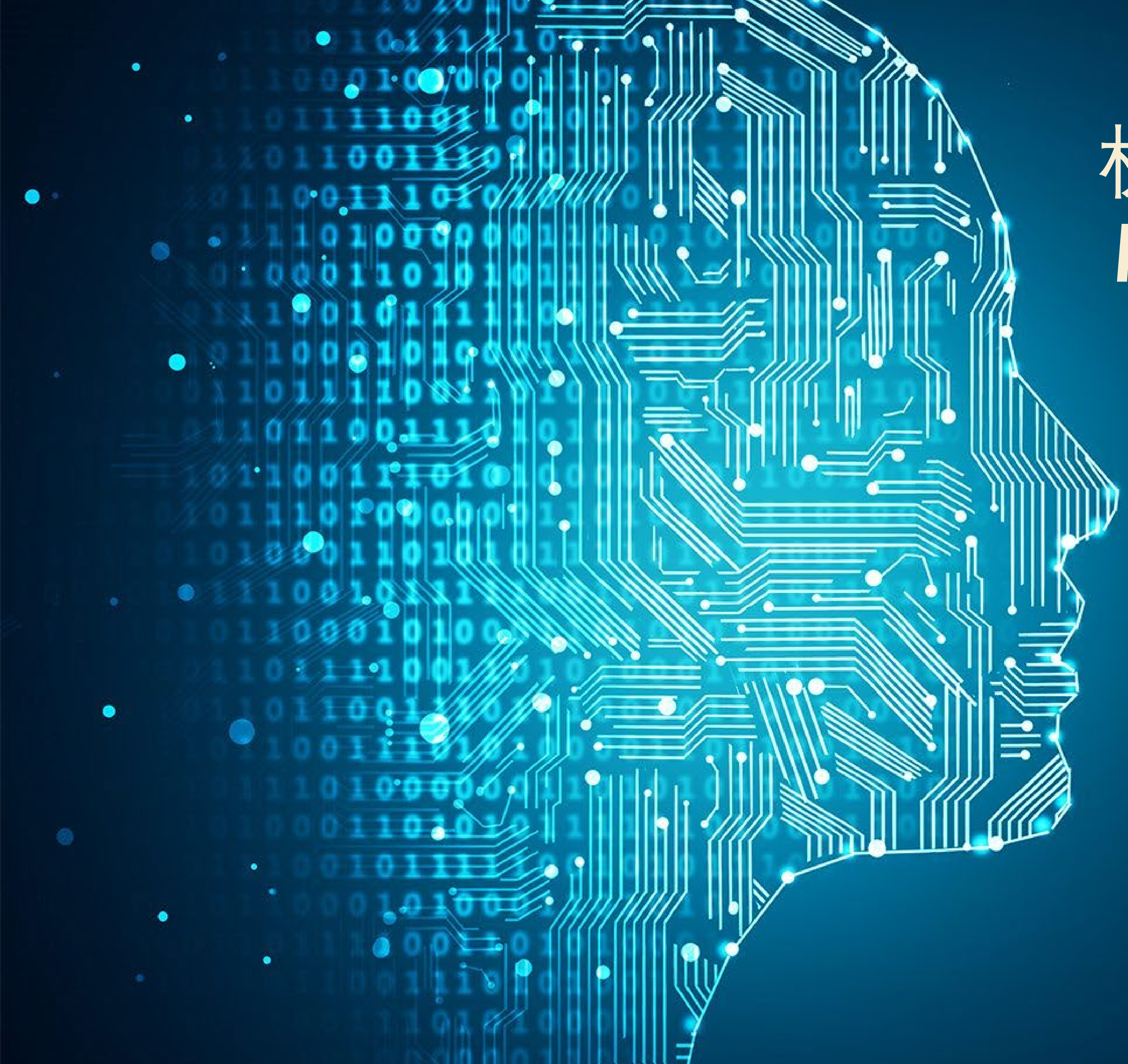


机器学习与人工智能
Machine Learning and Artificial Intelligence

Lecture 7 PCA

Yingjie Zhang (张颖婕)
Peking University
yingjiezhang@gsm.pku.edu.cn

2021 Fall

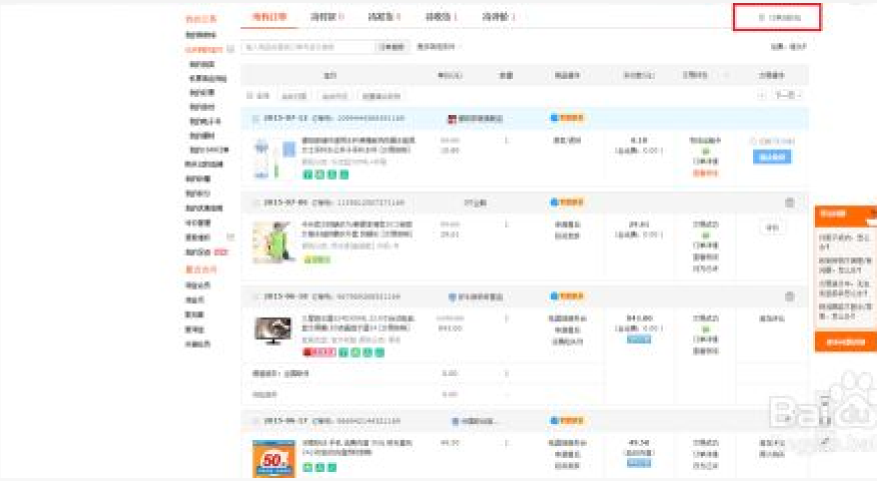
Principal Component Analysis

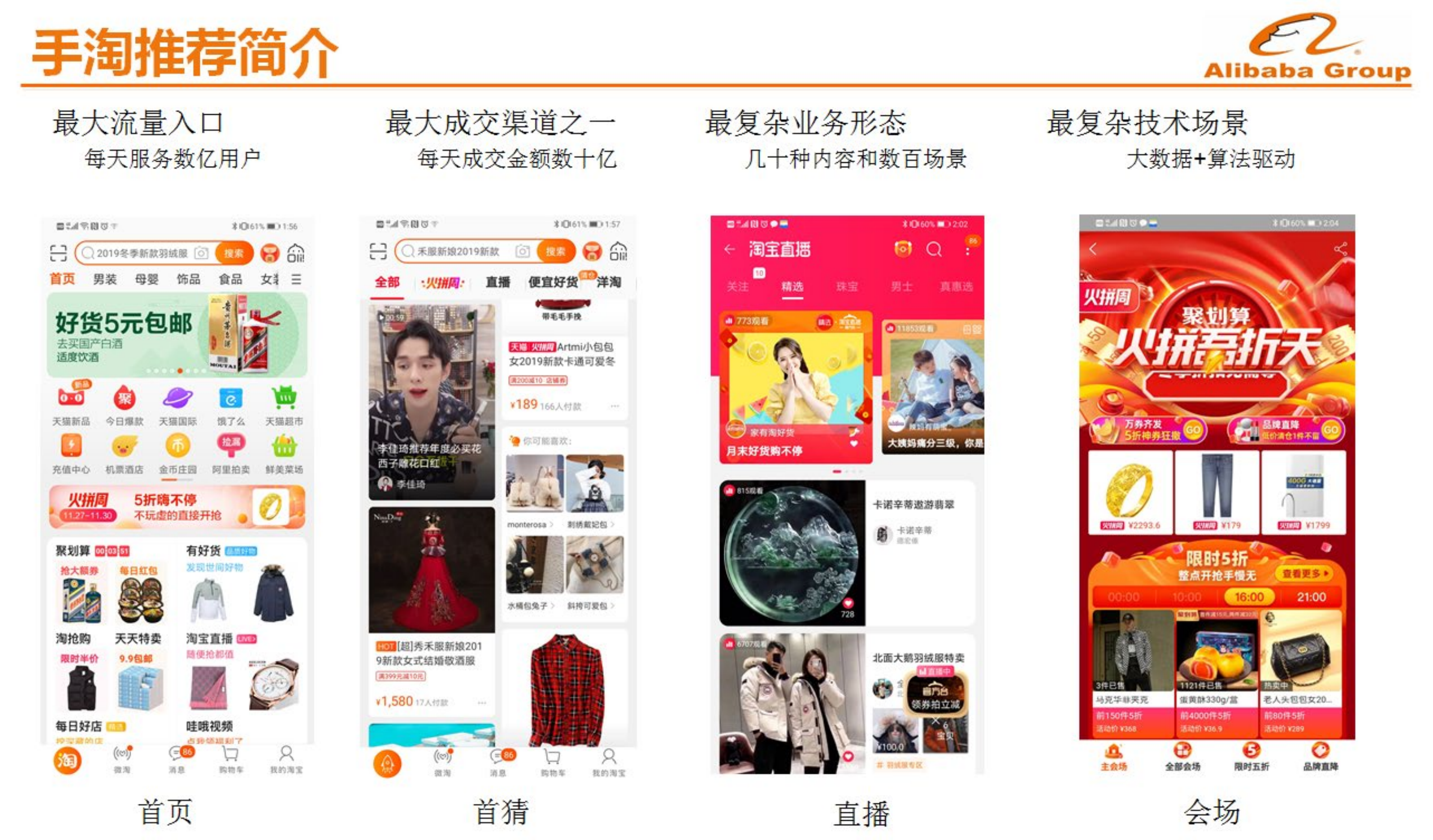
High Dimension Data

- High resolution images (millions of pixels)

High Dimension Data

- Customer purchase data





Useful for

- Visualization
- More efficient use of resources
  (e.g., time, memory, communication)
- Statistical: fewer dimensions → better generalization
- Noise removal (improving data quality)
- Further processing by ML algorithms

PCA Overview

- PCA is a technique that can simplify data
- It is a linear transformation that chooses a new coordinate system for the data set such that
  - greatest variance by any projection of the data set comes to lie on the first axis (then called the first principal component)
  - the second greatest variance on the second axis, and so on.

Toy Example

Consider the following 3D data points

1	2	4	3	6	5
2	4	8	6	12	10
3	6	12	9	18	15

If each component is stored in a byte,
we need 18 = 3 x 6 bytes

Toy Example

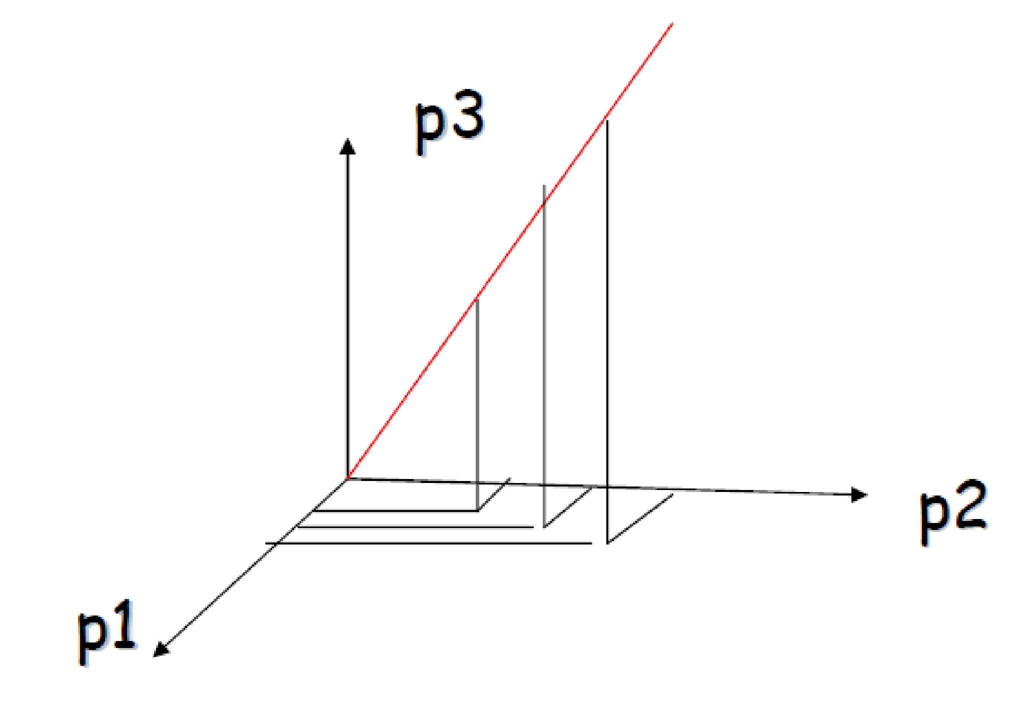
Consider the following 3D data points

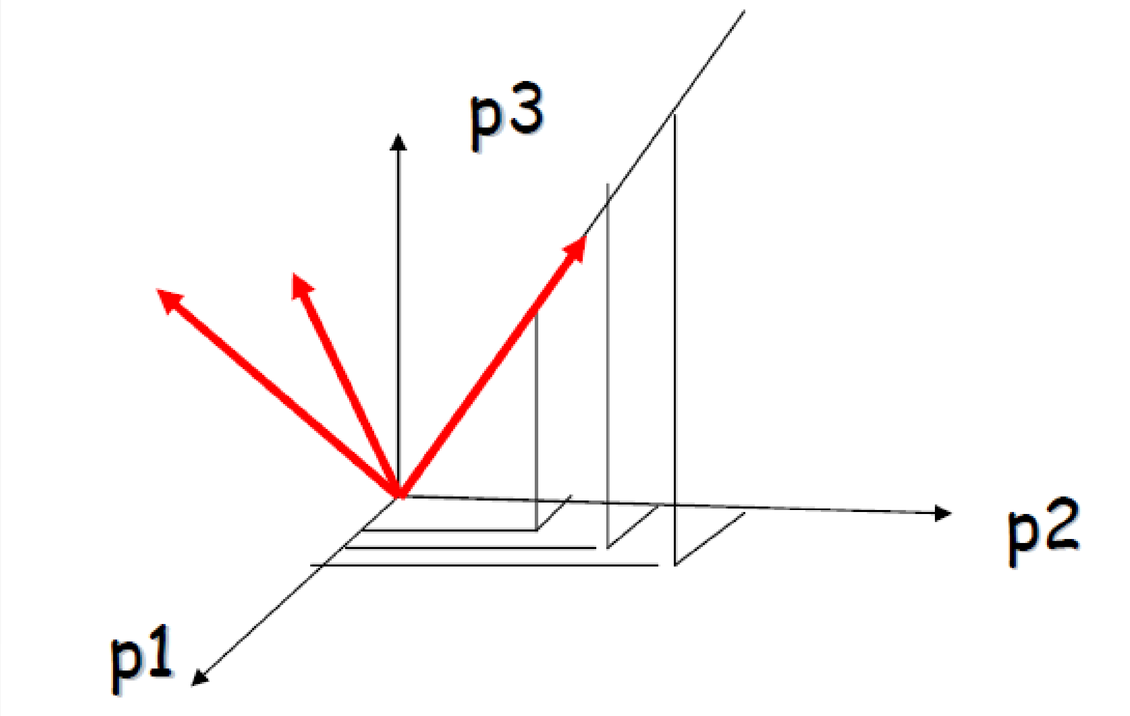
$$\begin{bmatrix}1\\2\\3\end{bmatrix} = 1*\begin{bmatrix}1\\2\\3\end{bmatrix} \qquad \begin{bmatrix}2\\4\\6\end{bmatrix} = 2*\begin{bmatrix}1\\2\\3\end{bmatrix} \qquad \begin{bmatrix}4\\8\\12\end{bmatrix} = 4*\begin{bmatrix}1\\2\\3\end{bmatrix}$$

$$\begin{bmatrix}3\\6\\9\end{bmatrix} = 3*\begin{bmatrix}1\\2\\3\end{bmatrix} \qquad \begin{bmatrix}6\\12\\18\end{bmatrix} = 6*\begin{bmatrix}1\\2\\3\end{bmatrix} \qquad \begin{bmatrix}5\\10\\15\end{bmatrix} = 5*\begin{bmatrix}1\\2\\3\end{bmatrix}$$

They can be stored using only 9 bytes (50% savings!)

Toy Example





Principle Component Analysis

- Identifying the axes is known as Principal Components Analysis, and can be obtained by using classic matrix computation tools (Eigen or Singular Value Decomposition).

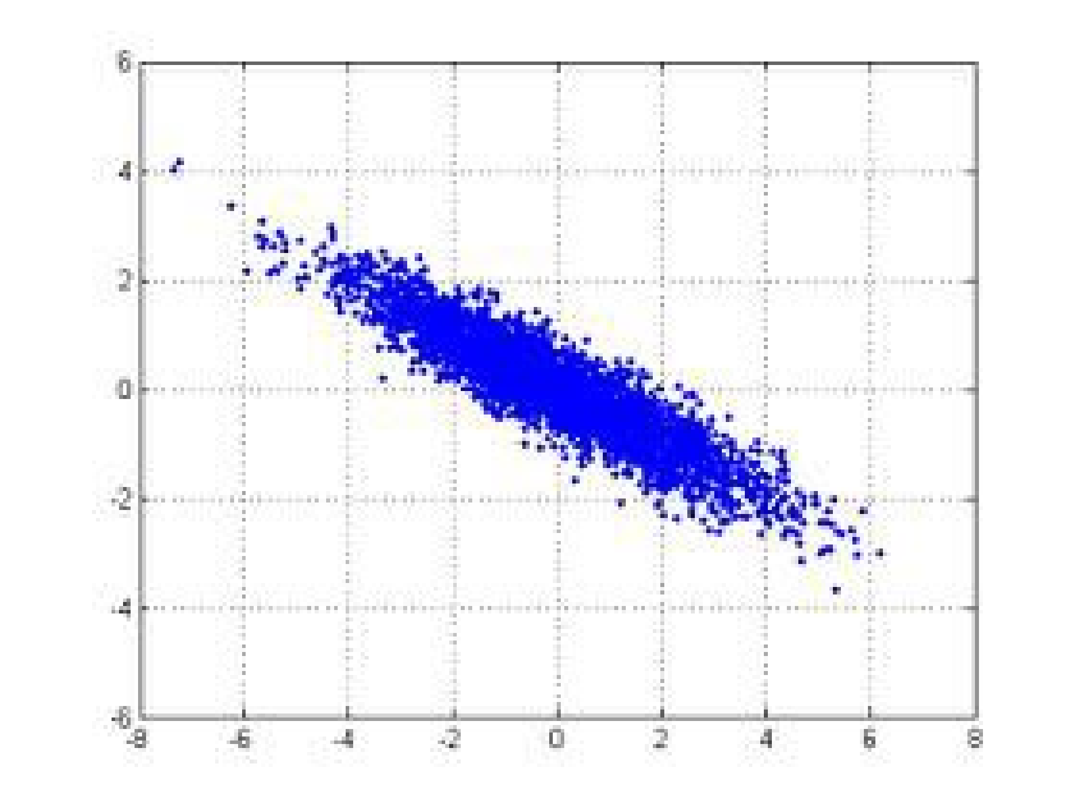
- Data for PCA:

$$\mathcal{D} = \{x^{(i)}\}_{i=1}^{N} \qquad X = \begin{bmatrix} \left(\boldsymbol{x}^{(1)}\right)^T \\ \left(\boldsymbol{x}^{(2)}\right)^T \\ \cdots \\ \left(\boldsymbol{x}^{(N)}\right)^T \end{bmatrix}$$

We assume the data is centered: $\mu = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}^{(i)} = \mathbf{0}$

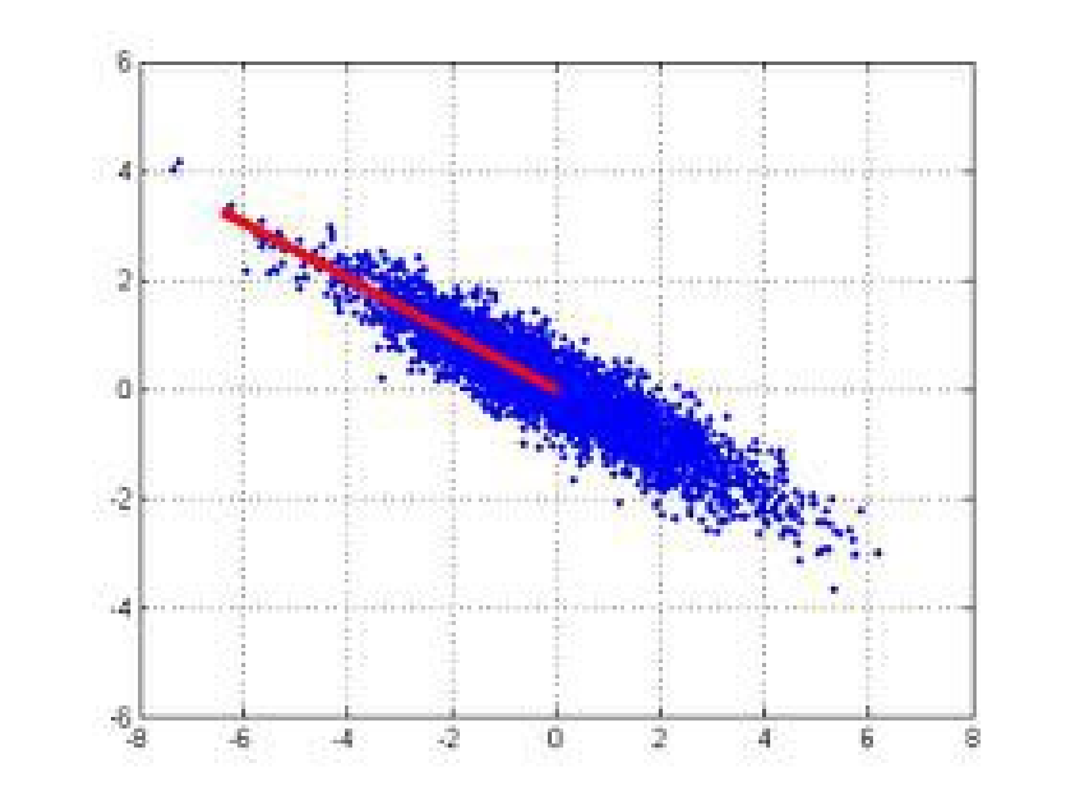
2D Gaussian Dataset

The original dataset:



2D Gaussian Dataset

First find the direction of maximum variance, labeled “Component 1”

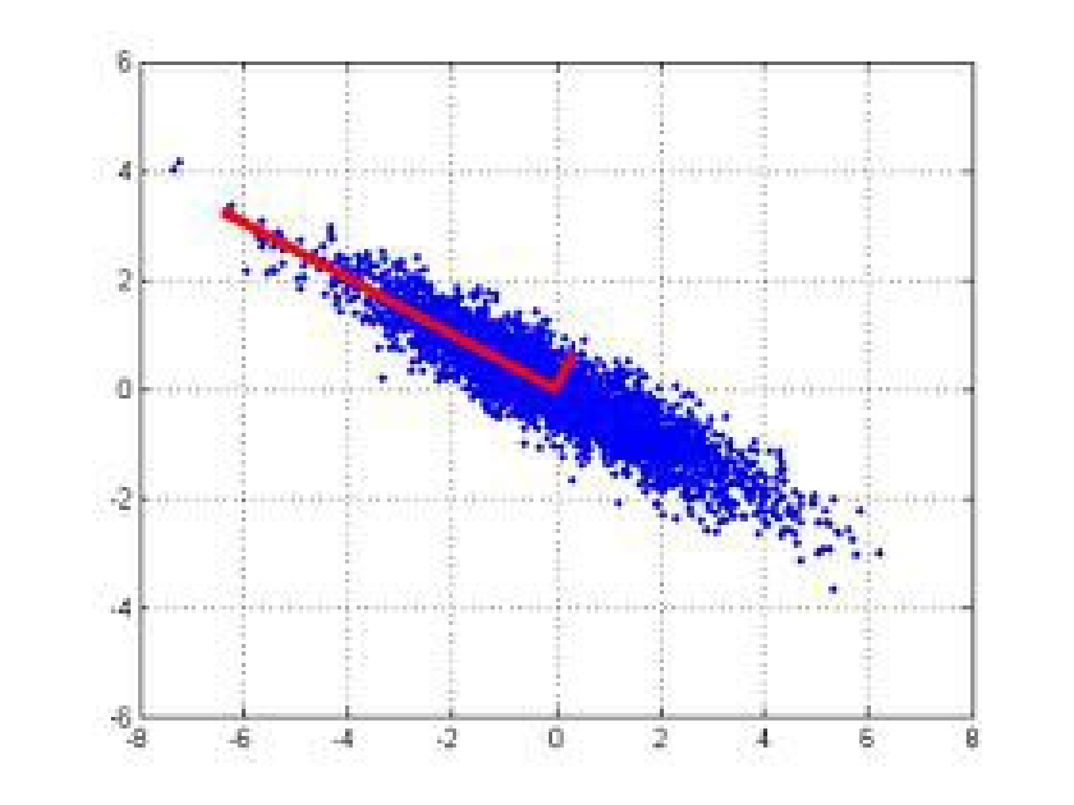


Along this direction:

- Features are most correlated with each other
- Contains the most of the information

2D Gaussian Dataset

Component 2: orthogonal to the first direction & maximized variance



Sample Covariance matrix

- The sample covariance matrix is given by:

$$\Sigma_{jk} = \frac{1}{N}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_j\right)\left(x_k^{(i)} - \mu_k\right)$$

- Since the data matrix is centered, we rewrite as:

$$\sum = \frac{1}{N}\boldsymbol{X}^T\boldsymbol{X}$$

Definition of PCA

- Given K vectors, $\overrightarrow{v_1}, \overrightarrow{v_2}, \dots, \overrightarrow{v_K}$, the projection of a vector $x^{(i)}$to a lower K-dimensional space is

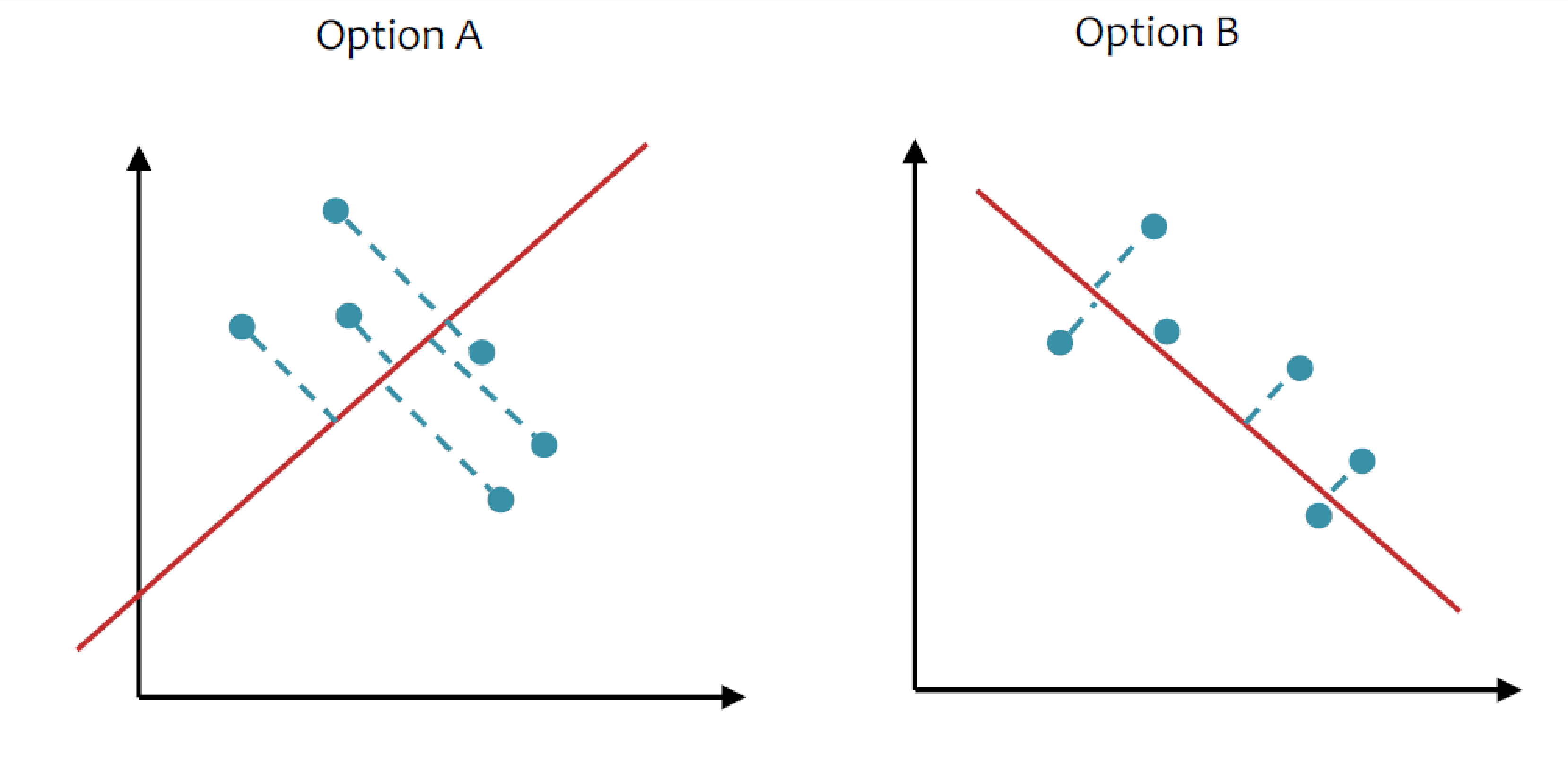
$$\vec{u}^{(i)} = \begin{bmatrix} \overrightarrow{v_1}^T \vec{x}^{(i)} \\ \dots \\ \overrightarrow{v_K}^T \vec{x}^{(i)} \end{bmatrix}$$

- Def: PCA repeatedly chooses a next $\overrightarrow{v_j}$ that minimize the reconstruction error, s.t., $\overrightarrow{v_j}$ is orthogonal to $\overrightarrow{v_1}$,…, $\overrightarrow{v_{j-1}}$

PCA: Maximize the Variance

Quiz: Consider the two projections below

1. Which maximizes the variance?
2. Which minimizes the reconstruction error?



Eigenvectors and Eigenvalues

- For a square matrix **A** ($n \times n$), the vector **v** ($n \times 1$) is an eigenvector iff there exists eigenvalue λ (scalar) such that

$$Ax = \lambda x$$

- **Theorem 1**: The vector that maximizes the variance is the eigenvector of Σ with largest eigenvalue
- **Theorem 2**: The eigenvector of a symmetric matrix are orthogonal to each other
- **Fact 1**: Σ is a symmetric matrix

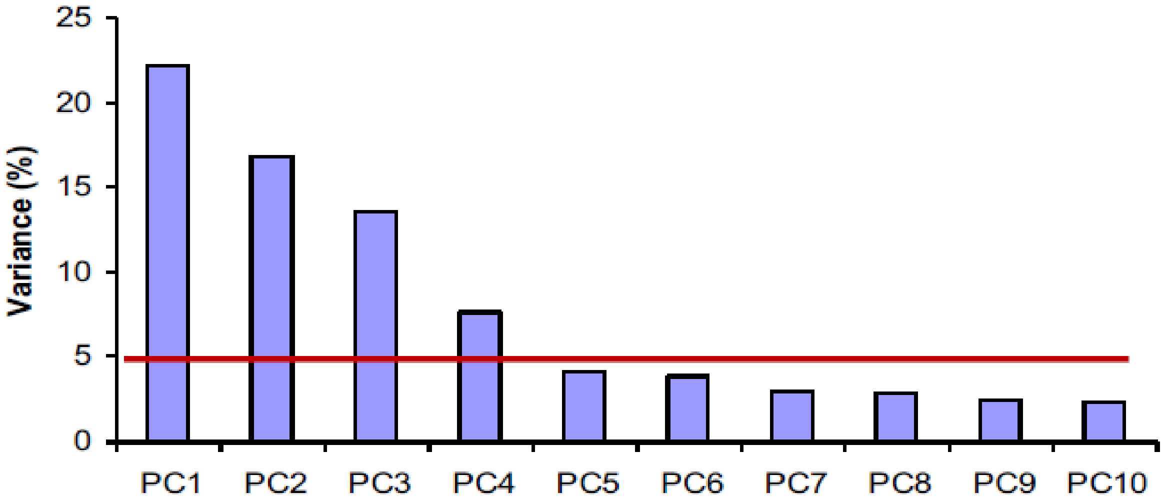
Algorithms for PCA

- Singular Value Decomposition (SVD)
  - Find all the principal components at once
  - Two options:
    - Option A: run SVD on $X^T X$
    - Option B: run SVD on X

How Many PCs?

- For M original dimensions, sample covariance matrix is MxM, and has up to M eigenvectors. So M PCs.
- Where does dimensionality reduction come from?

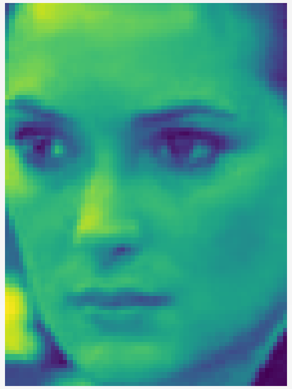
  Can *ignore the components of lesser significance*.



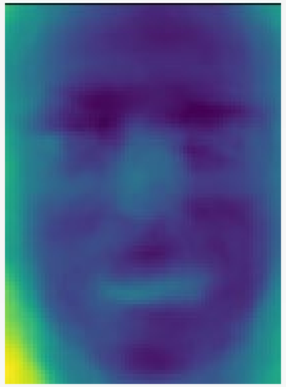
- You do lose some information, but if the eigenvalues are small, you don't lose much
  - M dimensions in original data
  - calculate M eigenvectors and eigenvalues
  - choose only the first D eigenvectors, based on their eigenvalues
  - final data set has only D dimensions

Example: Facial Recognition

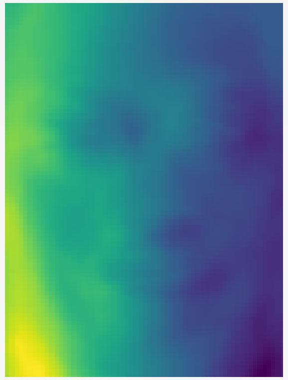
PCA Transformation



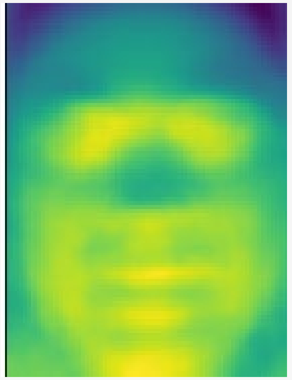
$\approx \boldsymbol{\beta_0} \times$



$+\boldsymbol{\beta_1} \times$



$+\boldsymbol{\beta_2} \times$



$+ \cdots$

β_0, β_1, and so on are the coefficients of the principal components for this data point.