Learning From Data
Lecture 12: Unsupervised Learning III

Yang Li **yangli@sz.tsinghua.edu.cn**

TBSI

December 15, 2022

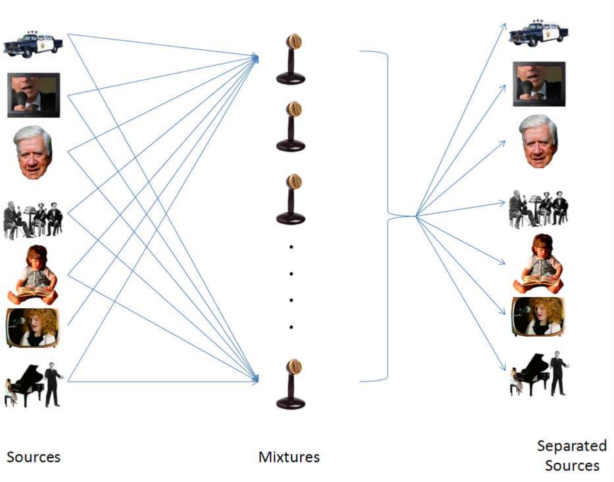
Today's Lecture

Unsupervised Learning (Part III)

- Independent Component Analysis (ICA)
- Canonical Correlation Analysis (CCA)

The cocktail party problem

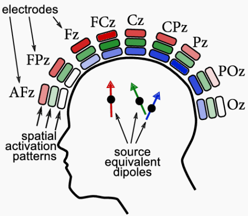
- n microphones at different locations of the room, each recording a mixture of n sound sources
- How to “unmix" the sound mixtures?

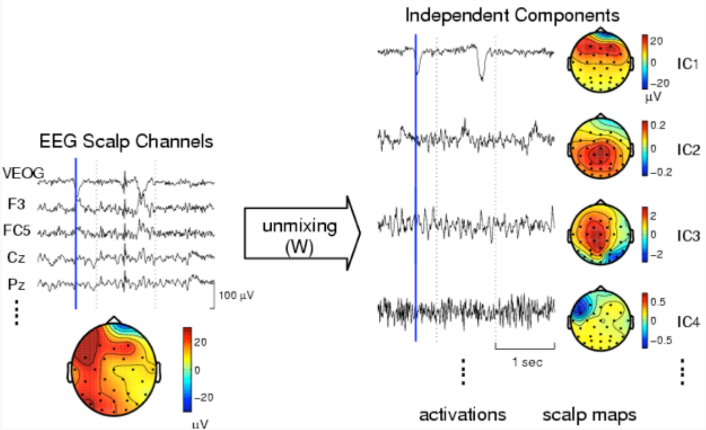


Sample audio: https://cnl.salk.edu/~tewon/Blind/blind_audio.html, http://www.kecl.ntt.co.jp/icl/signal/sawada/demo/bss2to4/index.html

EEG Analysis

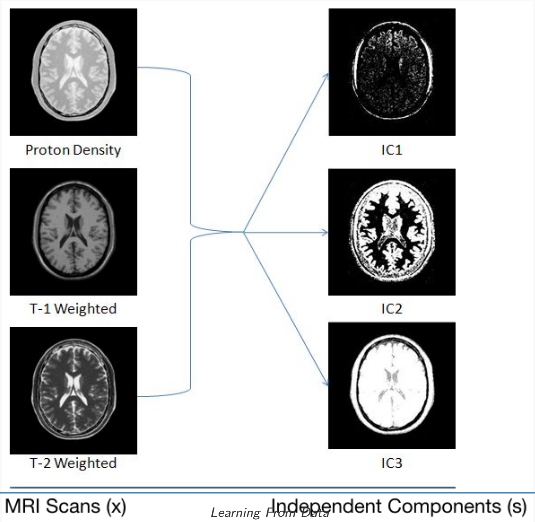
- Electrodes on patient scalp measure a mixture of different brain activations
- Finding independent activation sources helps removing artifacts in the signal





Brian imaging

- Different brain matters: gray matter, white matter, cerebrospinal fluid (CSF), fat, muscle/skin, glial matter etc.
- An MRI scan is a mixture of magnetic response signals from different brain matters



Problem Model

Case: $n = 2$

- Observed random variables: x_1, x_2
- Independent sources: $s_1, s_2 \in \mathbb{R}$

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

A is called the **mixing matrix**

$$x = As$$

The blind source separation (cocktail party) problem

Given repeated observation $\{x^{(i)}; i = 1, \dots, m\}$, recover sources $s^{(i)}$ that generated the data $(x^{(i)} = As^{(i)})$

Independent Component Analysis (ICA)

The blind source separation (cocktail party) problem

Given repeated observation $\{x^{(i)}; i = 1, \ldots, m\}$, recover sources $s^{(i)}$ that generated the data ($x^{(i)} = As^{(i)}$)

Let $W = A^{-1}$ be the **unmixing matrix**
Goal of ICA: Find W, such that given $x^{(i)}$, the sources can be recovered by $s^{(i)} = Wx^{(i)}$

$$W = \begin{bmatrix} -w_1^T- \\ \vdots \\ -w_n^T- \end{bmatrix}$$

ICA Ambiguities

Assume data is **non Gaussian**, ICA has two ambiguities:

- Variance of the sources: *We can fix the magnitude of s_i by setting $\mathbb{E}[s_i^2] = 1$*
- Order of the sources $s_1, \ldots, s_n$:
  Let P be a permutation matrix, then we have $x = APP^{-1}s$.

Why is Gaussian data problematic?

- The distribution of any rotation of Gaussian x has the same distribution as x.
- As long as at least one s_j is non-Gaussian, given enough data, we can recover the n independent sources.

Densities and Linear Transformations

Theorem 1

If random vector s has density p_s, and $x = As$ for a square, invertible matrix A, then the density of x is

$$p_x(x) = p_s(Wx) \cdot |W|$$

where $W = A^{-1}$.

ICA Algorithm

The joint distribution of *independent* sources $s = \{s_1, \dots, s_n\}$:

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

The density of observation $x = As = W^{-1}s$ is:

$$p_x(x) = p_s(s)|W| = \prod_{j=1}^{n} p_s(s_j)|W| = \prod_{j=1}^{n} p_s(w_j^T x)|W|$$

Choose the sigmoid function $g(s) = \frac{1}{1+e^{-s}}$ as the *non-Gaussian* cdf for p_s, then

$$p_s(s) = g'(s)$$

This appears to be a heuristic choice, yet it can be justified rigorously in other interpretations.

ICA Algorithm

Given i.i.d. training samples $\{x^{(1)}, \ldots, x^{(m)}\}$, the log likelihood is

$$
\begin{aligned}
l(W) = \sum_{i=1}^{m} log(p_x(x^{(i)})) &= \sum_{i=1}^{m} log(\prod_{j=1}^{n} p_s(w_j{}^T x)|W|) \\
&= \sum_{i=1}^{m} \left(\sum_{j=1}^{n} \log g'(w_j{}^T x^{(i)}) + \log |W| \right)
\end{aligned}
$$

Stochastic gradient ascent learning rule for sample $x^{(i)}$:

$$
W := W + \alpha \left(\begin{bmatrix} 1 - 2g(w_1{}^T x^{(i)}) \\ \vdots \\ 1 - 2g(w_n{}^T x^{(i)}) \end{bmatrix} x^{(i)}{}^T + (W^T)^{-1} \right)
$$

Check this at home!

Theoretical Motivation of ICA

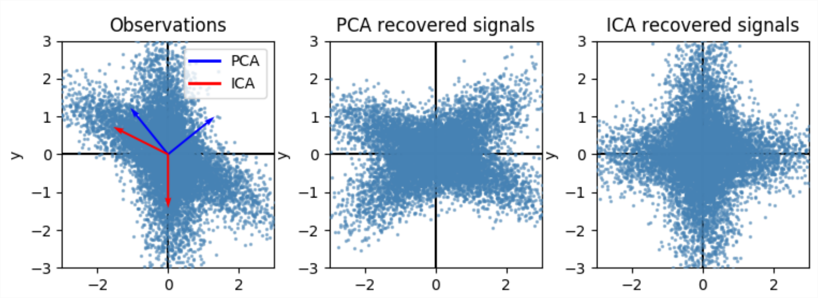
- Originally proposed by Jutten & Herault (1991) [1] *90 years later than PCA*
- Equivalent to learning projection directions $w_1, \ldots, w_n$ that
  - maximize the **sum of non-gaussianity** of the projected signals
  - minimize the **mutual information** of the projected signals

  under the constraint that $w_1^T x, \ldots, w_n^T x$ are uncorrelated. [2]

[1] Christian Jutten, Jeanny Herault, Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture, Signal Processing, Vol 24:1, 1991

[2] Hyvärinen, Aapo, and Erkki Oja. "Independent component analysis: algorithms and applications." Neural networks 13.4-5 (2000): 411-430.

ICA vs PCA



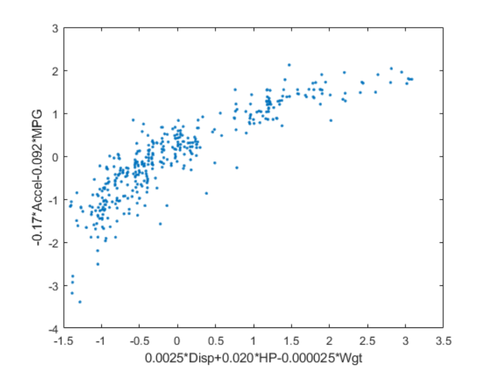
PCA	ICA
approximately Gaussian data	non-Gaussian data
removes correlation (low order dependence)	removes correlations and higher order dependence
ordered importance	all components are equally important
orthogonal	not orthogonal

Canonical Correlation Analysis

Canonical correlation analysis (CCA) finds the associations among two sets of variables.

Example: two sets of measurements of 406 cars:

- Specification: Engine displacement (Disp), horsepower (HP), weight (Wgt)
- Measurement: Acceleration (Accel), MPG



find important features that explain covariation between sets of variables

CCA Definitions

- Random vectors $X = \begin{bmatrix} x_1 \\ \vdots \\ x_{n_1} \end{bmatrix}$ and $Y = \begin{bmatrix} y_1 \\ \vdots \\ y_{n_2} \end{bmatrix}$
- Covariance matrix $\Sigma_{XY} = cov(X, Y)$
- CCA finds vectors a and b such that the random variables $a^T X$ and $b^T Y$ maximize the correlation

$$\rho = corr(a^T X, b^T Y)$$

- $U = a^T X$ and $V = b^T Y$ are called **the first pair of canonical variables**
- Subsequent pairs of canonical variables maximizes ρ while being *uncorrelated* with all previous pairs

Review: Singular Value Decomposition

A generalization of eigenvalue decomposition to rectangle ($m \times n$) matrices M.

$$M = U\Sigma V^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

- $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$ are orthogonal matrices
- $\Sigma \in \mathbb{R}^{m \times n}$ is a **rectangular diagonal matrix**.
  Examples:

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \\ 0 & 0 & 0 \end{bmatrix} \quad \Sigma = \begin{bmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & \sigma_3 & 0 \end{bmatrix}$$

  Diagonal entries $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k$, $k = \min(n, m)$ are called **singular values of** M.

Review: Singular Value Decomposition

A non-negative real number σ is a singular value for $M \in \mathbb{R}^{m \times n}$ if and only if there exist unit-length $u \in \mathbb{R}^m$ and $v \in \mathbb{R}^n$ such that

$$Mv = \sigma u$$
$$M^T u = \sigma v$$

u is called the **left singular vector** of σ, v is called the **right singular vector** of σ

Connection to eigenvalue decomposition

Given SVD of matrix $M = U\Sigma V^T$,

- $M^T M = (V\Sigma^T U^T)(U\Sigma V^T) = V(\Sigma^T\Sigma)V^T$ ← *v_i is an eigenvector of $M^T M$ with eigenvalue σ_i^2*
- $MM^T = (U\Sigma V^T)(V^T\Sigma^T U) = U(\Sigma\Sigma^T)U^T$ ← *u_i is an eigenvector of MM^T with eigenvalue σ_i^2*

CCA Derivations

The original problem:

$$(a_1, b_1) = \underset{a \in \mathbb{R}^{n_1}, b \in R^{n_2}}{\operatorname{argmax}} \ corr(a^T X, b^T Y) \tag{1}$$

Assume $\mathbb{E}[x_1] = \cdots = \mathbb{E}[x_{n_1}] = \mathbb{E}[y_1] = \cdots = \mathbb{E}[y_{n_2}] = 0$,

$$\begin{aligned} corr(a^T X, b^T X) &= \frac{\mathbb{E}[(a^T X)(b^T Y)]}{\sqrt{\mathbb{E}[(a^T X)^2]\mathbb{E}[(a^T Y)^2]}} \\ &= \frac{a^T \Sigma_{XY} b}{\sqrt{a^T \Sigma_{XX} a}\sqrt{b^T \Sigma_{YY} b}} \end{aligned}$$

(1) is equivalent to:

$$(a_1, b_1) = \underset{\substack{a \in \mathbb{R}^{n_1}, b \in R^{n_2} \\ a^T \Sigma_{XX} a = b^T \Sigma_{YY} b = 1}}{\operatorname{argmax}} \quad a^T \Sigma_{XY} b \tag{2}$$

CCA Derivations

Define $\Omega \in R^{n_1 \times n_2}$, $c \in \mathbb{R}^{n_1}$ and $d \in \mathbb{R}^{n_2}$,

$$\Omega = \Sigma_{XX}^{-\frac{1}{2}} \Sigma_{XY} \Sigma_{YY}^{-\frac{1}{2}}$$
$$c = \Sigma_{XX}^{\frac{1}{2}} a$$
$$d = \Sigma_{YY}^{\frac{1}{2}} b$$

(2) can be written as

$$(c_1, d_1) = \underset{\substack{c \in \mathbb{R}^{n_1}, d \in \mathbb{R}^{n_2} \\ ||c||^2 = ||d||^2 = 1}}{\operatorname{argmax}} c^T \Omega d \tag{3}$$

(c_1, d_1) can be solved by SVD, then the first pair of canonical variables are

$$a_1 = \Sigma_{XX}^{-\frac{1}{2}} c_1, \quad b_1 = \Sigma_{YY}^{-\frac{1}{2}} d_1$$

CCA Derivations

$$(c_1, d_1) = \underset{\substack{c \in \mathbb{R}^{n_1}, d \in \mathbb{R}^{n_2} \\ ||c||^2 = ||d||^2 = 1}}{\operatorname{argmax}} \quad c^T \Omega d$$

Proposition 1

c_1 and d_1 are the left and right unit singular vectors of Ω with the largest singular value.

Theorem 2

c_i and d_i are the left and right unit singular vectors of Ω with the ith largest singular value.

CCA Algorithm

Input: Covariance matrices for centered data X and Y:

- Σ_{XY} , invertible Σ_{XX} and Σ_{YY}
- Dimension $k \leq \min(n_1, n_2)$

Output: CCA projection matrices A_k and B_k:

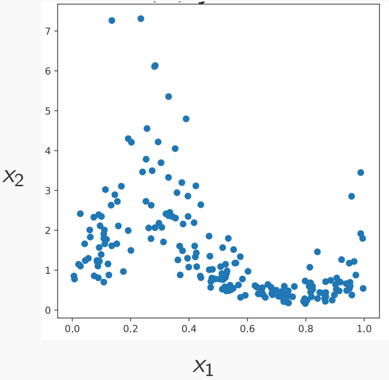
- Compute $\Omega = \Sigma_{XX}^{-\frac{1}{2}} \Sigma_{XY} \Sigma_{YY}^{-\frac{1}{2}}$
- Compute SVD decomposition of Ω

$$\Omega = \begin{bmatrix} | & \cdots & | \\ c_1 & \cdots & c_{n_1} \\ | & \cdots & | \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \\ & 0 & \end{bmatrix} \begin{bmatrix} -d_1^T- \\ \vdots \\ -d_{n_2}^T- \end{bmatrix}$$

- $A_k = \Sigma_{XX}^{-\frac{1}{2}} [c_1, \ldots, c_k]$ and $B_k = \Sigma_{YY}^{-\frac{1}{2}} [d_1, \ldots, d_k]$

Discussion of CCA

- CCA only measures linear dependencies
- Non-linear generalizations:
  - Kernel CCA (KCCA)
  - Deep CCA (DCCA)
  - Maximal HGR Correlation



Non-linear dependency between x_1 and x_2

PCA, ICA and CCA

Linear Subspace Learning

Given high dimensional random vector $\mathbf{x}$, transform it to a low-dimensional vector $\mathbf{y}$ through a projection matrix U:

$$y = U^T x$$

- PCA, ICA and CCA are all unsupervised linear subspace learning methods.

Name	What is U ?	goal	subspace
PCA	principal component (U)	remove (low order) correlation	single
ICA	unmixing matrix (W)	remove (high order) correlation	single
CCA	canonical projection matrices (A, B)	maximize correlation between feature pairs	paired