Unsupervised Learning
Principal Component Analysis

CMSC 422

Slides adapted from Prof. CARPUAT

Slides credit: Maria-Florina Balcan

Unsupervised Learning

- Discovering hidden structure in data

- What algorithms do we know for unsupervised learning?
  - K-Means Clustering

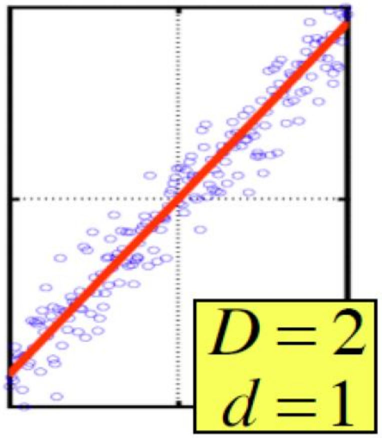
- Today: how can we learn better representations of our data points?

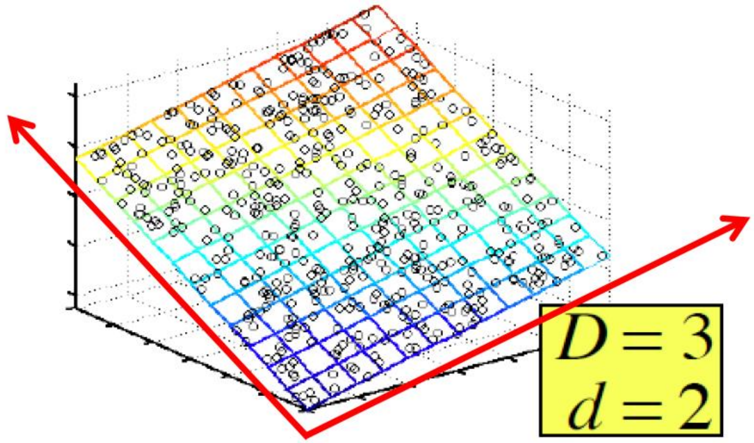
Dimensionality Reduction

- Goal: extract hidden lower-dimensional structure from high dimensional datasets

- Why?
  - To visualize data more easily
  - To remove noise in data
  - To lower resource requirements for storing/processing data
  - To improve classification/clustering

Low dimensional data embedded in high dimensional spaces





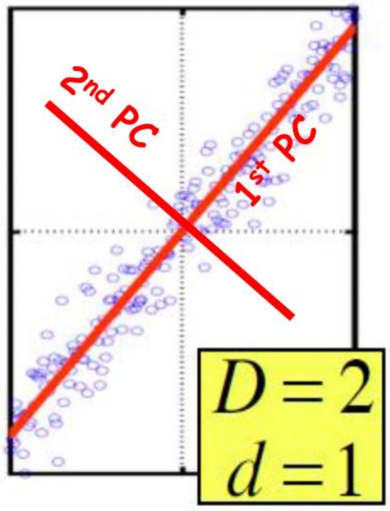
Examples of data points in D dimensional space that can be effectively represented in a d-dimensional subspace (d < D)

Principal Component Analysis

- Goal: Find a **projection** of the data onto directions that **maximize variance** of the original data set
  - Intuition: those are directions in which most information is encoded

- Definition: **Principal Components** are orthogonal directions that capture most of the variance in the data

PCA: finding principal components



- 1^{st} PC
  - Projection of data points along 1^{st} PC discriminates data most along any one direction
- 2^{nd} PC
  - next orthogonal direction of greatest variability
- And so on…

PCA: notation

- Data points
  - Represented by matrix X of size DxN
  - Let's assume data is centered
- Principal components are d vectors: $v_1, v_2, \ldots v_d$
  - $v_i . v_j = 0, i \neq j$ and $v_i . v_i = 1$
- The sample variance data projected on vector v is $\frac{1}{n}\sum_{i=1}^{n}(v^T x_i)^2 = \frac{1}{n} v^T X X^T v$

PCA formally

- Finding vector that maximizes sample variance of projected data:

$$argmax_v \, v^T X X^T v \text{ such that } v^T v = 1$$

- A constrained optimization problem
  - Lagrangian folds constraint into objective: $argmax_v \, v^T X X^T v \; - \; \lambda v^T v$
  - Solutions are vectors v such that $X X^T v = \; \lambda v$
    - i.e. eigenvectors of $X X^T$ (sample covariance matrix)

PCA formally

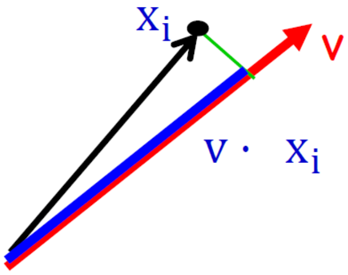
- The eigenvalue λ denotes the amount of variability captured along dimension v
  - Sample variance of projection $v^T XX^T v = \lambda$

- If we rank eigenvalues from large to small
  - The 1st PC is the eigenvector of XX^T associated with largest eigenvalue
  - The 2nd PC is the eigenvector of XX^T associated with 2nd largest eigenvalue
  - …

Alternative interpretation of PCA

- PCA finds vectors v such that projection on to these vectors minimizes reconstruction error

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



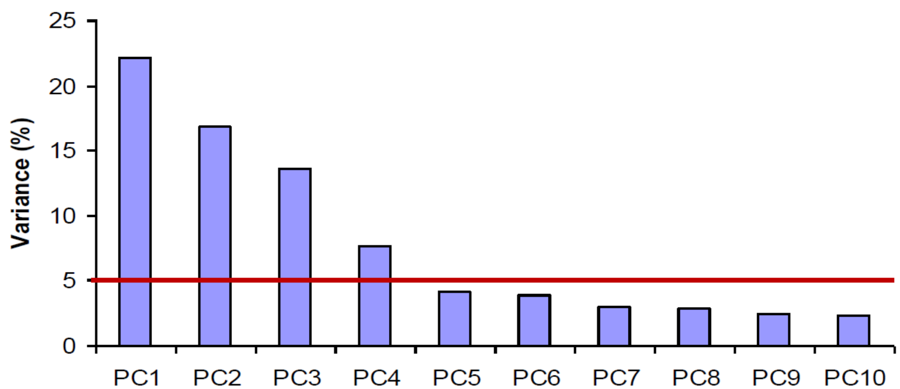
Resulting PCA algorithm

Algorithm 36 PCA($\mathbf{D}$, K)

1: $\boldsymbol{\mu} \leftarrow \textsc{mean}(\mathbf{X})$ // compute data mean for centering
2: $\mathbf{D} \leftarrow \left(\mathbf{X} - \boldsymbol{\mu}\mathbf{1}^\top\right)^\top \left(\mathbf{X} - \boldsymbol{\mu}\mathbf{1}^\top\right)$ // compute covariance, $\mathbf{1}$ is a vector of ones
3: $\{\lambda_k, \boldsymbol{u}_k\} \leftarrow$ top K eigenvalues/eigenvectors of $\mathbf{D}$
4: **return** $(\mathbf{X} - \boldsymbol{\mu}\mathbf{1})\,\mathbf{U}$ // project data using $\mathbf{U}$

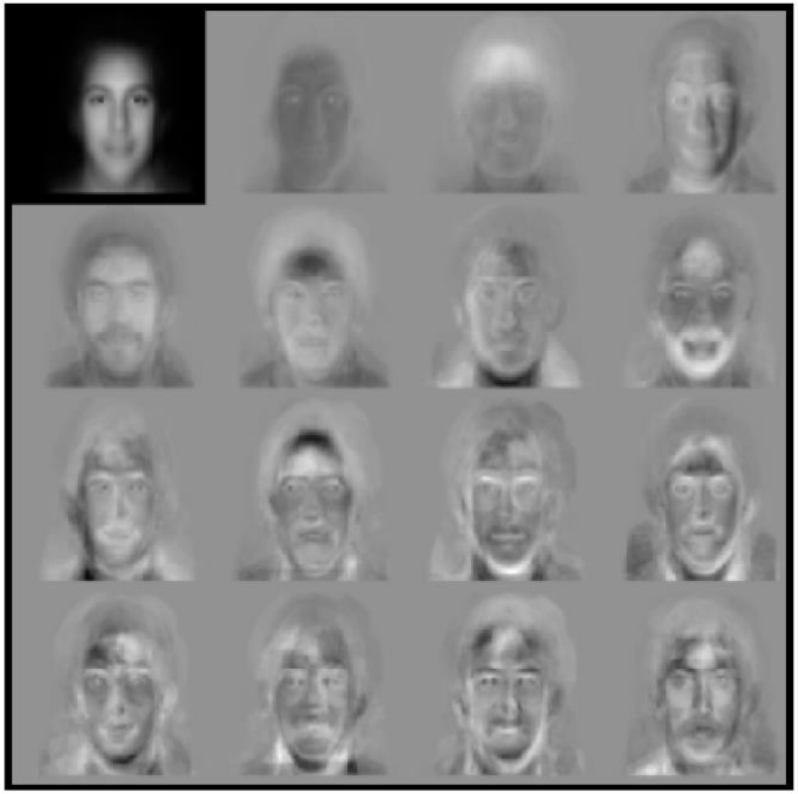
How to choose the hyperparameter K?

- i.e. the number of dimensions



- We can ignore the components of smaller significance

An example: Eigenfaces



Eigenfaces from 7562 images:

top left image is linear combination of rest.

Sirovich & Kirby (1987)
Turk & Pentland (1991)

PCA pros and cons

- Pros
  - Eigenvector method
  - No tuning of the parameters
  - No local optima

- Cons
  - Only based on covariance (2^{nd} order statistics)
  - Limited to linear projections

What you should know

- Principal Components Analysis
  - Goal: Find a **projection** of the data onto directions that **maximize variance** of the original data set
  - PCA **optimization objectives**and resulting **algorithm**
  - Why this is useful!