

# Machine Learning

# Contents

1. Introduction
2. K-Nearest Neighbor Algorithm
3. LDA(Linear Discriminant Analysis)
4. Perceptron
5. Feed-Forward Neural Networks
6. RNN(Recurrent Neural Networks)
7. SVM(Support Vector Machine)
8. Ensemble Learning
9. CNN(Convolutional Neural Network)
10. PCA(Principal Component Analysis)
- 11. ICA(Independent Component Analysis)**
12. Clustering
13. GAN(Generative Adversarial Network)

# 11.1. Problem Statement

- Cocktail Party Problem



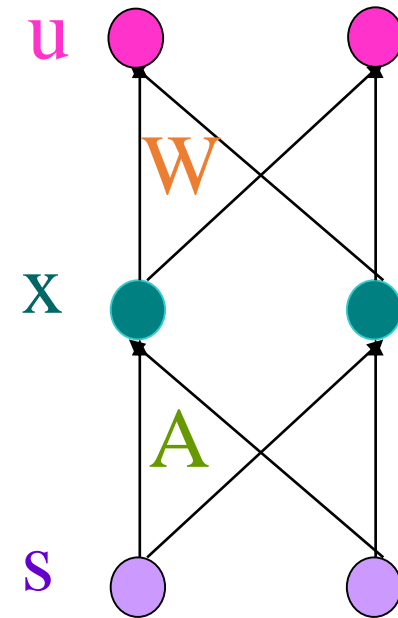
- Unknown mutually independent source signals with zero means

$$s_i(t), i = 1, \dots, n$$

- The model for sensor signals

$$\mathbf{x}(t) = \mathbf{A} \cdot \mathbf{s}(t)$$

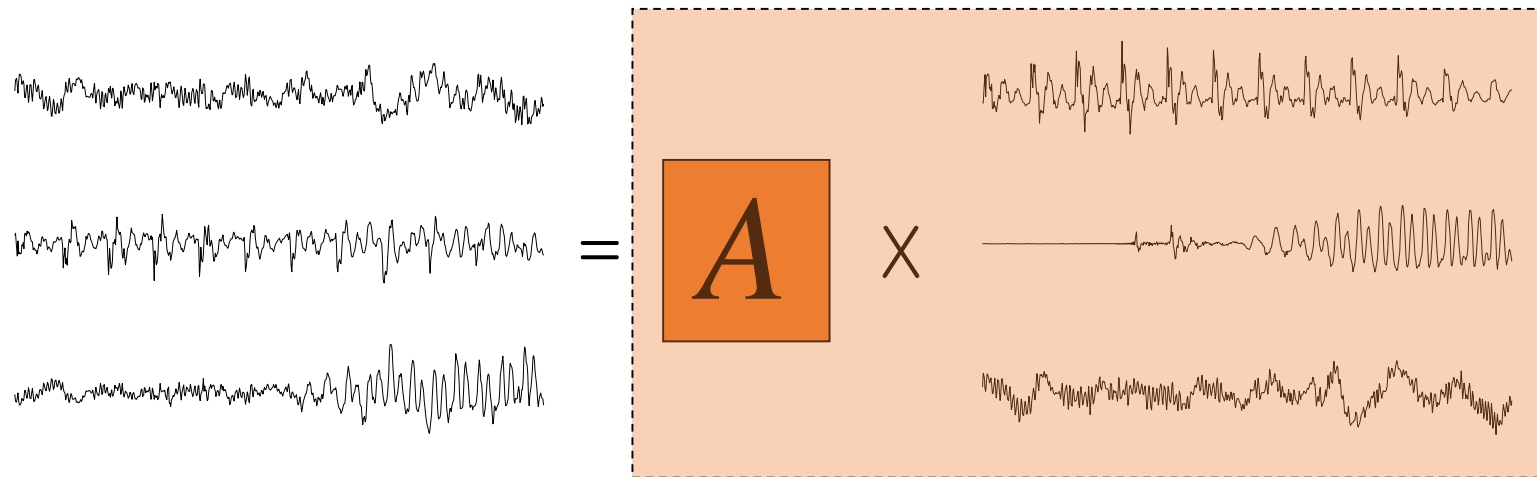
- Problem
  - To recover the original signals except for a permutation of indices and scales
  - Estimation of an unmixing matrix



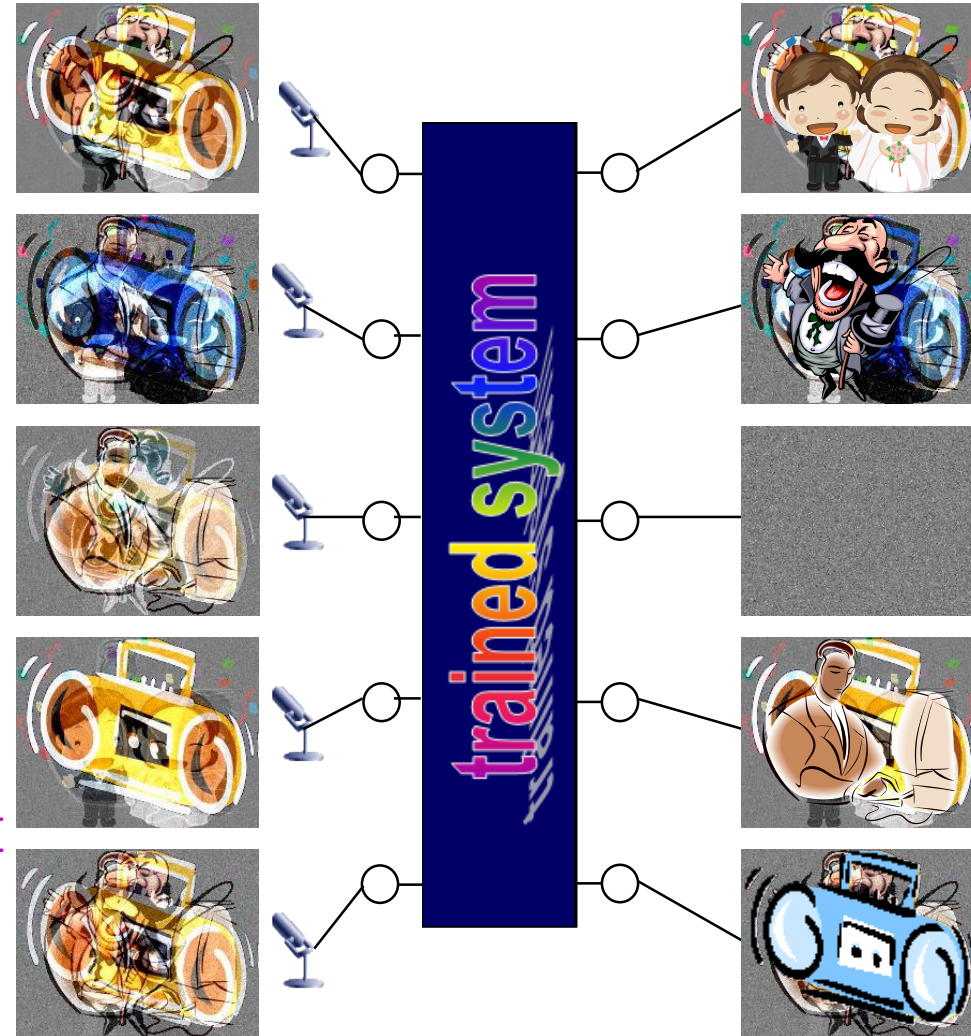
$$x_1(t) = a_{11}s_1(t) + a_{12}s_2(t) + a_{13}s_3(t)$$

$$x_2(t) = a_{21}s_1(t) + a_{22}s_2(t) + a_{23}s_3(t)$$

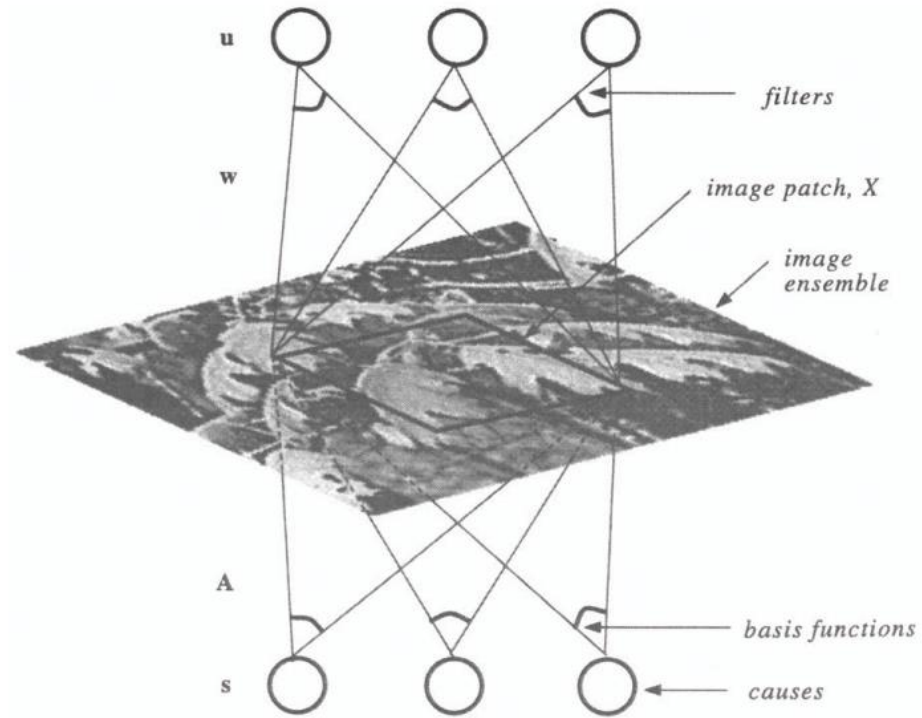
$$x_3(t) = a_{31}s_1(t) + a_{32}s_2(t) + a_{33}s_3(t),$$



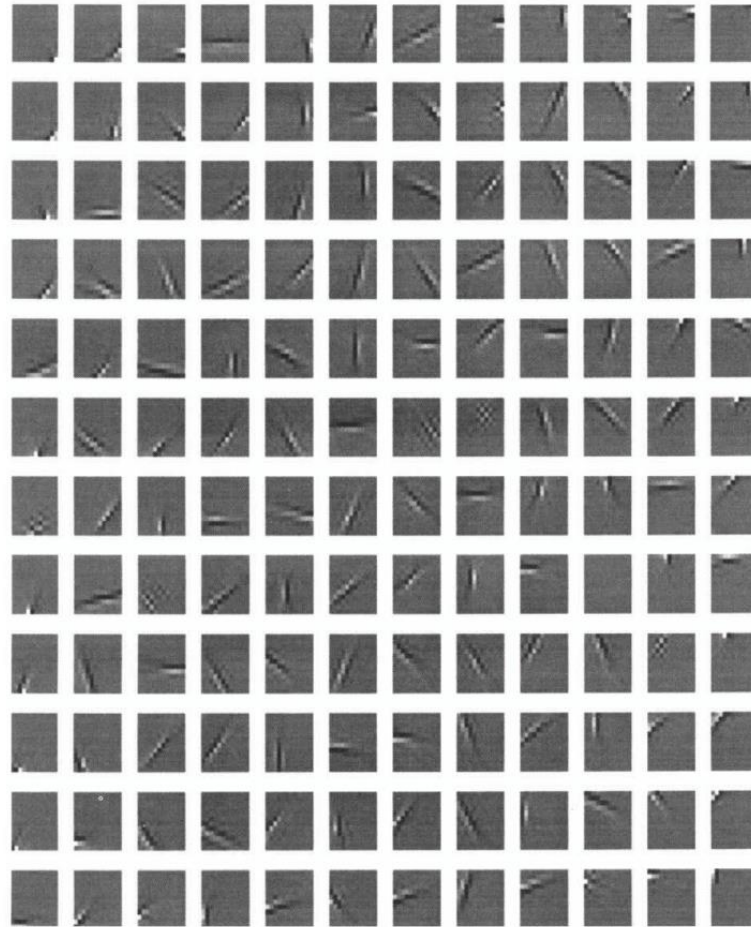
# Blind Source Separation by ICA



- Independent Components of Natural Scenes  
(Bell and Sejnowski, Vision Research 1996.[2])



- Independent Components are Edge Filters





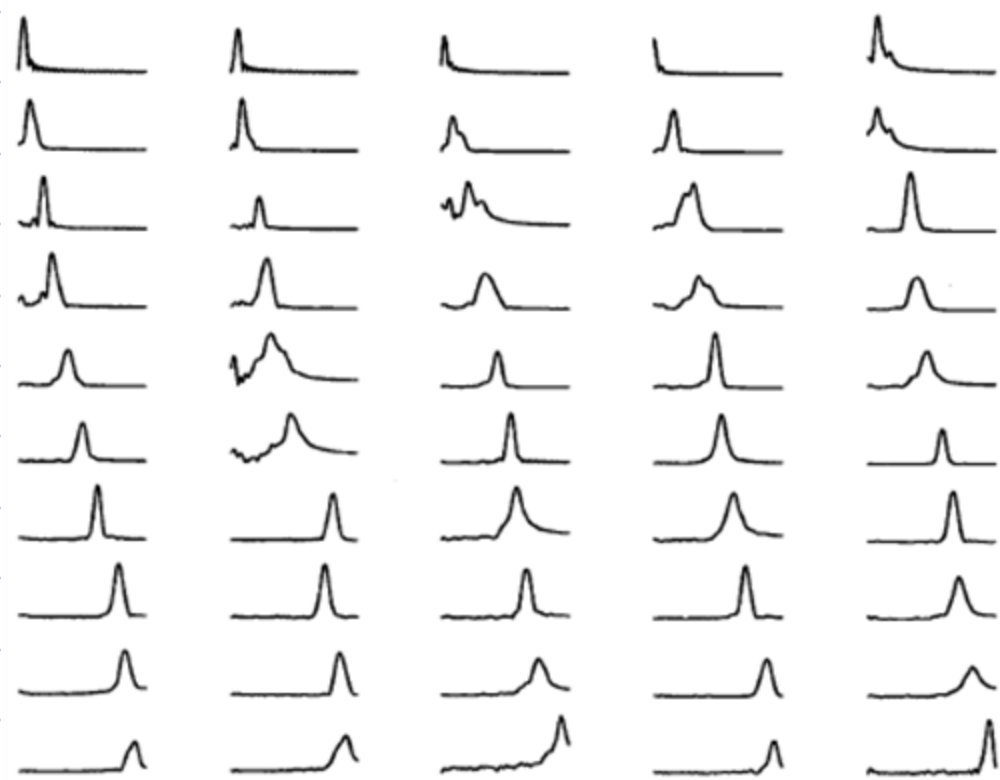


그림 11.3. 음성에서 추출한 독립성분

[41. Jong-Hwan Lee et al. 2002]

# Assumptions and limitations on ICA

- Basic assumptions:
  - Individual source signals are statistically independent of the other source signals.
  - # of sources  $\leq$  # of observations
    - $\mathbf{x} = \mathbf{A}\mathbf{s}$      $\mathbf{s} = \mathbf{W}\mathbf{x}$
- Limitations:
  - Scale and sign indeterminacy
    - We cannot determine the variance of sources
    - $\mathbf{x} = \mathbf{A}\mathbf{s} = (\mathbf{A}/\alpha)(\alpha\mathbf{s}) = \mathbf{A}'\mathbf{s}'$
  - Permutation indeterminacy
    - We cannot determine the order of sources
  - Sources should be non-Gaussian. At most one source can be Gaussian.

## 11.2. Information Theory

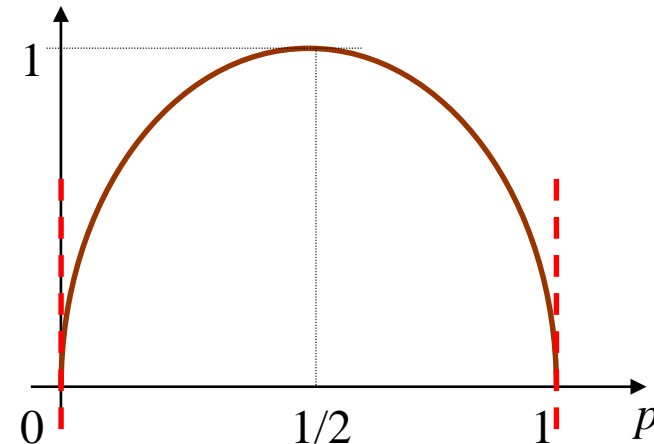
- What is Entropy?
- Definition: The average amount of information of the source.

$$H_r(S) \equiv \sum_{i=1}^q p_i I_r(p_i) = \sum_{i=1}^q p_i \log_r \frac{1}{p_i}$$

$$H_2(S) \equiv \sum_{i=1}^q p_i I_2(p_i) = \sum_{i=1}^q p_i \log_2 \frac{1}{p_i} \text{ (bits)}$$

- Ex) Two Symbol Sources

$$H_2(p) \equiv p \log_2 \frac{1}{p} + (1-p) \log_2 \frac{1}{1-p} \text{ (bits)}$$



- Differential Entropy  $H(X) \equiv -\int p(x) \log p(x) dx$

- Bounded Case: Uniform Distribution
- Unbounded Case: Normal Distribution (Gaussian)

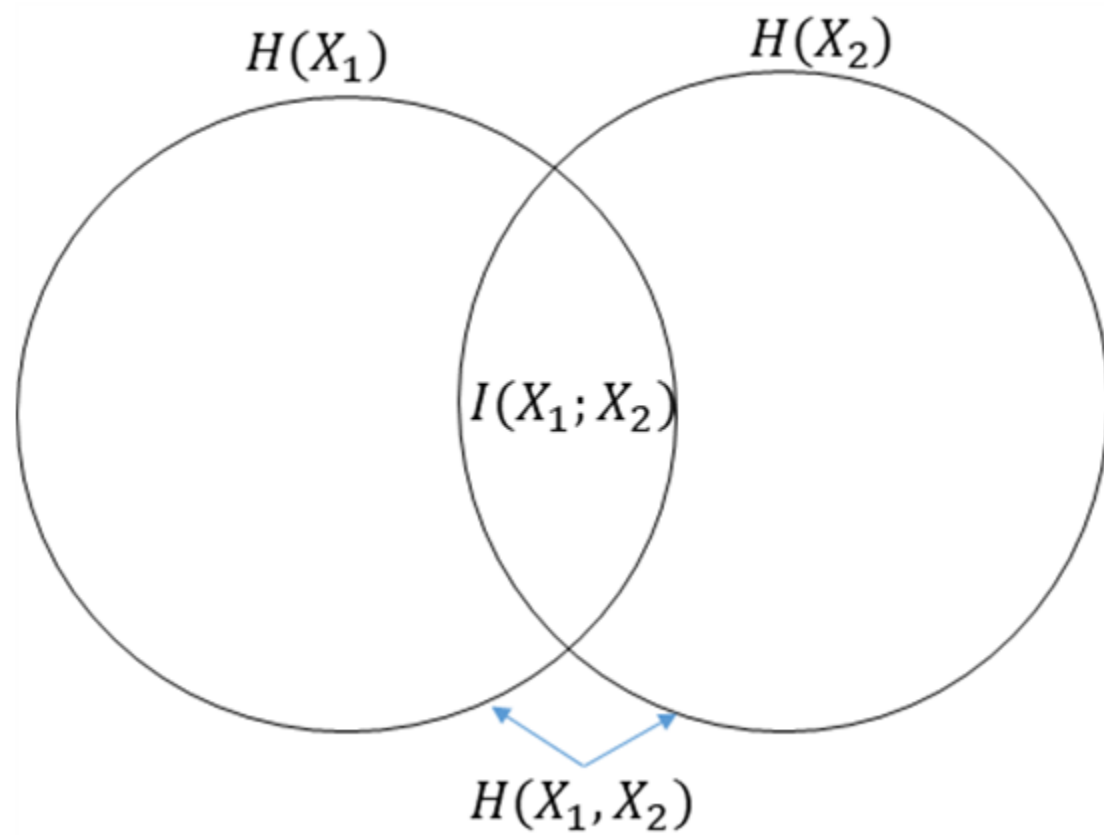
- Joint Entropy

$$H(X_1, X_2) \equiv -\int p(x_1, x_2) \log p(x_1, x_2) dx_1 dx_2$$

- Mutual Information

$$I(X_1; X_2) \equiv \int p(x_1, x_2) \log \frac{p(x_1, x_2)}{p(x_1)p(x_2)} dx_1 dx_2$$

$$H(X_1, X_2) = H(X_1) + H(X_2) - I(X_1; X_2)$$



참조: 상호정보와 엔트로피의 관계 유도

상호정보와 엔트로피의 관계를 유도해보자. 식 (11.2.6)으로 주어진 상호정보는

$$I(X_1; X_2) = \int p(x_1, x_2) \log p(x_1, x_2) dx_1 dx_2 - \int p(x_1, x_2) \log [p(x_1)p(x_2)] dx_1 dx_2 \quad (11.2.8)$$

와 같이 두개의 항으로 나누어진다. 두 번째 항은

$$\int p(x_1, x_2) \log [p(x_1)p(x_2)] dx_1 dx_2 = \int p(x_1) \log p(x_1) dx_1 + \int p(x_2) \log p(x_2) dx_2 \quad (11.2.9)$$

로 정리된다. 이때,  $p(x_1) = \int p(x_1, x_2) dx_2$ 을 적용하였다. 식 (11.2.9)를 (11.2.8)에 대입하고 엔트로피의 정의를 적용하면

$$I(X_1; X_2) = -H(X_1, X_2) + H(X_1) + H(X_2) \quad (11.2.10)$$

를 얻게 된다.

# 11.3. Probability

## Probability Density Function

$p_x(x)$  : probability density function of  $x$

$p_y(y)$  : probability density function of  $y = g(x)$

Assume  $g(x)$  is a monotonic increasing function, then

$$p_y(y)dy = \Pr\{y < \mathbf{y} < y + dy\}$$

This set consists of the interval  $x < \mathbf{x} < x + dx$

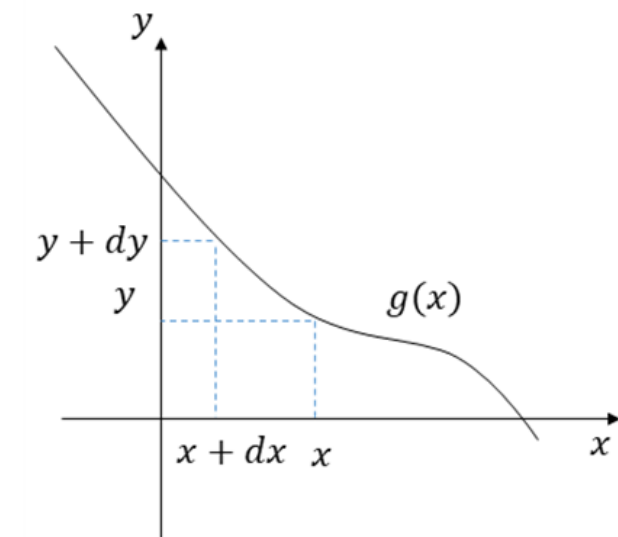
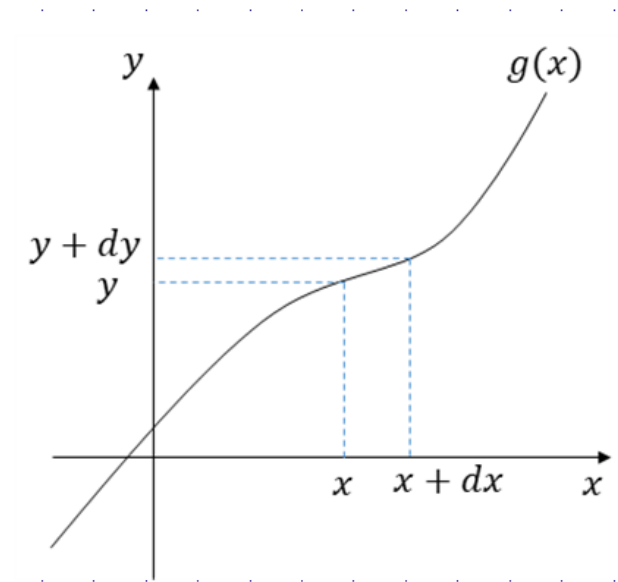
(decreasing :  $x + dx < \mathbf{x} < x$ ,  $dx < 0$ )

$$p_y(y)dy = \Pr\{y < \mathbf{y} < y + dy\} = \Pr\{x < \mathbf{x} < x + dx\} = p_x(x)dx$$

(decreasing :  $p_y(y)dy = \Pr\{x + dx < \mathbf{x} < x\} = p_x(x)|dx|$ )

Therefore, for monotonic function of  $y = g(x)$

$$p_y(y) = \frac{p_x(x)}{\left| \frac{dy}{dx} \right|} = \frac{p_x(x)}{|g'(x)|}$$



$$F_u(u) = \Pr\{U \leq u\}$$

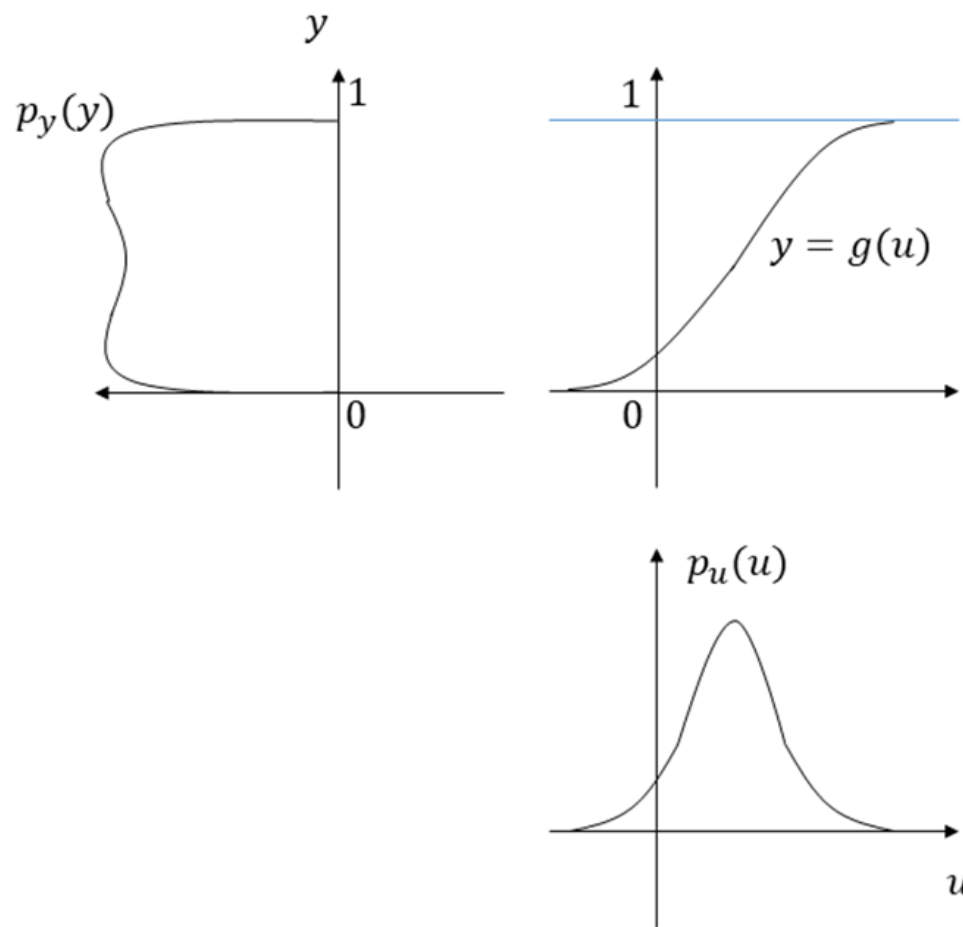
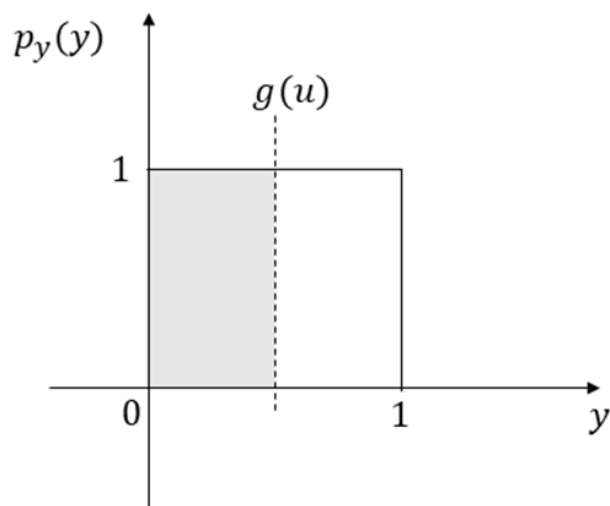


그림 11.11. 확률변수의 CDF 함수 효과

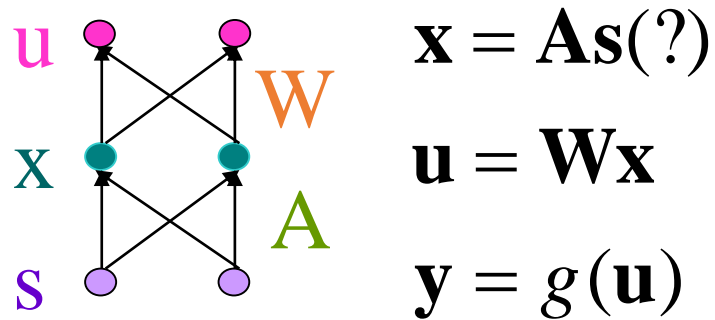
$$F_u(u) = \Pr\{U \leq u\} = \Pr\{g^{-1}(Y) \leq u\} = \Pr\{Y \leq g(u)\} = g(u)$$



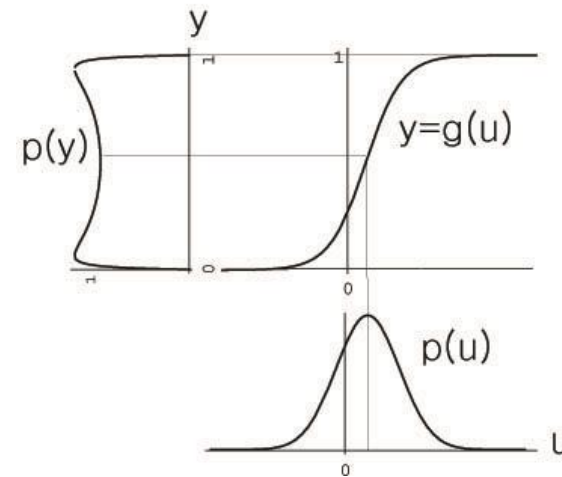
# 11.4. InfoMax Algorithm

## Objective Function of ICA

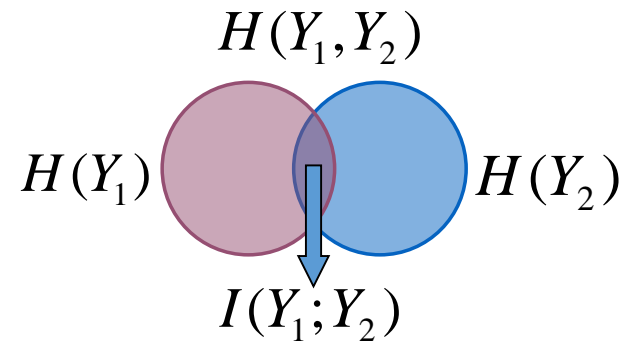
- Objective Function of ICA



$g(\cdot)$ : c.d.f. of source distribution



- Minimizing Mutual Information
- Maximizing Entropy  $H(\mathbf{y})$



# Maximizing Output Entropy: InfoMax

- Joint entropy at the outputs

$$H(y_1, \dots, y_N) = H(y_1) + \dots + H(y_N) - I(y_1, \dots, y_N)$$

where  $H(y_i) = -E\{\log p(y_i)\}$

- Pdf of the outputs

$$p(y) = \frac{p(x)}{|J(x)|}$$

- Joint entropy at the outputs

$$H(y) = -E\{\log p(y)\} = E\{\log |J(x)|\} - E\{\log p(x)\}$$

# One Input One Output

- Stochastic gradient ascent learning rule

$$u = wx + w_0, \quad y = \frac{1}{1+e^{-u}} \quad p_y(y) = \frac{p_x(x)}{\left| \frac{\partial y}{\partial x} \right|}$$

$$H(y) = -E[\log p_y(y)] = -\int p_y(y) \log p_y(y) dy = E[\log \left| \frac{\partial y}{\partial x} \right|] - E[\log p_x(x)]$$

$$\Delta w \propto \frac{\partial H(y)}{\partial w} = \frac{\partial}{\partial w} \log \left| \frac{\partial y}{\partial x} \right| = \left| \frac{\partial y}{\partial x} \right|^{-1} \frac{\partial}{\partial w} \left( \frac{\partial y}{\partial x} \right)$$

$$\frac{\partial y}{\partial x} = \frac{\partial y}{\partial u} \frac{\partial u}{\partial x} = wy(1-y)$$

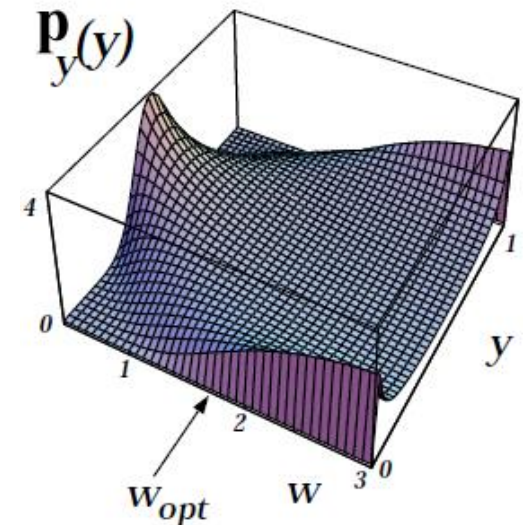
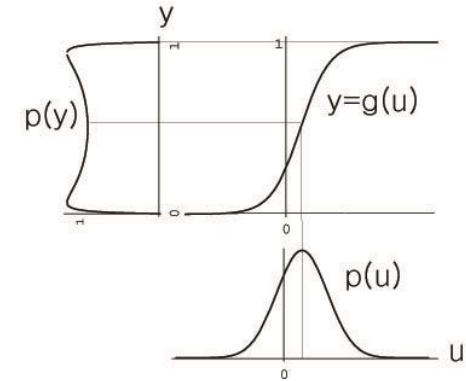
$$\frac{\partial}{\partial w} \frac{\partial y}{\partial x} = y(1-y)(1+wx(1-2y))$$

$$\Delta w \propto \frac{1}{w} + x(1-2y) \quad \text{Similarly} \quad \Delta w_0 \propto (1-2y)$$

$\Delta w_0$ : centre the steepest part of the sigmoid curve on the peak of  $p_x(x)$

$\Delta w$ : scale the slope of the sigmoid to match the variance of  $p_x(x)$

$p_y(y)$ : flat unit distribution on



## 11.4. InfoMax Algorithm: $N \rightarrow N$ Network

- Stochastic gradient ascent learning rule

$$\Delta \mathbf{W} \propto \frac{\partial H(\mathbf{y})}{\partial \mathbf{W}} = \frac{\partial}{\partial \mathbf{W}} \log |\mathbf{J}|$$

- Jacobian of the transform

$$\mathbf{J}(\mathbf{x}) = \det \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \dots & \frac{\partial y_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_n}{\partial x_1} & \dots & \frac{\partial y_n}{\partial x_n} \end{bmatrix} = \det(\mathbf{W}) \prod_{i=1}^N \frac{\partial y_i}{\partial u_i} \quad \frac{\partial \mathbf{y}}{\partial \mathbf{x}} = \frac{\partial \mathbf{y}}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \mathbf{x}}$$

- Learning rule

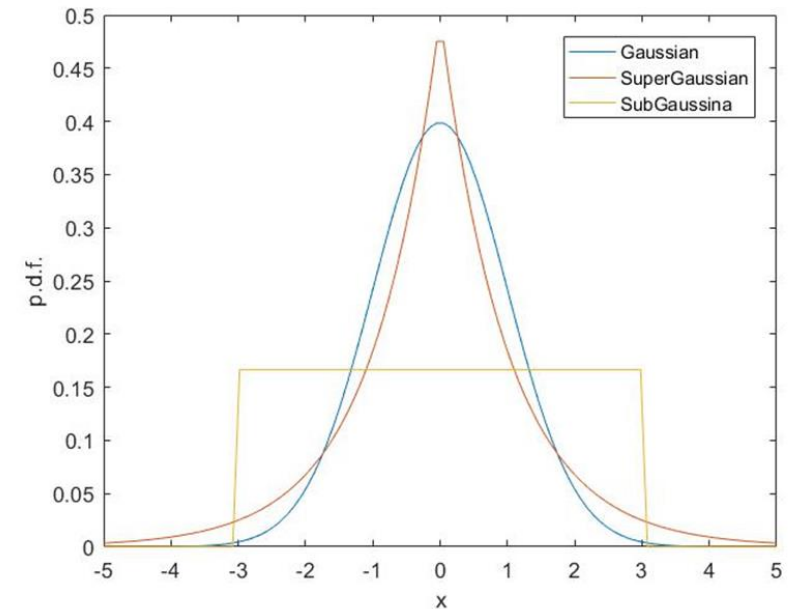
$$\begin{aligned}\Delta \mathbf{W} &\propto \frac{\partial H(\mathbf{y})}{\partial \mathbf{W}} = \frac{\partial}{\partial \mathbf{W}} \log |\det(\mathbf{W})| + \sum_{i=1}^N \frac{\partial}{\partial \mathbf{W}} \log \left| \frac{\partial y_i}{\partial u_i} \right| \\ &= (\mathbf{W}^T)^{-1} + \frac{\frac{\partial p(\mathbf{u})}{\partial \mathbf{u}}}{p(\mathbf{u})} \mathbf{x}^T \\ &= (\mathbf{W}^T)^{-1} - \varphi(\mathbf{u}) \mathbf{x}^T\end{aligned}$$

where

$$p(u_i) = \frac{\partial y_i}{\partial u_i} \quad \text{and} \quad \varphi(u_i) = -\frac{\frac{\partial p(u_i)}{\partial u_i}}{p(u_i)} \quad (\text{Score function})$$

- Natural Gradient (or Relative Gradient)  
(Amari, Neural Computation, 1998. [31])

$$\begin{aligned} \Delta \mathbf{W} &\propto \frac{\partial H(\mathbf{y})}{\partial \mathbf{W}} \mathbf{W}^T \mathbf{W} = [(\mathbf{W}^T)^{-1} - \varphi(\mathbf{u}) \mathbf{x}^T] \mathbf{W}^T \mathbf{W} \\ &= [\mathbf{I} - \varphi(\mathbf{u}) \mathbf{u}^T] \mathbf{W} \end{aligned}$$

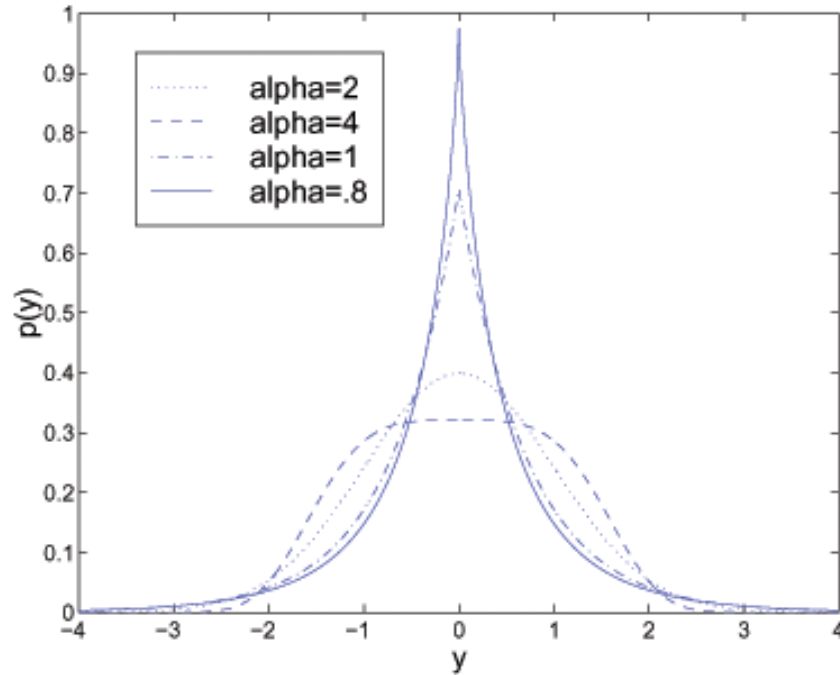


-How can we decide the score ft.:Ext. ICA (T.-W. Lee et al., 1999.[32])

$$\varphi(u_i) = u_i + k_i \tanh(u_i)$$

$$\text{where } k_i = \text{sign}(E\{\text{sech}^2(u_i)\}E\{u_i^2\} - E\{[\tanh(u_i)]u_i\})$$

# Flex. ICA: Generalized Gaussian



$\alpha = 2$  : Normal

(S. Choi. 2000.[34])

$$p(u; \alpha) = \frac{\alpha}{2\lambda\Gamma(\frac{1}{\alpha})} \exp\left\{-\left|\frac{u}{\lambda}\right|^\alpha\right\}, \quad \varphi(u_i) = |u_i|^{\alpha_i-1} \text{sgn}(u_i)$$

참조: 행렬의 미분

$m \times n$ 차원 행렬  $\mathbf{W} = (w_{ij})$ 에 대한 스칼라 함수  $g$ 가

$$g = g(\mathbf{W}) = g(w_{11}, \dots, w_{ij}, \dots, w_{mn}) \quad (11.5.12)$$

로 주어졌다. 행렬식(Determinant)이 이러한 함수의 전형적인 예이다. 그러면, 함수  $g$ 의 행렬  $\mathbf{W}$ 에 대한 미분은

$$\frac{\partial g}{\partial \mathbf{W}} = \begin{pmatrix} \frac{\partial g}{\partial w_{11}} & \dots & \frac{\partial g}{\partial w_{1n}} \\ \vdots & & \vdots \\ \frac{\partial g}{\partial w_{m1}} & \dots & \frac{\partial g}{\partial w_{mn}} \end{pmatrix} \quad (11.5.13)$$

이다. 이제,  $\mathbf{W} = (w_{ij})$ 를 역행렬을 구할 수 있는  $n \times n$  차원이라고 하자. 그러면,

$$\mathbf{W}^{-1} = \frac{1}{\det \mathbf{W}} \text{adj}(\mathbf{W}) \quad (11.5.14)$$

이 된다. 여기서,  $\det \mathbf{W}$ 는 행렬식을 나타내며,  $\text{adj}(\mathbf{W})$ 는 수반행렬(Adjoint)이다. 수반행렬은

$$\text{adj}(\mathbf{W}) = \begin{pmatrix} W_{11} & \dots & W_{n1} \\ \vdots & & \vdots \\ W_{1n} & \dots & W_{nn} \end{pmatrix} \quad (11.5.15)$$

이며, 여기서  $W_{ij}$ 는 여인수(Cofactor)이다. 이는 행렬  $\mathbf{W}$ 에서  $i$ 번째 열과  $j$ 번째 행을 제외한 나머지  $(n-1) \times (n-1)$  행렬을 취한 후  $(-1)^{i+j}$ 를 곱하여 얻어진다. 행렬식은 여인수의 함수로

$$\det \mathbf{W} = \sum_{k=1}^n w_{ik} W_{ik} \quad (11.5.16)$$

와 같이 표현된다. 식 (11.5.16)을  $w_{ij}$ 에 대하여 미분하면

$$\frac{\partial \det \mathbf{W}}{\partial w_{ij}} = W_{ij} \quad (11.5.17)$$

이 된다. 그러면, 식 (11.5.15)와 (11.5.16)에 의해

$$\frac{\partial \det \mathbf{W}}{\partial \mathbf{W}} = \text{adj}(\mathbf{W})^T \quad (11.5.18)$$

이다. 한편, 식 (11.5.14)에서  $\text{adj}(\mathbf{W}) = (\det \mathbf{W}) \mathbf{W}^{-1}$ 이므로

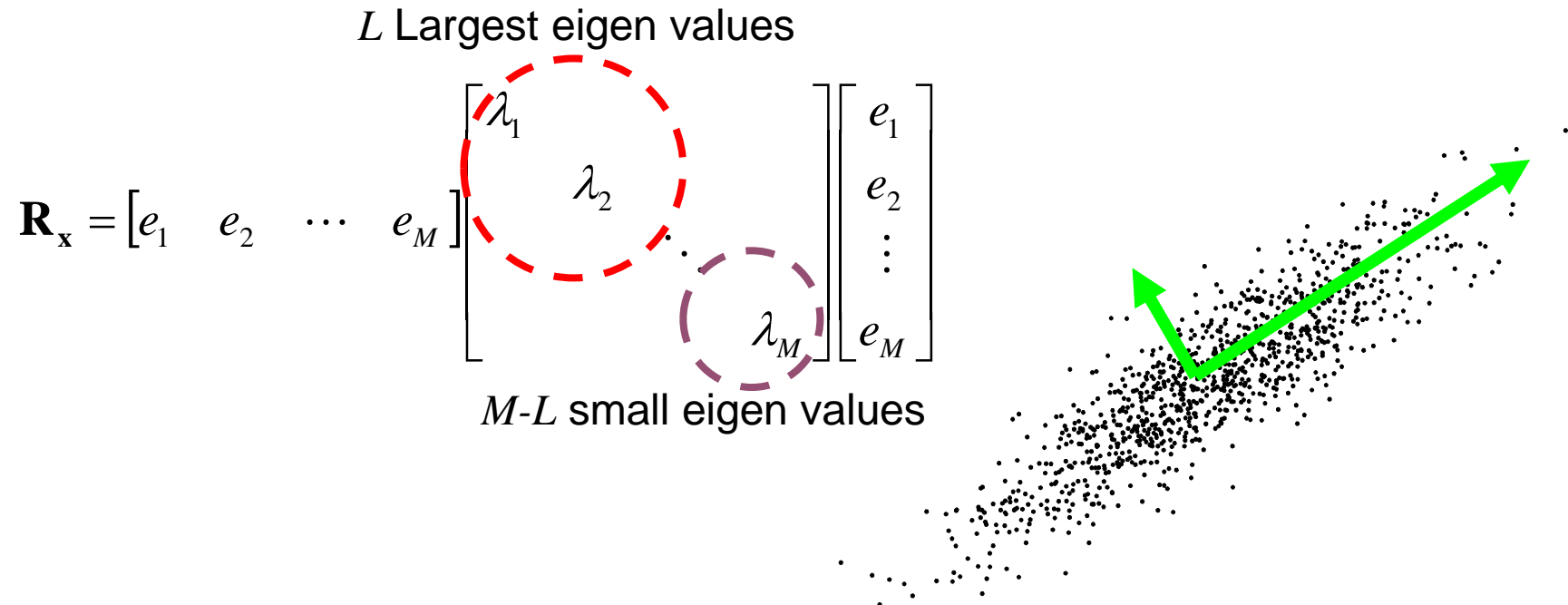
$$\frac{\partial \log |\det \mathbf{W}|}{\partial \mathbf{W}} = \frac{1}{|\det \mathbf{W}|} \frac{\partial \det \mathbf{W}}{\partial \mathbf{W}} = (\mathbf{W}^T)^{-1} \quad (11.5.19)$$

을 얻게 된다. 이 결과가 식 (11.5.6)의 유도에 사용되었다.



# 11.5. Comparison with PCA and Applications

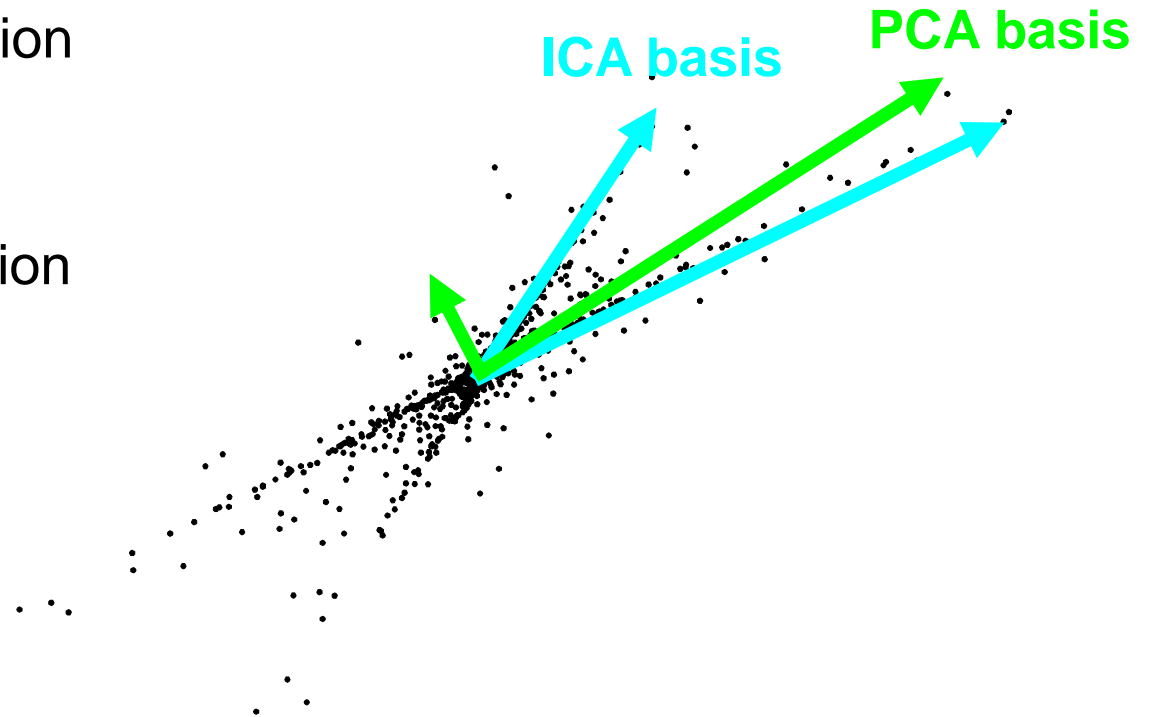
- PCA
- Calculate covariance matrix  $\mathbf{R}_x = E(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^H$
- Using SVD, find  $M$  eigenvalues and eigenvectors.
- Choose  $L$  eigen vectors corresponding to  $L$  largest eigen values.



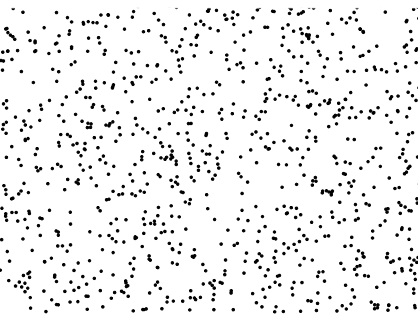
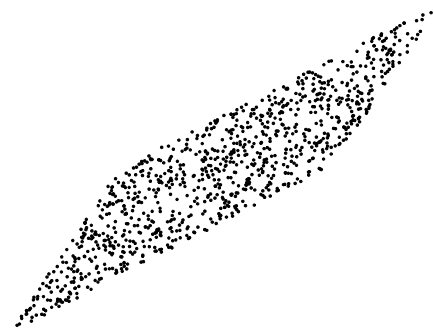
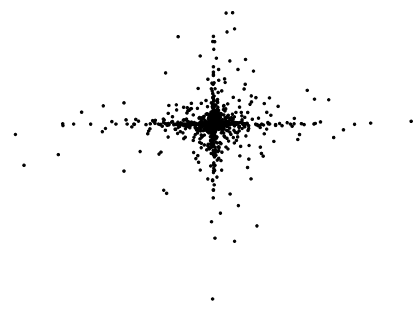
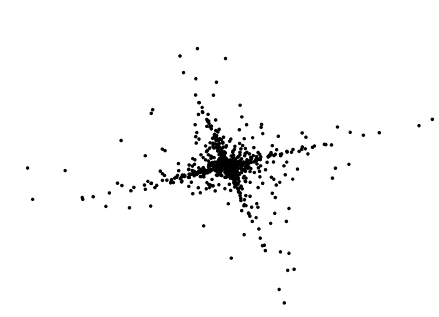
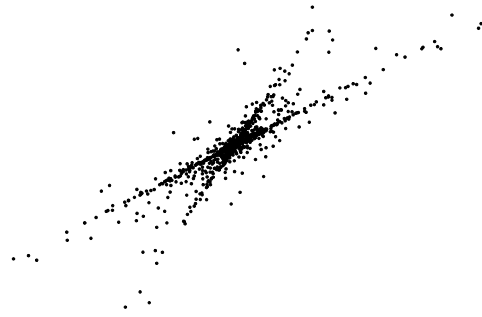
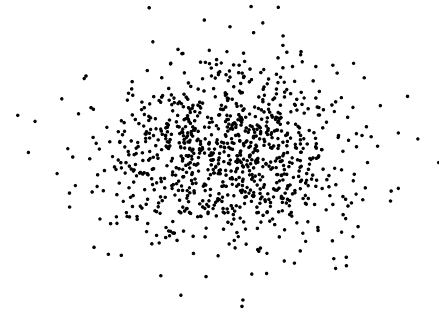
# Comparison of PCA and ICA

- Independent = uncorrelated ?

- ❖ Only for Gaussian data
- ❖ dependence = correlation
- ❖ Generally,
- ❖ dependence  $\neq$  correlation



# Whitened signal and independent signal

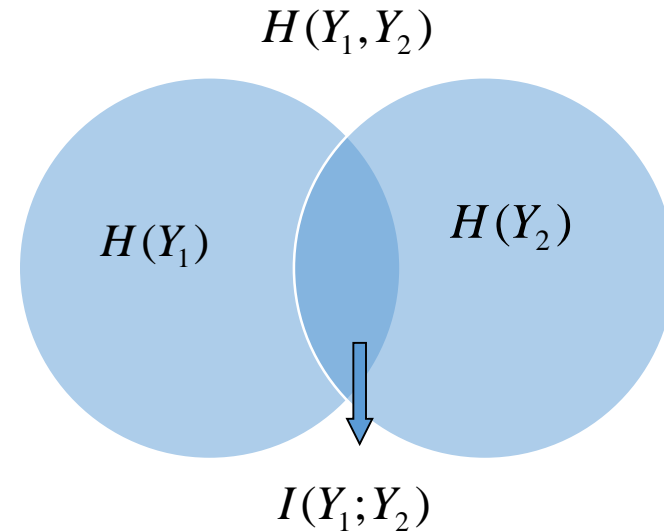


**Observations**

**Whitened**

**Independent**

# Unifying Information-Theoretic Framework(1)



- Max.  $H(y)$ : InfoMax
- Minimizing Mutual Information
- Negentropy Maximization

$$J(\mathbf{u}) = D(p(\mathbf{u}) \parallel p_G(\mathbf{u})) = \int p(\mathbf{u}) \log \frac{p(\mathbf{u})}{p_G(\mathbf{u})} d\mathbf{u}$$

## Unifying Information-Theoretic Framework(2)

- Maximum Likelihood Estimation

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

$$L(\mathbf{u}, \mathbf{W}) = \log|\det(\mathbf{W})| + \sum_{i=1}^N \log p_i(u_i)$$

$$\begin{aligned} \Delta \mathbf{W} \propto \frac{\partial L(\mathbf{u}, \mathbf{W})}{\partial \mathbf{W}} &= (\mathbf{W}^T)^{-1} + \frac{\frac{\partial p(\mathbf{u})}{\partial \mathbf{u}}}{p(\mathbf{u})} \mathbf{x}^T \\ &= (\mathbf{W}^T)^{-1} - \varphi(\mathbf{u}) \mathbf{x}^T \end{aligned}$$

## Unifying Information-Theoretic Framework(3)

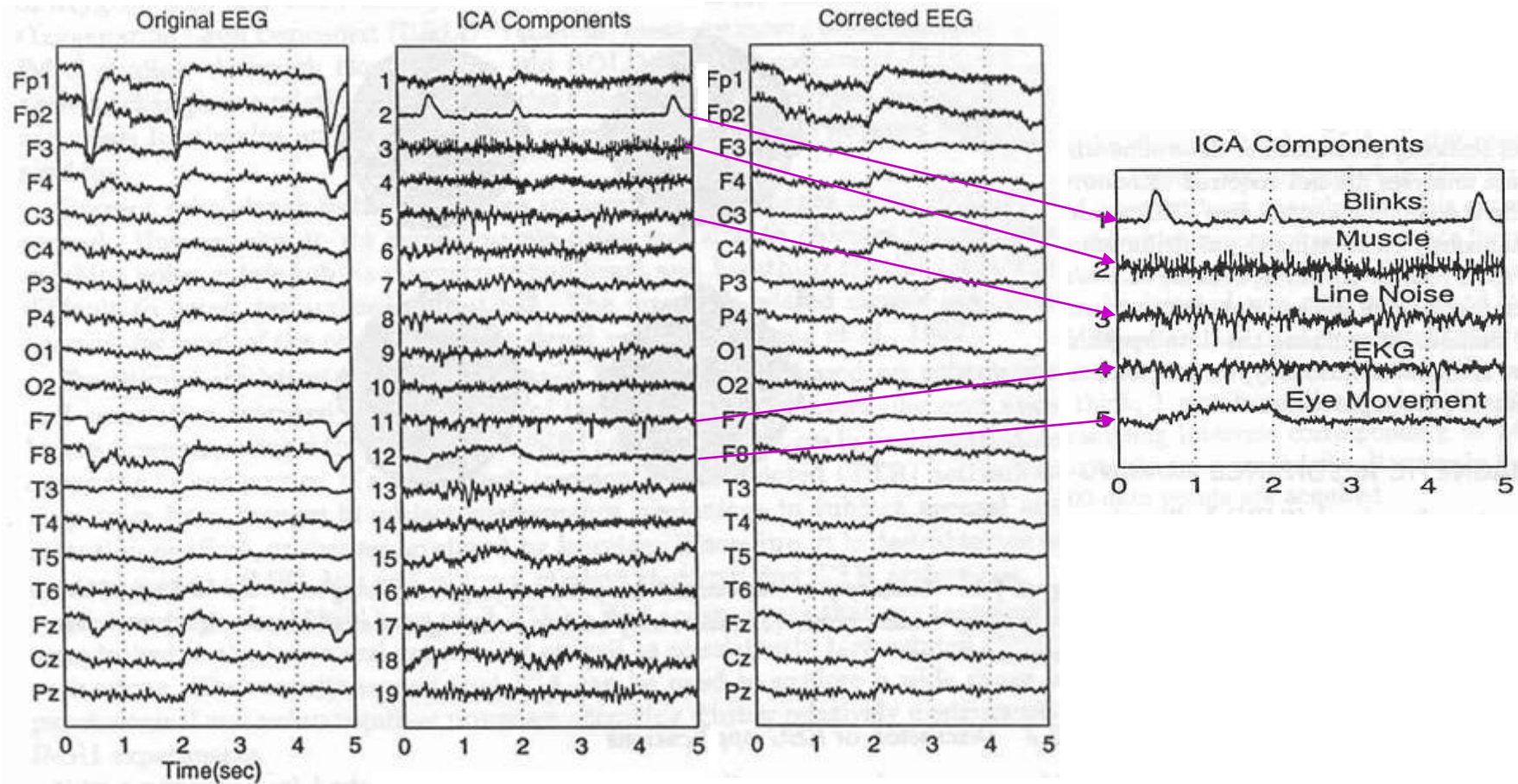
- High-Order Moments and Cumulants
- Nonlinear PCA
- Bussgang Algorithms

(Te-Won Lee et al., Computers and Mathematics with Applications, 2000. [1])

# Applications of ICA

- **Image processing**
  - e.g. denoising, feature extraction, etc
- **Biomedical data analysis**
  - e.g. analysis of EEG, MEG, ECG, etc...
- **Speech and audio processing**
  - e.g. speech separation, feature extraction
- **Telecommunication**
  - e.g. MIMO System, etc.
- **Bioinformatics**
  - e.g. micro-array data analysis, etc
- **Financial data analysis**

- Blind Source Separation of EEG Signals

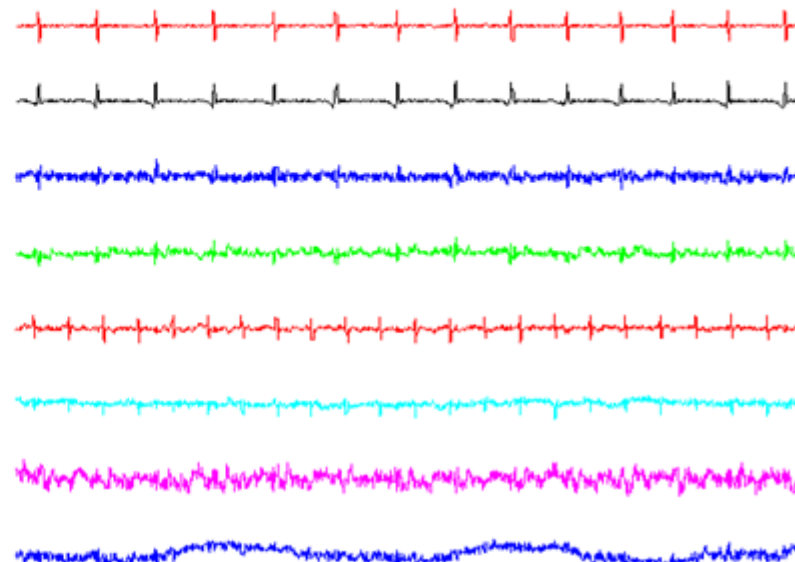




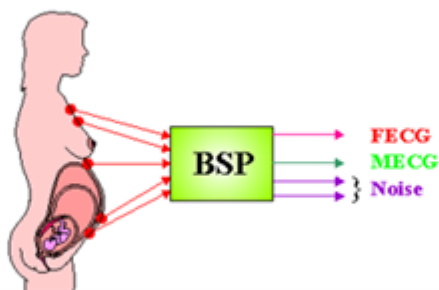


ECG data

Flex  
ICA



Separated Components



The 3<sup>rd</sup> & 4<sup>th</sup> components: Fetus Signals  
The 7<sup>th</sup> & 8<sup>th</sup> components: Mother Signals

그림 11.16. 유연한 독립성분분석(Flexible ICA)에 의한 임신부와 태아 신호 분리

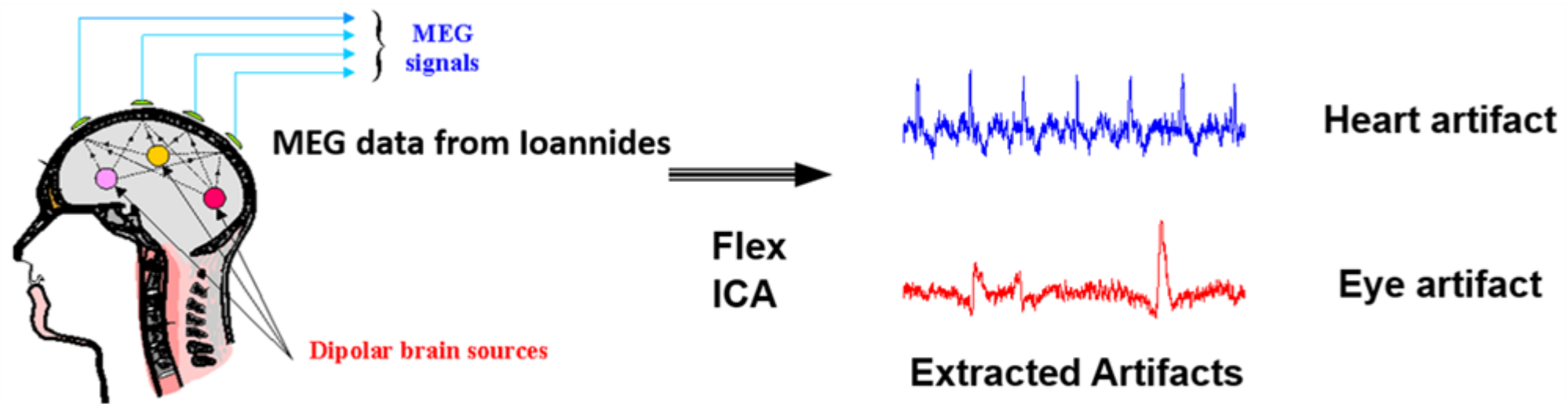


그림 11.17. MEG(Magnetoencephalography) 데이터에서 심장과 눈에 의한 성분 분리

## 11.6. Projection Pursuit Learning for Latent Structure

Data Model  $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t)$

where  $\mathbf{s}(t) \in R^M (M \leq N)$  are independent sources.

$$p(\mathbf{s}) = \prod_{i=1}^M p_i(s_i)$$

Objective

$$\mathbf{u}(t) = \mathbf{W}\mathbf{x}(t)$$

where  $\mathbf{W} \in R^{P \times N} (P \ll N)$

If  $P=2$ , visualization!

Negentropy

$$KL(p||p_G) = \int p_u(u_i) \log \frac{p_u(u_i)}{P_G(u_i)} du_i$$

$$KL(p_F||p_G) = \int p_u(\mathbf{u}) \log \frac{\prod_{i=1}^P p_i(u_i)}{P_G(\mathbf{u})} d\mathbf{u}$$

$$KL(p_F||p_G) = \frac{1}{2} \log((2\pi e)^P \det(\mathbf{R}_{uu})) + \int p_u(\mathbf{u}) \log \prod_{i=1}^P p_i(u_i) d\mathbf{u}$$

$$\mathbf{R}_{uu} = E[\mathbf{u}\mathbf{u}^T] = E[\mathbf{W}\mathbf{x}\mathbf{x}^T\mathbf{W}^T] = \mathbf{W}E[\mathbf{x}\mathbf{x}^T]\mathbf{W}^T = \mathbf{W}\mathbf{R}_{xx}\mathbf{W}^T$$

$$\frac{\partial}{\partial \mathbf{W}} \left[ \frac{P}{2} \log(2\pi e) + \frac{1}{2} \log(\det(\mathbf{W}\mathbf{R}_{xx}\mathbf{W}^T)) + \sum_{i=1}^P E[\log p_i(u_i)] \right]$$

$$\frac{\partial}{\partial \mathbf{W}} \left[ \frac{1}{2} \log(\det(\mathbf{W}\mathbf{R}_{xx}\mathbf{W}^T)) \right] = (\mathbf{W}\mathbf{R}_{xx}\mathbf{W}^T)^{-1} \mathbf{W}\mathbf{R}_{xx}$$

$$\frac{\partial}{\partial \mathbf{W}} \left[ \sum_{i=1}^P E[\log p_i(u_i)] \right] = E[\mathbf{f}(\mathbf{u})\mathbf{x}^T]$$

$$\frac{\partial}{\partial \mathbf{W}} KL(p_F || p_G) = (\mathbf{W} \mathbf{R}_{xx} \mathbf{W}^T)^{-1} \mathbf{W} \mathbf{R}_{xx} + E[\mathbf{f}(\mathbf{u}) \mathbf{x}^T]$$

$$\Delta \mathbf{W}_t = \eta_t \left( \frac{\partial}{\partial \mathbf{W}} KL(p_F || p_G) \right) \mathbf{W}_t \mathbf{W}_t^T$$

$$\Delta \mathbf{W}_t = \eta_t \left( \frac{\partial}{\partial \mathbf{W}} KL(p_F || p_G) \right) (\mathbf{W}_t \mathbf{W}_t^T + \epsilon_t \mathbf{I})$$

$$\Delta \mathbf{W}_t = \eta_t (\mathbf{I} + \mathbf{f}(\mathbf{u}_t) \mathbf{u}_t^T) \mathbf{W}_t + \eta_t \epsilon_t ((\mathbf{W} \mathbf{R}_{xx} \mathbf{W}^T)^{-1} \mathbf{W} \mathbf{R}_{xx} + \mathbf{f}(\mathbf{u}) \mathbf{x}^T)$$

$$\Delta \mathbf{W}_t = \eta_t (\mathbf{I} + \mathbf{f}(\mathbf{u}_t) \mathbf{u}_t^T) \mathbf{W}_t$$

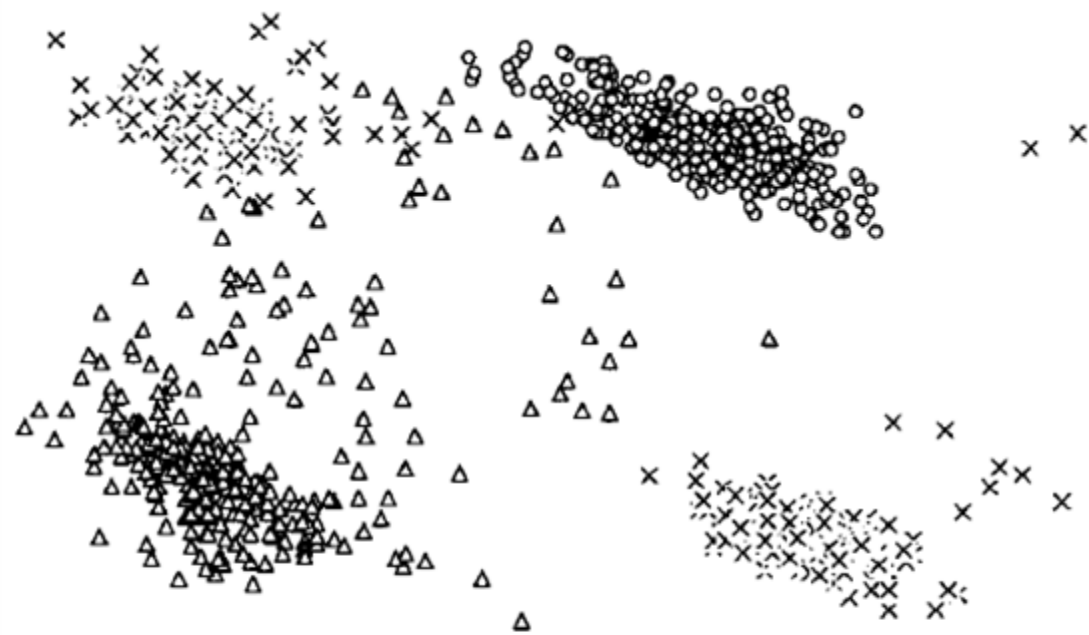


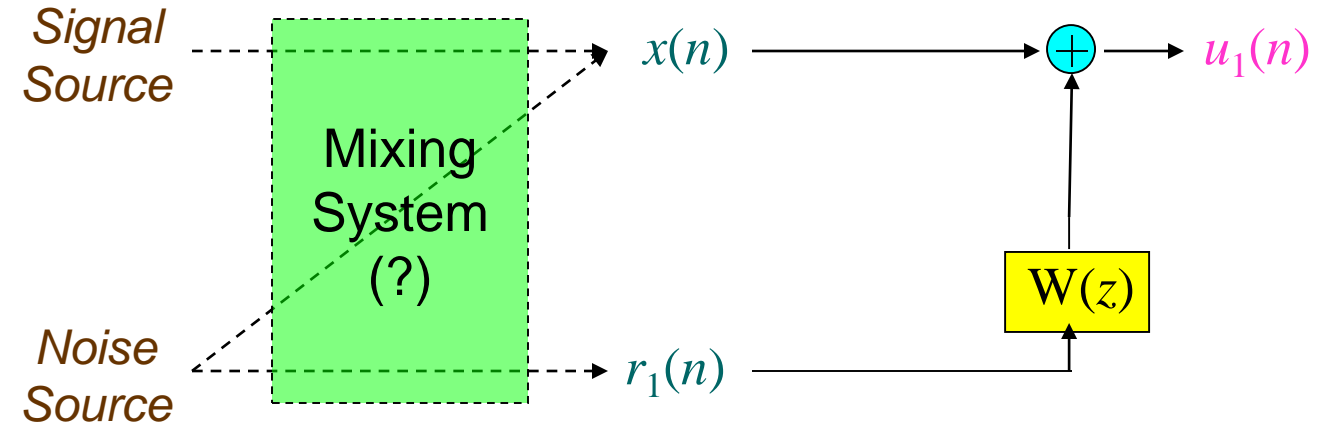
그림 11.18. ICA에 의한 데이터 시각화 예

( × 총류, Δ 고리류, ○ 균일류 )

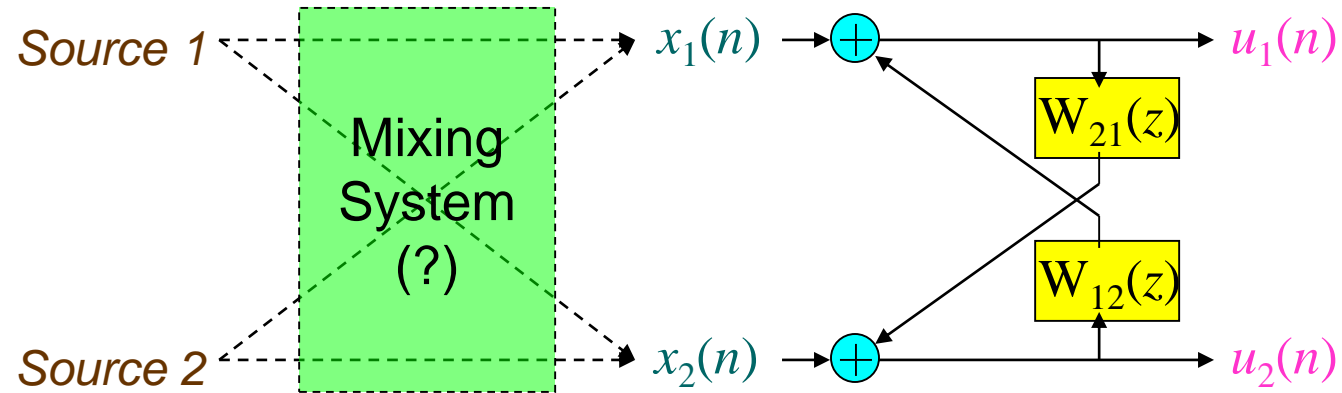
[33. Mark Girolami et al. 1998]

## 11.8 Convolved Mixtures: ANC vs. BSS

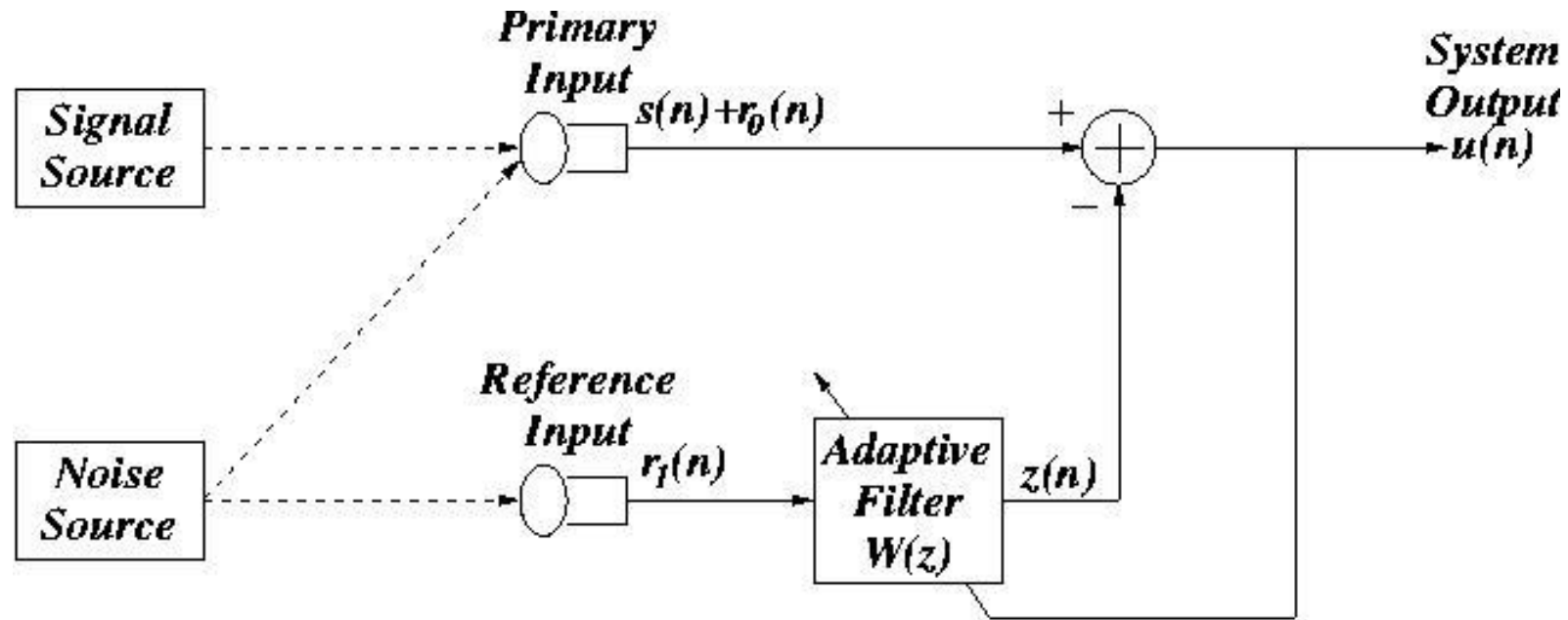
- ANC



- BSS



# Adaptive Noise Canceling(ANC)



- System output  $u(n) = s(n) + r_0(n) - \sum_{k=1}^{L_a} w(k)r_1(n-k)$
- Minimize the output power  $\min E[u^2] = E[s^2] + \min E[(r_0 - z)^2]$
- The LMS algorithm  $\Delta w(k) \propto u(n)r_1(n-k)$



# ICA-Based Approach to ANC [36]

- Maximizing entropy

- Set dummy output  $v = r_1$

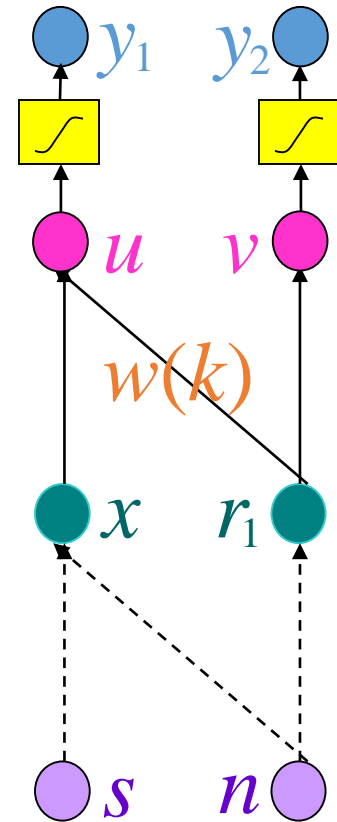
$$H(y_1, y_2) = -E[\log(p(y_1, y_2))] = -E[\log(\frac{p(x, r_1)}{|J|})]$$

$$\text{where } J = \frac{\partial y_1}{\partial x} \frac{\partial y_2}{\partial r_1} - \frac{\partial y_1}{\partial r_1} \frac{\partial y_2}{\partial x} = \frac{\partial y_1}{\partial u} \frac{\partial y_2}{\partial v}$$

- Learning rules of adaptive filter coefficients in ANC (Park et al. 2002)

$$\Delta w(k) \propto \frac{\partial H(y_1, y_2)}{\partial w(k)} = \frac{\partial}{\partial w(k)} \log |J| = \left( \frac{\partial y_1}{\partial u} \right)^{-1} \frac{\partial}{\partial w(k)} \left( \frac{\partial y_1}{\partial u} \right)$$

$$= \varphi(u(n)) r_1(n-k) \quad \text{where} \quad \varphi(u) = - \left( \frac{\partial y_1}{\partial u} \right)^{-1} \frac{\partial^2 y_1}{\partial u^2}$$





Known Noise Source



Input Signal with Noise



Output Signal after Process

# Time Domain Approach to BSS

- Convolved mixtures

$$x_i(n) = \sum_{j=1}^N \sum_{k=0}^{K-1} a_{ij}(k) s_j(n-k)$$

- Feedback architecture

$$u_i(n) = \sum_{k=0}^K w_{ii}(k) x_i(n-k) + \sum_{j=1, j \neq i}^N \sum_{k=1}^K w_{ij}(k) u_j(n-k)$$

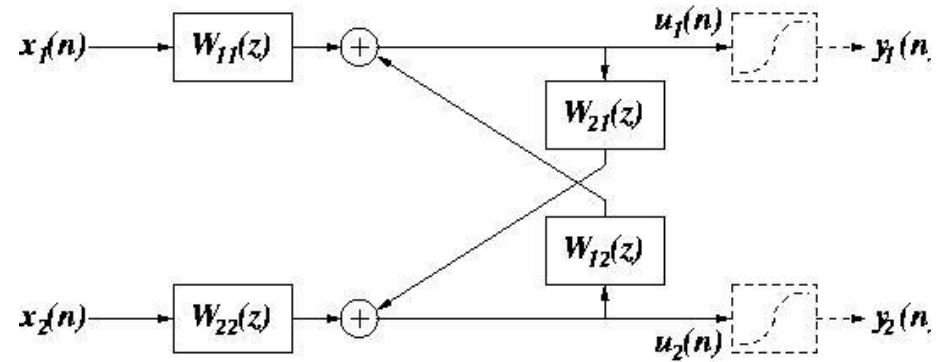
- Learning rules

$$\Delta w_{ii}(0) \propto 1/w_{ii}(0) - \varphi(u_i(n)) x_i(n),$$

$$\Delta w_{ii}(k) \propto -\varphi(u_i(n)) x_i(n-k), \quad k \neq 0,$$

$$\Delta w_{ij}(k) \propto -\varphi(u_i(n)) u_j(n-k), \quad i \neq j,$$

$$\varphi(u_i(n)) = -\frac{\partial p(u_i(n))}{\partial u_i(n)} \frac{1}{p(u_i(n))}$$



# Frequency Domain Approach to BSS

- In the frequency domain

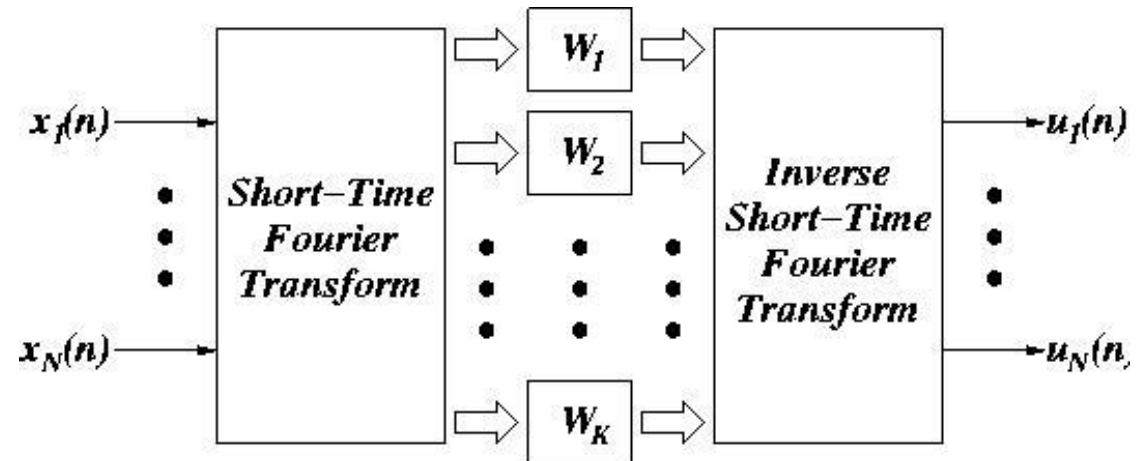
$$X_f(n) = A_f \cdot S_f(n), \quad \forall f$$

- Complex score function

$$\varphi(u_i) = -\frac{\frac{\partial p(|u_i|)}{\partial(|u_i|)}}{p(|u_i|)} \exp(j \cdot \angle u_i)$$

- Learning rule

$$\Delta W \propto [I - \varphi(u) u^H] W$$



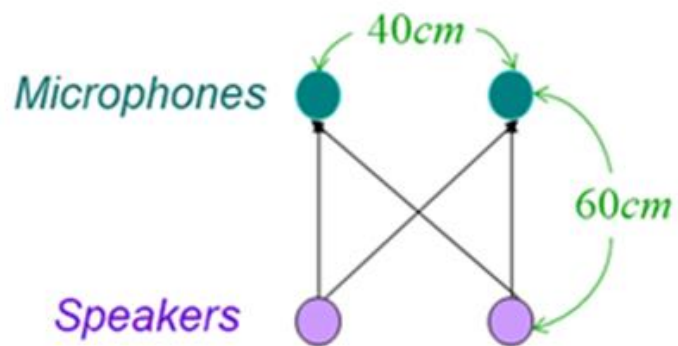


그림 11.24. 2×2 실험환경

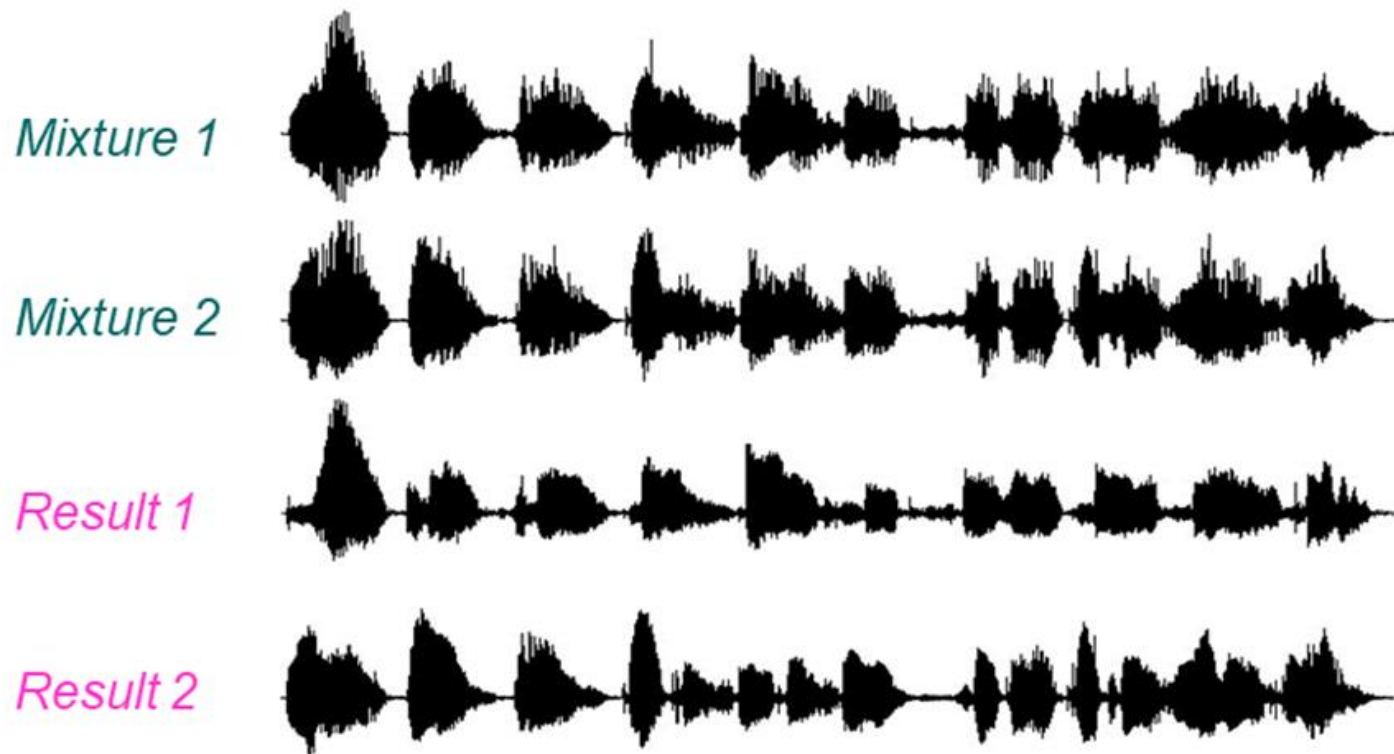


그림 11.25. 2×2 실험환경에서 필터뱅크 방법으로 암묵신호분리 한 결과  
[37. Hyung-Min Park et al. 2006]