Independent Component Analysis

Xing Yi Liu, Weiyu Huang, Landon Choi

March 10, 2021

Section 1

Introduction

What is ICA?

- A method of dimension reduction used to separate data into independent basis vectors
- Deciphers the latent non-Gaussian variables which make up a multivariate mixed signal
- **Applications of ICA**: Cocktail Party Problem, EEG (brain activity), feature extraction, image processing

Cocktail Party Problem

Figure 1: Cocktail Party

Blind Source Separation

- Given m recordings of mixed sound sources, blind source separation seeks to uncover the underlying individiual sources without any information other than the mixed data.
- Typically, we assume that if there are m recordings of a mixed sound, there are also m independent sound sources.

Graphical Illustration

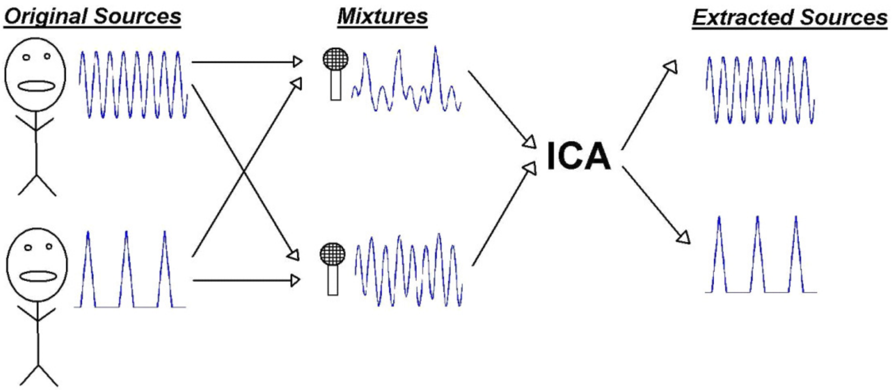

Figure 2: Illustration of ICA, $m = 2$

Section 2

Mathematical Formulation

Notation

Let the observed data be $\{\boldsymbol{x}_i\}_{i=1}^N$, where

$$\boldsymbol{x}_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{im} \end{pmatrix} = \begin{pmatrix} a_{11}s_{i1} + a_{12}s_{i2} + \ldots + a_{1m}s_{im} \\ a_{21}s_{i1} + a_{22}s_{i2} + \ldots + a_{2m}s_{im} \\ \vdots \\ a_{m1}s_{i1} + a_{m2}s_{i2} + \ldots + a_{mm}s_{im} \end{pmatrix} = \mathbf{A}\boldsymbol{s}_i.$$

Then,

$$\mathbf{W}\boldsymbol{x}_i = \boldsymbol{s}_i,$$

where $\mathbf{W} = \mathbf{A}^{-1}$. Our goal is to estimate $\mathbf{W}$ by assuming the distribution of $\boldsymbol{s}_i$, and approximate $\boldsymbol{s}_i$.

Preprocessing

Preprocessing involves centering and whitening the observation data.

Let $\boldsymbol{x} \in \{\boldsymbol{x}_i\}_{i=1}^N$. Let

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x} - \bar{\boldsymbol{x}}$$

represent the centered data. Our goal is to obtain whitened data $\boldsymbol{x}_w$ satisfying

$$\mathbb{E}(\boldsymbol{x}_w \boldsymbol{x}_w^\intercal) = \mathbf{I}.$$

To achieve this, perform eigen-decomposition on the covariance matrix of $\widetilde{\boldsymbol{x}}$:

$$\mathbb{E}(\widetilde{\boldsymbol{x}}\widetilde{\boldsymbol{x}}^\intercal) = \mathbf{VDV}^{-1},$$

where $\mathbf{V}$ is the matrix whose columns are eigenvectors of $\mathbb{E}(\widetilde{\boldsymbol{x}}\widetilde{\boldsymbol{x}}^\intercal)$, and $\mathbf{D}$ is the diagonal matrix of eigenvalues $\lambda_1, \ldots, \lambda_m$.

Preprocessing

The whitened signal can be obtained by

$$\boldsymbol{x}_w = \mathbf{V}\mathbf{D}^{-\frac{1}{2}}\mathbf{V}^\intercal\widetilde{\boldsymbol{x}} = \mathbf{A}_w\boldsymbol{s},$$

where $\mathbf{A}_w = \mathbf{V}\mathbf{D}^{-1/2}\mathbf{V}^\intercal\mathbf{A}$ is orthogonal.

$\mathbf{A}_w$ has only $\frac{m(m-1)}{2}$ degrees of freedom, less than the m^2 parameters of $\mathbf{A}$. Whitening thus allows us to estimate less parameters.

Maximum Likelihood Estimation

Since $\boldsymbol{x}_i = \mathbf{A}\boldsymbol{s}_i$,

$$p(\boldsymbol{x}_i) = \frac{1}{|\det(\mathbf{A})|} p(\boldsymbol{s}_i)$$
$$= |\det(\mathbf{W})| \prod_{j=1}^{m} p_i(\boldsymbol{w}_j^\intercal \boldsymbol{x}_i),$$

where $\boldsymbol{w}_j^\intercal$ is the j-th row (a row vector) of $\mathbf{W} = \mathbf{A}^{-1}$. Then, the likelihood is

$$L(\mathbf{W}) = |\det(\mathbf{W})|^N \prod_{i=1}^{N} \prod_{j=1}^{m} p_i(\boldsymbol{w}_j^\intercal \boldsymbol{x}_i)$$

$$\log\left(L(\mathbf{W})\right) = N \log|\det(\mathbf{W})| + \sum_{i=1}^{N} \sum_{j=1}^{m} \log\left(p_i(\boldsymbol{w}_j^\intercal \boldsymbol{x}_i)\right)$$

$$\frac{1}{N} \log\left(L(\mathbf{W})\right) = \log|\det(\mathbf{W})| + \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{m} \log\left(p_i(\boldsymbol{w}_j^\intercal \boldsymbol{x}_i)\right).$$

Maximum Likelihood Estimation

$$\frac{\partial}{\partial \mathbf{W}} \frac{1}{N} \log\left(L\left(\mathbf{W}\right)\right) = \left(\mathbf{W}^{T}\right)^{-1} + \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{g}(\mathbf{W}\boldsymbol{x}_i)\boldsymbol{x}_i^{\mathsf{T}},$$

where

$$\boldsymbol{g}(\boldsymbol{a}) = \begin{pmatrix} g_1(a_1) \\ g_2(a_2) \\ \vdots \\ g_m(a_m) \end{pmatrix}.$$

The functions $g_i = (\log(p_i))' = \frac{p_i'}{p_i}, i = 1, \ldots, m$ are called **score functions** and depend on the probability distribution p_i of the source s_i.

Bell-Sejnowski Algorithm

Require: $\{\boldsymbol{x}_n\}_{n=1}^{N}, \{p_i(s_i)\}_{i=1}^{m}$

1. Calculate the sample mean $\bar{\boldsymbol{x}}$ of the data and center the data by $\widetilde{\boldsymbol{x}}_i \leftarrow \boldsymbol{x}_i - \bar{\boldsymbol{x}}$ for $i = 1, \ldots, N$.
2. Calculate $\mathbf{V}, \mathbf{D}$ such that $\mathbb{E}(\widetilde{\boldsymbol{x}}_i \widetilde{\boldsymbol{x}}_i^{\intercal}) = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$ is the eigen-decomposition of $\mathbb{E}(\widetilde{\boldsymbol{x}}_i \widetilde{\boldsymbol{x}}_i^{\intercal})$.
3. Preprocess the data by $\boldsymbol{x}_{wi} = \mathbf{V}\mathbf{D}^{-1/2}\mathbf{V}^{\intercal}\widetilde{\boldsymbol{x}}_i$ for $i = 1, \ldots, N$.
4. Initialize $\mathbf{W} \in \mathbf{R}^{m \times m}$
5. Update $\mathbf{W} \leftarrow \mathbf{W} - \eta \left(\left(\mathbf{W}^{T} \right)^{-1} + \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{g}(\mathbf{W}\boldsymbol{x}_{wi})\boldsymbol{x}_{wi}^{\intercal} \right)$, where $\eta > 0$.
6. If stopping criterion is satisfied, proceed to step 7. Otherwise, return to step 5.
7. Estimates for the separated latent variables will be given by $\boldsymbol{y}_i = \mathbf{W}\boldsymbol{x}_i$.

Non-Gaussianity

Kurtosis is a classical measure of non-Gaussianity:

$$\kappa(y) = \mathbb{E}(y^4) - 3(\mathbb{E}(y^2))^2.$$

A more robust measure of non-Gaussianity involves entropy:

$$H(\boldsymbol{y}) = -\int f(\boldsymbol{y}) \log f(\boldsymbol{y}) d\boldsymbol{y}.$$

- Since Gaussian distributions have the most entropy, can compare using entropy
- Brings definition of negentropy and its maximization

Negentropy

$$J(\boldsymbol{y}) = H(\boldsymbol{y}_{\text{Gauss}}) - H(\boldsymbol{y})$$

- $H(\boldsymbol{y}_{\text{Gauss}})$ is the entropy a Gaussian variable with the same covariance as $\boldsymbol{y}$
- But this definition of negentropy is hard to work with
- Approximization is used instead, which is the foundation of FastICA

$$J(y) \propto [\mathbb{E}\{G(y)\} - \mathbb{E}\{G(\nu)\}]^2$$

- $G(y)$ is a nonquadratic function, ν is a Gaussian distribution of mean 0 and unit variance

FastICA

FastICA is a fixed point algorithm that attempts to maximize negentropy by maximizing the term $\mathbb{E}\{G(y)\}$ or $\mathbb{E}\{G(\boldsymbol{w}_i^\intercal x)\}$ where $\boldsymbol{w}_i$ is the i-th row of $\mathbf{W}$.

According to Kuhn-Tucker conditions and by also assuming that $\|\boldsymbol{w}_i\| = 1$ due to whitening earlier, the optima of $\boldsymbol{w}_i$ is where

$$\mathbb{E}\{\boldsymbol{x} g(\boldsymbol{w}_i^\intercal \mathbf{x})\} - \beta \boldsymbol{w}_i = 0$$

where $g = G'$ and $\beta = \mathbb{E}\{\boldsymbol{w}_{i,\text{opt}}^\intercal \boldsymbol{x} g(\boldsymbol{w}_{i,\text{opt}}^\intercal \boldsymbol{x})\}$ is a constant.

FastICA

Therefore the overall algorithm will go as follows.

1. Randomize a $\boldsymbol{w}_i$ vector
2. Calculate $\boldsymbol{w}_i^+ = \mathbb{E}\{\boldsymbol{x}g(\boldsymbol{w}_i^\intercal \boldsymbol{x})\} - \mathbb{E}\{g'(\boldsymbol{w}_i^\intercal \boldsymbol{x})\}$
3. Normalize $\boldsymbol{w}_i \leftarrow \frac{\boldsymbol{w}_i^+}{\|\boldsymbol{w}_i^+\|}$
4. If $\boldsymbol{w}_i$ has not converged then repeat step 2

Note that computing the expectation of $\boldsymbol{x}g(\boldsymbol{w}_i^\intercal \boldsymbol{x})$ and $g'(\boldsymbol{w}_i^\intercal \boldsymbol{x})$ is difficult so it is often easier to compute the sample mean for all the values $\boldsymbol{x}_n$ within the dataset.

Section 3

Experiments

Mixed Signals

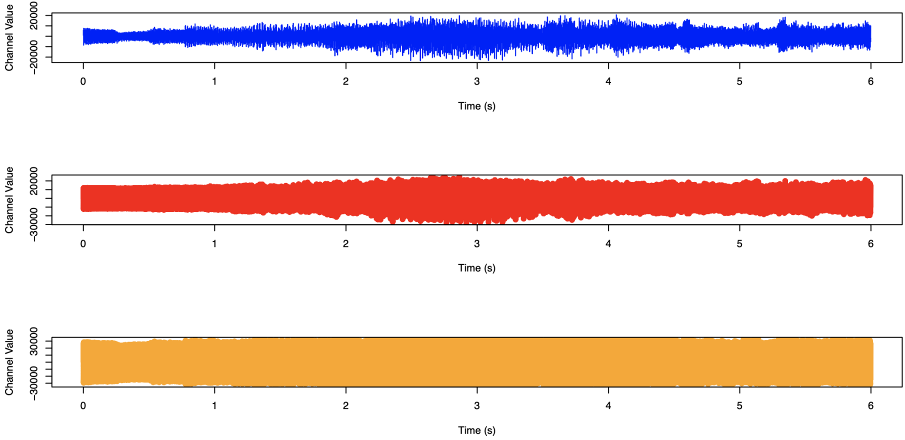

Figure 3: Recordings of Mixed Signals

Separated Sources

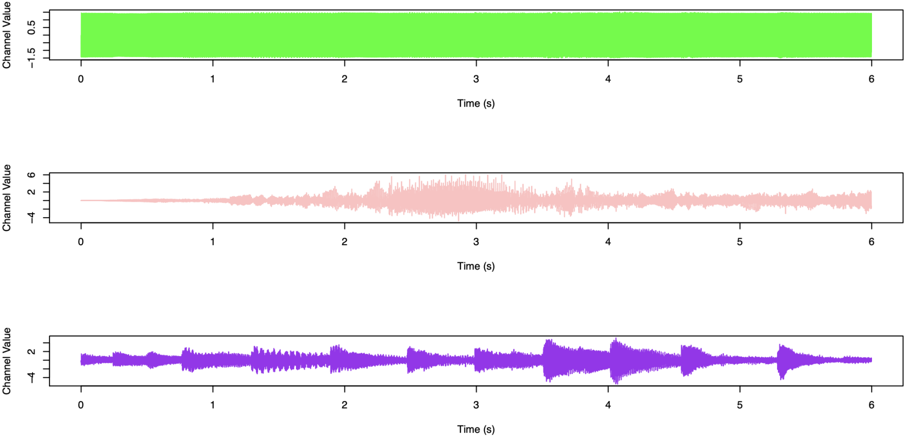

Figure 4: Separated Sources