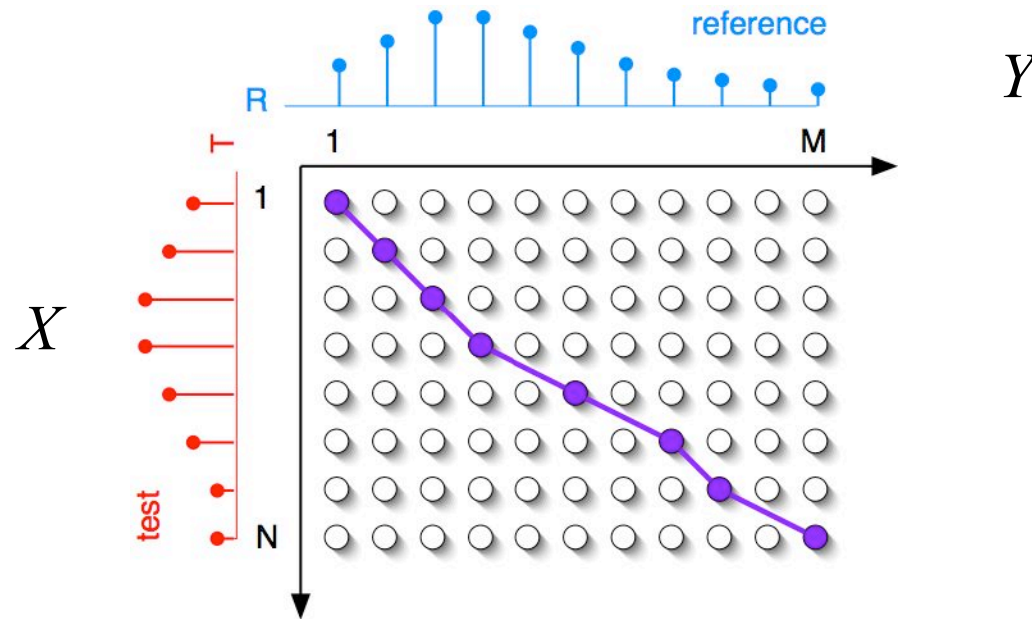


Part II: Alignment and Comparison

- Alignment
 - DTW
 - Matching what?
- Comparison
 - Features for comparison
 - Distance measures
- Audio Basis
 - Latent Semantic Analysis, PCA, ICA

Sequence Alignment



Find a warping function $c(k) = (i_k, j_k)$
that minimizes global error $D(X, Y) = \sum_{k=1}^K d(x_{i_k}, y_{j_k})$

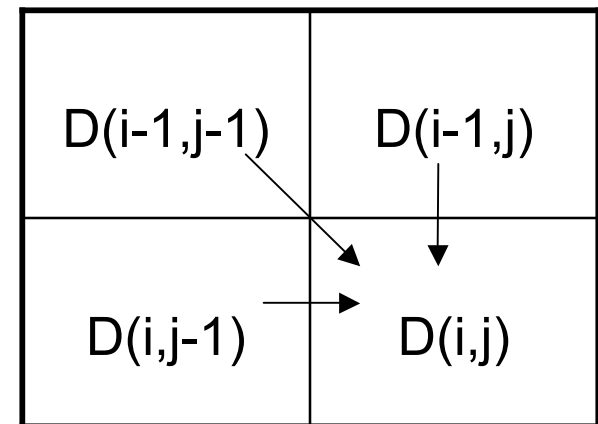
X, Y

- Audio: DTW
- Midi: Sequence Matching
- Mixed Midi & Audio : Score Alignment

$d(x, y)$

- Audio: Signal Distance
- Midi: Edit Distance
- Mixed Midi & Audio: Combined

$$D(i, j) = \min \begin{cases} D(i-1, j-1) + sub \\ D(i, j-1) + ins \\ D(i-1, j) + del \end{cases}$$



Principle:

MidiWavAlign.m

- Parse Midi into Score (Piano Roll)
- Extract Features from Audio
- Calculate $d(\text{Score}(i), \text{Feature}(j))$
- Find best alignment:
 - calculate $D(i, j)$
 - back-trace to find the warping function $c(i, j)$

Many types of alignment:

- GSA: Global Sequence Alignment
- LSA: Local Sequence Alignment
- LCS: Local Common Sequence
- ASM: Approximate String Matching
- OLM: Overlap Match
- DTW: Dynamic Time Warping
- TWLCS: Time Warp LCS

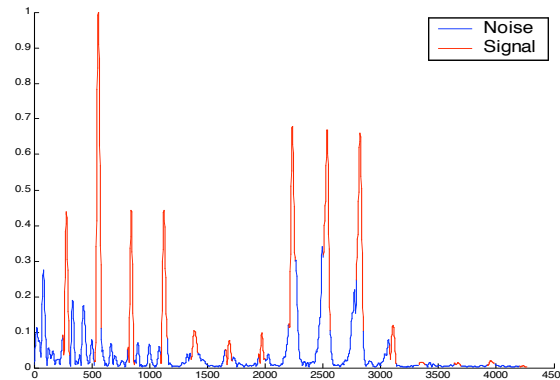
Align.m

Example of costs assignments:

- GSA: $D(i,j) = \min(D(i,j - 1) + 2, D(i - 1,j) + 2, D(i - 1,j - 1) + p)$, where if $X(i) \approx Y(j)$ then $p = -1$ else $p = 1$ and $D(i,0) = 2i$, for $i = 0..M$ and $D(0,j) = 2j$, for $j = 0..N$.
- LSA: $D(i,j) = \min(D(i,j - 1) + 2, D(i - 1,j) + 2, D(i - 1,j - 1) + p, 0)$, where if $X(i) \approx Y(j)$ then $p = -1$ else $p = 1$ and $D(i,0) = 0$, for $i = 0..M$ and $D(0,j) = 0$, for $j = 0..N$.
- LCS: $D(i,j) = \min(D(i,j - 1), D(i - 1,j), D(i - 1,j - 1) + p)$, where if $X(i) \approx Y(j)$ then $p = -1$ else $p = 0$ and $D(i,0) = 0$, for $i = 0..M$ and $D(0,j) = 0$, for $j = 0..N$.
- DTW: $D(i,j) = \min(D(i,j - 1) + \text{ins} * p(i,j), D(i - 1,j) + \text{del} * p(i,j), D(i - 1,j - 1) + p(i,j))$, where $p(i,j)$ is the dissimilarity between $X(i)$ and $Y(j)$ and $D(i,0) = i$, for $i = 0..M$ and $D(0,j) = j$, for $j = 0..N$.

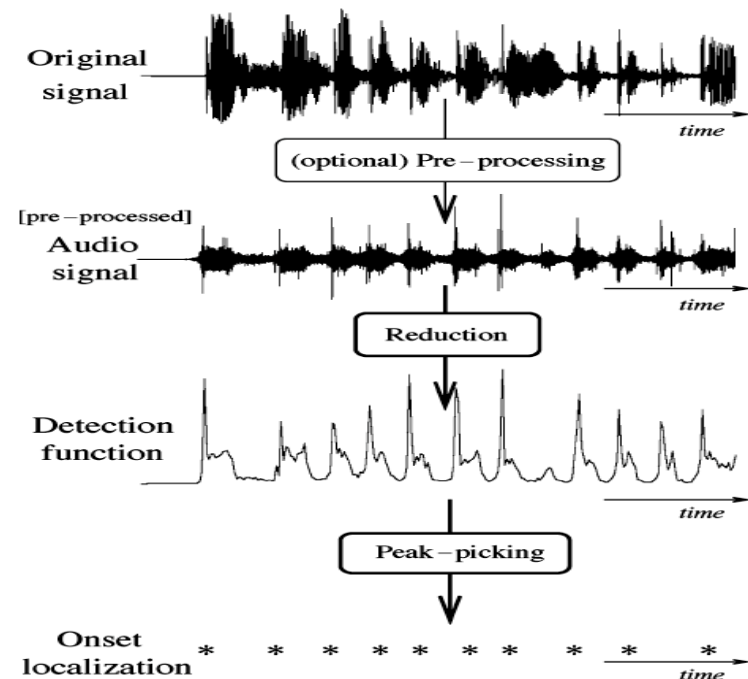
Matching functions

- Sustain matching
 - Harmonic model of note spectrum
 - $D(\text{Model}, \text{Signal})$



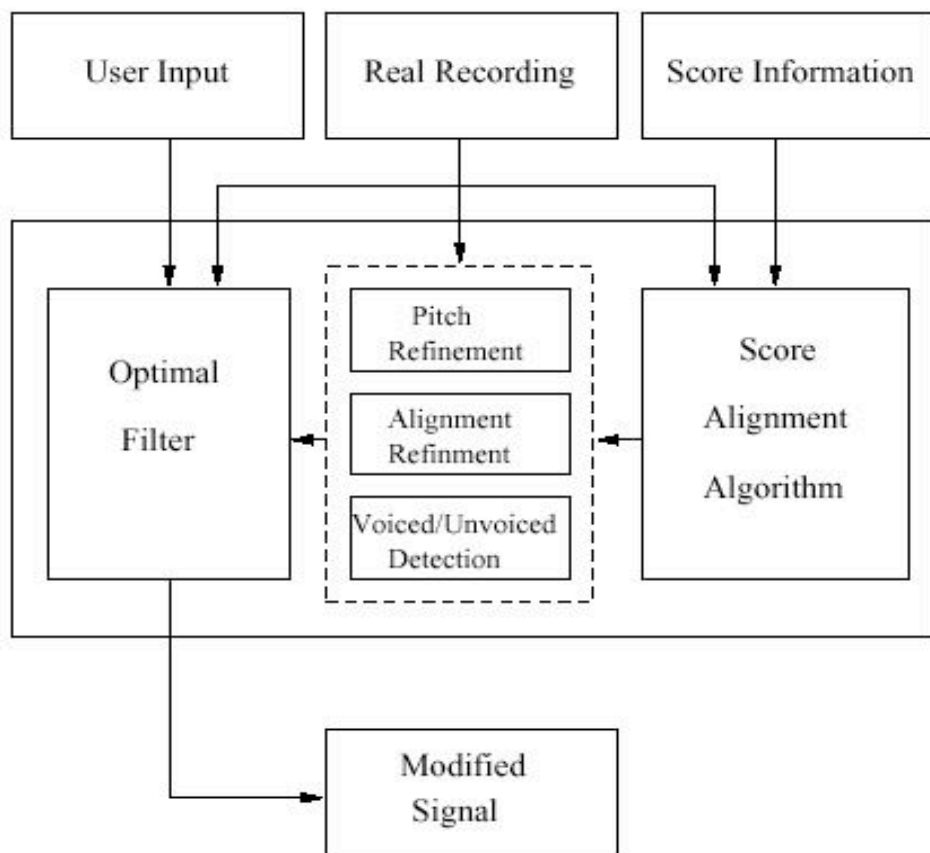
NoteLikelihood

- Onset detection
 - High frequency content
 - Spectral difference
 - Phase variation
 - Wavelet method
- Offset / Silence model



Onset

Example: score driven filtering



Mozart K454



Duo



Vln. only

Gershwin



All



Ella Fitzgerald

Audio Similarity

- Features
- Distances

Audio with Perceptual Features

<http://www.ofai.at/~elias.pampalk/ma/>

Speech

<http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html>

Audio Basis

[ABDist.m](#)

Features (speech)

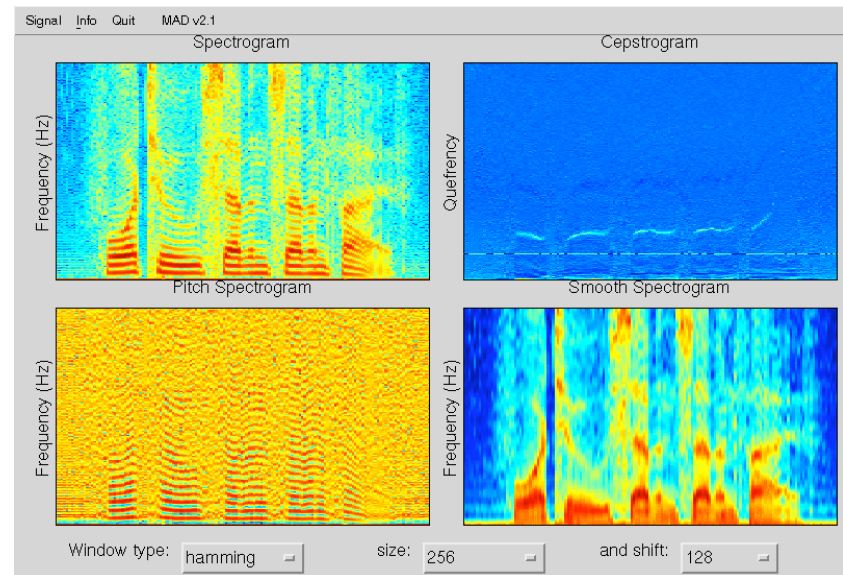
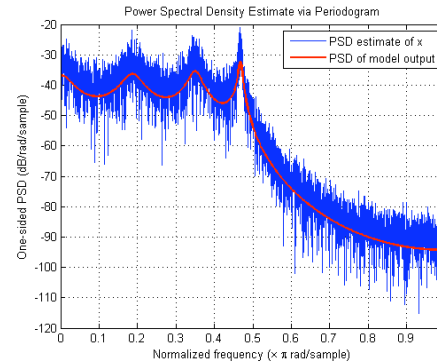
- AR coefficients
Fit linear AR filter

$$x(n) = \sum_{i=1}^p a_i x(n-1) + e(n)$$

- Cepstrum
Homomorphic Decomposition
and “liftering”

$$c(i) = \mathcal{F}^{-1} \{ \log(|\mathcal{F}\{x(n)\}|) \}$$

LPCAna.m, CepsAna.m



$$X(\omega) = \mathcal{F}\{x(n)\} = S(\omega) \cdot E(\omega)$$

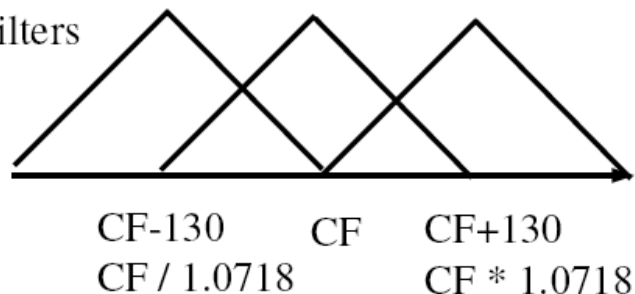
Features (perceptual audio)

- MFCC

Mel-scale

13 linearly-spaced filters

27 log-spaced filters



Mel-filtering

Log

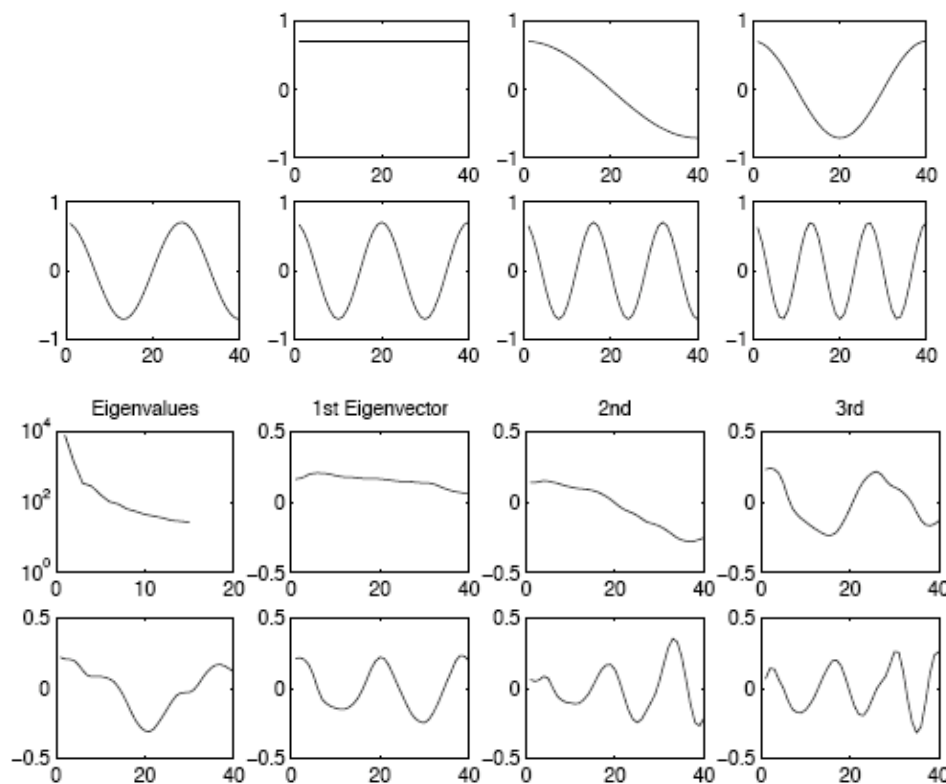
DCT

MFCCs

[mfcc.m](#)

Why MFCC?

- Perceptual Mel Frequency scale
- DCT is approx. optimal (PCA) transform of log-spectra



Distances

- AR -> Itakura-Saito

[distis.m](#)

$$D(S_x, S_y) = \int_{-\pi}^{\pi} \frac{d\omega}{2\pi} [e^{V(\omega)} - V(\omega) - 1], \quad V(\omega) = \log \frac{S_x(\omega)}{S_y(\omega)}$$

- Cepstrum, MFCC -> Euclidian

$$D(C_x, C_y) = \sum_{i=1}^p [C_x(i) - C_y(i)]^2$$

Similarity ~ dot product

[simm.m](#)

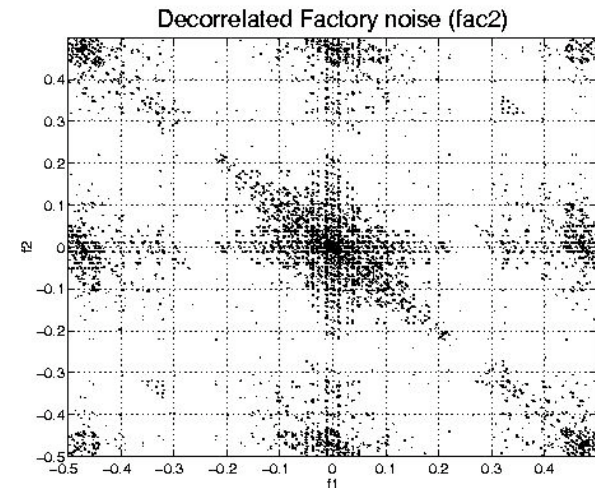
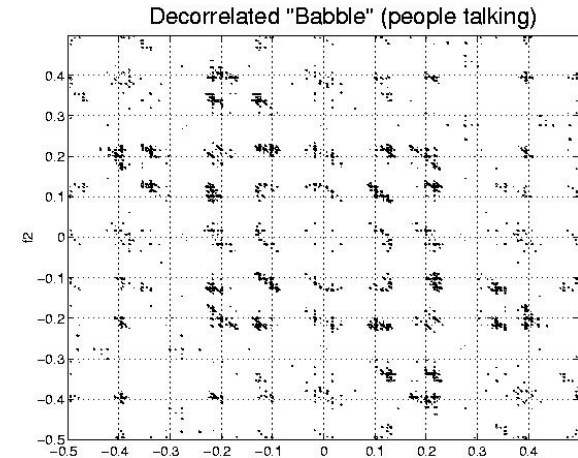
$$\text{Sim}(\hat{C}_x, \hat{C}_y) = 1 - \frac{1}{2} D(\hat{C}_x, \hat{C}_x) = \hat{C}_x \cdot \hat{C}_y, \quad \hat{C}_x = \frac{C_x}{\|C_x\|}$$

Other Measures

- Zwicker based measures
 - Bark-scale, Loudness in Sones
 - Statistics are derived from loudness measure at different frequencies
- Low level signal
 - RMS, Spectral centroid, bandwidth, zero-crossing, spectral roll-off, etc.
 - Static and dynamic features
- Bispectral features

$$B(\omega_1, \omega_2) = \lim_{N \rightarrow \infty} E \left\{ \frac{1}{N} X_N(\omega_1) X_N(\omega_2) X_N^*(\omega_1 + \omega_2) \right\}$$

- IS generalized to Bispectra
- Performs well for texture matching



Musical Content Features

- Chromagram

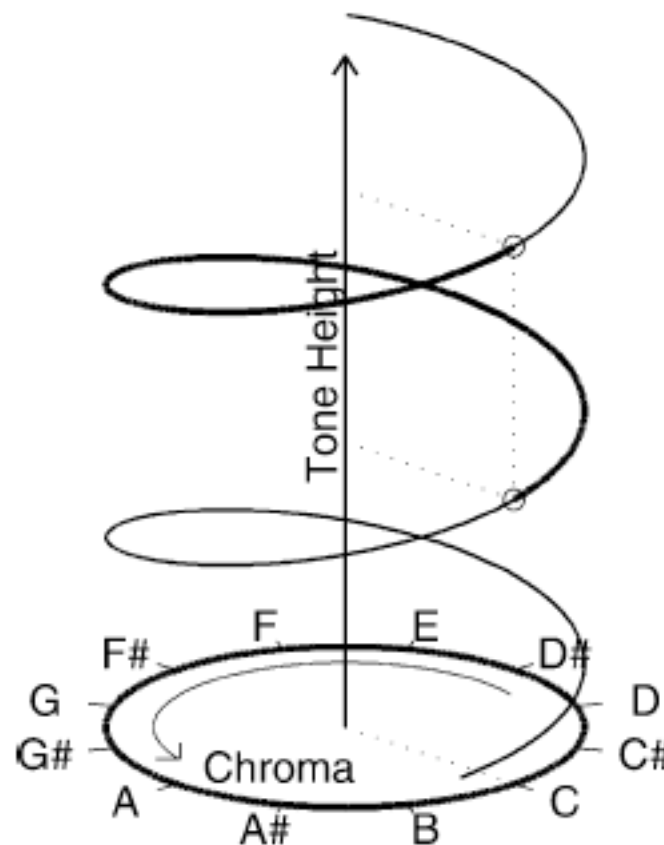
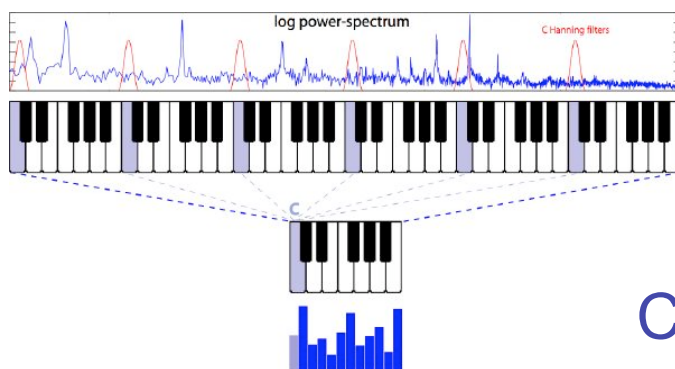
Frequency f

Height h

Chroma c

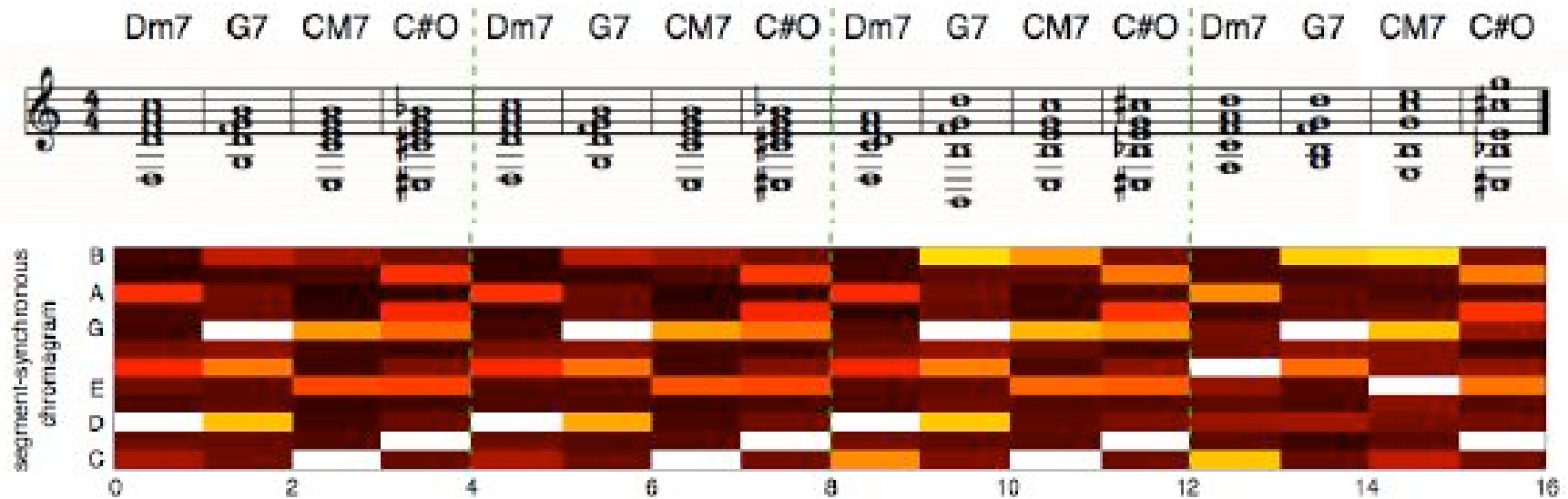
$$f = 2^{h+c}, \quad c \in [0,1) \text{ and } h \in \mathbb{Z}$$

$$c = \log_2 f - \lfloor \log_2 f \rfloor$$



Chroma.m

Chromagram example



Tristan Jehan, “**Creating Music by Listening**”, PhD thesis, MIT 2005

MARSYAS

Extraction of 30 features from 30-second audio tracks



Timbral Texture (19)

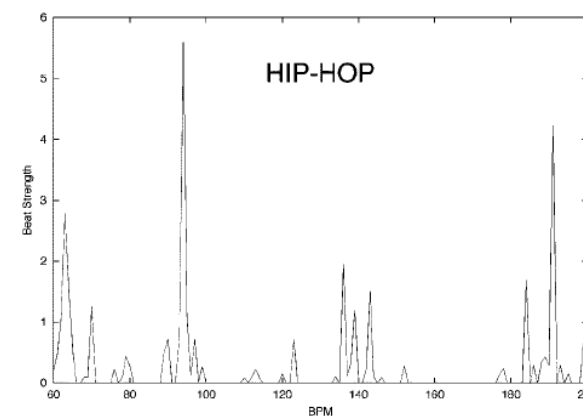
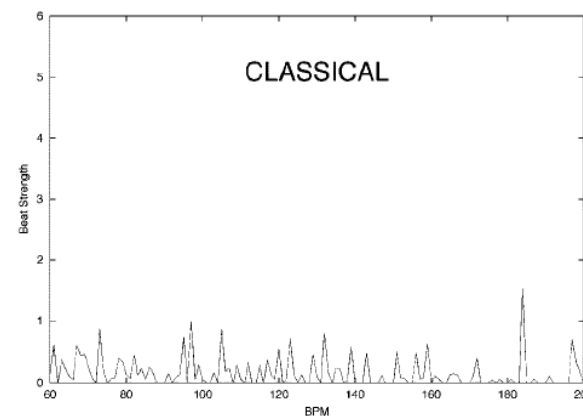
- Spectral Centroid – ‘brightness’ of sound
- Spectral Flux – local spectral change
- Zero Crossings – ‘noisiness’ of signal
- Low-Energy – amount of quiet time
- Mel-frequency Cepstral Coefficients (MFCC)

Rhythmic Content (6)

- Beat Strength, Amplitude, Tempo Analysis
 - Wavelet Transform: Frequencies of peaks, Relative amplitude of major peaks, Sum of all peaks

Pitch Content (5)

- Dominant Pitch , Pitch Intervals
 - Multipitch Detection Algorithm



Results

RBF networks: (Turnbull & Elkan 2005)
71% (std 1.5%)

- | | |
|-------------|----------|
| • Classical | • Rock |
| • Country | • Blues |
| • Disco | • Reggae |
| • Hip-Hop | • Pop |
| • Jazz | • Metal |

10 examples each

Human classification in similar experiment (Tzanetakis & Cook 2001):
70%

GMM with 3 Gaussians per class (Tzanetakis & Cook 2001):
61% (std 4%)

Support Vector Machine (SVM) (Li & Tzanetakis 2003):
69.1% (std 5.3%)

Linear Discriminant Analysis (LDA) (Li & Tzanetakis 2003):
71.1% (std 7.3%)

Audio Basis

- Geometric representation
- Feature matrix factorization
- Latent Semantic Analysis
- PCA versus ICA
- Audio Basis

The Model

- Geometric Representation $\bar{x} = A\bar{s}$

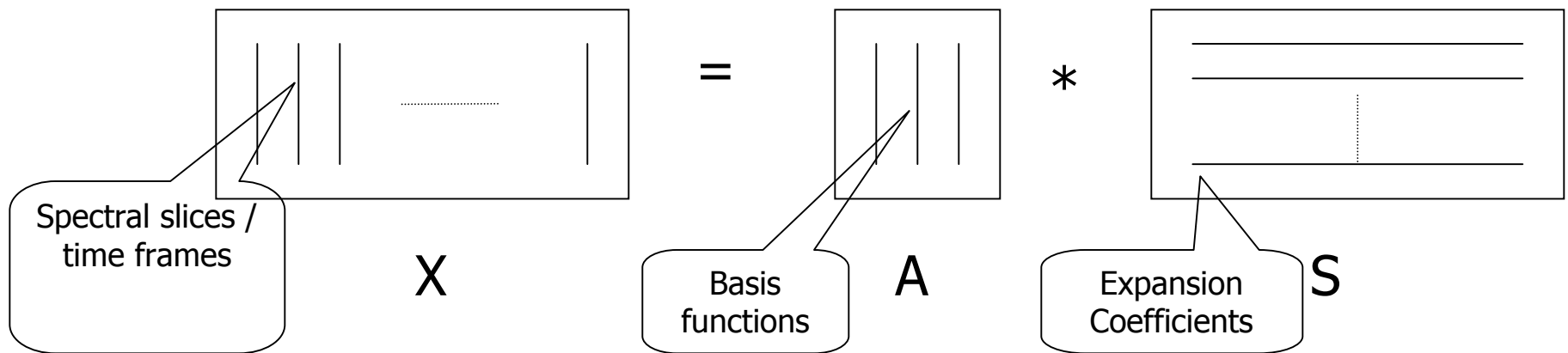
Samples: $\bar{x} = (x_1, x_2, \dots, x_n)^T$

Coefficients: $\bar{s} = (s_1, s_2, \dots, s_m)^T$

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = [\bar{a}_1 \bar{a}_2 \dots \bar{a}_m] \begin{bmatrix} s_1 \\ \vdots \\ s_m \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_n \end{bmatrix} \quad \begin{array}{l} \bar{n} = \bar{0} \\ m=n \end{array}$$

Geometric Representation

Finding a Basis



PCA & ICA

- PCA
 - Projects d -dimensional data onto a lower dimensional subspace in a way that is optimal in $\sum |\mathbf{x}_0 - \mathbf{x}|^2$ sense
 - Can be efficiently estimated using SVD
 - Used in Latent Semantic Indexing (LSI)
- ICA
 - Seek directions in signal / feature space such that resulting signals show independence.
 - Motivated by an idea the sound is a linear combination of independent “sound objects”

Latent Semantic Indexing

- problem #1: text - LSI: find 'concepts'

term	data	information	retrieval	brain	lung
document					
CS-TR1	1	1	1	0	0
CS-TR2	2	2	2	0	0
CS-TR3	1	1	1	0	0
CS-TR4	5	5	5	0	0
MED-TR1	0	0	0	2	2
MED-TR2	0	0	0	3	3
MED-TR3	0	0	0	1	1

LSI (cont.)

C. Faloutsos, icde01

- $X = U L V^T$

doc-to-concept
similarity matrix

CS-concept
MD-concept

	retrieval								
	inf. ↓		brain	lung					
	data								
↑	$\left[\begin{array}{cc} \text{CS} \\ \downarrow \\ \uparrow \\ \text{MD} \\ \downarrow \end{array} \right]$	$\left[\begin{array}{cc} 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 1 \end{array} \right]$	=	$\left[\begin{array}{cc} 0.18 & 0 \\ 0.36 & 0 \\ 0.18 & 0 \\ 0.90 & 0 \\ 0 & 0.53 \\ 0 & 0.80 \\ 0 & 0.27 \end{array} \right]$	x	$\left[\begin{array}{cc} 9.64 & 0 \\ 0 & 5.29 \end{array} \right]$	x	$\left[\begin{array}{ccccc} 0.58 & 0.58 & 0.58 & 0 & 0 \\ 0 & 0 & 0 & 0.71 & 0.71 \end{array} \right]$	

CS-concept (pointing to 0.18)
MD-concept (pointing to 0.18)

LSI (cont.)

C. Faloutsos, icde01

- $X = U L V^T$

‘strength’ of CS-concept

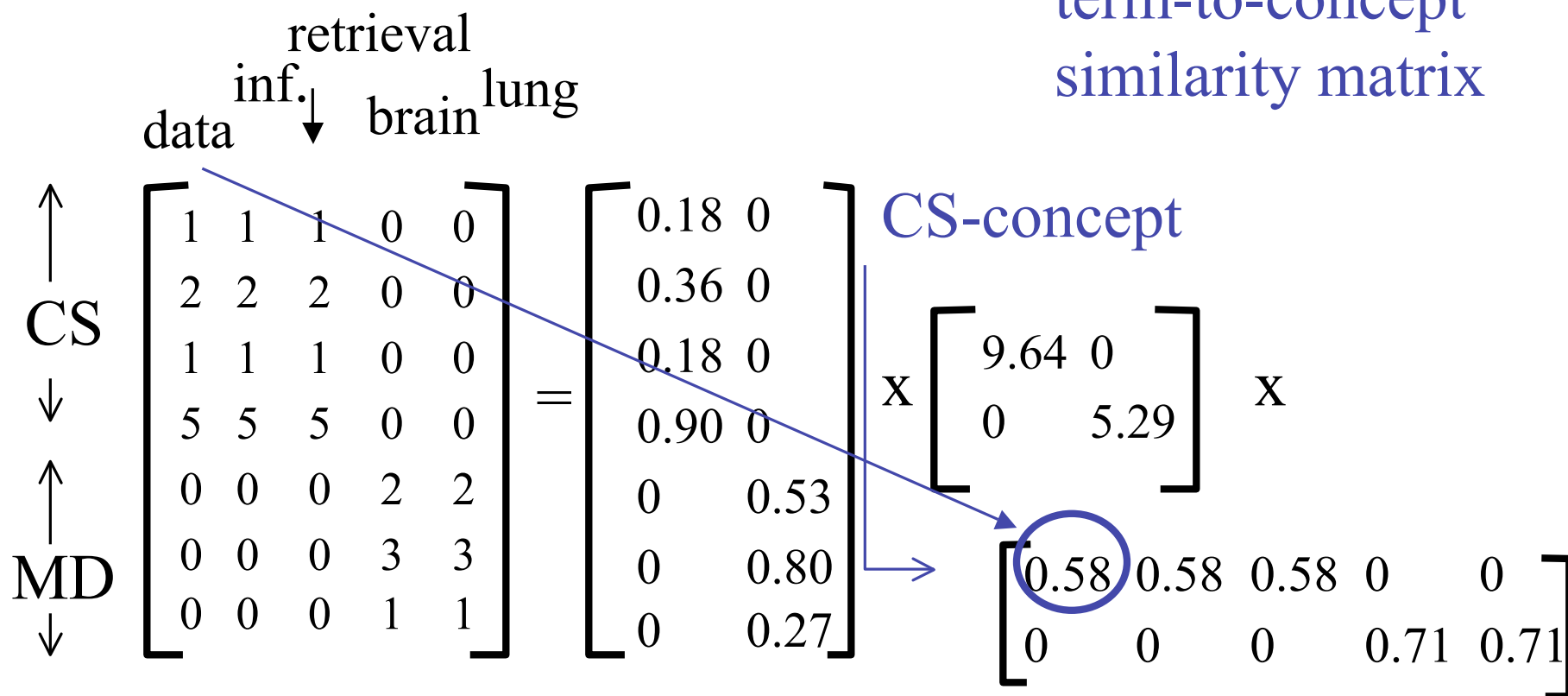
$$\begin{array}{c}
 \uparrow \\
 \text{CS} \\
 \downarrow \\
 \uparrow \\
 \text{MD} \\
 \downarrow
 \end{array}
 \begin{array}{c}
 \text{data} \\
 \text{inf.} \\
 \text{retrieval} \\
 \text{brain} \\
 \text{lung}
 \end{array}
 \begin{bmatrix}
 1 & 1 & 1 & 0 & 0 \\
 2 & 2 & 2 & 0 & 0 \\
 1 & 1 & 1 & 0 & 0 \\
 5 & 5 & 5 & 0 & 0 \\
 0 & 0 & 0 & 2 & 2 \\
 0 & 0 & 0 & 3 & 3 \\
 0 & 0 & 0 & 1 & 1
 \end{bmatrix}
 =
 \begin{bmatrix}
 0.18 & 0 \\
 0.36 & 0 \\
 0.18 & 0 \\
 0.90 & 0 \\
 0 & 0.53 \\
 0 & 0.80 \\
 0 & 0.27
 \end{bmatrix}
 \times
 \begin{bmatrix}
 9.64 & 0 \\
 0 & 5.29
 \end{bmatrix}
 \times
 \begin{bmatrix}
 0.58 & 0.58 & 0.58 & 0 & 0 \\
 0 & 0 & 0 & 0.71 & 0.71
 \end{bmatrix}$$

SVD - Example




C. Faloutsos, icde01

- $A = U L V^T$

term-to-concept
similarity matrix



LSI: Sound -Text analogy

- Document  Sound Frame
- Terms  Feature Values
- Concepts  Sound Objects

Examples – m spectral frames

Features – fft bins

Objects – spectral shapes

Summary: LSI Interpretation

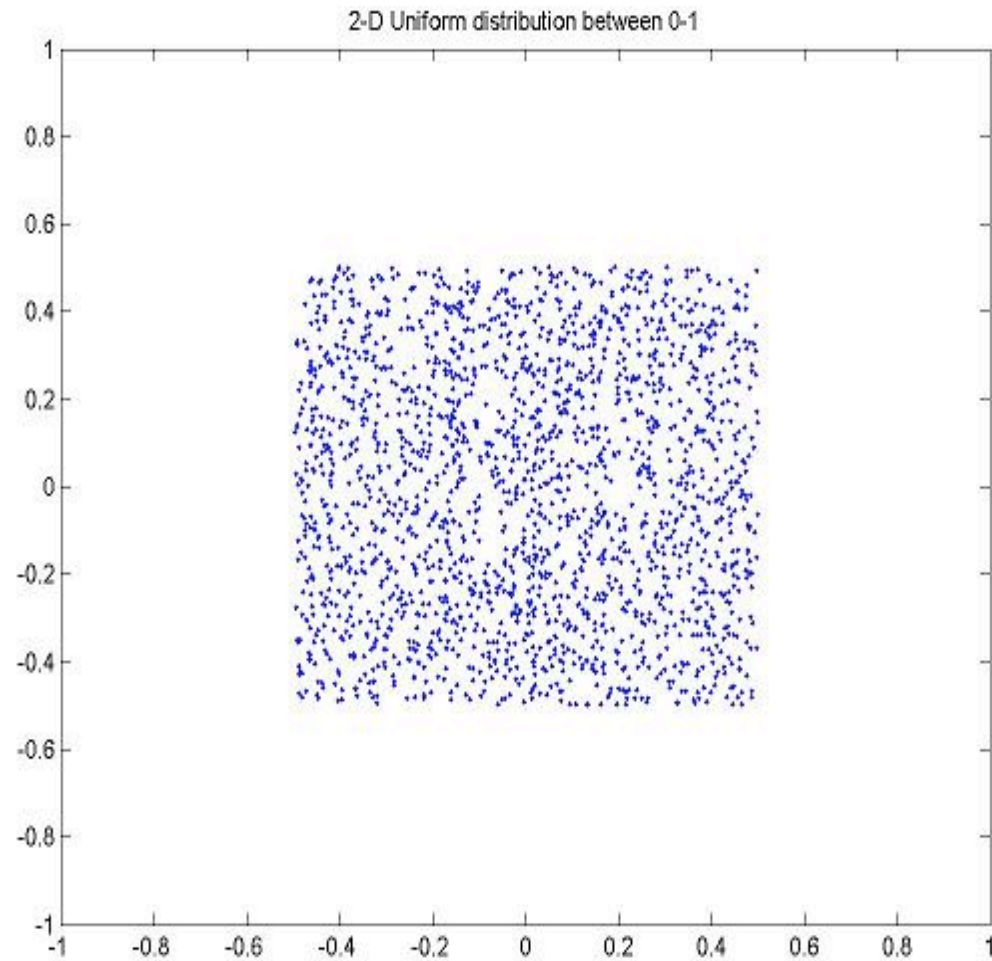
‘features’, ‘examples’ and ‘objects’:

- U : features-to-objects similarity matrix
- V : examples-to-object sim. matrix
- L : diagonal elements: ‘strength’ of each object

ICA vs. PCA

Let's generate 2 random coefficients from 2-D uniform distribution

$$s = [s_1, s_2]$$



ICA vs. PCA

Let A be the “objects” matrix

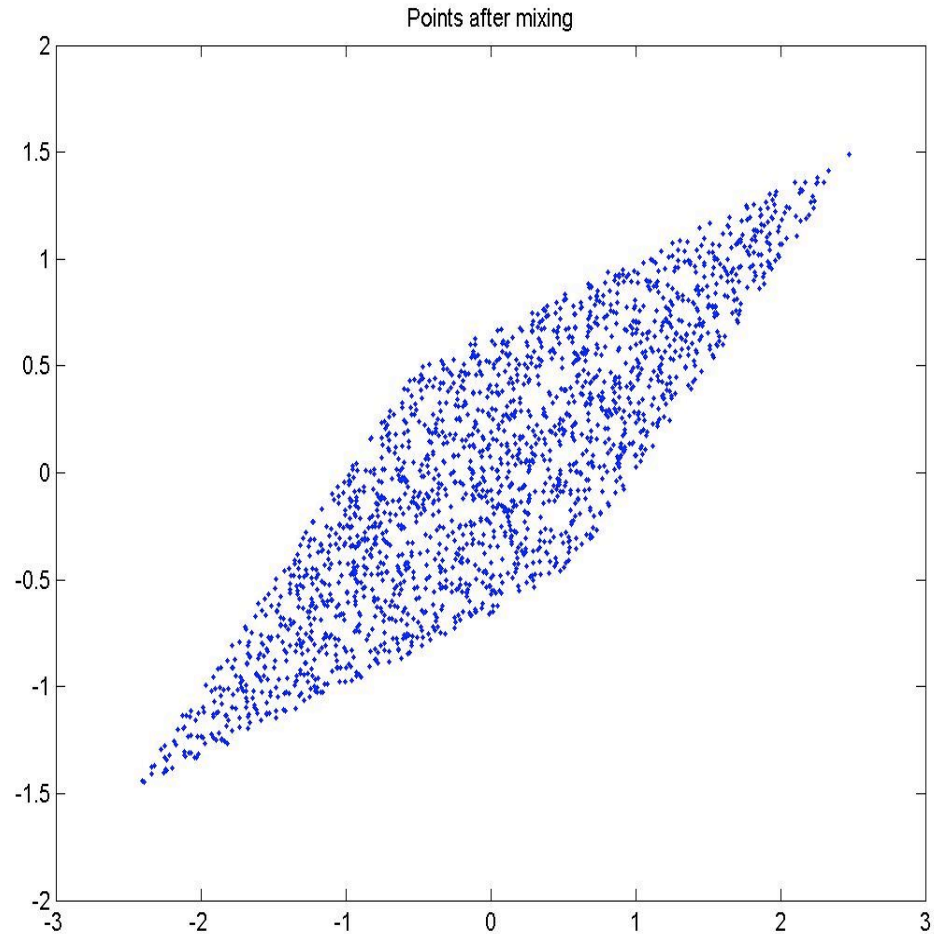
$$A = \begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix}$$

that results in features x

$$x = As$$

The features are no longer statistically independent

Are the components of x correlated one with another?



ICA vs. PCA

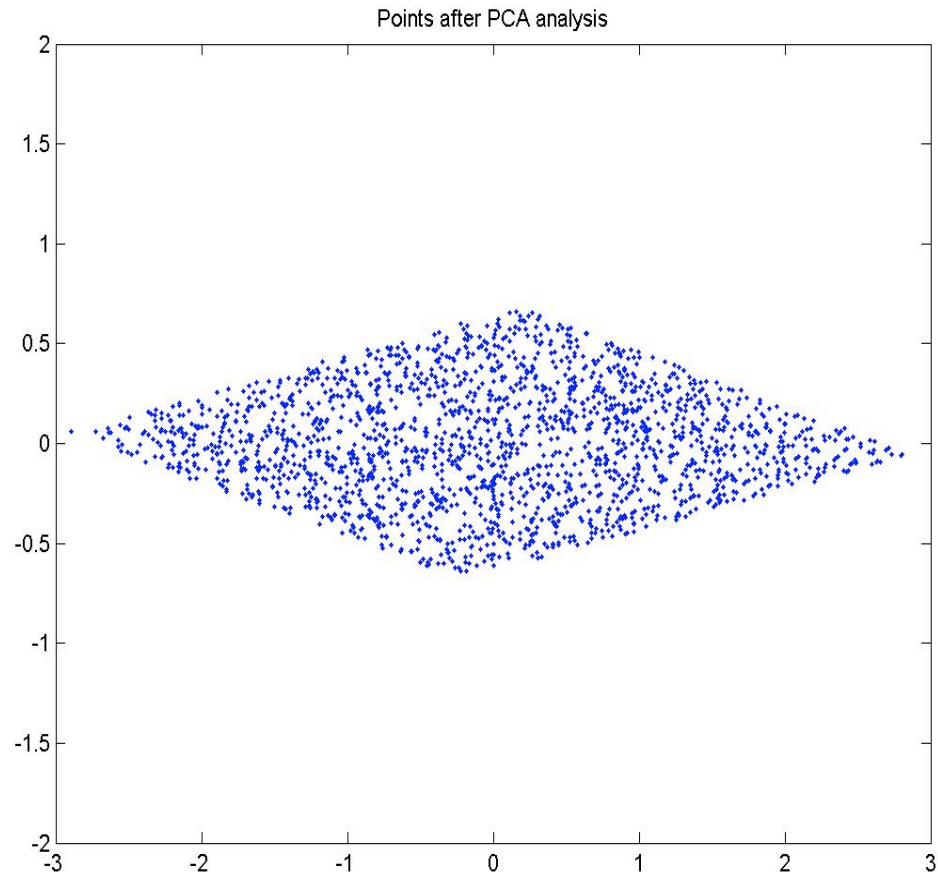
We apply PCA transform to \mathbf{x} and get a new vector \mathbf{y}

Are the components of \mathbf{y} statistically independent?

Are the components of \mathbf{y} correlated to one each other?

We conclude that PCA estimated a vector \mathbf{y} that has uncorrelated components.

However, it failed to estimate the independent components of the source \mathbf{x}

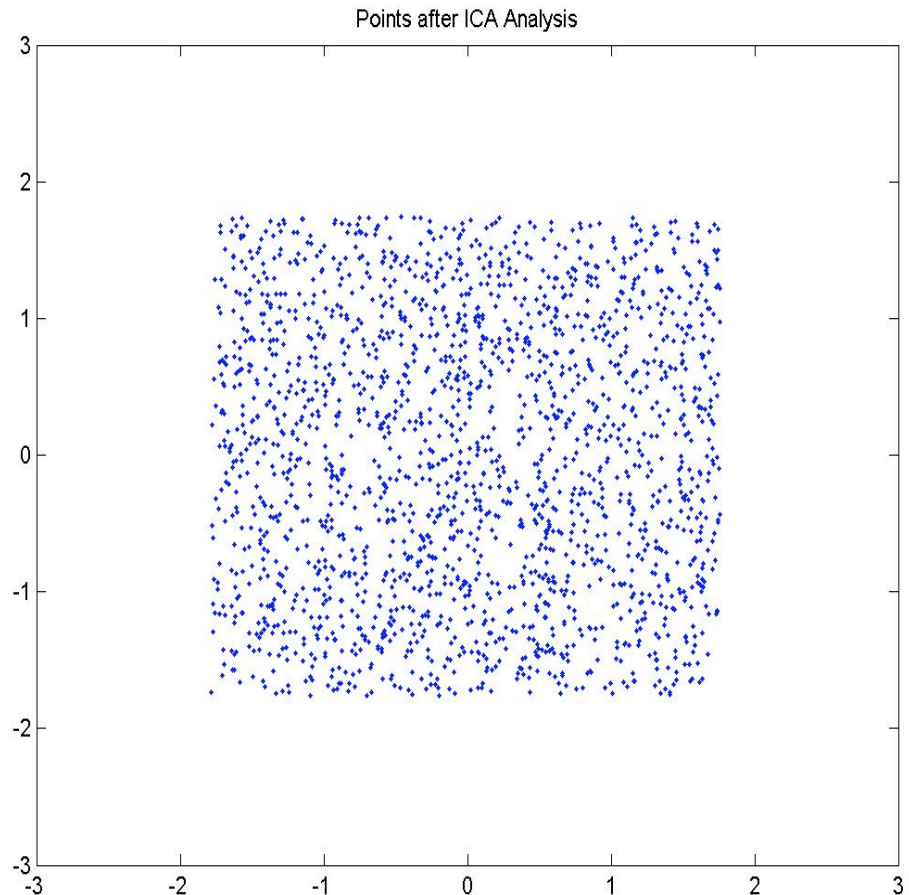


ICA vs. PCA

The points after ICA analysis.

We see that ICA has successfully estimated the independent components of the mixed sources.

The obvious question is how ICA does that? And why PCA fails to do it?



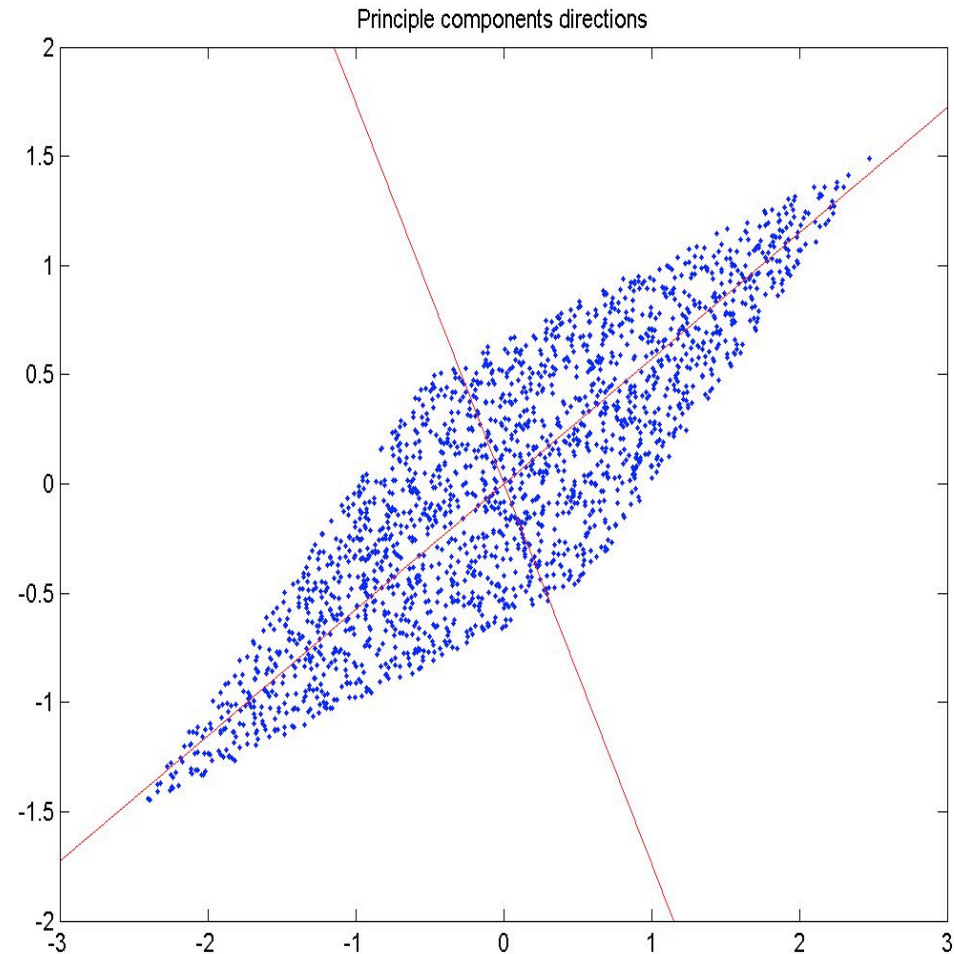
ICA vs. PCA

The directions of the principle components of x

The principle components directions are chosen so that they explain the maximum amount of variance of x

e.g. The first principle component:

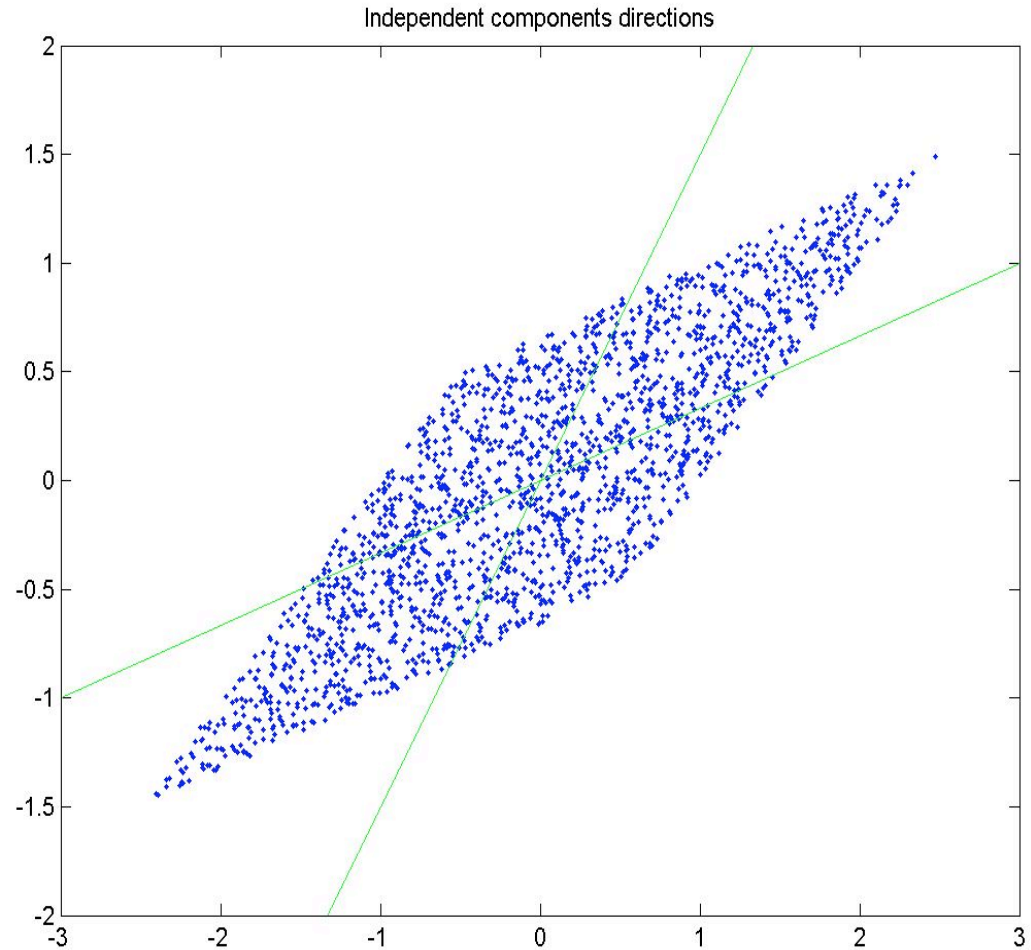
$$w_1 = \arg \max E\{(w^T x)^2\}$$



ICA vs. PCA

The directions on which ICA project x' to estimate the sources.

How does ICA choose these directions?

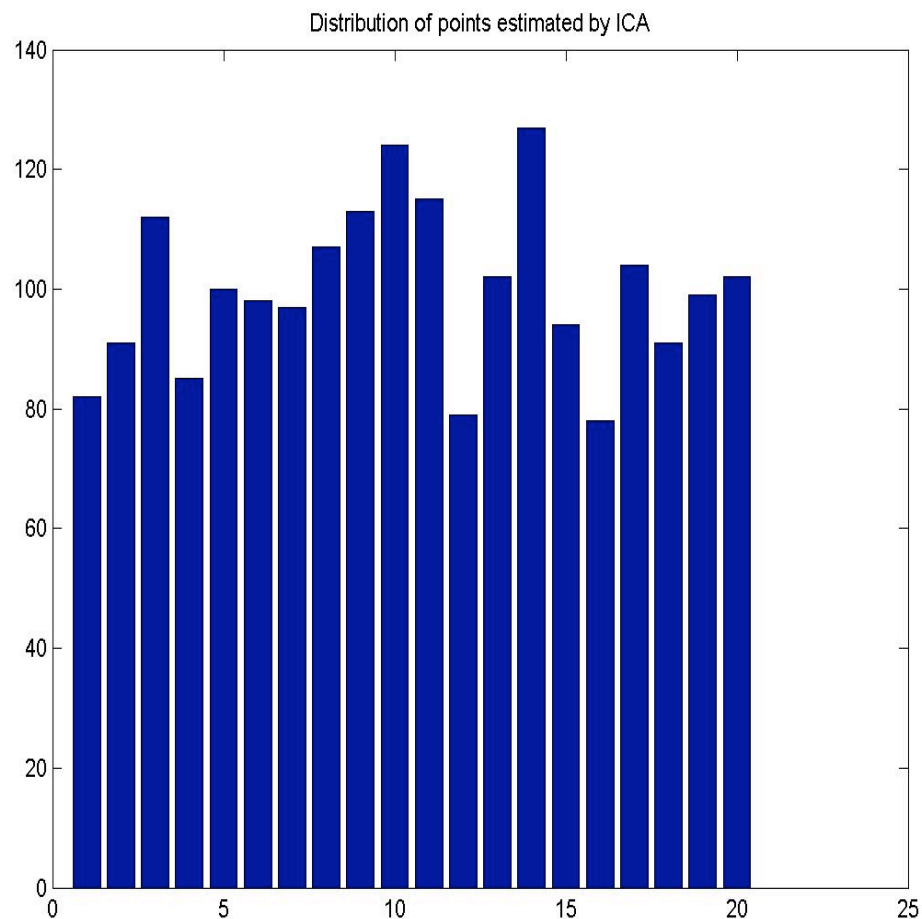


Principles of ICA estimation - Example

The distribution of one of the estimated independent components.

This distribution is the most non-gaussian one, that ICA found.

On the other hand..

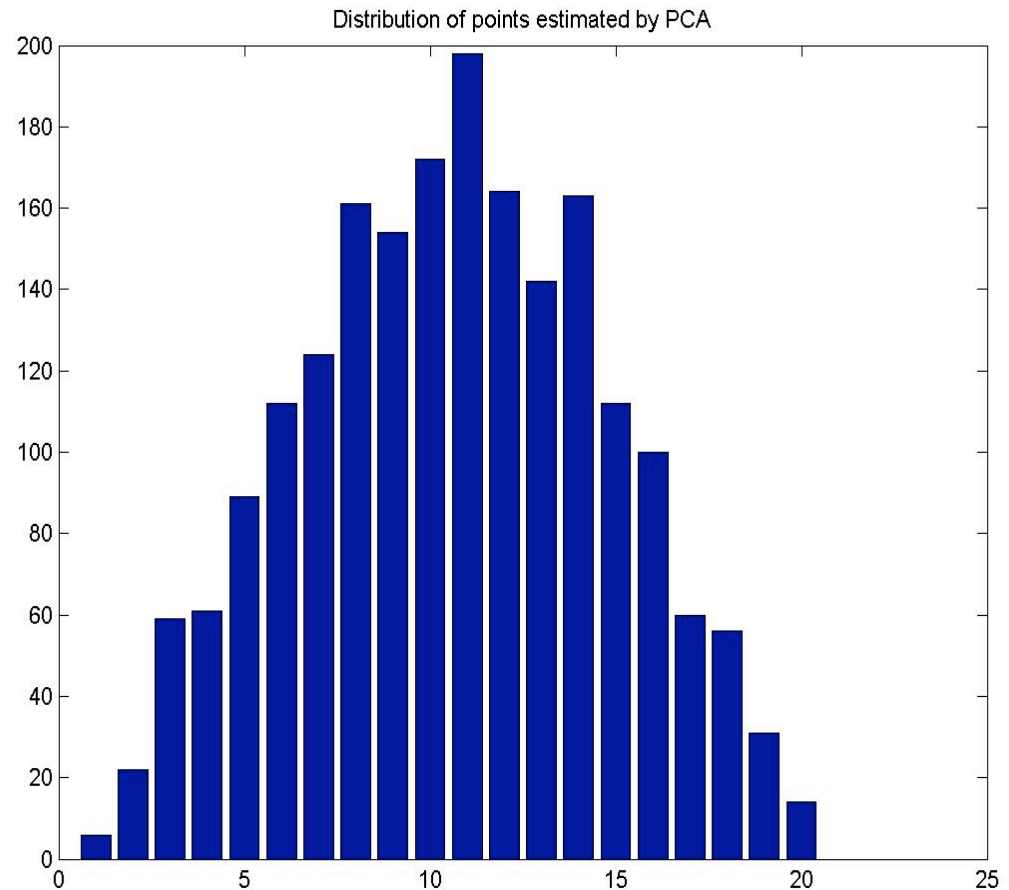


Principles of ICA estimation - Example

The Distribution of one of the components after PCA analysis.

By successive rotations of the PCA vectors, a local maximum non-Gaussian solution is found

The procedure is repeated for the next component



Audio Basis Algorithm

-Data Reduction (SVD) to r dimensions

$$(1) \quad \tilde{X} = U \Sigma V^T$$

$$(2) \quad \tilde{X} = X * V_r$$

- ICA

$$(3) \quad [A, W, S] = \text{ICA} (\tilde{X}^T)$$

- Independent Coefficients: $C_x = \tilde{X} * W^T$

- ICA matrix: $B = V_r * W^{-1}$

AudioBasis

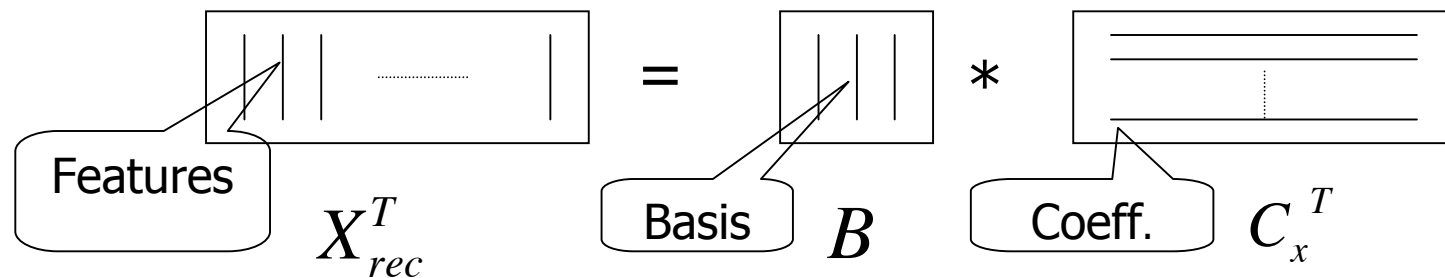
Audio Basis Algorithm (cont.)

We can use C_x , B for reconstruction of the data by:

$$(5) X_{rec} = \tilde{X} * V_r^T = \tilde{X} * W^T * (W^{-1})^T * V_r^T = C_x * B^T$$

or

$$X_{rec}^T = B * C_x^T \quad (\text{without ICA } X_{rec}^T = V_r * \tilde{X}^T)$$



AudioBasis.m

AB Distance

Likelihood of sound y given AB model $X_{rec}^T = B * C_x^T$

- Represent y in the basis of x

$$\tilde{Y}^T = B * C_y^T$$

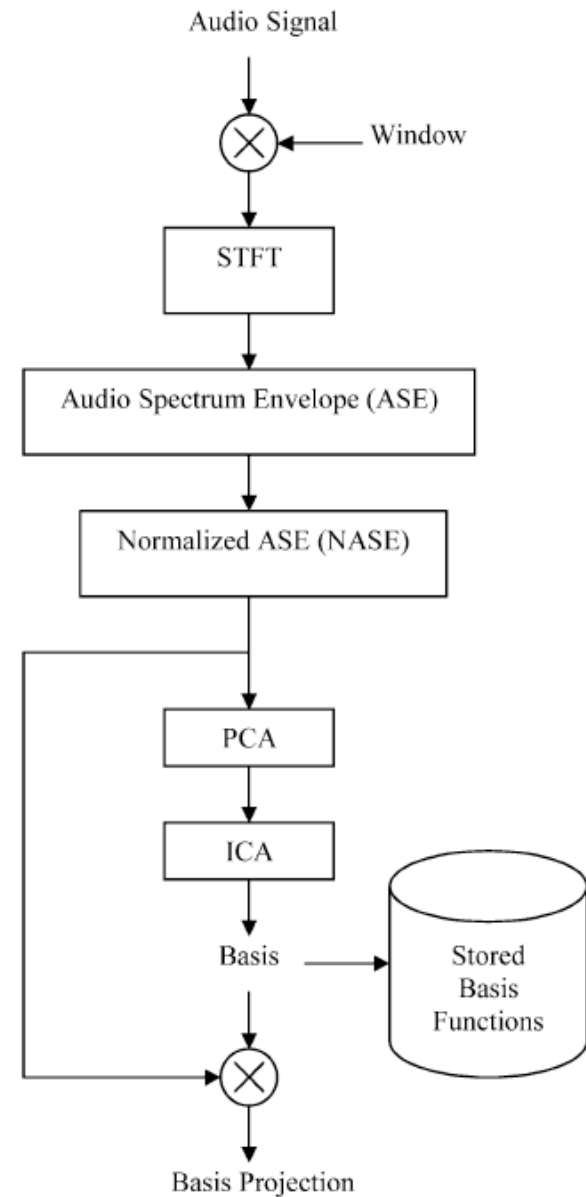
- Compare between C_y and C_x

$$Dist(C_y, C_x) = D[prob(C_y) \parallel prob(C_x)]$$

$$\approx \sum_i D[prob(C_y^i) \parallel prob(C_x^i)]$$

ABDist.m

```
% [L1,SX,SY,VY] = ABDist(x,y,ABType,DistType)
% Distance between sounds using Audio Basis
% Input:
% x - query sound
% y - reference sound
% ABtype - type of Audio Basis (AB):
% 'MAG' - Short time FFT magnitudes (default)
% 'ENV' - MPEG-7 AudioSpectrumEnvelope
%         (4 Bands Per Octave)
% 'ERB' - ERB Auditory filter Bank
% DistType- type of distance measure
% 'KL' - Kulback Liebler distance using GMM model
% 'Like' - log probability using GMM
% 'IS' - Itakura Saito Distance
```



MPEG7 matching with HMM

