Applied Mathematics
007 – Dimensionality Reduction

Christian Wallraven
Cognitive Systems Lab
Departments of Artificial Intelligence, Brain and Cognitive Engineering
christian.wallraven+AMF2023@gmail.com
http://cogsys.korea.ac.kr

Discovering “factors” / dimensions

CognitiveSystems

- In the previous lectures we focused on finding **clusters**
  - dimensionality fixed / known
  - clusters are structures in a known parameter space
- What about when the dimensions are not known?
- We would like to observe underlying (so-called ‘latent’) factors in data
  - factors influencing perception of attractiveness in faces
  - ego, personality, intelligence in psychology
  - topics in news articles
  - transcription factors in genomics

Discovering “factors” / dimensions

- Another problem is if there are too many observations and/or dimensions in our data, this makes it hard to
  - classify
  - visualize relationships
- Many dimensions of our data might actually only contain noise
  - we need to “reduce” data to a smaller set of factors / dimensions
- The goal is to embed the data in a lower-dimensional space without losing much information
  - In this space, we can create more effective data analyses: classification, clustering, pattern recognition
- Combinations of observed variables may be more effective bases for insights, even if physical meaning is obscure

Basic Idea

CognitiveSystems

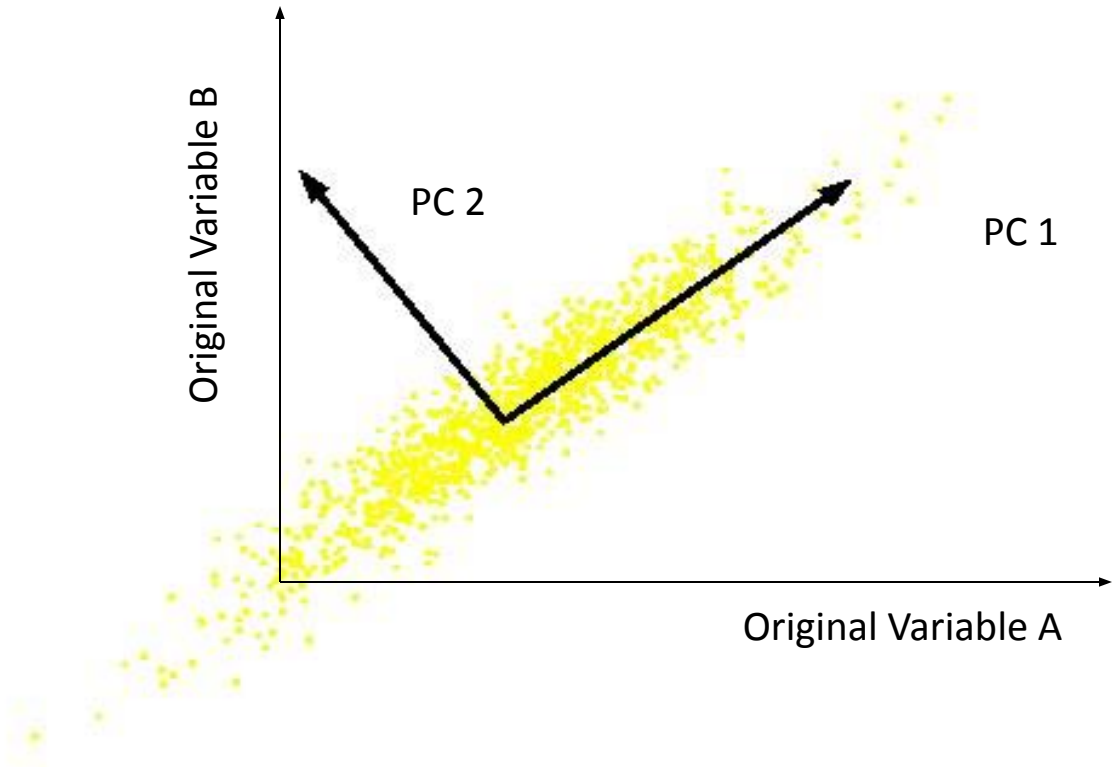
- Identify regions of high **variance** in data
  - yields insight into where items can be best **discriminated,** or into underlying factors
- If two items or dimensions are highly correlated or dependent
  - They are likely to represent highly related phenomena
  - If they tell us about the same underlying variance, we combine them
    - Occam's razor or parsimonious representations
    - Reduces noise
- Fuse related variables, focus on uncorrelated / independent ones
- Goal: Create a smaller set of variables that explain as much of the variance in the original data as possible
- These variables are called "factors" (Factor Analysis), or "principal components" (Principal Component Analysis), or "independent components" (Independent Component Analysis)

Principal Component Analysis

CognitiveSystems

- Also known as the Karhunen-Loeve transform
- Most common form of factor analysis and dimensionality reduction
- The new factors/dimensions uncovered by PCA are called principal components and they
  - Are **linear** combinations of the original ones
  - Are uncorrelated with one another, i.e., they are orthogonal in the original parameter space
  - Can be sorted according to the variance they capture in the original data

Intuitive understanding



- Orthogonal directions along directions of greatest variance in data

Principal Components

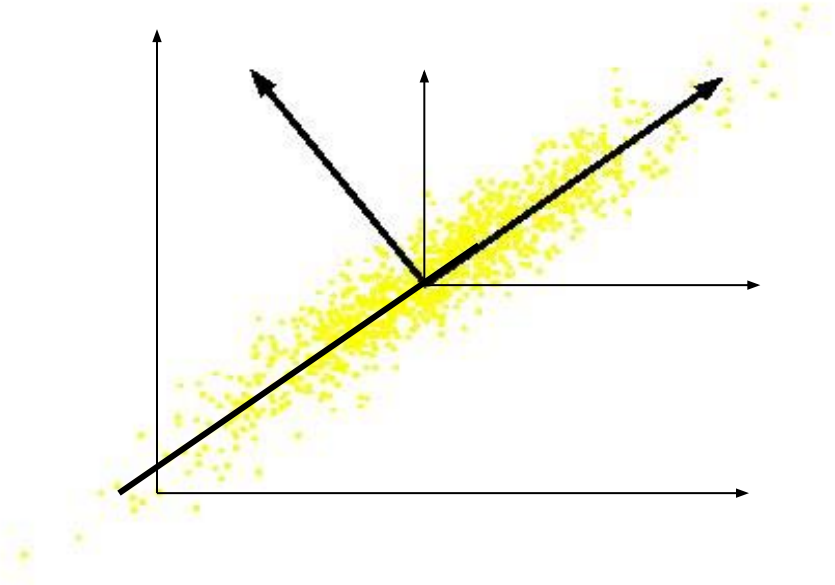
- First principal component is the direction of greatest variability (covariance) in the data
- Second is the next orthogonal (uncorrelated) direction of greatest variability
  - So first remove all the variability along the first component, and then find the next direction of greatest variability
- And so on …
- This is – in principle – an iterative algorithm

Principal Components Analysis (PCA)

- Principle
  - Linear projection method to reduce the number of parameters
  - Transfer a set of correlated variables into a new set of uncorrelated variables
  - Map the data into a space of lower dimensionality
  - Form of unsupervised learning
- Properties
  - It can be viewed as a rotation of the existing axes to new positions in the space defined by original variables
  - New axes are orthogonal and represent the directions with maximum variability

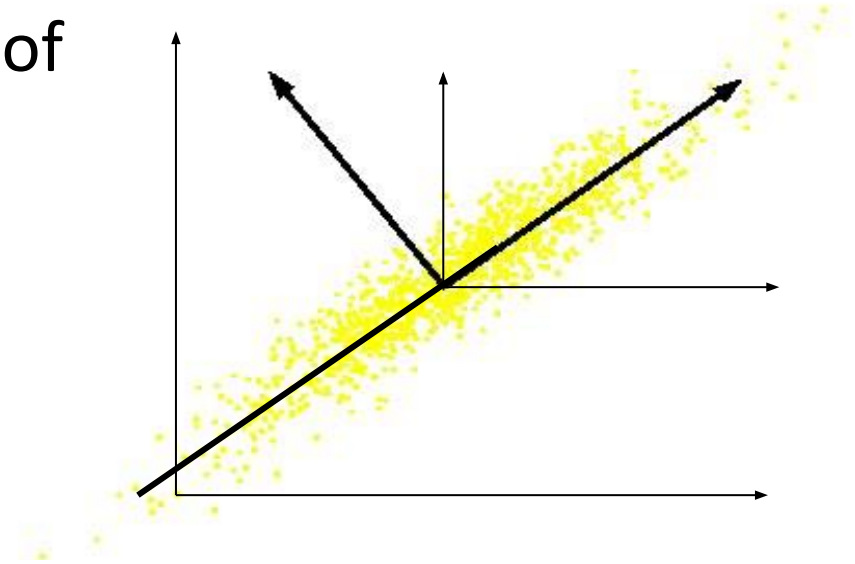
Computing the Components

- Data points are vectors in a multidimensional space
- First we center the original axis system at the mean of all data
- Projection of vector **x** onto an axis (dimension) **u** is **u**.**x**
- Direction of greatest variability is that in which the average square of the projection is greatest
  - choose **u** such that $E((\mathbf{u}.\mathbf{x})^2)$ over all **x** is maximized
  - This direction of **u** is the direction of the first Principal Component



Computing the Components

- The new axes (Principal Components, PCs) are the eigenvectors of the matrix of correlations of the original variables
- Geometrically: centering followed by rotation
  - Linear transformation



PCs, Variance and Least-Squares

CognitiveSystems

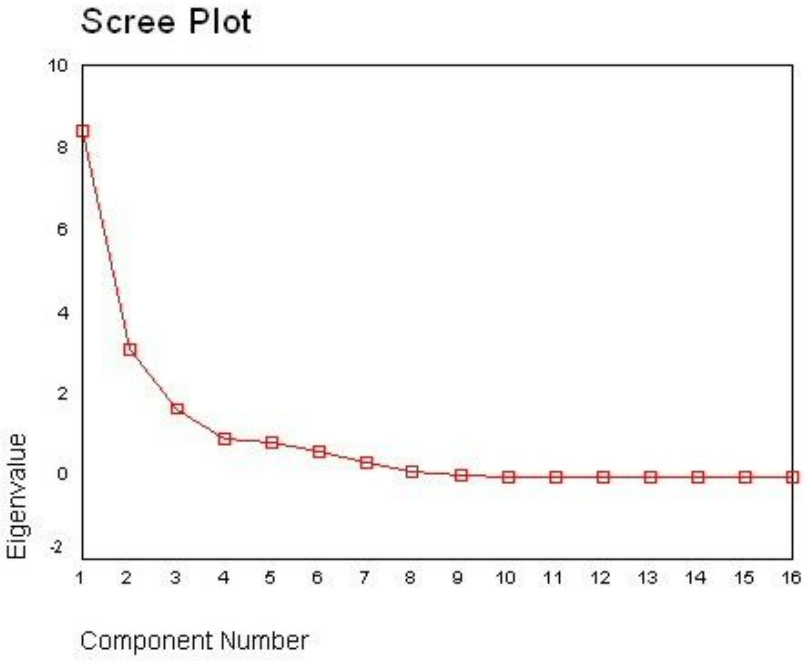
- The first PC retains the greatest amount of variation in the sample
- The k-th PC retains the k-th greatest fraction of the variation in the sample
- The k-th largest eigenvalue of the correlation matrix C is the variance in the sample along the k-th PC
- There are many different interpretations of PCA:
  - In least-squares terms, e.g.: PCs are a series of linear least squares fits to a sample, each orthogonal to all previous ones

How Many PCs?

- For n original dimensions, correlation matrix is nxn, and has up to n eigenvectors.
- Since the decomposition is unique, we get n PCs.
- How can we use PCA for dimensionality reduction?

How Many PCs?

- As each eigenvalue captures the amount of variance along that dimension, simply order dimensions from highest to lowest eigenvalue.
- Then sum up contributions of eigenvalues
- Choose m < n and variance threshold t, such that $\sum_{i=1}^{m} \lambda_i < t$
- If the eigenvalues are small, you don't lose much
- Other option: identify m from elbow in scree plot



Demos of PCA

CognitiveSystems

- http://setosa.io/ev/principal-component-analysis/

- Script

PCA applications - Eigenfaces

- Classic paper by Turk and Pentland (1991), that ignited the interest in PCA as an image processing tool
- Eigenfaces are the eigenvectors of the covariance matrix of the probability distribution of the vector space of human faces
- Eigenfaces are the 'standardized face ingredients' derived from the statistical analysis of many pictures of human faces
- A human face may be considered to be a **combination** of these standard faces

Cited by 14455

A theory for multiresolution signal decomposition: The wavelet representation
SG Mallat – Pattern Analysis and Machine Intelligence, IEEE ..., 1989

Cited by 13235

A computational approach to edge detection
J Canny – Pattern Analysis and Machine Intelligence, IEEE ..., 1986

Cited by 12841

Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images
Geman and Geman - Pattern Analysis and Machine ..., 1984

Cited by 12801 + 4129 (Object recognition from local scale-invariant features)

Distinctive image features from scale-invariant keypoints
DG Lowe - International journal of computer vision, 2004

Cited by 12251

Snakes: Active contour models
M Kass, A Witkin, Demetri Terzopoulos - International journal of computer ..., 1988

Cited by 9358 + 3206 (Face Recognition using Eigenfaces)

Eigenfaces for Recognition
Turk and Pentland, Journal of cognitive neuroscience Vol. 3, No. 1, Pages 71-86, 1991 (9358 citations)

PCA applications - Eigenfaces

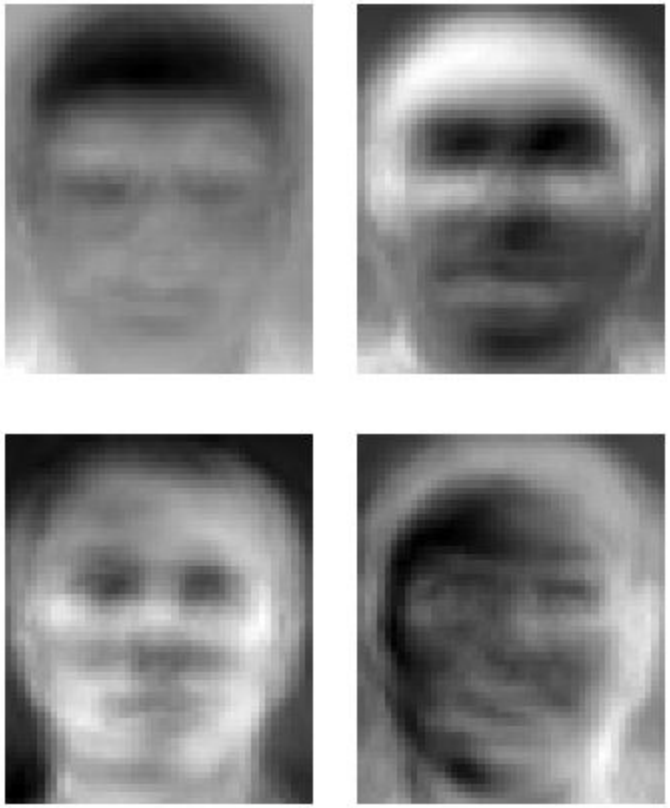
CognitiveSystems

- To generate a set of eigenfaces:

- Large set of digitized images of human faces is taken under the same lighting conditions.
- The images are normalized to line up the eyes and mouths.
- The eigenvectors of the covariance matrix of the statistical distribution of face image vectors are then extracted.
- These eigenvectors are called eigenfaces.

PCA applications - Eigenfaces

- The principal eigenface looks like a bland androgynous average human face



http://en.wikipedia.org/wiki/Image:Eigenfaces.png

Eigenfaces – Face Recognition

CognitiveSystems

- When properly weighted, eigenfaces can be summed together to create an approximate a human face.
- Due to the similar structure of human faces, this is already possible with surprisingly few eigenfaces
  - good for data compression
  - good for classification

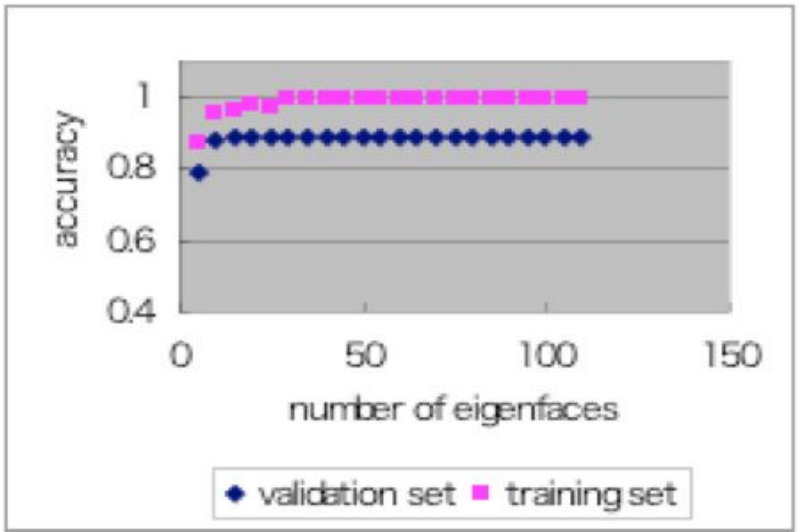
Eigenfaces – An experiment

- Data used here are from the ORL database of faces. Frontal face images of 16 persons each with 10 views are used.
  - Training set contains 16×7 images.
  - Test set contains 16×3 images.
- First three eigenfaces :

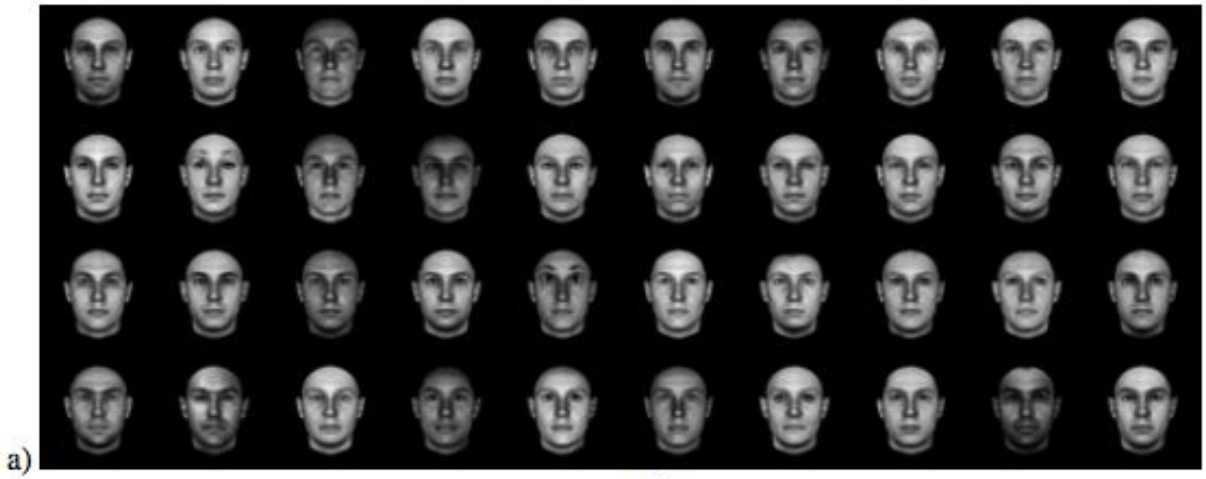


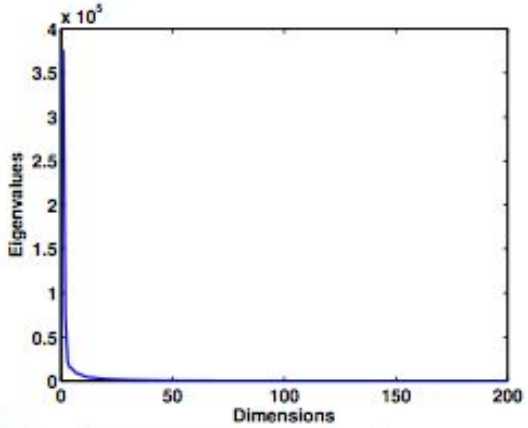
Classification Using Nearest Neighbor

- Save average coefficients for each person. Classify new face as the person with the closest average.
- Recognition accuracy increases with number of eigenfaces till roughly 15 dimensions.
- Later eigenfaces do not help much with recognition.
- Best recognition rates
  - Training set 99%
  - Test set 89%



Eigenfaces with PCA



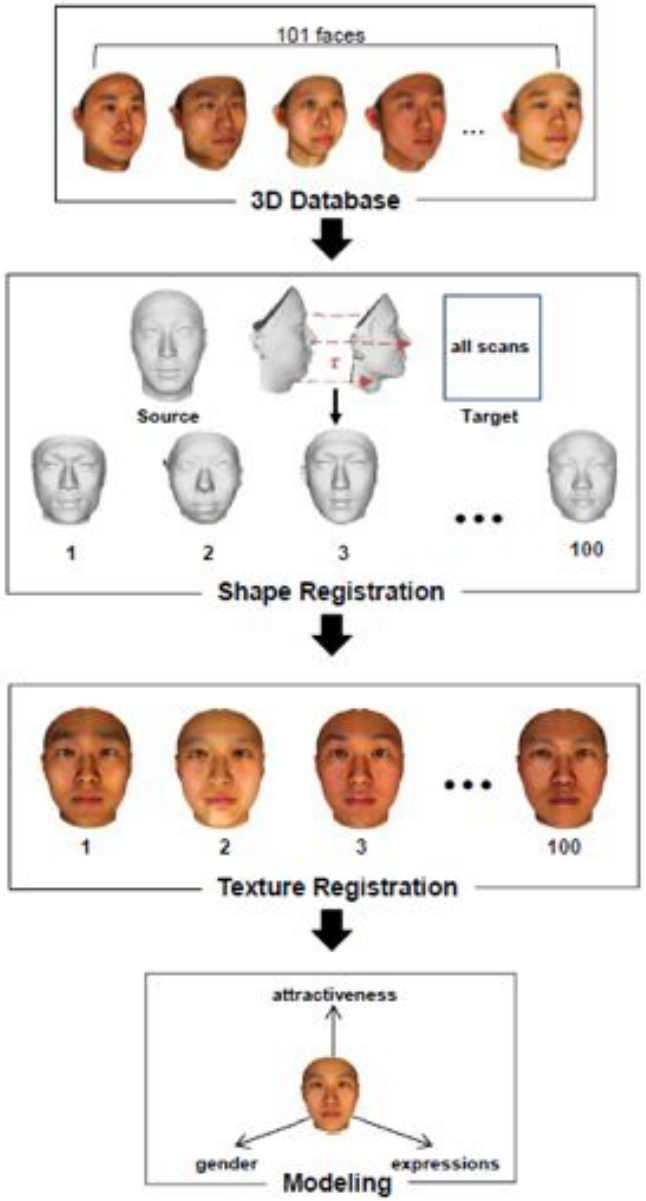


3D Eigenfaces with PCA

- If faces are captured in 3D, brought into correspondence, and then a PCA is calculated, a morphable model is created
- Classic paper by Blanz and Vetter (1999)
- We repeated this here at KU with >100 3D scans
  - shape and texture information

$$S = (X_1, Y_1, Z_1, X_2, \ldots, Y_n, Z_n)^T \in R^{3n}$$

$$T = (R_1, G_1, B_1, R_2, \ldots, G_n, B_n)^T \in R^{3n}$$



3D Eigenfaces with PCA

- Calculate mean shape and texture information
- Determine covariance matrix for shape and texture
- Do PCA to retrieve eigenvectors and eigenvalues
- After faces are brought into correspondence, shape and texture information can be represented as

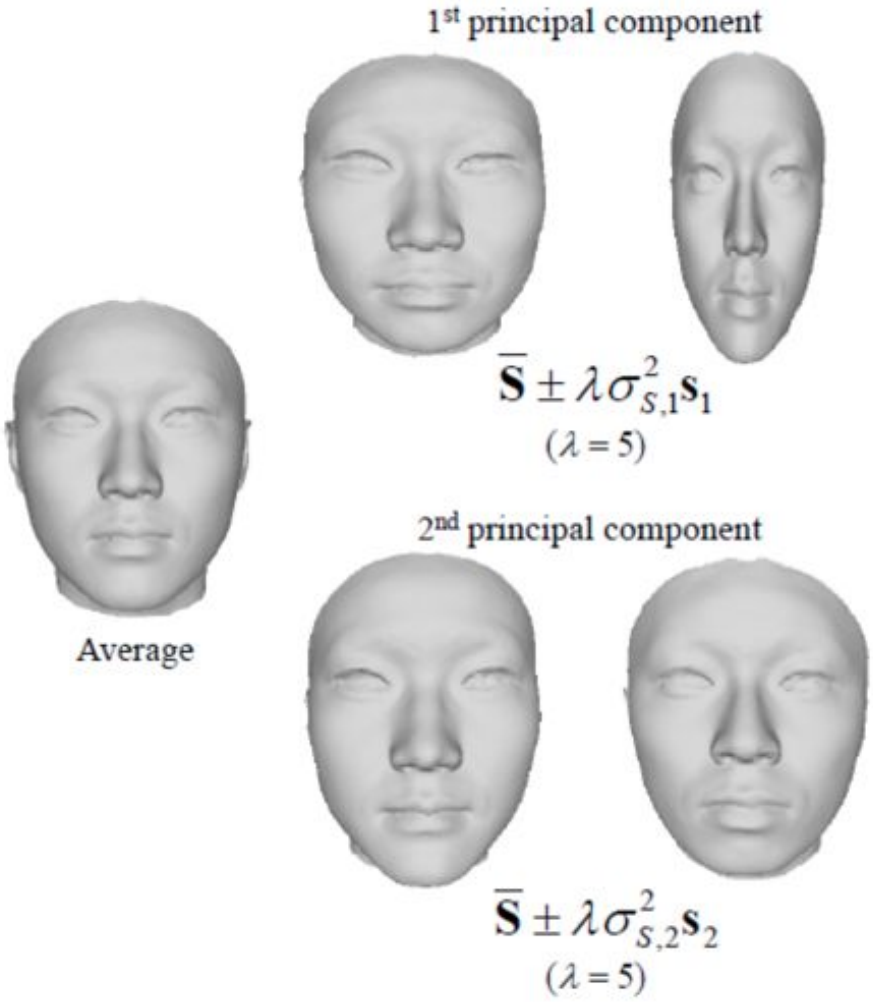
$$\bar{\mathbf{s}} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{S}_i, \quad \bar{\mathbf{t}} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{T}_i$$

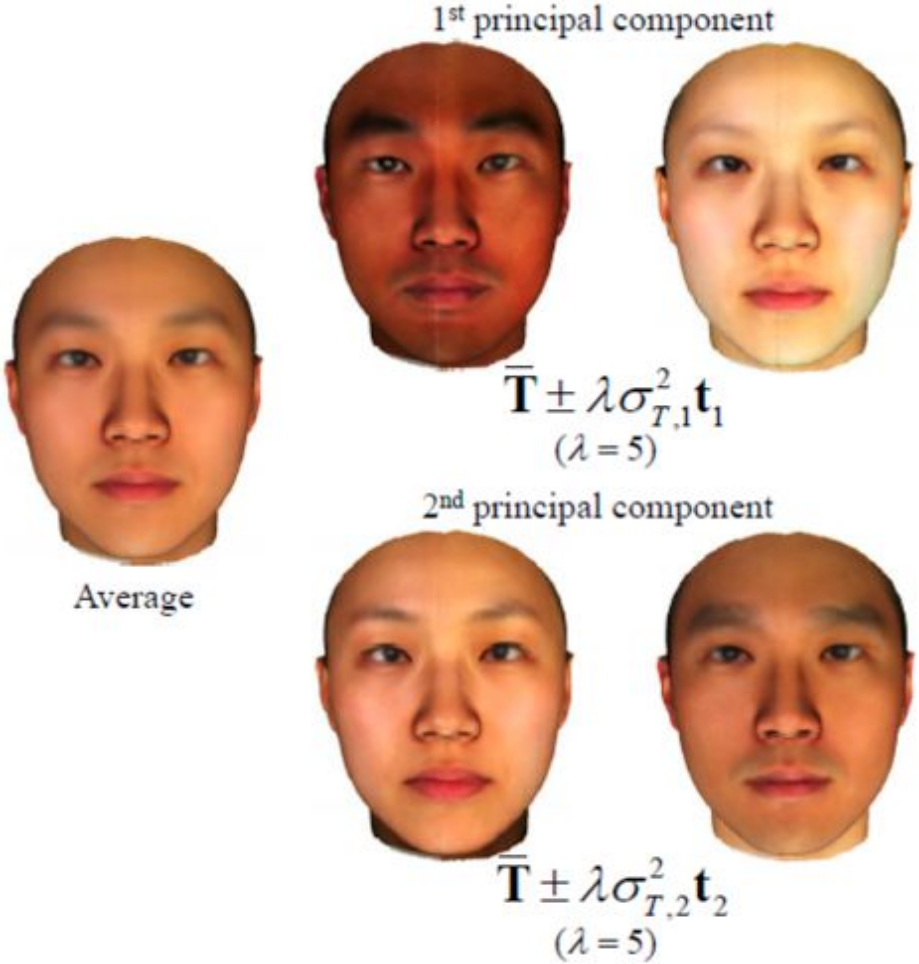
$$\mathbf{C} = \frac{1}{m}\mathbf{A}\mathbf{A}^T$$

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots\ldots, \mathbf{a}_{m-1}, \mathbf{a}_m], \mathbf{a}_i = \mathbf{S}_i - \bar{\mathbf{s}}$$

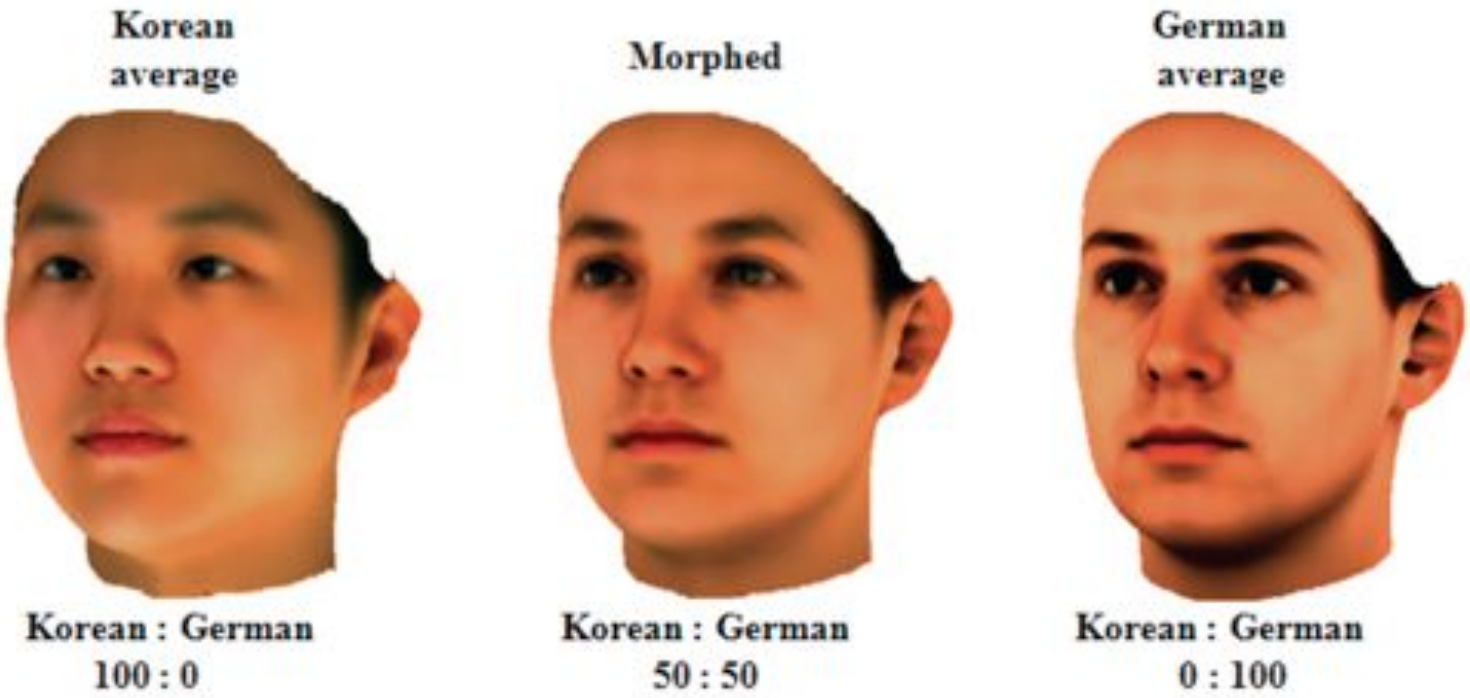
$$\mathbf{S} = \bar{\mathbf{s}} + \sum_{i=1}^{m-1}\alpha_i \cdot \mathbf{s}_i, \quad \mathbf{T} = \bar{\mathbf{t}} + \sum_{i=1}^{m-1}\beta_i \cdot \mathbf{t}_i$$

Shape and texture eigenvectors





Morphing between faces



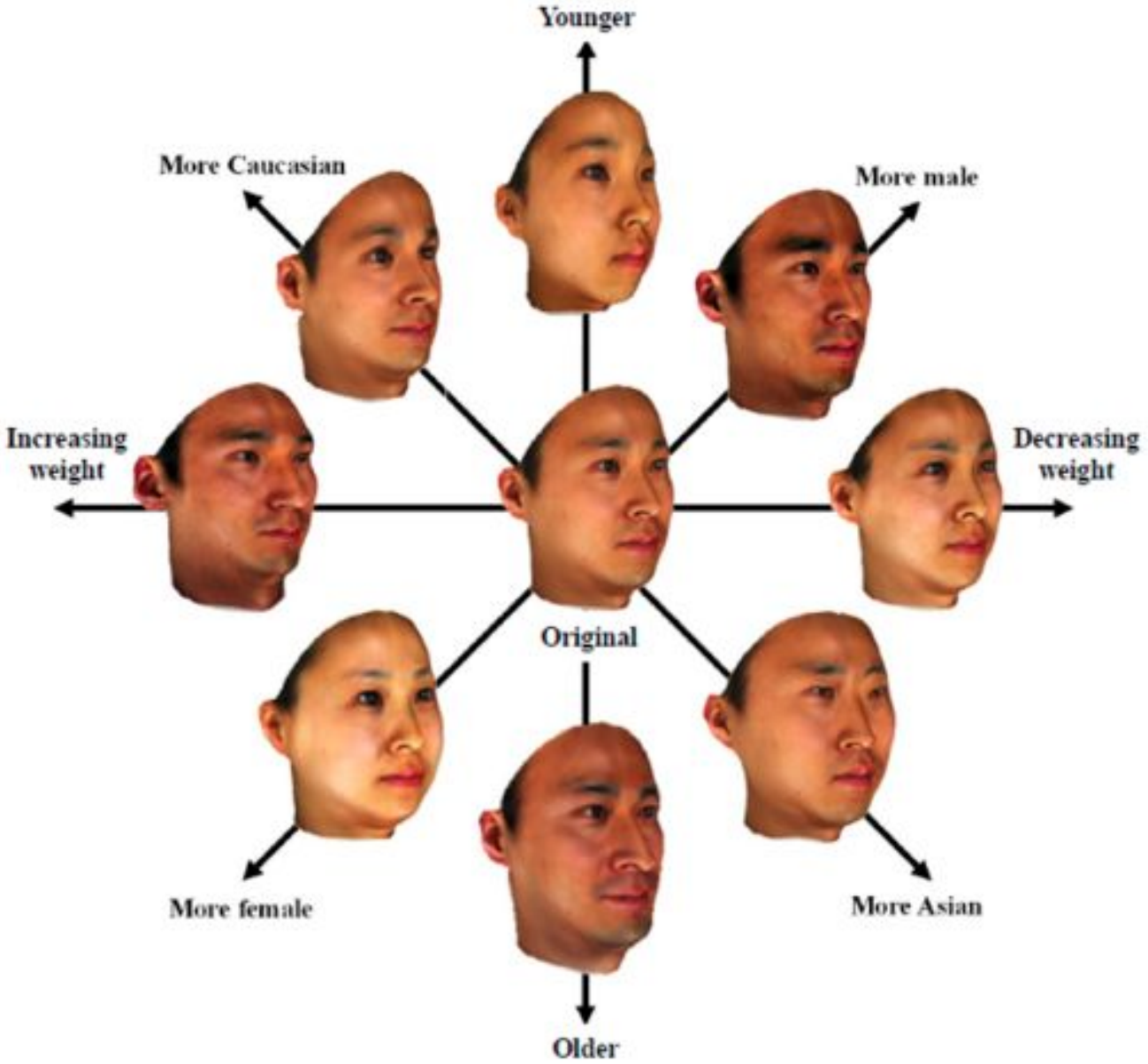
Making faces distinctive

Average

Original

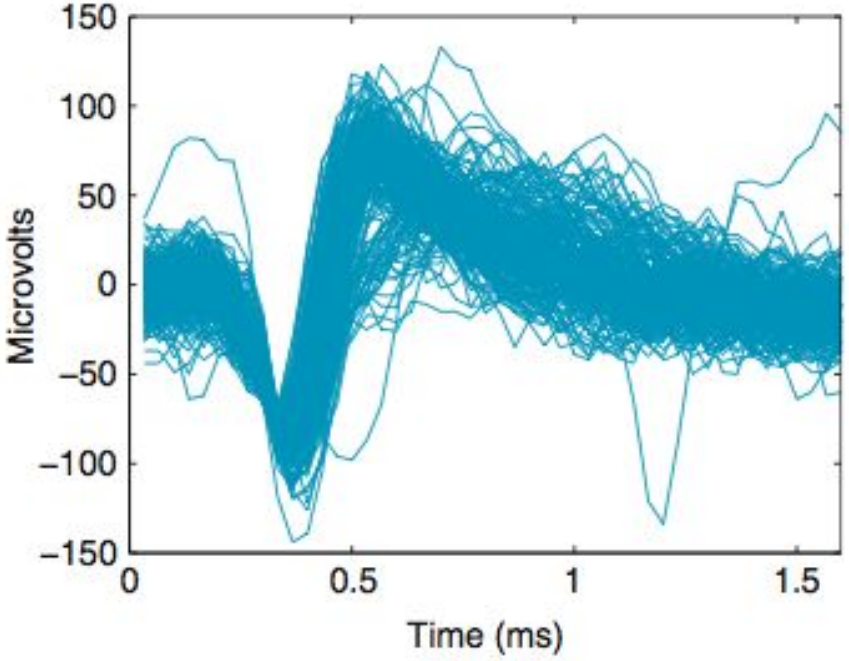
Caricature

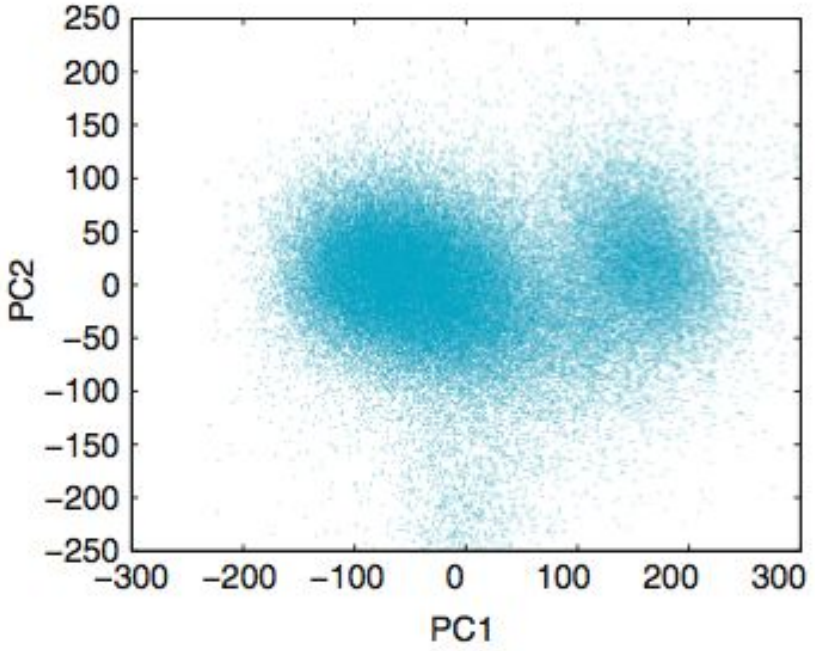
Manipulating faces



Spike-sorting with PCA

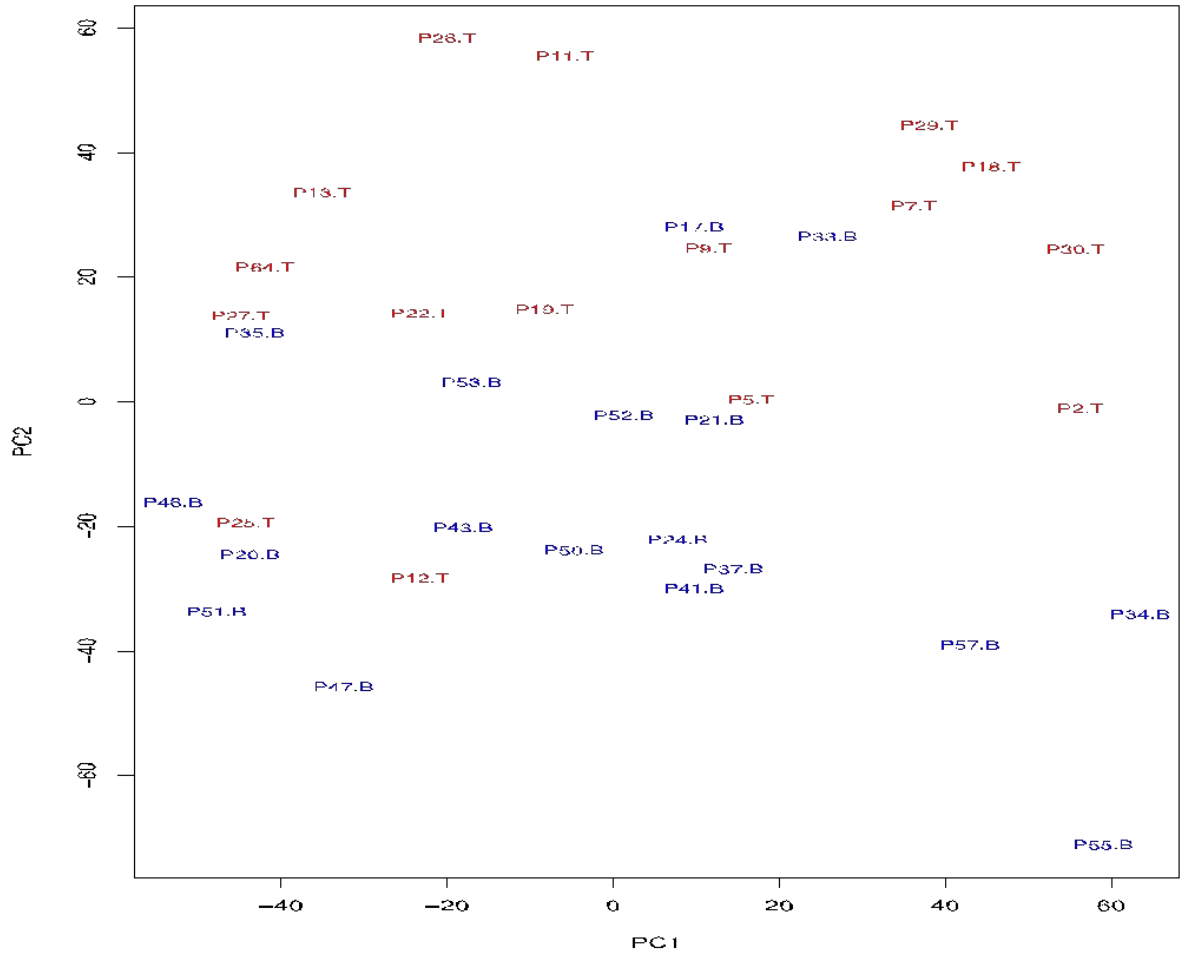
- pca_spikesorting.m





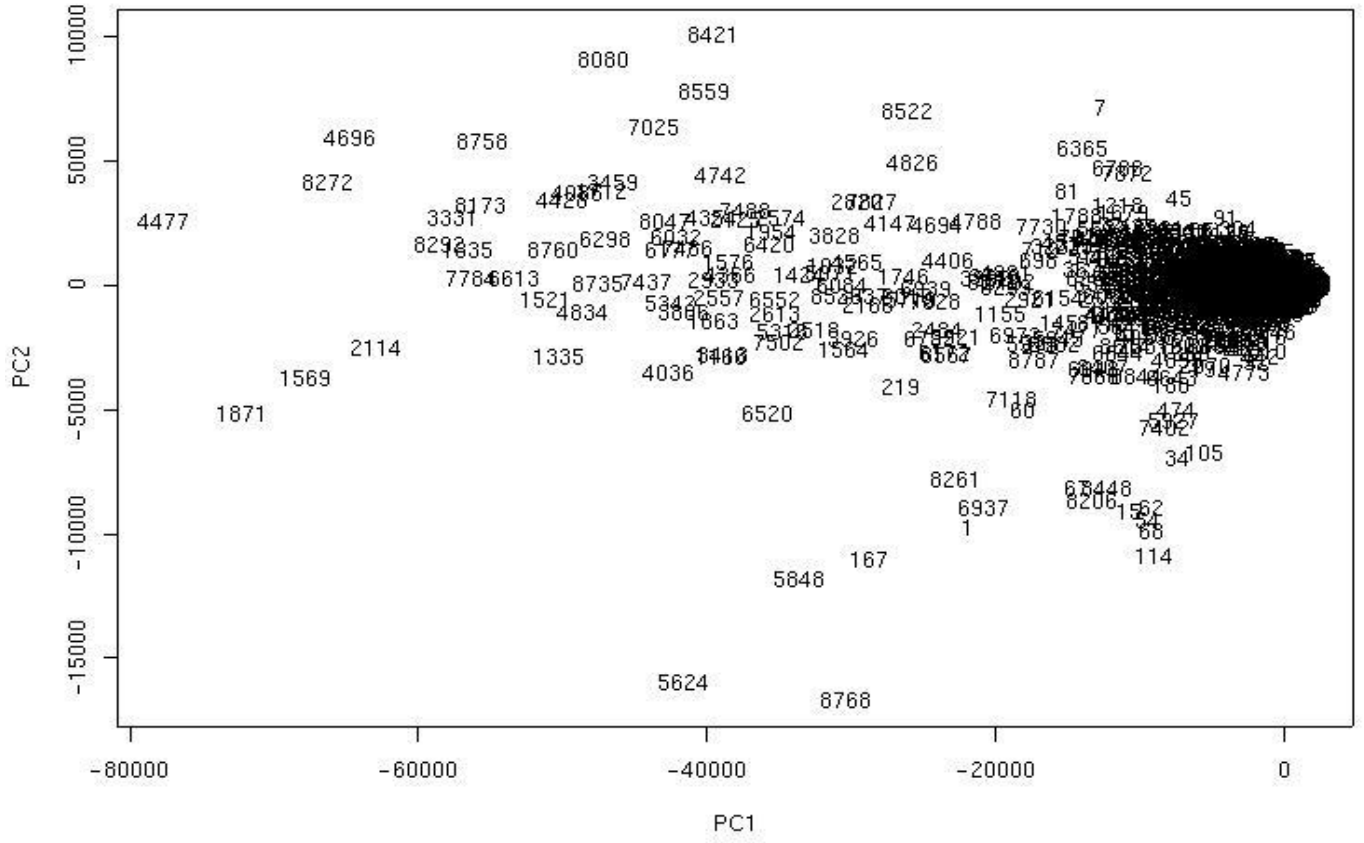
PCA of Genes (Leukemia data, precursor B and T cells)

- **34 patients**, dimension of 8973 genes reduced to 2

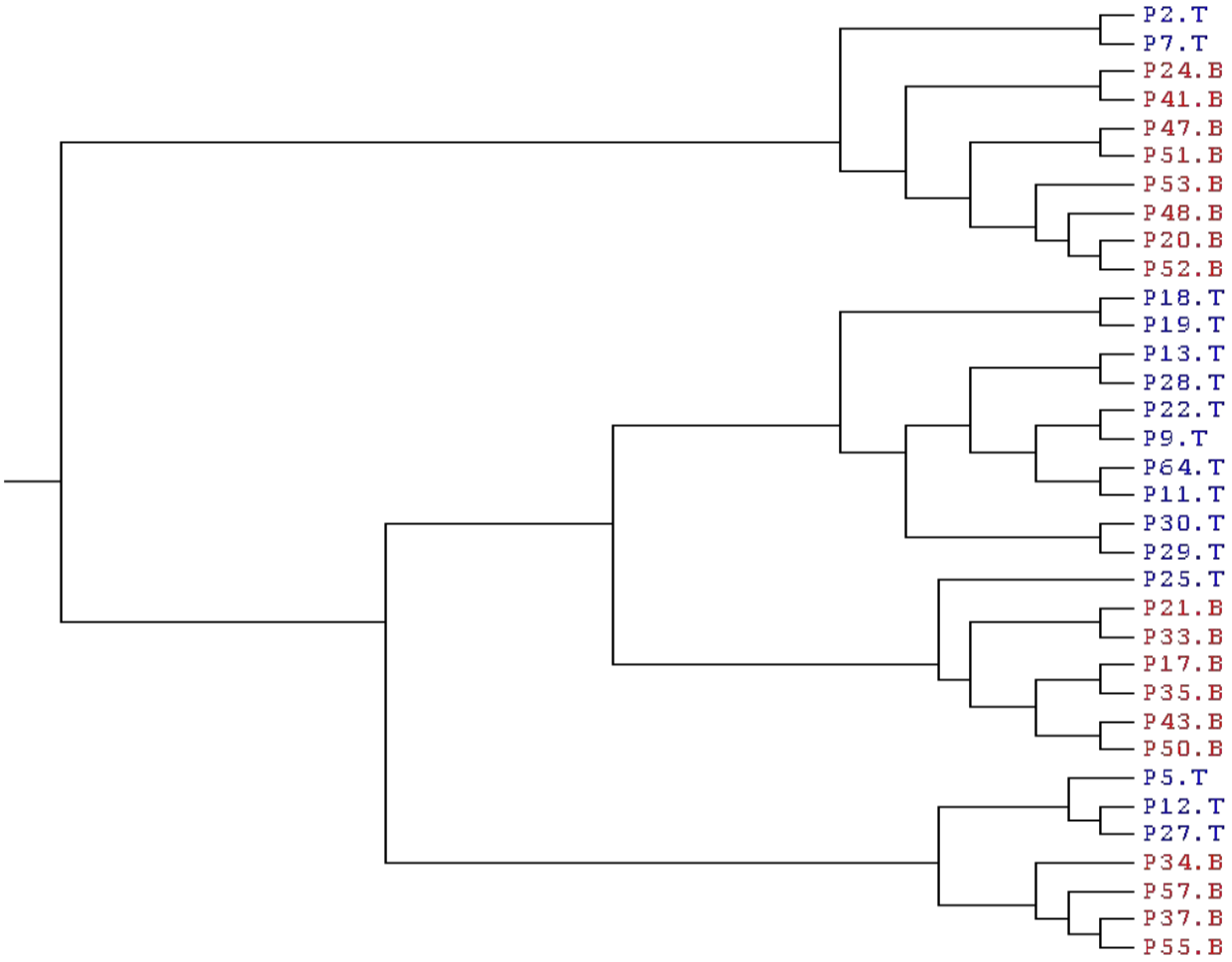


PCA of genes (Leukemia data)

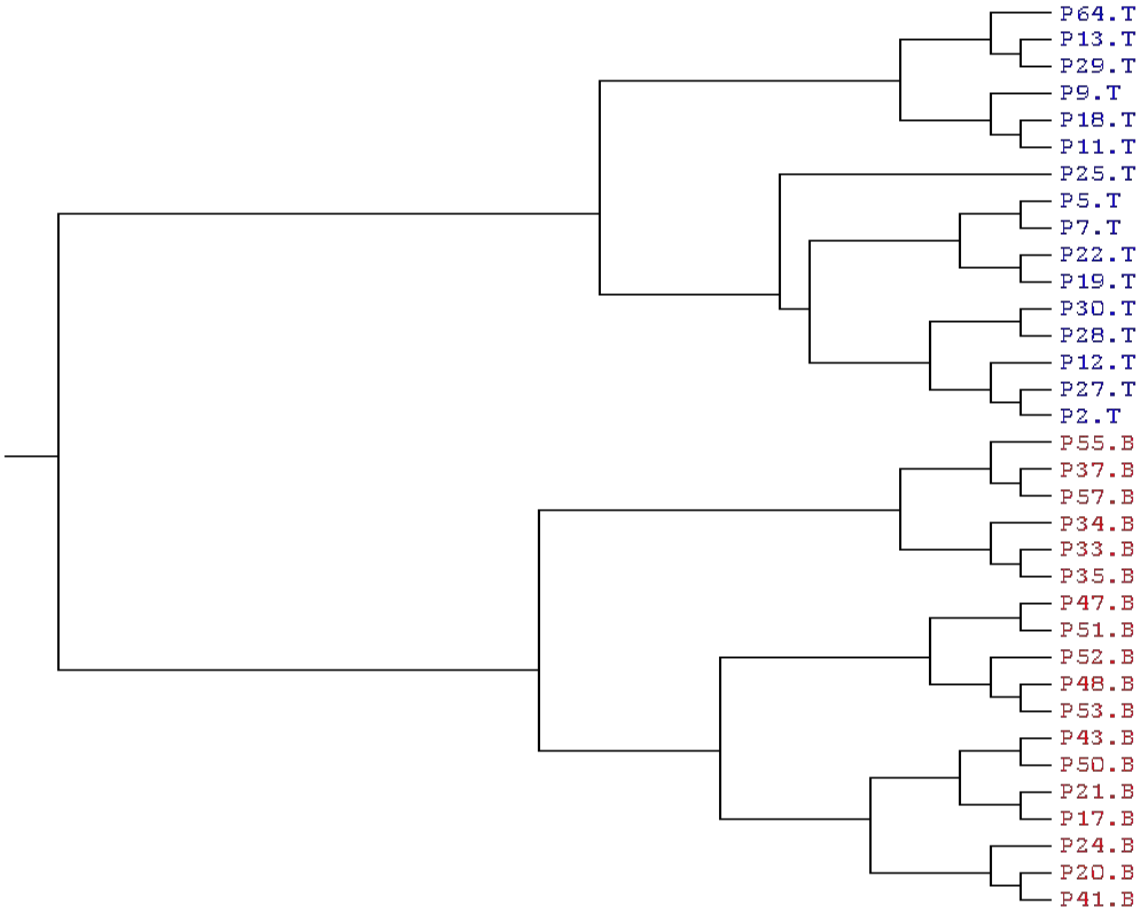
- **Plot of 8973 genes**, dimension of 34 patients reduced to 2



Leukemia data - clustering of patients on original gene dimensions



Leukemia data - clustering of patients on top 100 eigen-genes



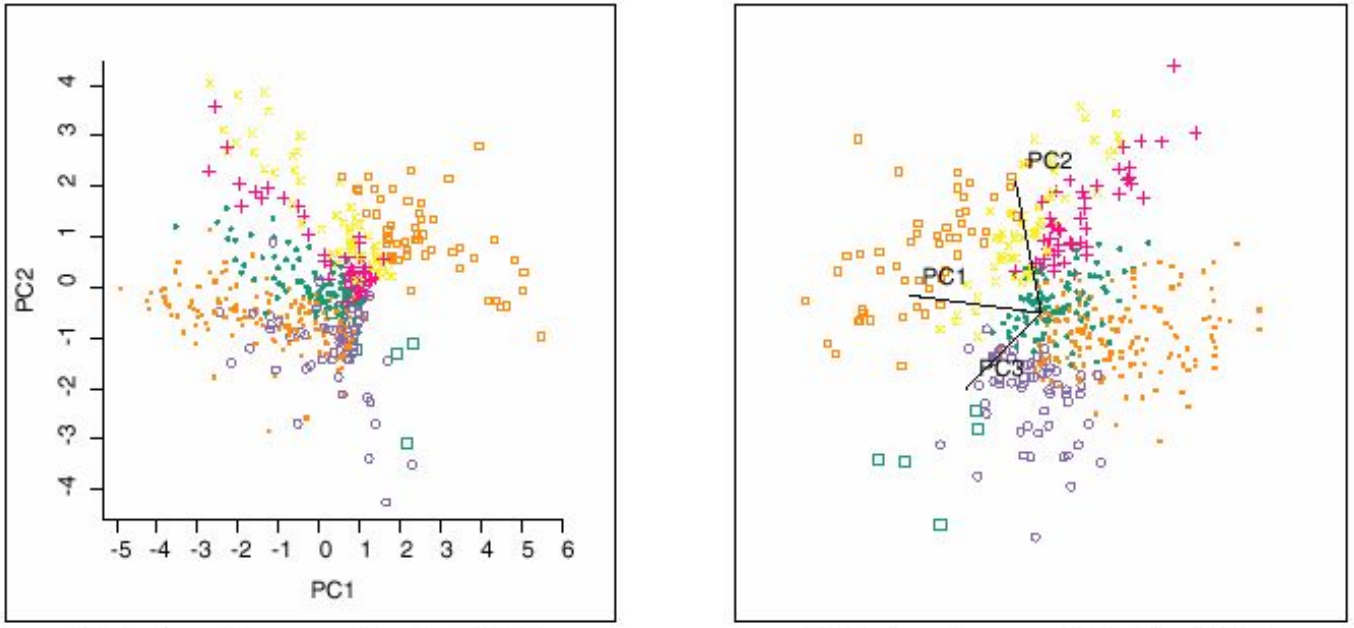
Sporulation Data Example

CognitiveSystems

- Data: A subset of yeast sporulation data (477 genes) were classified into seven temporal patterns (Chu et al., 1998)
- The first 2 PCs contains 85.9% of the variation in the data. (Figure 1a)
- The first 3 PCs contains 93.2% of the variation in the data. (Figure 1b)

Sporulation Data

- The patterns overlap around the origin in (**1a**).
- The patterns are much more separated in (**1b).**



(a) In the subspace of the first 2 PC's

(b) In the subspace of the first 3 PC's

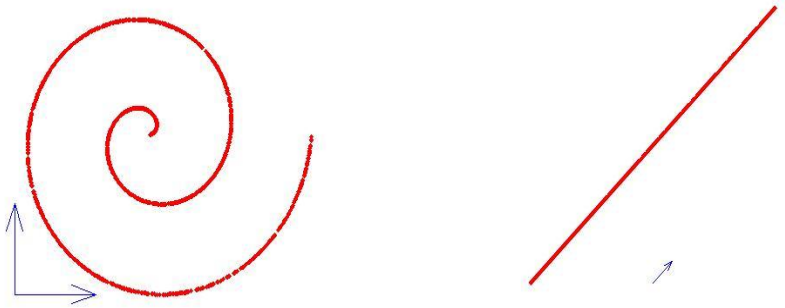
Fig. 1. Visualization of a subset of the sporulation data. (a) In the subspace of the first two PCs. (b) In the subspace of the first three PCs.

Limitations of PCA

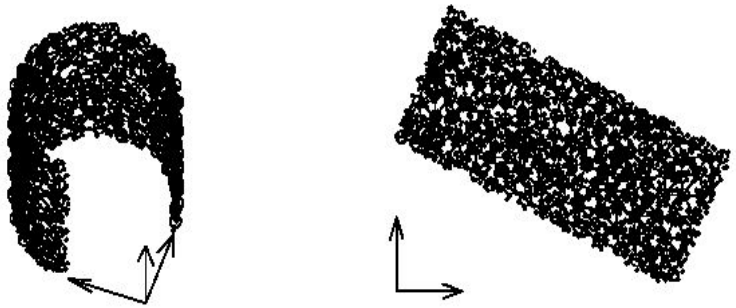
CognitiveSystems

- PCA assumes that data come from a uni-modal Gaussian distribution
- The reduction of dimensions for complex distributions may need non linear processing
- Curvilinear Component Analysis (CCA)
  - Non linear extension of PCA
  - Preserves the proximity between the points in the input space i.e. local topology of the distribution
  - Enables to unfold some varieties in the input data
  - Keep the local topology

Example of data representation using CCA



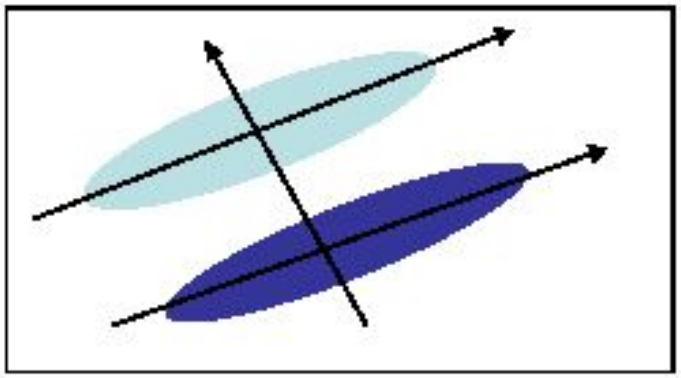
Non linear projection of a spiral



Non linear projection of a horseshoe

PCA and Discrimination

- PCA may not find the best directions for discriminating between two classes.
- Example: suppose the two classes have 2D Gaussian densities as ellipsoids.
- 1st eigenvector is best for representing the probabilities.
- 2nd eigenvector is best for discrimination.

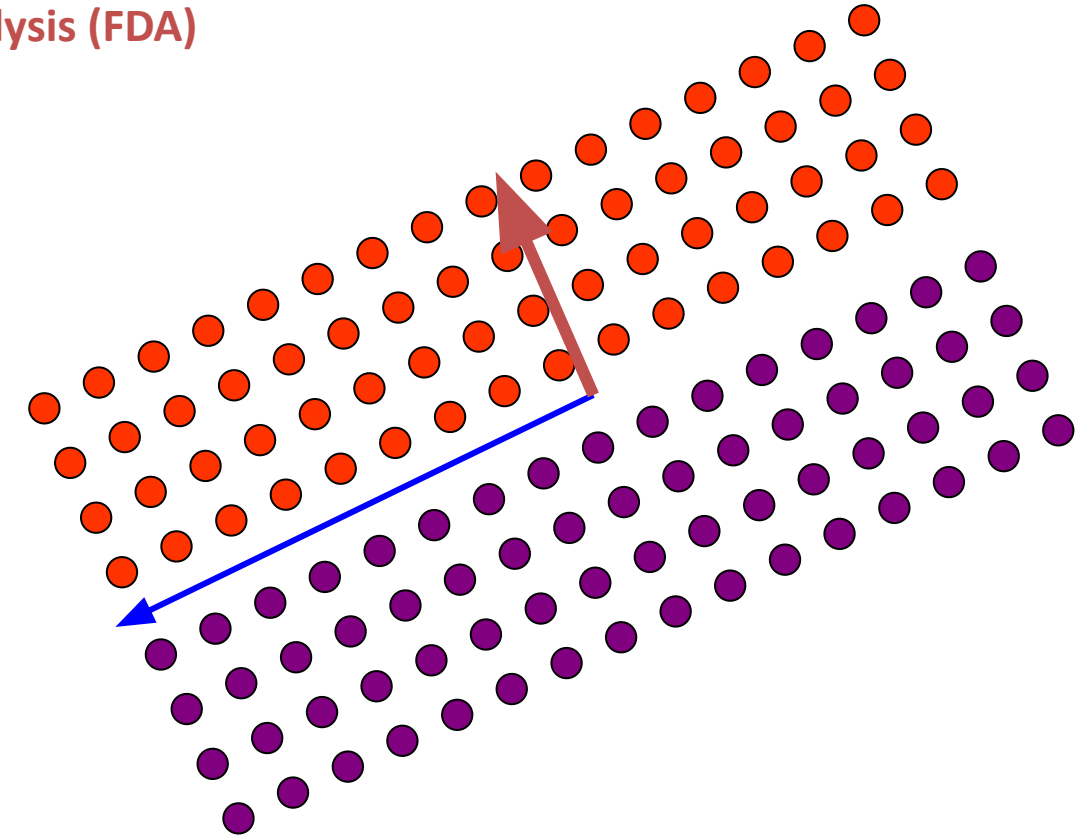


Limitations of PCA

CognitiveSystems

Are the maximal variance dimensions the relevant dimensions for preservation?

- Relevant Component Analysis (RCA)
- Fisher Discriminant analysis (FDA)

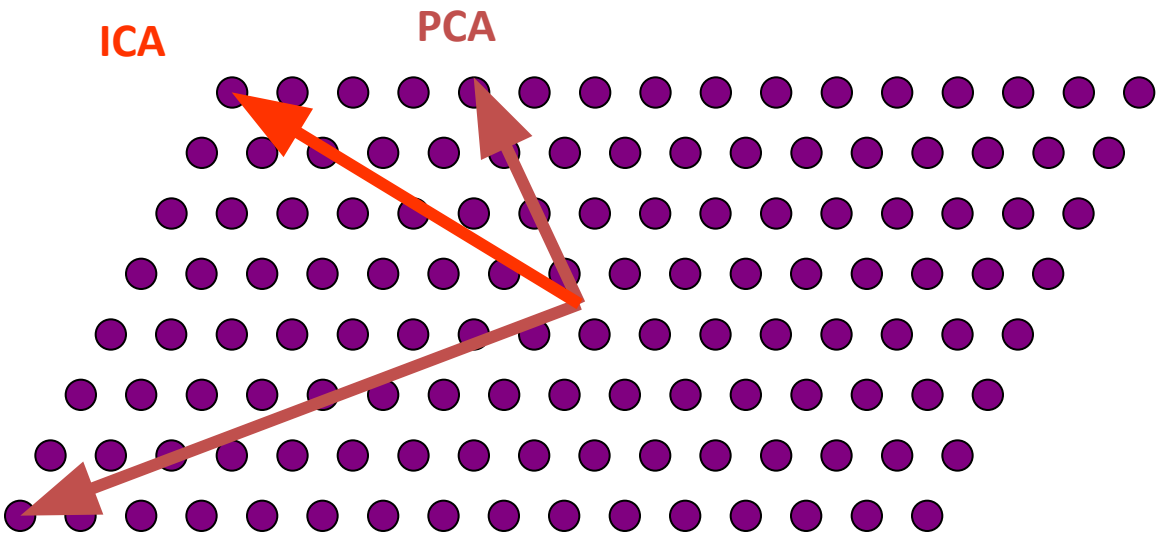


Limitations of PCA

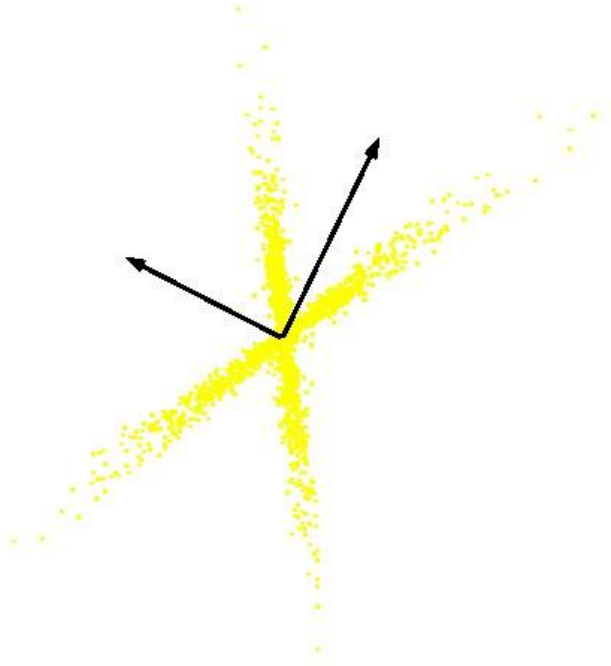
CognitiveSystems

Should the goal be finding independent rather than pair-wise uncorrelated dimensions

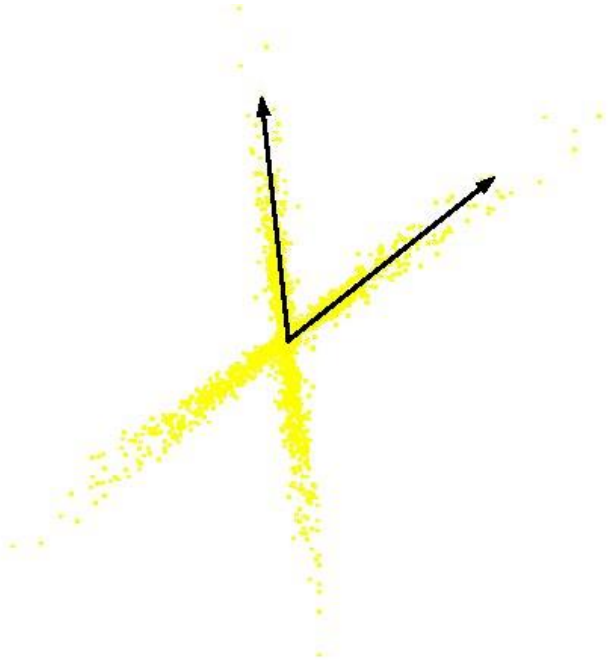
- Independent Component Analysis (ICA)



PCA vs ICA



PCA
(orthogonal axes)



ICA
(non-orthogonal axes)

Cocktail-party Problem

CognitiveSystems

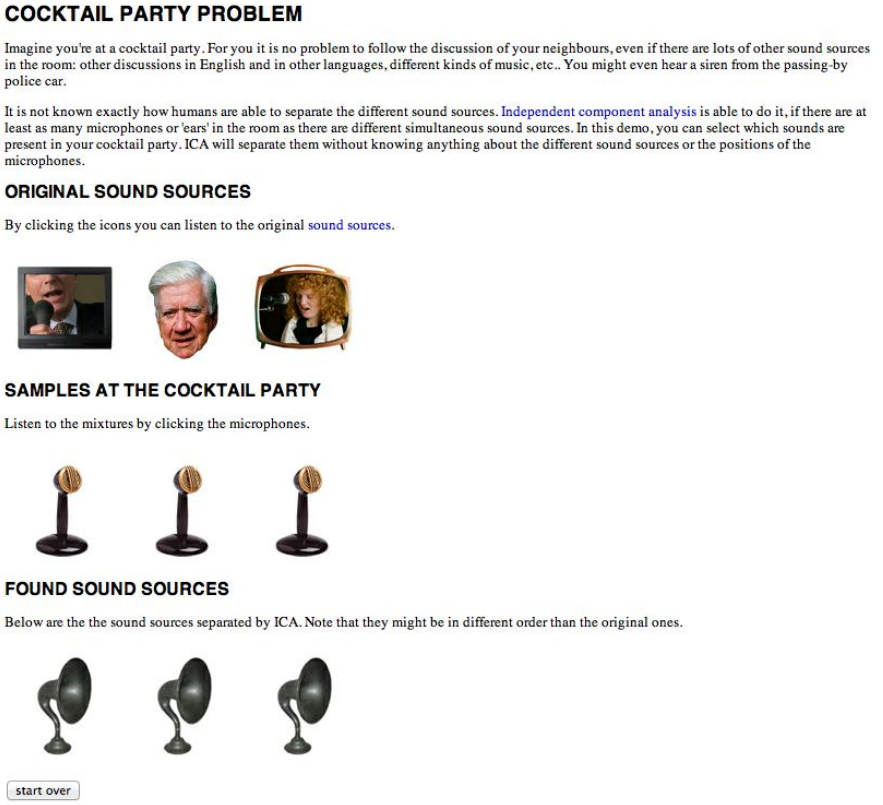
- Problem setup
  - Multiple sound sources in room (each source is an independent source)
  - Multiple sensors receiving signals, which are a mixture of original signals
- Goal:
  - Estimate original source signals from mixture of received signals
- This problem is also called “Blind-Source Separation” (BSS) as the mixing parameters are not known, and hence the sources need to be reconstructed

Demo: ICA for Blind Source Seperation

- http://www.cis.hut.fi/projects/ica/cocktail/cocktail_en.cgi

  or

- http://cnl.salk.edu/~tewon/Blind/blind_audio.html



COCKTAIL PARTY PROBLEM

Imagine you're at a cocktail party. For you it is no problem to follow the discussion of your neighbours, even if there are lots of other sound sources in the room: other discussions in English and in other languages, different kinds of music, etc.. You might even hear a siren from the passing-by police car.

It is not known exactly how humans are able to separate the different sound sources. Independent component analysis is able to do it, if there are at least as many microphones or 'ears' in the room as there are different simultaneous sound sources. In this demo, you can select which sounds are present in your cocktail party. ICA will separate them without knowing anything about the different sound sources or the positions of the microphones.

ORIGINAL SOUND SOURCES

By clicking the icons you can listen to the original sound sources.

SAMPLES AT THE COCKTAIL PARTY

Listen to the mixtures by clicking the microphones.

FOUND SOUND SOURCES

Below are the the sound sources separated by ICA. Note that they might be in different order than the original ones.

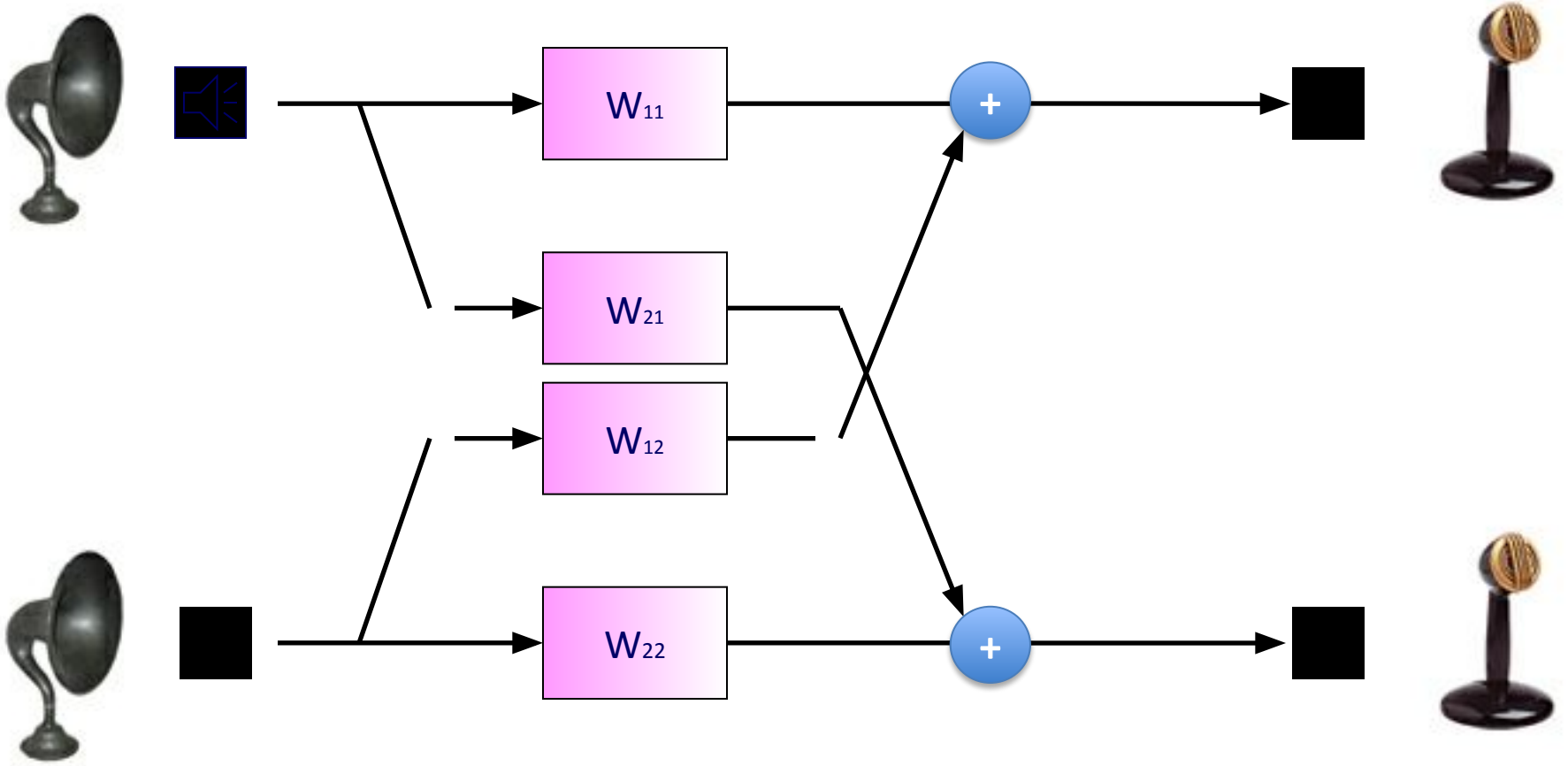
start over

BSS and ICA

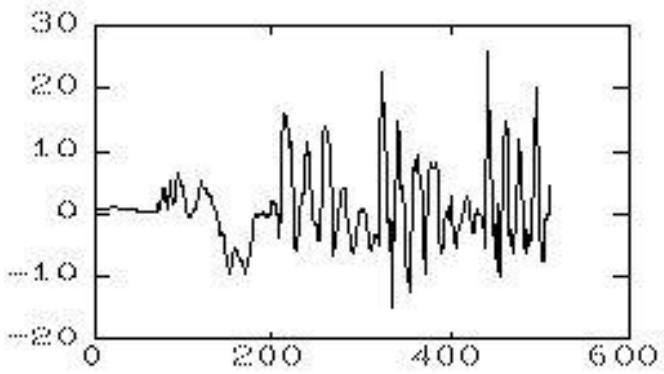
CognitiveSystems

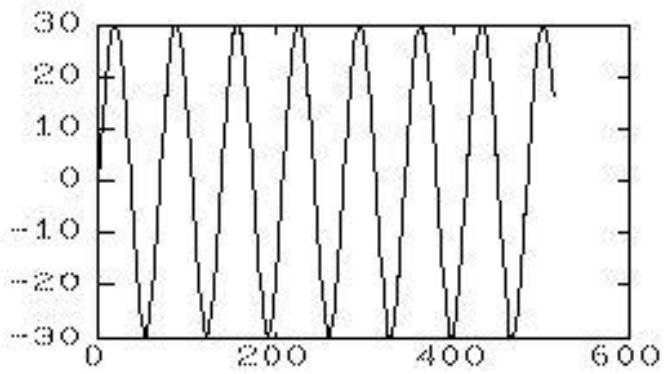
- Blind Source Separation (BSS) problem is ill-posed, unless assumptions are made!
- Most common assumption is that source signals are **statistically independent**.
  - Knowing value of one signal gives no information about the other.
- Methods based on this assumption are called Independent Component Analysis (ICA) methods
- It can be shown that under some reasonable conditions, if the ICA assumption holds, then the source signals can be recovered up to permutation and scaling.

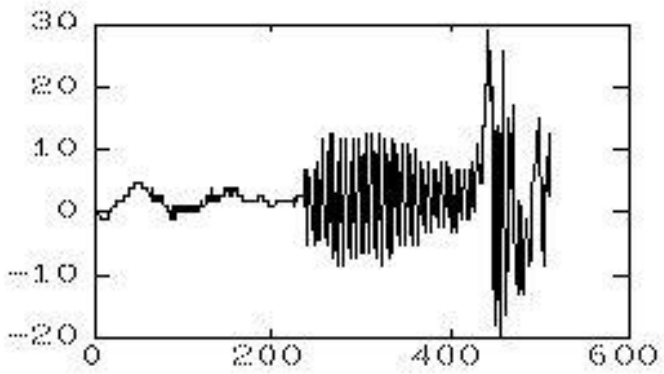
Source Separation Using ICA

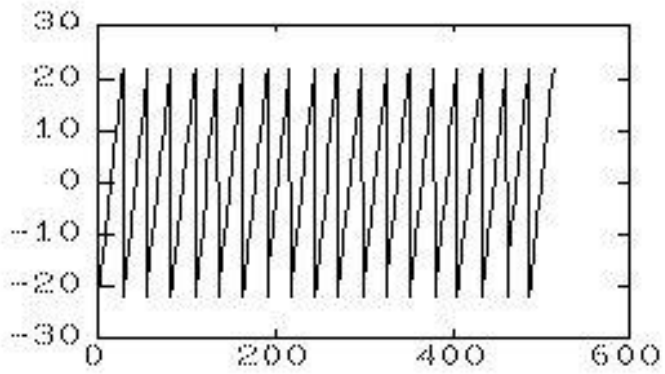


Source signals







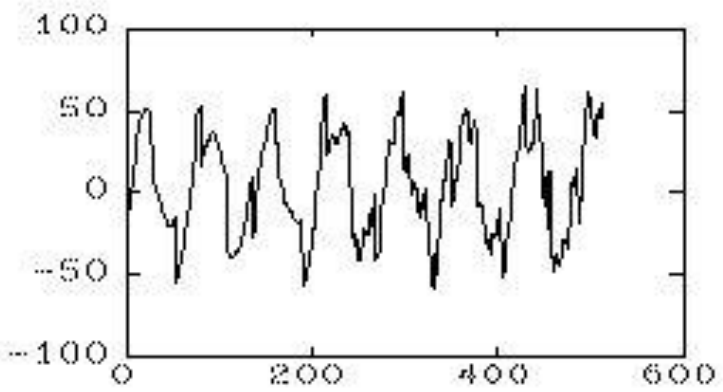


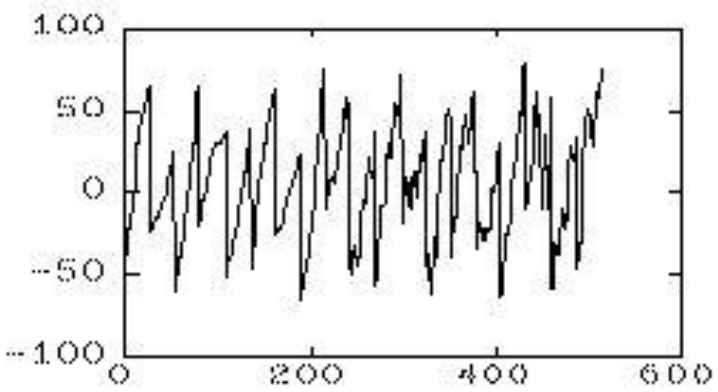
Original signals (hidden sources) $s_1(t)$, $s_2(t)$, $s_3(t)$, $s_4(t)$, t=1:T

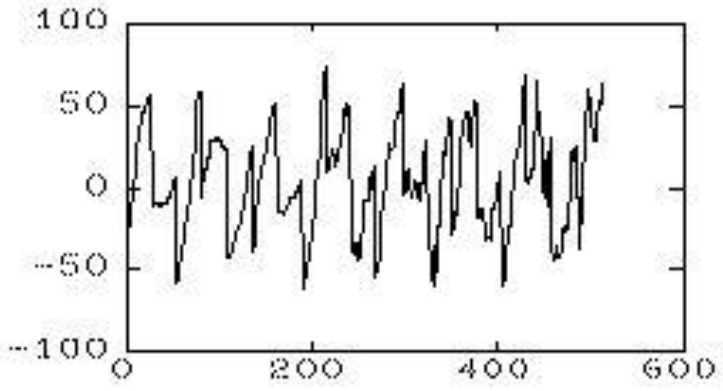
What the microphones hear

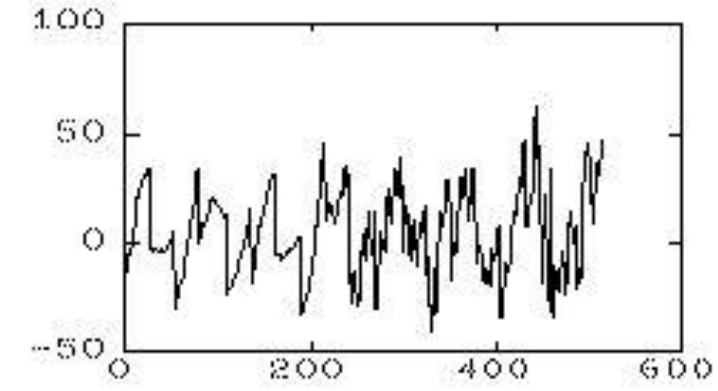
- $x_i(t) = a_{i1} * s_1(t) + a_{i2} * s_2(t) + a_{i3} * s_3(t) + a_{i4} * s_4(t)$
- i=1:4
- In vector-matrix notation, and dropping the time-variable t, this is
  x = **A** * **s**
- Note, that this is **linear mixing**!

What the microphones hear





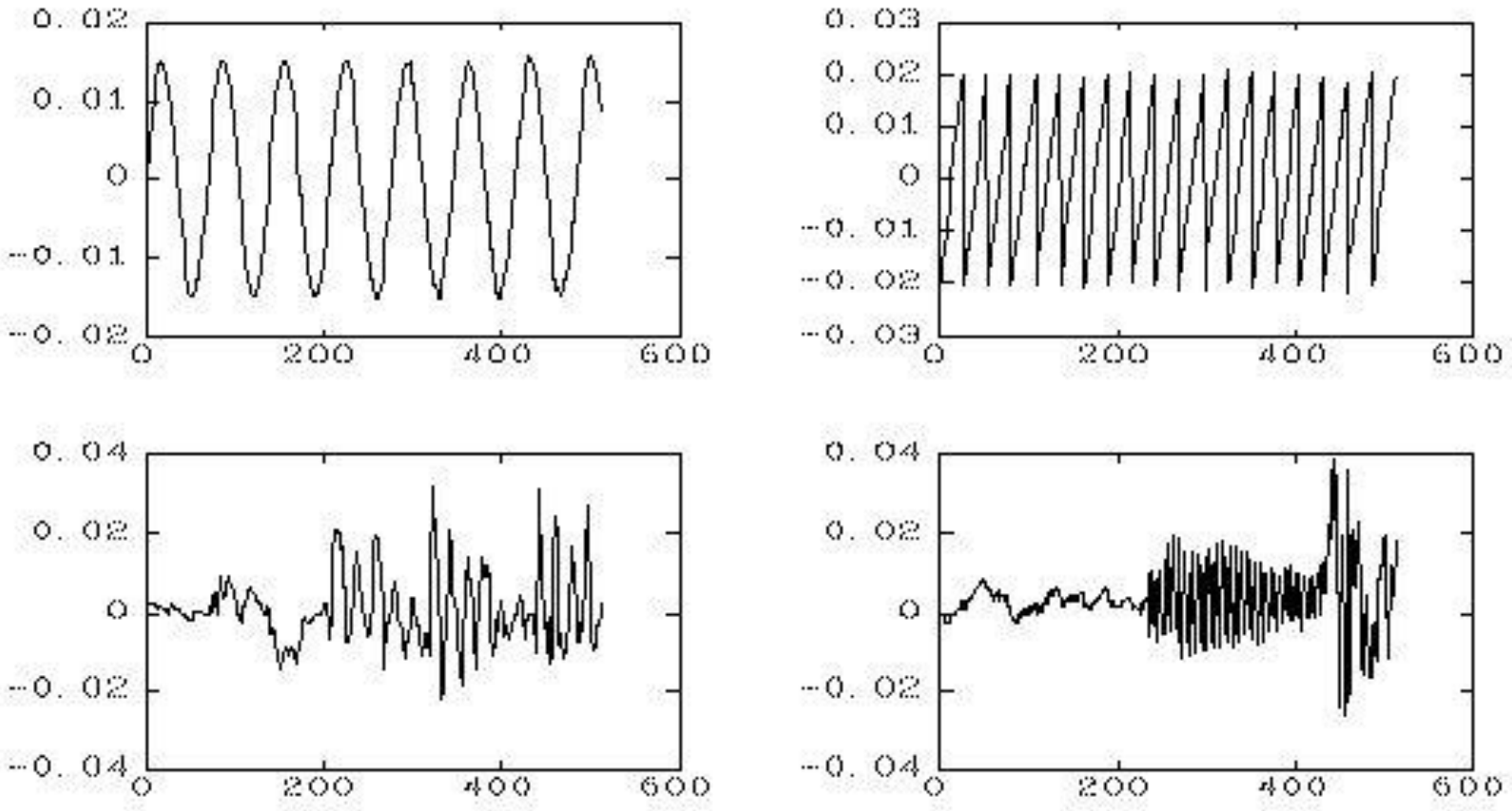




a linear mixture of the sources

$$x_i(t) = a_{i1} * s_1(t) + a_{i2} * s_2(t) + a_{i3} * s_3(t) + a_{i4} * s_4(t)$$

ICA recovery



Recovered signals

ICA Solution and Applicability

CognitiveSystems

- If we knew the mixing parameters a_{ij} then we would just need to solve a linear system of equations.
  - But: we know neither a_{ij} nor s_i.
- Given two-dimensional vector , $x = [x_1\ x_2]^T$, ICA aims at finding the following decomposition

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} s_1 + \begin{bmatrix} a_{12} \\ a_{22} \end{bmatrix} s_2$$

$$\mathbf{x} = \mathbf{a}_1 s_1 + \mathbf{a}_2 s_2$$

- where $\mathbf{a_1}$, $\mathbf{a_2}$ are basis vectors and s_1, s_2 are basis coefficients
- subject to the constraint: Basis coefficients s_1 and s_2 are statistically independent.

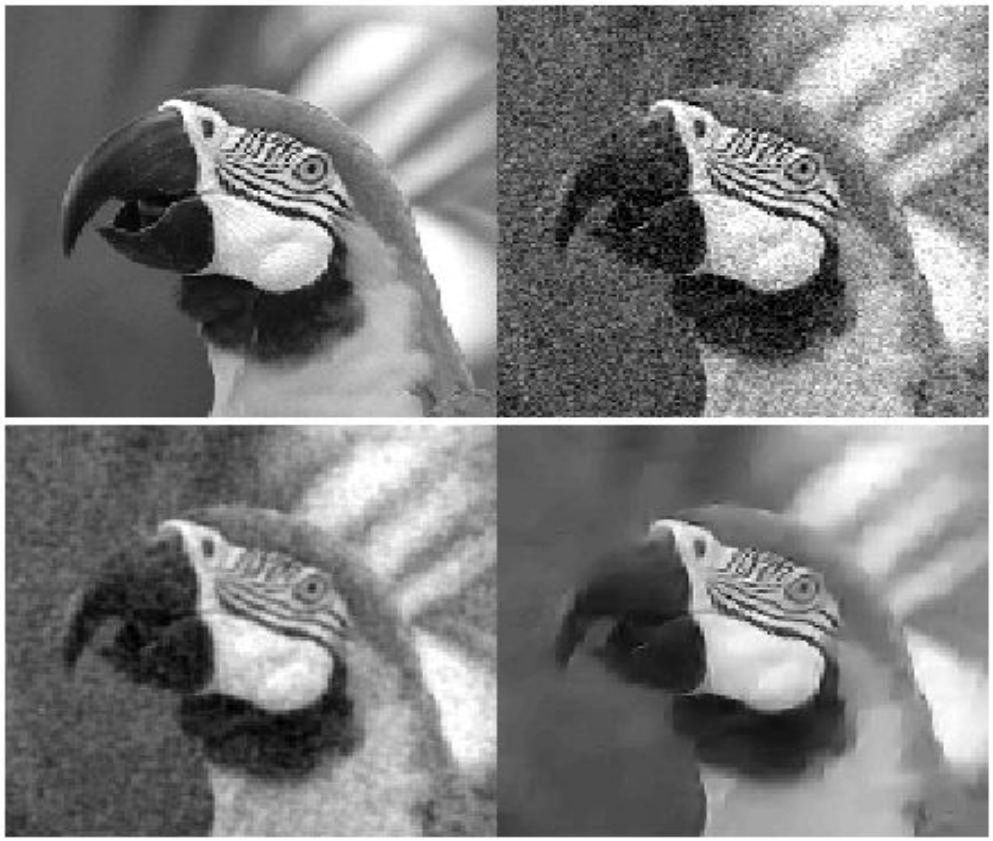
Application domains of ICA

CognitiveSystems

- Blind source separation
- Image denoising
- Medical signal processing – fMRI, ECG, EEG
- Modelling of the hippocampus and visual cortex
- Feature extraction, face recognition
- Compression, redundancy reduction
- Watermarking
- Clustering
- Time series analysis (stock market, microarray data)
- Topic extraction
- Econometrics: Finding hidden factors in financial data

Image denoising

Original image



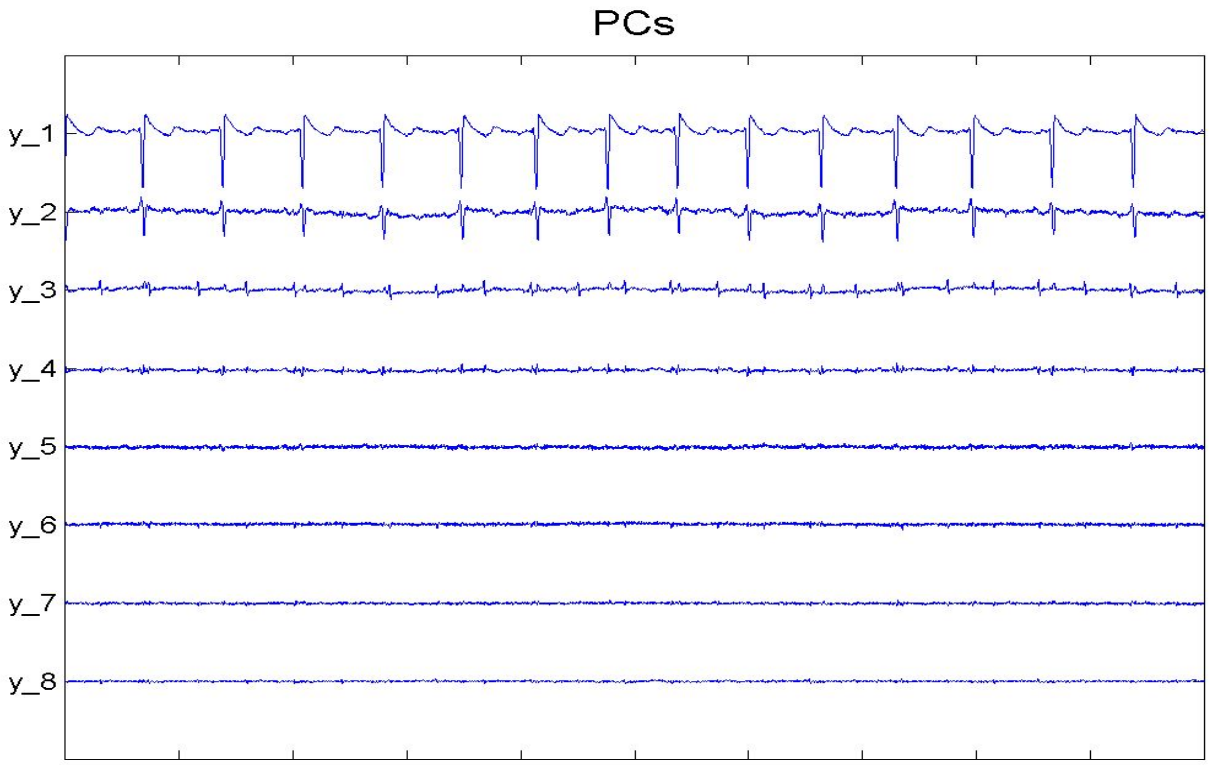
Noisy image

Wiener filtering

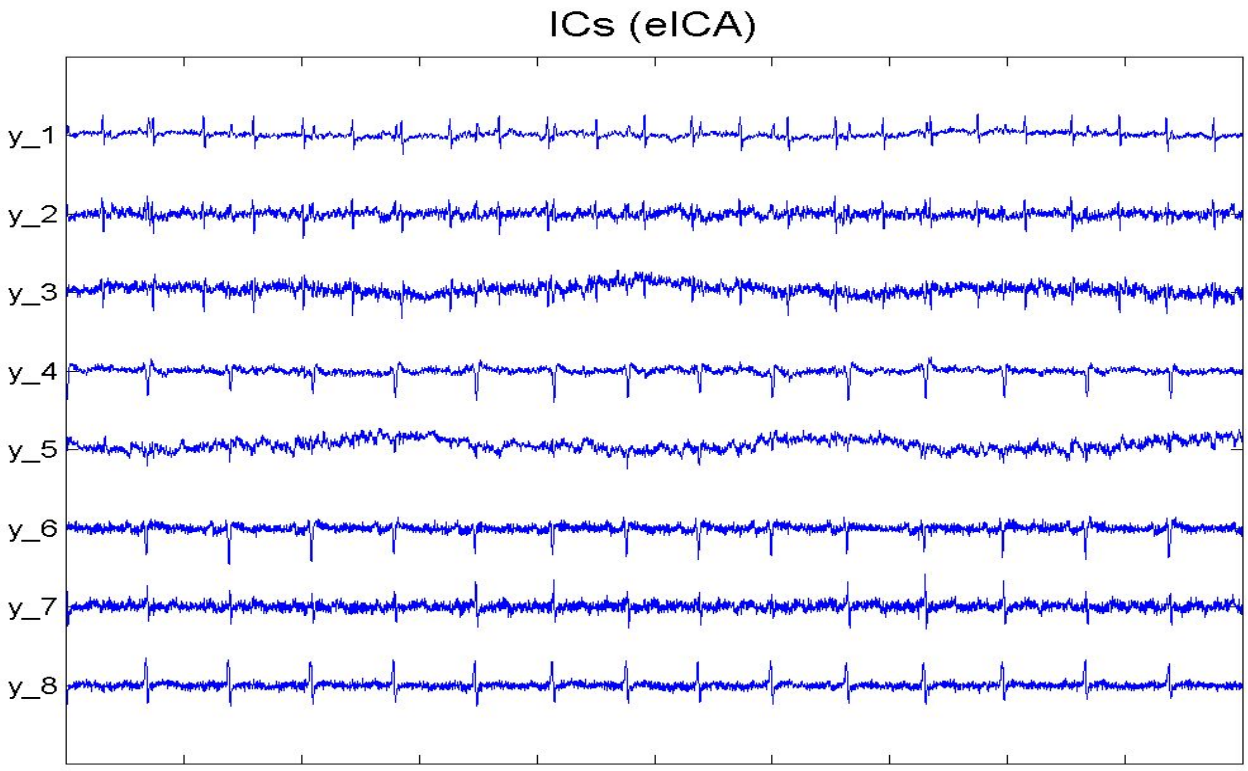
ICA filtering

Feature Extraction in ECG data (Raw Data)

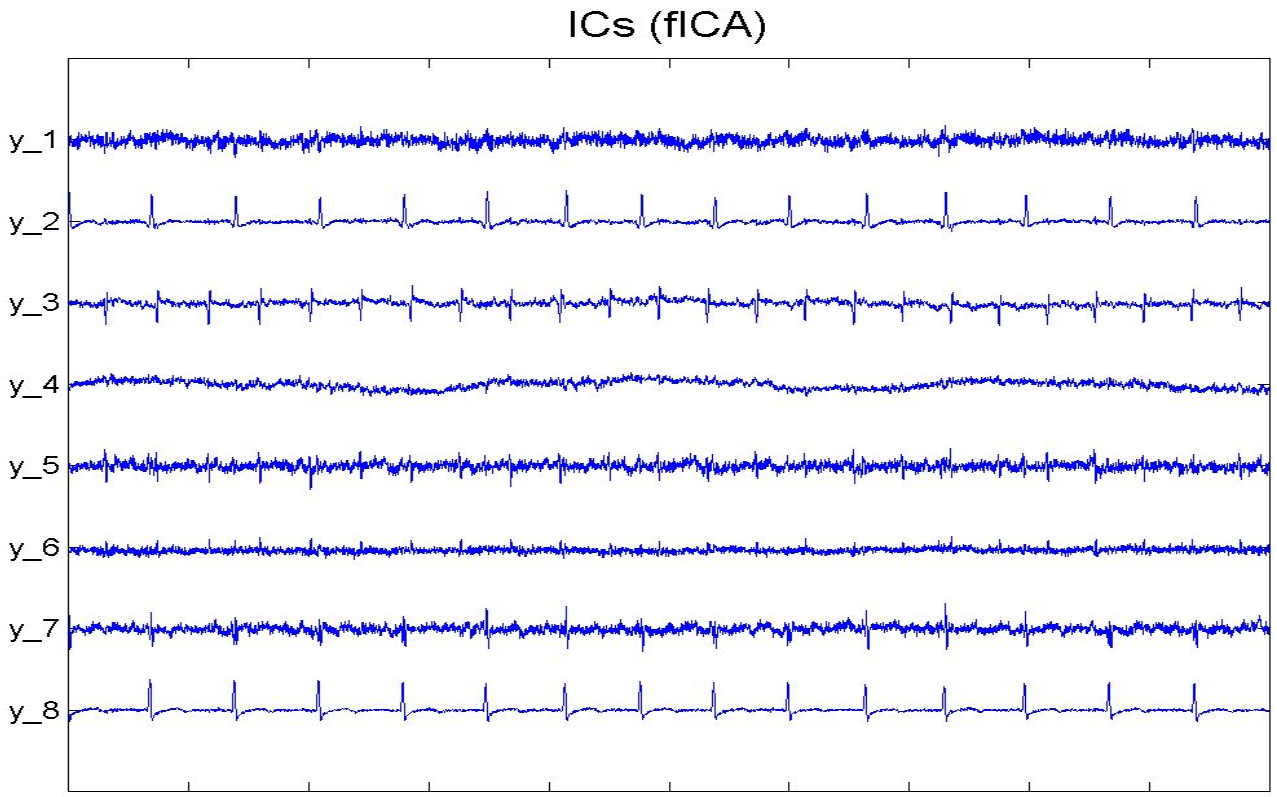
Feature Extraction in ECG data (PCA)



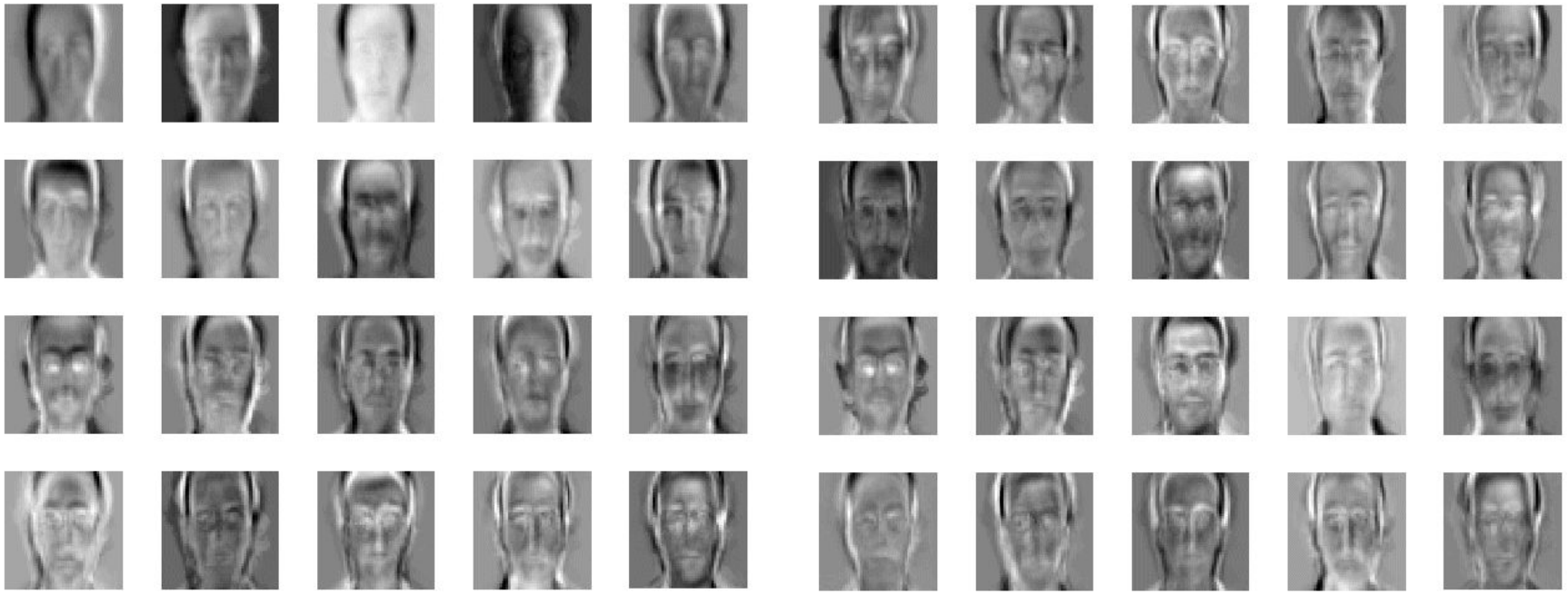
Feature Extraction in ECG data (Extended ICA)



Feature Extraction in ECG data (flexible ICA)



Eigenfaces vs Factorial Faces

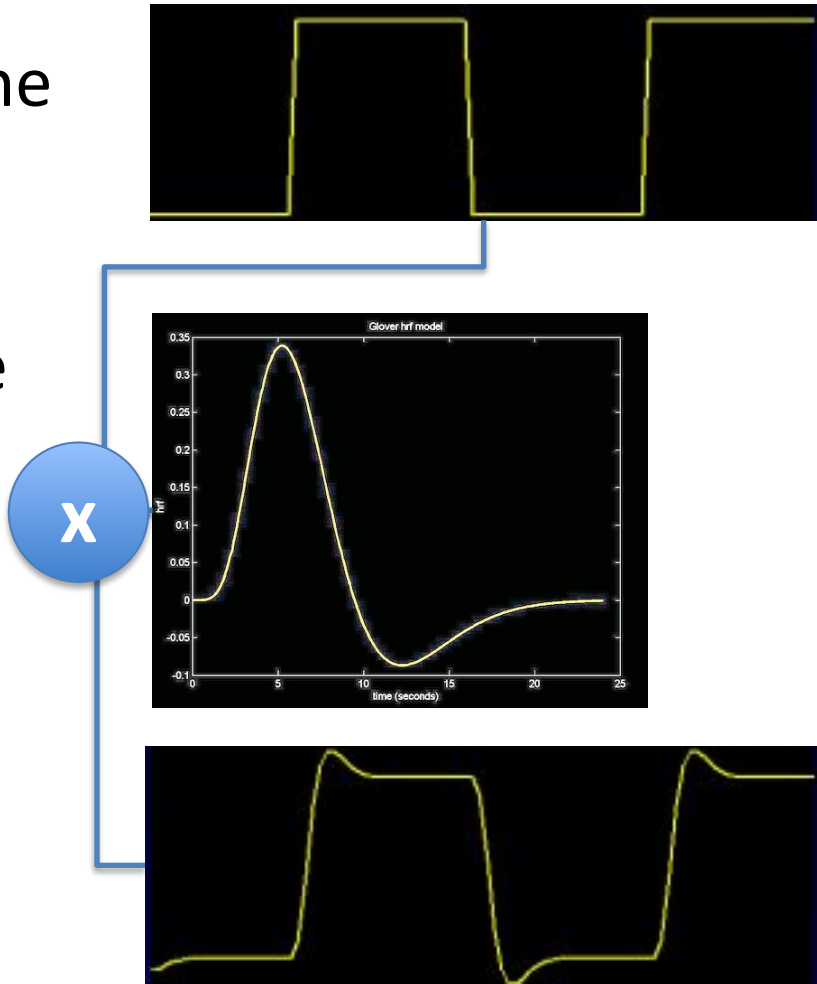


(a) (b)

Figure 3. First 20 basis images: (a) in eigenface method; (b) factorial code. They are ordered by column, then, by row.

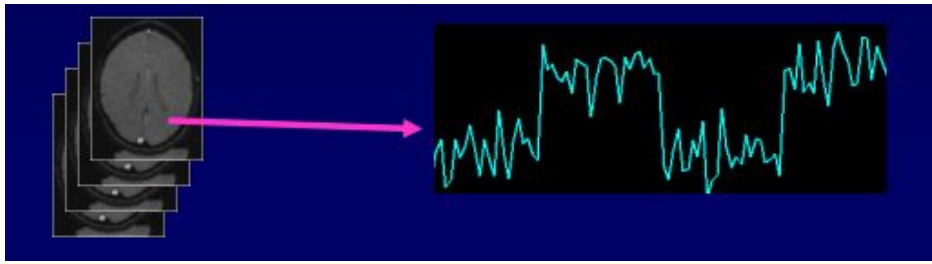
Analyzing fMRI data

- Standard approach for fMRI data is General Linear Model (GLM)
- Take the design matrix (when the task is switched on and off)
- Filter with the hemodynamic response model (oxygen uptake of blood)
- Obtain a hypothetic model for the voxel's BOLD response, if it fully correlates with the task



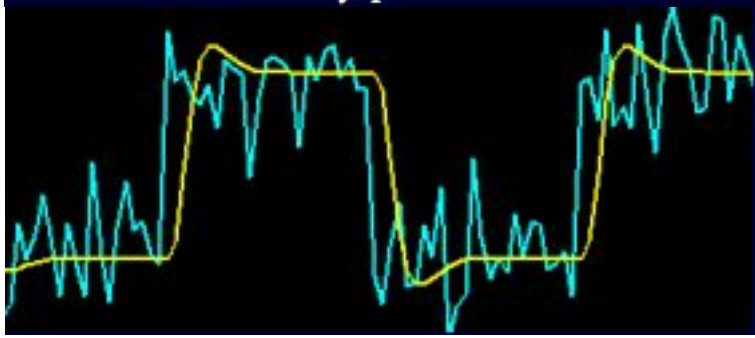
Analyzing fMRI data

- Extract time courses from each voxel

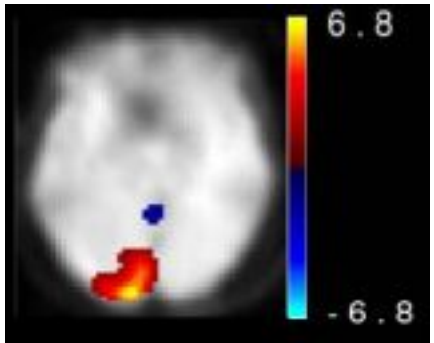


- Fit the model to the time course using a linear model

$$y(j) = \hat{\beta}_0 + \sum_{i=1}^{M} \hat{\beta}_i x_i(j) + e(j)$$



- Encode regression results into an activation map



GLM versus ICA

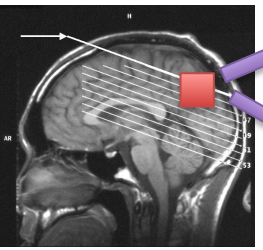
CognitiveSystems

GLM

Design matrix

Activation maps (for each column of G)

Data (X) = G . β



Time courses (supplied)

Mixing matrix

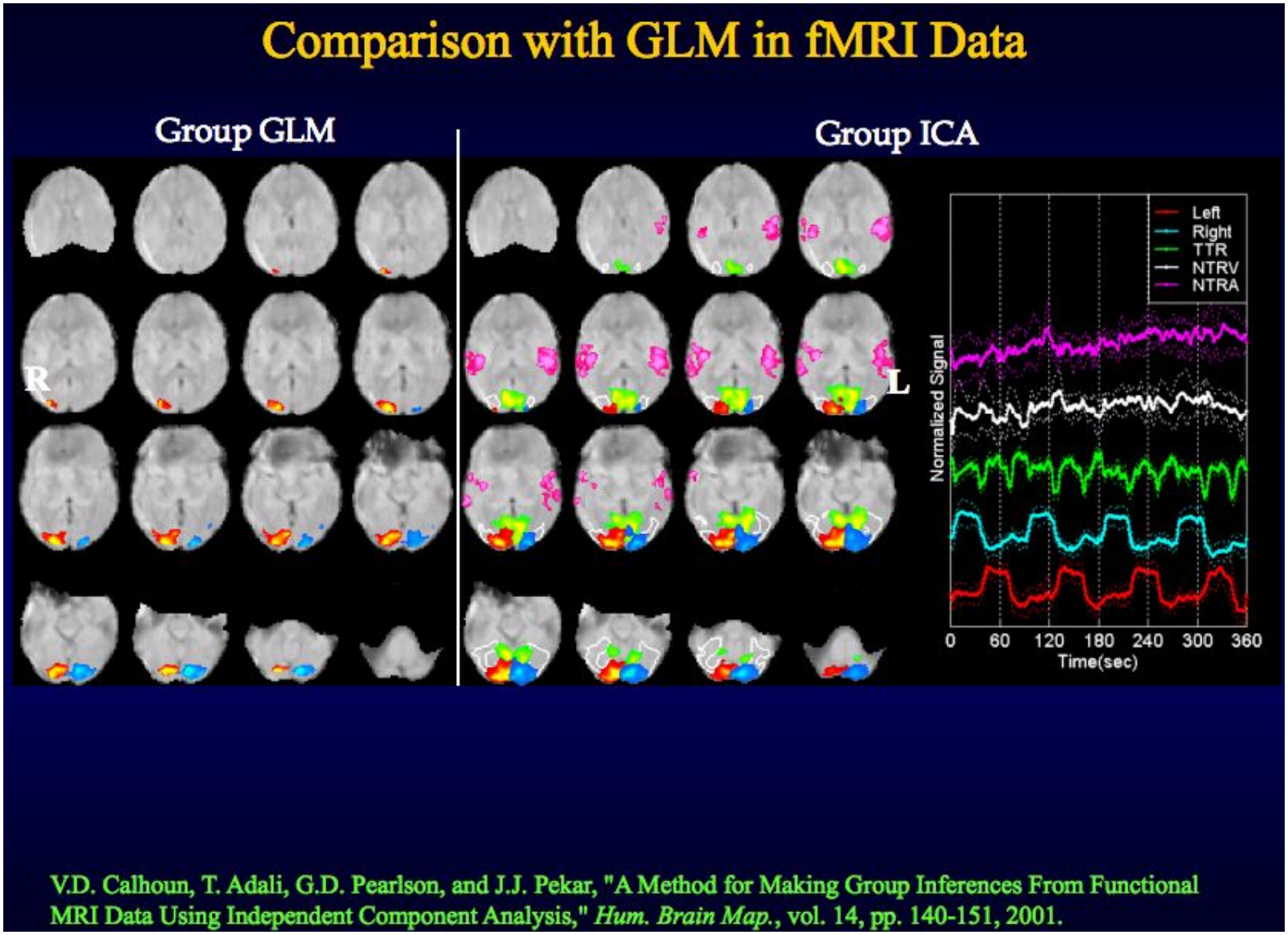
Spatially independent components

Data (X) = W^{-1} . Components (C)

Time courses (estimated)

ICA

Comparison of GLM and ICA



PCA vs ICA

CognitiveSystems

- Both methods are based on linear transformations
  - Compression
  - Classification
- PCA
  - Focus on uncorrelated (Gaussian) components
  - Second-order statistics (variance)
  - Orthogonal transformation
- ICA
  - Focus on independent and non-Gaussian components
  - Higher-order statistics (kurtosis)
  - Non-orthogonal transformation