Independant Componant Analysis

Nathan de Lara, Florian Tilquin et Vincent Vidal
Master Mathématiques, Vision et Apprentissage

Problem statement

Let $x \in \mathbb{R}^p$ be some random variables whose components correspond to different mix of some primitive sources $s_i \in \mathbb{R}^n$. The aim is to retrieve an estimation y of every source s_i, given only x. We note A the mixing matrix and W the separation matrix such that:

$$x = As \quad \text{and} \quad y = Wx. \tag{1}$$

In order to retrieve the sources, we suppose that:

- the sources are independents
- the mix is linear and instantaneous
- at most one source has a Gaussian distribution.

Measure of independance

We want to find W that maximises the independence of $y = Wx$.
The information theory gives us quantity to work with, based on the Kullback-Leibler divergence:

$$\mathrm{K}(P \parallel Q) = \int_{\mathbb{R}^n} P(x) \log \frac{P(x)}{Q(x)} \mathrm{d}x. \tag{2}$$

With $\mathcal{G}$ the gaussian distribution manifold and $\mathcal{P}$ the product one, we define

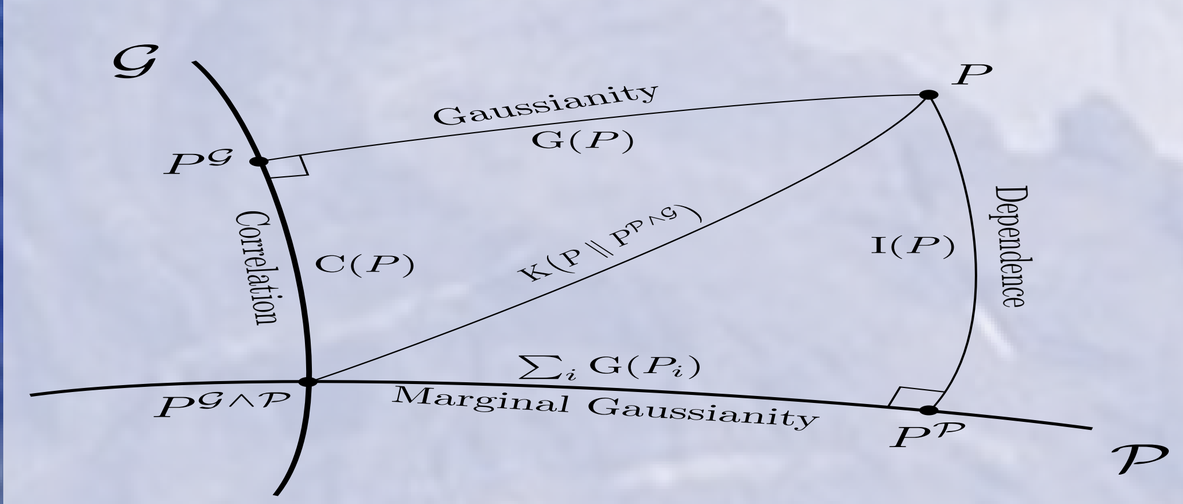

Figure: Representation of the distribution and the different projections on the manifolds $\mathcal{P}$ and $\mathcal{G}$ and the different quantities that can be define with the Kullback Leibler divergence.

The Pythagorean theorem gives the relation

$$\mathrm{I}(Y) + \sum_i \mathrm{G}(Y_i) = \mathrm{G}(Y) + \mathrm{C}(Y)\,. \tag{3}$$

If the mutual information $\mathrm{I}(P)$ appears to be the best one to use, it is to hard to compute. The equation 3 justifies the use of the non-gaussianity, correlation or even negentropy as contrast functions.

Performance evaluation

We use the "Amari distance", equation 4, which gives a criterion of proximity between two matrices, to evaluate the performance of an algorithm.
If U and V are two n-by-n matrices, the Amari distance is defined by:

$$d(U,V) = \frac{1}{2n}\sum_{i=1}^{n}\left(\frac{\sum_{j=1}^{n}|a_{ij}|}{\max_j |a_{ij}|} - 1\right) + \frac{1}{2n}\sum_{j=1}^{n}\left(\frac{\sum_{i=1}^{n}|a_{ij}|}{\max_i |a_{ij}|} - 1\right) \tag{4}$$

with $a_{ij} = (UV^{-1})_{ij}$. This function has the advantage to be invariant by scaling factors and permutation of the components of the matrices.

Algorithms

Several algorithms have been developed to perform ICA, among those we can cite:

- HJ
- JADE
- FastICA
- KernelICA

Hérault and Jutten (HJ) algorithm

This method is based on the neural network principle. We write $W = \left(I_n + \widetilde{W}\right)^{-1}$ and for a pair of given functions $(f,\ g)$, we adapt the $\widetilde{W}$ as follows:

$$\widetilde{W}_{ij} = f(y_i)g(y_j). \tag{5}$$

Jade algorithm

Several methods are based on the cumulants. The aim here is to annul all the cross cumulants of order 4. Thus, we diagonalize the cumulant tensor which is equivalent to minimise the following contrast function:

$$c\,(x) = \sum_{i,k,l} \left|\mathrm{Cum}\,(x_i, x_i^*, x_k, x_l)\right|^2. \tag{6}$$

FastICA algorithm

The FastICA algorithm is based on the information theory. We want here to maximise the marginal non-gaussianity on the whitened data, relying on a non linear quadratic function f with the following rule:

$$\widetilde{W}_{t+1} = \mathbb{E}\left[\, X.f(W_t^{\mathsf{T}}X)^{\mathsf{T}}\right] - \mathbb{E}\left[\, f''(W_t^{\mathsf{T}}X)\right] W_t, \tag{7}$$

with W_t the normalise vector of $\widetilde{W}_t$. In our case, we will use $f(x) = \frac{x^4}{4}$. We may use $f(x) = \log\cosh x$ or $f(x) = \exp\left(-\frac{x^2}{2}\right)$ too.

Kernel ICA

... À faire ...

Modus operandi

We have consider m distribution of probability, each sampled N times.
We mixed them with a random matrix, whitened this data and apply the ICA algorithm.
Eventually, we compare the matrix found with the true matrix using the Amari distance.

Results

Figure: Results of the main ICA algorithms on simulated data. Each colour corresponds to a signal. The "Sources" graph shows the unmixed signal s and the other ones the results of the algorithms.