Dimension Reduction

CS 760@UW-Madison

Goals for the lecture

you should understand the following concepts

- dimension reduction
- principal component analysis: definition and formulation
- two interpretations
- strength and weakness

Introduction

Big & High-Dimensional Data

- High-Dimensions = Lot of Features

Document classification

Features per document =

thousands of words/unigrams

millions of bigrams, contextual

information

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

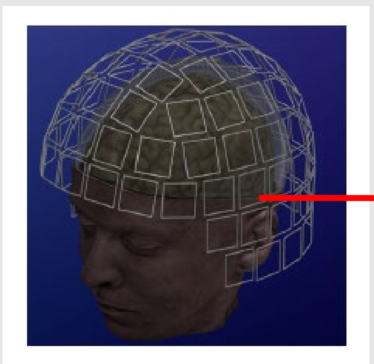
Big & High-Dimensional Data

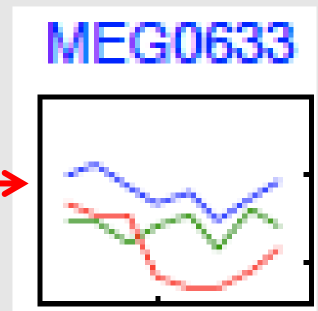
- High-Dimensions = Lot of Features

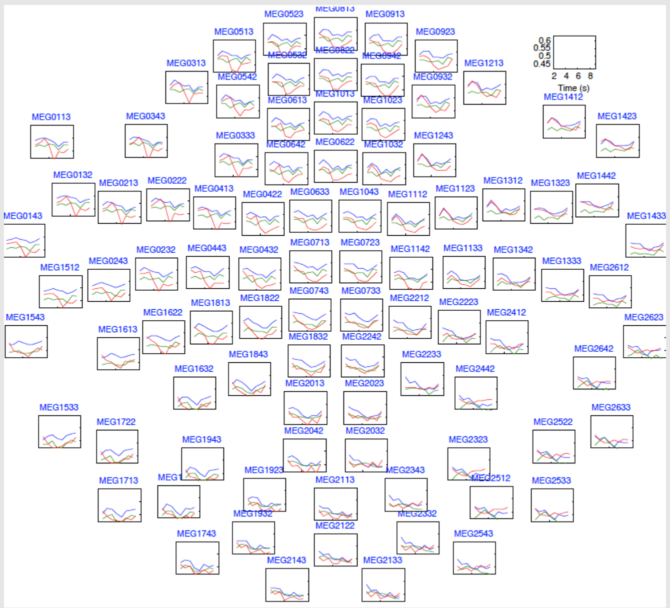
MEG Brain Imaging

120 locations x 500 time points

x 20 objects







Or any high-dimensional image data

- Big & High-Dimensional Data.
- Useful to learn lower dimensional representations of the data.

Learning Representations

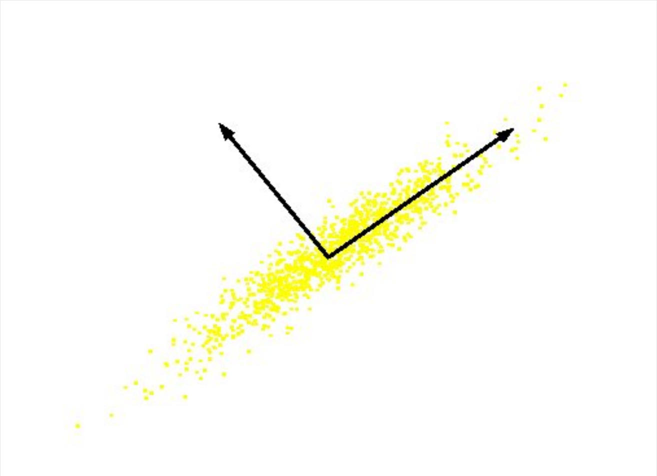
PCA, Kernel PCA, ICA: Powerful unsupervised learning techniques for extracting hidden (potentially lower dimensional) structure from high dimensional datasets.

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 machine learning algorithms

Principal Component Analysis (PCA)

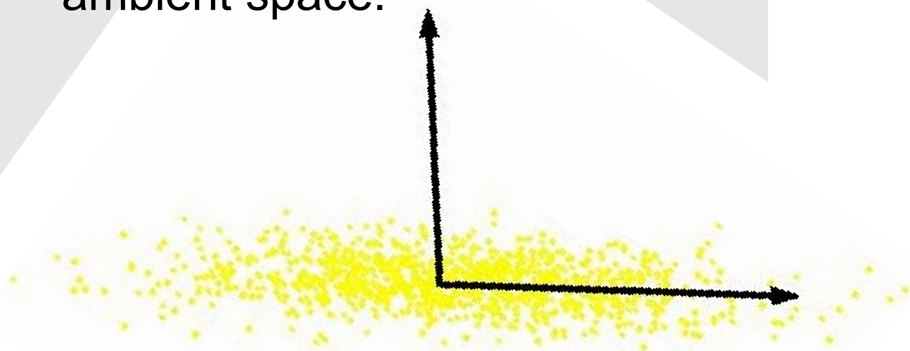
What is PCA: Unsupervised technique for extracting variance structure from high dimensional datasets.



- PCA is an orthogonal projection or transformation of the data into a (possibly lower dimensional) subspace so that the variance of the projected data is maximized.

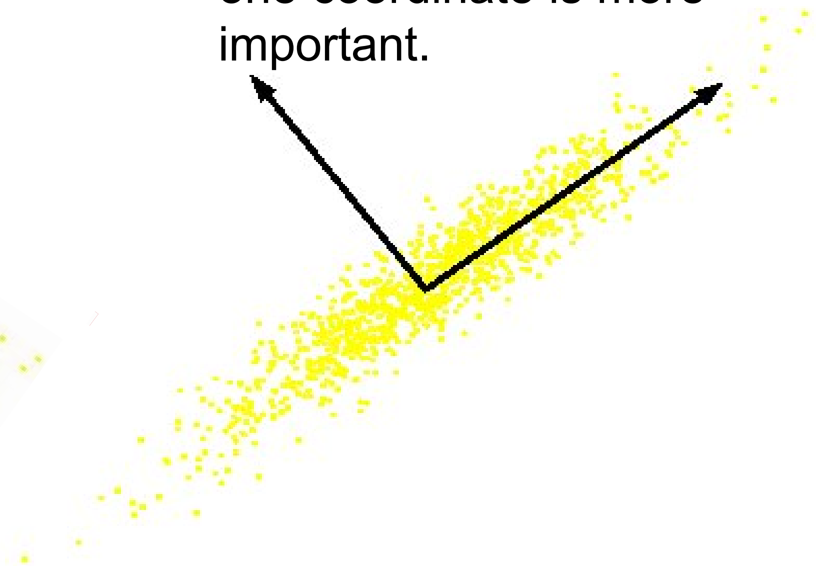
Principal Component Analysis (PCA)

Intrinsically lower dimensional than the dimension of the ambient space.



Only one relevant feature

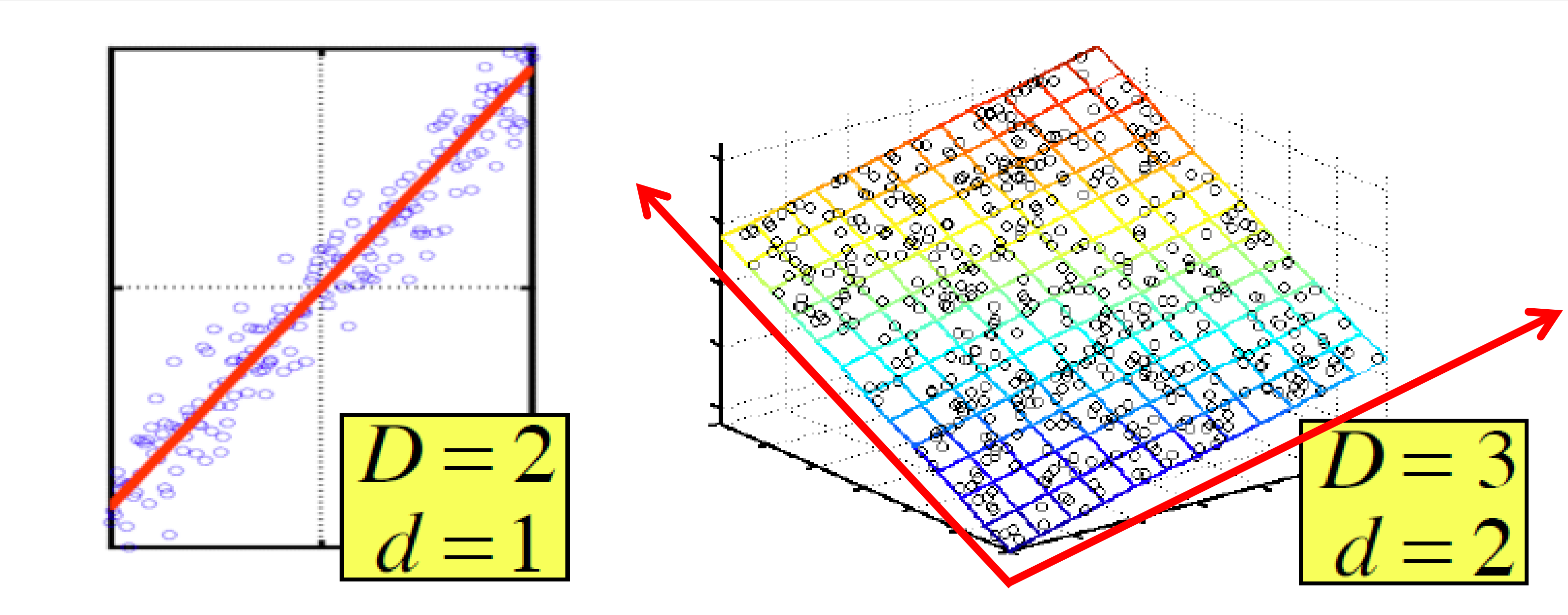
If we rotate data, again only one coordinate is more important.



Both features are relevant

Question: Can we transform the features so that we only need to preserve one latent feature?

Principal Component Analysis (PCA)



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.

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

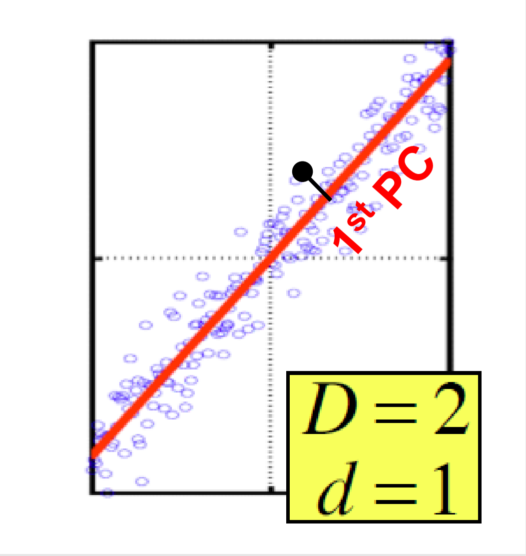
Formulation

Principal Component Analysis (PCA)

Principal Components (PC) are orthogonal directions that capture most of the variance in the data.

- First PC – direction of greatest variability in data.

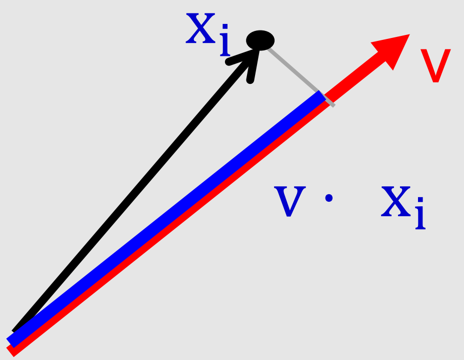
- Projection of data points along first PC discriminates data most along any one direction (pts are the most spread out when we project the data on that direction compared to any other directions).



Principal Component Analysis (PCA)

Principal Components (PC) are orthogonal directions that capture most of the variance in the data.

Quick reminder:



||v||=1, Point $\mathrm{x_i}$ (D-dimensional vector)

Projection of $\mathrm{x_i}$ onto v is $\mathrm{v} \cdot \mathrm{x_i}$

Let μ be the mean of data points, then PCA for the first PC is:
max $\sum_i (\mathrm{v} \cdot \mathrm{x_i} - \mathrm{v} \cdot \mu)^2$ s.t. ||v||=1

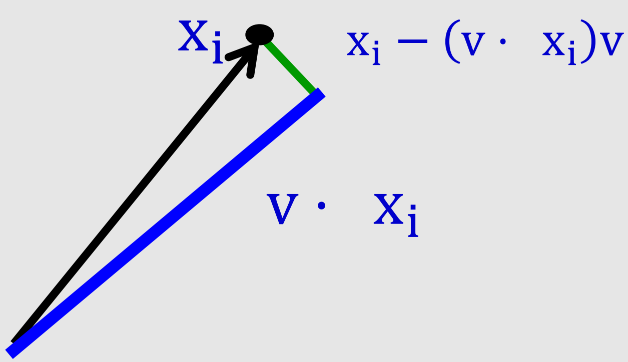
Usually, we first centralize the data points by subtracting their mean, then $\mu = 0$, and the optimization is simplified to:
max $\sum_i (\mathrm{v} \cdot \mathrm{x_i})^2$ s.t. ||v||=1

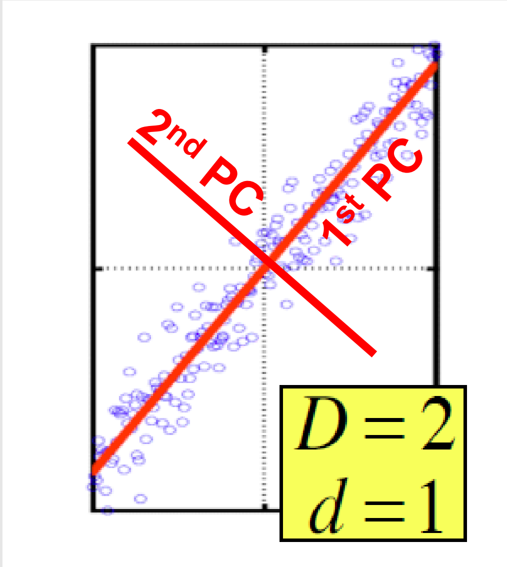
Let $\mathrm{X} = [\mathrm{x_1}, \mathrm{x_2}, \ldots, \mathrm{x_n}]$ (columns are the datapoints). Then:
max $\mathrm{v}^t X X^t \mathrm{v}$ s.t. ||v||=1

Principal Component Analysis (PCA)

Principal Components (PC) are orthogonal directions that capture most of the variance in the data.

- 1^{st} PC – direction of greatest variability in data.





- 2^{nd} PC – Next orthogonal (uncorrelated) direction of greatest variability

  (remove all variability in first direction, then find next direction of greatest variability)

- And so on …

Two Interpretations

Two Interpretations

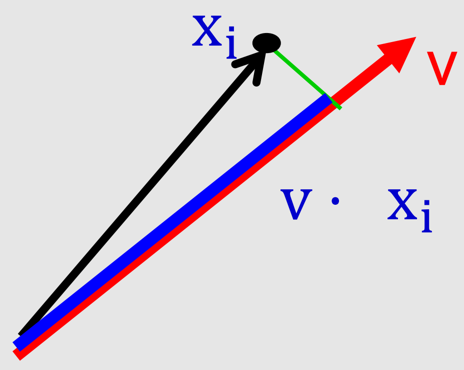
Consider only the first component.

Maximum Variance Direction: 1st PC is a direction v such that projection on to this direction has maximum variance. Assume data mean=0:

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

Minimum Reconstruction Error: 1st PC is a direction v such that projection on to this direction yields minimum MSE reconstruction

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



Why? Pythagorean Theorem

E.g., for the first component.

Maximum Variance Direction: 1st PC is a direction v such that projection on to this direction has maximum variance

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

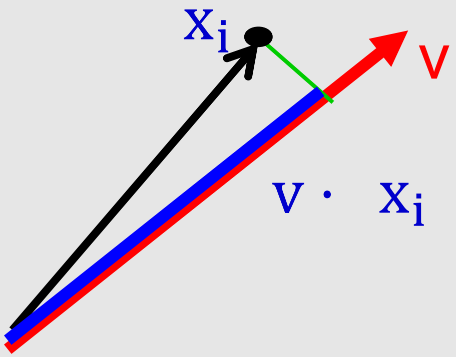
$$\sum_{i=1}^{n} \|\mathbf{x}_i - (\mathbf{v}^T \mathbf{x}_i)\mathbf{v}\|^2$$

Minimum Reconstruction Error: 1st PC is a direction v such that projection on to this direction yields minimum MSE reconstruction

blue^2 + green^2 = black^2

black^2 is fixed (it's just the data)

So, maximizing blue^2 is equivalent to minimizing green^2



Computation

Principal Component Analysis (PCA)

Let $v_1, v_2, \ldots, v_d$ denote the d principal components.

$v_i \cdot v_j = 0, i \neq j$ and $v_i \cdot v_i = 1,$

Assume data is centered (we extracted the sample mean).

Let $X = [x_1, x_2, \ldots, x_n]$ (columns are the datapoints)

First PC: find vector that maximizes sample variance of projected data

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

$$\max_{\mathbf{v}} \quad \mathbf{v}^T \mathbf{X} \mathbf{X}^T \mathbf{v} \quad \text{s.t.} \quad \mathbf{v}^T \mathbf{v} = 1$$

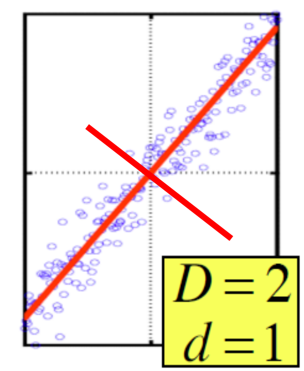
Lagrangian: $\max_{\mathbf{v}} \mathbf{v}^T \mathbf{X} \mathbf{X}^T \mathbf{v} - \lambda \mathbf{v}^T \mathbf{v}$

Wrap constraints into the objective function

$\partial / \partial \mathbf{v} = 0$ $\quad (\mathbf{X}\mathbf{X}^T - \lambda \mathbf{I})\mathbf{v} = 0$ $\quad \Rightarrow (\mathbf{X}\mathbf{X}^T)\mathbf{v} = \lambda \mathbf{v}$

Principal Component Analysis (PCA)

$(X X^T)v = \lambda v$, so v (the first PC) is the eigenvector of sample correlation/covariance matrix $X X^T$



The variance of projection $v^T X X^T v = \lambda v^T v = \lambda$

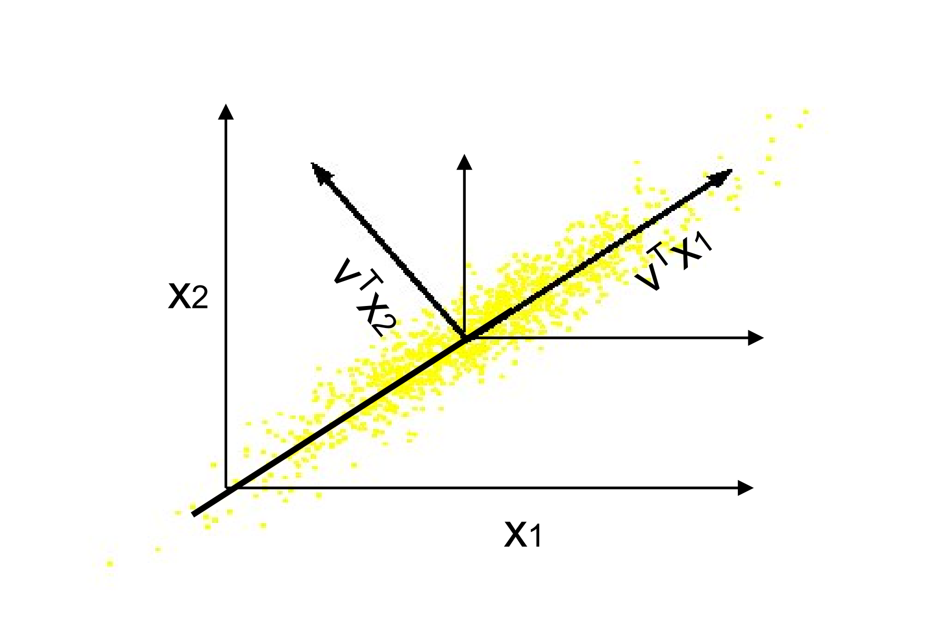
Thus, the eigenvalue λ denotes the amount of variability captured along that dimension (aka amount of energy along that dimension).

Eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \cdots$

- The 1st PC v_1 is the first eigenvector of the sample covariance matrix $X X^T$ associated with the largest eigenvalue. The variance of the projected data on it is just the largest eigenvalue.
- The 2nd PC v_2 is the second eigenvector of the sample covariance matrix $X X^T$ associated with the second largest eigenvalue. The variance of the projected data on it is just the second largest eigenvalue.
- And so on …

Principal Component Analysis (PCA)

- So, the new axes (the PCs) are the eigenvectors of the matrix of sample correlations $X\ X^T$ of the centralized data.



- Geometrically: centering followed by rotation (all are linear transformation)

Key computation: eigendecomposition of XX^T (closely related to SVD of X).

Dimensionality Reduction using PCA

The eigenvalue λ denotes the amount of variability captured along that dimension (aka amount of energy along that dimension).

Zero eigenvalues indicate no variability along those directions => data lies exactly on a linear subspace

Only keep data projections onto top principal components, say $v_1, \ldots, v_d$

Original representation

Data point

$$x_i = (x_i^1, \ldots, x_i^D)$$

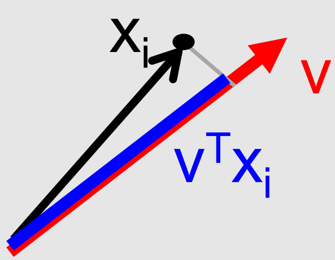
D-dimensional vector

Transformed representation

projection

$$(v_1 \cdot x_i, \ldots, v_d \cdot x_i)$$

d-dimensional vector



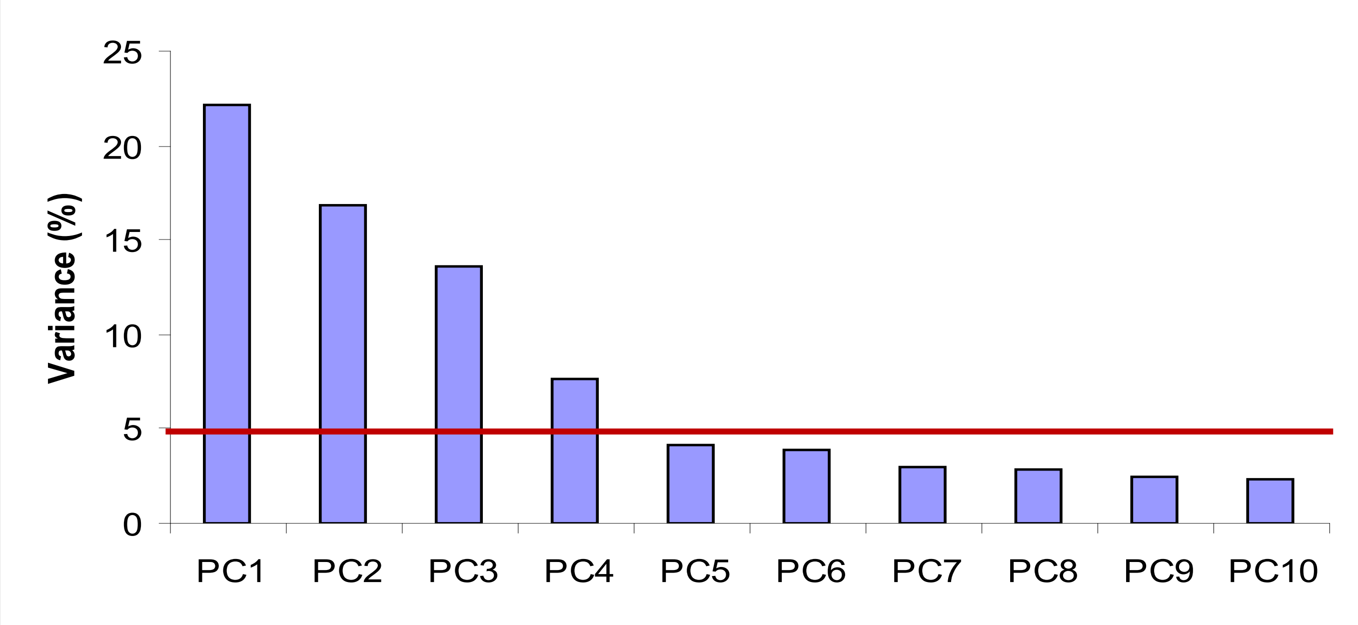
Application Examples

Dimension Reduction using PCA

In high-dimensional problems, data sometimes lies near a linear subspace, as noise introduces small variability

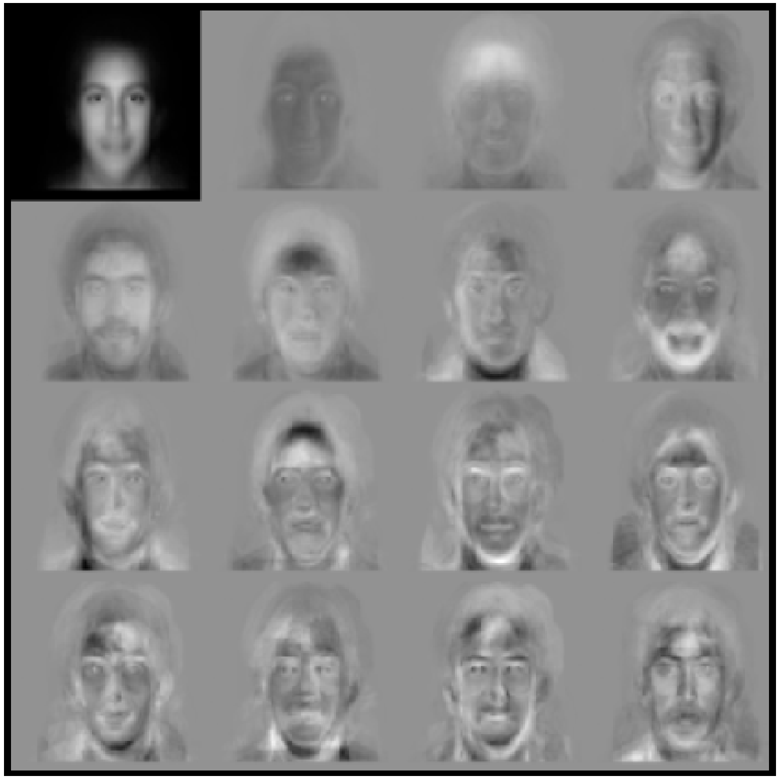
Only keep data projections onto principal components with **large** eigenvalues

Can *ignore* the components of smaller significance.



Might lose some info, but if eigenvalues are small, do not lose much

Example: faces



Eigenfaces from 7562 images:

top left image is linear combination of rest.

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

Can represent a face image using just 15 numbers!

PCA Discussion

Strengths

Eigenvector method

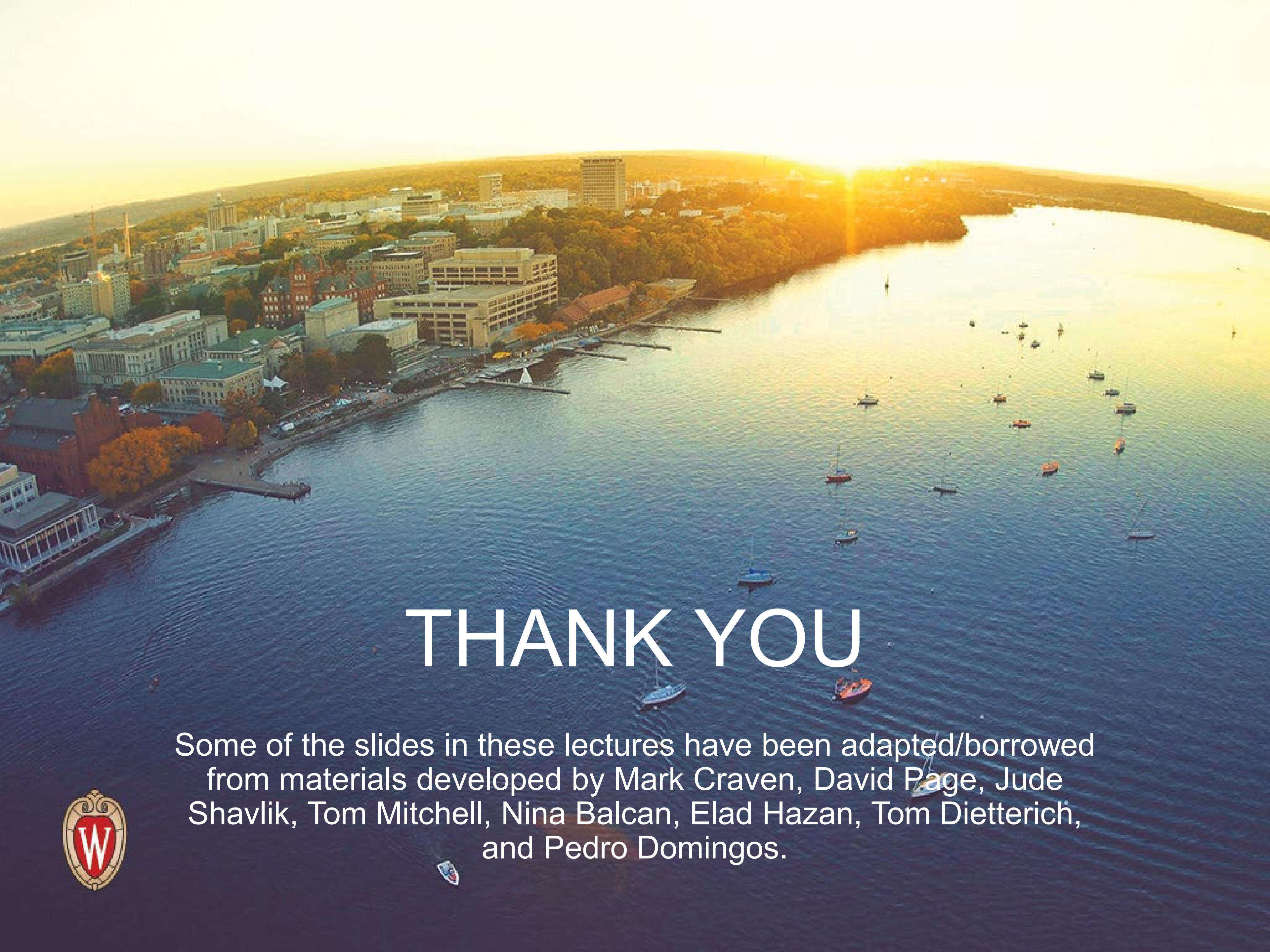
No tuning of the parameters

No local optima

Weaknesses

Limited to second order statistics

Limited to linear projections



THANK YOU

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, and Pedro Domingos.