Convex Optimization
CMU-10725
Independent Component Analysis
(and matrix differentials)

Barnabás Póczos & Ryan Tibshirani
Independent Component Analysis
Independent Component Analysis

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

Model

Observations (Mixtures)

ICA estimated signals

original signals
We observe

\[
\begin{align*}
    x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) \\
    x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t)
\end{align*}
\]

We want

\[
\begin{align*}
    \begin{pmatrix} x_1(1) \\ x_2(1) \end{pmatrix}, \begin{pmatrix} x_1(2) \\ x_2(2) \end{pmatrix}, \ldots, \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}
\end{align*}
\]

But we don’t know \{a_{ij}\}, nor \{s_i(t)\}

Goal: Estimate \{s_i(t)\}, (and also \{a_{ij}\})
The Cocktail Party Problem

SOLVING WITH PCA

Sources: s(t)

Mixing: A ∈ ℝ^{M×M}

Observation: x(t) = As(t)

PCA Estimation: y(t) = Wx(t)
The Cocktail Party Problem
SOLVING WITH ICA

Sources

Mixing

Observation

ICA Estimation

\[ x(t) = As(t) \]

\[ y(t) = Wx(t) \]

\[ A \in \mathbb{R}^{M \times M} \]
ICA Theory
Statistical (in)dependence

**Definition (Independence)**

\[ Y_1, Y_2 \text{ are independent } \iff p(y_1, y_2) = p(y_1)p(y_2) \]

**Definition (Shannon entropy)**

\[ H(Y) = H(Y_1, \ldots, Y_m) = - \int p(y_1, \ldots, y_m) \log p(y_1, \ldots, y_m) dy. \]

**Definition (KL divergence)**

\[ 0 \leq KL(f \| g) = \int f(x) \log \frac{f(x)}{g(x)} dx \]

**Definition (Mutual Information)**

\[ 0 \leq I(Y_1, \ldots, Y_M) = \int p(y_1, \ldots, y_M) \log \frac{p(y_1, \ldots, y_M)}{p(y_1)\ldots p(y_M)} dy \]
Solving the ICA problem with i.i.d. sources

ICA problem:  $x = As$, $s = [s_1; \ldots; s_M]$ are jointly independent.

Ambiguity: $s = [s_1; \ldots; s_M]$ sources can be recovered only up to sign, scale and permutation.

Proof:
- $P = $ arbitrary permutation matrix,
- $\Lambda = $ arbitrary diagonal scaling matrix.

$\Rightarrow x = [AP^{-1}\Lambda^{-1}][\Lambda P s]$
Solving the ICA problem

Lemma:
We can assume that \( E[s] = 0 \).

Proof:
Removing the mean does not change the mixing matrix.
\[
x - E[x] = A(s - E[s]).
\]

In what follows we assume that \( E[ss^T] = I_M, E[s] = 0 \).
Whitening

  (We assumed centered data)

- Do **SVD**: $\Sigma \in \mathbb{R}^{N \times N}$, $\text{rank}(\Sigma) = M$,
  $\Rightarrow \Sigma = U D U^T$,
  where $U \in \mathbb{R}^{N \times M}$, $U^T U = I_M$, **Signular vectors**
  $D \in \mathbb{R}^{M \times M}$, diagonal with rank $M$. **Singular values**
Whitening

- Let $Q = D^{-1/2}U^T \in \mathbb{R}^{M \times N}$ whitening matrix
- Let $A^* = QA$
- $x^* = Qx = QA s = A^* s$ is our new (whitened) ICA task.

We have,

$$E[x^*x^{*T}] = E[Qxx^TQ^T] = Q\Sigma Q^T = (D^{-1/2}U^T)UDU^T(UD^{-1/2}) = I_M$$

$$\Rightarrow E[x^*x^{*T}] = I_M, \text{ and } A^*A^{*T} = I_M.$$
Whitening solves half of the ICA problem

Note:
The number of free parameters of an N by N orthogonal matrix is \((N-1)(N-2)/2\).

⇒ whitening solves half of the ICA problem

After whitening it is enough to consider orthogonal matrices for separation.
**Solving ICA**

**ICA task:** Given $\mathbf{x}$,

- find $\mathbf{y}$ (the estimation of $\mathbf{s}$),
- find $\mathbf{W}$ (the estimation of $\mathbf{A}^{-1}$)

**ICA solution:** $\mathbf{y} = \mathbf{Wx}$

- Remove mean, $E[\mathbf{x}] = 0$
- Whitening, $E[\mathbf{xx}^T] = \mathbf{I}$
- Find an orthogonal $\mathbf{W}$ optimizing an objective function
  - Sequence of 2-d Jacobi (Givens) rotations

---

![Images of data visualization](image)

**original**  |  **mixed**  |  **whitened**  |  **rotated** (demixed)
Optimization Using Jacobi Rotation Matrices

\( G(p, q, \theta) \triangleq \begin{pmatrix}
1 & \ldots & 0 & \ldots & 0 & \ldots & 0 \\
\vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\
0 & \ldots & \cos(\theta) & \ldots & -\sin(\theta) & \ldots & 0 \\
\vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\
0 & \ldots & \sin(\theta) & \ldots & \cos(\theta) & \ldots & 0 \\
\vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\
0 & \ldots & 0 & \ldots & 0 & \ldots & 1
\end{pmatrix} \)

\( \in \mathbb{R}^{M \times M} \)

Observation: \( x = As \)

Estimation: \( y = Wx \)

\[ W = \arg \min_{\tilde{W} \in \mathcal{W}} J(\tilde{W}x), \]

where \( \mathcal{W} = \{ W | W = \prod_i G(p_i, q_i, \theta_i) \} \)
ICA Cost Functions

Let \( y = Wx \), \( y = [y_1; \ldots; y_M] \), and let us measure the dependence using Shannon’s mutual information:

\[
J_{ICA_1}(W) = I(y_1, \ldots, y_M) = \int p(y_1, \ldots, y_M) \log \frac{p(y_1, \ldots, y_M)}{p(y_1) \cdots p(y_M)} dy,
\]

Let \( H(y) = H(y_1, \ldots, y_m) = -\int p(y_1, \ldots, y_m) \log p(y_1, \ldots, y_m) dy \).

**Lemma**

\[
H(Wx) = H(x) + \log | \det W |
\]  
Proof: Homework

Therefore,

\[
I(y_1, \ldots, y_M) = \int p(y_1, \ldots, y_M) \log \frac{p(y_1, \ldots, y_M)}{p(y_1) \cdots p(y_M)}
\]

\[
= -H(y_1, \ldots, y_M) + H(y_1) + \ldots + H(y_M)
\]

\[
= -H(x_1, \ldots, x_M) - \log | \det W | + H(y_1) + \ldots + H(y_M).
\]
ICA Cost Functions

\[ I(y_1, \ldots, y_M) = \int p(y_1, \ldots, y_M) \log \frac{p(y_1, \ldots, y_M)}{p(y_1) \ldots p(y_M)} \]

\[ = -H(y_1, \ldots, y_M) + H(y_1) + \ldots + H(y_M) \]

\[ = -H(x_1, \ldots, x_M) - \log |\det W| + H(y_1) + \ldots + H(y_M). \]

\[ H(x_1, \ldots, x_M) \text{ is constant, } \log |\det W| = 0. \]

Therefore,

\[ J_{ICA_2}(W) \doteq H(y_1) + \ldots + H(y_M) \]

The covariance is fixed: I. Which distribution has the largest entropy?

⇒ go away from normal distribution
Central Limit Theorem

The sum of independent variables converges to the normal distribution

$\Rightarrow$ For separation go far away from the normal distribution

$\Rightarrow$ Negentropy, $|\text{kurtozis}|$ maximization

Figs from Ata Kaban
ICA Algorithms
Maximum Likelihood ICA Algorithm

- simplest approach
- requires knowing densities of hidden sources \( \{ f_i \} \)

\[
x(t) = As(t), \quad s(t) = Wx(t), \quad \text{where} \quad A^{-1} = W = [w_1; \ldots; w_M] \in \mathbb{R}^{M \times M}
\]

\[
L = \frac{T}{t=1} \log P_x(x(t)) = \frac{T}{t=1} \log P_x(AS(t)) \quad \Rightarrow \quad \max_A \quad A^{-1}P_S(A^{-1}x(t))
\]

\[
\rho_{As}(u) = A^{-1}P_S(A^{-1}u) \quad \Rightarrow \quad L = \frac{T}{t=1} \log A^{-1}P_S(s(t)) = T \log |W| + \frac{T}{t=1} \log P_S(s(t))
\]

\[
= T \log |W| + \sum_{t=1}^{T} \sum_{i=1}^{M} \log P_{S,i}(w_i; x(t)) \quad \Rightarrow \quad \max_w \quad f_i(w_i; x(t))
\]

David J.C. MacKay (97)
Maximum Likelihood ICA Algorithm

\[ L = T \log |W| + \sum_{t=1}^{T} \sum_{b=1}^{M} \log f_{eb}(w_{eb} x(t)) \]

\[ \Rightarrow \max_{W} L \Rightarrow \frac{\partial L}{\partial W_{ij}} = 0 \]

\[ \frac{\partial L}{\partial W_{ij}} = T (W^T)^{-1} + \sum_{t=1}^{T} \frac{\partial}{\partial W_{ij}} \left[ \sum_{b=1}^{M} \log f_{eb}(w_{eb} x(t)) \right] \]

\[ = T (W^T)^{-1} + \sum_{t=1}^{T} \frac{f_i'(w_i x(t))}{f_i(w_i x(t))} x_j(t) \]

\[ \Rightarrow \Delta W \propto [W^T]^{-1} + \frac{1}{T} \sum_{t=1}^{T} g(Wx(t))x^T(t), \text{ where } g_i = f_i'/f_i \]
ICA algorithm based on Kurtosis maximization

Kurtosis = 4\textsuperscript{th} order cumulant

Measures

- the distance from normality
- the degree of peakedness

\[ \kappa_4(y) = \mathbb{E}\{y^4\} - \frac{3}{\mathbb{E}\{y^2\}^2} = 3 \text{ if } \mathbb{E}\{y\} = 0 \text{ and whitened} \]

\[ \kappa_4(y) = -\frac{2}{15} \quad \kappa_4(y) = 0 \quad \kappa_4(y) = 12 \]
The Fast ICA algorithm (Hyvarinen)

- Given whitened data $z$
- Estimate the $1^{st}$ ICA component:

\[ y = w^T z, \quad \|w\| = 1, \quad \iff w^T = 1^{st} \text{ row of } W \]

\[ \text{maximize kurtosis } f(w) = \kappa_4(y) = \mathbb{E}[y^4] - 3 \]
\[ \text{with constraint } h(w) = \|w\|^2 - 1 = 0 \]

\[ \text{At optimum } f'(w) + \lambda h'(w) = 0^T \quad (\lambda \text{ Lagrange multiplier}) \]
\[ \Rightarrow 4\mathbb{E}[(w^T z)^3 z] + 2\lambda w = 0 \]

Solve this equation by Newton–Raphson’s method.
The Fast ICA algorithm (Hyvarinen)

Solve: \( F(w) = 4\mathbb{E}[(w^Tz)^3z] + 2\lambda w = 0 \)

Note:

\[ y = w^Tz, \quad ||w|| = 1, \quad z \text{ white} \Rightarrow \mathbb{E}[(w^Tz)^2] = 1 \]

The derivative of \( F \):

\[
F'(w) = 12\mathbb{E}[(w^Tz)^2zz^T] + 2\lambda I \\
\sim 12\mathbb{E}[(w^Tz)^2]\mathbb{E}[zz^T] + 2\lambda I \\
= 12\mathbb{E}[(w^Tz)^2]I + 2\lambda I \\
= 12I + 2\lambda I
\]
The Fast ICA algorithm (Hyvarinen)

The Jacobian matrix becomes diagonal, and can easily be inverted.

\[ w(k + 1) = w(k) - [F'(w(k))]^{-1} F(w(k)) \]

\[ w(k + 1) = w(k) - \frac{4 \mathbb{E}[(w(k)^T z)^3 z] + 2\lambda w(k)}{12 + 2\lambda} \]

\[ (12 + 2\lambda)w(k + 1) = (12 + 2\lambda)w(k) - 4 \mathbb{E}[(w(k)^T z)^3 z] - 2\lambda w(k) \]

\[ -\frac{12 + 2\lambda}{4} w(k + 1) = -3w(k) + \mathbb{E}[(w(k)^T z)^3 z] \]

Therefore,

Let \( w_1 \) be the fix point of:

\[ \tilde{w}(k + 1) = \mathbb{E}[(w(k)^T z)^3 z] - 3w(k) \]

\[ w(k + 1) = \frac{\tilde{w}(k + 1)}{||\tilde{w}(k + 1)||} \]

- Estimate the 2\(^{nd}\) ICA component similarly using the \( w \perp w_1 \) additional constraint... and so on...
Other Nonlinearities

\[
\max_{\mathbf{w}} \mathbb{E} G(\mathbf{w}^T z) \quad \text{such that} \quad \mathbf{w}^T \mathbf{w} = 1
\]

\[
F(\mathbf{w}) = \mathbb{E} z g(\mathbf{w}^T z) - \alpha \mathbf{w} = 0 \in \mathbb{R}^n \quad \text{with} \quad \mathbf{g}(z) = g'(z) \quad g''(y) = 12y^2
\]

\[
\nabla F(\mathbf{w}) = \mathbb{E} [zz^T g'(\mathbf{w}^T z)] - 2\mathbf{I}
\]

\[
\approx \mathbb{E} [zz^T] \mathbb{E} [g'(\mathbf{w}^T z)] - 2\mathbf{I}
\]

\[
= \mathbb{E} [g'(\mathbf{w}^T z)] \mathbf{I} - 2\mathbf{I}
\]

\[
\text{\textsc{scalar}} \quad \text{\textsc{diagonal matrix with identical elements}}
\]
Other Nonlinearities

Newton method:

\[ \mathbf{w}_{l+1} = \mathbf{w}_l - \left[ \nabla F(\mathbf{w}_l) \right]^{-1} F(\mathbf{w}_l) \]

\[ = \mathbf{w}_l - \frac{\mathbf{E}[\mathbf{z} g(\mathbf{w}_l^T \mathbf{z})] - \mathbf{w}_l}{\mathbf{E}[g'(\mathbf{w}_l^T \mathbf{z})]} \]

Algorithm:

\[ \tau_{C \in \mathbb{R}} \mathbf{w}_{l+1} = \mathbf{E}[\mathbf{z} g(\mathbf{w}_l^T \mathbf{z})] - \mathbf{E}[g'(\mathbf{w}_l^T \mathbf{z})] \mathbf{w}_l \]

\[ \mathbf{w}_{l+1} = \frac{\mathbf{w}_l}{\| \mathbf{w}_{l+1} \|} \sqrt{\frac{\mathbf{g}^T(\mathbf{w}_l)}{\mathbf{g}_{-1}(\mathbf{w}_l) = \text{TANH}(\mathbf{A}, \mathbf{u})}} \]

\[ \mathbf{g}_{-1}(\mathbf{u}) = \mathbf{u} \exp(-\mathbf{u}^2) \]
Fast ICA for several units

\[
W = \begin{bmatrix}
    w_1^T \\
    \vdots \\
    w_n^T
\end{bmatrix}
\]

\[WW^T = I\]

We need to prevent different vectors converging to the same maxima.

If we already estimated \( w_1 \ldots w_p \), then update \( w_{p+1} \).

And after that,

\[
w_{p+1} = w_{p+1} - \sum_{j=1}^{p} (w_{p+1}^T w_j) w_j
\]

Option 2:

\[
W = (WW^T)^{-1/2} W
\]

From SVD

\[
WW^T = UDV^T
\]

\[(WW)^{-1/2} = UD^{-1/2} V^T\]