Independent Component Analysis

Pattern Recognition

University of Groningen

Introduction

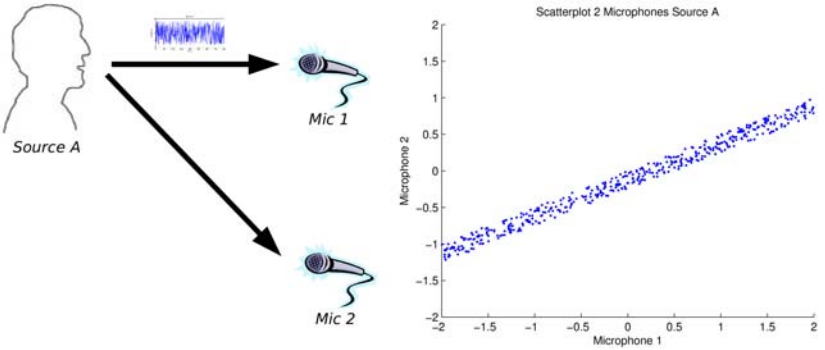
ICA

Independent Component Analysis (ICA) finds linear combinations of the original features, such that the new features are statistically independent. (Works for non-Gaussian data)

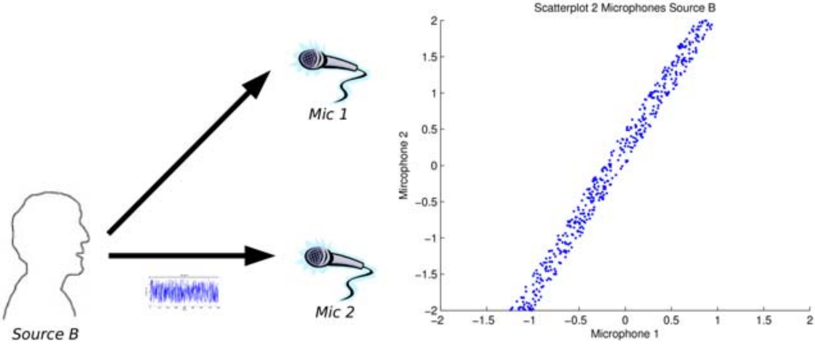
Example applications:

- Blind source separation (e.g. EEG data)
- Image feature extraction

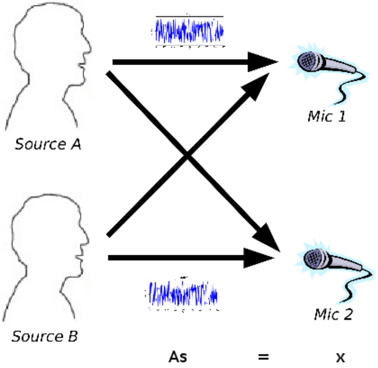
The cocktail party problem

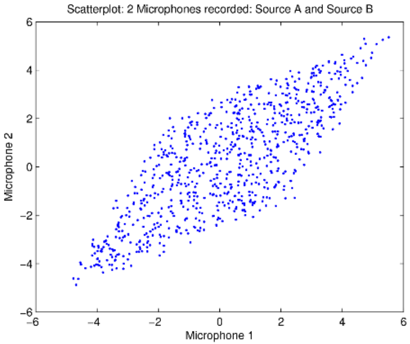


The cocktail party problem



The cocktail party problem





The cocktail party problem

Consider a data set $S = \{(s_1(t), s_2(t)) \mid i = 1, 2, ..., n\}$ formed by samples of the two **source speech signals**, and a data set $X = \{(x_1(t), x_2(t)) \mid i = 1, 2, ..., n\}$ formed by samples of the two **recorded signals**

The recorded signals are weighted sums of the source signals:

$$x_1(t) = a_{11}s_1(t) + a_{12}s_2(t)$$

$$x_2(t) = a_{21}s_1(t) + a_{22}s_2(t)$$

A depends on the distances from the source to the microphone

How do we estimate the original signals?

- If the parameters a_{ij} are known, then solve a system of linear equations
- Else, use ... ICA

With ICA the source signals $s_i(t)$ can be recovered from their mixtures $x_i(t)$, if the source signals are **statistically independent** and **non-Gaussian**.

Definition

- Assume observations of n variables $x_1, ..., x_n$ that are linear mixtures of m ($m <= n$) unknown independent variables $s_1, ..., s_m$ called the independent components:

$$x_j = a_{j1}s_j + ... + a_{jn}s_n \quad \text{for all } j$$

- The ICA model is given by:

$$\mathbf{x} = A\mathbf{s} \quad \text{or} \quad \mathbf{x} = \sum_{i=1}^{m} \mathbf{a}_i s_i$$

- Independent components are obtained by:

$$\mathbf{s} = W\mathbf{x} \quad \text{or} \quad \mathbf{s} = \sum_{i=1}^{n} \mathbf{w}_i x_i$$

where W is the inverse matrix of A

What is independence?

Definition

Two scalar-valued random variables $\mathbf{s}_1$ and $\mathbf{s}_2$ are said to be independent if information about the value of $\mathbf{s}_1$ does not give any information about the value of $\mathbf{s}_2$ and vice versa. $\mathbf{s}_1$ and $\mathbf{s}_2$ are independent if and only if the joint pdf is factorizable.

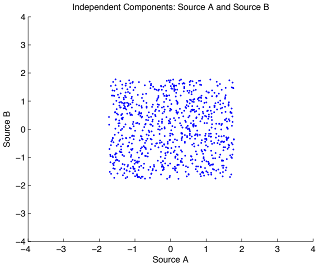


Figure: Independent components $\mathbf{s}_1$, s_2

PCA vs. ICA

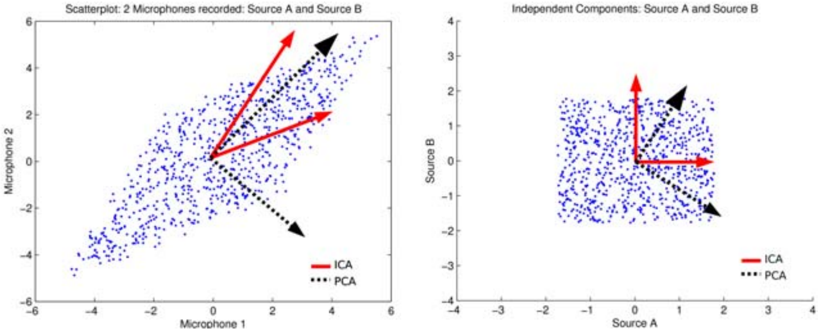
Similarity:

- both are linear transformations of the original features to new features

Differences:

- in PCA, the axes of the new coordinate system (black) are orthogonal; in ICA (red) they need not be
- goal of PCA: find a new orthogonal coordinate system (data representation), such that the covariance matrix is diagonal(i.e. the new features are uncorrelated)
- goal of ICA: find a new (not necessary orthogonal) coordinate system (data representation), such that the new features are statistically independent (the covariance matrix is diagonal, but statistically independent is a stronger condition than uncorrelated)
- in PCA, the first PC gives a new feature for which variance is maximal, but this feature needs not have any relation to an independent source variable

PCA vs. ICA


Scatterplot: 2 Microphones recorded: Source A and Source B
Microphone 1
Microphone 2
ICA
PCA
Independent Components: Source A and Source B
Source A
Source B
ICA
PCA

Independent components are maximally non-Gaussian

Assume that the independent components have non-Gaussian distributions.

According to the central limit theorem:
Sums of non-Gaussian random variables are closer to Gaussian then the original ones.

When we take a linear combination of the observed mixture variables, which corresponds to one independent component, the distribution of that component will be maximally non-Gaussian, compared to other (mixture) linear combinations

$$\mathbf{s} = \sum_i \mathbf{w}_i x_i$$

Mixing signals

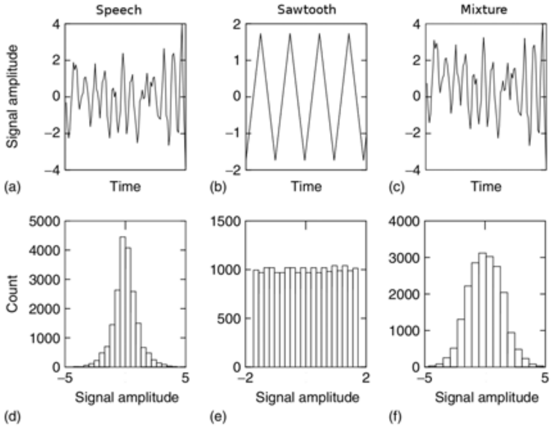


Figure: Signal mixtures have Gaussian (or normal) histograms.

Kurtosis for non-Gaussianity estimation

The classical measure of **non-Gaussianity** is kurtosis or the fourth-order cumulant. The kurtosis of y is defined as:

$$kurt(y) = \frac{E[(y-\mu)^4]}{\sigma^4} - 3$$

The kurtosis is:

- Zero for **Gaussian** y (mixed signal)
- Negative for **sub-Gaussian** y (sawtooth signal)
- Positive for **super-Gaussian** y (speech signal)

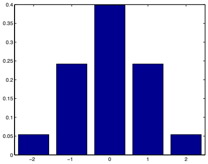
Drawback of kurtosis:
Very sensitive to outliers, therefore not a robust measure of non-Gaussianity

Kurtosis example

Kurtosis of a Gaussian distribution

y	-2	-1	-1	0	0	0	0	1	1	2

$\mu = 0 \ \sigma^2 = 1.3$

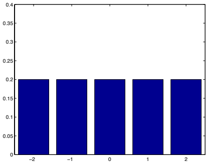


$$k(y) = \frac{\sum_{i=1}^{10} y_i^4}{(10-1)1.3^2} - 3 = 0.75$$

Kurtosis of an uniform distribution

y	-2	-1	0	1	2	2	1	0	-1	-2

$\mu = 0 \ \sigma^2 = 2.2$



$$k(y) = \frac{\sum_{i=1}^{10} y_i^4}{(10-1)2.2^2} - 3 = -1.5$$

Negentropy for non-Gaussianity estimation

Negentropy is based on the information theoretic quantity of (differential) entropy.

$$H(Y) = -\sum_{j} P(Y) log P(Y)$$

A **Gaussian** variable has the **largest entropy** among all random variables of equal variance. **Non-Gaussianity** is measured by:

$$J(y) = H(y_{gauss}) - H(y)$$

where y_{gauss} is a Gaussian random variable with the same covariance as y

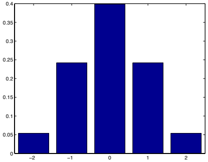
Drawback of Negentropy:
difficult to estimate the (non-parametric) pdf $P(Y)$.

Negentropy example

Negentropy of a Gaussian distribution:

$$P(y) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)$$
$$\mu = 0 \ \sigma^2 = 1.3$$
$$y_{gauss} = N(0, 1.3)$$

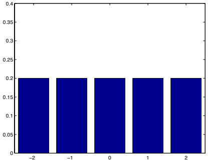
J(Y) = 28.54 - 24.95 = 3.59



Negentropy of an uniform distribution:

$$P(y) = \frac{1}{n}$$
$$\mu = 0 \ \sigma^2 = 2.2$$
$$y_{gauss} = N(0, 2.2)$$

J(Y) = 28.54 - 4.61 = 23.94



General Contrast Function

A general contrast function to evaluate the non-Gaussianity can be formulated as:

$$J(y) \approx [E\{G(y)\} - E\{G(y_{gauss})\}]^2$$

The following functions were proven (Hyvärinen) to be useful as **robust estimators** of G:

$$G_1(u) = \frac{1}{a_1} \log \cosh a_1 u, \;\; G_2(u) = -exp(\frac{-u^2}{2})$$

Whitening: parameter reduction

Whitening transforms x linearly into vector $\tilde{x}$, with uncorrelated components with unit variance. For example using eigen-value decomposition of the covariance matrix $E\{xx^T\} = EDE^T$

$$\tilde{x} = ED^{-1/2}E^T x,$$

where **E=orthogonal matrix** of **eigenvectors**, **D=diagonal matrix** of the **eigenvalues** and **x=random variable ($\mu = 0$)**.

$A = W^{-1}$ becomes an orthogonal mixing matrix $\tilde{A}$:

$$\tilde{x} = \tilde{A}y = ED^{-1/2}E^T Ay$$

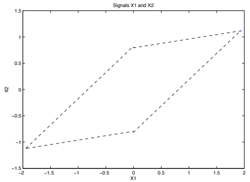
$$E\{\tilde{x}\tilde{x}^T\} = \tilde{A}E\{yy^T\}\tilde{A}^T = \tilde{A}\tilde{A}^T = I$$

only $n(n-1)/2$ instead of n^2 parameters have to be estimated

Whitening example

Consider two mixed signals $X1$ and $X2$:

X1	1.94	0.01	-0.01	-1.90	1.94	0.01
X2	1.11	0.80	-0.80	-1.11	1.11	0.80



1. Compute the covariance matrix A:

$$A = E[(X1 - \mu_1)(X2 - \mu_2)] =$$

2.1221	1.1734
1.1734	1.0071

2. Calculate the Eigenvalues λ and Eigenvectors V: $\mathrm{Av} = \lambda\ \mathrm{v}$

$$\lambda_1 = 0.2654 \ \lambda_2 = 2.8637$$

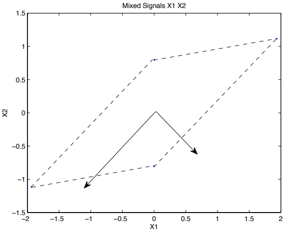
$$v1 = \begin{pmatrix} 0.534 \\ -0.845 \end{pmatrix} v2 = \begin{pmatrix} -0.845 \\ -0.534 \end{pmatrix}$$

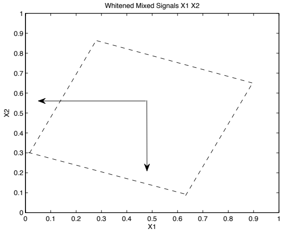
Whitening example

3. Transform the signals using Eigen-value decomposition:

$$\tilde{X} = V\lambda^{-1/2}V^{T}X$$

$\tilde{X}1$	1.93	-0.66	0.40	-1.46	1.19	-0.66
$\tilde{X}2$	0.28	1.03	-1.69	-0.93	0.28	1.03

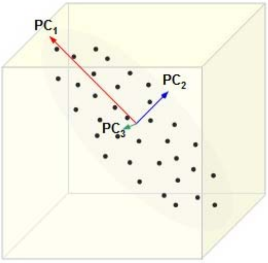




Principle component analysis: dimension reduction

In pre-processing **Principle Component Analysis** (Whitening) can be used to determine the **number of independent components** if the noise level is low.

The n largest principle components show the number of independent components. When having m sample points take at least: $n \leq \sqrt{m}$, because for every value in the $n * n$ weight matrix there must be at least one sample.



FastICA for one unit

FastICA can be used to derive a **single independent component**, it is based on a fixed-point iteration schema for finding the maximum of the non-Gaussianity of $w^T x$ based on finding the maxima of the general contrast function J.

The basic form of the FastICA algorithm is:

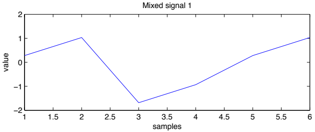
1. Choose an initial (random) weight vector w.
2. Update weights: $w^+ = E\{xg(w^T x)\} - E\{g'(w^T x)\}w$
3. Normalize w: $w = \frac{w^+}{||w^+||}$
4. If not converged, go back to 2.

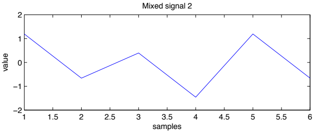
where g is the derivative of the contrast function G and g' the derivative of g

FastICA for one unit: Example

Consider the two pre-whitened mixed signals x1 and x2:

X1	1.93	-0.66	0.40	-1.46	1.19	-0.66
X2	0.28	1.03	-1.69	-0.93	0.28	1.03





1. Choose an random weight vector: w = [0.3273 0.1746]'
2. Update the weight vector using the learning rule:

$$g = tanh(X^T * w) \quad g' = 1 - g^2$$

$$w^+ = \frac{\sum X * g}{n} - \frac{\sum g'}{n} * w = [0.0090\ 0.0055]'$$

3. Normalize $w^+ = \frac{w^+}{||w^+||} = [0.8542\ 0.5199]'$

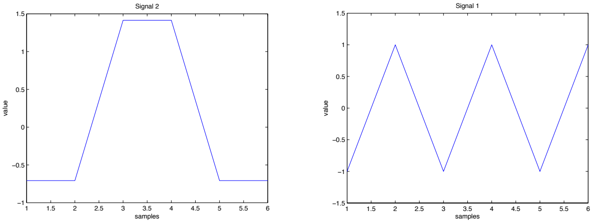
FastICA for one unit: Example

4.Check for convergence: $w \cdot w^{+} \approx 1$
Convergence after 4 iterations: w = [0.3760 0.9266]'

5. Obtain the estimated independent component: $y1 = wX$
Optionally: to obtain a different component, $y2$, re-start at 1

The estimated independent components Y are:

y1	0.71	0.71	-1.41	-1.41	0.71	0.71
y2	1	-1	1	-1	1	-1



FastICA for multiple units

To obtain **multiple independent components**, the one-unit FastICA algorithm can be used using **multiple weight vectors** $w_1, \ldots, w_n$.

To prevent different vectors from converging to the same **maxima**, the outputs $w_1^T x, \ldots, w_n^T x$ have to be **decorrelated** after every iteration.

If p independent components have been estimated then for $p+1$ do after **every iteration** of the one-unit FastICA algorithm:

1. $w_{p+1} = w_{p+1} - \sum_{j=1}^{p} w_{p+1}^T w_j w_j$
2. $w_{p+1} = w_{p+1} / \sqrt{w_{p+1}^T w_{p+1}}$

Properties of FastICA

The FastICA algorithm has compared to other (classical) ICA methods a number of desirable properties:

- Very fast convergence: cubic/quadratic
- Finds directly independent components without a known PDF
- Performance optimization by changing function G
- Independent components can be estimated one by one

Applications

- Separation of MEG and EEG data (biomedical signal processing)
- Finding hidden factors in financial data (economics)
- Telecommunications (parameter estimation)
- Reducing noise in natural images (gaussian noise removal)
- Image feature extraction (image processing)

Image Feature Extraction

Denote by X a vector of pixel gray levels from an image window.
It can be represented as a mixture (linear combination) of a set of some basis vectors (images of the same window size) $a_1, \ldots, a_n$, such that the coefficients $s_1, \ldots, s_n$ are (as) independent (as possible)

$$\mathbf{x} = s_1 \cdot \mathbf{a}_1 + s_2 \cdot \mathbf{a}_2 + \ldots + s_n \cdot \mathbf{a}_n$$

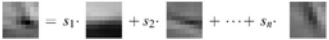
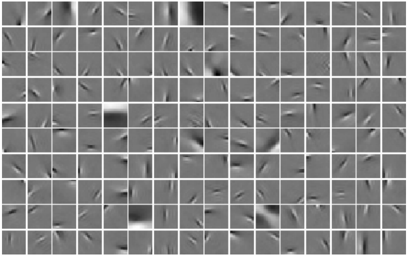


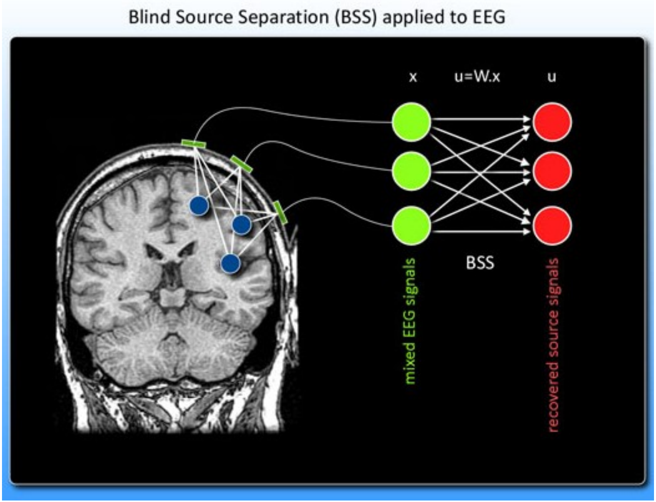
Image Feature Extraction

How do these basis images look like? ICA applied to natural images gives the following result:



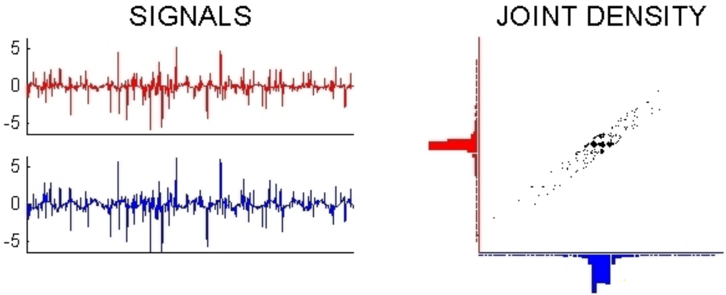
The resulting images are similar to the receptive field functions of neurons in the visual cortex. [DH Hubel, TN Wiesel, "Receptive fields and functional architecture of monkey striate cortex", The Journal of Physiology, 1968]

Separation of EEG data



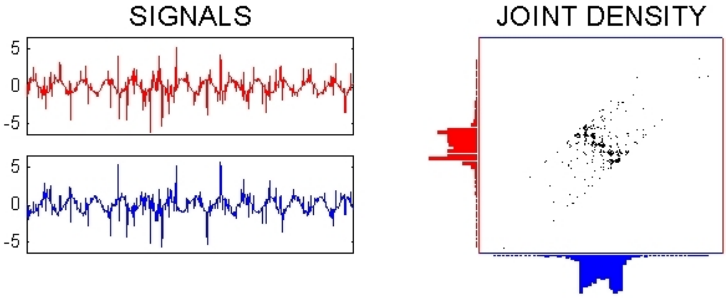
http://amouraux.webnode.com

Blind signal separation



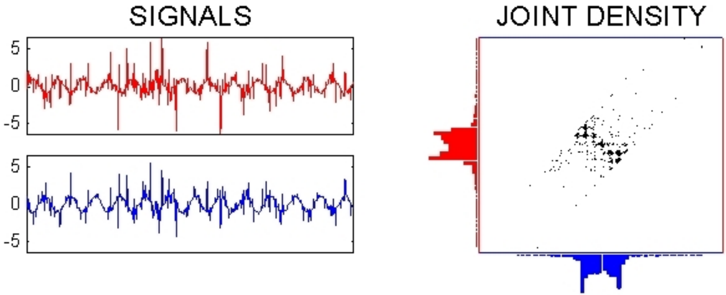
Input signals and density

Blind signal separation



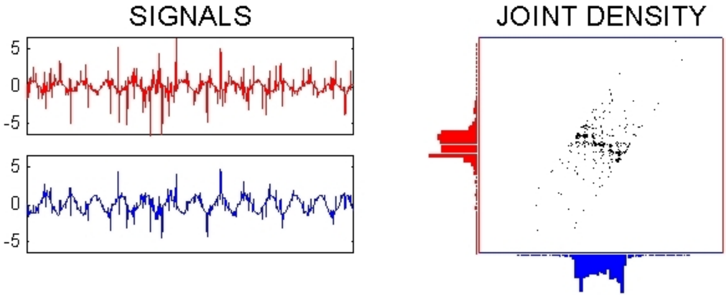
Whitened signals and density

Blind signal separation



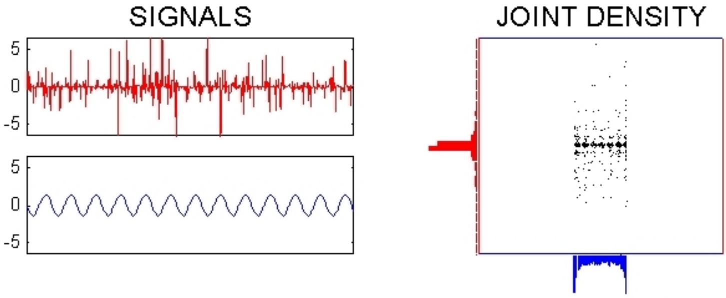
Separated signals after 1 step of FastICA

Blind signal separation



Separated signals after 2 steps of FastICA

Blind signal separation



Separated signals after 5 steps of FastICA

http://www.cis.hut.fi/projects/ica/icademo/

Limitations of ICA

- The **variances**, or **energies**, of the independent components can **not be determined**. This is because both s and A are unknown, any scalar multiplier in one of the sources s_i could always be cancelled by dividing the corresponding column $\mathbf{a}_i$ of the same scalar.

$$\mathbf{x} = \sum_{i=1}^{n} \mathbf{a}_i s_i$$

- The **order** of independent components can **not be determined**.

- Not all independent components can be derived if the amount of sources is larger then the number of observed mixtures X.

Summary

ICA is a general-purpose statistical technique in which observed random data are **linearly transformed** into components that are **maximally independent** from each other.

Maximum non-Gaussianity can be used to derive **different objective functions** whose optimization enables the estimation of the ICA model.

The **maximum likelihood estimation** or **minimization of mutual information** can be used to estimate ICA.

A **computationally efficient** method for performing the estimations is given by the **FastICA** algorithm.

References

R.O. Duda, E. Hart, G. Stortk, "*Pattern Classification*", ISBN10: 041056693

A. Hyvärinen, E. Oja,"*Independent Component Analysis: Algorithms and Applications*", Neural Networks, 13(4-5):411-430, 2000

A. Hyvärinen, "*Survey on Independent Component Analysis*",Neural Computing Surveys, Vol. 2 (1999), pp. 94-128.

A. Hyvärinen, "*Fast and Robust Fixed-Point Algorithms for Independent Component Analysis*", IEEE Trans. on Neural Networks, 10(3):626-634, 1999.

J. V. Stone, "*Independent Component Analysis*", Encyclopedia of Statistics in Behavioral Science, Volume 2, pp. 907-912. 2005