A brief overview of the ICA algorithms used in stabilized ICA

Nicolas Captier (PhD student at Institut Curie U900)

Abstract

This is a supplementary material for the Stabilized ICA package (sica) [1]. It aims to guide the user through the different algorithms that are included in this package to solve the ICA problem. It provides some mathematical insights about two well-known algorithms : FastICA and infomax with a maximum-likelihood approach. The idea is to clearly state which problem each of these algorithms solves and what are the latent assumptions they make. At the end of this work, a glossary lists the key parameters for the choice of the solving algorithm in the stabilized ICA method.

1 Introduction

1.1 Definition of ICA

To define ICA, we use a statistical "latent variables" model (cf. section 2 [1]). We assume that we observe n linear mixtures $x_1, ..., x_n$ of latent sources $s_1, ..., s_n$:

$$x_j = a_{j1}s_1 + a_{j2}s_2 + ... + a_{jn}s_n \quad \text{for all j} \tag{1}$$

It is convenient to use a vector-matrix notation introducing the observed random vector $\boldsymbol{x} \in \mathbb{R}^n$, the latent random vector $\boldsymbol{s} \in \mathbb{R}^n$ and the unknown mixing matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$:

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s} \tag{2}$$

We can also introduce $\boldsymbol{W}$ the inverse of the mixing matrix $\boldsymbol{A}$ and obtain the latent components with :

$$\boldsymbol{s} = \boldsymbol{W}\boldsymbol{x} \tag{3}$$

Independent component analysis adds three fundamental assumptions to the mixing model (2) :

1. The sources are statistically independent of one another.
2. The mixing matrix is assumed to be invertible.
3. The sources have a non-Gaussian distribution, or more precisely, at most one of them can be a Gaussian signal.

Note 1 : *Here, we used a squared mixing matrix for simplicity. Please note, that this assumption can be relaxed and cases where there are more observations than sources still fall into the scope of ICA.*

Since, in general these assumptions do not necessarily hold, we can reformulate the problem saying that the goal is to find the linear transformation $\boldsymbol{A}$ so that the observations $\boldsymbol{x}$ best satisfy the generative model describe above.[2] We could also say that we look for the linear transformation $\boldsymbol{W}$ that maximizes the statistical independence of the sources $\boldsymbol{s}$ defined by (3).

1.2 Minimization of mutual information

Mutual information being the natural information-theoretic measure of the independence of random variables, we can use it as a criterion for finding the ICA transform :

$$\boldsymbol{W}^* = \underset{\boldsymbol{W} \in GL_n(\mathbb{R})}{argmin}\ I(s_1, ..., s_n) \quad (\text{with } s_i = \boldsymbol{W_i}\boldsymbol{x}) \tag{4}$$

[1] https://github.com/ncaptier/Stabilized_ICA

[2] We will see that the terms "best satisfy" can be mathematically translated by the minimization of the Kullback-Leibler divergence

In the following, we will describe several popular approaches for solving the ICA problem and show how they relate to the mutual information minimization problem. First we will tackle the information-maximization principle and show that it can be solved with a maximum-likelihood strategy. Then, we will shed light on the well-known FastICA method and show that it relates to the mutual information minimization under the constraint of uncorrelatedness.

2 Information-maximization for ICA

2.1 The original idea

To solve the ICA problem, Bell and Sejnowski proposed to apply the infomax approach [2]. This approach was originally developed as an unsupervised strategy to learn a "good" representation of an input $\boldsymbol{x}$ with a single-layer feed forward neural network with non-linear activation (Figure 1).

$$\begin{cases} \boldsymbol{y} = \boldsymbol{g}(\boldsymbol{W}\boldsymbol{x}) \\ \boldsymbol{W} \in \mathbb{R}^{n\times n}, \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{y} \in \mathbb{R}^n, \boldsymbol{g} : \mathbb{R}^n \longrightarrow \mathbb{R}^n \end{cases} \tag{5}$$

A "good" representation is defined using an information theoretic criteria : $\boldsymbol{W}^*$ maximizes the mutual information between the output $\boldsymbol{y}$ and the input $\boldsymbol{x}$.

$$\boldsymbol{W}^* = \underset{\boldsymbol{W}\in\mathbb{R}^{n\times n}}{argmax}\, I(\boldsymbol{x},\boldsymbol{y}) = \underset{\boldsymbol{W}\in\mathbb{R}^{n\times n}}{argmax}\, H(\boldsymbol{y}) - H(\boldsymbol{y}|\boldsymbol{x}) \quad (\text{with } \boldsymbol{y} = \boldsymbol{g}(\boldsymbol{W}\boldsymbol{x})) \tag{6}$$

They showed that for their simple case $H(\boldsymbol{y}|\boldsymbol{x})$ did not depend on the weight matrix $\boldsymbol{W}$ and thus the infomax approach boiled down to the maximization of the entropy of the output. Finally, they developed an online stochastic gradient ascent algorithm to solve such a problem.

$$\boldsymbol{W}^* = \underset{\boldsymbol{W}\in\mathbb{R}^{n\times n}}{argmax}\, \Phi_I(\boldsymbol{W}) \quad (\text{with } \Phi_I(\boldsymbol{W}) = H(\boldsymbol{g}(\boldsymbol{W}\boldsymbol{x}))) \tag{7}$$

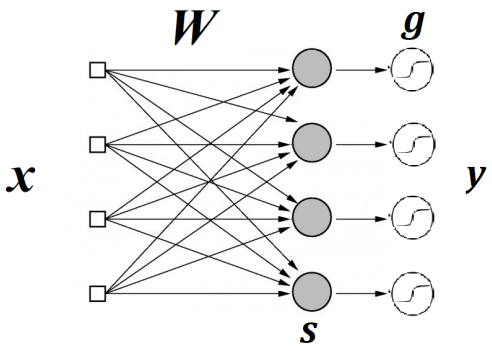

Figure 1: Single-layer feed forward neural network with weight matrix $\boldsymbol{W} \in \mathbb{R}^{n\times n}$ and a continuous monotonic activation function $\boldsymbol{g} : \mathbb{R}^n \longrightarrow \mathbb{R}^n$. For simplicity we did not consider any bias.

In [2], Bell and Sejnowski linked the representation problem described above with ICA and showed that we could apply the same strategy to solve it. Indeed, looking at Figure 1, the ICA model (3) fits perfectly in our neural network : an additional non-linear transformation of the sources $\boldsymbol{s}$ is simply added to the model.

$$\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{s}) = \boldsymbol{g}(\boldsymbol{W}\boldsymbol{x}) \tag{8}$$

The question that remains is then : under which conditions and for which choice of the non-linearity $\boldsymbol{g}$ does the infomax problem given by (7) solve ICA and ensure maximally independent sources $\boldsymbol{s}$?

Bell and Sejnowski recommended to use scalar sigmoïds $g_1, ..., g_n$ ($\boldsymbol{g} = (g_1, ..., g_n)^T$) that map the real line to the interval $(0, 1)$. With this choice, g_i can be seen as the cumulative distribution function (c.d.f) of some p.d.f f_i :

$$g_i(u) = \int_{-\infty}^{u} f_i(v)dv \tag{9}$$

We can then easily show that, for such non-linearity $\boldsymbol{g}$, maximizing the entropy (7) boils down to minimize the mutual information of the sources $s_1, ..., s_n$ (assuming that they are distributed according to $f_1, ..., f_n$) [3].

$$\Phi_I(\boldsymbol{W}) = -KL(\boldsymbol{W}\boldsymbol{x}||\tilde{\boldsymbol{s}}) = -I(s_1, ..., s_n) \quad (\text{where } \tilde{\boldsymbol{s}} \sim f_1 \times ... \times f_n) \tag{10}$$

In other words, solving the infomax problem for the network defined by (8) and (9) allows us to maximize the independence of the sources and get as close as possible (in the sense of KL divergence) to a target distribution $\tilde{\boldsymbol{s}} \sim f_1 \times ... \times f_n$.

2.2 Maximum-likelihood formulation

In [3] Jean-François Cardoso proposed a maximum-likelihood formulation for the ICA problem and showed that it was equivalent to the infomax approach described above.

We denote $\boldsymbol{X} = \left[\boldsymbol{x}^1, ..., \boldsymbol{x}^T\right] \in \mathbb{R}^{n \times T}$ T observations supposed to come from the generative model (2). Introducing the matrix of latent sources $\boldsymbol{S} \in \mathbb{R}^{n \times T}$ we obtain the matrix-formulation :

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S} \quad (\text{with } \boldsymbol{A} = \boldsymbol{W}^{-1} \in \mathbb{R}^{n \times n}) \tag{11}$$

Similarly to the precedent approach, we assume the marginal densities of the independent sources f_i to be known and, under the independence hypothesis, we write the negative log-likelihood associated with the observations (11) [4]:

$$\mathcal{L}(\boldsymbol{W}) = -\log\left(|det\boldsymbol{W}|\right) - \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n}\log\left(f_i\left(\boldsymbol{W}_i\boldsymbol{x}^t\right)\right) \tag{12}$$

The link between the log-likelihood and the Kullback Leibler divergence is well known and we can thus easily show that maximizing $\mathcal{L}(\boldsymbol{W})$ boils down to maximize and empirical estimate of (10) [5].

In a nutshell, both original and maximum-likelihood formulations for the infomax approach for ICA are to the minimization of the mutual information of sources (4). The notable difference is that they both assumed the marginal densities f_i to be known. The choice of such densities is a vast question, widely discussed in the literature. In [2] two main strategies were proposed : either integrate the estimation of the marginal densities in the learning process using "flexible" sigmoïds (cf. section 2.5 [2]) or use a common model for all the marginals with $-\log(f_i(.)) = 2\log(\cosh(./2)) + cste$.

Note 2 : *The algorithms we are going to briefly present in the following paragraph only propose the simplest method for the building of the marginals f_i : a single model is chosen for all the marginals (e.g* $2\log(\cosh(./2)) + cste$*).*

The infomax ICA approach thus deeply depends on the choice of the source model (i.e choice and/or estimation for f_i). For a model that captures the "true" densities of the sources, the local convergence of the infomax method is guaranteed [6]. However, it is unclear what happens with a wrong source model. Many articles stated that this approach is fairly robust to misspecifying source distributions and that it should be enough to estimate whether the sources are sub- or super-Gaussian.

2.3 Fast algorithms for the infomax approach

Many algorithms have been proposed to solve the infomax ICA problem, including the stochastic gradient ascent developed by Bekk and Sejnowski in [2]. Since we would like to integrate such an algorithm in our stabilized ICA method (i.e multiple runs with random initialisations), we need a fast and accurate one.

In [4], Pierre Ablin *et al.* developed a very promising second-order scheme to solve the infomax problem through the minimization of the negative log-likelihood (12):

$$\boldsymbol{W}^* = \underset{\boldsymbol{W}}{argmin}\,\mathcal{L}(\boldsymbol{W}; f) \quad (f \text{ the marginal density of a source}) \tag{13}$$

[3] Please refer to article [3] for a detailed proof of (10)
[4] We used the change of variables $f_{\boldsymbol{x}}(\boldsymbol{u}) = |det\boldsymbol{W}|f_{\boldsymbol{s}}(\boldsymbol{W}\boldsymbol{u})$
[5] Please refer to article [3] for more details
[6] The criterion being non-convex, we only have results for local convergence

Note 3 : *We purposely wrote $\mathcal{L}(\boldsymbol{W}; f)$ instead of $\mathcal{L}(\boldsymbol{W})$ to insist on the fact that this algorithm requires the choice of the marginal density f (the same for all the sources).*

Note 4 : *A python implementation can be found in a package called picard* [7] *(Preconditioned ICA for Real Data).*

Roughly, this optimization scheme corresponds to a quasi-Newton strategy for which the hessian matrix is accurately and efficiently estimated using the past steps (L-BFGS). A regularization step is added to avoid not positive definite hessian matrices (it could happen since the objective is not convex) as well as a line-search to optimize the size of each step.

Finally, let us note that this algorithm is claimed to achieve a kind of convergence (in the sense of gradient decrease) even when it is applied to real data for which ICA assumptions do not hold exactly. For us, this is particularly interesting since we aim to apply our stabilized ICA method to real biological data for which the ICA model is not expected to hold exactly.

2.3.1 Extended infomax

The choice of a single model for f in (13) means that we assume all the sources to have the same distribution (or at least the same "kind" of distribution). Thus we can deal with cases where all sources have a super-Gaussian distribution, cases where they all have a sub-Gaussian distribution but not cases that mix both.

In [5], Lee *et al.* proposed an extension of the infomax approach in order to separate mixtures of super- and sub-Gaussian sources while preserving computational simplicity. They showed that using two specific models for f, one sub-Gaussian and the other super-Gaussian, we get two very similar expressions for the gradient (and the hessian) of our objective function $\mathcal{L}$ that we want to optimize :

$$\begin{cases} f_{sub}(u) = \frac{1}{2}\left(\mathcal{N}(-1,1) + \mathcal{N}(1,1)\right) & \implies \psi^{-}(u) = u - \tanh(u) \\ f_{sup}(u) \propto \phi(u)\dfrac{1}{\cosh(u)} & \implies \psi^{+}(u) = u + \tanh(u) \end{cases} \quad (\text{with } \phi \sim \mathcal{N}(0,1)) \tag{14}$$

where ψ is a non-linearity that appears in the gradient of the objective function [8].

To be more specific, at each step of the optimization algorithm, a non-linearity ψ_i appears for each source i in the gradient of the objective and we need a rule to choose between ψ_i^+ and ψ_i^- (i.e between super- and sub-Gaussian models). In [5], a simple rule is derived from the fact that the sources should constitute a local minimum for the objective function [9].

2.3.2 Orthogonal constraints

In [7], Pierre Ablin *et al.* proposed an extension of their approach constraining the sources to be uncorrelated (i.e $\frac{1}{T}SS^T = I_N$). Indeed, this resonates with the strategy of many ICA algorithms which consists in performing a pre-whitening step before searching for the rotation that maximizes the independence of the sources.

To do so, they adapted the minimization of the negative non-Gaussian ICA likelihood (13) by restraining the search space :

$$\boldsymbol{O^*} = \underset{\boldsymbol{O} \in \mathcal{O}(n)}{argmin}\, ; \mathcal{L}(\boldsymbol{O W_0}; f) \qquad \boldsymbol{W^*} = \boldsymbol{O^* W_0}\,,\ \boldsymbol{W_0} = \left(\frac{1}{T}\boldsymbol{X X}^T\right)^{-\frac{1}{2}} \tag{15}$$

where $\boldsymbol{W_0}$ corresponds to the initial whitening step.

They used the same quasi-Newton-like solver as the one developed in [4] and adapted the gradient and the hessian matrix in order to ensure that each iteration remained in the orthogonal group.

Note 5 : *in [7] they showed that their algorithm actually mixed orthogonal constraints and the extended infomax strategy. They also showed that this mix converged towards the fixed points of FastICA, yet faster on real data. They developed a very interesting discussion about the connections between their approach and FastICA fixed point strategy (see section 4. [7]).*

[7] https://pierreablin.github.io/picard/
[8] Please refer to the articles [2] and [5] for more details
[9] Please refer to the article [5] and [6] for more details

3 Fast ICA

Work in progress...

4 Glossary for the stabilized ICA method

4.1 Algorithm parameter

infomax : see section 2.3 (picard algorithm with parameters ortho = False and extended = False)

infomax_ext : see section 2.3.1 (picard algorithm with parameters ortho = False and extended = True)

infomax_orth : see section 2.3.2 (picard algorithm with parameters ortho = True and extended = False)

fastica_picard : see Note 5 (picard algorithm with parameters ortho = True and extended = Truee)

fastica_par : see section 3 (sklearn.FastICA with parameter algorithm = 'parallel')

fastica_def : see section 3 (sklearn.FastICA with parameter algorithm = 'deflation')

4.2 Fun parameter

4.2.1 Using picard algorithm

The fun parameter defines the density model of the sources. When a string is used it refers to a non-linearity score function ψ that appears in the expression of the gradient and the hessian matrix :

$$\psi(u) = -\frac{f'(u)}{f(u)} \quad \Longrightarrow \quad f(u) \propto \exp\left(-\int_a^u \psi(v)dv\right) \tag{16}$$

tanh : $\psi(u) = \tanh(u) \quad \Longrightarrow \quad f(u) \propto \frac{1}{\cosh(u)}$

cube : $\psi(u) = u^3 \quad \Longrightarrow \quad f(u) \propto \exp\left(-\frac{u^4}{4}\right)$

exp : $\psi(u) = u\exp\left(-\frac{u^2}{2}\right) \quad \Longrightarrow \quad f(u) \propto$ **?**

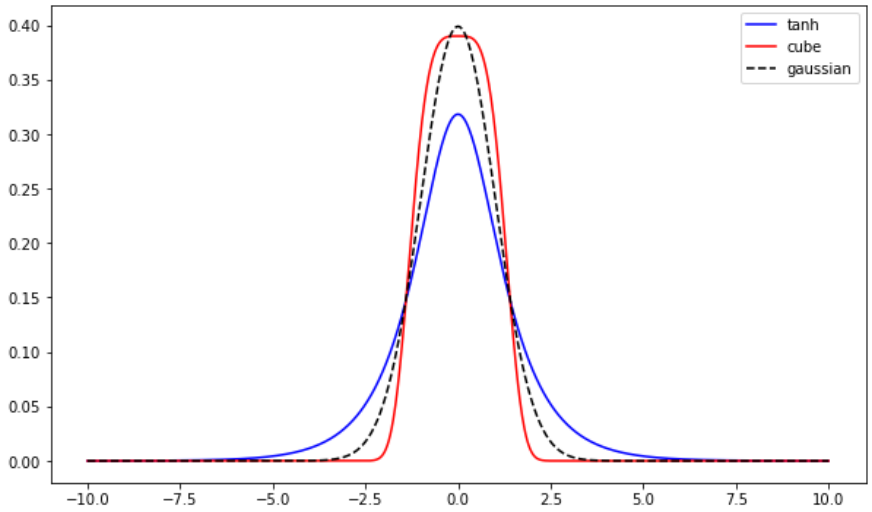

Figure 2: tanh and cube models compared to the Gaussian model.

Note 6 : *The "exp" model does not seem to lead to a proper density function, we do not have any justification for its presence here. The "cube" model leads to a sub-Gaussian distribution while the "tanh" model leads to a super-Gaussian distribution.*

Finally, let us precise that for the case of extended infomax (i.e "infomax_ext" and "fastica_picard" algorithms) the algorithm is well defined only for the "tanh" model. Besides this model is actually quite different from the one defined above since the score function is $\psi(u) = u + \tanh(u)$ (see section 2.3.1).

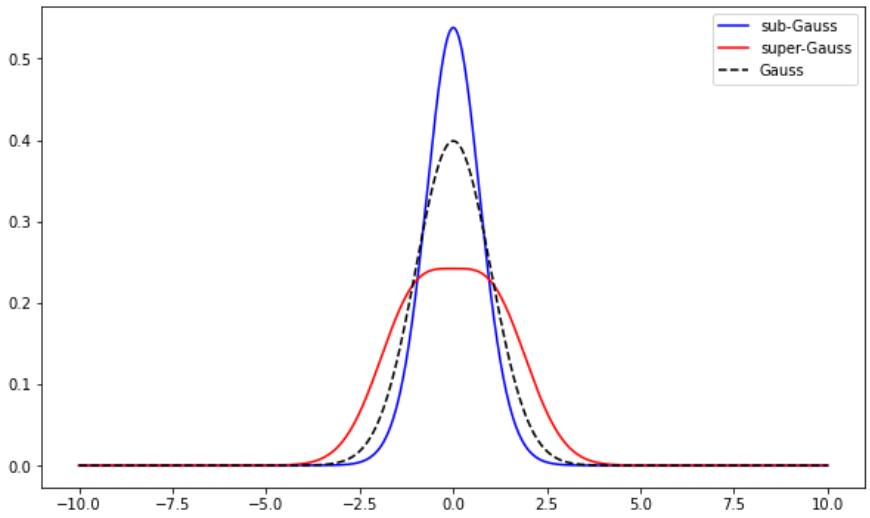

Figure 3: tanh model in the case of extended infomax compared to the Gaussian model. Here tanh gives rise to two source distribution : one sub-Gaussian and the other super-Gaussian.

4.2.2 Using FastICA algorithm

Work in progress...

logcosh :

cube :

exp :

References

[1] A. Hyvärinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Netw.*, 13(4–5):411–430, May 2000.

[2] Anthony J. Bell and Terrence J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Comput.*, 7(6):1129–1159, November 1995.

[3] J. F. Cardoso. Infomax and maximum likelihood for blind source separation. *IEEE Signal Processing Letters*, 4(4):112–114, 1997.

[4] P. Ablin, J. Cardoso, and A. Gramfort. Faster independent component analysis by preconditioning with hessian approximations. *IEEE Transactions on Signal Processing*, 66(15):4040–4049, 2018.

[5] Te-Won Lee, Mark Girolami, and Terrence Sejnowski. Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources. *Neural Computation*, 11:417–441, 02 1999.

[6] Tianping Chen. Stability analysis of learning algorithm for blind source separation. *Neural Networks*, 10:1345–1351, 08 1997.

[7] P. Ablin, J. Cardoso, and A. Gramfort. Faster ica under orthogonal constraint. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4464–4468, 2018.