Machine Learning: Chapter 10 Summary

Dimensionality Reduction I: Principal Componoent Analysis & Singular Value Decomposition

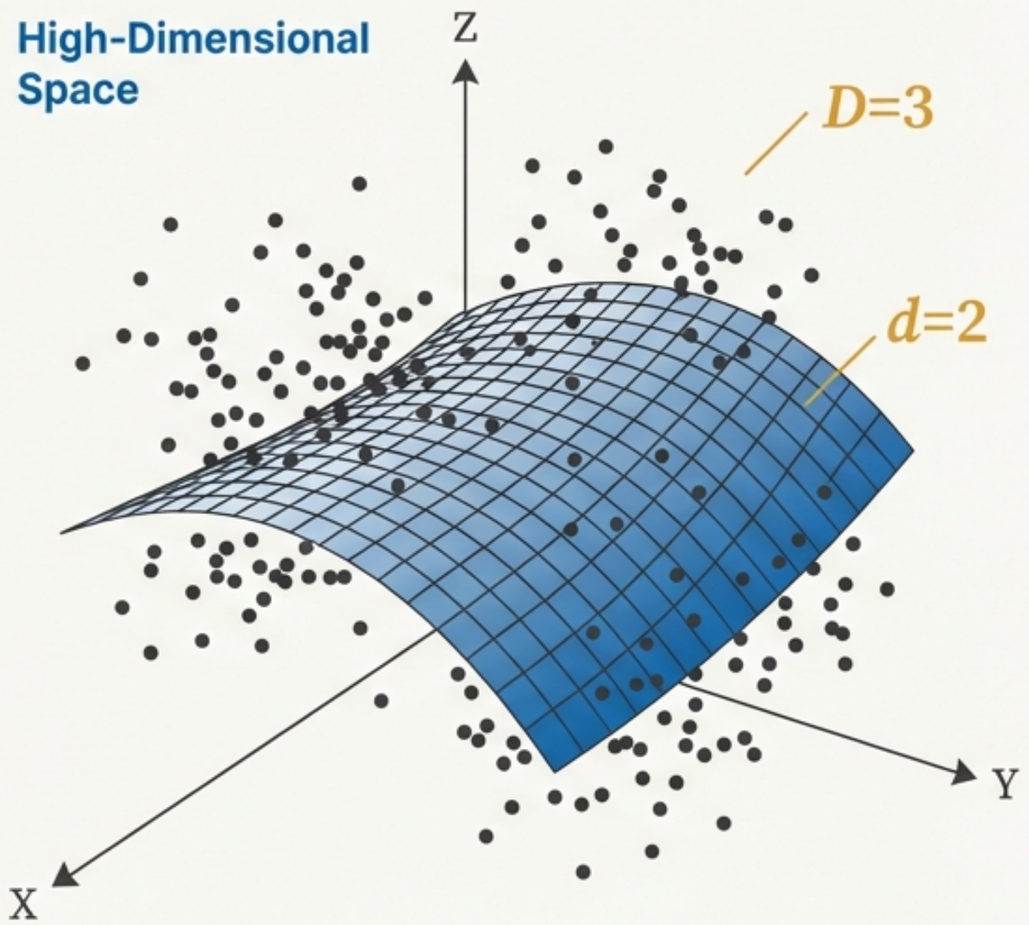
IN2064 | Technical University of Munich

The Challenge of High-Dimensional Data

Key Concept

Many datasets have a large number of features, which presents several challenges:

- **The Curse of Dimensionality**
  We need an exponential amount of data to characterize a space as its dimensionality increases.
- **Computational Expense**
  Distance and similarity calculations become very costly in high dimensions.
- **Visualization**
  It is difficult to visualize and interpret data beyond three dimensions.



Our Goal

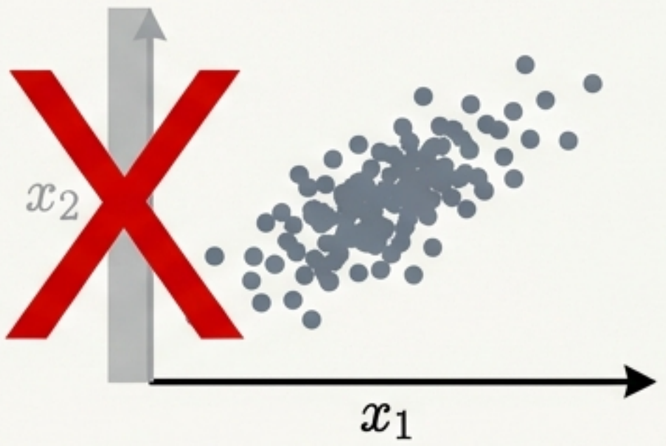
Reduce the number of dimensions while preserving as much meaningful information as possible to save computation, reveal the data's intrinsic structure, and enable visualization.

Our Strategy: Finding a Better Coordinate System

Two Approaches to Dimensionality Reduction:

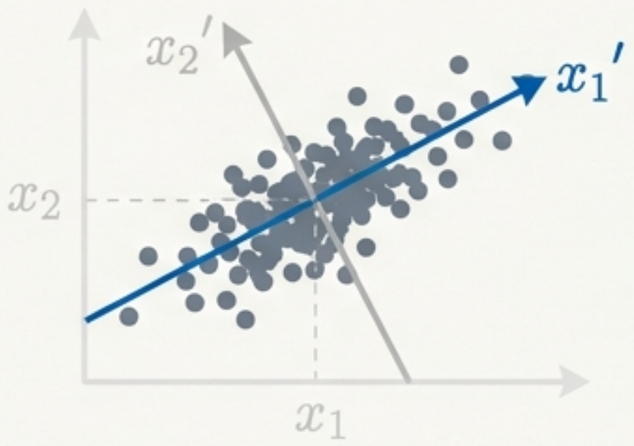
Feature Selection

Simply discarding entire features. This is simple but ignores correlations between features.



Linear Transformation

A more powerful approach that rotates the coordinate system to better align with the data's structure, then discards the new, less informative dimensions.



The Mathematics of Transformation

We use an orthonormal transformation matrix F to project the original data matrix X into a new coordinate system X'.

Transformation of Data: $X' = X \cdot F$ Transformation of Covariance: $\Sigma_{X'} = F^T \cdot \Sigma_X \cdot F$

The key question this chapter answers is: How do we find the optimal transformation matrix F?

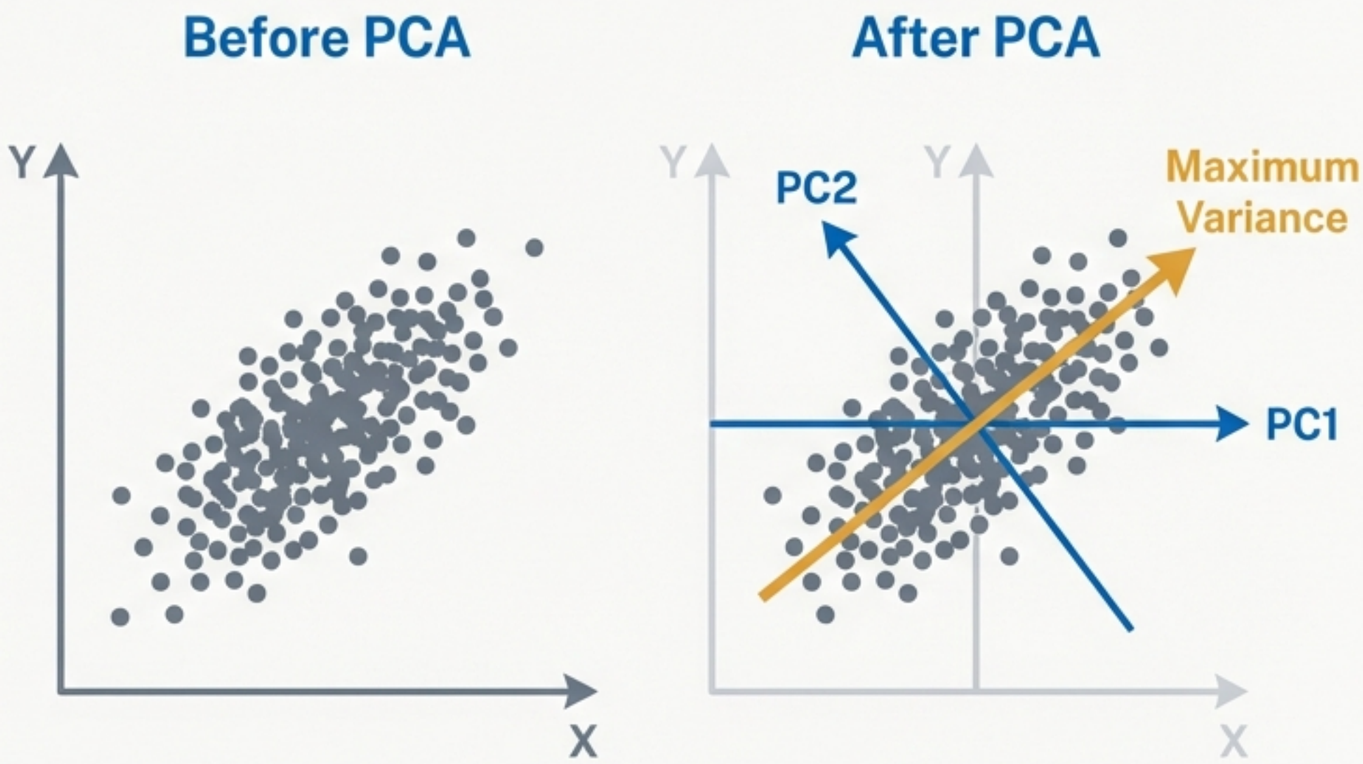
Principal Component Analysis (PCA): The Goal

Core Objective

PCA aims to find an optimal orthogonal transformation F that makes the dimensions in the new coordinate system linearly uncorrelated.

What this means in practice:

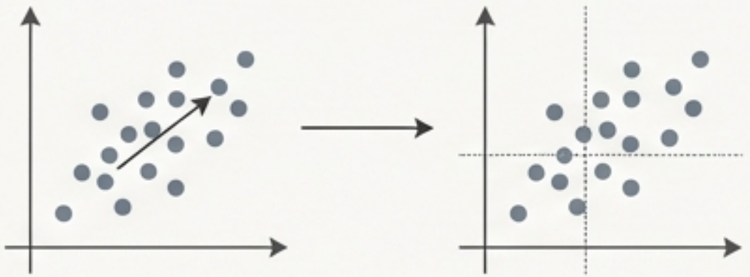
- The covariance matrix of the transformed data becomes a diagonal matrix. All off-diagonal elements (covariances between different dimensions) are zero.
- The new axes, called Principal Components (HEX #0065BD), are ordered by the amount of variance they capture from the original data. The first principal component is the direction of maximum variance.



The PCA Algorithm

1. Center the Data

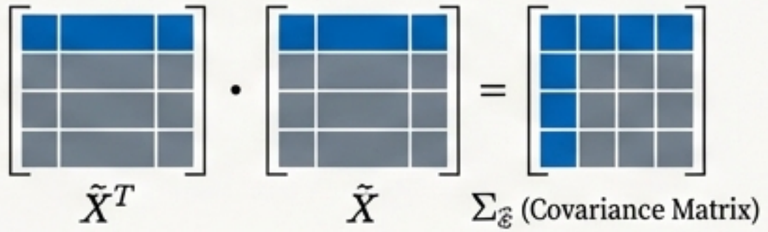
Calculate the mean vector $\bar{x}$ of the data points and subtract it from every data point to get the centered data $\tilde{x}_i = x_i - \bar{x}$. This results in a centered data matrix $\tilde{X}$.



2. Compute the Covariance Matrix

Calculate the covariance matrix of the centered data:

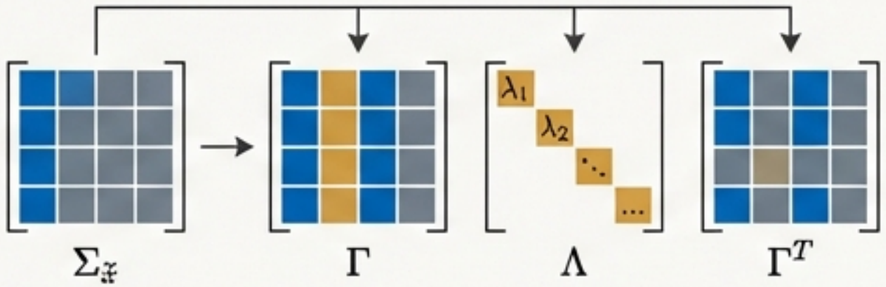
$$\Sigma_{\tilde{x}} = \frac{1}{N}\tilde{X}^T\tilde{X}$$



3. Perform Eigendecomposition

The covariance matrix is symmetric, so we can perform an eigendecomposition (spectral decomposition) to find its eigenvectors and eigenvalues:

$$\Sigma_{\tilde{x}} = \Gamma\Lambda\Gamma^T$$



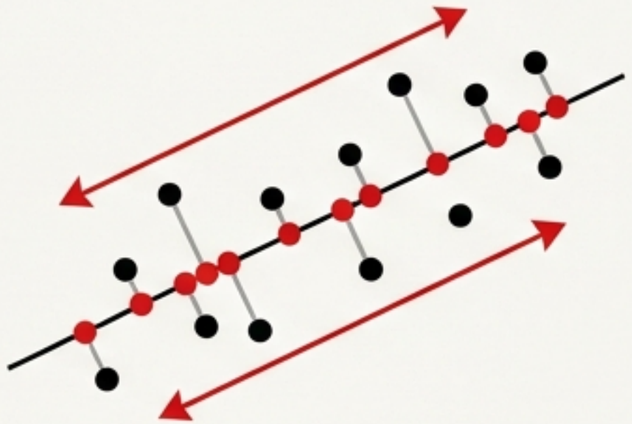
The Result

Principal Components: The columns of the matrix Γ are the eigenvectors, which are the principal components of the data. They form the new, optimal coordinate system.
Variances: The diagonal matrix Λ contains the eigenvalues, which represent the variance of the data along each corresponding principal component.

The Duality of PCA's Optimality

Two Equivalent Perspectives on What PCA Achieves

Maximize Variance

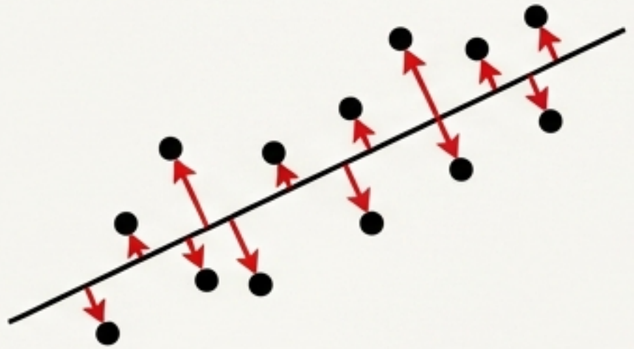


PCA finds the linear projection that maximizes the variance of the projected data.

Insight from Exercise 1

This is proven to be optimal for any M-dimensional subspace. The projection onto the M eigenvectors with the largest eigenvalues maximizes the variance. This is shown via proof by induction.

Minimize Reconstruction Error



PCA also finds the linear projection that minimizes the average squared distance (reconstruction error) between the original data points and their projected counterparts.

Insight from Exercise 2

Maximizing the variance of the projected data is equivalent to minimizing the reconstruction error.

PCA by the Numbers: A Worked Example

Based on Exercise 4: Given N=4 data points in $\mathbb{R}^{\wedge}3$.

Step 1: Original Data X and Centered Data X_c

$$\mathbf{X} = \begin{bmatrix} 4 & 3 & 2 \\ 2 & 1 & -2 \\ 4 & -1 & 2 \\ -2 & 1 & 2 \end{bmatrix}, \quad \text{Mean } \bar{\mathbf{x}} = [2, 1, 1] \quad \mathbf{X_c} = \begin{bmatrix} 2 & 2 & 1 \\ 0 & 0 & -3 \\ 2 & -2 & 1 \\ -4 & 0 & 1 \end{bmatrix}$$

Step 2: Covariance Matrix Σ_{X_c}

$$\Sigma_{X_c} = \frac{1}{4}\mathbf{X_c}^T\mathbf{X_c} = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}.$$

Step 3: Principal Components & Variances

Since Σ_{X_c} is diagonal, the principal components are the standard basis vectors [1,0,0], [0,1,0], and [0,0,1].

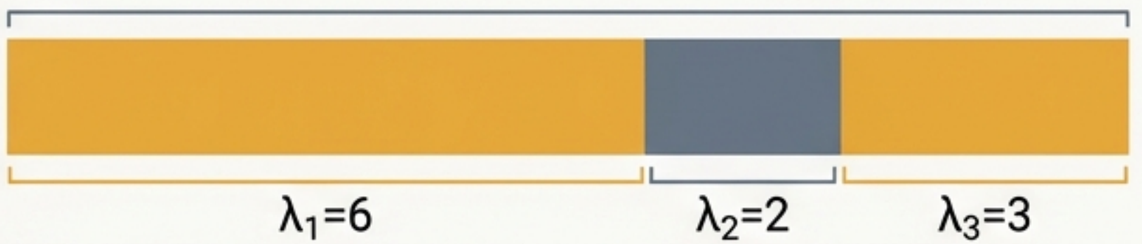
The variances (eigenvalues) are $\lambda_1=6$, $\lambda_2=2$, $\lambda_3=3$.

Step 4: Project to 2D

We select the PCs with the two largest eigenvalues: $\lambda_1=6$ (PC1) and $\lambda_3=3$ (PC3).

$$\mathbf{Y} = \mathbf{X_c} \cdot \mathbf{\Gamma}_{\text{trunc}} = \begin{bmatrix} 2 & 1 \\ 0 & -3 \\ 2 & 1 \\ -4 & 1 \end{bmatrix}.$$

Step 5: Variance Preserved



The fraction of total variance preserved is $(\lambda_1 + \lambda_3) / (\lambda_1 + \lambda_2 + \lambda_3) = (6 + 2 + 3) = 9/11$.

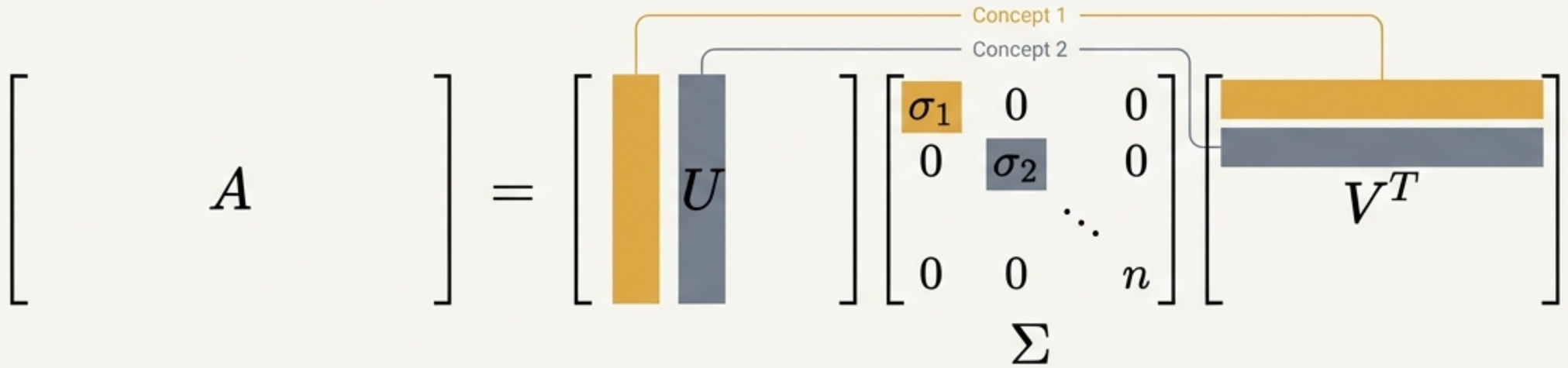
How Common Transformations Affect PCA

From Exercise 3: We apply PCA to a dataset X and find that the top 5 principal components preserve 70% of the variance. What happens if we first transform X?

Transformation	Variance Preserved	Justification
Isotropic Scaling (XS, where $S = \lambda I$)	**70%**	All eigenvalues are scaled by λ^2. The ratio of variances remains unchanged.
Rotation/Reflection (XR, where $RR^T = I$)	**70%**	The data is rotated, changing the eigenvectors, but the eigenvalues (the variances themselves) are invariant to rotation.
Shifting ($X + 1\mu^T$)	**70%**	PCA's first step is to center the data by subtracting the mean. An initial shift has no effect on the final result.
Anisotropic Scaling (XQ, $Q = \text{diag}(1..D)$)	**Cannot Tell**	Each dimension is scaled by a different factor. This fundamentally alters the covariance structure and the relative importance of components.
Rank Reduction (XA, where $\text{rank}(A) = 5$)	**100%**	The transformed data $Y_6 = XA$ will have a rank of at most 5. This means the data lies in a subspace of 5 or fewer dimensions, so the first 5 PCs will capture all of its variance.

Beyond PCA: Singular Value Decomposition (SVD)

- A **More General Decomposition**: SVD is a factorization that applies to *any* real matrix A (not just symmetric covariance matrices).
- This is an **exact** representation of A.



U

The matrix of **left singular vectors**. Its columns are orthonormal. Maps rows of A to a "concept space".

Σ

A **diagonal matrix** whose entries are the **singular values** ($\sigma_1 \geq \sigma_2 \geq \cdots > 0$). They represent the "strength" of each underlying concept or latent factor.

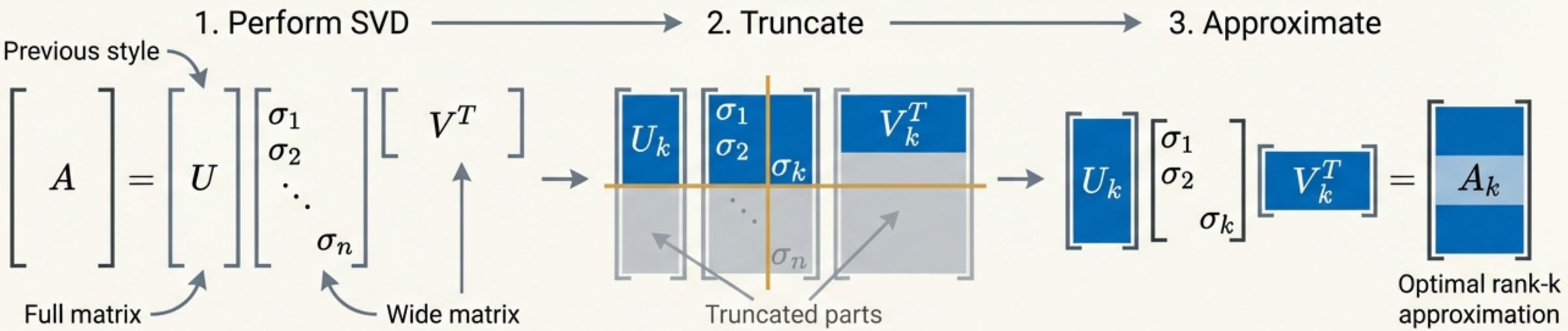
V

The matrix of **right singular vectors**. Its columns are also orthonormal. Maps columns of A to the "concept space".

SVD's Superpower: Optimal Low-Rank Approximation

While SVD provides an exact decomposition of a matrix A, its real power comes from approximation. We can create the best possible rank-k approximation of A by truncating the SVD.

Visual Diagram: How it Works



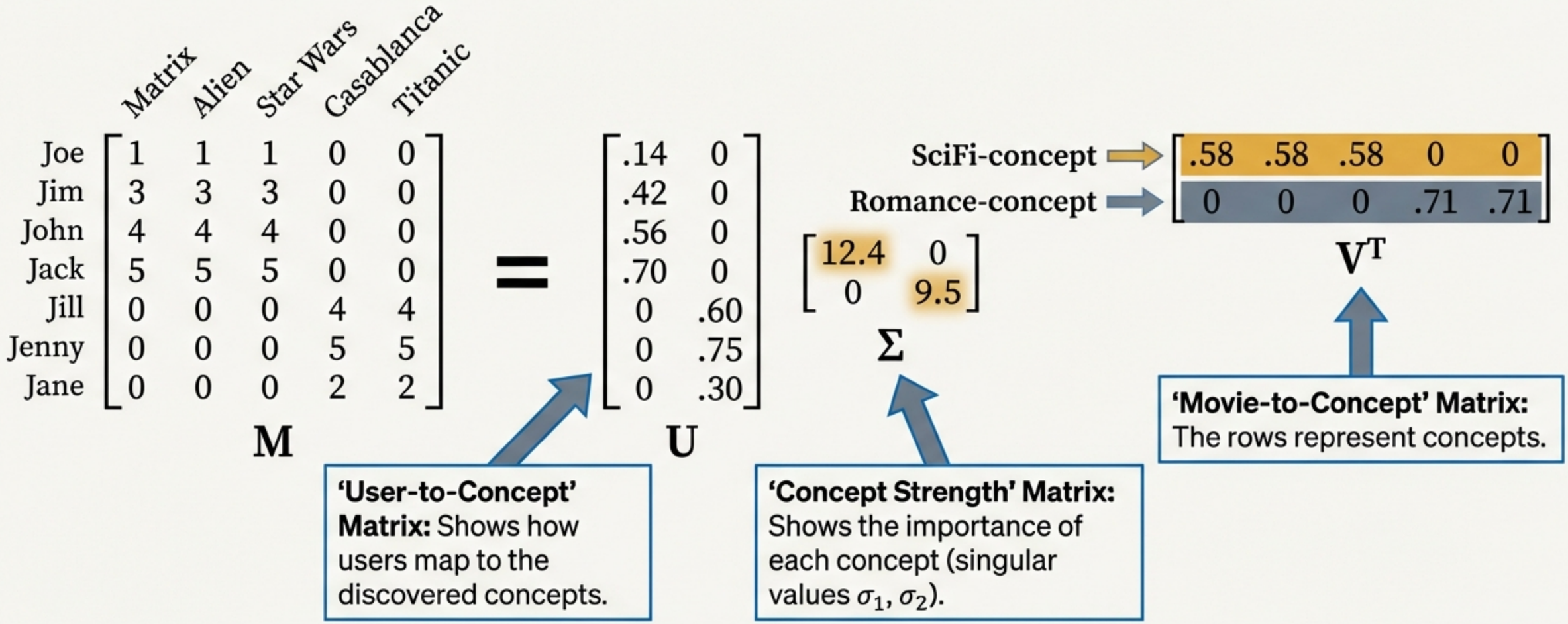
The Approximation Equation: $A_k = U_k \Sigma_k V_k^T$

What 'Best' Means

The matrix A_k is the rank-k matrix that minimizes the Frobenius norm of the reconstruction error: $||A - A_k||_F^2$.

Application: Uncovering 'Concepts' with SVD

A user-movie rating matrix M, where rows are users and columns are movies.



SVD automatically discovers these hidden dimensions from the data.

SVD for Recommendations: A Worked Example

CALCULATION

Scenario (from Exercise 5):
A new user, Leslie, has a rating vector [0, 3, 0, 0, 4] for the movies (Matrix, Alien, Star Wars, Casablanca, Titanic).

Goal: Find Leslie's representation in the 'concept space' to understand her preferences.

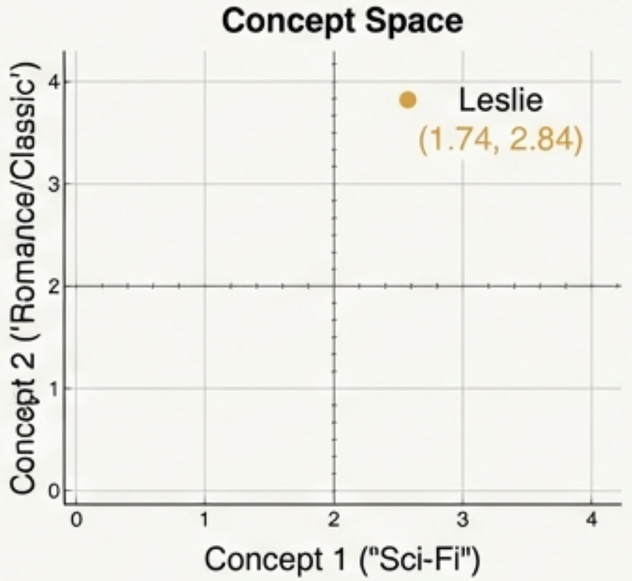
Method:
Project her rating vector into the concept space using the right singular vectors: new_user_vector · **V**.

$$\begin{bmatrix}0 & 3 & 0 & 0 & 4\end{bmatrix} \times \begin{bmatrix}.58 & .58 & .58 & 0 & 0\\ 0 & 0 & 0 & .71 & .71\end{bmatrix}^T$$

$$\rightarrow \begin{bmatrix}1.74 & 2.84\end{bmatrix}$$

Leslie's new concept coordinates.

INTERPRETATION



Interpretation:

- Leslie's score for Concept 1 ('Sci-Fi') is **1.74**.
- Leslie's score for Concept 2 ('Romance/Classic') is **2.84**.

Conclusion:
Leslie has a stronger preference for the 'Romance/Classic' concept. Since she has already seen 'Titanic,' the model would suggest 'Casablanca' as a good recommendation.

Application: SVD for Efficient Linear Regression

The Problem (from Exercise 6): Find the optimal weights $\mathbf{w}^*$ for linear regression given features $\mathbf{X}$ and targets $\mathbf{y}$.

Standard Solution (Normal Equation)

Source Serif Pro is ta inversion, which has a computational complexity of $O(D^3)$ and its numerically.

$$\mathbf{w}^* = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

This requires a **matrix inversion**, which has a **computational complexity** of $O(D^3)$ and can be numerically unstable.

SVD-based Solution

Derivation: Substitute $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$ into the normal equation. After simplification, the expression becomes:

$$\mathbf{w}^* = \mathbf{V}\Sigma^{-1}\mathbf{U}^T\mathbf{y}$$

This form **avoids the explicit, expensive** $(\mathbf{X}^T\mathbf{X})$ inversion. The inversion of the diagonal matrix Σ is trivial $(O(D))$.

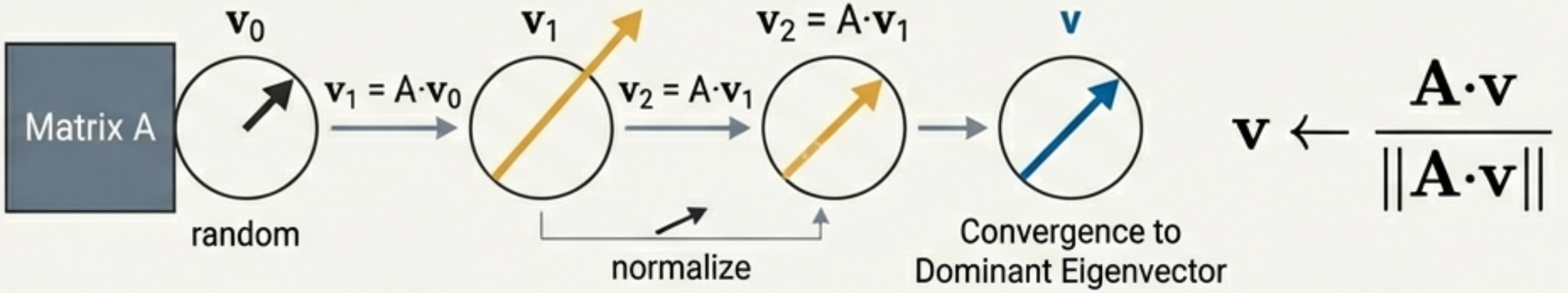
The total complexity is reduced to $O(ND)$ (for $N > D$), a significant improvement over $O(D^3)$.

Computational Considerations: Power Iteration

How are eigenvectors and eigenvalues computed for large matrices where full decomposition is infeasible?

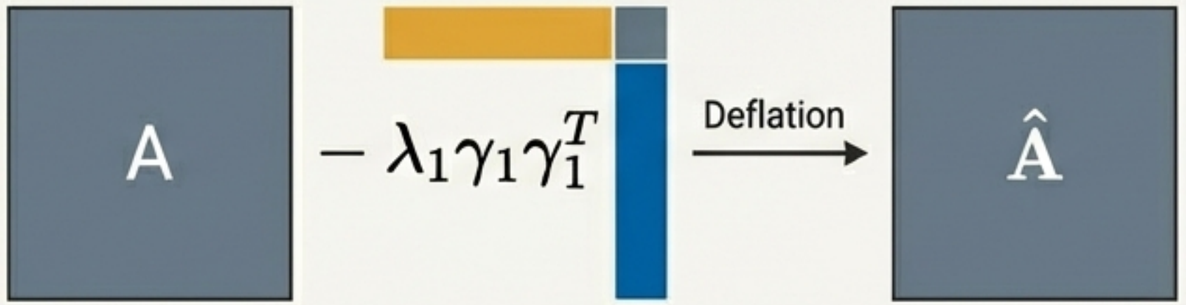
Power Iteration

An iterative algorithm to find the single eigenvector corresponding to the largest eigenvalue.



Deflation

Once the first eigenpair (λ_1, γ_1) is found, a "deflated" matrix $\hat{A} = A - \lambda_1 \gamma_1 \gamma_1^T$ is constructed. The dominant eigenvector of $\hat{A}$ is the second eigenvector of A. This process can be repeated.



Chapter 10: Your Dimensionality Reduction Toolkit

Principal Component Analysis (PCA)

Primary Goal

Find a new, uncorrelated coordinate system that maximizes the variance of the projected data.

Method

Eigendecomposition of the data's **covariance matrix** ($\Sigma_x = \Gamma\Lambda\Gamma^T$).

Operates On

Relationships between features (covariance).

Core Use Cases

Finding principal axes of variation, data compression and noise reduction, visualization.

Singular Value Decomposition (SVD)

Primary Goal

Decompose any data matrix to find its optimal low-rank approximation and uncover latent factors.

Method

Decomposition of the **data matrix itself** ($A = U\Sigma V^T$).

Operates On

The raw data matrix.

Core Use Cases

Latent semantic analysis, recommendation systems, data compression, a stable numerical tool for other algorithms (e.g., regression).