Principal Component Analysis (PCA)

GENOME 560, Spring 2026

Saori Sakaue (sakaue@uw.edu)

Section 1

Recap: Matrix Operations

Symmetric Matrix

$$\mathbf{A} = \mathbf{A}^T \qquad a_{ij} = a_{ji}$$

Example: covariance matrices are always symmetric.

Diagonal Matrix

$$\mathbf{D} = \begin{pmatrix} d_1 & 0 & 0 \\ 0 & d_2 & 0 \\ 0 & 0 & d_3 \end{pmatrix}$$

Identity Matrix

$$\mathbf{I} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{AI} = \mathbf{IA} = \mathbf{A}$$

Matrix Multiplication (Recap)

$$\mathbf{A}_{m \times n} \cdot \mathbf{B}_{n \times p} = \mathbf{C}_{m \times p}$$

Inverse (Recap)

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$$

Matrix Multiplication is Projection

Multiplying a vector by a matrix **projects** it into different axes.

$$\mathbf{A}\mathbf{v} = \mathbf{w}$$

This is the key idea behind PCA: we find a matrix that projects data onto the axes of maximum variance.

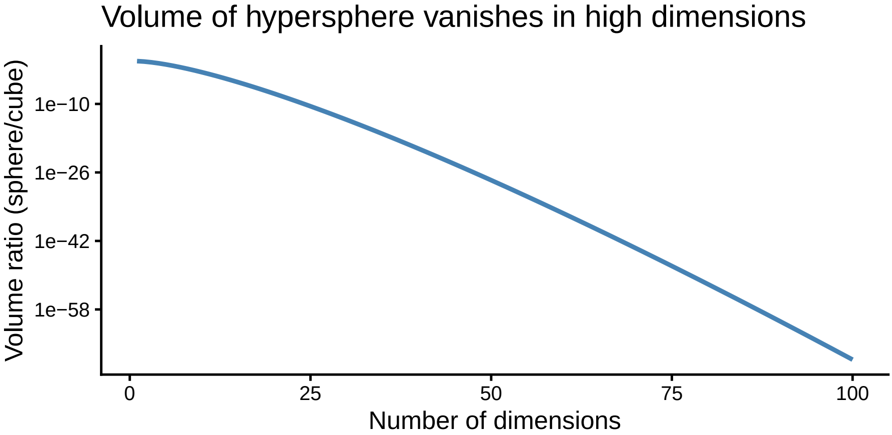
Section 2

Why PCA?

Why and When Do We Need PCA?

- High-dimensional data is hard to visualize and analyze
- Many features may be correlated (redundant)
- We want a **low-dimensional summary** that captures the most important variation

The Curse of Dimensionality



The Curse of Dimensionality (continued)

- In high dimensions, **all points become approximately equidistant**
- Distance-based methods (clustering, KNN) lose discriminating power
- Most of the volume concentrates near the surface

"You really shouldn't try to use your 20K gene expression data to compare and characterize your cells!" — without dimensionality reduction

Learning Low-Dimensional Representations

Linear methods:

- **PCA**, Kernel PCA, Sparse PCA, ICA, CCA, NMF

Non-linear methods:

- t-SNE, UMAP, Autoencoders

Section 3

PCA Definition and Theory

Definition of PCA

Given N samples with M features, PCA finds new axes (principal components) such that:

1. Each PC captures **maximal variance**
2. PCs are **mutually uncorrelated** (orthogonal)
3. PCs are ordered by the amount of variance they explain

Relationship to Linear Regression

- **Regression:** minimize vertical distance to line (in Y)
- **PCA:** minimize perpendicular distance to line (in all dimensions)

PCA Objective

Two equivalent formulations:

1. Maximize variance of projected data
2. Minimize reconstruction error

$$\mathsf{PC}_1 = \arg\max_{\|\mathbf{v}\|=1} \mathsf{Var}(\mathbf{X}\mathbf{v})$$

Covariance

$$\text{Cov}(X, Y) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$$

- Positive covariance: variables increase together
- Negative covariance: one increases as other decreases
- Zero covariance: no linear relationship

Covariance Matrix

For N samples $\times$ M features matrix $\mathbf{X}$ (centered):

$$\mathbf{C} = \frac{1}{N-1}\mathbf{X}^T\mathbf{X}$$

$\mathbf{C}$ is $M \times M$, symmetric, with variances on diagonal and covariances off-diagonal.

Covariance Matrix (continued)

$$\mathbf{C} = \begin{pmatrix} \mathsf{Var}(X_1) & \mathsf{Cov}(X_1, X_2) & \cdots \\ \mathsf{Cov}(X_2, X_1) & \mathsf{Var}(X_2) & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

Section 4

Eigen-decomposition

Eigen-decomposition

$$\mathbf{C}\mathbf{v} = \lambda\mathbf{v}$$

- $\mathbf{v}$: **eigenvector** (direction)
- λ: **eigenvalue** (magnitude of variance along that direction)

Eigen-decomposition

$$\mathbf{C} = \mathbf{V}\Lambda\mathbf{V}^T$$

where $\mathbf{V}$ = matrix of eigenvectors (columns), Λ = diagonal matrix of eigenvalues.

Simple Example

$$\mathbf{C} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

Eigenvalues: $\lambda_1 = 3, \lambda_2 = 1$

Eigenvectors: $\mathbf{v}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \mathbf{v}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$

Section 5

Steps to Perform PCA

Step 1: Center (and Optionally Normalize)

Given $N \times M$ data matrix $\mathbf{X}$:

$$\mathbf{X}_c = \mathbf{X} - \mu$$

where μ is the column mean vector.

Step 2: Compute Covariance Matrix

$$\mathbf{C} = \frac{1}{N-1}\mathbf{X}_c^T\mathbf{X}_c$$

Step 3: Eigen-decomposition

$$\mathbf{C} = \mathbf{V}\Lambda\mathbf{V}^T$$

Sort eigenvectors by decreasing eigenvalue: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_M$

Step 4: Select Top S Eigenvectors

Choose $S \ll M$ components that capture most variance.

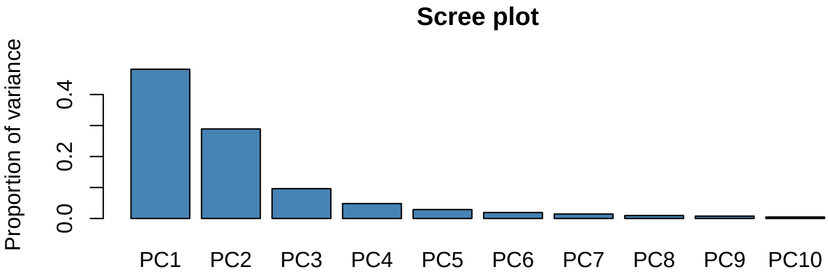
Step 5: Project

$$\mathbf{X}_{\text{projected}} = \mathbf{X}_c \cdot \mathbf{V}_S$$

where $\mathbf{V}_S$ contains the top S eigenvectors.

Variance Explained

$$\text{Variance explained by PC}_k = \frac{\lambda_k}{\sum_{j=1}^{M} \lambda_j}$$



Does the First Eigenvector Maximize Variance?

Yes! The eigenvector corresponding to the largest eigenvalue of $\mathbf{C}$ defines the direction of maximum variance in the data.

Proof sketch: Maximize $\mathbf{v}^T\mathbf{C}\mathbf{v}$ subject to $\|\mathbf{v}\| = 1$. Using Lagrange multipliers:

$$\mathbf{C}\mathbf{v} = \lambda\mathbf{v}$$

The maximum of $\mathbf{v}^T\mathbf{C}\mathbf{v} = \lambda$ is achieved at the largest eigenvalue.