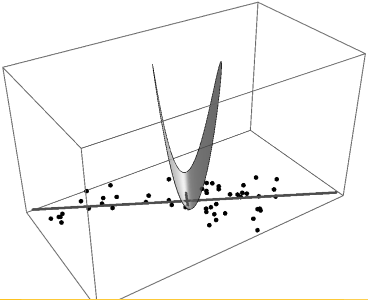
Directions corresponding to min and max $I(\mathbf{d})$ are orthogonal
$\Rightarrow$ indicates variation in data

Principal Component Analysis

Arrange data $\mathbf{x}_i$ in a data matrix and formulate problem in matrix form

$$X = \begin{bmatrix} \mathbf{x}_1^{\mathrm{T}} \\ \mathbf{x}_2^{\mathrm{T}} \\ \vdots \\ \mathbf{x}_n^{\mathrm{T}} \end{bmatrix} \quad \text{then} \quad I(\mathbf{d}) = \|X\mathbf{d}\|^2 = (X\mathbf{d}) \cdot (X\mathbf{d}) = \mathbf{d}^{\mathrm{T}} X^{\mathrm{T}} X \mathbf{d} \quad (*)$$

Let $C = X^{\mathrm{T}} X$ then C is a symmetric positive definite 2×2 matrix
(*) is a *quadratic form*



Principal Component Analysis

For which $\mathbf{d}$ is $I(\mathbf{d})$ maximal?

Answer: $\mathbf{d}$ that corresponds to C's dominant eigenvector

$$\mathbf{d}^{\mathrm{T}} C \mathbf{d} = \mathbf{d}^{\mathrm{T}} \lambda \mathbf{d}$$

$I(\mathbf{d})$ is minimal for $\mathbf{d}$ being the eigenvector corresponding to C's smallest eigenvalue

These eigenvectors form the major and minor axis of the *action ellipse*

— C symmetric $\Rightarrow$ Eigenvectors orthogonal

Look more closely at C

$$\begin{aligned} c_{1,1} &= x_{1,1}^2 + x_{2,1}^2 + \ldots + x_{n,1}^2 \\ c_{1,2} &= c_{2,1} = x_{1,1}x_{1,2} + x_{2,1}x_{2,2} + \ldots + x_{n,1}x_{n,2} \\ c_{2,2} &= x_{1,2}^2 + x_{2,2}^2 + \ldots + x_{n,2}^2. \end{aligned}$$

If each element of C is divided by n it is called the covariance matrix

— Summary of the variation in each coordinate and between coordinates

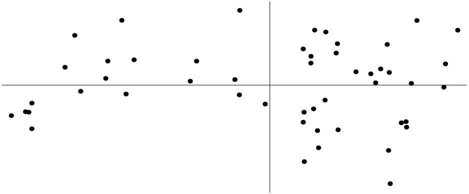
— Dividing by n will result in scaled eigenvalues (eigenvectors will not change)

Principal Component Analysis

Eigenvectors provide a convenient *local coordinate frame* for the data set
— Idea behind the principle of the *eigendecomposition*
— This frame is commonly called the principal axes

Let $V = [\mathbf{v}_1 \ \mathbf{v}_2]$ hold the normalized eigenvectors as column vectors
— $\mathbf{v}_1$ is the dominant eigenvector
Orthogonal transformation of the data X
— aligns $\mathbf{v}_1$ with $\mathbf{e}_1$ and $\mathbf{v}_2$ with $\mathbf{e}_2$

$$\hat{X} = XV \quad \Rightarrow \quad \hat{\mathbf{x}}_i = \begin{bmatrix} \mathbf{x}_i \cdot \mathbf{v}_1 \\ \mathbf{x}_i \cdot \mathbf{v}_2 \end{bmatrix}$$



Principal Component Analysis

Summary:

• Established a principal components coordinate system

— Defined by the eigenvectors of the covariance matrix

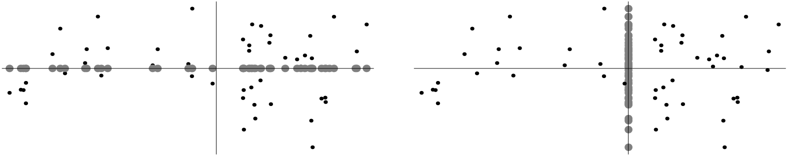
— Greatest variance corresponds to the first coordinate

• Data coordinates are now in terms of the trend lines

— Coordinates directly measure the distance from each trend line

⇒ Name of this method: Principal Component Analysis (PCA)

Principal Component Analysis

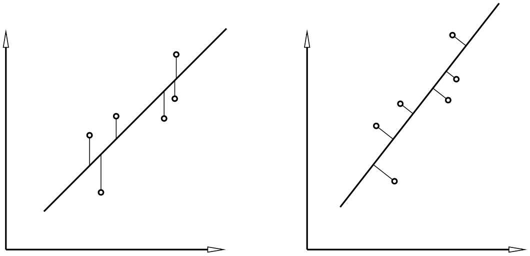


PCA can also be used for *data compression* by reducing dimensionality

Let V hold only some eigenvectors
— Example: most significant then $V = \mathbf{v}_1$ (left Figure)
— Example: $V = \mathbf{v}_2$ (right Figure)

Greater spread of the data corresponds to higher variance

Best Fit Line

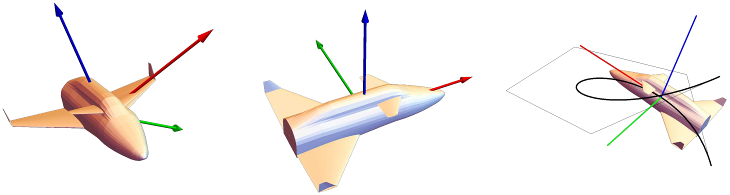


Left: least squares best fit line minimizes the sum of the vertical distances

Right: PCA best fit line minimizes the sum of the perpendicular distances to the dominant line

Principal Component Analysis

3D axis alignment: PCA can be used to align a 3D object with the coordinate axes in preparation for animation



Left: input space shuttle model

Middle: PCA axis aligned model

Right: moving the shuttle through space

Principal Component Analysis

Here we demonstrated PCA for 2D and 3D data but the real power of PCA comes with higher dimensional data

— Helpful when it is difficult to visualize and understand relationships between dimensions

— Possible to identify insignificant dimensions and eliminate them

Application: Face Authentication

Suppose a company has installed a face authentication system for access to the building

Each of the 1000 employees has their picture taken,

— 200×200 pixel image I_i for $i = 1, \ldots, 1000$

When a person arrives at the building's door, their picture I is taken and it must be checked if it is in the employee database

How can this be done quickly?

Application: Face Authentication

Convert each image I_i column-by-column to a vector $\mathbf{f}_i$ with 40,000 entries
— All employee images live in a 40,000-dimensional face vector space $\mathcal{F}$

A person arrives at the door:
— a new image I is taken
— converted to a vector $\mathbf{f}$
— A comparison against the faces in the database

$$||\mathbf{f}_i - \mathbf{f}|| < \epsilon \qquad \text{for } i = 1, \ldots, 1000$$

(ϵ is a pre-determined tolerance)

High dimensionality of $\mathcal{F}$ makes this an expensive operation

Faster approach: eigenfaces

Reduce the number of defining features for each face from 40,000 to something more manageable

— Let's select the first 100 eigenvectors $\mathbf{v}_j$ corresponding to the largest eigenvalues

— Each eigenvector has 40,0000 components and it can be reorganized into a (ghostly) image

Application: Face Authentication

Each face $\mathbf{f}_i$ in the database can be approximated by a linear combination of the eigenfaces in vector form $\mathbf{v}_j$

$$\mathbf{f}_i \approx s_{i,1}\mathbf{v}_1 + \ldots + s_{i,100}\mathbf{v}_{100}$$

$\Rightarrow$ 40,000 $\times$ 100 overdetermined linear system
— same procedure is applied to the new face $\mathbf{f}$ (resulting in $\mathbf{s}$)

Check whether new face is close to a face in the database of faces

$$||\mathbf{s}_i - \mathbf{s}|| < \tau, \qquad \text{for } i = 1, \ldots, 1000,$$

(τ predetermined tolerance)

Computational cost comparison:
1000 vectors that are 100-dimensional versus
1000 vectors that are 40,000-dimensional

WYSK

- Singular Value Decomposition (SVD)
- singular values
- right singular vector
- left singular vector
- SVD matrix dimensions
- SVD column, row, and null spaces
- SVD steps
- volume in terms of singular values
- eigendecomposition
- matrix decomposition
- action ellipse axes length
- pseudoinverse
- generalized inverse
- least squares solution via the pseudoinverse
- quadratic form
- contour ellipse
- Principal Components Analysis (PCA)
- covariance matrix