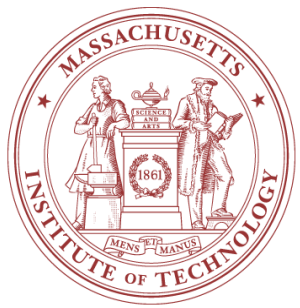
HST.582J / 6.555J / 16.456J Biomedical Signal and Image Processing
Spring 2007

Blind Source Separation: PCA & ICA

HST.582J/6.555J/16.456J

Gari D. Clifford

http://www.mit.edu/~gari



Harvard–MIT Health Sciences & Technology

What is BSS?

Assume an observation (signal) is a **linear** mix of >1 unknown **independent** *source* signals

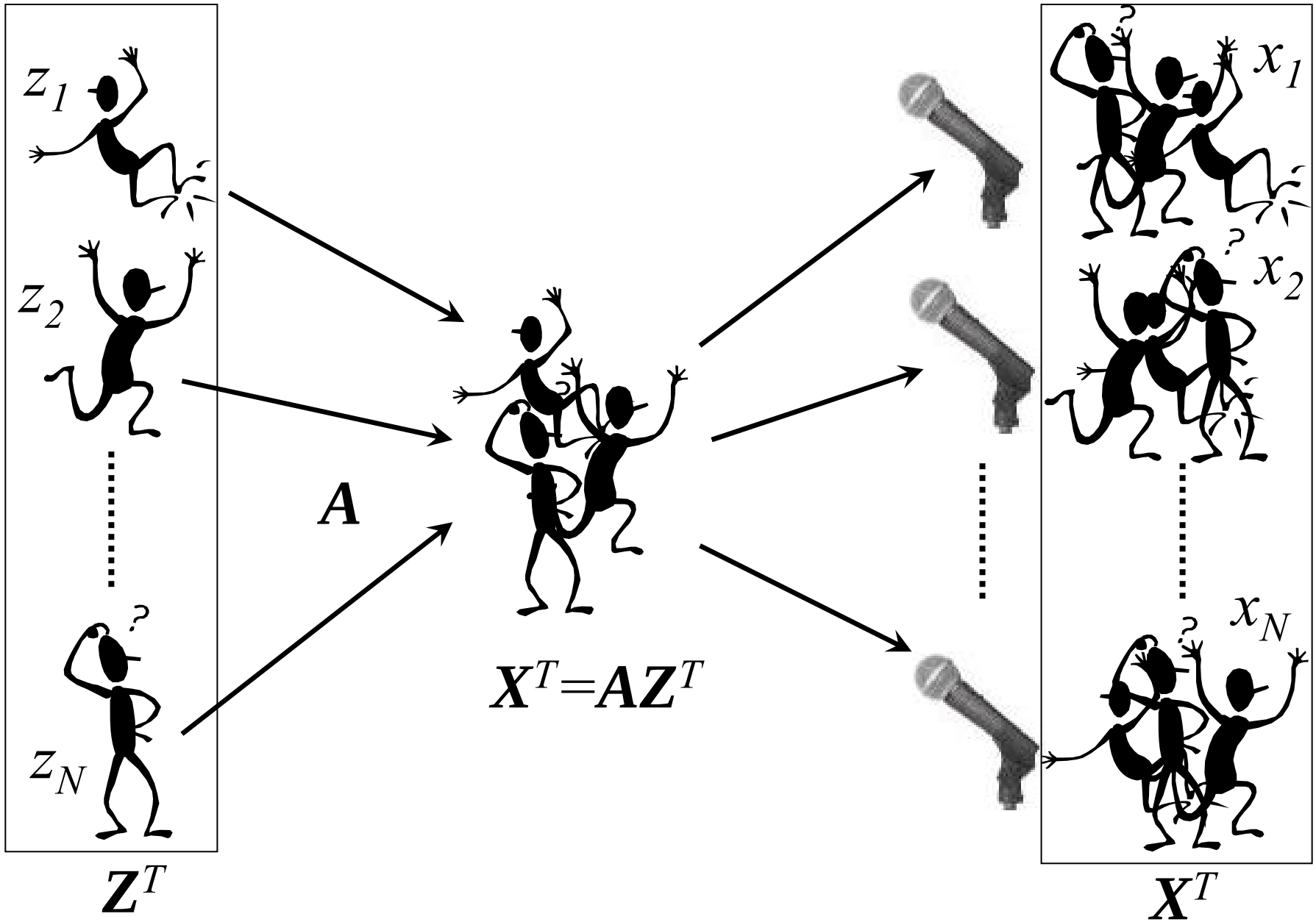
The *mixing* (not the signals) is **stationary**

We have as many **observations** as unknown **sources**

To find sources in observations
- need to define a suitable **measure of independence**

… For example - the *cocktail party problem* (sources are speakers and background noise):

The cocktail party problem - find $\boldsymbol{Z}$



Formal statement of problem

- *N independent* sources … $\boldsymbol{Z}_{mn}$ (MxN)
- linear square mixing … $\boldsymbol{A}_{nn}$ (NxN) (#sources=#sensors)
- produces a set of observations … $\boldsymbol{X}_{mn}$ (MxN)

$$\ldots\ldots \quad \boldsymbol{X}^T = \boldsymbol{A}\boldsymbol{Z}^T$$

Formal statement of solution

- 'demix' observations … $\boldsymbol{X}^T$ (NxM)

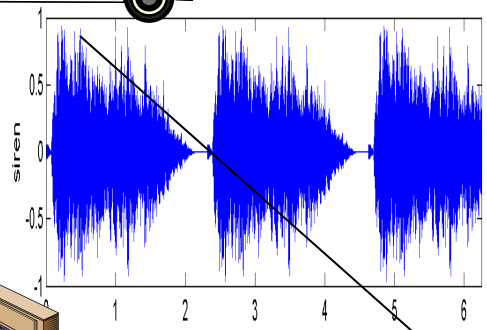
  into $\boldsymbol{Y}^T = \boldsymbol{W}\boldsymbol{X}^T$

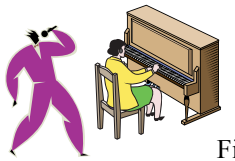
$\boldsymbol{Y}^T$ (NxM) $\approx \boldsymbol{Z}^T$ $\boldsymbol{W}$ (NxN) $\approx \boldsymbol{A}^{-1}$

How do we recover the independent sources?

(We are trying to estimate $\boldsymbol{W} \approx \boldsymbol{A}^{-1}$)

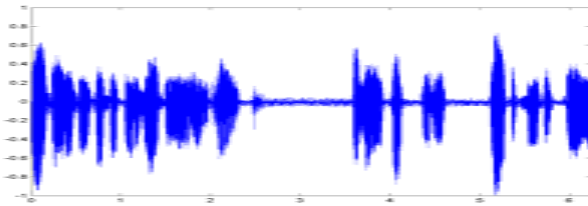
…. We require **a measure of independence**!

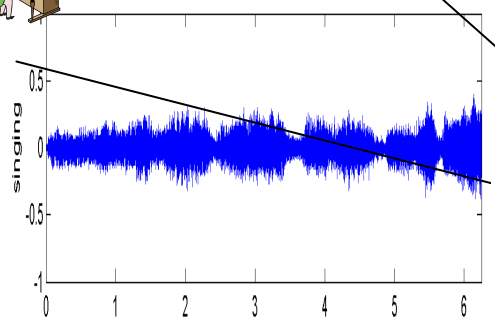


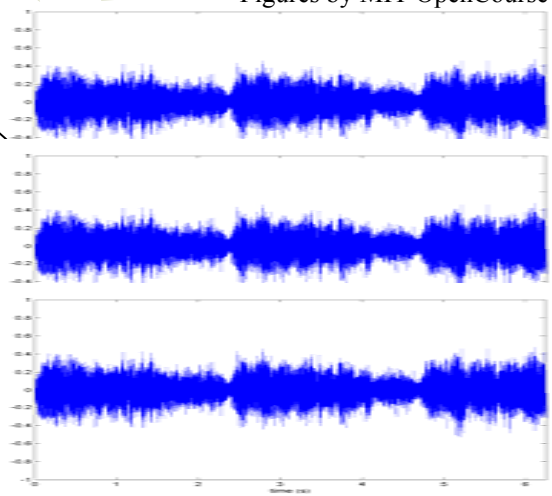


Figures by MIT OpenCourseWare.

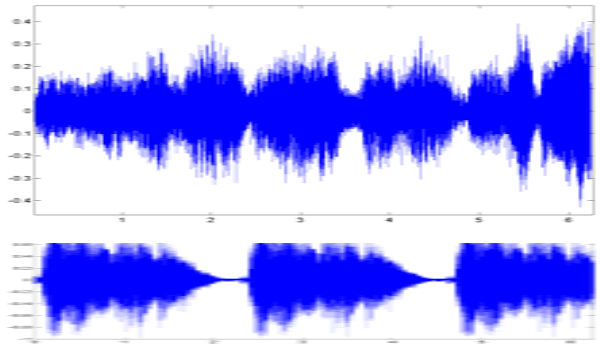
'Signal' source

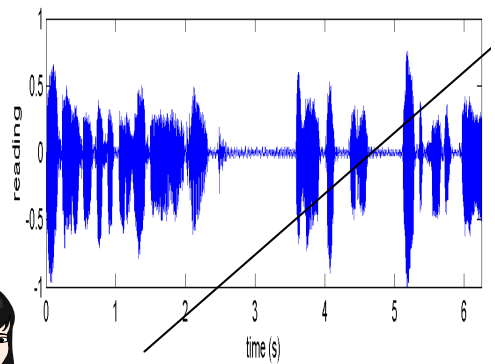






'Noise' sources





Observed mixtures

$$X^T = AZ^T$$

$$Z^T$$

$$Y^T = WX^T$$

$$\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ x_{M1} & x_{M2} & \cdots & x_{MN} \end{bmatrix}^T = \begin{bmatrix} a_{11} & \cdots & a_{N1} \\ \vdots & \vdots & \vdots \\ a_{1N} & \cdots & a_{NN} \end{bmatrix} \cdot \begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1N} \\ z_{21} & z_{22} & \cdots & z_{2N} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ z_{M1} & z_{M2} & \cdots & z_{MN} \end{bmatrix}^T$$

$$\boldsymbol{X}^T \quad = \quad \boldsymbol{A} \quad \boldsymbol{Z}^T$$

$$\mathbf{Y} = \hat{\mathbf{Z}}$$

$$\mathbf{W} = \hat{\mathbf{A}}^{-1}$$

$$\begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1N} \\ y_{21} & y_{22} & \cdots & y_{2N} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ y_{M1} & y_{M2} & \cdots & y_{MN} \end{bmatrix}^T = \begin{bmatrix} w_{11} & \cdots & w_{N1} \\ \vdots & \vdots & \vdots \\ w_{1N} & \cdots & w_{NN} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ x_{M1} & x_{M2} & \cdots & x_{MN} \end{bmatrix}^T$$

$$\boldsymbol{Y}^T \quad = \quad \boldsymbol{W} \quad \boldsymbol{X}^T$$

The Fourier Transform

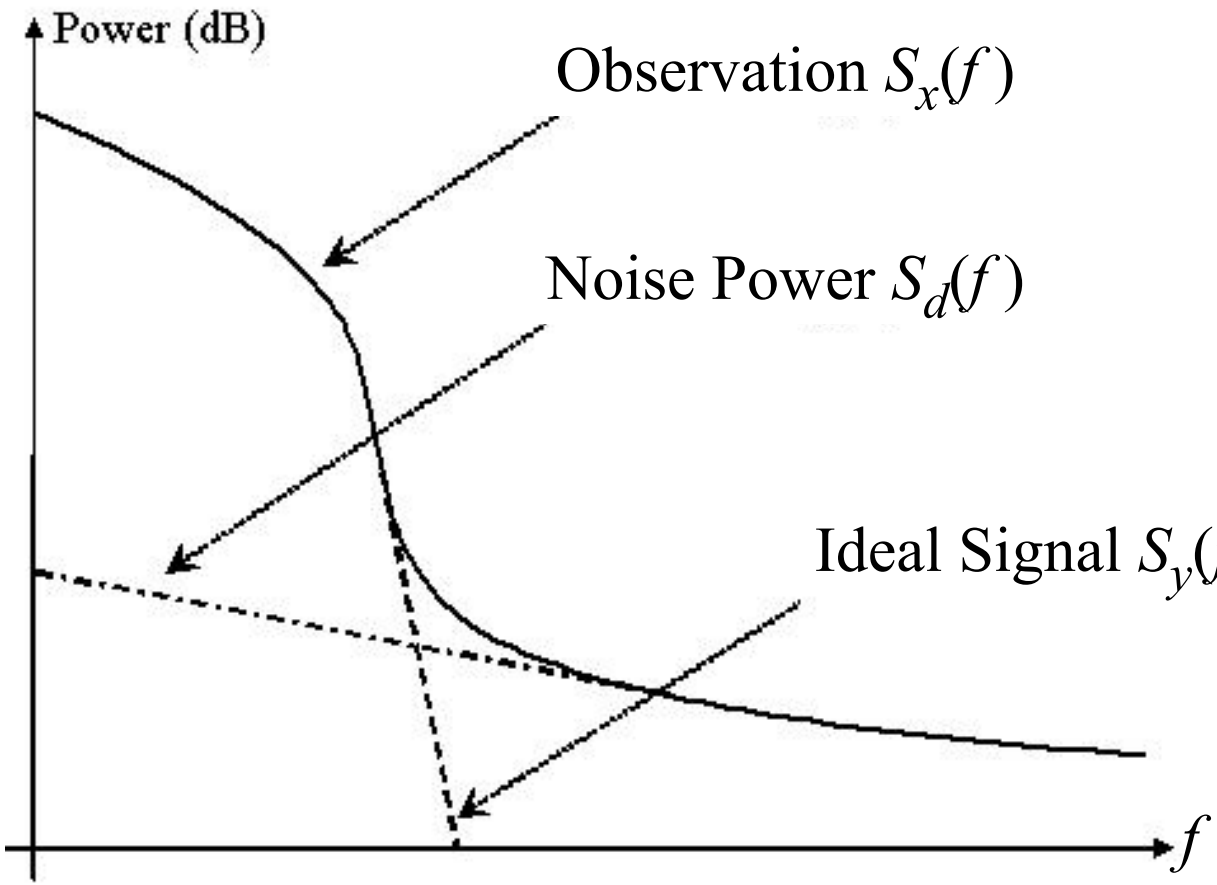
$$\mathbf{Y}_k = \sum_{n=1}^{N} \mathbf{W}_{kn}\mathbf{X}_n \qquad \mathbf{W}_{kn} = e^{-j2\pi kn/N}$$

$$\mathbf{W} = \begin{bmatrix} e^{-j2\pi} & e^{-j4\pi} & \dots & e^{-j2\pi N} \\ e^{-j4\pi} & e^{-j8\pi} & \dots & e^{-j4\pi N} \\ \vdots & \vdots & & \vdots \\ e^{-j2\pi M} & e^{-j4\pi M} & \dots & e^{-j2\pi MN} \end{bmatrix}$$

(Independence between components is assumed)

Recap: Non-causal Wiener filtering

$$H(f) = \frac{S_y(f)}{S_y(f) + S_d(f)}$$



$x[n]$ - observation

$y[n]$ - ideal signal

$d[n]$ - noise component

Filtered signal:

$S_{filt}(f) = S_x(f).H(f)$

BSS is a transform?

- Like Fourier, we decompose into components by transforming the observations into another vector space which maximises the separation between interesting (*signal*) and unwanted (*noise*).
- Unlike Fourier, separation is not based on frequency- It's based on ***independence***
- Sources can have the same frequency content
- No assumptions about the signals (other than they are *independent* and *linearly* mixed)
- So you can filter/separate in-band noise/signals with BSS

Principal Component Analysis

- Second order *decorrelation* = ***independence***
- Find a set of orthogonal axes in the data (independence metric = variance)
- Project data onto these axes to *decorrelate*
- ***Independence*** is forced onto the data through the orthogonality of axes
- Conventional noise / signal separation technique

Singular Value Decomposition

Decompose observation $\boldsymbol{X=AZ}$ into….

$$\boldsymbol{X=USV^T}$$

- $\boldsymbol{S}$ is a diagonal matrix of singular values with elements arranged in descending order of magnitude (the singular spectrum)
- The columns of $\boldsymbol{V}$ are the eigenvectors of $\boldsymbol{C=X^TX}$ (the orthogonal subspace … $dot(v_i,v_j)=0$) … they '*demix*' or rotate the data
- $\boldsymbol{U}$ is the matrix of projections of $\boldsymbol{X}$ onto the eigenvectors of $\boldsymbol{C}$ **…** the *'source'* estimates

Singular Value Decomposition

Decompose observation $\boldsymbol{X=AZ}$ into….

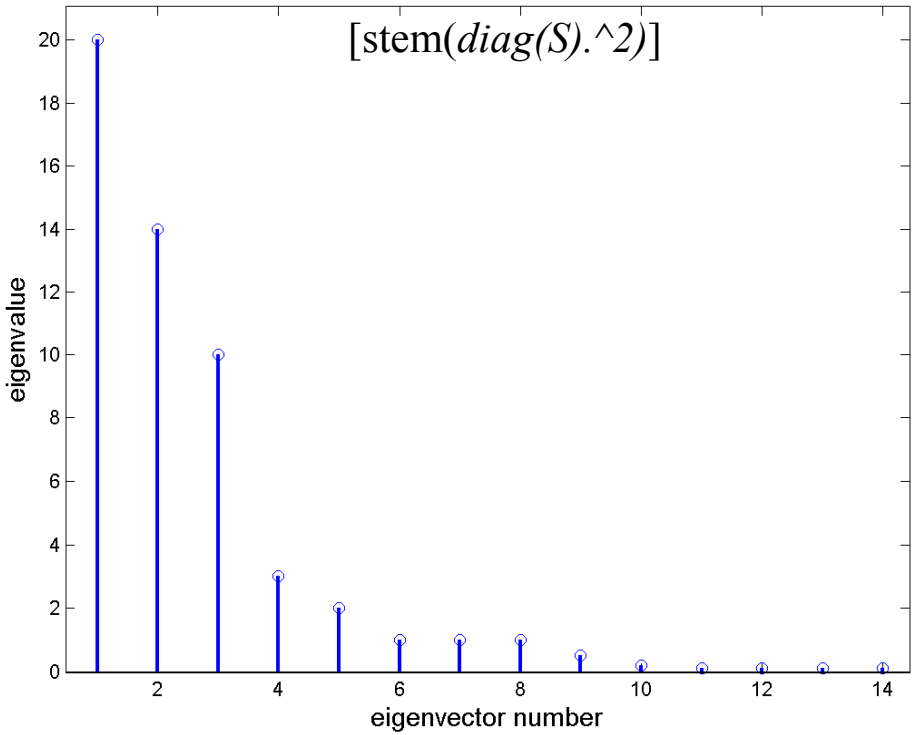
$$\boldsymbol{X=USV^T}$$

$$\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ x_{M1} & x_{M2} & \cdots & x_{MN} \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1N} \\ u_{21} & u_{22} & \cdots & u_{2N} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ u_{M1} & u_{M2} & \cdots & u_{MN} \end{bmatrix} \cdot \begin{bmatrix} s_{11} & 0 & \cdots & 0 \\ 0 & s_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & s_{NN} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \cdot \begin{bmatrix} v_{11} & \cdots & v_{1N} \\ v_{21} & \cdots & v_{2N} \\ \vdots & \vdots & \vdots \\ v_{N1} & \cdots & v_{NN} \end{bmatrix}$$

$$\mathbf{X} = \mathrm{U} \cdot \mathbf{S} \cdot \mathbf{V}^T$$

Eigenspectrum of decomposition

- $\boldsymbol{S}$ = *singular* matrix … zeros except on the leading diagonal
- $\boldsymbol{S}_{ij}$ (i=j) are the *eigenvalues*$^{1/2}$
- Placed in order of descending magnitude
- Correspond to the magnitude of projected data along each *eigenvector*
- Eigenvectors are the axes of maximal variation in the data



Eigenspectrum=
Plot of eigenvalues

Variance = power

(analogous to Fourier components in power spectra)

SVD: Method for PCA

A routine for performing SVD is as follows:

1. Find the N non-zero eigenvalues, λ_i of the matrix $\mathbf{C} = \mathbf{X^T X}$ and form a non-square diagonal matrix $\mathbf{S}$ by placing the square roots $s_i = \sqrt{\lambda_i}$ of the N eigenvalues in descending order of magnitude on the leading diagonal and setting all other elements of $\mathbf{S}$ to zero.

2. Find the orthogonal eigenvectors of the matrix $\mathbf{X^T X}$ corresponding to the obtained eigenvalues, and arrange them in the same order. this ordered collection of column-vectors forms the matrix $\mathbf{V}$.

3. Find the first N column-vectors of the matrix $\mathbf{U}$: $\mathbf{u}_i = s_i^{-1}\mathbf{X}\mathbf{v}_i$ $(i = 1 : N)$. Note that s_i^{-1} are the elements of $\mathbf{S}^{-1}$.

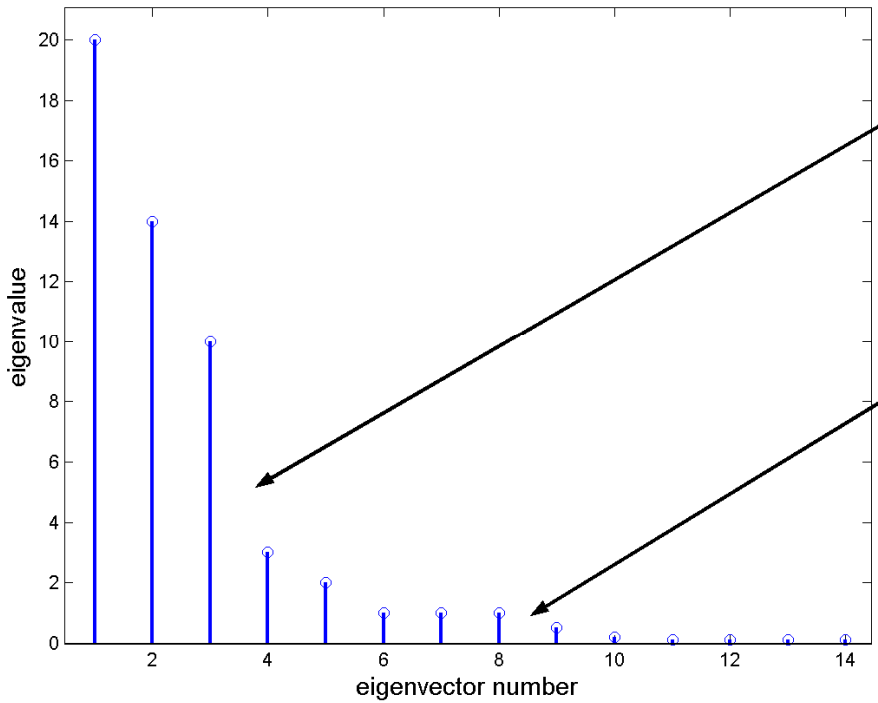
4. Add the rest of $M - N$ vectors to the matrix $\mathbf{U}$ using the Gram-Schmidt orthogonalization process (see appendix 15.9.2).

SVD noise/signal separation

To perform SVD filtering of a signal, use a truncated SVD decomposition (using the first p eigenvectors)

$$\boldsymbol{Y=US}_p\boldsymbol{V}^T$$

[Reduce the dimensionality of the data by discarding noise projections $\boldsymbol{S}_{\text{noise}}=0$
Then reconstruct the data with just the signal subsapce]

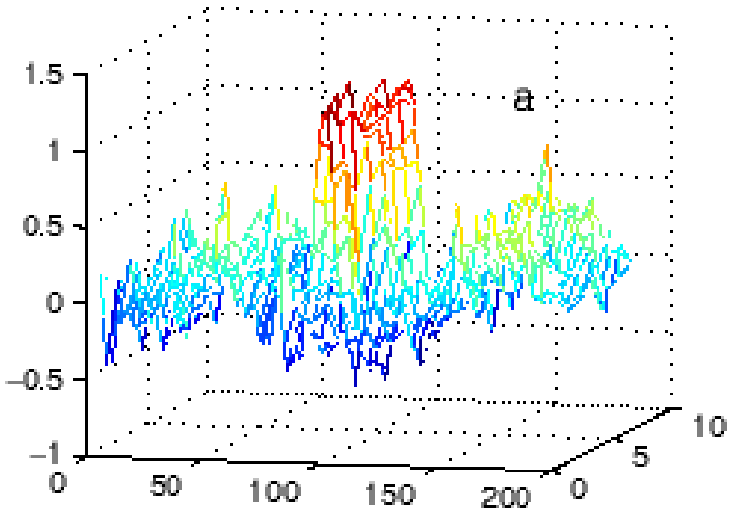


Most of the signal is contained in the first few principal components.

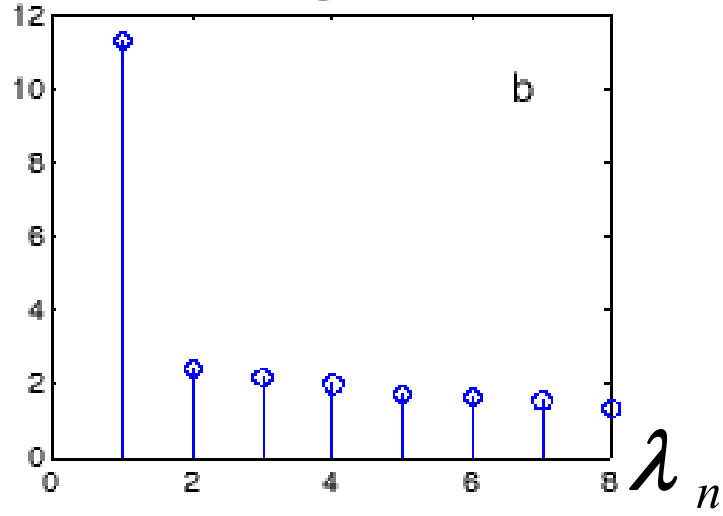
Discarding these and projecting back into the original observation space effects a noise-filtering or a noise/signal separation

Real data

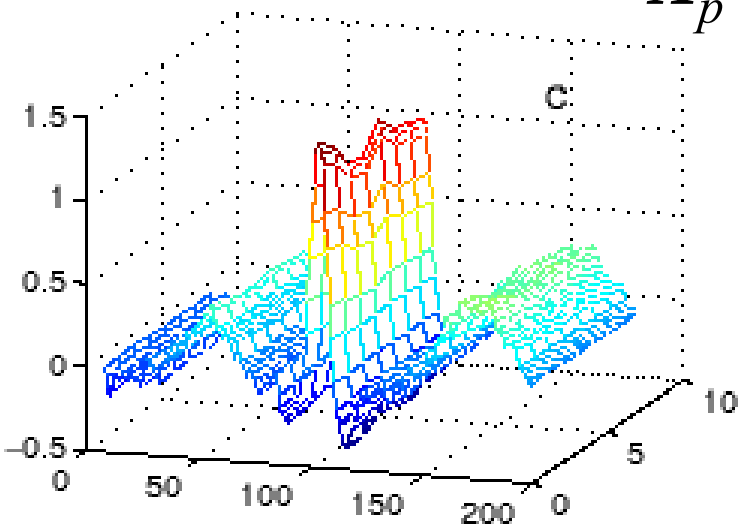
X



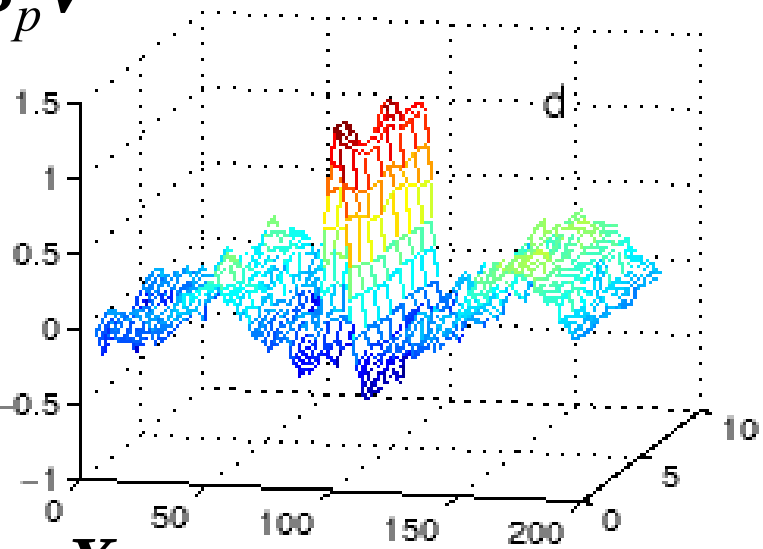
S^2



$$X_p = US_pV^T$$

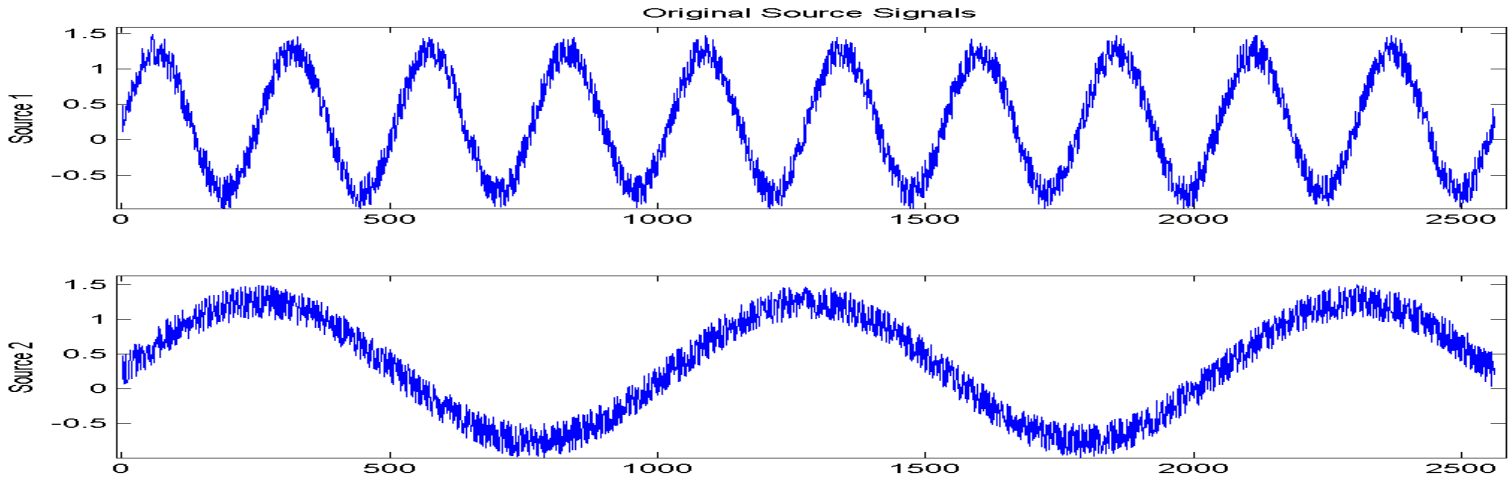


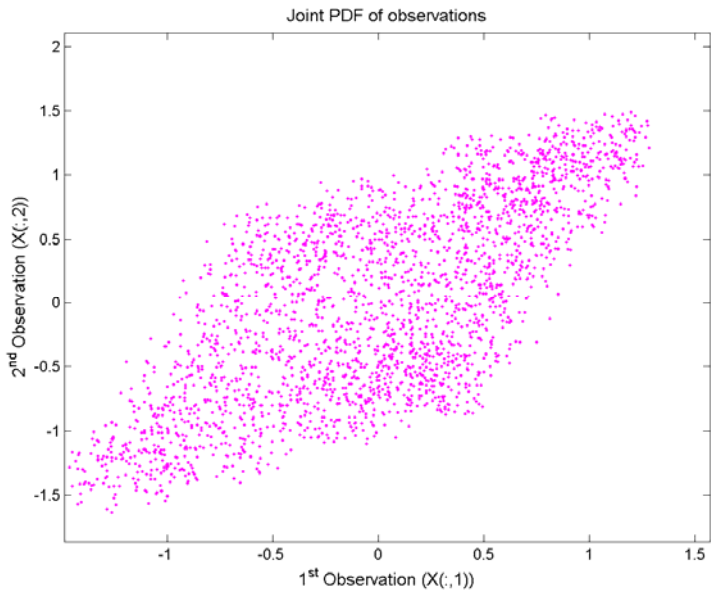
X_p ... $p=2$

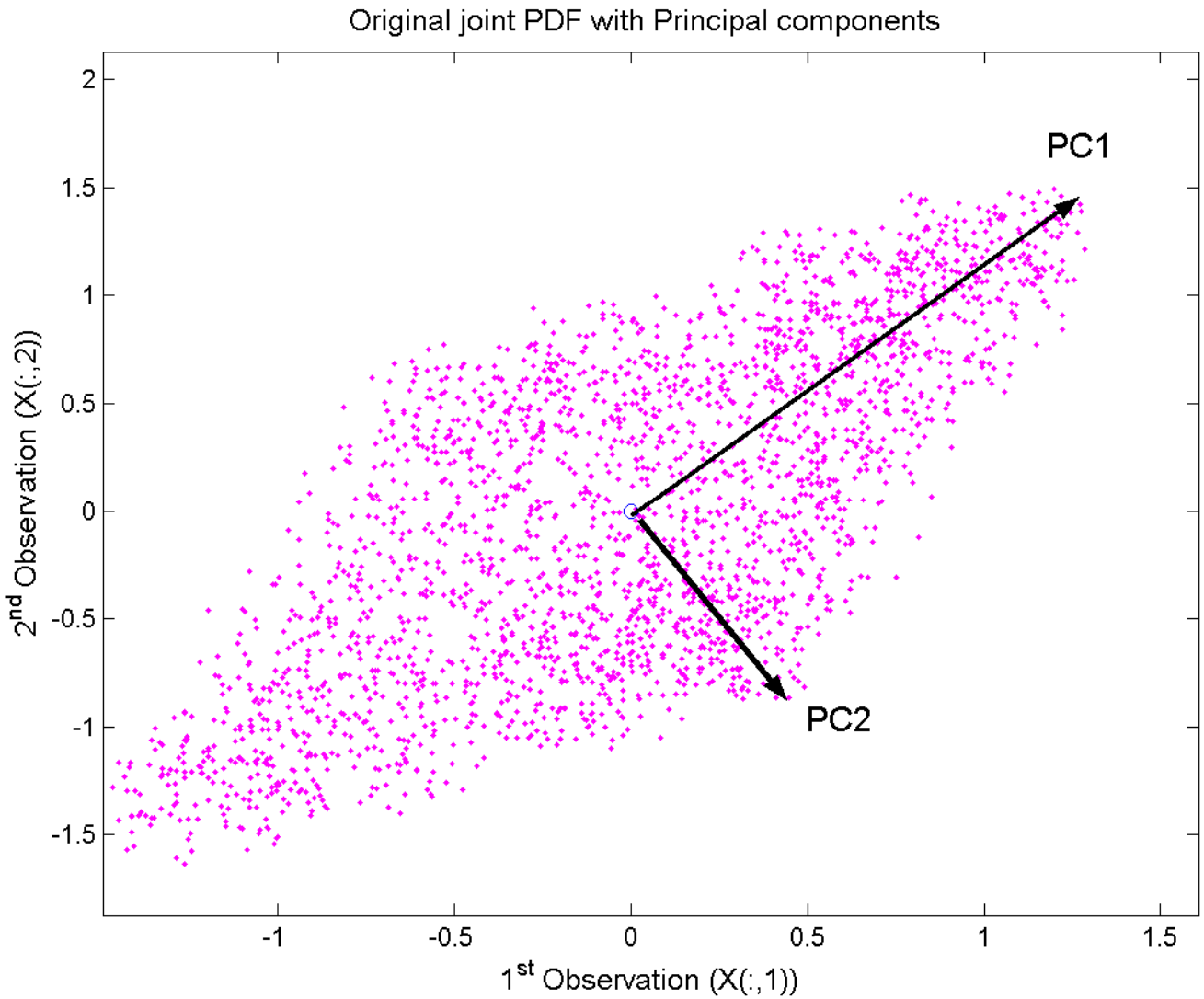


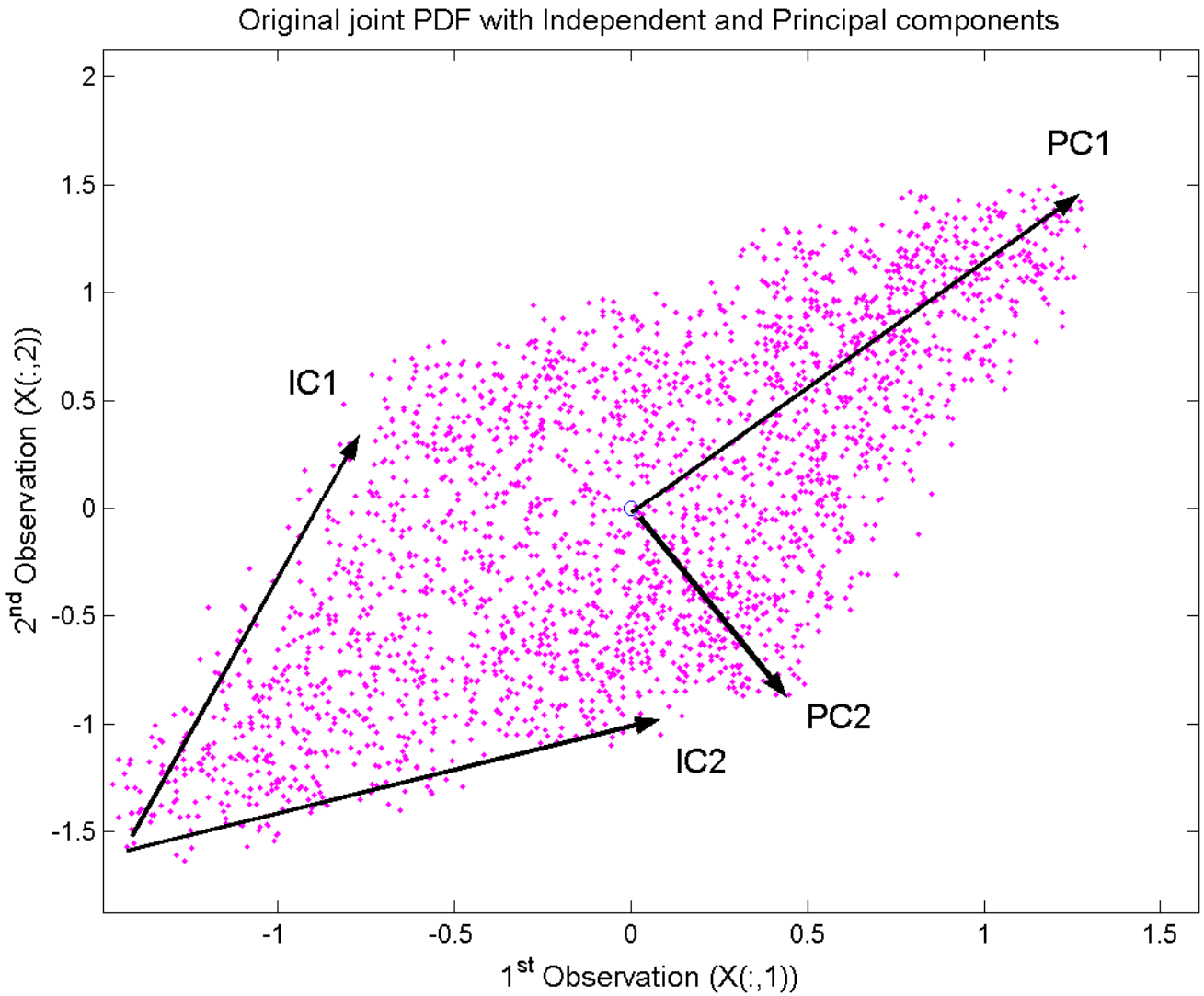
X_p ... $p=4$

Two dimensional example









Independent Component Analysis

As in PCA, we are looking for *N* different vectors onto which we can project our observations to give a set of *N* maximally independent signals (*sources*)

output data (discovered sources) dimensionality = dimensionality of observations

Instead of using *variance* as our independence measure (i.e. decorrelating) as we do in PCA, we use a measure of how statistically independent the sources are.

ICA: The basic idea ...

Assume underlying source signals ($\boldsymbol{Z}$) are independent.

Assume a linear mixing matrix ($\boldsymbol{A}$)… $\boldsymbol{X}^T=\boldsymbol{A}\boldsymbol{Z}^T$

in order to find $\boldsymbol{Y}$ ($\approx\boldsymbol{Z}$), find $\boldsymbol{W}$, ($\approx\boldsymbol{A}^{-1}$) ...

$$\boldsymbol{Y}^T=\boldsymbol{W}\boldsymbol{X}^T$$

How? Initialise $\boldsymbol{W}$ & iteratively update $\boldsymbol{W}$ to minimise or maximise a cost function that measures the (statistical) *independence* between the columns of the $\boldsymbol{Y}^T$.

Non-Gaussianity ⇒ statistical independence?

From the ***Central Limit Theorem***,

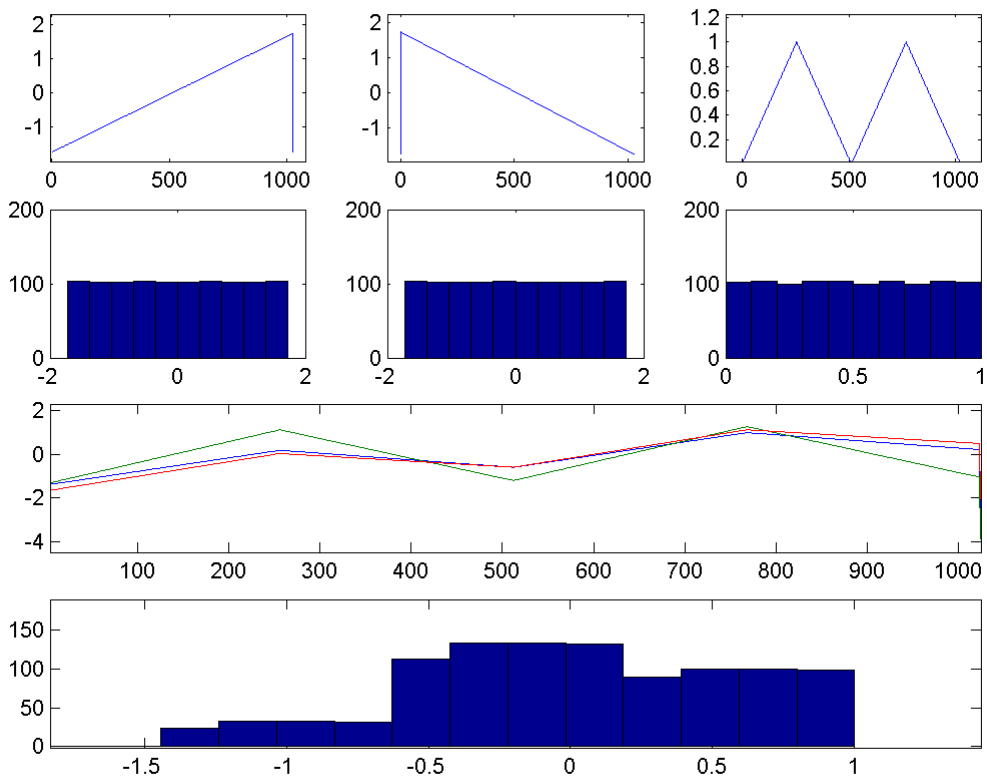
- add enough independent signals together, → Gaussian PDF

Sources, $\boldsymbol{Z}$ →

P($\boldsymbol{Z}$) (subGaussian) →

Mixtures ($\boldsymbol{X}^T=\boldsymbol{A}\boldsymbol{Z}^T$) →

P($\boldsymbol{X}$) (Gaussian) →



Recap: Moments of a distribution

$$\mu_x = E\{x\} = \int_{-\infty}^{+\infty} x p_x(x) \mathrm{d}x$$

$$\hat{\mu}_x = \frac{1}{M}\sum_{i=1}^{M} x_i$$

$$\sigma_x^2 = E\{(x-\mu_x)^2\} = \int_{-\infty}^{+\infty} (x-\mu_x)^2 p_x(x) \mathrm{d}x$$

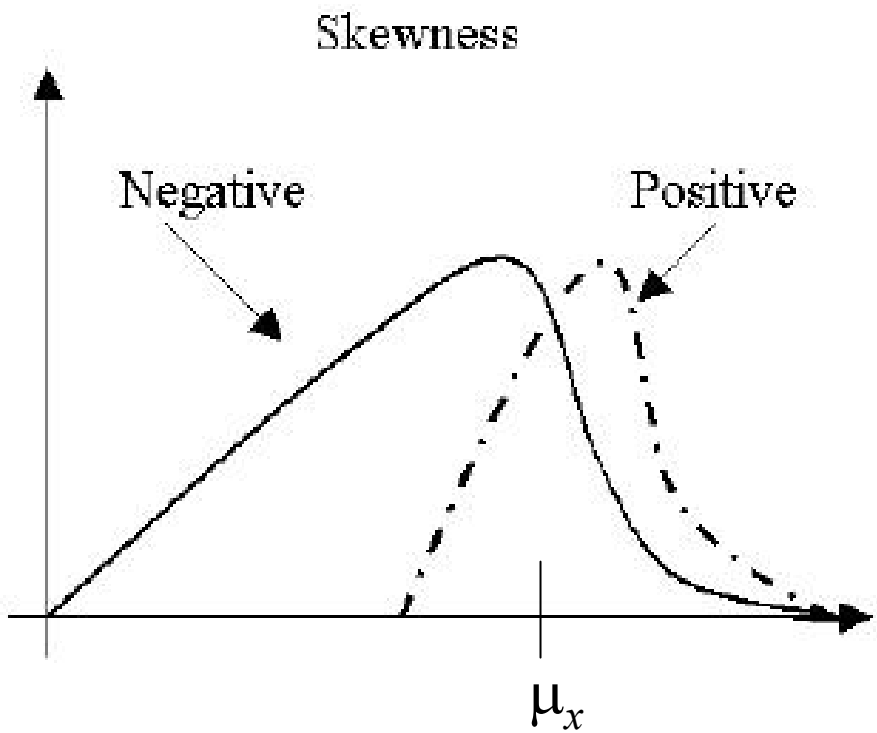
$$\hat{\sigma}^2(\mathbf{x}) = \frac{1}{M}\sum_{i=1}^{M} (x_i - \hat{\mu}_x)^2$$

$$\sigma(\mathbf{x}) = \sqrt{\sigma^2}$$

$$\upsilon_n = E\{(x-\mu_x)^n\} = \int_{-\infty}^{+\infty} (x-\mu_x)^n p_x(x) \mathrm{d}x$$

Higher order moments (3rd -*skewness*)

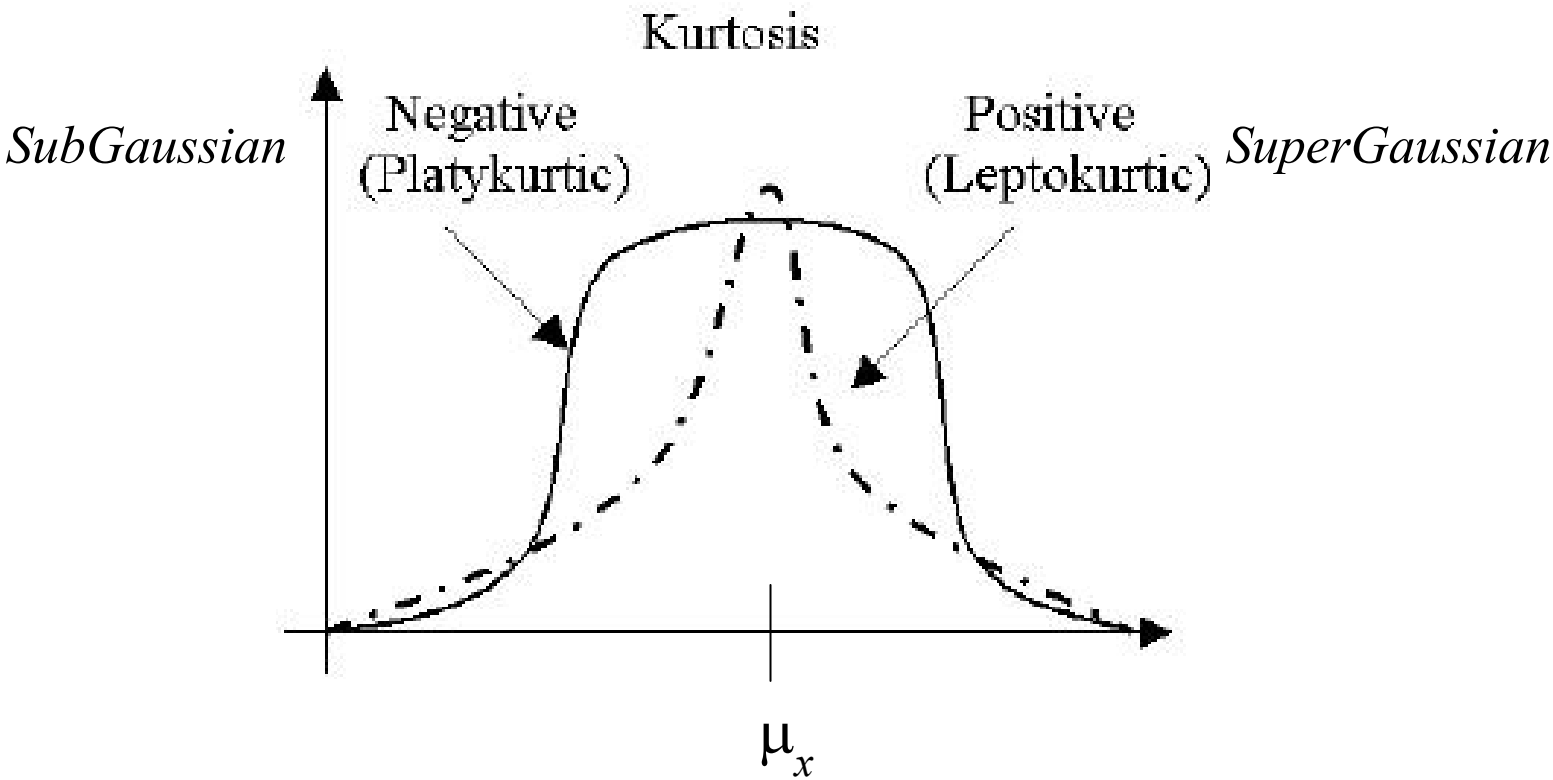
$$\hat{\zeta}(\mathbf{x}) = \frac{1}{M}\sum_{i=1}^{M}\left[\frac{x_i - \hat{\mu}_x}{\hat{\sigma}}\right]^3$$



Higher order moments (4th-*kurtosis*)

$$\hat{\kappa}(\mathbf{x}) = \frac{1}{M}\sum_{i=1}^{M}\left[\frac{x_i - \hat{\mu}_x}{\hat{\sigma}}\right]^4$$

Gaussians are *mesokurtic* with $\kappa = 3$



Non-Gaussianity ⇒ statistical independence?

Central Limit Theorem: add enough independent signals together,
⟶ Gaussian PDF ($\kappa = 3$)

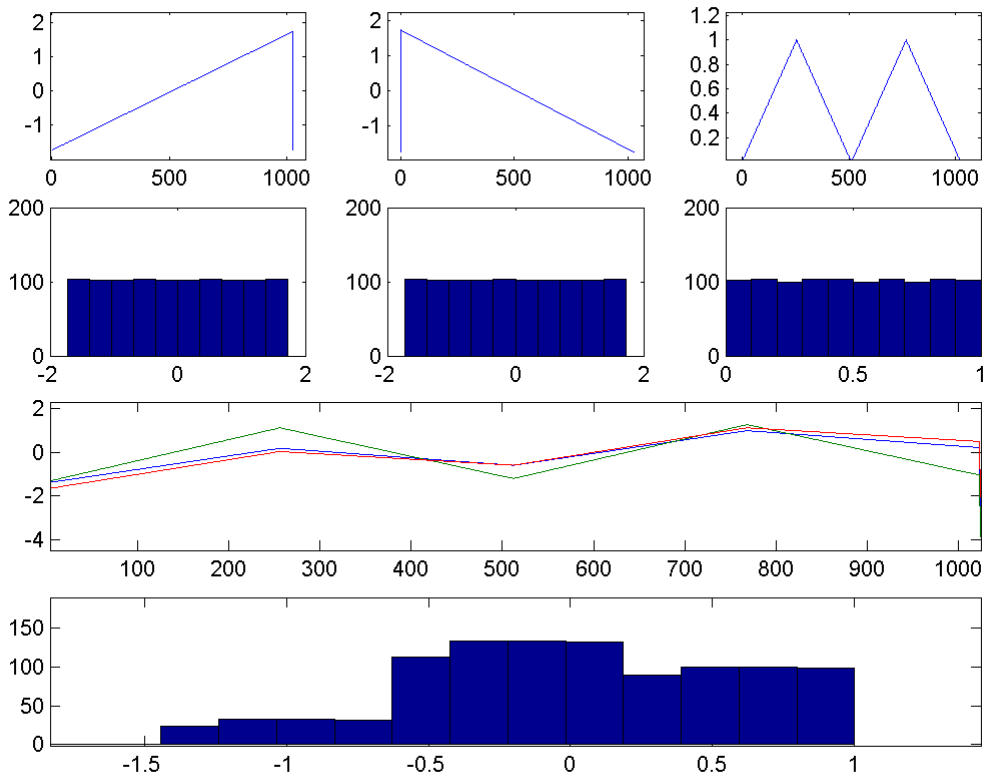
∴ make data components non-Gaussian to find independent sources

Sources, $\boldsymbol{Z}$ ⟶

$P(\boldsymbol{Z})$ ($\kappa < 3$ (1.8)) ⟶

Mixtures ($\boldsymbol{X}^T = \boldsymbol{A}\boldsymbol{Z}^T$) ⟶

$P(\boldsymbol{X})$ ($\kappa = 3.4$) ⟶



Recall – trying to estimate $\boldsymbol{W}$

Assume underlying source signals ($\boldsymbol{Z}$) are independent.

Assume a linear mixing matrix ($\boldsymbol{A}$)… $\boldsymbol{X}^T=\boldsymbol{A}\boldsymbol{Z}^T$

in order to find $\boldsymbol{Y}$ ($\approx\boldsymbol{Z}$), find $\boldsymbol{W}$, ($\approx\boldsymbol{A}^{-1}$) ...

$$\boldsymbol{Y}^T=\boldsymbol{W}\boldsymbol{X}^T$$

Initialise $\boldsymbol{W}$ & iteratively update $\boldsymbol{W}$ with gradient descent to maximise kurtosis.

Gradient descent to find $\boldsymbol{W}$

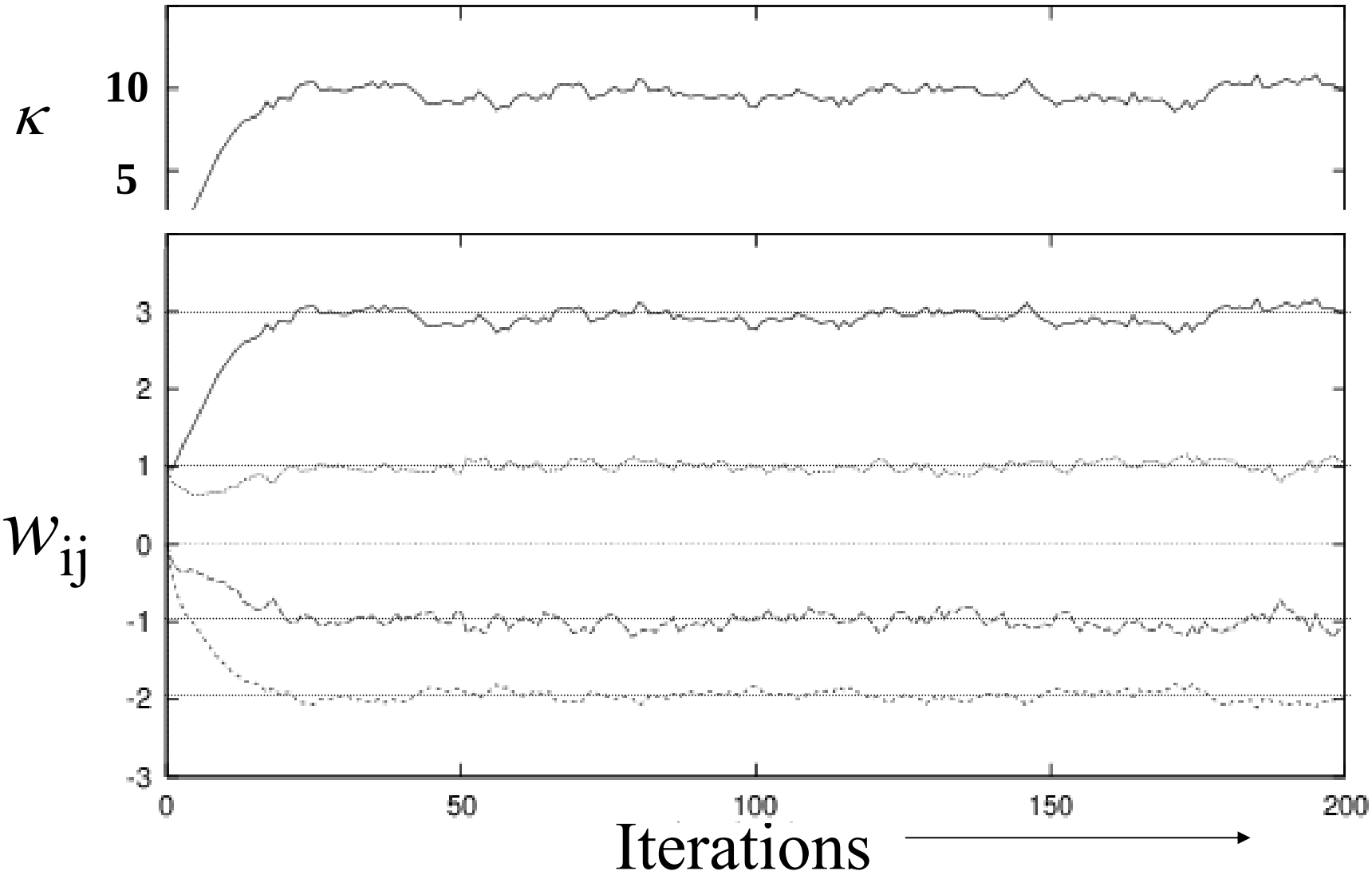
- Given a cost function, ξ, we update each element of $\boldsymbol{W}$ (w_{ij}) at each step, τ,

$$w_{ij}^{(\tau+1)} = w_{ij}^{(\tau)} - \eta \frac{\partial \xi}{\partial w_{ij}}$$

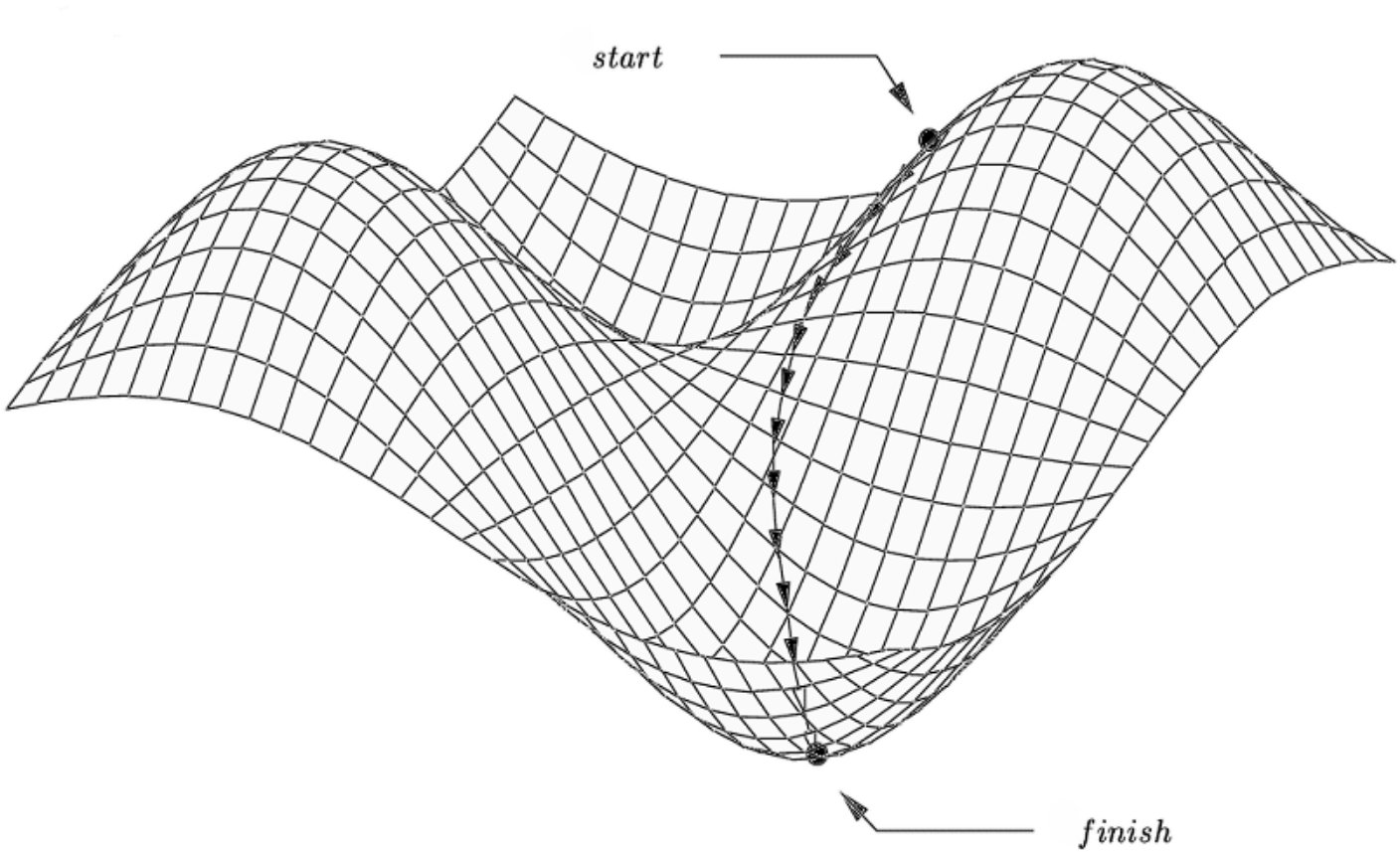
- … and recalculate cost function
- (η is the learning rate (~ 0.1), and speeds up convergence.)

Weight updates to find: (Gradient ascent)

$$\mathbf{w} = \begin{bmatrix} 1 & 3 \\ -2 & -1 \end{bmatrix}$$



Gradient descent



Gradient Descent

- Cost function, ξ, can be maximum κ or minimum $1/\kappa$

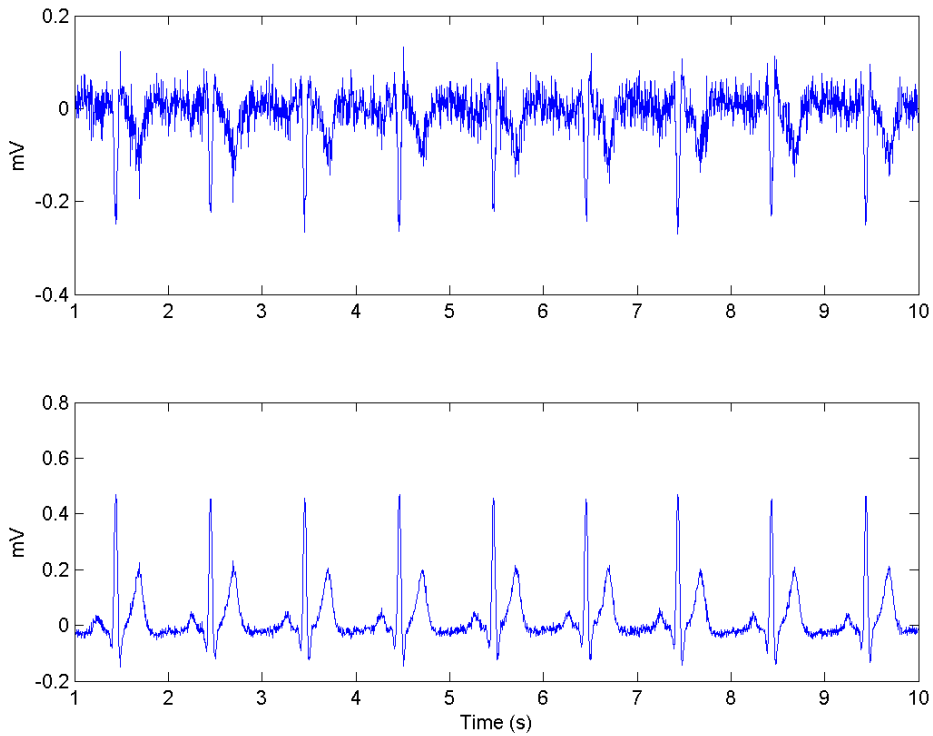
$$w_{ij}^{(\tau+1)} = w_{ij}^{(\tau)} - \eta \frac{\partial \xi}{\partial w_{ij}}$$

$$\xi = \min\left(1/|\kappa_1|, 1/|\kappa_2|\right) \Big|_{\kappa = \max}$$

Gradient descent example

- Imagine a 2-channel ECG, comprised of two sources;
  - Cardiac
  - Noise

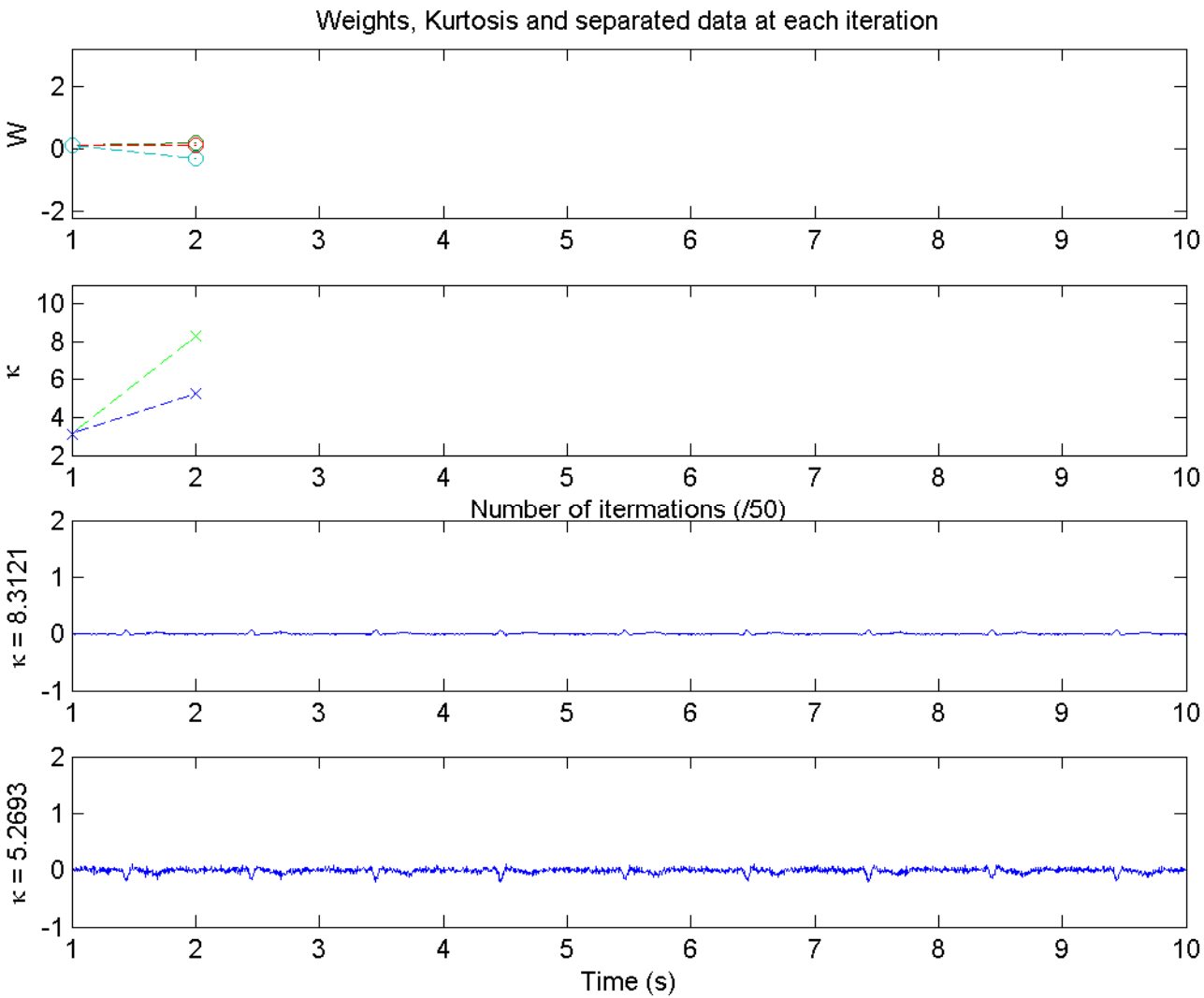
… and SNR=1



X_1

X_2

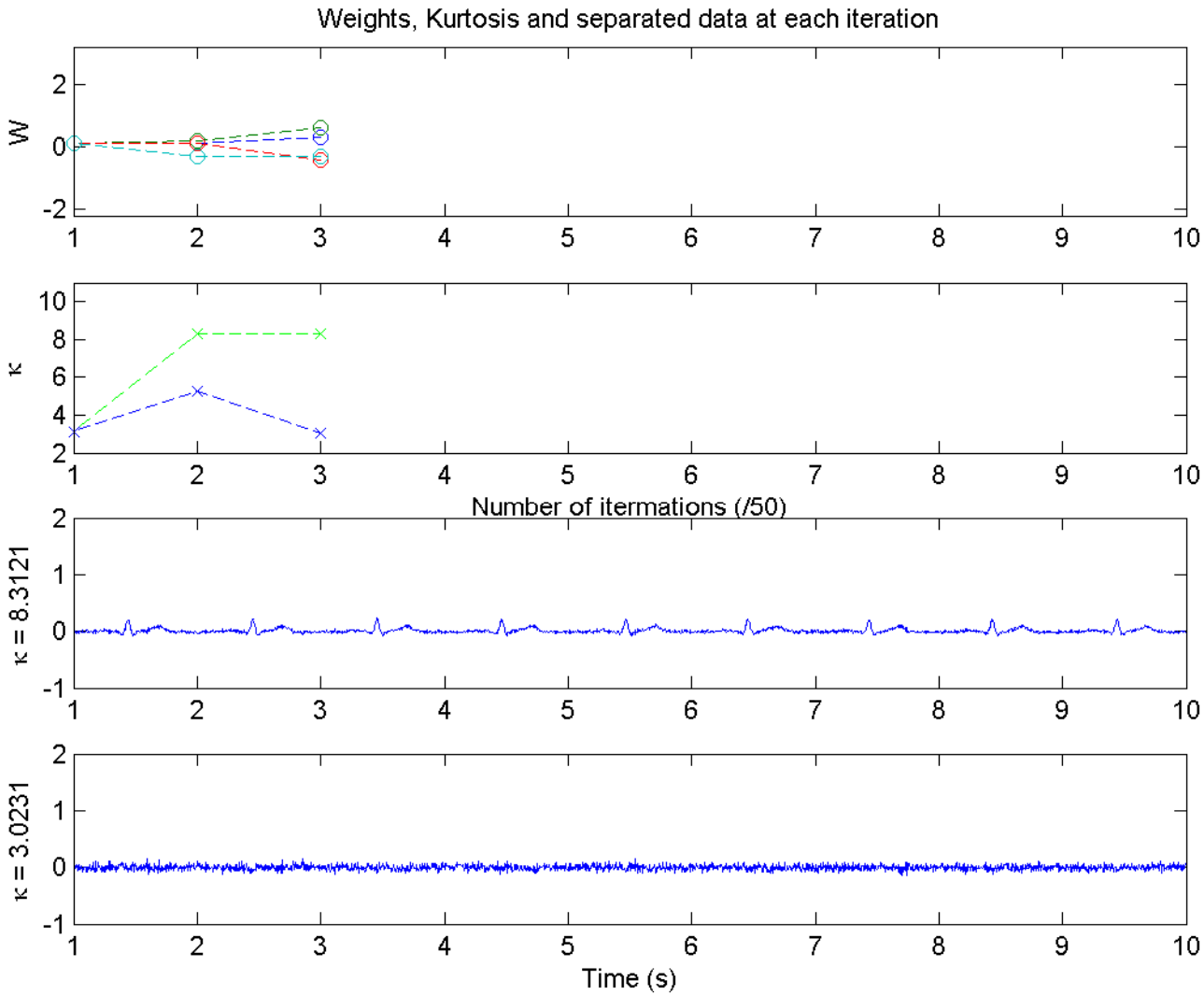
Iteratively update W and measure κ



Y_1

Y_2

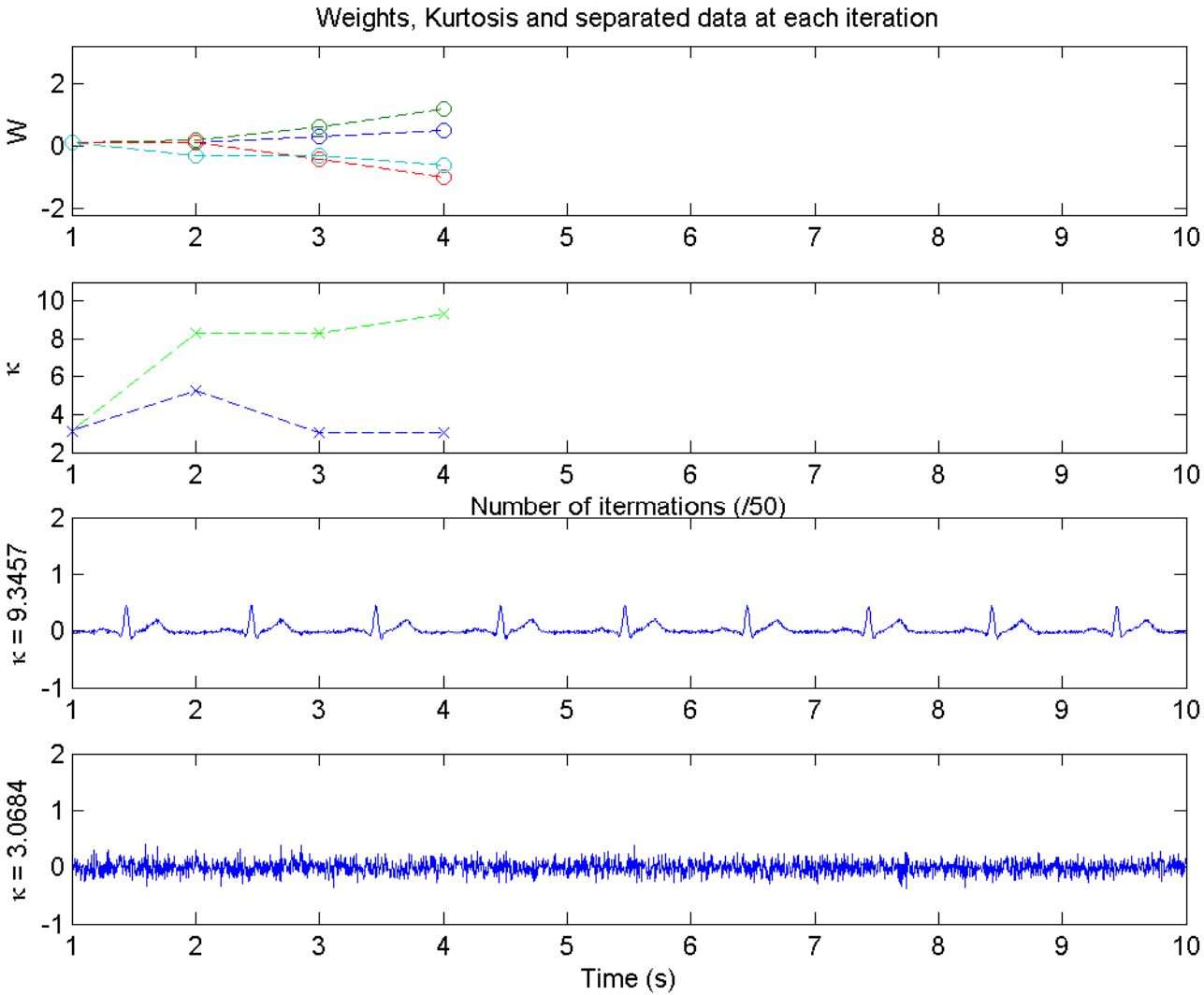
Iteratively update W and measure κ



Y_1

Y_2

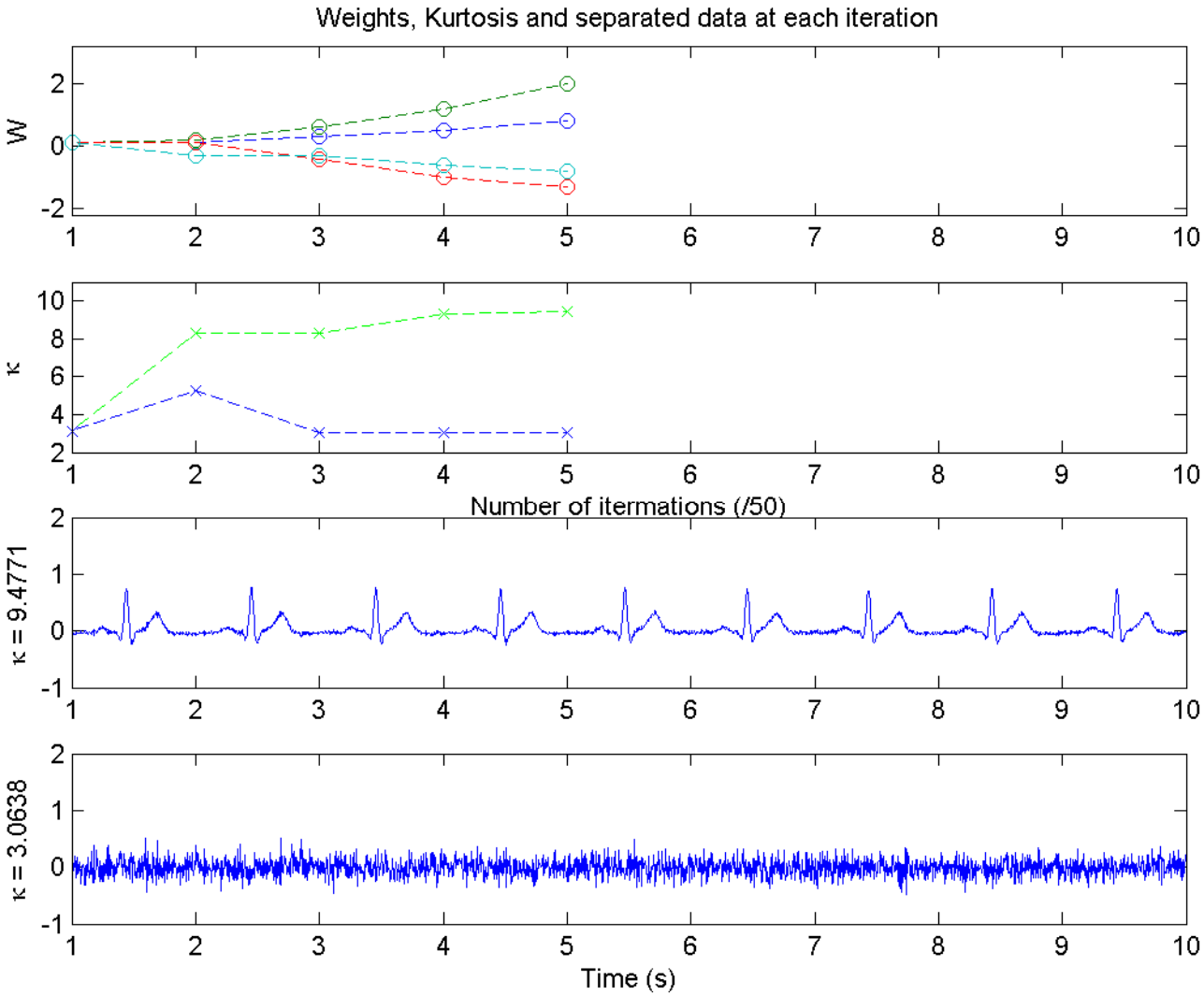
Iteratively update W and measure κ



Y_1

Y_2

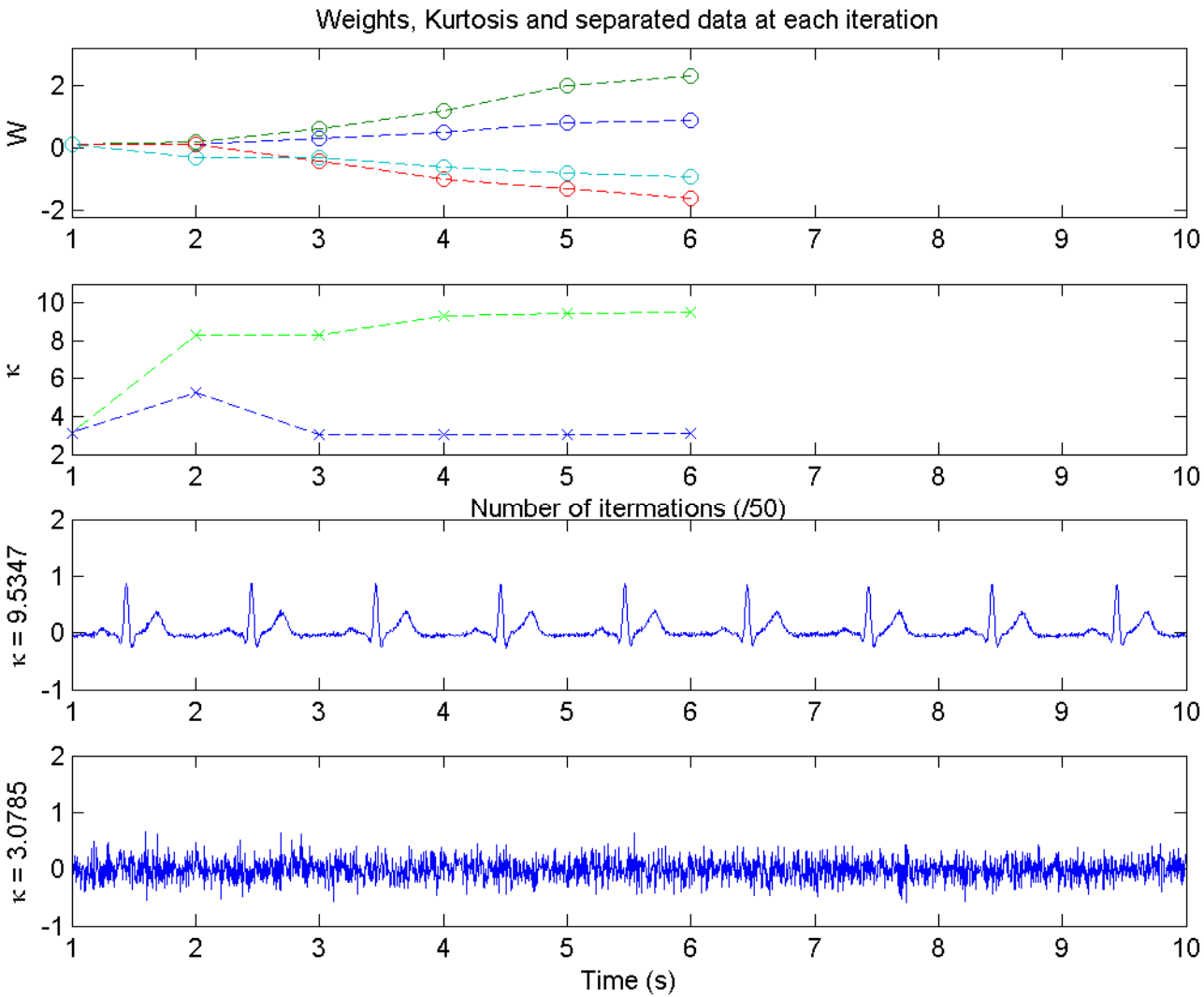
Iteratively update W and measure κ



Y_1

Y_2

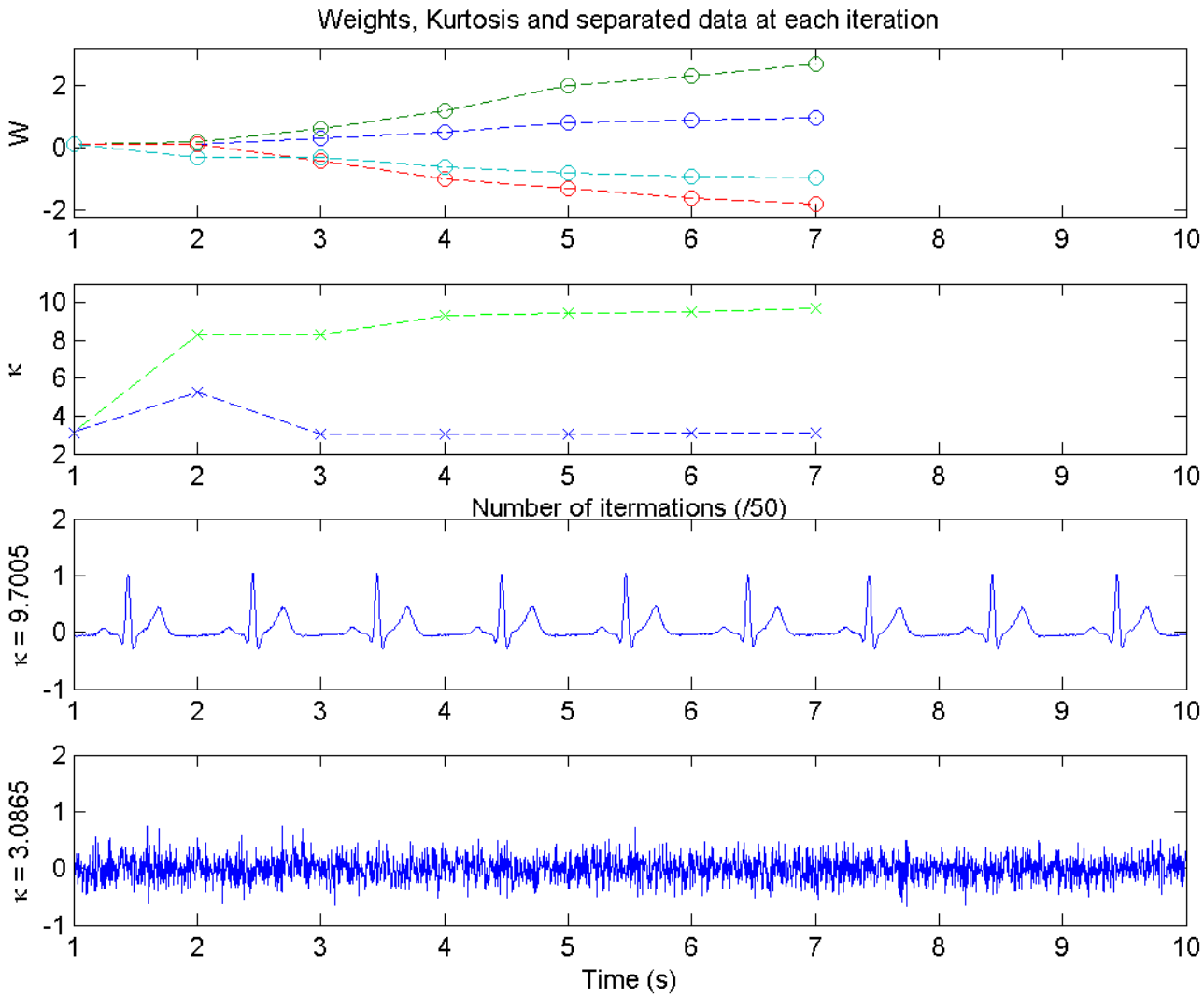
Iteratively update W and measure κ



Y_1

Y_2

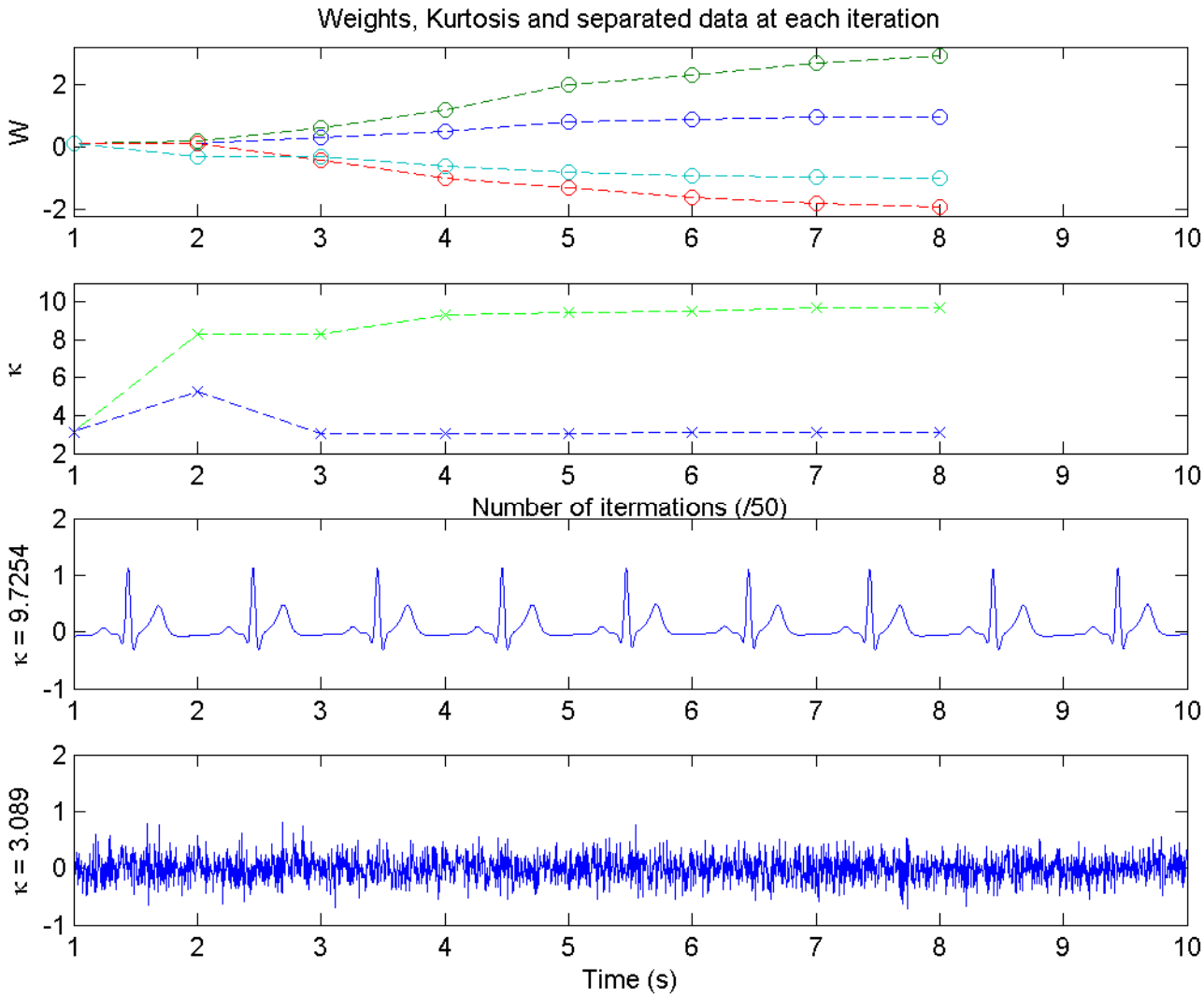
Iteratively update W and measure κ



Y_1

Y_2

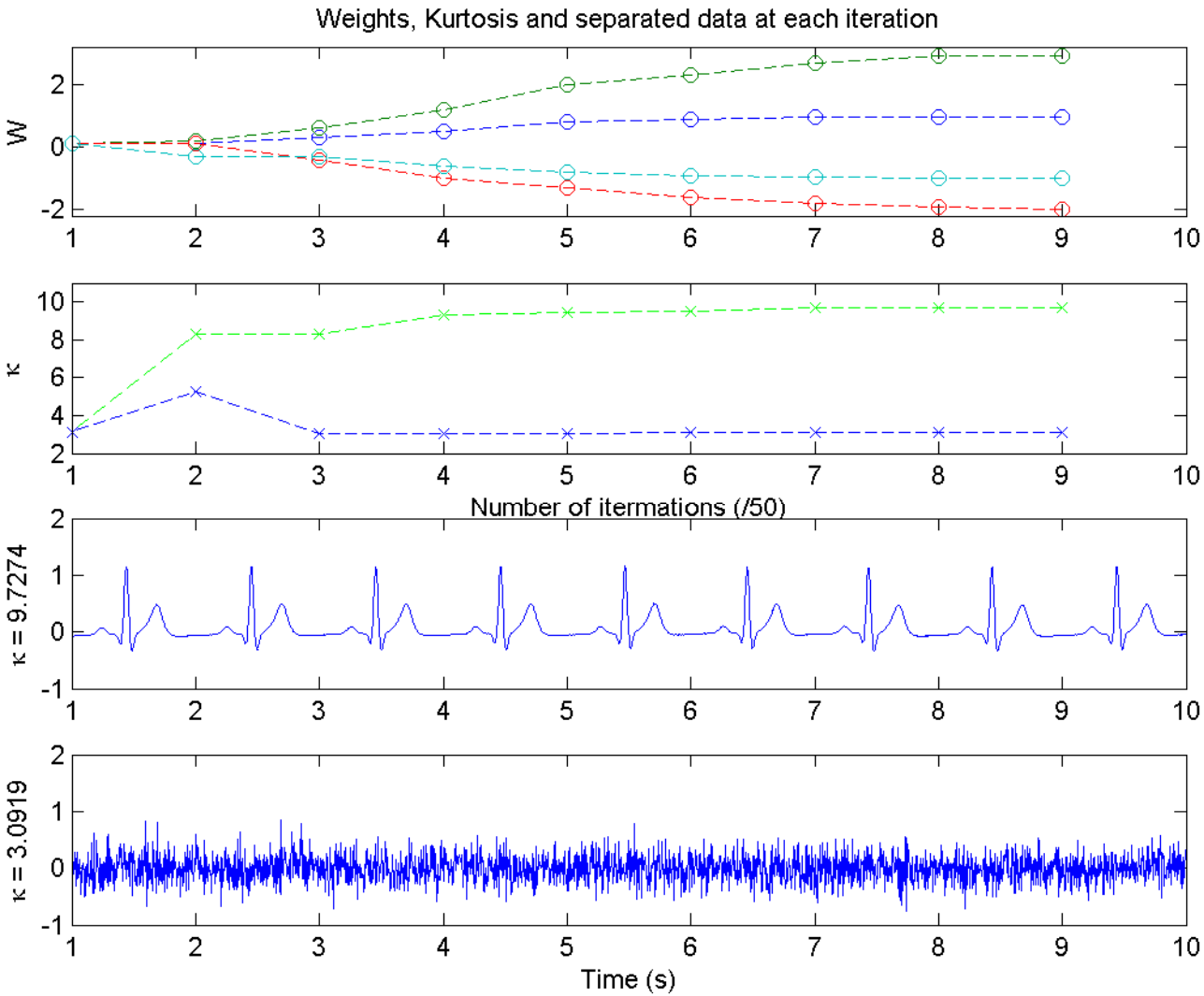
Iteratively update W and measure κ



Y_1

Y_2

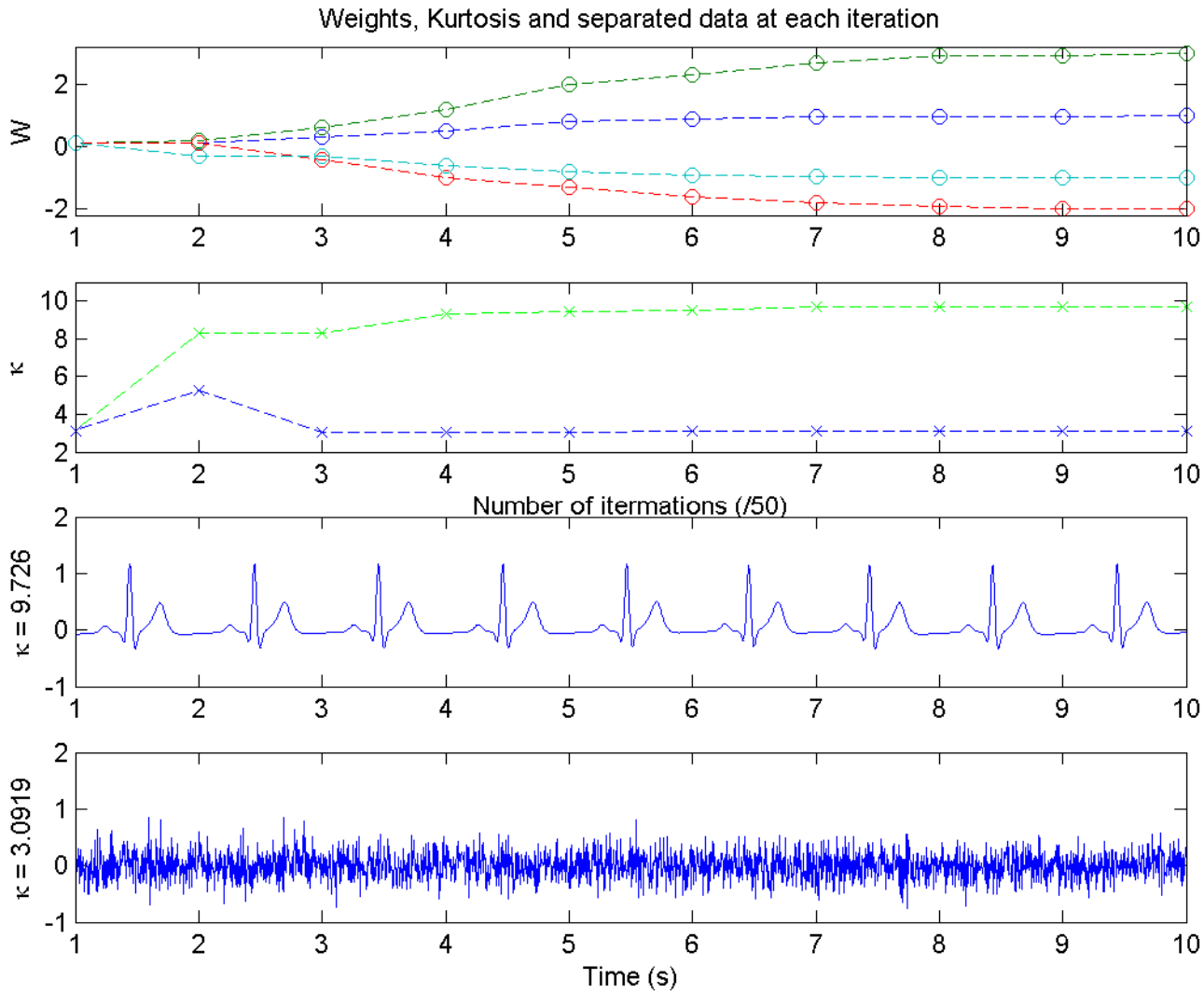
Iteratively update W and measure κ



Y_1

Y_2

Maximized κ for non-Gaussian signal



Y_1

Y_2

Outlier insensitive ICA cost functions

Measures of statistical independence

In general we require a measure of statistical independence which we maximise between each of the N components.

Non-Gaussianity is one approximation, but sensitive to small changes in the distribution tail.

Other measures include:

- Mutual Information I,
- Entropy (Negentropy, $\mathcal{J}$)… and
- Maximum (Log) Likelihood $\mathrm{L}(\mathbf{W})$

(Note: all are related to κ)

Entropy-based cost function

Kurtosis is highly sensitive to small changes in distribution tails.

A more robust measures of Gaussianity is based on differential entropy ***H*(*y*)**,

$$H(\mathbf{y}) = -\int P(\mathbf{y}) \log_2 P(\mathbf{y}) \mathrm{d}\mathbf{y}$$

… ***negentropy:***

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y})$$

where $\mathbf{y}_{gauss}$ is a Gaussian variable with the same covariance matrix as **y**. $\mathcal{J}(y)$ can be estimated from kurtosis …

$$\mathcal{J}(y) \approx \frac{1}{12} E\{y^3\}^2 + \frac{1}{48} \kappa(y)^2$$

Entropy: measure of randomness- Gaussians are maximally random

Minimising Mutual Information

Mutual information (MI) between two vectors ***x*** and ***y :***

$$I = H_x + H_y - H_{xy}$$

always non-negative and zero if variables are independent …

therefore we want to minimise MI.

MI can be re-written in terms of negentropy …

$$I(y_1, y_2, ..., y_m) = c - \sum_{i=1} J(y_i)$$

where *c* is a constant.

… differs from negentropy by a constant and a sign change

Independent source discovery using Maximum Likelihood

Generative latent variable modelling N observables, $\boldsymbol{X}$ ... from N sources, z_i through a linear mapping $\boldsymbol{W}=w_{ij}$

Latent variables assumed to be independently distributed

Find elements of $\boldsymbol{W}$ by gradient ascent - iterative update by

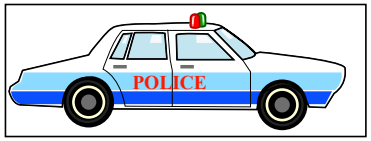
$$\Delta w_{ij} = \eta \frac{\partial Ł}{\partial w_{ij}}$$

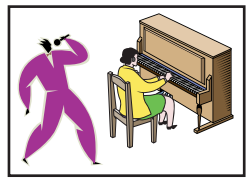
where η is some learning rate (const) … and $Ł(\mathbf{W})$ is our objective *cost function*, the **log likelihood**

$$\log_2 P(\mathbf{x}^m|\mathbf{A}) = \log_2 \det \mathbf{A} + \sum_i \log_2 p_i(a_{ij}\mathbf{x}_j)$$

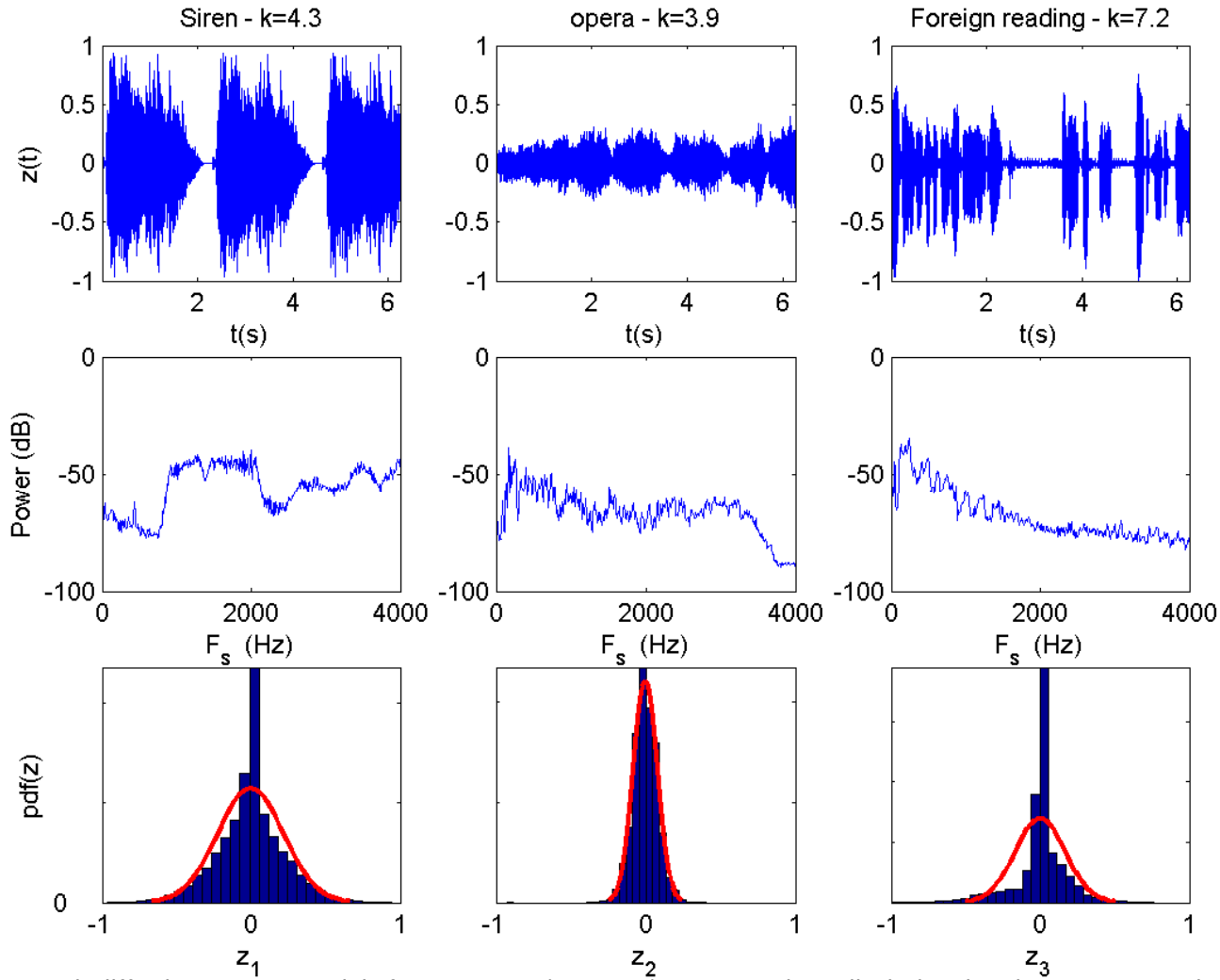
The cocktail party problem revisited

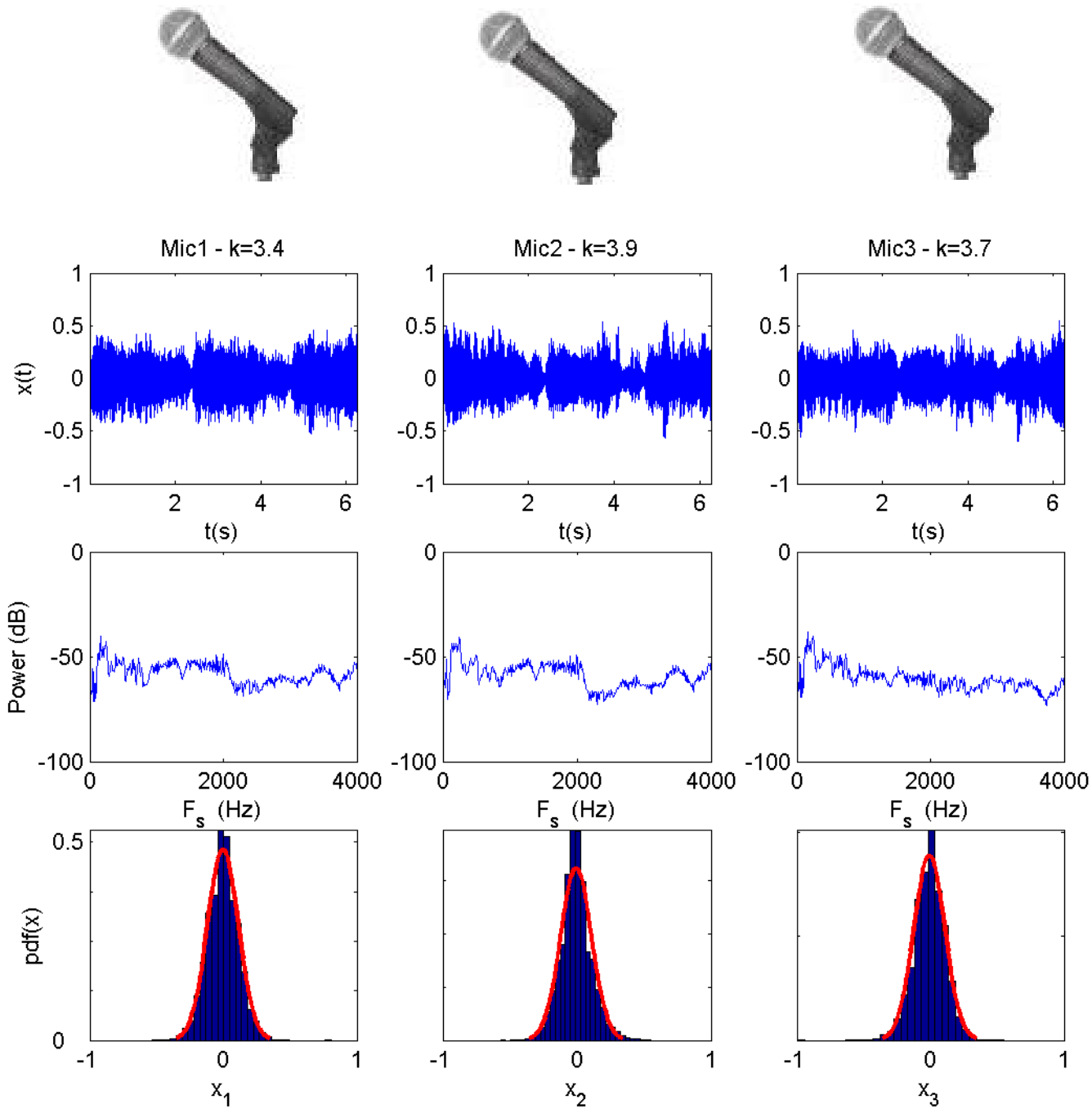
… some real examples using ICA

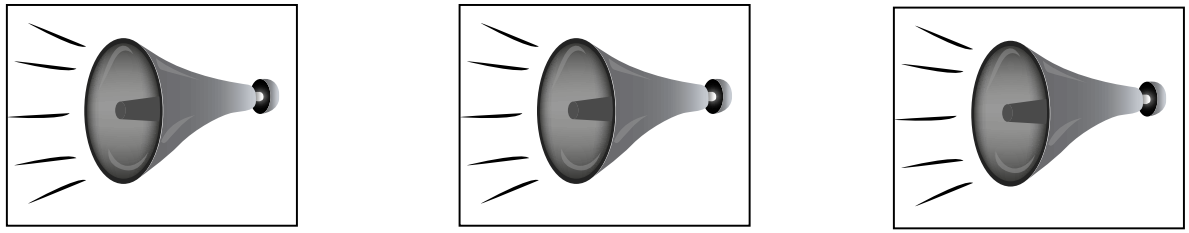




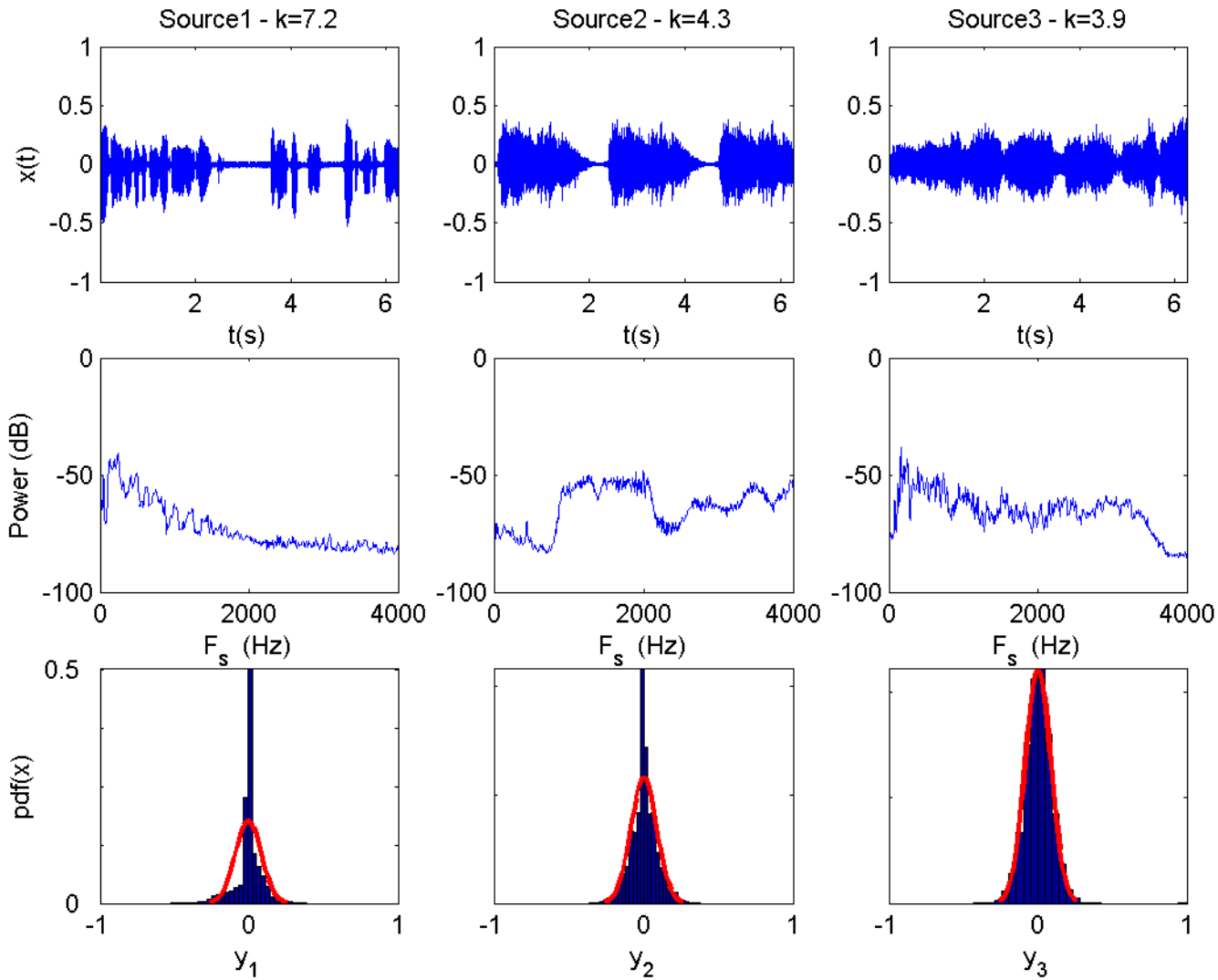
Figures by MIT OpenCourseWare.







Figures by MIT OpenCourseWare.



Observations

Separation of mixed observations into source estimates is excellent … apart from:

- Order of sources has changed
- Signals have been scaled

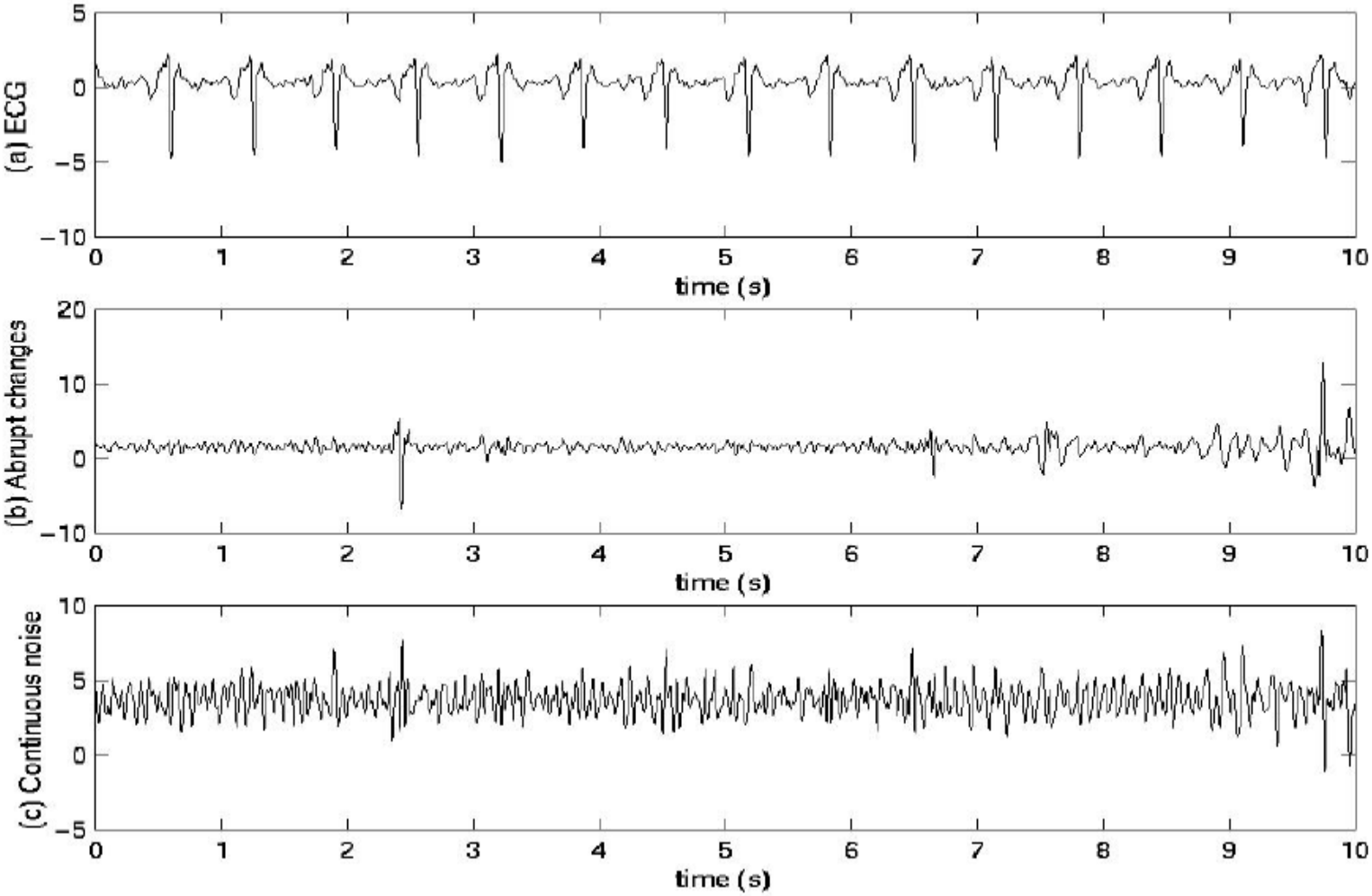
Why? … In $\boldsymbol{X}^T=\boldsymbol{A}\boldsymbol{Z}^T$, insert a permutation matrix $\boldsymbol{B}$ …

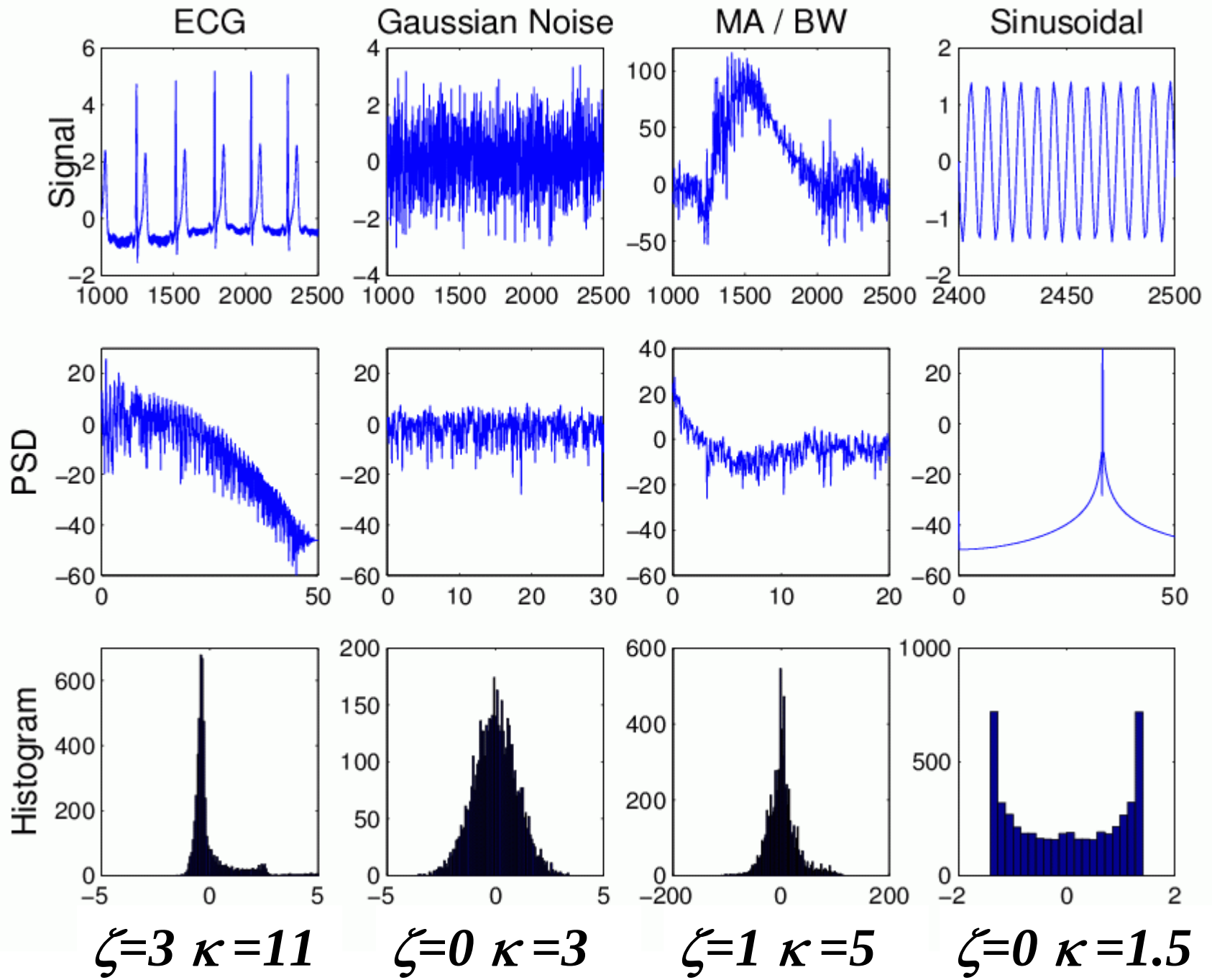
$\boldsymbol{X}^T=\boldsymbol{A}\boldsymbol{B}\boldsymbol{B}^{-1}\boldsymbol{Z}^T \Rightarrow \boldsymbol{B}^{-1}\boldsymbol{Z}^T$ … = sources with different col. order.

$\Rightarrow$ sources change by a scaling $\boldsymbol{A} \longrightarrow \boldsymbol{A}\boldsymbol{B}$

… ICA solutions are order and scale independent because κ is dimensionless

Separation of sources in the ECG





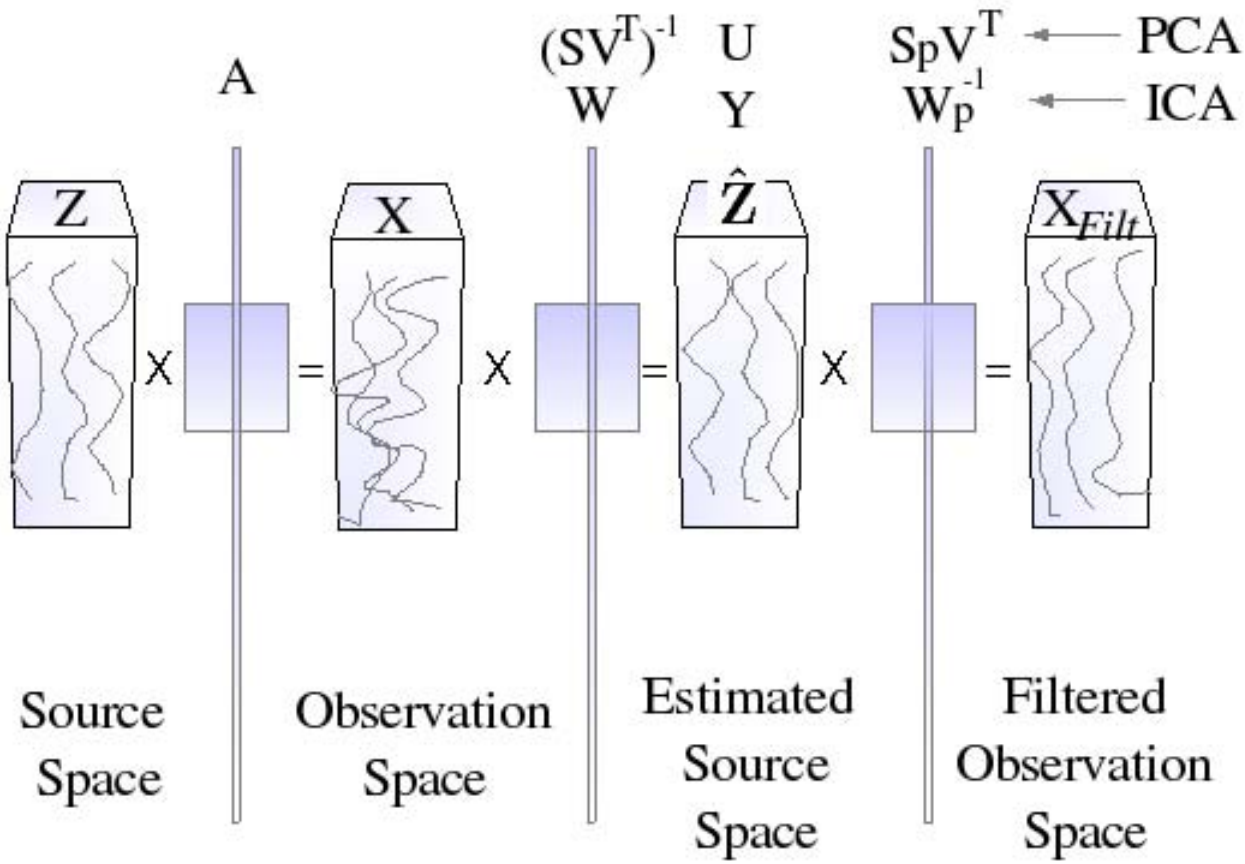
Transformation inversion for filtering

- Problem - can never know if sources are really reflective of the actual source generators - no gold standard

- De-mixing might alter the clinical relevance of the ECG features

- Solution: Identify unwanted sources, set corresponding (p) columns in $\boldsymbol{W}^{-1}$ to zero ($\boldsymbol{W}_p^{-1}$), then multiply back through to remove 'noise' sources and transform back into original observation space.

Transformation inversion for filtering



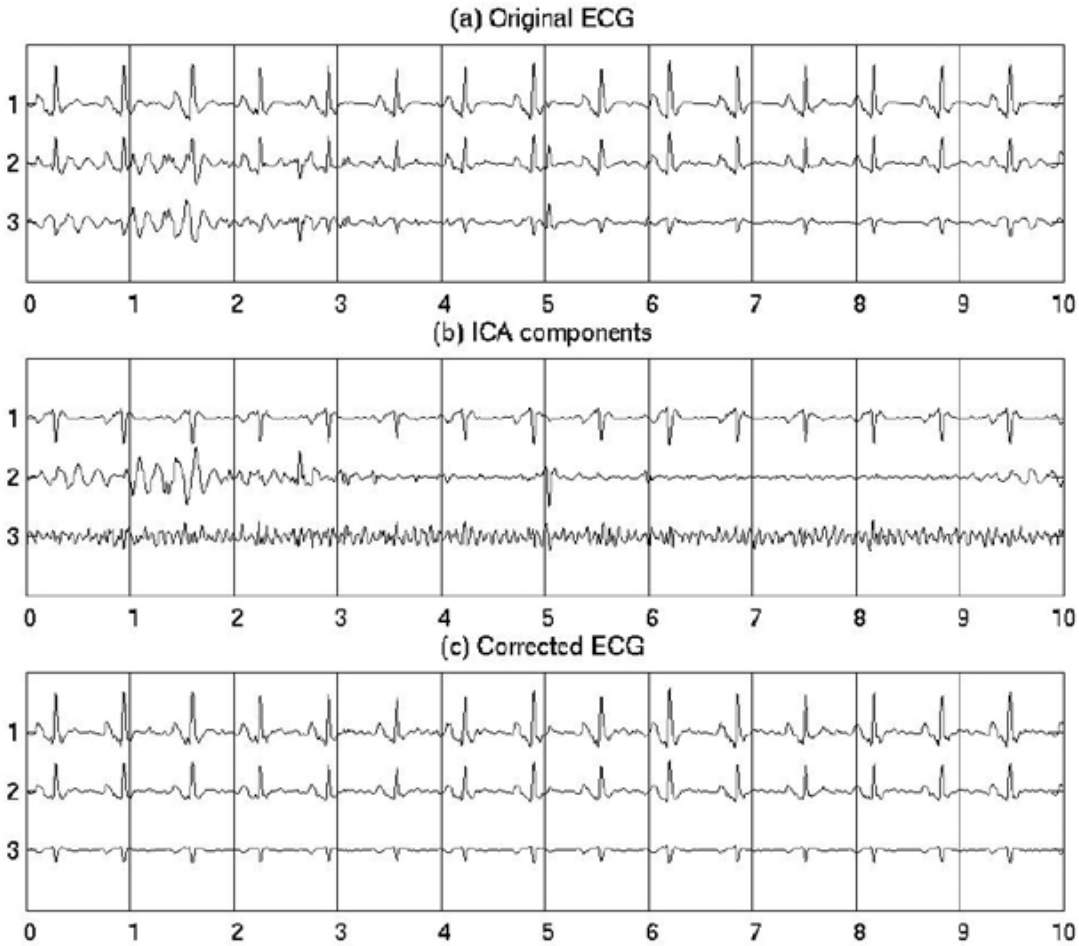
Real data

X

$Y=WX$

Z

$X_{filt} = W_p^{-1} Y$

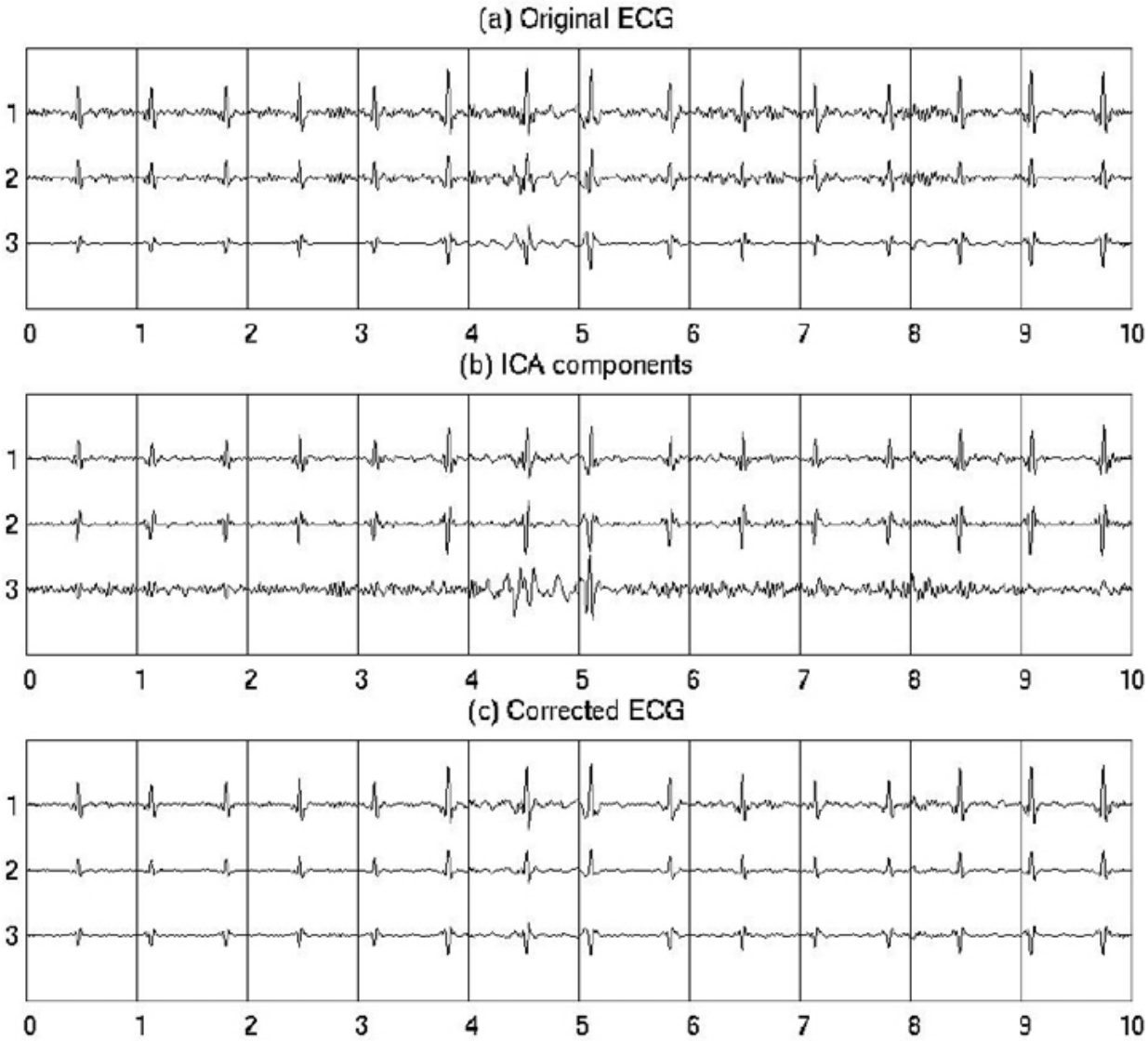


Source: He, Clifford, and Tarassenko. *Neural Computing & Applications* 15 no. 2 (April 2006): 105-116. doi:10.1007/s00521-005-0013-y.

X

$Y=WX$
Z

$X_{filt} = W_p^{-1} Y$

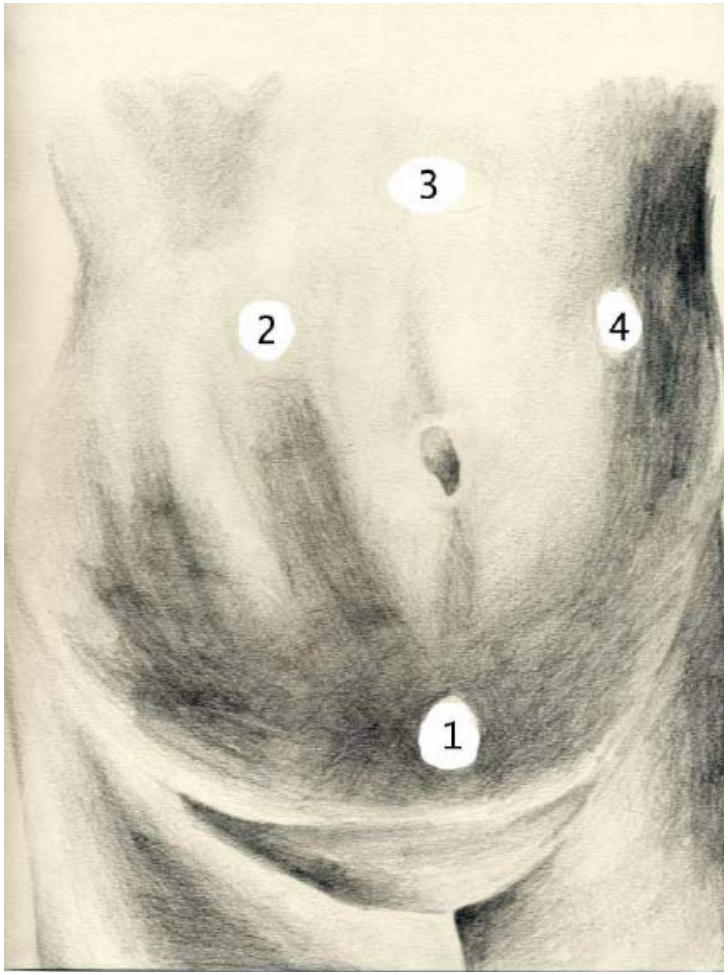


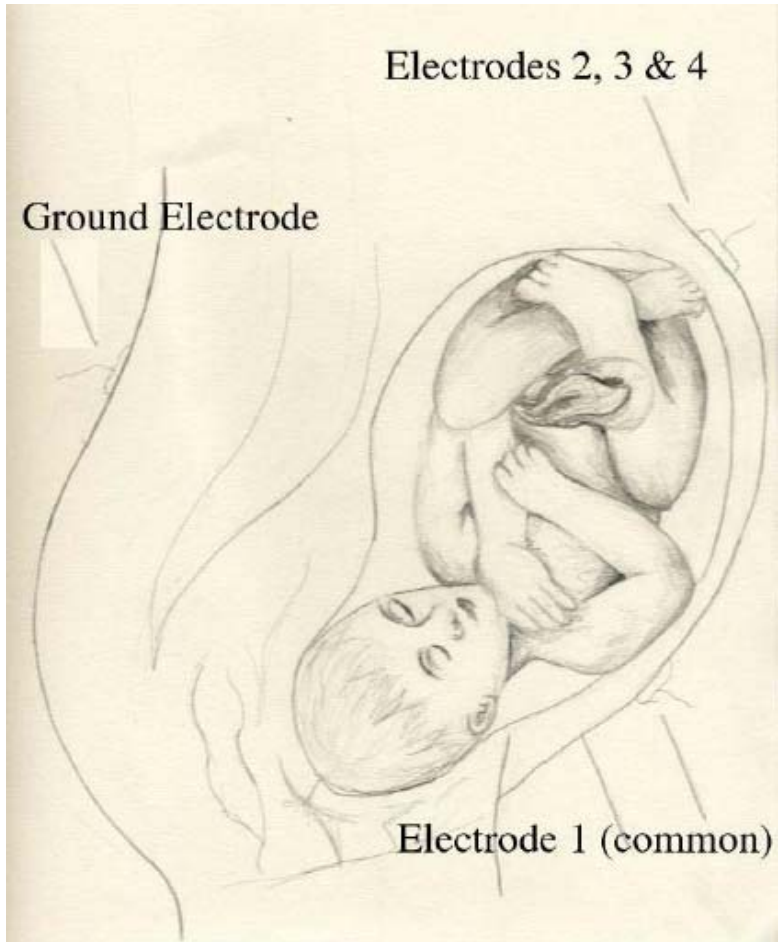
Source: He, Clifford, and Tarassenko. *Neural Computing & Applications* 15 no. 2 (April 2006): 105-116. doi:10.1007/s00521-005-0013-y.

Summary

- PCA is good for **Gaussian** noise separation
- ICA is good for **non-Gaussian** 'noise' separation
- PCs have obvious meaning - highest *energy* components
- ICA - derived sources : **arbitrary scaling/inversion & ordering**
  …. need energy-independent heuristic to identify signals / noise
- Order of ICs change - IC space is **derived** from the data.
  - PC space only changes if SNR changes.
- ICA assumes **linear** *mixing matrix*
- ICA assumes **stationary** mixing
- De-mixing performance is function of lead position
- ICA requires as many *sensors* (ECG leads) as *sources*
- Filtering - discard certain dimensions then *invert transformation*
- In-band noise can be removed - unlike Fourier!

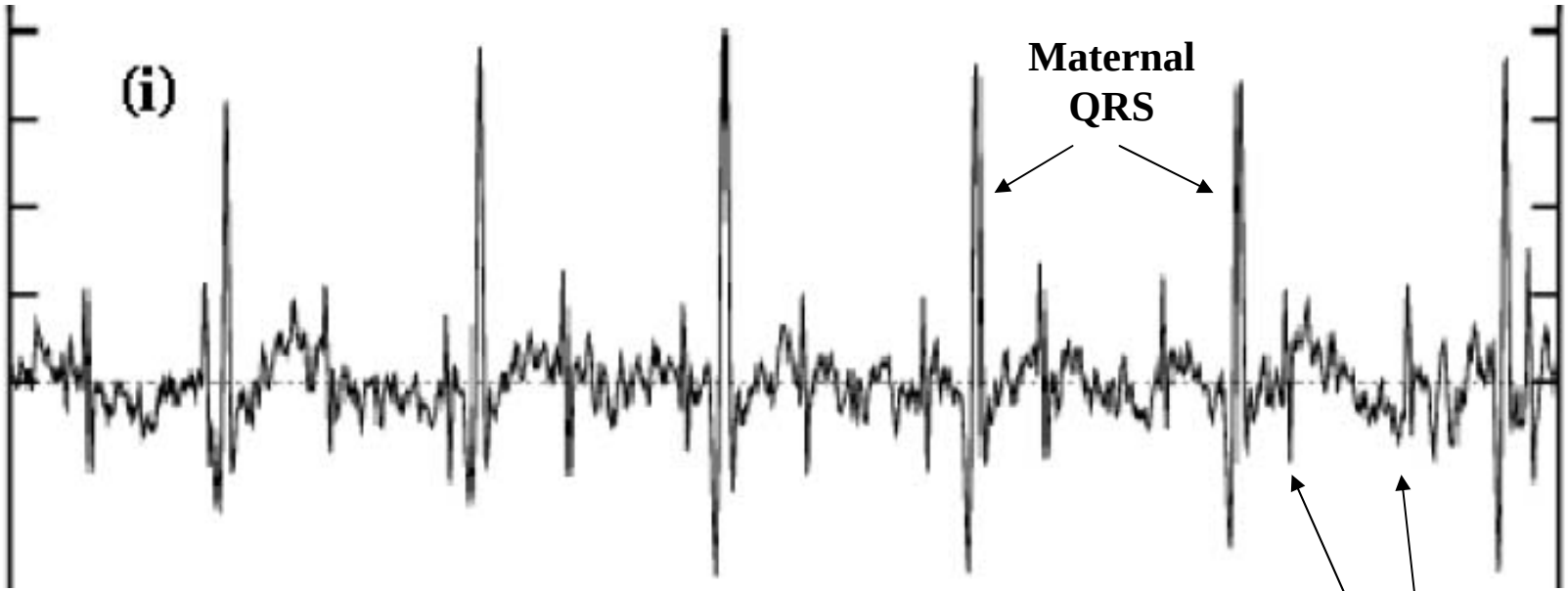
Fetal ECG lab preparation





Courtesy of B. Campbell. Used with permission.

Fetal abdominal recordings

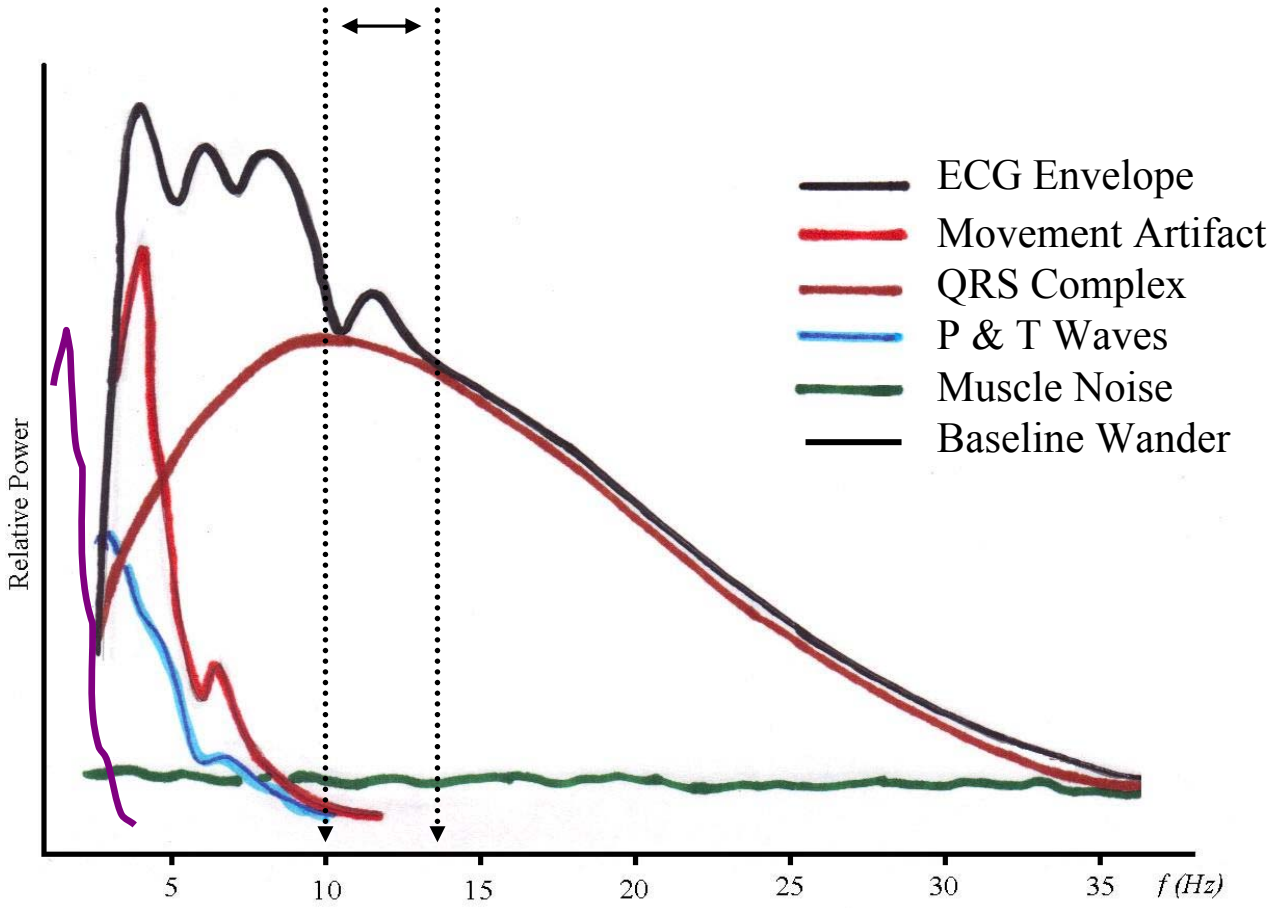


Richter M., T. Schreiber, and D. T. Kaplan. "Fetal EEG Extraction with Nonlinear Phase Space Projections." *IEEE Trans Bio Med Eng* 45 (January 1998): 133–137. Copyright © 1998 IEEE. Used with permission.

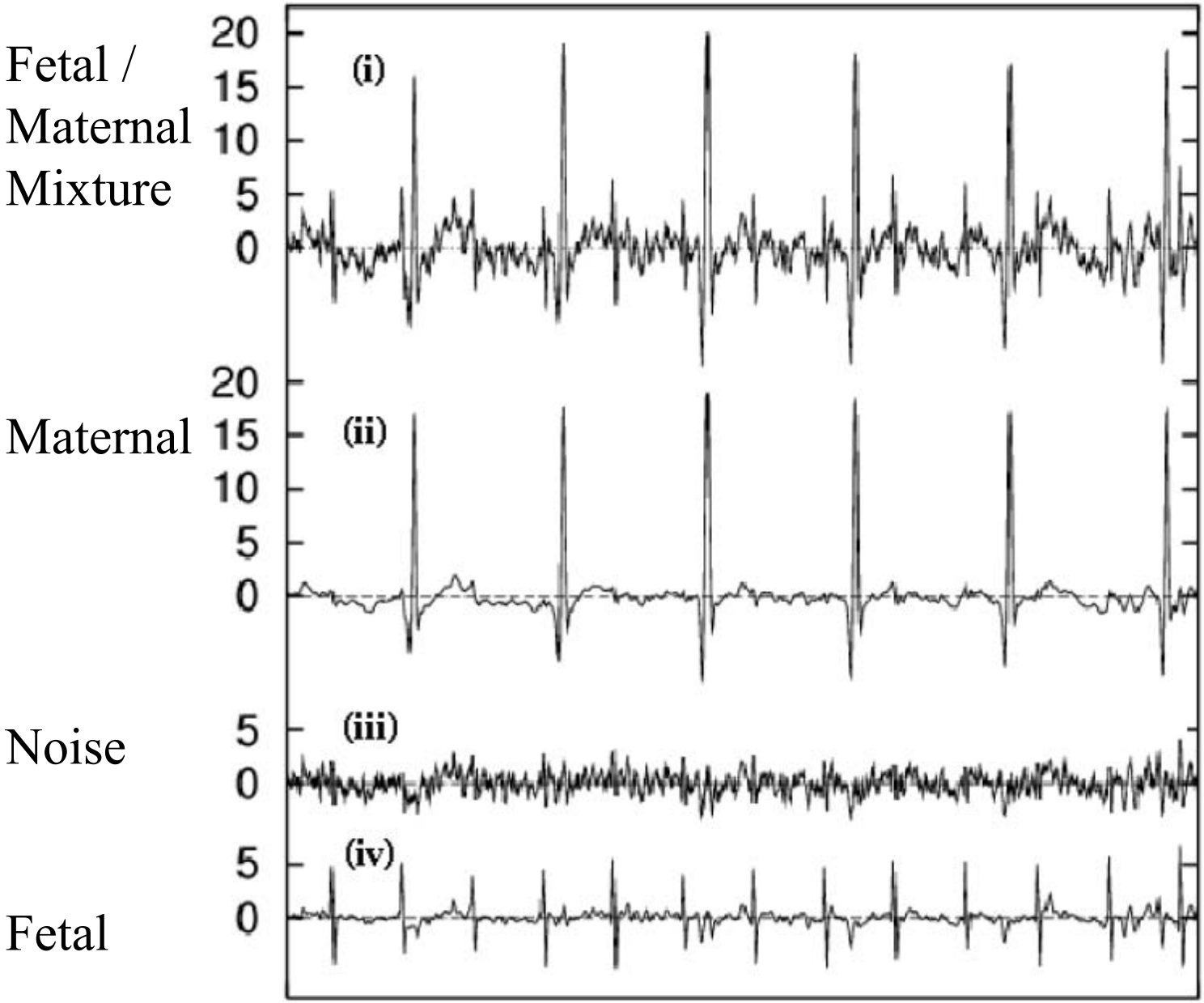
- Maternal ECG is much larger in amplitude
- Maternal and fetal ECG overlap in time domain
- Maternal features are broader, but
- Fetal ECG is *in-band* of maternal ECG
  (they overlap in freq domain)
- 5 second window … Maternal HR=72 bpm / Fetal HR = 156bpm

MECG & FECG spectral properties

Fetal QRS power region



Adapted from W. J. Tompkins (ed.) *Biomedical Digital Signal Processing: C Language Examples and Laboratory Experiments for the IBM PC*. Englewood Cliffs, NJ: Prentice Hall, 1993.



Richter M., T. Schreiber, and D. T. Kaplan. "Fetal EEG Extraction with Nonlinear Phase Space Projections." *IEEE Trans Bio Med Eng* 45 (January 1998): 133–137. Copyright © 1998 IEEE. Used with permission.