Probabilistic Graphical Model: Independent Component Analysis

Sofiane Horache - Aloïs Pourchot - Adil Rhoulam

April 3, 2018

Abstract

We give a short introduction to the ICA framework. We start by describing the ICA model and discuss its applications. We then dig into the theory to highlight the links with information theory, and explain why we can't use Gaussian variables. Finally we introduce a hyper-parameter free algorithm that solves the ICA problem quickly and efficiently: the FastICA algorithm. We then try to apply the ICA on a clustering problem.

1 The ICA Model

Given a random variable $\mathbf{x} = (x_1, ..., x_n)^T \in \mathbf{R}^n$, the ICA model aims at finding another random variable $\mathbf{s} = (s_1, ..., s_k)^T \in \mathbf{R}^k$, and a matrix $\mathbf{A} \in \mathbf{R}^{n \times k}$ with $rank(\mathbf{A}) = k$ such that:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \qquad \text{and} \qquad \forall i \neq j, \quad s_i \perp s_j \tag{1}$$

Where no more than one s_i is gaussian. $\mathbf{A}$ is called the mixing matrix. The idea behind ICA is to find, using a linear transformation, a new representation of observed data such that in the new space, the components of the vectors are statistically independent. It is used in many problems such as source separation, dimension reduction, or feature extraction. Here is a short example: given two recordings (x_1, x_2) (where x_i is a temporal signal that we can think of as observations a random variable) of two person talking together the aim is to extract each individual speech s_1 and s_2.

2 Information Theory

The ICA Model has strong ties with information theory. Let's consider for now that A is a non-singular square matrix (i.e. $n = k$). If $\mathbf{W} = \mathbf{A}^{-1}$ was known, then the ICA would be completely determined by $\mathbf{s} = \mathbf{W}\mathbf{x}$. The challenge is to estimate a $\mathbf{W}$ that maximizes the independence of the components of $\mathbf{s}$. A first idea is to use mutual information as a measure of the s_i's independence. Recall that mutual information is given by:

$$I(s_1, s_2, .., s_n) = \mathrm{E}_S[\log \frac{p(\mathbf{s})}{\prod p(s_i)}] \tag{2}$$

The larger the mutual information, the higher the mutual dependence between the s_i's. If we knew the distribution of $\mathbf{s}$ we could deduce the distribution of $\mathbf{x}$ in terms of $\mathbf{W}$, and use stochastic gradient descent to solve this optimization problem in $\mathbf{W}$. However, in practice we rarely have such information. Thus, other approaches were developed. The one leading to the FastICA algorithm consists in noticing that this mutual information reduction paradigm is equivalent to maximizing the "nongaussianity" of $\mathbf{s}$.

3 Non-Gaussianity

We now explain why the sources described in the ICA model [1] cannot be Gaussian random variables, and explore how to measure the non-gaussianity of random variables. Here are some assumptions that we're going to use from now on:

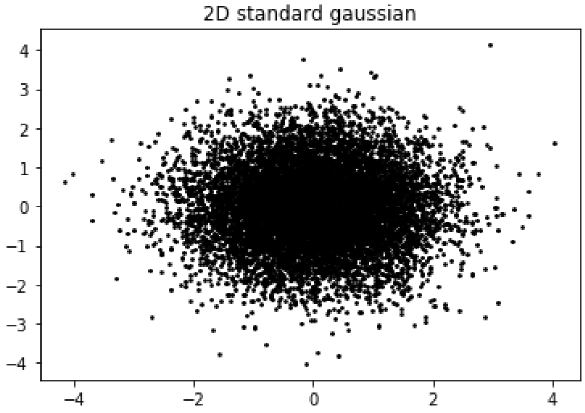


Figure 1: 2D Standard Gaussian distribution

- We consider that the mixing matrix is square and nonsingular, we note $\mathbf{W} = \mathbf{A}^{-1}$.
- We consider that $\mathbf{x}$ is centered, or equivalently consider $\mathbf{x}$ - E($\mathbf{x}$). Note that this also implies that the independent components are centered, since $\mathbf{s} = \mathbf{W}\mathbf{x}$
- We consider that the energies of the independent components are normalized to avoid variance ambiguities and large magnitudes. Since they are also independent, we could write E[$\mathbf{s}\mathbf{s}^T$] $=$ $\mathbf{I}$

3.1 Illustration of non-gaussianity

Suppose that $\mathbf{s} \sim \mathcal{N}(0, \mathbf{I})$ and that the mixing matrix A is orthogonal. We can show that $\mathbf{x} \sim \mathcal{N}(0, \mathbf{A}\mathbf{A}^T = \mathbf{I})$ because $(s_i)_{i=1...,n}$ are independent Gaussian sources. We consider in Figure[1] an example of two variables. We can see that it doesn't contain any information on the direction of the columns of $\mathbf{A}$ which means that if we replace $\mathbf{A}$ by any other orthogonal mixing matrix we will get the same results. This makes $\mathbf{A}$ impossible to estimate, illustrating why ICA won't work when the components are Gaussians.

3.2 ICA principle

The Central Limit Theorem (CLT) tells us that the distribution of a sum of i.i.d. random variables tends to a Gaussian distribution. Let $y = \mathbf{w}^T\mathbf{x}$ where $\mathbf{w}$ is unknown. If $\mathbf{w}$ was a column of $\mathbf{W}$ then y would be one of the independent components s_i. Let $\mathbf{z} = \mathbf{A}^T\mathbf{w}$, then $y = \mathbf{z}^T\mathbf{s}$, CLT tells us that y is least Gaussian when it's only to one non-zero component. This means that only one element of $\mathbf{z}$ is non zero. Therefore, maximizing (with respect to $\mathbf{x}$) the nongaussianity of y gives a vector $\mathbf{z}$ described as previously. This means that $\mathbf{w}^T\mathbf{x} = \mathbf{z}^T\mathbf{s}$ is equal to one of the independent components. Maximizing the nongaussianity (with respect to $\mathbf{w}$) is thus an optimization problem with 2n local maxima, two for each independent component (s_i and -s_i because we have considered that they have a normalized energy). CLT has given us a general idea of how to solve our ICA problem. Now, we introduce measures of nongaussianity for random variables.

3.3 Quantitative measures of non-Gaussianity

3.3.1 Kurtosis

Kurtosis is a classical measure of nongaussianity, it's defined as:

$$k(u) = E(u^4) - 3E(u^2)^2 = E(u^4) - 3, \tag{3}$$

for any random variable u centered with unit variance. The kurtosis of a Gaussian distribution is null and it's a linear operator for independent random variables and verifies k$(\alpha x_1) = \alpha^4$k(x_1), The kurtosis can be either positive or negative and nongaussianity is measured with the absolute value of the kurtosis. Let's illustrate this on a simple case, with n=2 as done before. For simplicity, let's consider that y has a unit variance, then with the same notations as before, we get: E(y^2) $= z_1^2 + z_2^2 = 1$ and k(y) $= z_1^4$k(s_1)$+z_2^4$k(s_2). The optimization problem of nongaussianity becomes max $|z_1^4$k(s_1)$+z_2^4$k(s_2)$|$ constrained on the unit circle $z_1^2 + z_2^2 = 1$. The resolution of this problem gives $\mathbf{z} \in \{(1,0),(-1,0),(0,-1),(0,1)\}$ which is same result that we stated earlier. It's worth to note that the kurtosis is null for all the Gaussian variables but the reciprocal is not always true. It's worth to note that if the kurtosis is null for a random variable, this doesn't aimply in general that's a Gaussian distribution. Moreover, kurtosis is very sensitive to outliers when we estimate it, this make it not a robust measure of nongaussianity and other measures have shown better performances.

3.3.2 Negentropy

The negentropy of a random variable y is given by:

$$J(y) = H(y_{gaus}) - H(y), \tag{4}$$

Where H is the entropy and y_{gaus} is a Gaussian random variable with the same covariance matrix as y. The entropy measures the "randomness" of a random variable. The larger is the entropy the more "random" is the random variable. It can be shown that the Gaussian distribution has the largest entropy for a fixed variance. We deduce that the negentropy is always positive and the larger it is, the further the distribution of y is from a Gaussian. Negentropy is well justified by statistical theory since it's defined as a difference of two entropy functions. The problem in using negentropy is the fact that is computationally difficult and expensive. Therefore, it's reasonable to use some negentropy approximations.

3.3.3 Negentropy approximations

A classical method to approximate the negentropy uses the higher order moments and gives:

$$J(y) \approx \frac{1}{12} E(y^3)^2 + \frac{1}{48} k(y)^2, \tag{5}$$

where we have assumed that y is centered and with unit variance. Despite its simplicity, this approximation depends on the kurtosis approximation and thus has its drawbacks, in particular it suffers from outliers. Robust approximations were developed, these approximations were generally based on the maximum-entropy principle. They have the below general formulation:

$$J(y) \approx \sum_{i=1}^{p} k_i \big[E\{G_i(y)\} - E\{G_i(y_{gaus})\}\big], \tag{6}$$

where k_i are positive constants, y_{gaus} is a standard Gaussian variable, y is always considered centered with unit variance and the G_i are some non-quadratic functions. When we use only one non quadratic function G, we get the simpler expression:

$$J(y) \sim \big[E\{G(y)\} - E\{G(y_{gauss})\}\big] \tag{7}$$

Empirical results show that u → $\frac{1}{a}$log cosh(au) and u → -exp(-$u^2/2$) where $1 \leq$ a ≤ 2 give good approximations. Those approximations have shown during experiences a very good compromise between their robustness to outliers that kurtosis suffers from and their fastness in term of computations unlike the negentropy.

4 FastICA Algorithm

We now describe the FastICA algorithm.

4.1 Optimization Problem

We've seen in the previous section that we can approximate the negentropy of a random variable y, by computing the quantity:

$$J(y) \sim c[\mathbb{E}\{G(y)\} - \mathbb{E}\{G(y_{\text{gauss}})\}]^2, \tag{8}$$

This approximation gives an objective function for estimating the ICA transform. To find one independent component we indeed want to maximize the quantity:

$$J_G(\mathbf{w}) = \mathbb{E}\{G(\mathbf{w}^T\mathbf{x})\} - \mathbb{E}\{G(y_{gaus})\}^2, \tag{9}$$

where $\mathbf{w}$ is a vector of dimension m constrained such that $||\mathbf{w}||^2 = 1$. To get the whole matrix W, we add the constraint that:

$$\forall i, j, \quad \mathbb{E}\{({w_i}^T x)({w_j}^T x)\} = \delta_{i,j} \tag{10}$$

4.2 Fixed-point iterations

To optimize this objective function we write the Lagrangian:

$$L(\mathbf{w}, \beta) = \mathbb{E}\{G(\mathbf{w}^T\mathbf{x})\} + \beta(1 - ||w||^2) \tag{11}$$

We compute the gradient of our Lagrangian, we get :

$$\nabla L(\mathbf{w}, \beta) = \mathbb{E}\{\mathbf{x}G'(\mathbf{w}^T\mathbf{x})\} - \beta\mathbf{w} \tag{12}$$

This equation can be solved numerically using the Newton method. We can compute the Hessian of the Lagrangian. We obtain :

$$\nabla^2 L(\mathbf{w}, \beta) = \mathbb{E}\{\mathbf{x}\mathbf{x}^T G''(\mathbf{w}^T\mathbf{x})\} - \beta\mathbf{I} \tag{13}$$

The problem is it is too complicated to invert the Hessian. Therefore we approximate the Hessian by :

$$\nabla^2 L(\mathbf{w}, \beta) \simeq \mathbb{E}\{\mathbf{x}\mathbf{x}^T\}\mathbb{E}\{G''(\mathbf{w}^T\mathbf{x})\} - \beta\mathbf{I} = \mathbb{E}\{G''(\mathbf{w}^T\mathbf{x})\}\mathbf{I} - \beta\mathbf{I} \tag{14}$$

This matrix is diagonal so it is easy to invert Finally we get the algorithm to optimize the approximation of the negentropy:

$$\mathbf{w}_{n+1} = \frac{\mathbf{w}_n(\mathbb{E}\{G''(\mathbf{w}_n^T\mathbf{x})) - \mathbb{E}\{\mathbf{x}G'(\mathbf{w}_n^T\mathbf{x})\}}{\mathbb{E}\{G''(\mathbf{w}_n^T\mathbf{x}) - \beta\}} \tag{15}$$

as β just appears in the numerator, we just compute at each step the numerator and we normalize then the result. We can notice that the objective function is not convex. So it will be a local minimum. In practice we have to restart the experiment a few time to have a proper solution.

4.3 Properties of FastICA

The FastICA algorithm has some very neat properties that we list now:

- The convergence of the algorithm is cubic under the assumptions of the ICA model. This means that the convergence is very fast, especially compared to other gradient based approach.
- The algorithm is hyper-parameter free, there is no step-size to choose.
- The algorithm doesn't require any prior knowledge about the probability density functions.

5 Experiment on hyper-spectral images

For this experiment, we want to do dimensionality reduction in order to perform segmentation on hyper-spectral image. hyper-spectral image are image with many bands(for this example the number of band is 198). We can see in Figure 2 a band of the the hyper-spectral Jasper Ridge(an image from a biological preserve in California owned by Stanford university). What is interesting with hyper-spectral image is that the response varies according to the frequency of the wave(Radar or visible wave). Therefore,

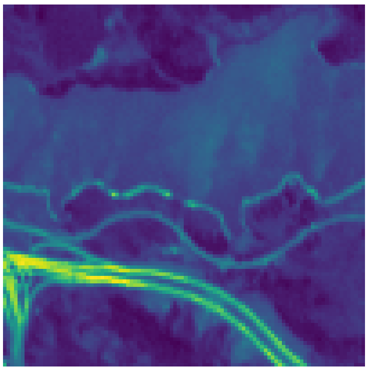


Figure 2: Example of the 5th band of a hyper-spectral image

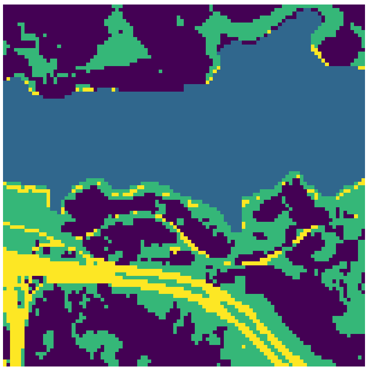


Figure 3: segmentation in four classes of the image(Ground Truth)

we can segmentate image using ICA then the KMeans algorithm. For this image, we have the ground truth(available here 3) to compare quantitatively the result. To formalize this problem, we have X the collection of pixels(let say N pixels) of our image each pixel contains d bands. We want to represent $\mathbf{x}$ as :

$$\mathbf{x} = \mathbf{A}\mathbf{s}$$

where $\mathbf{A}$ is a $N \times c$ matrix and $\mathbf{s}$ is a $c \times d$ matrix. the rows of $\mathbf{s}$ are mutually independent. And c is the number of component. So the first step to segmentate the image is to represent the data with A. we will reduce the dimension of $\mathbf{x}$. For this example we can take $c = 4$. We can visualize in Figure 4 the mixing matrix found by the ICA. This helps us to interpret the image because we can see that the road and the water emit different responses(see Figure 7). If we now apply Kmeans, we can perform a rather precise segmentation(see Figure 5). However we still have some mistakes because especially around the lake. If we compare with PCA, we can see that ICA outperforms ICA(see Figure 6). Therefore by analyzing the independent components of an image, we can learn the structure of the image.

6 Conclusion

In this report, we briefly introduced the ICA model and its links to gaussianity and experimented with the FastICA algorithm.

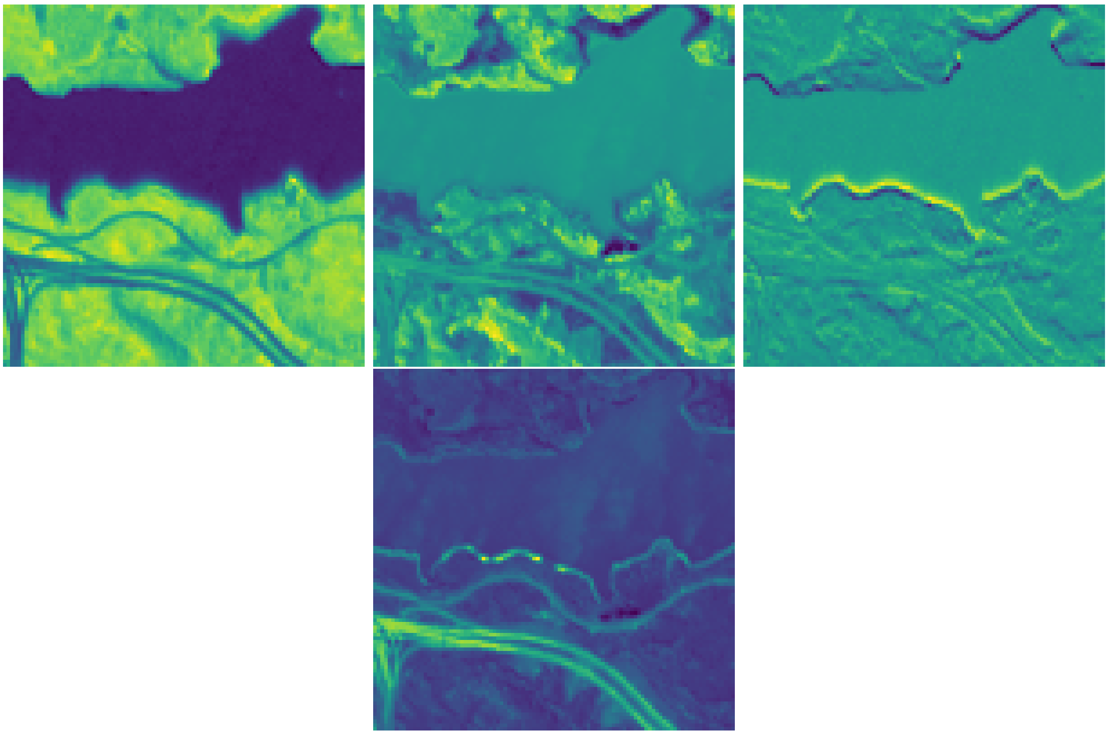
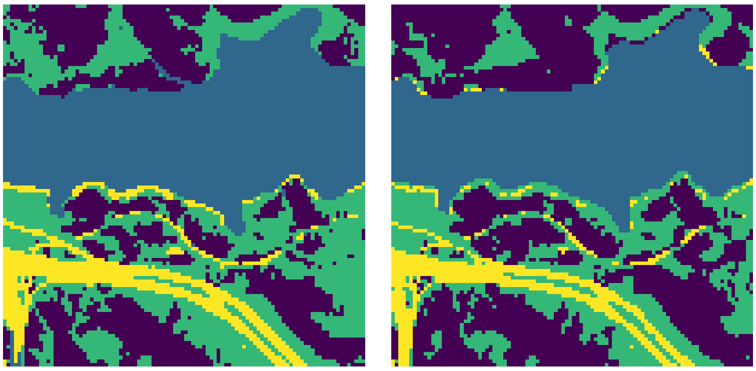


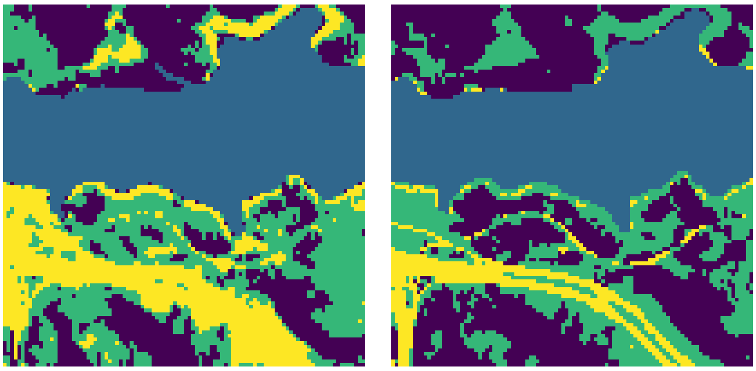
Figure 4: the column of the mixing matrix found by the ICA algorithm



(a) segmentation with 89% of accuracy

(b) ground truth

Figure 5: segmentation of the image with Kmeans+ICA



(a) segmentation with 72% of accuracy

(b) ground truth

Figure 6: segmentation of the image after Kmeans+PCA

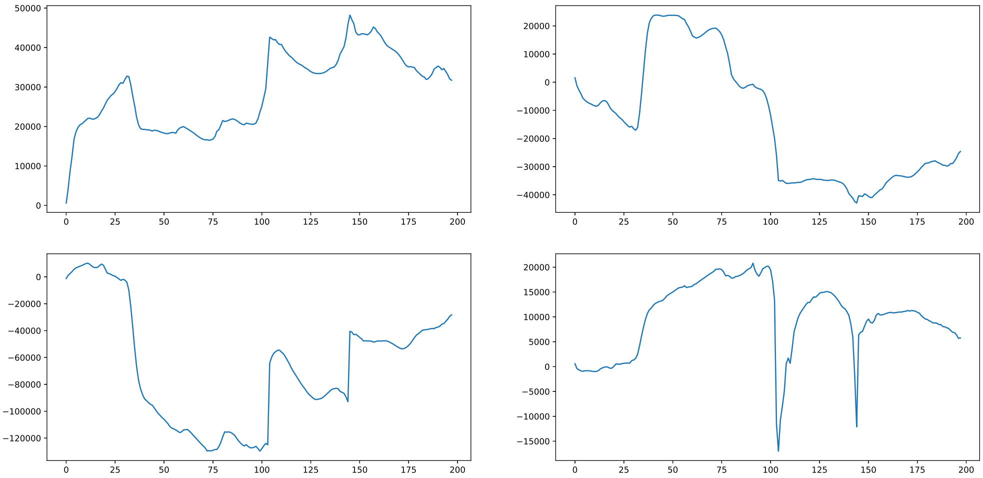


Figure 7: the four source separates by the ICA algorithm