Independent Component Analysis

- Independent Component Analysis (ICA)
  - Also called: "blind source separation"
- Suppose *m* independent signals are mixed, and sensed by *m* independent sensors.
  - Cocktail party with speakers and microphones [demo]
  - EEG with brain wave sources and sensors
    - Brain Computer Interface videos: [demo1, demo2, demo3].
  - etc.
- Can we reconstruct the original signals, given the mixed data from the sensors?

Independent Component Analysis

- The sources **s** must be independent.
  - And they must be non-Gaussian.
  - (If Gaussian, then there is no way to find unique independent components.)
- Linear mixing to get the sensor signals **x**.
  - $\mathbf{x} = \mathbf{A}\mathbf{s}$
  - *or* $\mathbf{s} = \mathbf{W}\mathbf{x}$ (i.e., $\mathbf{W} = \mathbf{A}^{-1}$)
- **A** is called bases; **W** is called filters

Algorithm for ICA

- There are several formulations of ICA:
  - **Maximum likelihood**
  - Maximizing non-Gaussianity

Maximum-likelihood

- Maximum likelihood learning for W
  - By definition, the sources are independent

$$p(\mathbf{s}) = \prod_{j=1}^{m} p_s(s_j)$$

  - Then, the observed data distribution is given as:

$$p(\mathbf{x}) = \prod_{j=1}^{m} p_s(\mathbf{w}_j^T \mathbf{x}) \cdot |W|$$

  - We model CDF of source distribution as sigmoid:

$$\int_{-\infty}^{s} p_s(s')\, ds' = g(s) \rightarrow p_s(s) = g'(s)$$

$$g(s) = 1/\left(1 + e^{-s}\right) \qquad\qquad = g(s)(1 - g(s))$$

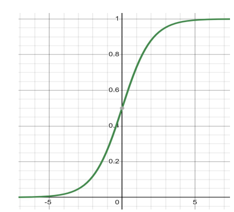
Use "change of variables" trick given:

$$\mathbf{s} = \mathbf{W}\mathbf{x}$$

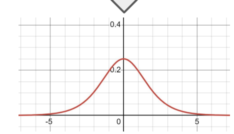
$$p(\mathbf{x}) Vol(d\mathbf{x}) = p(\mathbf{s}) Vol(d\mathbf{s})$$

$$\begin{aligned} p(\mathbf{x})|d\mathbf{x}| &= p(\mathbf{s})|d\mathbf{s}| \\ &= p(\mathbf{s})|W d\mathbf{x}| \\ &= p(\mathbf{s})|W| \cdot |d\mathbf{x}| \end{aligned}$$

$$p(\mathbf{x}) = p(\mathbf{s})|W|$$

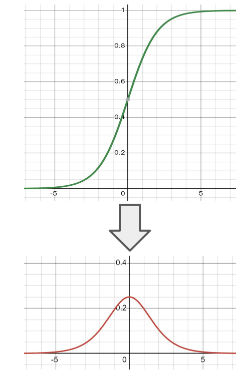


CDF g(s)



PDF g'(s)

Maximum-likelihood (cont'd)



CDF g(s)

PDF g'(s)

- Maximum likelihood learning for W
  - We model CDF of source distribution as sigmoid:

$$p_s(s) = g'(s) = g(s)(1 - g(s)) \qquad g(s) = 1/\left(1 + e^{-s}\right)$$

  - Our loss is the log-likelihood of data

$$\ell(W) = \sum_{i=1}^{N} \left(\sum_{j=1}^{m} \log g'(\mathbf{w}_j^\top \mathbf{x}^{(i)}) + \log |W| \right)$$

Maximum-likelihood (cont'd)

- Maximum likelihood learning for W
  - To get the update rule,

$$\ell(W) = \sum_{i=1}^{N} \left(\sum_{j=1}^{m} \log g'(\mathbf{w}_j^\top \mathbf{x}^{(i)}) + \log |W| \right)$$

  - SGD by taking derivative and using $\nabla_W |W| = |W| \left(W^{-1}\right)^\top$

$$W := W + \alpha \left(\begin{bmatrix} 1 - 2g(\mathbf{w}_1^\top \mathbf{x}^{(i)}) \\ 1 - 2g(\mathbf{w}_2^\top \mathbf{x}^{(i)}) \\ \vdots \\ 1 - 2g(\mathbf{w}_m^\top \mathbf{x}^{(i)}) \end{bmatrix} \mathbf{x}^{(i)\top} + \left(W^\top\right)^{-1} \right)$$

Algorithm for ICA

- There are several formulations of ICA:
  - Maximum likelihood
  - **Maximizing non-Gaussianity**

ICA by Maximizing non-Gaussianity

- Common steps of ICA (e.g., FastICA):
  - Apply PCA whitening (aka sphering) to the data
  - Find orthogonal unit vectors along which that the non-Gaussianity are maximized

$$\max_{W} L(W\widetilde{\mathbf{x}})$$
$$\text{s.t.} \quad WW^\top = I$$

  - where L(x) can be Kurtosis, L1-norm, etc.

PCA Whitening

- To whiten the input data,
  - We want a linear transformation

$$\widetilde{\mathbf{x}} = \mathbf{V}\mathbf{x}$$

  - So the components are uncorrelated:

$$\mathbb{E}\left[\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}^\top\right] = \mathbf{I}$$

  - From PCA transformation matrix, $\Sigma = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$
    - We can use

$$\mathbf{V} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top$$

  - Because

$$\mathbb{E}\left[\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}^\top\right] = \mathbb{E}\left[\mathbf{V}\mathbf{x}\mathbf{x}^\top\mathbf{V}^\top\right] = \mathbf{I}$$

Maximizing non-Gaussianity

- Kurtosis

μ_4 is the fourth central moment

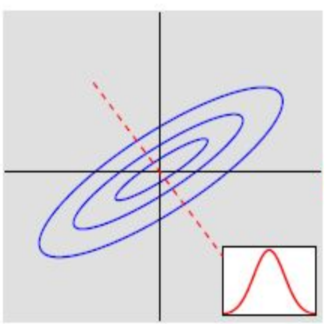
$$\mathrm{Kurt}[X] = \mathbb{E}\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] = \frac{\mathbb{E}\left[(X-\mu)^4\right]}{\left(\mathbb{E}[(X-\mu)^2]\right)^2} = \frac{\mu_4}{\sigma_4}$$

  - Measure the “tailed-ness” of a distribution
  - All Gaussian distributions have Kurt=3
  - By maximizing Kurtosis, we can increase the “non-gaussianity”.

Kurtosis - Wikipedia https://en.wikipedia.org/wiki/Kurtosis

PCA whitening (preprocessing for ICA): data from Gaussian

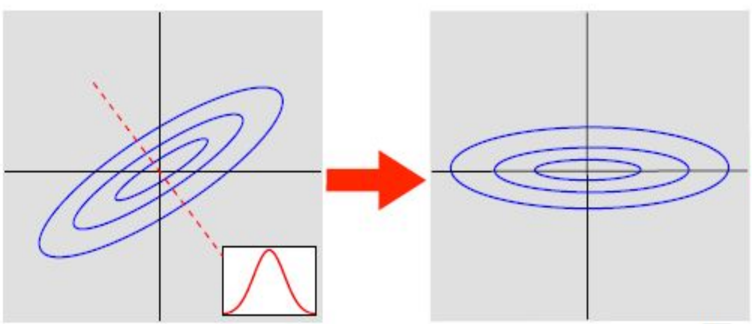
- Apply PCA: $\Sigma = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$



$\mathbf{x}$

PCA whitening (preprocessing for ICA): data from Gaussian

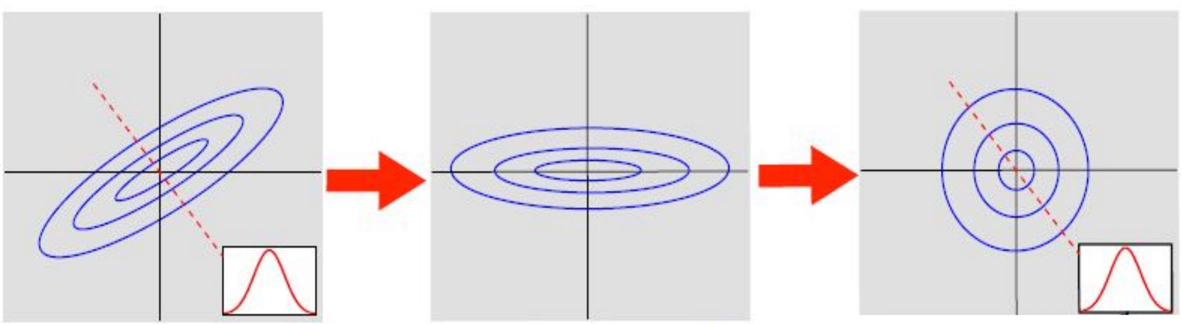
- Apply PCA: $\Sigma = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$
- Project (rotate) to the principal components



$\mathbf{x}$ $\qquad$ $\mathbf{x}_{\mathrm{PCA}} = \mathbf{U}^\top \mathbf{x}$

PCA whitening (preprocessing for ICA): : data from Gaussian

- Apply PCA: $\Sigma = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$
- Project (rotate) to the principal components
- “Scale” each axis so that the transformed data has identity as covariance.



$\mathbf{x}$ $\qquad$ $\mathbf{x}_{\text{PCA}} = \mathbf{U}^\top\mathbf{x}$ $\qquad$ $\mathbf{x}_{\text{white}} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top\mathbf{x}$

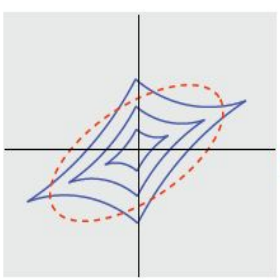
Note: this is a visualization for data with Gaussian distribution.

(Next slide:) For non-Gaussian data, ICA further rotates the whitened data to maximize non-gaussianity along each axis.

ICA illustration: data from non-Gaussian distribution

- PCA whitening
  - Apply PCA: $\Sigma = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$

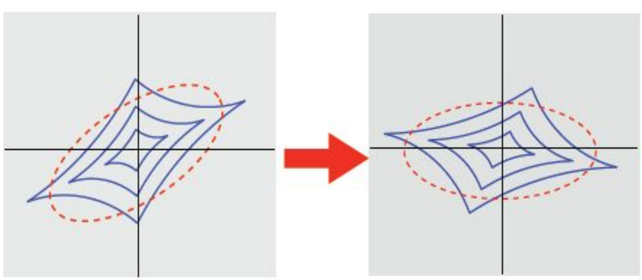
$\mathbf{x}$



ICA illustration: data from non-Gaussian distribution

- PCA whitening
  - Apply PCA: $\Sigma = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$
  - Project (rotate) to the principal components

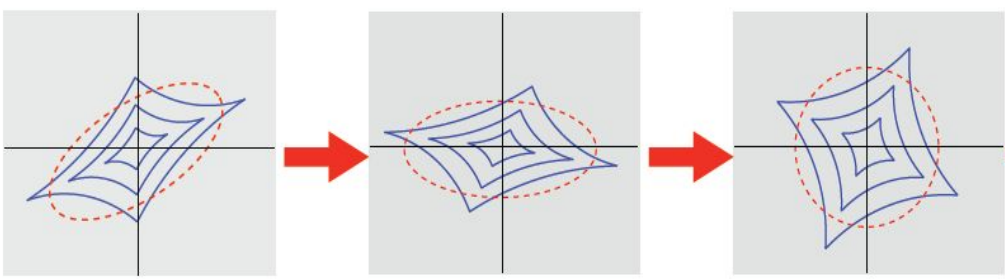
$\mathbf{x}$ $\qquad$ $\mathbf{x}_{\mathrm{PCA}} = \mathbf{U}^\top \mathbf{x}$



ICA illustration: data from non-Gaussian distribution

- PCA whitening
  - Apply PCA: $\Sigma = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$
  - Project (rotate) to the principal components
  - “Scale” each axis so that the transformed data has identity as covariance.

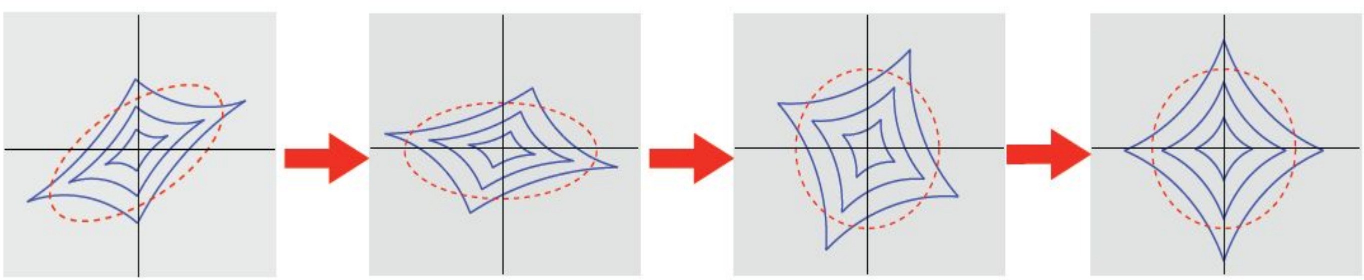
$\mathbf{x}$ $\qquad$ $\mathbf{x}_{\mathrm{PCA}} = \mathbf{U}^\top \mathbf{x}$ $\qquad$ $\mathbf{x}_{\mathrm{white}} = \boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top \mathbf{x}$



ICA illustration: data from non-Gaussian distribution

- PCA whitening
  - Apply PCA: $\Sigma = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$
  - Project (rotate) to the principal components
  - "Scale" each axis so that the transformed data has identity as covariance.
- Rotate to maximize non-Gaussianity

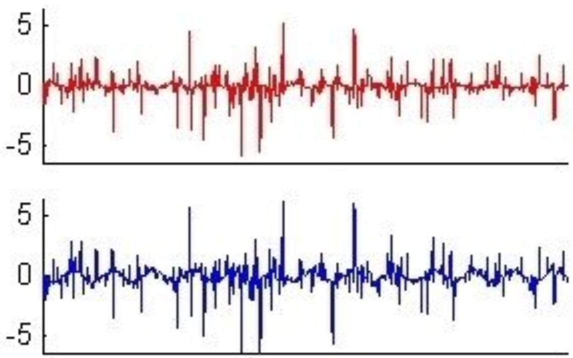
$\mathbf{x}$ $\quad$ $\mathbf{x}_{\text{PCA}} = \mathbf{U}^\top\mathbf{x}$ $\quad$ $\mathbf{x}_{\text{white}} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top\mathbf{x}$ $\quad$ $\mathbf{x}_{\text{ICA}} = \mathbf{W}\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top\mathbf{x}$



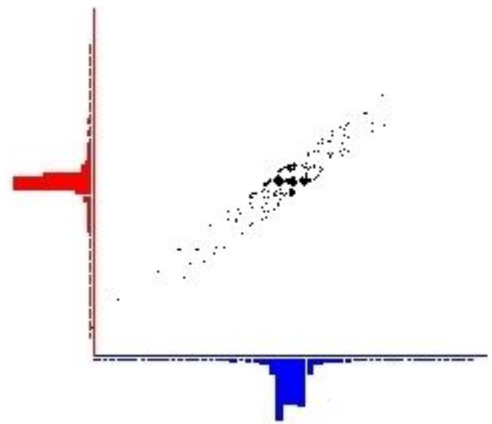
Independent Component Analysis

- Mixture example.

Signals



Joint Density

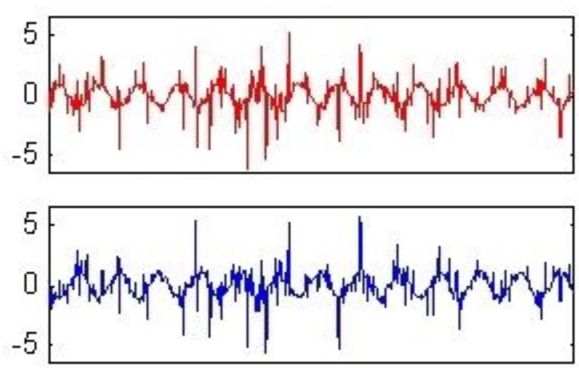


Input signals and density

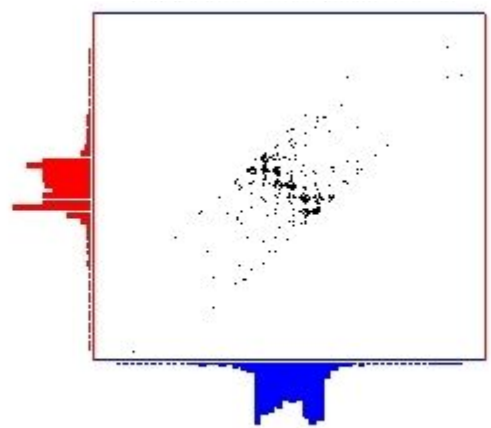
Independent Component Analysis

- Remove correlations by whitening (sphering) the data.

Signals



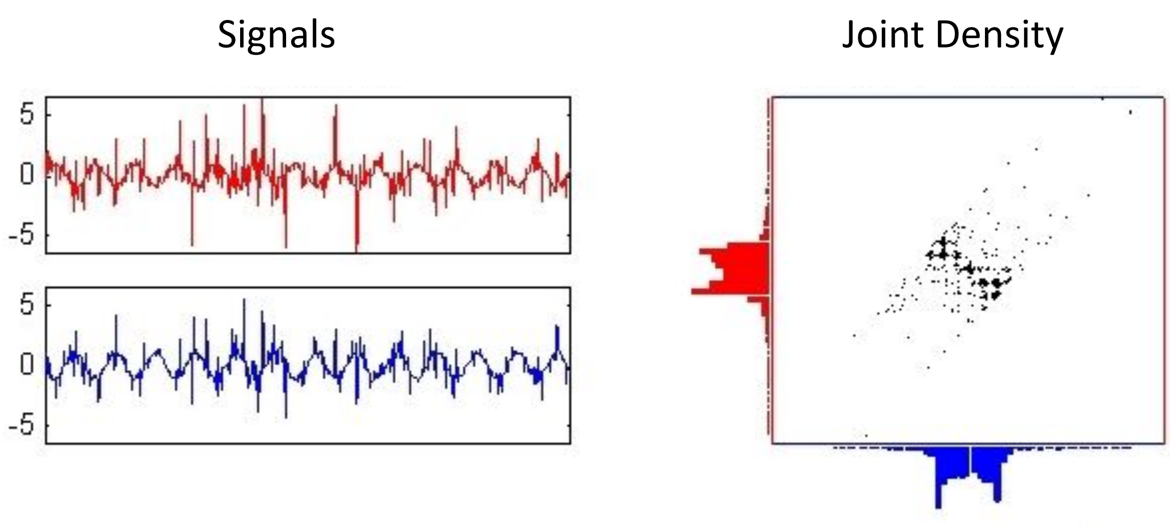
Joint Density



Whitened signals and density

Independent Component Analysis

- Step 1 of FastICA

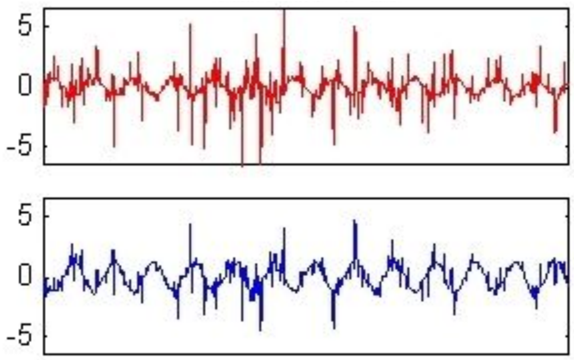


Separated signals after 1 step of FastICA

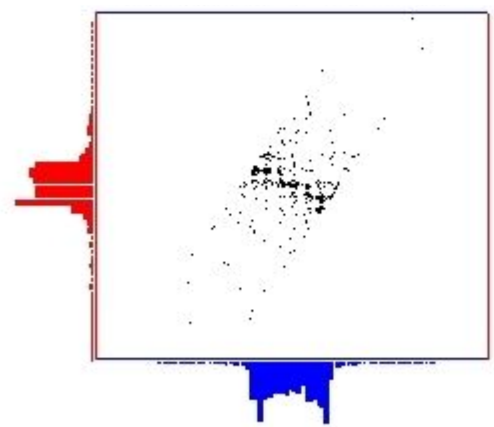
Independent Component Analysis

- Step 2 of FastICA

Signals



Joint Density

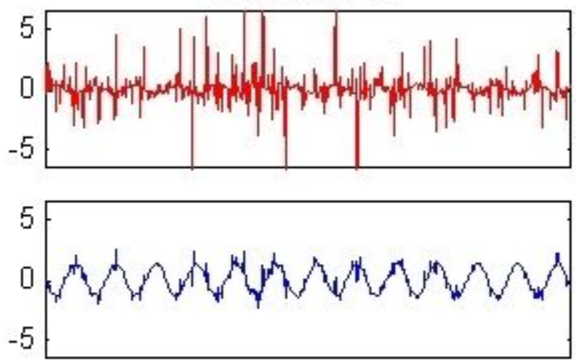


Separated signals after 2 steps of FastICA

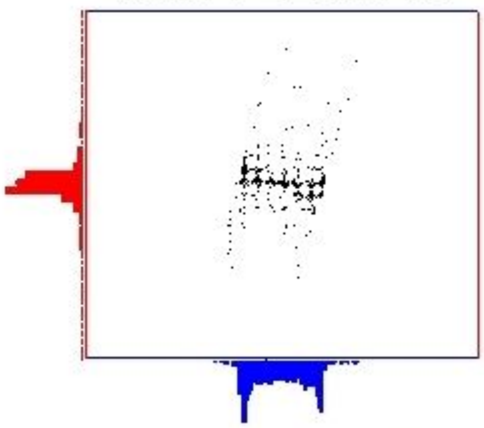
Independent Component Analysis

- Step 3 of FastICA

Signals



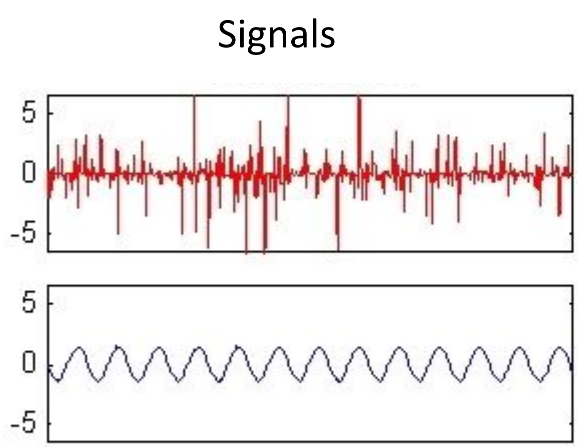
Joint Density

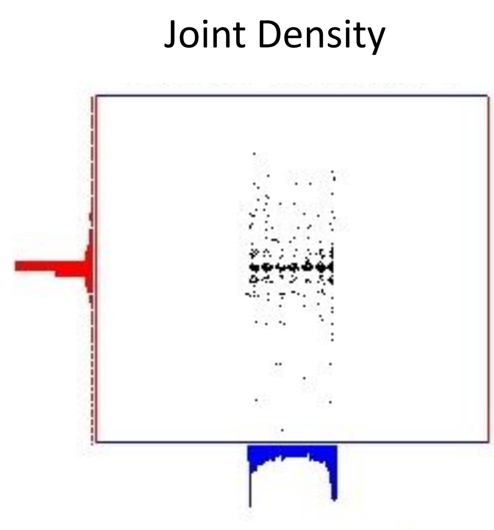


Separated signals after 3 steps of FastICA

Independent Component Analysis

- Step 4 of FastICA



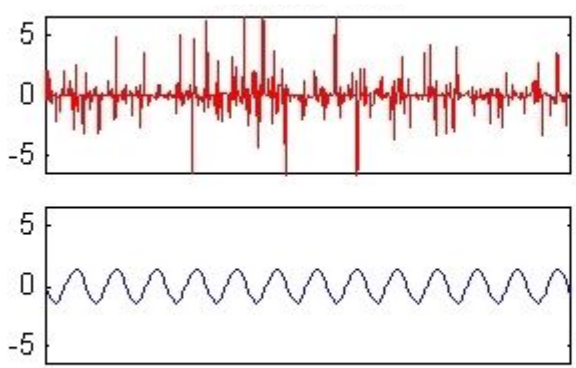


Separated signals after 4 steps of FastICA

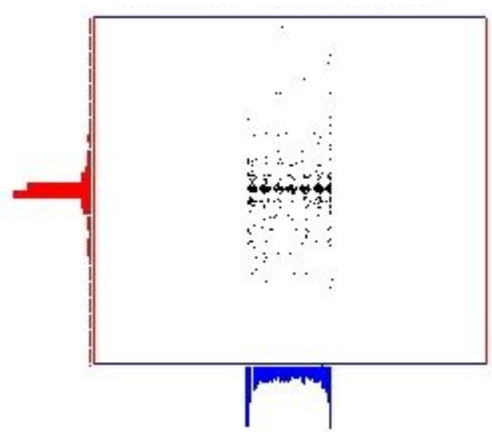
Independent Component Analysis

- Step 5: note that $p(\mathbf{x}_{\mathrm{ICA},1}, \mathbf{x}_{\mathrm{ICA},2}) = p(\mathbf{x}_{\mathrm{ICA},1})p(\mathbf{x}_{\mathrm{ICA},2})$

Signals



Joint Density



Separated signals after 5 steps of FastICA

ICA: summary

- Learning is done by PCA whitening followed by maximizing non-Gaussianity after transformations (kurtosis maximization).
- ICA is widely used for “blind-source separation.”
- The ICA components can be used for features.
- Limitations:
  - Difficult to learn overcomplete bases due to the orthogonality constraint
  - Difficult to handle situations where mixing is non-linear.

Blind Source Separation: Audio Examples
https://www.kecl.ntt.co.jp/icl/signal/sawada/demo/bss2to4/index.html
https://cnl.salk.edu/~tewon/Blind/blind_audio.html