南方科技大学

MAT8034: Machine Learning

Principal Components Analysis

Fang Kong

https://fangkongx.github.io/Teaching/MAT8034/Spring2025/index.html

Part of slide credit: Stanford CS229

Motivation

Motivation

Detecting Redundancy in Data

🚗 Example 1: Redundant Features in Car Attributes

- Dataset: $\{x^{(i)}\}_{i=1}^{n}$, where each $x^{(i)} \in \mathbb{R}^d$
- Each $x^{(i)}$ contains attributes of a different automobile (e.g., max speed, turn radius)
- Unknown to us:
  - One feature x_i: max speed in **miles per hour**
  - Another feature x_j: max speed in **kilometers per hour**
- These two features are **almost linearly dependent**
  - Only minor differences due to rounding
- Therefore, data lies approximately on an $n-1$ dimensional subspace
- **Goal**: Automatically detect and remove this redundancy

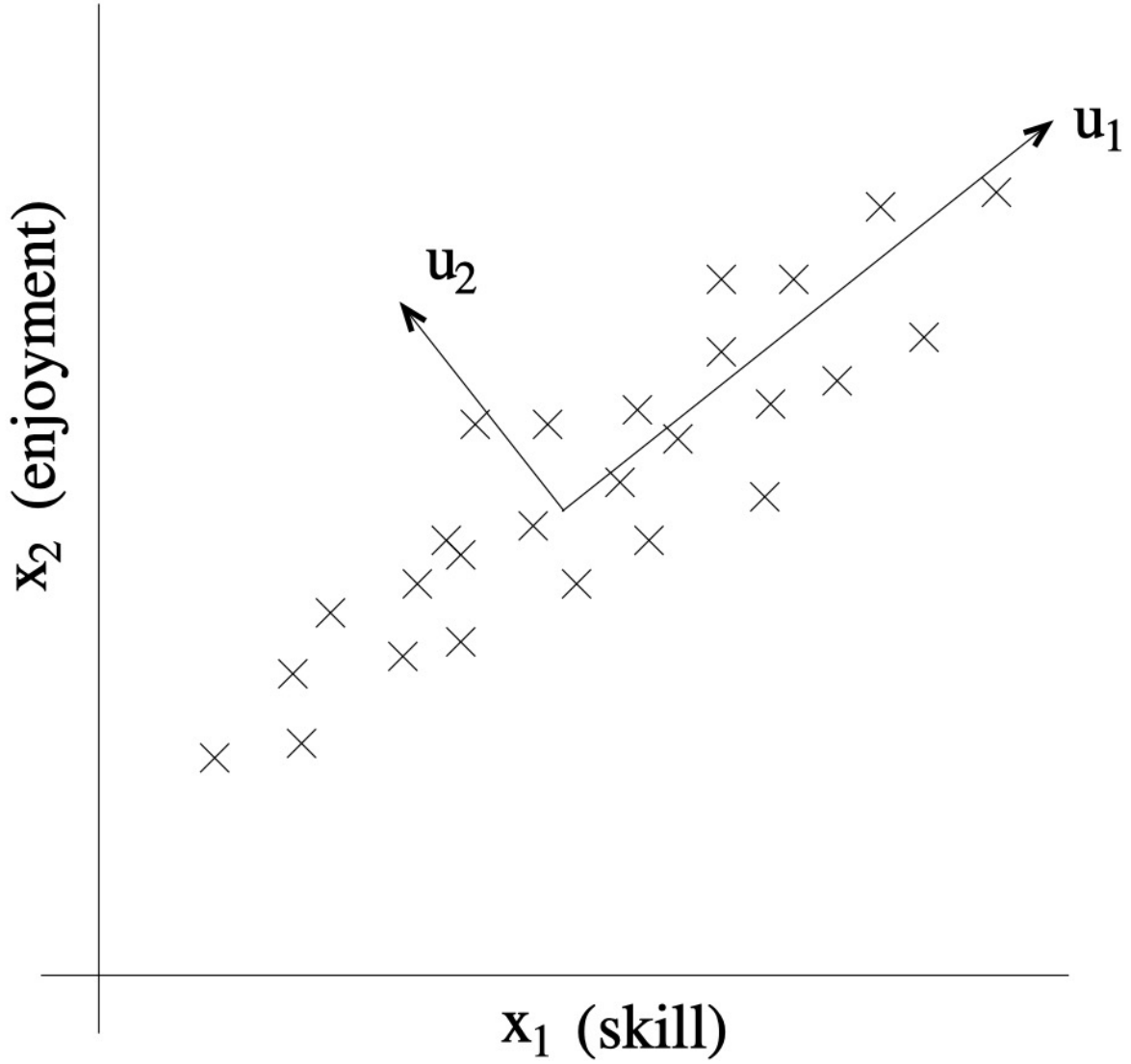
Motivation (cont'd)

- ## Detecting Redundancy in Data

🚁 Example 2: RC Helicopter Pilot Survey

- Data from a survey of RC helicopter pilots
- Two attributes:
  - x_1: pilot's skill level
  - x_2: enjoyment of flying
- RC helicopters are hard to fly → only those who **enjoy** it become **skilled**
- Strong **correlation** between x_1 and x_2
- Data likely lies along a **diagonal axis** (denoted u_1)
  - Captures intrinsic "piloting karma"
  - Only small noise lies off this axis
- **Goal**: Automatically compute this meaningful direction u_1

Illustration



Pre-processing

Intuition

- Features with larger scales (e.g., 1000) can dominate those with smaller scales (e.g., 0.01)
- Normalization gives all features equal weight in the analysis

- If different attributes are all on the same scale, rescaling may be omitted

Normalization

- Subtracting the mean and dividing by the empirical standard deviation

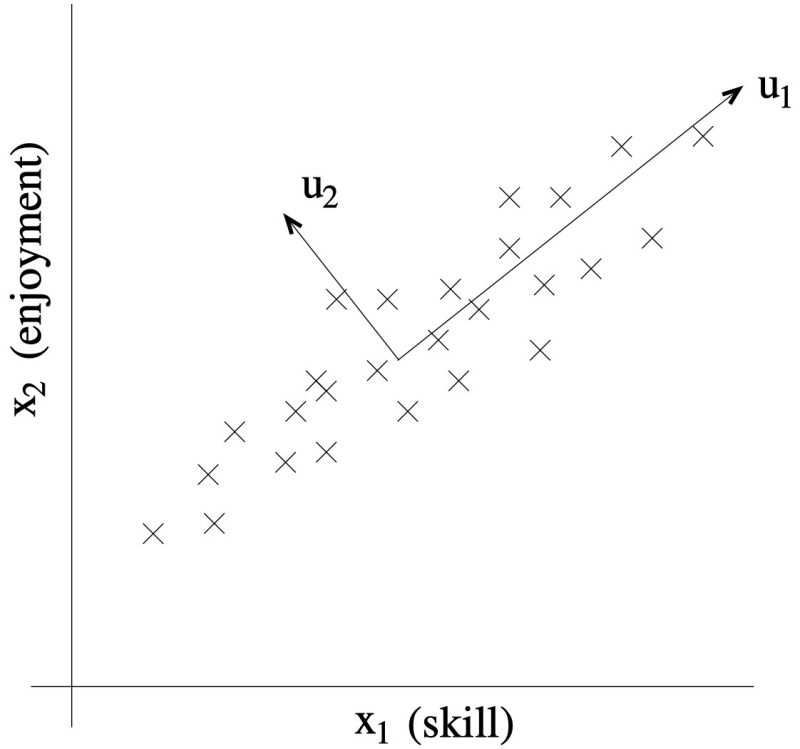
$$x_j^{(i)} \leftarrow \frac{x_j^{(i)} - \mu_j}{\sigma_j}$$

- $\mu_j = \frac{1}{n} \sum_{i=1}^{n} x_j^{(i)}$

$\sigma_j^2 = \frac{1}{n} \sum_{i=1}^{n} (x_j^{(i)} - \mu_j)^2$

Algorithm

Which basis to select?

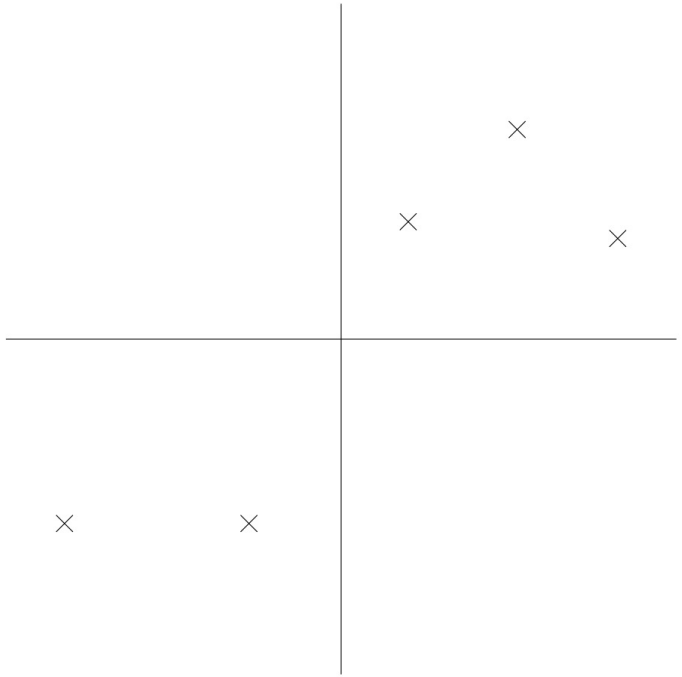


- The direction on which the data approximately lies

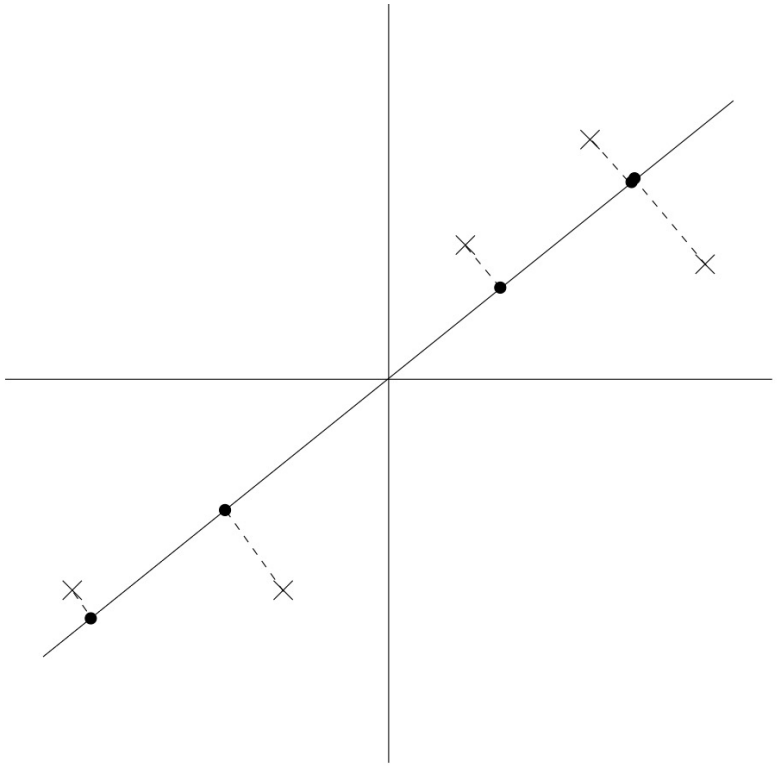
Intuition

- The data has natural "spread" in some directions more than others
- The **major axis** is the direction where data varies the most
- If we project data onto this axis, we **retain the most information** (variance)

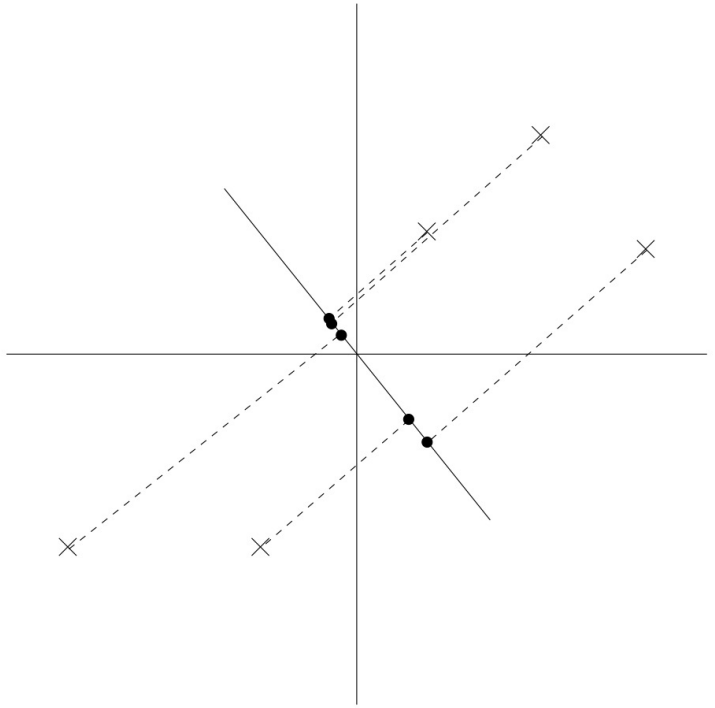
Example



Data



Selection 1



Selection 2

Mathematical Formulation

- The length of the projection of x onto u is $x^\top u$
- Maximizing the variance of the projections is equivalent to maximize

$$\frac{1}{n}\sum_{i=1}^{n}(x^{(i)T}u)^2 = \frac{1}{n}\sum_{i=1}^{n}u^T x^{(i)}x^{(i)T}u$$

$$= u^T\left(\frac{1}{n}\sum_{i=1}^{n}x^{(i)}x^{(i)T}\right)u.$$

Solution

- We want to maximize $\mathbf{u}^T \Sigma \mathbf{u}$

  Subject to: $\mathbf{u}^T \mathbf{u} = 1$

  Lagrangian:

$$\mathcal{L}(\mathbf{u}, \lambda) = \mathbf{u}^T \Sigma \mathbf{u} - \lambda(\mathbf{u}^T \mathbf{u} - 1)$$

  Set gradient to 0:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{u}} = 0 \Rightarrow \Sigma \mathbf{u} = \lambda \mathbf{u}$$

- The objective becomes finding the principal eigenvector of Σ

Extension to larger dimension

- If we wish to project our data into a k-dimensional subspace (k < d)
- Choose to be the top k eigenvectors of Σ

- Due to that Σ is symmetric, u_i's will be orthogonal to each other
- u_i's now form a new orthogonal basis for the data

Obtain new, low-dimension features

- Represent the data in the new basis

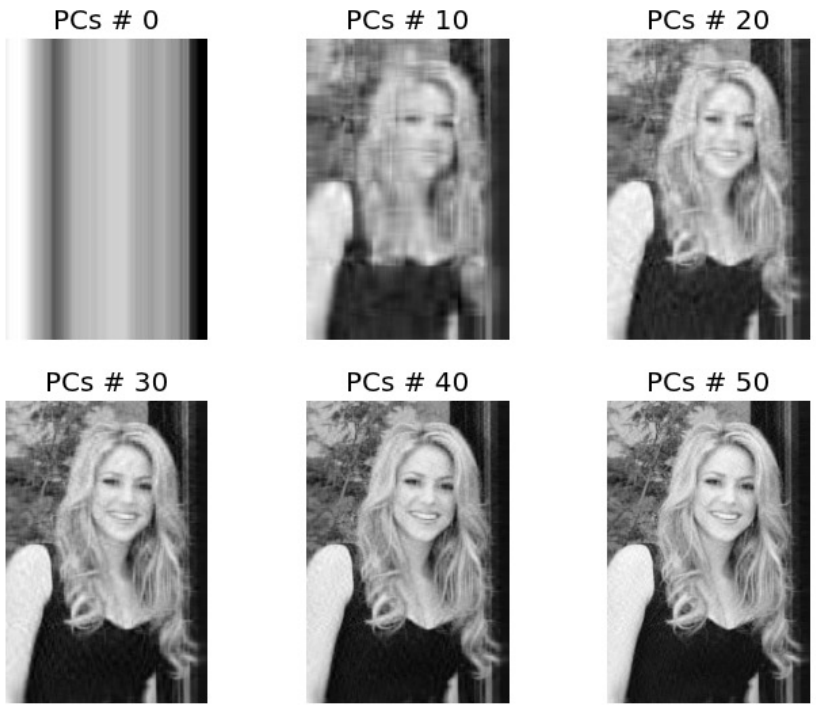
$$y^{(i)} = \begin{bmatrix} u_1^T x^{(i)} \\ u_2^T x^{(i)} \\ \vdots \\ u_k^T x^{(i)} \end{bmatrix} \in \mathbb{R}^k$$

- PCA is also referred to as a dimensionality reduction algorithm
- The vectors $u_1, \ldots, u_k$ are called the first k principal components of the data

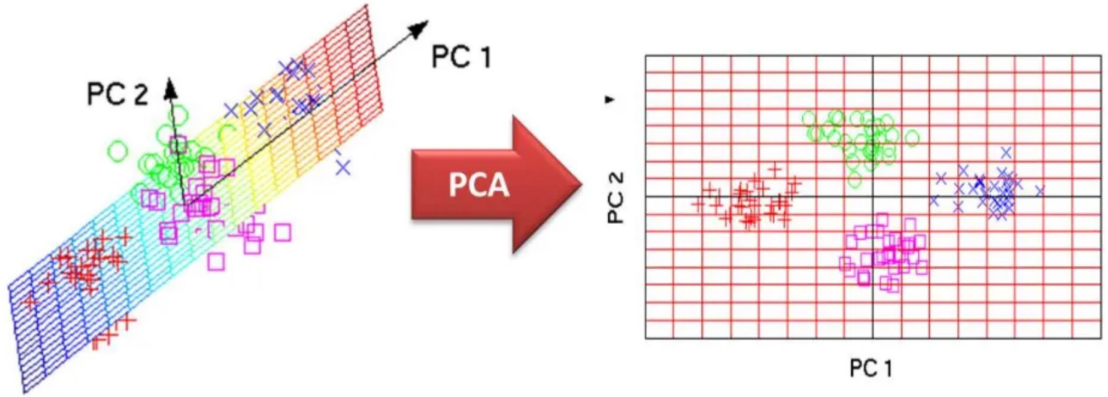
Other interpretation of PCA

- In this class: maximize the variance
- In the homework:
  - You will show that PCA minimizes the approximation error

Applications of PCA



Compression



Data preprocessing

https://glowingpython.blogspot.com/2011/07/pca-and-image-compression-with-numpy.html
https://ashutoshtripathi.com/2019/07/11/a-complete-guide-to-principal-component-analysis-pca-in-machine-learning/

Summary

- Principal components analysis (PCA)
  - Motivation: remove redundancy in data
  - Main idea: maximize the projection variance