CS 760: Machine Learning
Unsupervised Learning II: Dimensionality Reduction

Kirthi Kandasamy

University of Wisconsin-Madison

March 20, 2023

Announcements

- **HW4** was due today morning
- **HW5** due on Apr 3

High-Dimensional Data

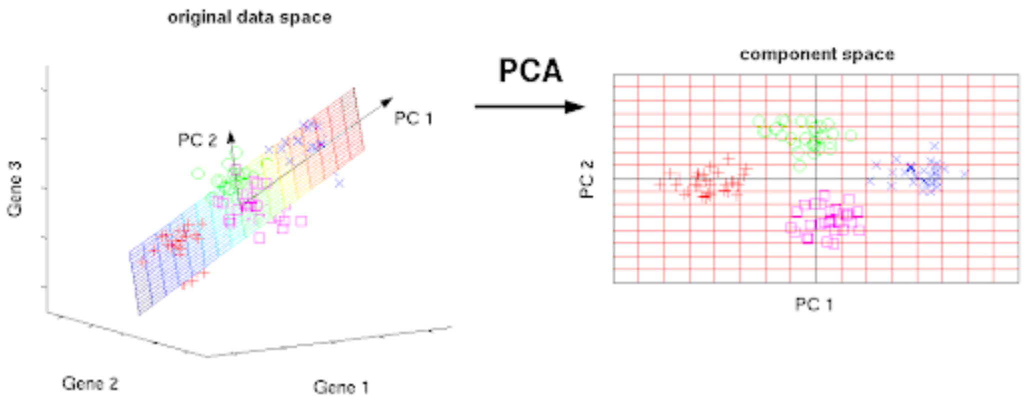
- High-dimensions = lots of features
- Document classification
  - Features per document = thousands of words/unigrams millions of bigrams, contextual information
- **Example**: Surveys - Netflix

  480189 users x 17770 movies

	movie 1	movie 2	movie 3	movie 4	movie 5	movie 6
Tom	5	?	?	1	3	?
George	?	?	3	1	2	5
Susan	4	3	1	?	5	1
Beth	4	3	?	2	4	2

Dealing with Dimensionality

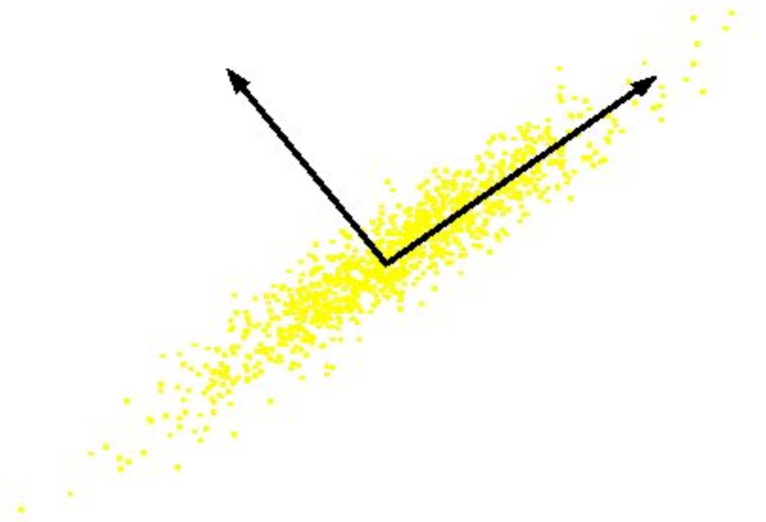
- **PCA, Kernel PCA, ICA**: Powerful unsupervised learning techniques for extracting hidden (potentially lower dimensional) structure from high dimensional datasets.
- Some uses:
  - Visualization
  - More efficient use of resources (e.g., time, memory, communication)
  - Noise removal (improving data quality)



PCA Intuition

- The dimension of the ambient space (ie, R^d) might be much higher than the **intrinsic** data dimension

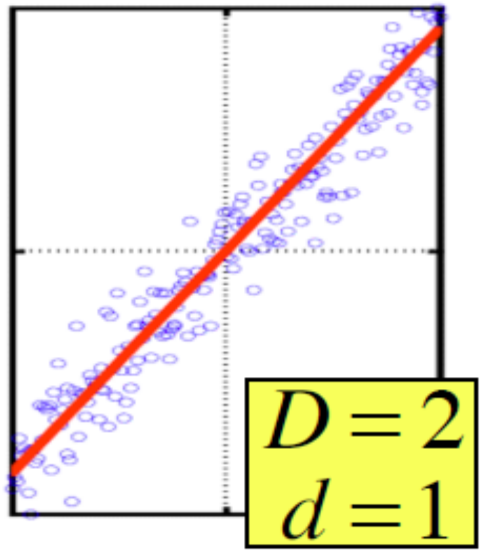
- Can we transform the features so that we store each point using fewer coordinates and still preserve most of the information?

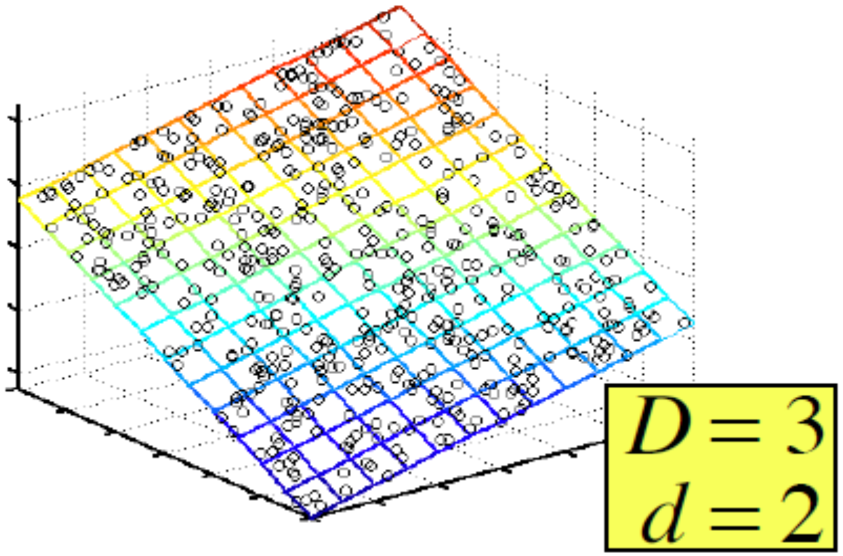


- PCA: Projects the data into a lower dimensional subspace so that the variance of the projected data is maximized.

PCA Intuition

- Some more visualizations





- In case where data lies on or near a low d-dimensional linear subspace, axes of this subspace are an effective representation of the data.

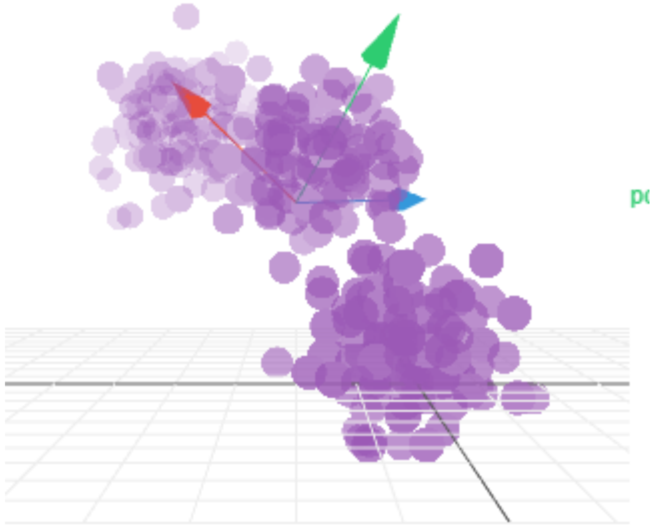
PCA: Principal Components

- **Principal Components (PCs)** are orthogonal directions that capture most of the variance in the data.
  - First PC – direction of greatest variability in data.
  - Projection of data points along first PC stores most of the information in the data most along any one direction

PCA Overview

- How does dimensionality reduction work? From *d* dimensions to *r* dimensions:
  - Get
  - Orthogonal!

$$v_1, v_2, \ldots, v_r \in \mathbb{R}^d$$



Victor Powell

PCA First Step

- First component,

$$v_1 = \arg\max_{\|v\|=1} \sum_{i=1}^{n} \langle v, x_i \rangle^2$$

- Same as getting

$$v_1 = \arg\max_{\|v\|=1} \|Xv\|^2$$

PCA Recursion

- Once we have *k-1* components, next?

$$\hat{X}_k = X - \sum_{i=1}^{k-1} X v_i v_i^T$$

Deflation

- Then do the same thing

$$v_k = \arg\max_{\|v\|=1} \|\hat{X}_k w\|^2$$

PCA Interpretations

- The v's are eigenvectors of $X^T X$ **(Gram matrix)**

- $X^T X$ is the sample covariance matrix!
  - When data has 0 mean.
  - I.e., *PCA is eigendecomposition of sample covariance*

- Finding $$v_1 = \arg\max_{\|v\|=1} \|Xv\|^2$$
  - First eigenvector of the covariance matrix!
  - Or, equivalently, first right singular vector of the data matrix X.

PCA Interpretations: Equivalence

- Interpretation 1.

Maximum variance direction

$$\sum_{i=1}^{n} (\mathbf{v}^T \mathbf{x}_i)^2 = \mathbf{v}^T \mathbf{X} \mathbf{X}^T \mathbf{v}$$

- Interpretation 2.

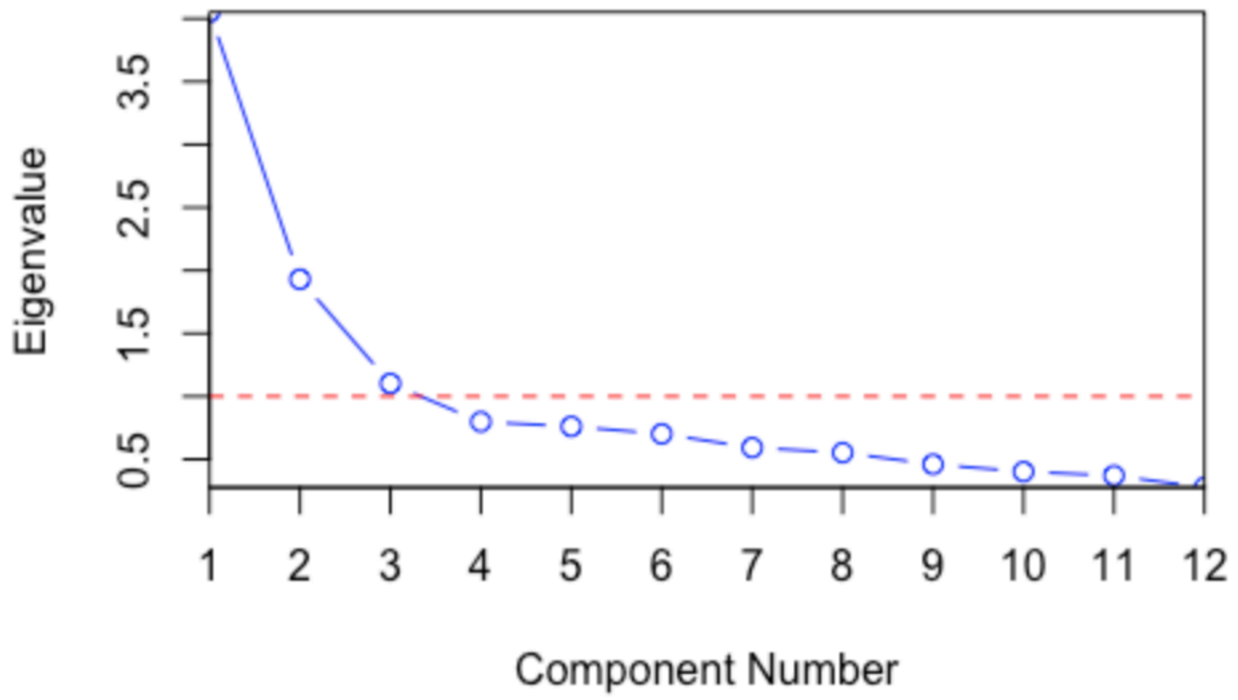
Minimum reconstruction error

$$\sum_{i=1}^{n} \|\mathbf{x}_i - (\mathbf{v}^T \mathbf{x}_i)\mathbf{v}\|^2$$

- Do at home (show that these two are equivalent)

How to choose r?

- Only keep data projections onto principal components with **large** eigenvalues of X^TX (singular values of X)
- Look for "knee point"



Application: Image Compression

- Start with image; divide into 12x12 patches
  - I.E., 144-D vector
  - **Original image:**

Application: Image Compression

- Project to 6D,

Compressed

Original

Thanks Everyone!

Some of the slides in these lectures have been adapted/borrowed from materials developed by Mark Craven, David Page, Jude Shavlik, Tom Mitchell, Nina Balcan, Elad Hazan, Tom Dietterich, Pedro Domingos, Jerry Zhu, Yingyu Liang, Volodymyr Kuleshov