Independent Component Analysis

Lecture 14

ICA: Motivation

■ Cocktail party problem:

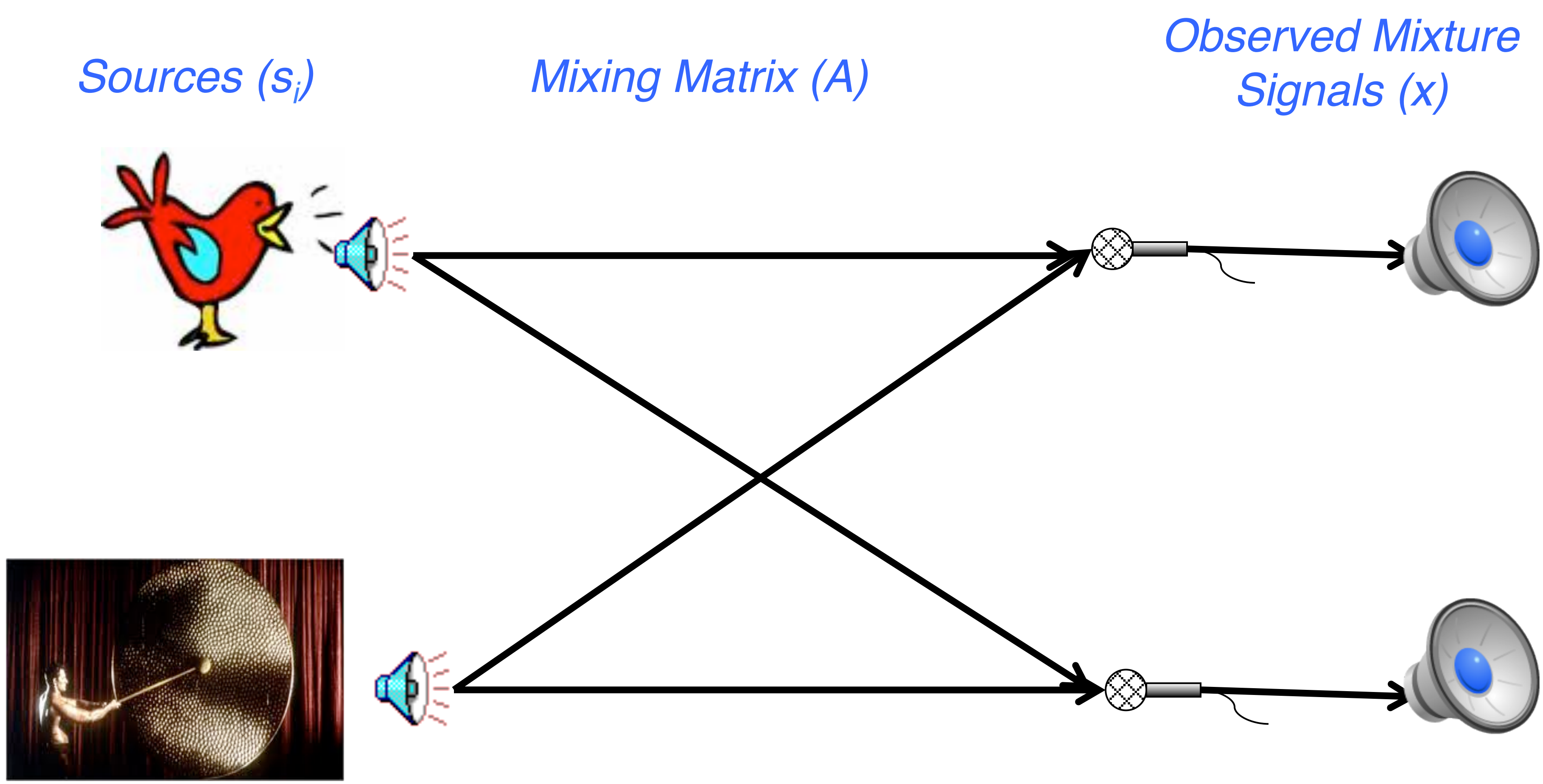
- Imagine you are in a room where two people are speaking simultaneously.
- You have two microphones placed in two different locations.
  - Microphones will give you two recorded time signals which we are denoted by $x_1(t)$ and $x_2(t)$, with x_1 and x_2 the amplitudes and t the time index.

$$x_1(t) = a_{11}s_1(t) + a_{12}s_2(t)$$
$$x_2(t) = a_{21}s_1(t) + a_{22}s_2(t)$$

  - This is of course a simple model where we have omitted time delays and reverberances in the room.

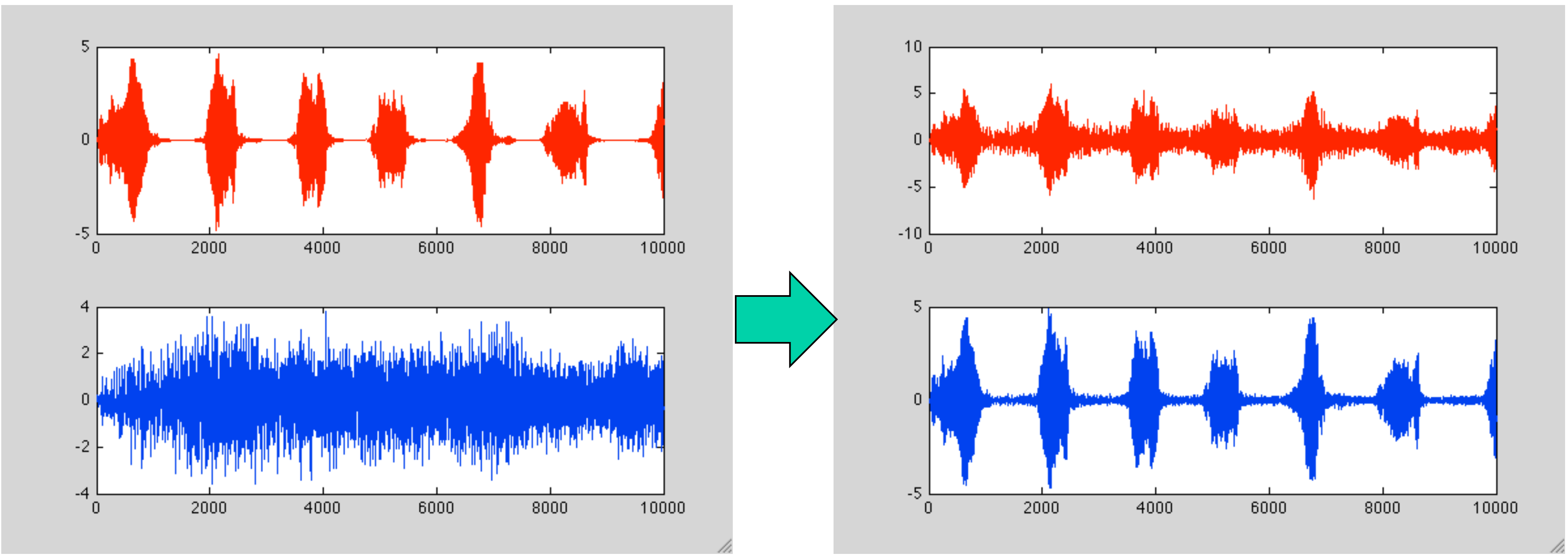
ICA: Motivation

- **Cocktail party problem:**



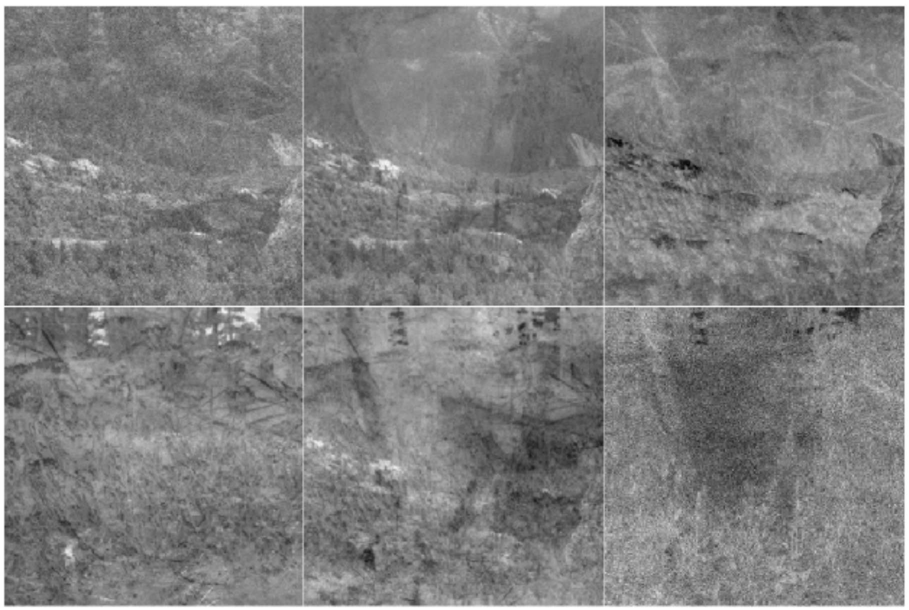
ICA

Sources (s_i) | Observed Mixture Signals (x)



ICA

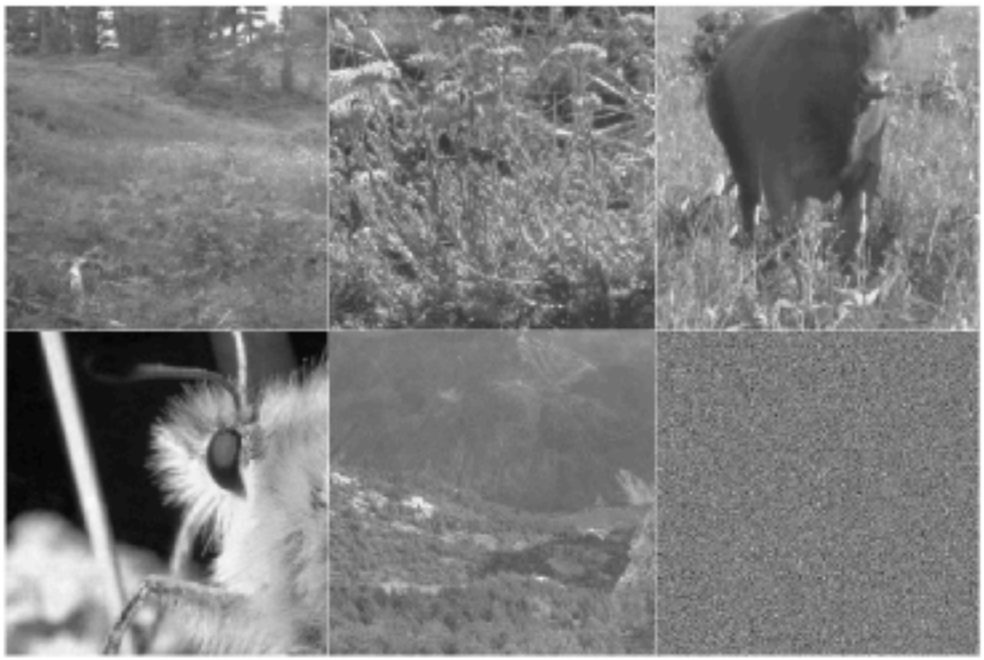
Image processing



- 6 images
- linear mixtures of 6 originals
- determine originals

ICA

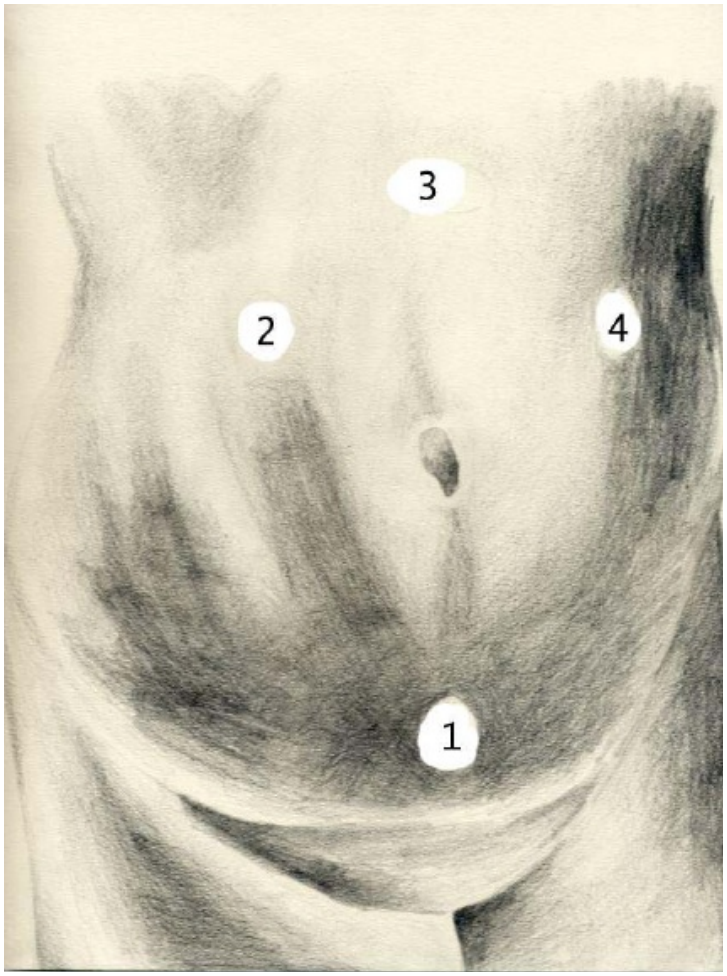
■ Image processing

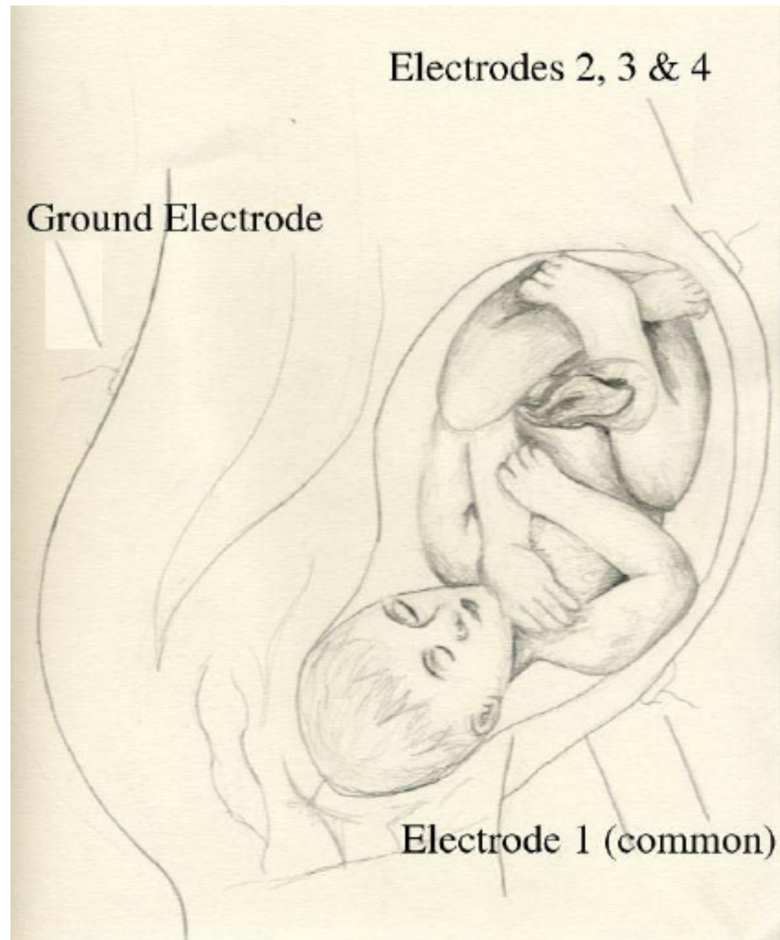


- independent latent (hidden) variables
- linear phenomenon

ICA Motivation

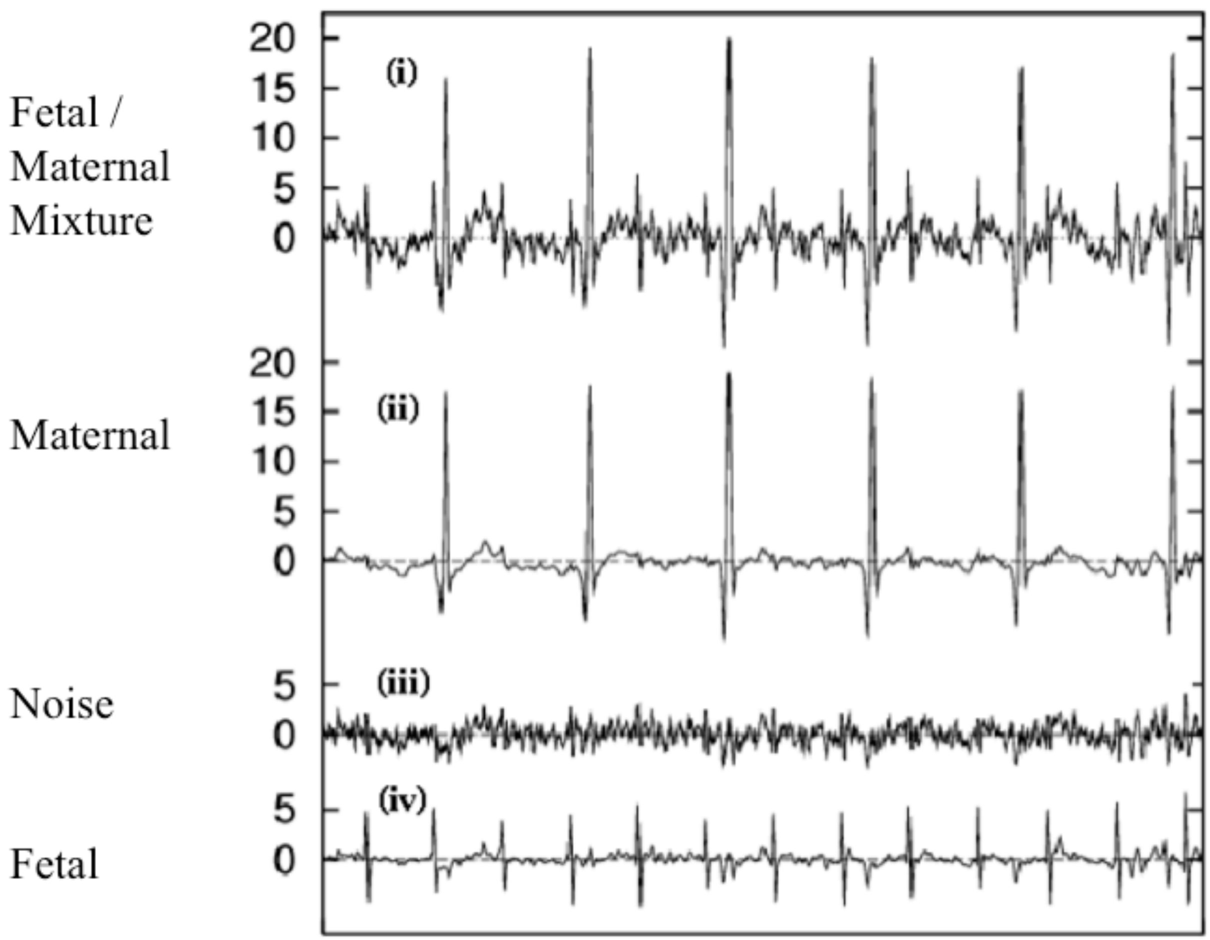
- **Fetal ECG**





ICA Motivation

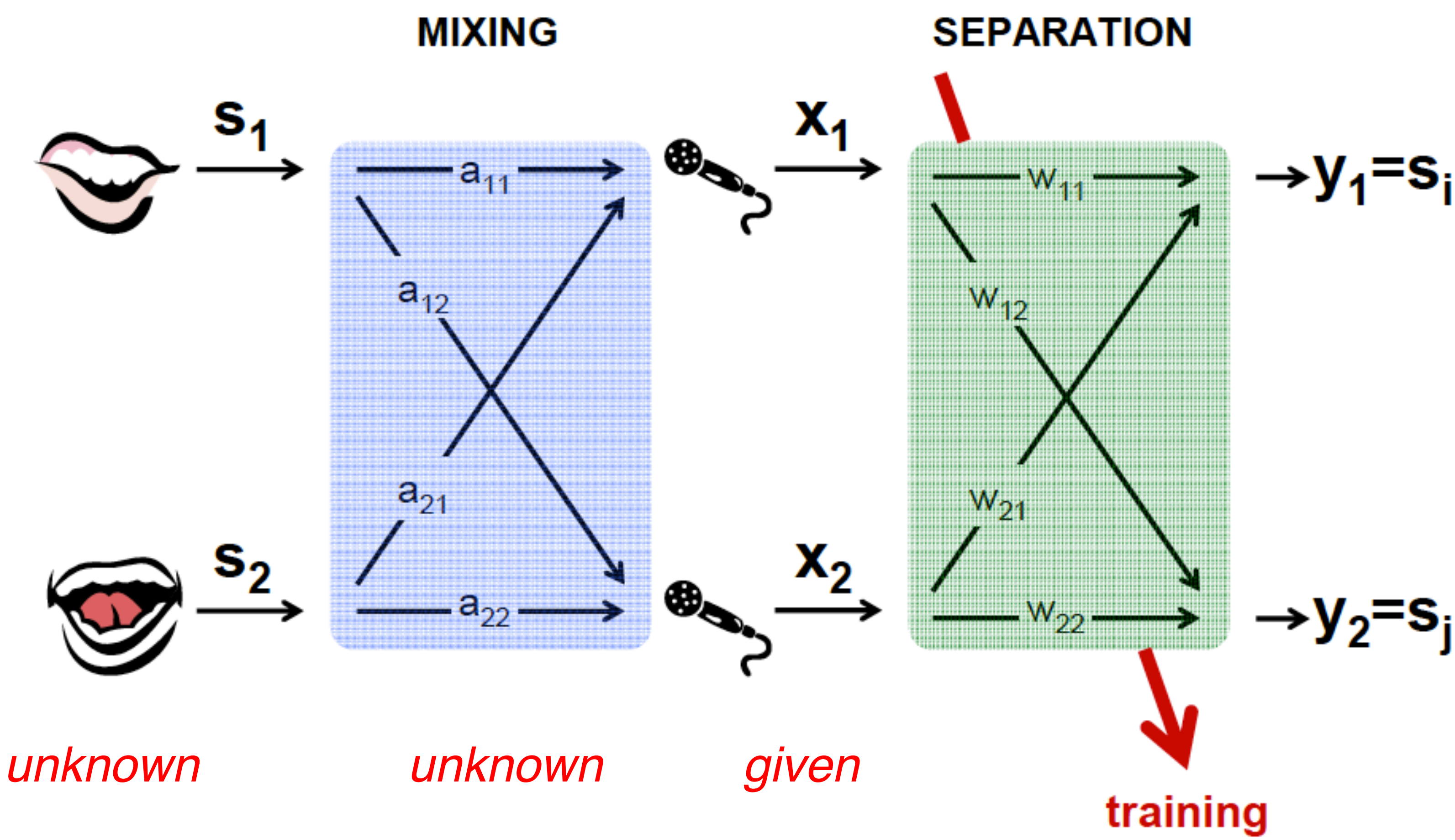
- **Fetal ECG**



ICA: Motivation

- **Electroencephalogram(EEG):**
  - The EEG data consists of recordings of electrical potentials generated by mixing some underlying components of brain activity
  - We would like to the original components of brain activity but we can only observe mixture of components

ICA



From Gutierrez-Osuna

ICA Problem:

- **Assume that we observe n linear mixtures x_1, x_2, …x_n, from n independent observers**

$$x_j(t) = a_{j1}s_1(t) + a_{j2}s_2(t) + \cdots + a_{jn}s_n(t)$$

- **Or, using matrix notation**

$$x = As$$

- **Our goal is to find a de-mixing matrix W such that**

$$s = Wx$$

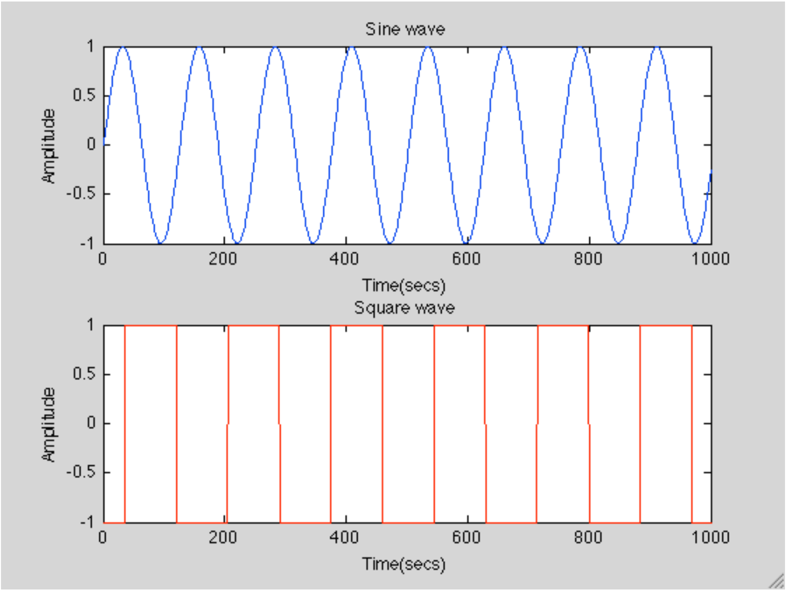
- **Assumptions**
  - Both mixture signals and source signals are zero-mean (i.e. $E[x_i]=E[s_j]=0,\ \forall i,j$)
    - If not, we simply subtract their means
  - The sources have non-Gaussian distributions
    - More on this in a minute
  - The mixing matrix is square, i.e., there are as many sources as mixing signals
    - This assumption, however, can sometimes be relaxed

From Gutierrez-Osuna

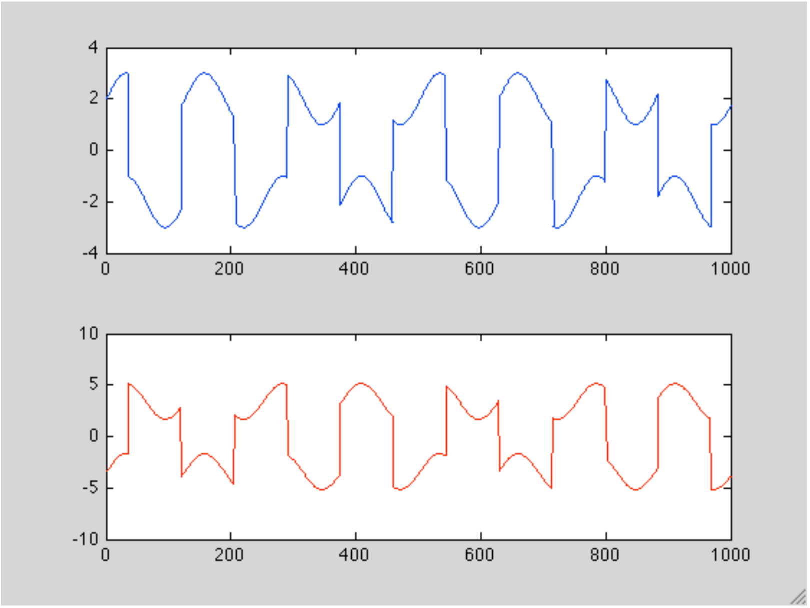
An Example

- **Given the observed signal can we find the sources?**

Sources (s_i)



Observed Mixture Signals (x)



Independence vs. uncorrelatedness

- **What is independence?**
  - Note: Variables s_1 and s_2 are independent but mixture variables x_1 and x_2 are not
  - Two random variables y1 and y2 are said to be independent if knowledge of the value of y1 does not provide any information about the value of y2, and viceversa

$$p(y_1 l y_2)=p(y_1)=>p(y_1,y_2)=p(y1)p(y_2)$$

- **What is uncorrelatedness?**
  - Two random variables y1 and y2 are said to be uncorrelated if their covariance is zero

$$E[y_1^2 y_2^2=0]$$

- **Equivalences**
  - Independence implies uncorrelatedness
  - Uncorrelatedness DOES NOT imply independence…
    - Unless the random variables y1 and y2 are Gaussian, in which case uncorrelatedness and independence are equivalent

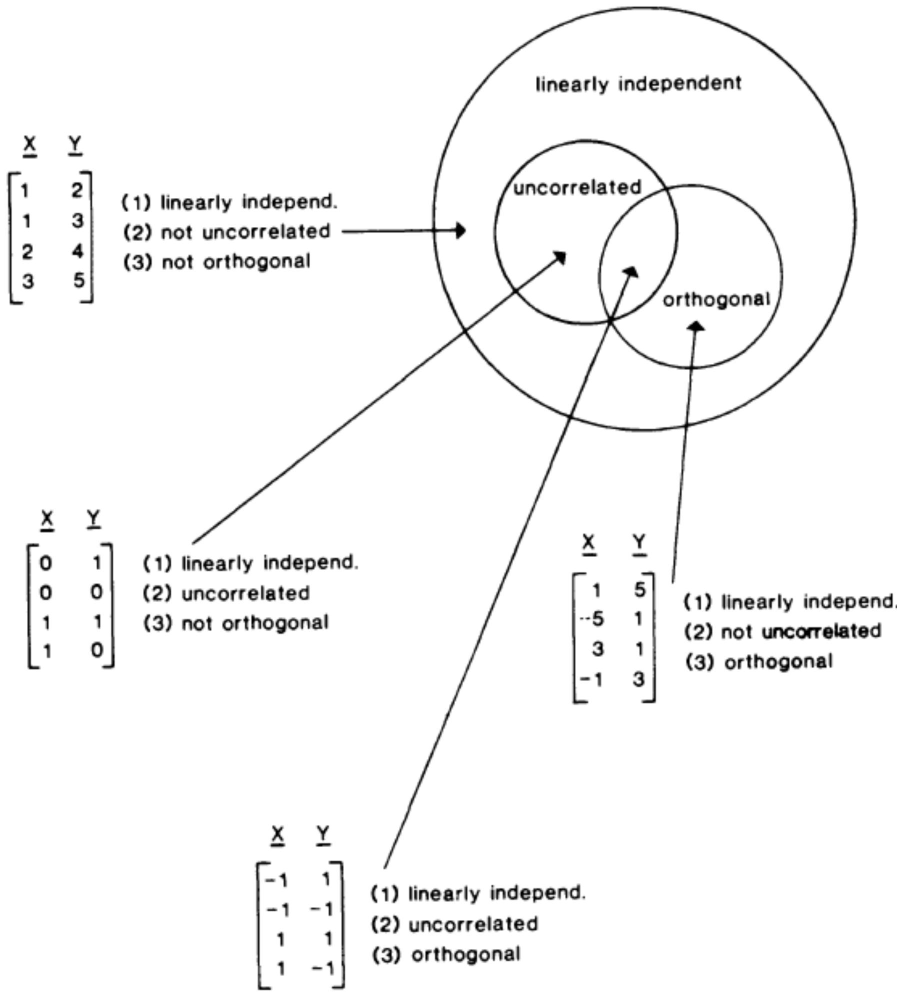
From Gutierrez-Osuna

Geometric View

- Linearly independent variables are those with vectors that do not fall along the same line; that is, there is no multiplicative constant that will expand, contract, or reflect one vector onto the other

- Orthogonal variables are a special case of linearly independent variables

- "uncorrelated" implies that once each variable is centered (i.e., the mean of each vector is subtracted from the elements of that vector), then the vectors are perpendicular.



Rodgers et al., 1984

Independence and non-Gaussianity

- **A necessary condition for ICA to work is that the signals be non-Gaussian. Otherwise, ICA cannot resolve the independent directions due to symmetries**
  - The joint density of unit variance gaussian s1 & s2 is symmetric. So it doesn‘t contain any information about the directions of the cols of the mixing matrix A. So A cannot be estimated.
  - Besides, if signals are Gaussian, one may just use PCA to solve the problem (!)
- **We will now show that finding the independent components is equivalent to finding the directions of largest non-Gaussianity**
  - For simplicity, let us assume that all the sources have identical distributions
  - Our goal is to find the vector *w such that* $y=w^T x$ *is equal to the sources*

From Gutierrez-Osuna

Why non-Gaussianity?

- **Consider an example n =2, such that**

$$s \sim N(0, I)$$

  - I is a 2x2 identity matrix
  - Contours are circles centered at origin, and is rotationally symmetric

- **We observed**

$$x = As$$

  - x will be Gaussian, with zero mean and the covariance is:

$$E\left[xx^T\right] = E\left[Ass^T A^T\right] = AA^T$$

  - *$x \sim N(0, AA^T)$*

From Ng's notes, Stanford

Why non-Gaussianity?

- **Now let R be an arbitrary orthogonal matrix**

$$RR^T = R^T R = I$$

- **Let**

$$A' = AR$$

  - *If that data has been mixed using A' instead of A, we would observe x'=A's.*
  - *x' is also Gaussian distributed with zero mean and the covariance matrix is:*

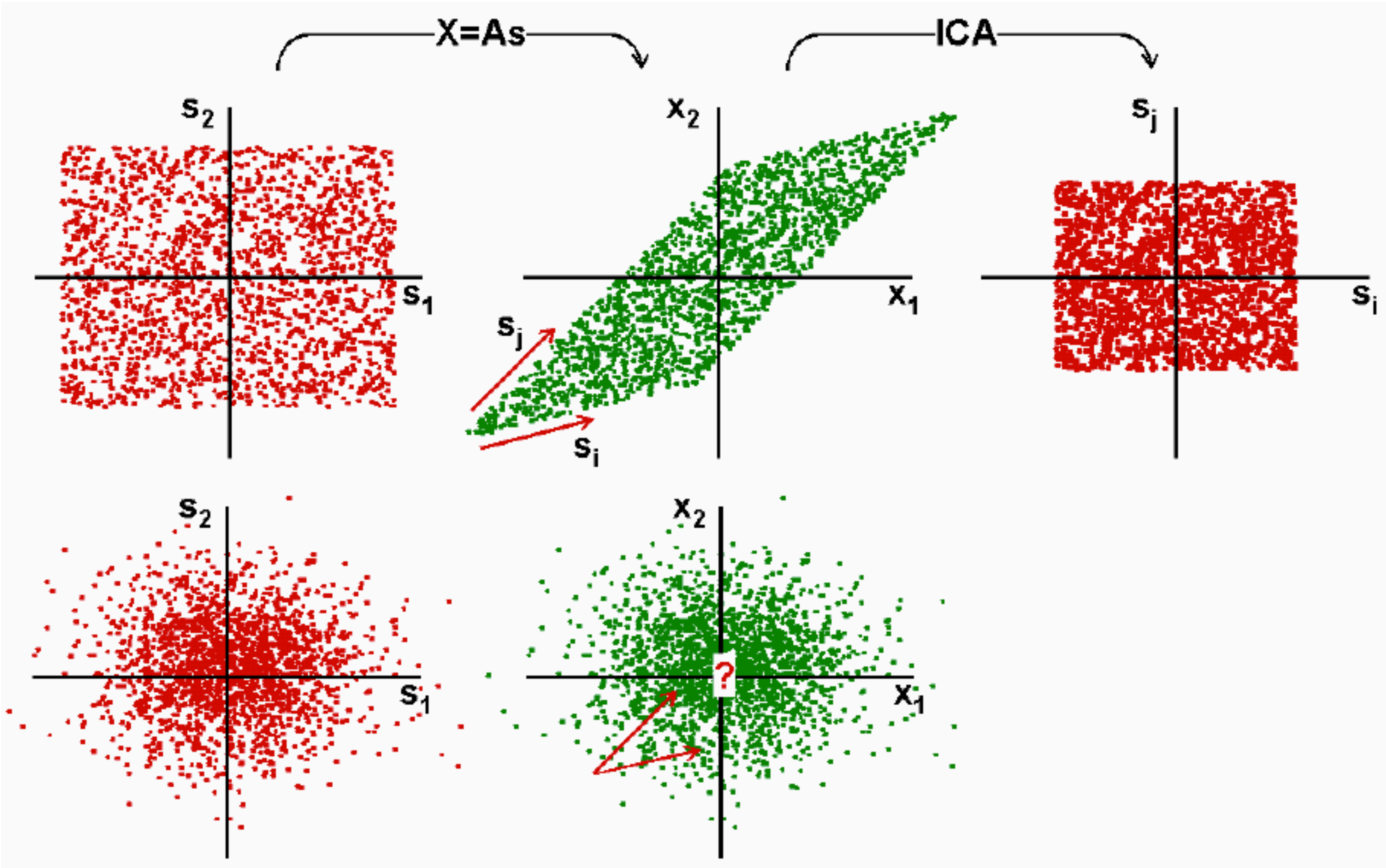
$$E\left[x' x'^T\right] = E\left[A' s s^T A'^T\right] = A' A'^T = A R R^T A^T = A A^T$$

  - *x' ~ N(0,AA^T)*

- **This implies that is an arbitrary rotational component that cannot be determined form the data**

From Ng's notes, Stanford

Why can't Gaussian variables be used with ICA?



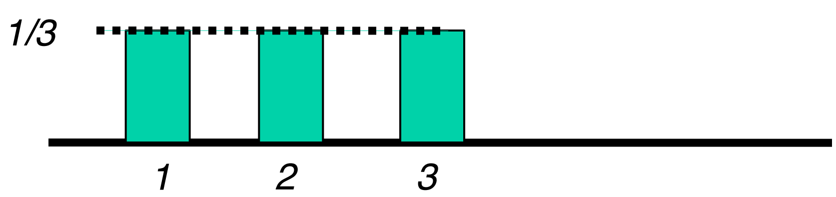
From Gutierrez-Osuna

Central Limit Theorem

- **The distribution of sum of independent random variables, which itself is a random variable, tends toward a Gaussian Distribution as the number of terms in the sum increases**

$$X = \begin{cases} 1 & \text{with probability } 1/3, \\ 2 & \text{with probability } 1/3, \\ 3 & \text{with probability } 1/3. \end{cases}$$

Probability Mass Function

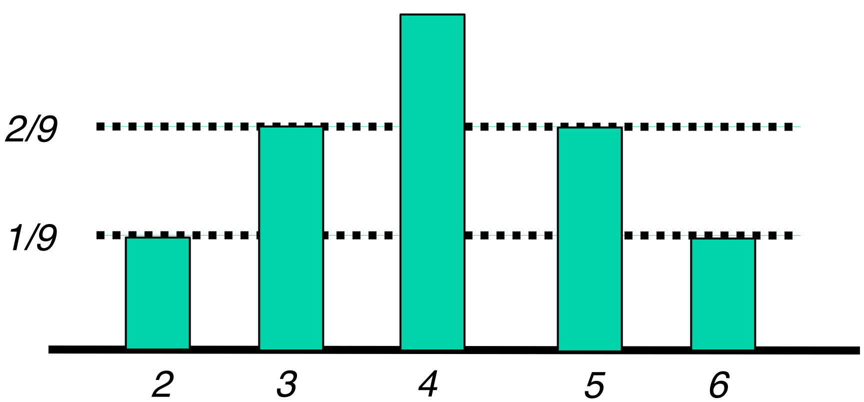


From Wikipedia

Central Limit Theorem

- **Sum of two independent copies of X**

$$\left\{\begin{array}{rcl} 1+1 & = & 2 \\ 1+2 & = & 3 \\ 1+3 & = & 4 \\ 2+1 & = & 3 \\ 2+2 & = & 4 \\ 2+3 & = & 5 \\ 3+1 & = & 4 \\ 3+2 & = & 5 \\ 3+3 & = & 6 \end{array}\right\} = \left\{\begin{array}{ll} 2 & \text{with probability } 1/9 \\ 3 & \text{with probability } 2/9 \\ 4 & \text{with probability } 3/9 \\ 5 & \text{with probability } 2/9 \\ 6 & \text{with probability } 1/9 \end{array}\right\}$$

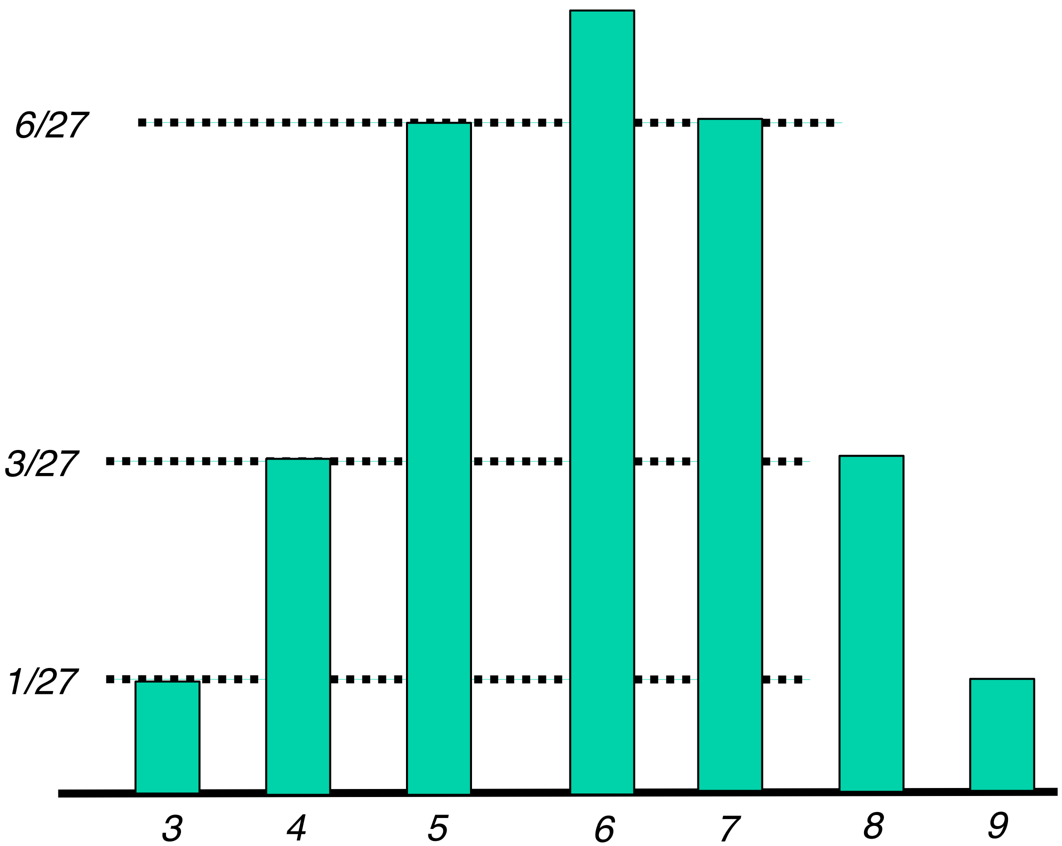


Probability Mass Function

From Wikipedia

Central Limit Theorem

- **Sum of three independent copies of X**

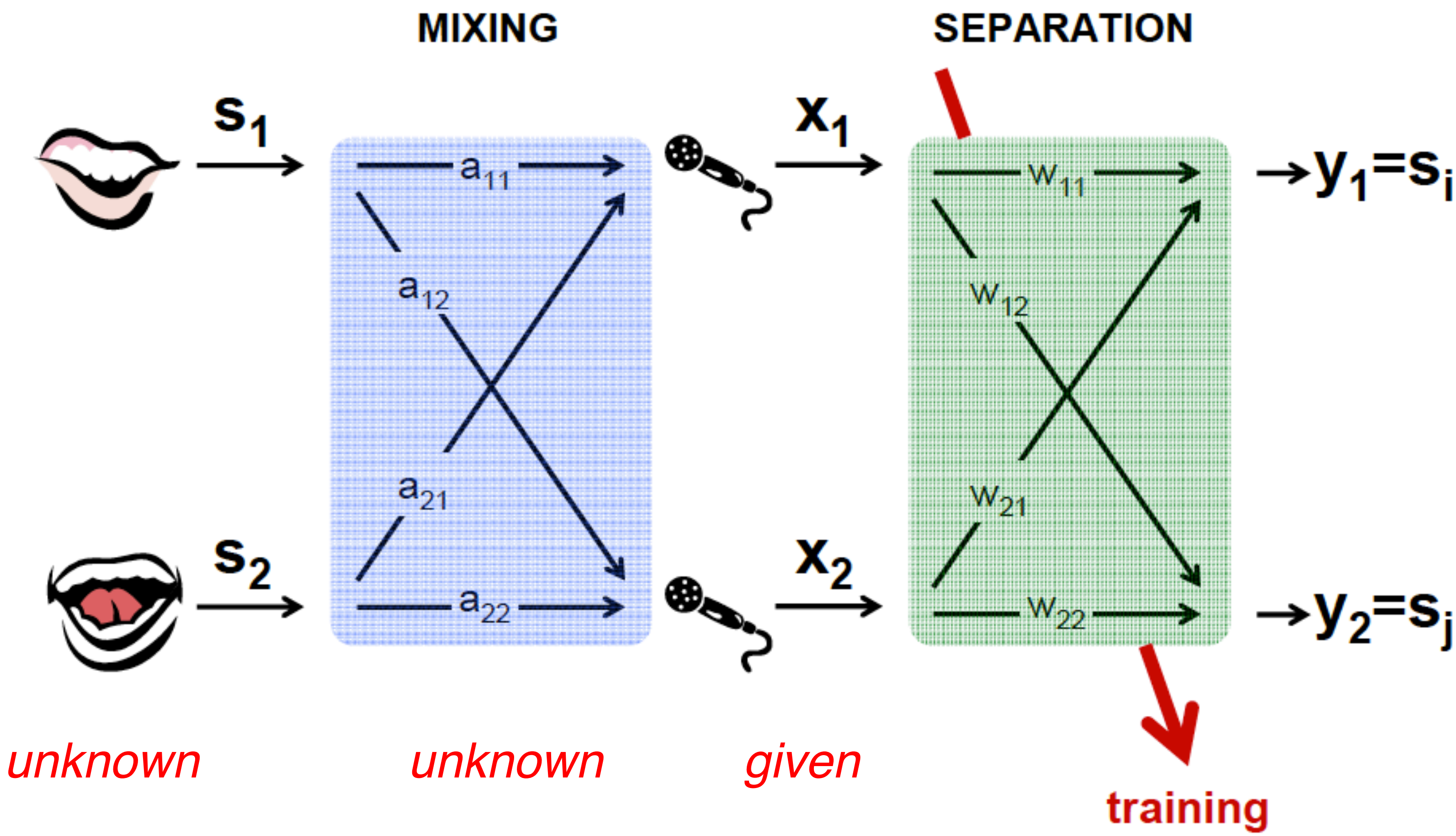


Probability Mass Function

$$\left\{\begin{array}{rcl} 1+1+1 & = & 3 \\ 1+1+2 & = & 4 \\ 1+1+3 & = & 5 \\ 1+2+1 & = & 4 \\ 1+2+2 & = & 5 \\ 1+2+3 & = & 6 \\ 1+3+1 & = & 5 \\ 1+3+2 & = & 6 \\ 1+3+3 & = & 7 \\ 2+1+1 & = & 4 \\ 2+1+2 & = & 5 \\ 2+1+3 & = & 6 \\ 2+2+1 & = & 5 \\ 2+2+2 & = & 6 \\ 2+2+3 & = & 7 \\ 2+3+1 & = & 6 \\ 2+3+2 & = & 7 \\ 2+3+3 & = & 8 \\ 3+1+1 & = & 5 \\ 3+1+2 & = & 6 \\ 3+1+3 & = & 7 \\ 3+2+1 & = & 6 \\ 3+2+2 & = & 7 \\ 3+2+3 & = & 8 \\ 3+3+1 & = & 7 \\ 3+3+2 & = & 8 \\ 3+3+3 & = & 9 \end{array}\right\} = \left\{\begin{array}{ll} 3 & \text{with probability } 1/27 \\ 4 & \text{with probability } 3/27 \\ 5 & \text{with probability } 6/27 \\ 6 & \text{with probability } 7/27 \\ 7 & \text{with probability } 6/27 \\ 8 & \text{with probability } 3/27 \\ 9 & \text{with probability } 1/27 \end{array}\right\}$$

From Wikipedia

ICA



From Gutierrez-Osuna

ICA and Central Limit Theorem

- **According to the CLT, the signal *y is more Gaussian than the sources s since it* is a linear combination of them, and becomes the least Gaussian when it is equal to one of the sources**

- **Therefore, the optimal *w is the vector that maximizes the non-Gaussianity of* $w^T x$*, since this will make y equal to one of the sources***

- **The trick is now how to measure "non-Gaussianity"…**

Gaussian Distribution

- **The Gaussian distribution, also known as the normal distribution, is a widely used model for the distribution of continuous variables**
  - One dimensional Gaussian distribution

$$P\left(x|\mu,\sigma^2\right) = \frac{1}{\left(2\pi\sigma^2\right)^{1/2}}\exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

$$\mu \rightarrow mean$$
$$\sigma^2 \rightarrow \text{var}\,iance$$

  - Multi-dimensional Gaussian distribution

$$P\left(x|\mu,\Sigma\right) = \frac{1}{\left(2\pi\right)^{D/2}|\Sigma|^{1/2}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

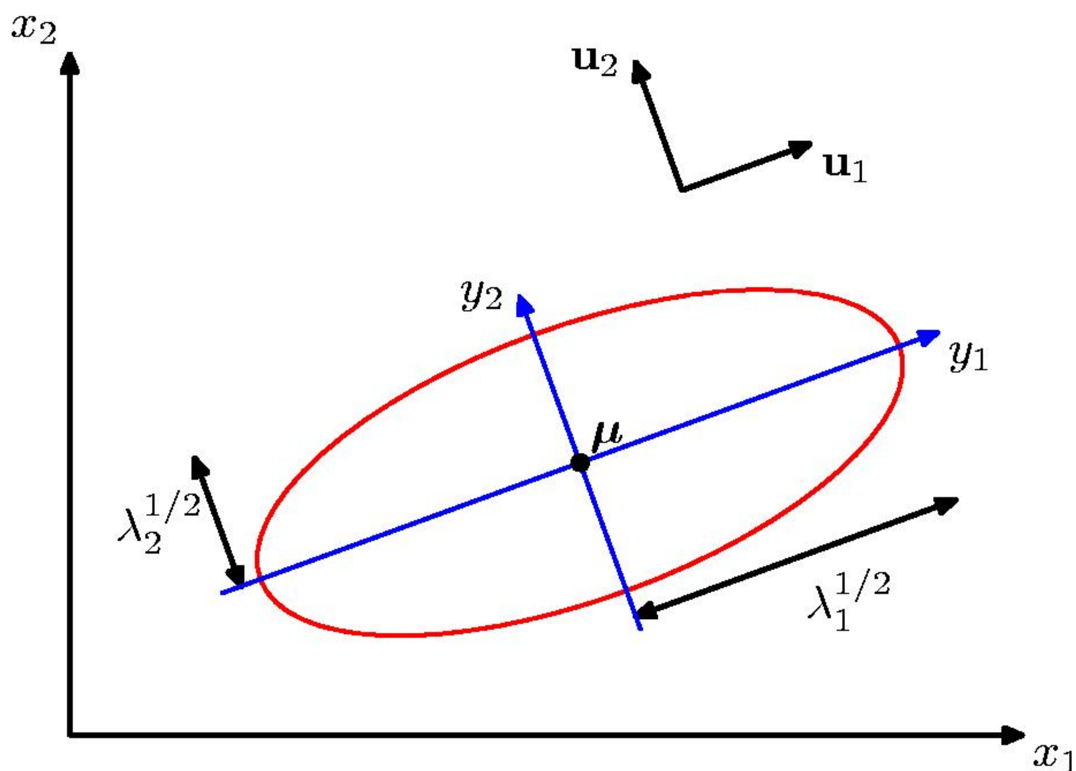
$$\mu \rightarrow mean \qquad D-\text{dim}\,ensional$$
$$\sigma^2 \rightarrow \text{var}\,iance \quad DxD \quad \text{cov}\,ariance \quad matrix$$
$$|\Sigma| \rightarrow \det er\min ant \quad \text{cov}\,ariance \quad matrix$$

Geometry of Multivariate Gaussian

- **The red curve shows elliptical surface of constant probability density for a 2-d gaussian**
  - Axis of ellipse are defined by eigenvectors (u_i) of the covariance matrix
  - Scaling factors in the directions of u_i are given by sqrt(λ_i)

$$\Delta^2 = (\mathbf{x} - \boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})$$

Mahalanobis distance



From Bishop PRML

Moments of a Gaussian Distribution

First moment (mean):

$$\overline{x} = \frac{1}{N}\sum_{j=1}^{N} x_j$$

Second moment:

$$\mathrm{Var}(x_1 \dots x_N) = \frac{1}{N-1}\sum_{j=1}^{N}(x_j - \overline{x})^2$$

Third moment:

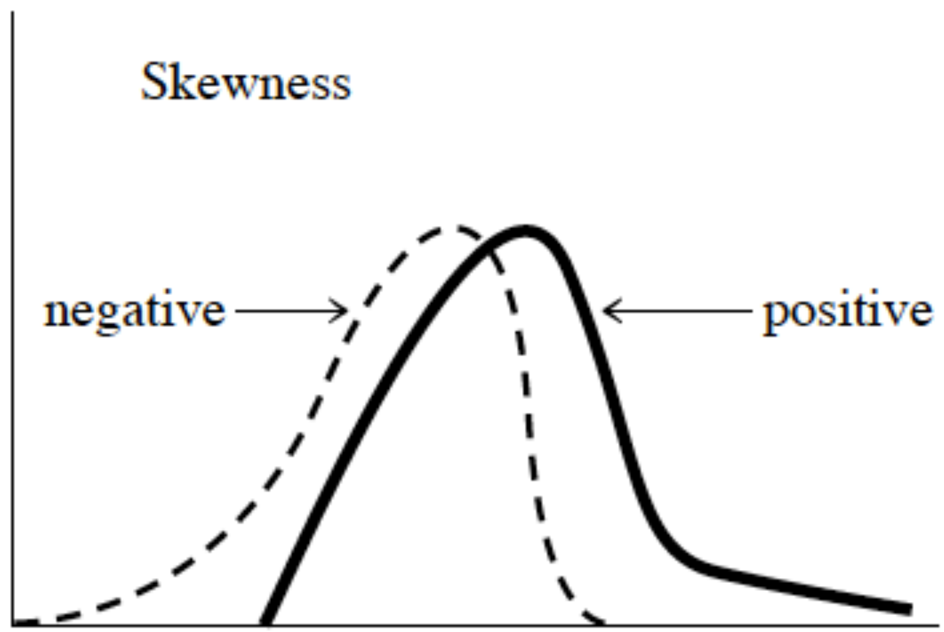
$$\mathrm{Skew}(x_1 \dots x_N) = \frac{1}{N}\sum_{j=1}^{N}\left[\frac{x_j - \overline{x}}{\sigma}\right]^3$$

Fourth moment:

$$\mathrm{Kurt}(x_1 \dots x_N) = \left\{\frac{1}{N}\sum_{j=1}^{N}\left[\frac{x_j - \overline{x}}{\sigma}\right]^4\right\} - 3$$

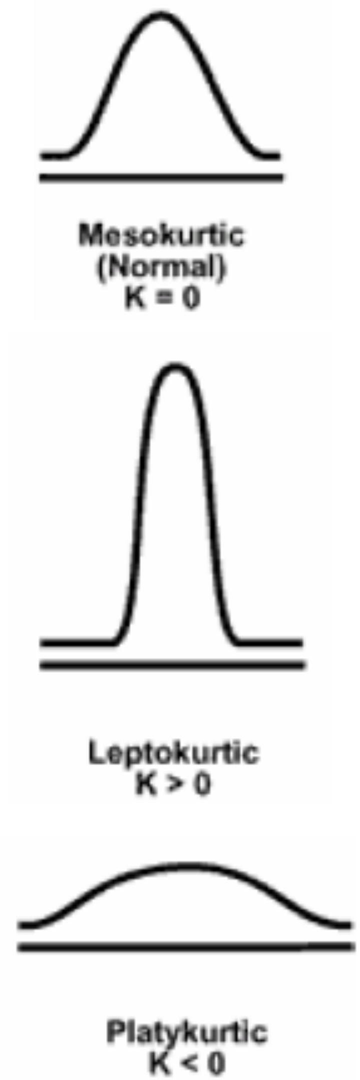
Skewness

- **Characterizes asymmetry of distribution around the mean**
  - A positively skewed distribution has a "tail" which is pulled in the positive direction.
  - A negatively skewed distribution has a "tail" which is pulled in the negative direction



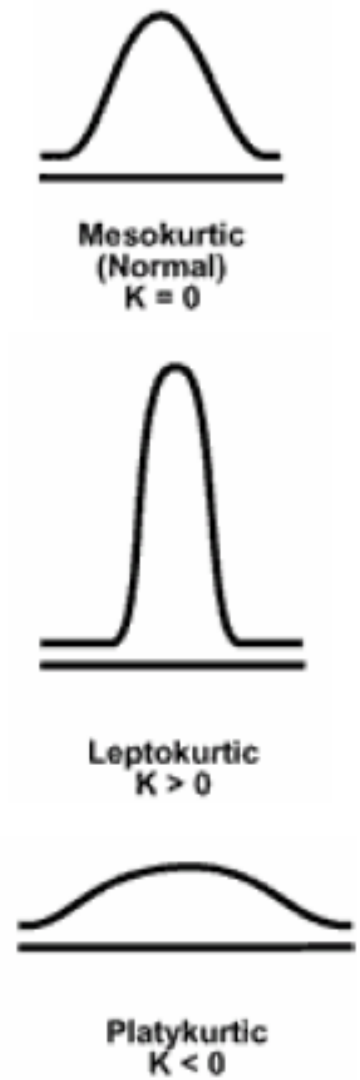
Measures of Gaussianity

- **Kurtosis**
  - Measures “peakedness” or “flatness” of a distribution relative to a normal distribution
- **Kurtosis can be both positive or negative**
  - When kurtosis is zero, the variable is Gaussian
  - When kurtusis is positive, the variable is said to be supergaussian or leptokurtic
    - Supergaussians are characterized by a “spiky” pdf with heavy tails, i.e., the Laplace pdf
  - When kurtosis is negative, the variable is said to be subgaussian or platykurtic
    - Subgaussians are characterized by a rather “flat” pdf



Measures of Gaussianity

- **Kurtosis**
  - Measures “peakedness” or “flatness” of a distribution relative to a normal distribution
- **Kurtosis can be both positive or negative**
  - When kurtosis is zero, the variable is Gaussian
  - When kurtusis is positive, the variable is said to be supergaussian or leptokurtic
    - Supergaussians are characterized by a “spiky” pdf with heavy tails, i.e., the Laplace pdf
  - When kurtosis is negative, the variable is said to be subgaussian or platykurtic
    - Subgaussians are characterized by a rather “flat” pdf
- **Thus, the absolute value of the kurtosis can be used as a measure of non-Gaussianity**
  - Kurtosis has the advantage of being computationally cheap
  - Unfortunately, kurtosis is rather sensitive to outliers



Preprocessing for ICA

- **The computation of independent components can be made simpler and better conditioned it the data is preprocessed prior to the analysis**
- **Centering**
  - This step consists of subtracting the mean of the observation vector

    $$x' = x - E[x]$$

  - The mean vector can be added to the estimates of the sources afterwards

- **Whitening**
  - Whitening consists of applying a linear transform to the observations so that its components are uncorrelated and have unit variance

    $$z = \tilde{W}x \Rightarrow E\left[zz^T\right] = I$$

  - This can be achieved through principal components

    $$z = VD^{-1/2}V^T x \qquad Note: \quad C_x^{-1/2} = VD^{-1/2}V^T$$

  - where (the columns of) V and (the diagonal of) D are the are eigenvector and eigenvalues of $E[xx^T]$, respectively

Preprocessing for ICA

Whitening

- Note that whitening makes the mixing matrix orthogonal

$$z = VD^{-1/2}V^T x$$

$$z = VD^{-1/2}V^T As = \tilde{A}s$$

$$E\left[zz^T\right] = E\left[\tilde{A}ss^T\tilde{A}\right] = \tilde{A}E\left[ss^T\right]\tilde{A}^T = \tilde{A}\tilde{A}^T = I$$

- Which has the advantage of halving the number of parameters that need to be estimated , since an orthogonal matrix only has n(n-1)/2 free parameters

FastICA algorithm for kurtosis maximization

$$J(w) = \left|kurt\left(w^T z\right)\right| = \left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|$$

$$zz^T = I$$

$$w^T w = I$$

FastICA algorithm for kurtosis maximization

$$J(w) = \left|kurt\left(w^T z\right)\right| = \left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z z^T w\right)\right]^2\right|\right)}{\partial w} \qquad \left[zz^T = I\right]$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3\left(w^T w\right)^2\right|\right)}{\partial w}$$

FastICA algorithm for kurtosis maximization

$$J(w) = \left|kurt\left(w^T z\right)\right| = \left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z z^T w\right)\right]^2\right|\right)}{\partial w} \qquad \left[zz^T = I\right]$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3\left(w^T w\right)^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|\frac{1}{N}\sum_{t=1}^{N}\left(w^T z(t)\right)^4 - 3\left(w^T w\right)^2\right|\right)}{\partial w}$$

FastICA algorithm for kurtosis maximization

$$J(w) = \left|kurt\left(w^T z\right)\right| = \left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T zz^T w\right)\right]^2\right|\right)}{\partial w} \qquad \left[zz^T = I\right]$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3\left(w^T w\right)^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|\frac{1}{N}\sum_{t=1}^{N}\left(w^T z(t)\right)^4 - 3\left(w^T w\right)^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \left|\frac{4}{N}\sum_{t=1}^{N} z(t)\left(w^T z(t)\right)^3 - 3\cdot 2\left(w^T w\right)\cdot 2w\right|$$

$$\frac{\partial J(w)}{\partial w} = 4\left|E\left[z(t)\left(w^T z(t)\right)^3\right] - 3w(w^T w)\right|$$

$$\frac{\partial J(w)}{\partial w} = 4\left|E\left[z(t)\left(w^T z(t)\right)^3\right] - 3w\right| \qquad \left[w^T w = 1\right]$$

FastICA algorithm for kurtosis maximization

$$J(w) = \left|kurt\left(w^T z\right)\right| = \left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z\right)^2\right]^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3E\left[\left(w^T z z^T w\right)\right]^2\right|\right)}{\partial w} \qquad \left[zz^T = I\right]$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|E\left[\left(w^T z\right)^4\right] - 3\left(w^T w\right)^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \frac{\partial\left(\left|\frac{1}{N}\sum_{t=1}^{N}\left(w^T z(t)\right)^4 - 3\left(w^T w\right)^2\right|\right)}{\partial w}$$

$$\frac{\partial J(w)}{\partial w} = \left|\frac{4}{N}\sum_{t=1}^{N} z(t)\left(w^T z(t)\right)^3 - 3\cdot 2\left(w^T w\right)\cdot 2w\right|$$

$$\frac{\partial J(w)}{\partial w} = 4\left|E\left[z\left(w^T z\right)^3\right] - 3w(w^T w)\right|$$

$$\frac{\partial J(w)}{\partial w} = 4\left|E\left[z\left(w^T z\right)^3\right] - 3w\right|$$

FastICA algorithm for kurtosis maximization

- **Fast ICA algorithm**

$$w_{i+1} = E\left[z\left(w_i^T z\right)^3\right] - 3w_i$$

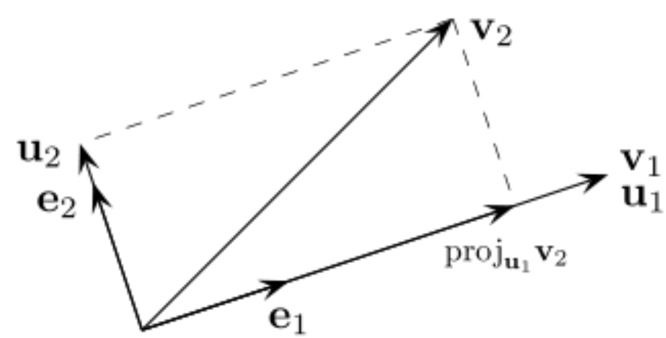
$$w_{i+1} = \frac{w_{i+1}}{norm(w_{i+1})}$$

FastICA algorithm for kurtosis maximization

- **To estimate several independent components, we run the one-unit FastICA with several units w_1, w_2, …, w_n**
  - To prevent several of these vectors from converging to the same solution, we decorrelate outputs $w_1^T x$, $w_2^T x$, …, $w_n^T x$ at each iteration
  - This can be done using a deflation scheme based on Gram-Schmidt
    - We estimate each independent component one by one
    - With p estimated components $w_1, w_2, \ldots, w_p$, we run the one-unit ICA iteration for w_{p+1}
    - After each iteration, we subtract from w_p+1 its projections $(w^T{}_p+1 w_j)w_j$ on the previous vectors w_j
    - Then, we renormalize w_{p+1}

$$w_{p+1} = w_{p+1} - \sum_{i=1}^{p} w_{p+1}^T w_j w_j$$

$$w_{p+1} = \frac{w_{p+1}}{\sqrt{w_{p+1}^T w_{p+1}}}$$



ICA Ambiguities

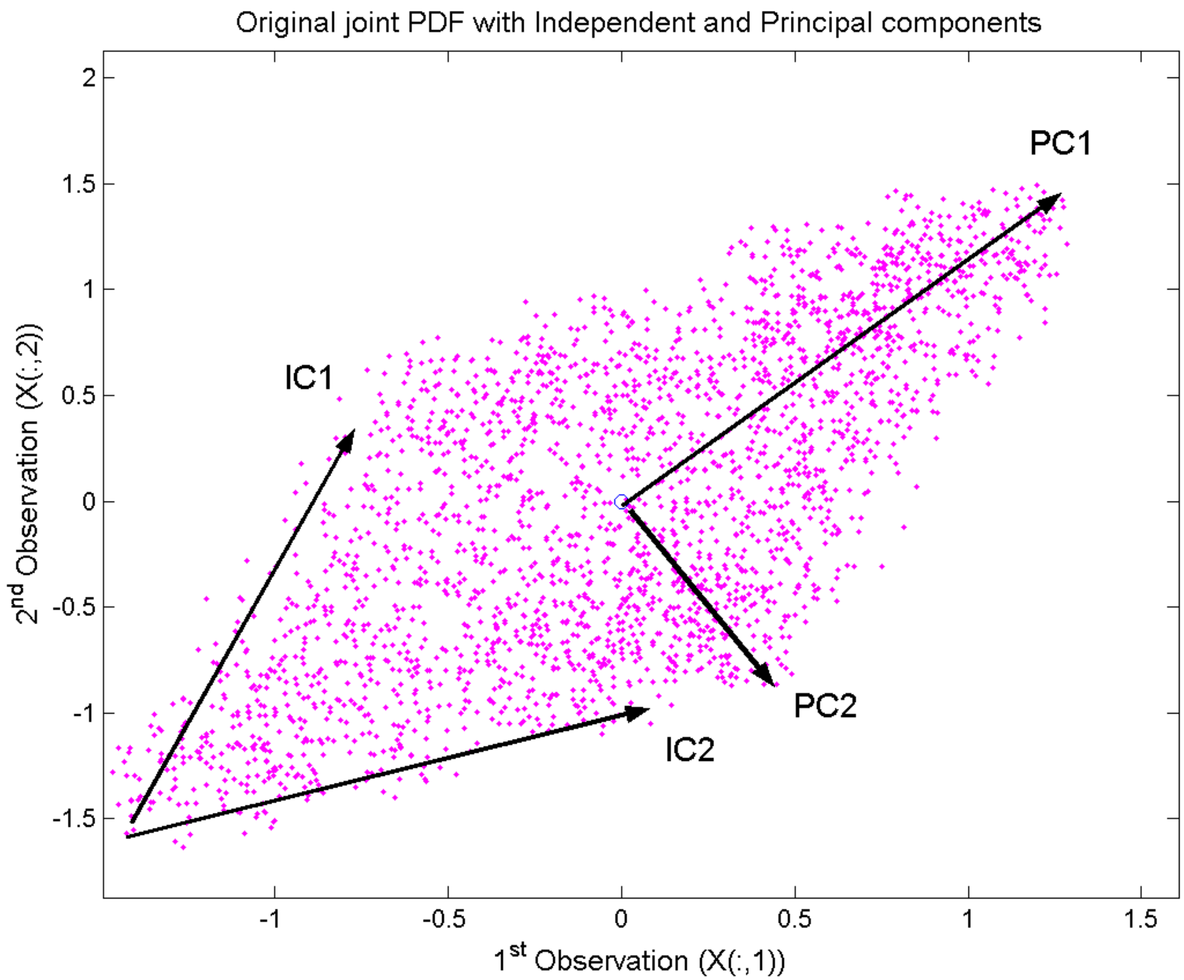
- **The variance of the independent components cannot be determined**
  - Since both s and A are undetermined, any multiplicative factor in s, including a change of sign, could be absorbed by the coefficients of A

$$x_j(t) = (ka_{j1})s_1(t) + (ka_{j2})s_2(t)$$
$$= a_{j1}(ks_1(t)) + a_{j2}(ks_2(t))$$

  - To resolve this ambiguity, source signals are assumed to have unit variance

- **The order of independent components cannot be determined**
  - Since both s and A are unknown, any permutation of the mixing terms would yield the same result
  - Compare this with Principal Components Analysis, where the order of the components can be determined by their eigenvalues (their variance)

PCA vs. ICA



Original joint PDF with Independent and Principal components
PC1
PC2
IC1
IC2
2
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
1st Observation (X(:,1))
2nd Observation (X(:,2))

Matrix Calculus

- Let **x** be a **n by 1** vector and **y** be **m by 1** vector, where each component y_i may be a function all x_j

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \qquad y = f(x)$$

Matrix Calculus

- **Derivative of the vector y with respect to vector x is n by m matrix**

$$\frac{\partial y}{\partial x} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_2}{\partial x_1} & \cdots & \frac{\partial y_m}{\partial x_1} \\ \frac{\partial y_1}{\partial x_2} & \frac{\partial y_2}{\partial x_2} & \cdots & \frac{\partial y_m}{\partial x_{21}} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial y_1}{\partial x_n} & \frac{\partial y_2}{\partial x_n} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix}$$

Matrix Calculus

- **Derivative of a scalar y with respect to vector x is n by 1 matrix**

$$\frac{\partial y}{\partial x} = \begin{bmatrix} \dfrac{\partial y}{\partial x_1} \\ \dfrac{\partial y}{\partial x_2} \\ \vdots \\ \dfrac{\partial y}{\partial x_2} \end{bmatrix}$$

Matrix Calculus

- Derivative of a vector y with respect to a scalar x is 1 by m matrix

$$\frac{\partial y}{\partial x} = \begin{bmatrix} \frac{\partial y_1}{\partial x} & \frac{\partial y_2}{\partial x} & \cdots & \frac{\partial y_m}{\partial x} \end{bmatrix}$$

Matrix Calculus

- **An Example**

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \qquad A = \begin{bmatrix} 2 & 1 \\ 1 & -2 \\ 0 & 1 \end{bmatrix}$$

$$y = Ax$$

$$y = \begin{bmatrix} 2x_1 + x_2 \\ x_1 - 2x_2 \\ x_2 \end{bmatrix}$$

Matrix Calculus

■ An Example

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \qquad A = \begin{bmatrix} 2 & 1 \\ 1 & -2 \\ 0 & 1 \end{bmatrix}$$

$$y = A\ x$$

$$y = \begin{bmatrix} 2x_1 + x_2 \\ x_1 - 2x_2 \\ x_2 \end{bmatrix}$$

$$\frac{\partial y}{\partial x} = \begin{bmatrix} 2 & 1 & 0 \\ 1 & -2 & 1 \end{bmatrix}$$

Matrix Calculus

$\mathbf{y}$	$\dfrac{\partial \mathbf{y}}{\partial \mathbf{x}}$
$\mathbf{A}\mathbf{x}$	$\mathbf{A}^T$
$\mathbf{x}^T\mathbf{A}$	$\mathbf{A}$
$\mathbf{x}^T\mathbf{x}$	$2\mathbf{x}$
$\mathbf{x}^T\mathbf{A}\mathbf{x}$	$\mathbf{A}\mathbf{x} + \mathbf{A}^T\mathbf{x}$

Note: A is a matrix

LDA worked example

- **Compute the Linear Discriminant projection for the following two-dimensional dataset**
  - X1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)}
  - X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}

$$S_i = \frac{1}{N_i} \sum_{x \in Ci} (x - \mu_i)(x - \mu_i)^T$$

$$S_W = S_1 + S_2$$

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

From Gutierrez-Osuna

LDA worked example

- **Compute the Linear Discriminant projection for the following two-dimensional dataset**
  - X1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)}
  - X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}

$$S_i = \frac{1}{N_i} \sum_{x \in Ci} (x - \mu_i)(x - \mu_i)^T$$

$$S_W = S_1 + S_2$$

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

- **Class Means:**

$$\mu_1 = \begin{bmatrix} 3.00 & 3.60 \end{bmatrix}$$

$$\mu_2 = \begin{bmatrix} 8.40 & 7.60 \end{bmatrix}$$

LDA worked example

- **Compute the Linear Discriminant projection for the following two-dimensional dataset**
  - X1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)}
  - X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}

$$S_i = \frac{1}{N_i} \sum_{x \in Ci} (x - \mu_i)(x - \mu_i)^T$$

$$S_W = S_1 + S_2$$

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

- **Class Means:**

$$\mu_1 = \begin{bmatrix} 3.00 & 3.60 \end{bmatrix}^T$$

$$\mu_2 = \begin{bmatrix} 8.40 & 7.60 \end{bmatrix}^T$$

- **Within Class Scatter (class1):**

$$x_{C1} - \mu_1 = \begin{bmatrix} 1 & -2.6 \\ -1 & 0.4 \\ -1 & -0.6 \\ 0 & 2.4 \\ 1 & 0.4 \end{bmatrix}^T$$

$$S_1 = \frac{1}{5} \sum_{x \in C1} (x - \mu_1)(x - \mu_1)^T$$

$$S_1 = \begin{bmatrix} 0.8 & -0.4 \\ -0.4 & 2.64 \end{bmatrix}$$

LDA worked example

- **Compute the Linear Discriminant projection for the following two-dimensional dataset**
  - X1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)}
  - X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}

$$S_i = \frac{1}{N_i} \sum_{x \in Ci} (x - \mu_i)(x - \mu_i)^T$$

$$S_W = S_1 + S_2$$

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

- **Class Means:**

$$\mu_1 = \begin{bmatrix} 3.00 & 3.60 \end{bmatrix}^T$$

$$\mu_2 = \begin{bmatrix} 8.40 & 7.60 \end{bmatrix}^T$$

- **Within Class Scatter (class2):**

$$x_{C1} - \mu_1 = \begin{bmatrix} 0.6 & 2.4 \\ -2.4 & 0.4 \\ 0.6 & -2.6 \\ -0.4 & -0.6 \\ 1.6 & 0.4 \end{bmatrix}^T$$

$$S_2 = \frac{1}{5} \sum_{x \in C1} (x - \mu_1)(x - \mu_1)^T$$

$$S_2 = \begin{bmatrix} 1.84 & -0.04 \\ -0.04 & 2.64 \end{bmatrix}$$

LDA worked example

- **Compute the Linear Discriminant projection for the following two-dimensional dataset**
  - X1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)}
  - X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}

$$S_i = \frac{1}{N_i} \sum_{x \in C_i} (x - \mu_i)(x - \mu_i)^T$$

$$S_W = S_1 + S_2$$

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

- **Class Means:**

$$\mu_1 = \begin{bmatrix} 3.00 & 3.60 \end{bmatrix}^T$$

$$\mu_2 = \begin{bmatrix} 8.40 & 7.60 \end{bmatrix}^T$$

- **Total Within Class Scatter :**

$$S_1 = \begin{bmatrix} 0.8 & -0.4 \\ -0.4 & 2.6 \end{bmatrix}$$

$$S_w = \begin{bmatrix} 2.64 & -0.44 \\ -0.44 & 5.28 \end{bmatrix}$$

$$S_2 = \begin{bmatrix} 1.84 & -0.04 \\ -0.04 & 2.64 \end{bmatrix}$$

LDA worked example

- **Compute the Linear Discriminant projection for the following two-dimensional dataset**
  - X1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)}
  - X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}

$$S_i = \frac{1}{N_i} \sum_{x \in Ci} (x - \mu_i)(x - \mu_i)^T$$

$$S_W = S_1 + S_2$$

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

- **Class Means:**

$$\mu_1 = \begin{bmatrix} 3.00 & 3.60 \end{bmatrix}^T$$

$$\mu_2 = \begin{bmatrix} 8.40 & 7.60 \end{bmatrix}^T$$

- **Total Within Class Scatter :**

$$S_w = \begin{bmatrix} 2.64 & -0.44 \\ -0.44 & 5.28 \end{bmatrix}$$

- **Between Class Scatter:**

$$S_B = \begin{pmatrix} -5.4 \\ -4 \end{pmatrix} \begin{pmatrix} -5.4 & -4 \end{pmatrix}$$

$$S_w = \begin{bmatrix} 29.16 & 21.6 \\ 21.6 & 16 \end{bmatrix}$$

LDA worked example

- **Compute the Linear Discriminant projection for the following two-dimensional dataset**
  - X1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)}
  - X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}

$$S_i = \frac{1}{N_i} \sum_{x \in Ci} (x - \mu_i)(x - \mu_i)^T$$

$$S_W = S_1 + S_2$$

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

- **Class Means:**

$$\mu_1 = \begin{bmatrix} 3.00 & 3.60 \end{bmatrix}^T$$

$$\mu_2 = \begin{bmatrix} 8.40 & 7.60 \end{bmatrix}^T$$

- **Scatter Matrices :**

$$S_w = \begin{bmatrix} 2.64 & -0.44 \\ -0.44 & 5.28 \end{bmatrix} \qquad S_B = \begin{bmatrix} 29.16 & 21.6 \\ 21.6 & 16 \end{bmatrix}$$

- **The LDA projection is then obtained as the solution of the generalized eigenvalue problem:**

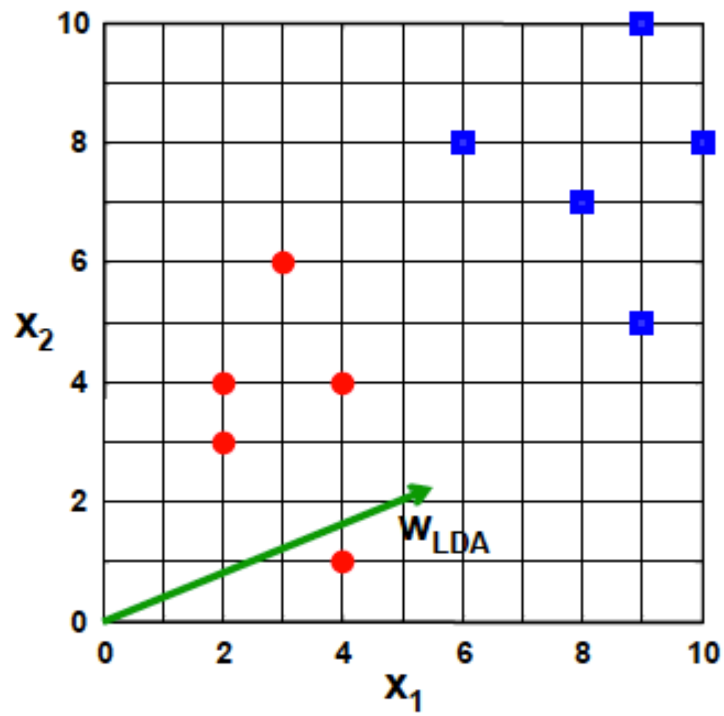
$$S_w^{-1} S_B w = \lambda w \Rightarrow \left| S_w^{-1} S_B - \lambda I \right| = 0 \Rightarrow \begin{vmatrix} 11.89 - \lambda & 8.81 \\ 5.08 & 3.76 - \lambda \end{vmatrix} = 0 \Rightarrow \lambda = 15.65$$

$$\begin{vmatrix} 11.89 & 8.81 \\ 5.08 & 3.76 \end{vmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = 15.65 \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 0.91 \\ 0.39 \end{bmatrix}$$

LDA worked example

- **Compute the Linear Discriminant projection for the following two-dimensional dataset**
  - X1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)}
  - X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}



- **Class Means:**

$$\mu_1 = \begin{bmatrix} 3.00 & 3.60 \end{bmatrix}^T$$

$$\mu_2 = \begin{bmatrix} 8.40 & 7.60 \end{bmatrix}^T$$

- **Scatter Matrices :**

$$S_w = \begin{bmatrix} 2.64 & -0.44 \\ -0.44 & 5.28 \end{bmatrix} \qquad S_B = \begin{bmatrix} 29.16 & 21.6 \\ 21.6 & 16 \end{bmatrix}$$

- **Eigenvectors of $S_w^{-1}S_B$:**

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 0.91 \\ 0.39 \end{bmatrix}$$

From Gutierrez-Osuna