Université Paris Dauphine

Probabilistic Graphical Models

Independent Component Analysis

Student:
Guillaume Ausset

Advisors:
Francis Bach
Guillaume Obozinski

Contents

1. The ICA framework

All code, examples, images and any other material can be found in the `.zip` *file provided. This zip file is actually a* `Git` *repository that can be found here:* `https://gitlab.com/aussetg/pgm-ica/tree/master`

In the physical world it is rare for the quantity of interest and the quantity measured to perfectly match. Scientists are used to dealing with measurements mixed with random noise but this is only one way information may be unavailable. A very common situation is when our signal of interest, that we will from now on call $s \in \mathcal{R}^{m \times n}$ is actually mixed with itself. To try to recover our signal s we will set up a certain number of recording devices, generally n and from our recorded signals try to recover the sources. If we assume the sources are linearly mixed our problem is then:

$$x = As$$

where $x \in \mathcal{R}^{k \times n}$, $A \in \mathcal{R}^{k \times m}$. While it is not necessarily the case or even needed from now on, we will assume $k = m$ so that the number of recorded signals equals the number of sources, the problem is then well posed.

If we somehow knew A the mixing matrix then this problem would be easy to solve and one would only need to solve $s = A^{-1}x$, but in general it's impossible to obtain the matrix A and the problem seems intractable, the number of unknowns dwarfing the number of known quantities.

We will see that in fact and quite surprisingly the sole hypothesis of the independence of the components of s is enough to find a solution to the problem.

There are, however, some limitations: because of the way the problem is posed one cannot hope to recover the real s as for example a scaling or permutation of A leaves the problem unchanged and we therefore cannot hope to recover the magnitudes (and therefore signs) of the s_i or their order. We will also not be able to recover signals if more than 1 is Gaussian as the distribution of the orthogonal transformation of a gaussian vector is the same as the original gaussian vector. See Hyvärinen et al., 2001 for a broader view of ICA.

2. Preprocessing

Before proceeding with the rest, we will quickly give two important preprocessing steps.

This preprocessing has two important functions: it greatly simplifies the theory at no additional cost, and it while reduce the computational complexity.

2.1. Centering

The first step is to centre the x such that $\mathbb{E}[x] = 0$, we therefore just have to form $\tilde{x} = x - \mathbb{E}[x]$.

This preprocessing greatly simplify the formulas and theory and have the advantage of not adding any constraint: once W is found we only have to add back $W\mathbb{E}[x]$ to the s found.

We will also work with RKHS, we therefore reminds that is possible to find the gram matrix of the observations centered in the feature space without explicitly knowing how to centre in the feature space. In the rest of the report all kernel matrices will be considered centred that way. If we note K the uncentred gram matrix then $\tilde{K}$ the centred gram matrix is given by:

$$
\begin{aligned}
H &= \mathbb{I} - \frac{1}{N}\mathbb{1}_{\mathbb{N}} \\
\tilde{K} &= HKH
\end{aligned}
$$

2.2. Whitening

ICA can be seen a generalization of PCA: while PCA gives uncorrelated components, ICA gives independent components. One implication of this fact is that an ICA solution is necessarily a PCA solution. It is therefore a good idea to first *whiten* the data, i.e. diagonalize its covariance matrix before treating it with ICA.

The whitening process can be performed efficiently using the semi-definite positiveness of the covariance matrix by forming its eigenvalue decomposition:

$$\mathbb{E}[xx^\intercal] = UDU^\intercal$$

Therefore if we note $P = UD^{-\frac{1}{2}}U^\intercal$, where $D^{-\frac{1}{2}}$ can be very efficiently computed, we have with $\tilde{x} = Px$

$$\mathbb{E}[xx^\intercal] = \mathbb{I}$$

Computationally, this transformation has the advantage of making W orthogonal, the parameters to estimate are therefore reduced from n^2 to $\frac{n(n-1)}{2}$. The new constraint "$WW^\intercal = \mathbb{I}$" also forces W to lie on the Stiefel Manifold, giving very efficient optimization techniques.

3. Measuring independence

It is possible to treat the problem as a parametric one and to fix a certain form for the distributions but that case is quite limiting. We while therefore work in a semi-parametric framework with parameter W the unmixing matrix:

$$s = Wy$$

In the case $m = k$ we then have $W = A^{-1}$. It is natural to study W directly as it is our parameter of interest and we will see that the problem is more easily expressed in terms of W.

We will suppose here that $m = k$ and A is invertible to make this paragraph easier.

Given our semi-parametric formulation we want to find the maximum likelihood estimate of A. We will note $p^\star(x)$ the unknown ground truth of x and $p(x)$ the model. Our problem is then:

$$\text{minimize } D\left(p^\star(x) \mid\mid p(x)\right)$$

We know that the KL divergence is invariant under an invertible transformation therefore the problem is equivalent to:

$$\text{minimize } D\left(p^\star(s) \mid\mid p(s)\right)$$

if we use the relation $x = As$.

We also have the following relation:

$$D(p^\star(s) \mid\mid p(s)) = D(p^\star(s) \mid\mid \tilde{p}(s)) + D(\tilde{p}(s) \mid\mid p(s))$$

with $\tilde{p}(s)$ the product of the marginal densities of $p^\star(s)$ i.e. the density if the components were independent.

The minimum is then obtained for $p^\star(s) = \tilde{p}(s)$ and the problem is:

$$\text{minimize } D\left(p^\star(x) \mid\mid \tilde{p}(s)\right)$$

We want to minimize the *mutual information between the components of s*.

A natural objective to optimize would therefore be the empirical mutual information of $s = Wx$ but the empirical mutual information is complicated to compute, to make the problem tractable two different approaches are possible:

1. Optimize an approximation of the mutual information.
2. Optimize a *contrast function* J, i.e a function with the same properties as the mutual information: $J(s) \geq 0, \forall s$ and $J(s) = 0$ iff the components of s are independent.

Fast-ICA will use the first approach while Kernel-ICA will use the second approach.

4. A quick reminder of Fast-ICA

We will use Fast-ICA as our reference algorithms as well as the method we will use to initialize our Kernel-ICA procedure. We therefore give a quick overview of the Fast-ICA algorithm.

Fast-ICA maximizes an approximation of the negentropy $J(s) = H(s_{\text{gauss}}) - H(s)$ which is equivalent to the minimizing the mutual information up to a constant.

Of course forming the empirical negentropy is equivalent to forming the empirical mutual information and the problem is therefore intractable. The approach taken is therefore to approximate the negentropy using non-linear functions.

In the case of non-quadratic function G, Hyvärinen, 1998, proves that

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

where $\nu \sim \mathcal{N}(0,1)$.

Two good (optimal in some sense) choices for G are:

$$G_1(u) = \log\cosh(au)$$
$$G_2(u) = e^{-\frac{u^2}{2}}$$

Fast-ICA is then a relatively simple fixed point algorithm obtained from the newton steps.

Algorithm 1 Fast-ICA

```
procedure FastICA($X, G, \epsilon$)
    $w \leftarrow$ Random orthogonal matrix
    while $1 - \min(|\text{diag}(w_i w_{i-1}^\intercal)|) > \epsilon$ do
        $w_{i+1} \leftarrow G(w_i X)X^\intercal - \left(G'(w_i X)\mathbb{1} \times \mathbb{I}\right) w_i$
        $w_{i+1} \leftarrow (w_{i+1} w_{i+1}^\intercal)^{-\frac{1}{2}} w_{i+1}$
    end while
    return $w, wX$
end procedure
```

By only working with matrix operations all computations can be done in-place and using a `BLAS`, making Fast-ICA extremely efficient.

5. Kernel-ICA

5.1. Kernel contrast function

We want to optimize a measure of independence of our estimated sources. Most approaches try to optimize an approximation of the mutual information. The measure chosen here by Bach and Jordan, 2002 is for two variables:

$$\rho_{\mathcal{F}} = \max_{f_1,f_2\in\mathcal{F}} \operatorname{corr}(f_1(x_1), f_2(x_2))$$

We see that it is indeed a contrast function: it is always greater than 0 and equal to 0 iff the variables are independent if the family $\mathcal{F}$ is *large enough* (if it includes Gaussian densities for example).

By exploiting the kernel trick we can obtain

$$\rho_{\mathcal{F}} = \max_{f_1,f_2\in\mathcal{F}} \operatorname{corr}(\langle\Phi_1(x_1), f_1\rangle, \langle\Phi_2(x_2), f_2\rangle)$$

We recognize a CCA problem and if we note K_1 and K_2 respective Gram matrices (assuming they are centred) we get the problem:

$$\begin{pmatrix} 0 & K_1K_2 \\ K_2K_1 & 0 \end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix} = \rho\begin{pmatrix} K_1^2 & 0 \\ 0 & K_2^2\end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix}$$

This problem is unfortunately not well posed and will always be equal to 1 for most kernels, we therefore adopt a regularized version as an estimator:

$$\rho_{\mathcal{F}} = \max_{f_1,f_2\in\mathcal{F}} \frac{\operatorname{cov}(f_1(x_1), f_2(x_2))}{(\operatorname{var} f_1(x_1) + \kappa\|f_1\|^2_{\mathcal{F}})^{1/2}(\operatorname{var} f_2(x_2) + \kappa\|f_2\|^2_{\mathcal{F}})^{1/2}}$$

The problem estimated at the first order is then:

$$\begin{pmatrix} 0 & K_1K_2 \\ K_2K_1 & 0 \end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix} = \rho\begin{pmatrix} (K_1+\frac{N\kappa}{2}\mathbb{I})^2 & 0 \\ 0 & (K_2+\frac{N\kappa}{2}\mathbb{I})^2\end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix}$$

Of course we are interested in solving the m-variables problems, we can easily extend CCA to m variables.

$$\begin{pmatrix} 0 & K_1K_2 & \cdots & K_1K_m \\ K_2K_1 & 0 & \cdots & K_2K_m \\ \vdots & \vdots & \ddots & \vdots \\ K_mK_1 & K_mK_2 & \cdots & 0 \end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\\ \vdots \\ \alpha_m\end{pmatrix} = \rho\begin{pmatrix} (K_1+\frac{N\kappa}{2}\mathbb{I})^2 & 0 & \cdots & 0 \\ 0 & (K_2+\frac{N\kappa}{2}\mathbb{I})^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (K_m+\frac{N\kappa}{2}\mathbb{I})^2\end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\\ \vdots \\ \alpha_m\end{pmatrix}$$

We can then transform that problem in the problem of finding the eigenvalues of:

$$\tilde{\mathcal{K}}_\kappa = \begin{pmatrix} \mathbb{I} & r_\kappa(K_1)r_\kappa(K_2) & \cdots & r_\kappa(K_1)r_\kappa(K_m) \\ r_\kappa(K_2)r_\kappa(K_1) & \mathbb{I} & \cdots & r_\kappa(K_2)r_\kappa(K_m) \\ \vdots & \vdots & \ddots & \vdots \\ r_\kappa(K_m)r_\kappa(K_1) & r_\kappa(K_m)r_\kappa(K_2) & \cdots & \mathbb{I} \end{pmatrix}$$

$$r_\kappa(K_i) = K_i(K_i + \frac{N\kappa}{2}\mathbb{I})^{-1}$$

We then optimize:

$$J(W) = -\frac{1}{2}\log C\left(\tilde{\mathcal{K}}_\kappa\right)$$

With $C(U) = \det U$ for the KGV contrast function or $C(U) = \lambda_{\min}(U)$ for KCCA. Both have strong links with the mutual information.

5.2. Reducing complexity

In practice the previous computation is intractable because of the dimensions involved and even forming the Gram matrices is impossible. We will therefore use the fact that the Gram matrices are semi-definite positive and use an incomplete Choleski decomposition to find a low rank approximation of the matrices involved.

Algorithm 2 Incomplete Choleski

```
procedure IncompleteCholeski($K \in \mathbb{R}^{n\times n}, \eta$)
    $i, K', P \leftarrow 1, K, \mathbb{I}$
    $G_{jj} \leftarrow K_{jj}, \forall j$
    while $\sum_{j=1}^{n} G_{jj} > \eta$ do
        $j \leftarrow \arg\max_{j\in[1,n]} G_{jj}$
        $P_{ii} \leftarrow 0, P_{j^\star j^\star} \leftarrow 0, P_{ij^\star} \leftarrow 1, P_{j^\star i} \leftarrow 1$
        $K_{1:n,i} \leftrightarrow K_{1:n,j^\star}$
        $K_{i,1:n} \leftrightarrow K_{j^\star,1:n}$
        $G_{i,1:i} \leftrightarrow G_{j^\star,1:i}$
        $G_{ii} = \sqrt{G_{jj}}$
        $G_{i+1:n,i} \leftarrow \frac{1}{G_{ii}}\left(K'_{i+1:n,i} - \sum_{j=1}^{i-1} G_{i+1:n,j}G_{i,j}\right)$
        for $j \in [i+1, n]$ do
            $G_{jj} = K_{jj} - \sum_{k=1}^{i} G_{jk}^2$
        end for
        $i \leftarrow i + 1$
    end while
    return $P, G, M = i - 1$
end procedure
```

If we decompose the K_i as

$$K_i = G_i G_i^\intercal = U_i \Lambda_i U_i^\intercal$$

with Λ_i diagonal then if R_i is Λ_i regularized by $\lambda \to \frac{\lambda}{\lambda + N\kappa/2}$ we have

$$\tilde{\mathcal{K}}_\kappa = (\mathcal{U}\mathcal{V}) \begin{pmatrix} \mathcal{R}_\kappa & 0 \\ 0 & \mathbb{I} \end{pmatrix} (\mathcal{U}\mathcal{V})^\intercal$$

with

$$\mathcal{R}_\kappa = \begin{pmatrix} \mathbb{I} & R_1 U_1^\intercal U_2 R_2 & \cdots & R_1 U_1^\intercal U_m R_m \\ R_2 U_2^\intercal U_1 R_1 & \mathbb{I} & \cdots & R_2 U_2^\intercal U_m R_m \\ \vdots & \vdots & \ddots & \vdots \\ R_m U_m^\intercal U_1 R_1 & R_m U_m^\intercal U_2 R_2 & \cdots & \mathbb{I} \end{pmatrix}$$

And therefore

$$\det \tilde{\mathcal{K}}_\kappa = \det \mathcal{R}_\kappa$$

We have therefore reduced the computational complexity from $O(m^2 n^3)$ to $O(m^2 M^2 n)$ with $M \ll n$.

5.3. Optimization on a manifold

As seen in 2.2, the whitening constrained our problem to:

$$\min_W J(W)$$
$$\text{s.t } WW^\intercal = \mathbb{I}$$

The set $\{W \mid WW^\intercal = \mathbb{I}\}$ has a particular geometry: it is Riemannian manifolds and most common optimization procedures can be performed on it Edelman et al., 1998. We use the fact that common procedures using parallel translations in $\mathbb{R}^d$ can be used in a differentiable manifold by using parallel transports along geodesic.

The simplest implementation is steepest descent along geodesics in the direction of the gradient using the following: if $W, H \in \mathcal{R}^{m\times n}$ s.t $W^\intercal W = \mathbb{I}$ and $A = W^\intercal H$ skew-symmetric then the geodesic on the Stiefel manifold emanating from W in direction H is given by the curve

$$\begin{aligned} & & W(t) &= WM(t) + QN(t) \\ &\text{where} & QR &= (\mathbb{I} - WW^\intercal)H \\ &\text{and} & \begin{pmatrix} M(t) \\ N(t) \end{pmatrix} &= \exp\left(t \begin{pmatrix} A & -R^\intercal \\ R & 0 \end{pmatrix}\right) \begin{pmatrix} \mathbb{I}_n \\ 0 \end{pmatrix} \end{aligned}$$

Given the cost of the gradient computations a natural extension is to perform conjugate gradient on the manifold, see Edelman et al., 1998 for the procedure. Unfortunately because of the complexity of the implementation I only used steepest descent along a geodesic, as in Bach and Jordan, 2002. We also note that clever use of the block structure of $\tilde{\mathcal{K}}_\kappa$ gives an efficient $O(m^2)$ algorithm to compute first order differences and that Bach and Jordan, 2002 provides a closed form of the gradient in the cases of polynomials and Gaussian kernels. I chose to only use finite differences because of their generality.

5.4. Possible improvements

My current implementation is straight forward and inspired by the implementation of Bach and Jordan, 2002 but several improvements can be obtained.

The first improvement, the easiest, is to notice that building the $\mathcal{R}_\kappa$ is done by block, each block being independent of the others, it is therefore possible to build this matrix entirely in parallel. The finite differences can also be computed in parallel but if parallelism is introduced in the contrast computation then this does not seem necessary.

A second more difficult improvement is to make use of conjugate gradient, the procedure is well understood on the Stiefel manifold and fully described in Edelman et al., 1998, but the complexity of the implementation is very high.

It would be worthwhile to implement Shen et al., 2009 which uses an approximate newton method.

Finally, it could be worthwhile to compare the implemented contrast functions with other kernel based contrast functions such as HSIC Shen et al., 2007 or COCO A. Gretton et al., 2005 and Arthur Gretton et al., 2005. HSIC in particular seems like a perfect fit:

- The empirical estimate of HSIC is simple, it's the trace of a product of Gram matrices.
- No regularization is needed.
- The empirical HSIC converges to HSIC quickly $(1/\sqrt{N})$.
- The bias is $O(N^{-1})$.
- Experiments show that HSIC achieve superior performances.

HSIC is based on the following expression:

$$\begin{aligned} &\mathrm{HSIC}(p_{xy}, \mathcal{F}, \mathcal{G}) := \|C_{xy}\|^2_{\mathrm{HS}} \\ \text{with} \quad &C_{xy} := \mathbb{E}_{x,y}\left[(\phi_{\mathcal{F}}(x) - \mu_x) \otimes (\phi_{\mathcal{G}}(y) - \mu_y)\right] \end{aligned}$$

While this formula doesn't seem simple, the finite sample estimator has the form:

$$\mathrm{HSIC}(Z_n, \mathcal{F}, \mathcal{G}) := (m-1)^2 \operatorname{tr} KHLH$$

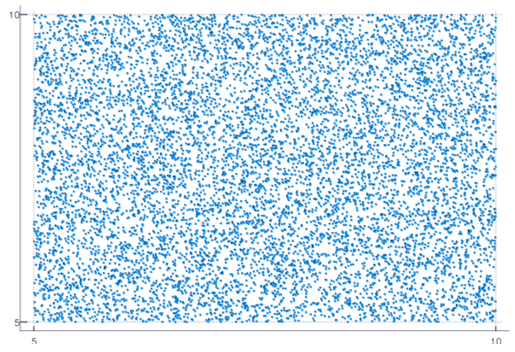
With K and L kernel matrices and $H = \mathbb{1} - \frac{1}{m}$.

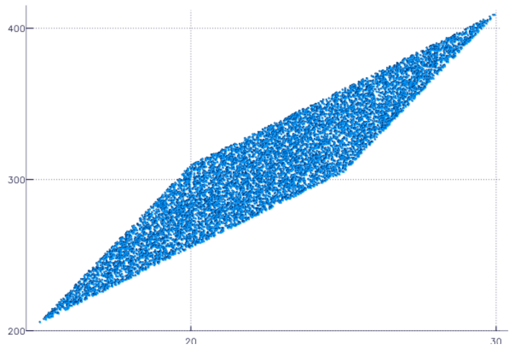
While I didn't have time to finish I started implemented HSIC with approximate newton in the branch `https://gitlab.com/aussetg/pgm-ica/tree/FastKICA`. I intend to finish implementing it.

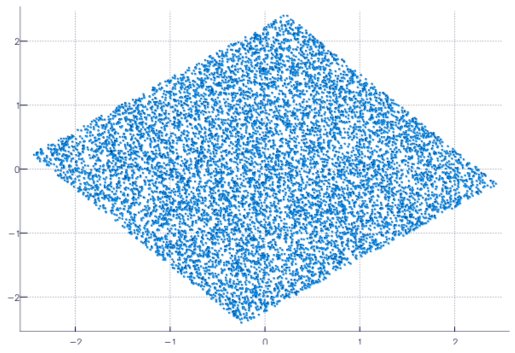
A. Experimental results

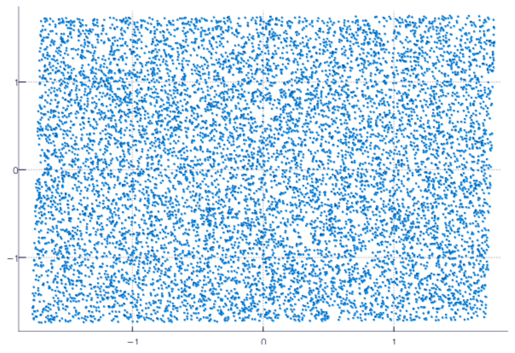
A.1. Visual representation of ICA

This diagram illustrate how ICA works be first whitening the data then finding the optimal rotation.


(a) Original sources


(b) Mixed sources


(c) Whitening


(d) ICA

A.2. Unmixing Images

Here we just illustrate source separation by applying it to images. We are interested in unmixing images that have been mixed linearly.

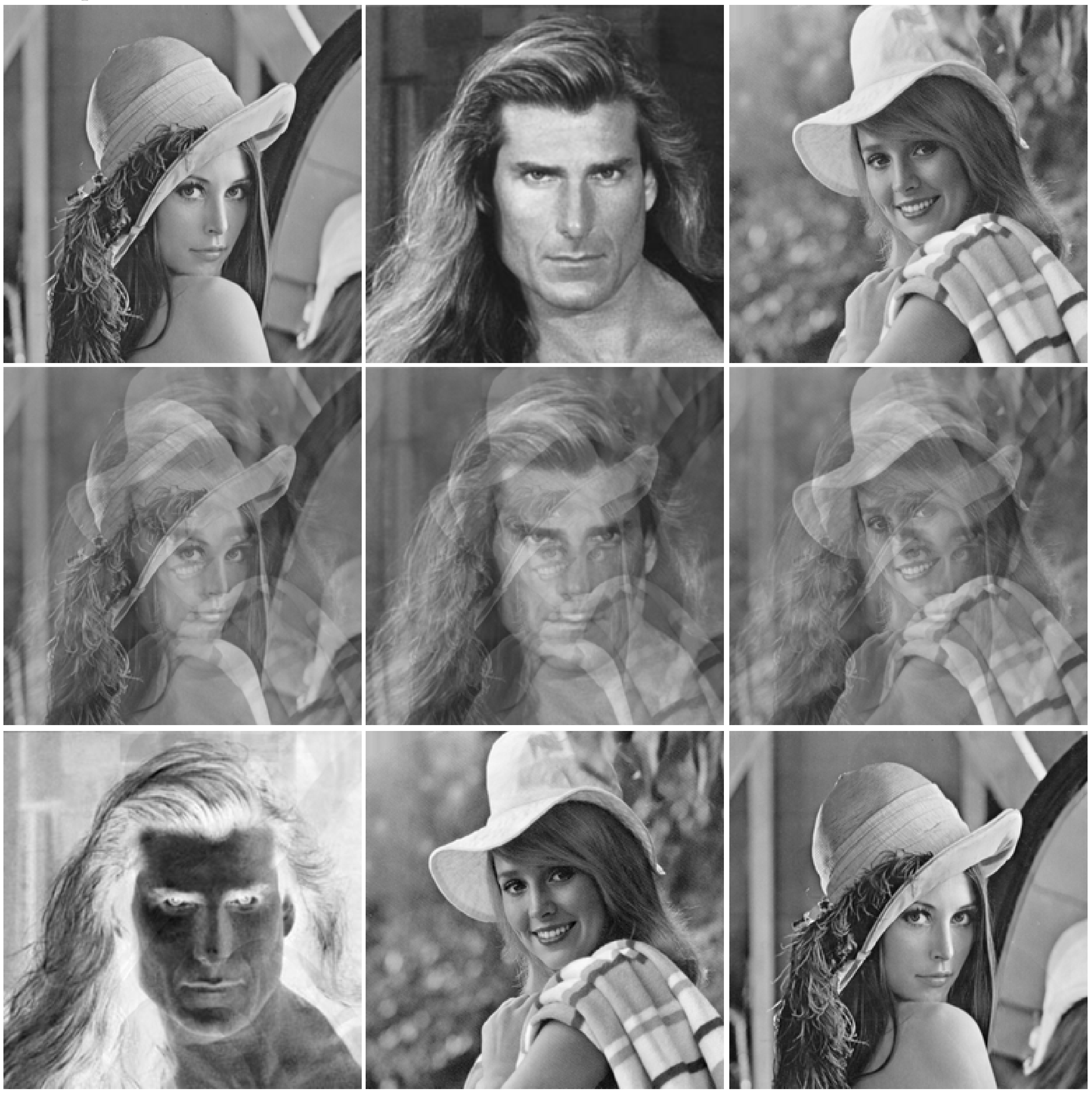


Figure A.1.: Original images, mixed then unmixed.

A.3. Finding a natural basis of images

In this problem the quantity of interest isn't s anymore but A or w. We interpret this problem as finding a basis of images (if we use A) or of detectors (if we use w) and s being the realizations of the coefficients for this basis. We actually either find a basis of edges or the corresponding Gabor filters, Bell and Sejnowski, 1997.

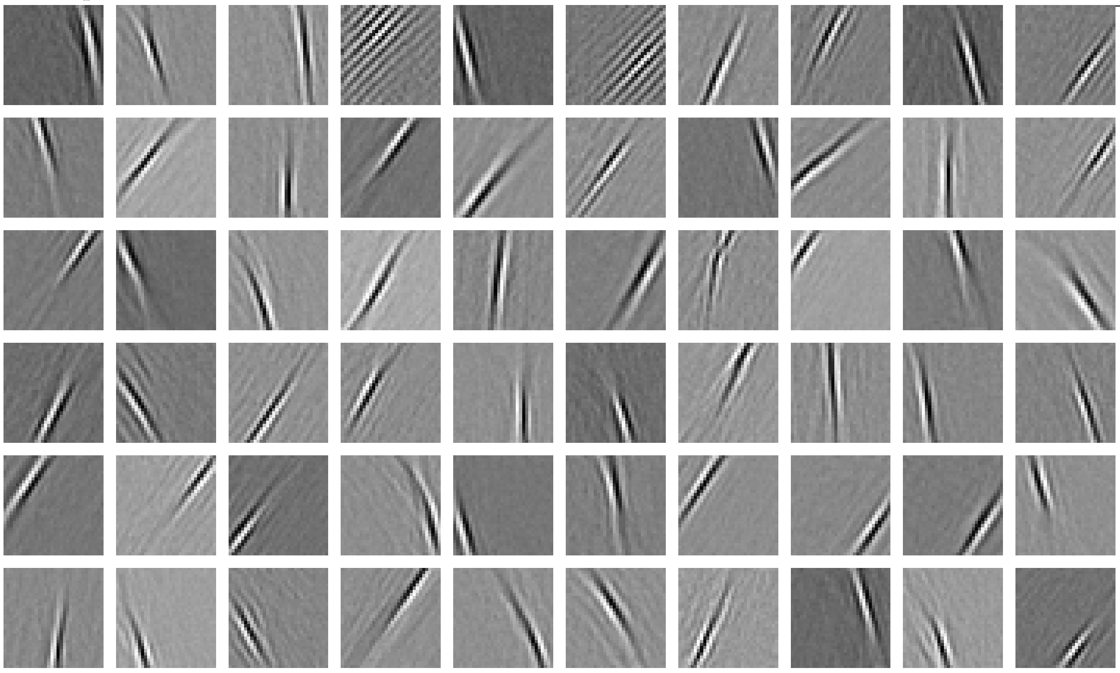


Figure A.2.: Gabor filters for the previous images.

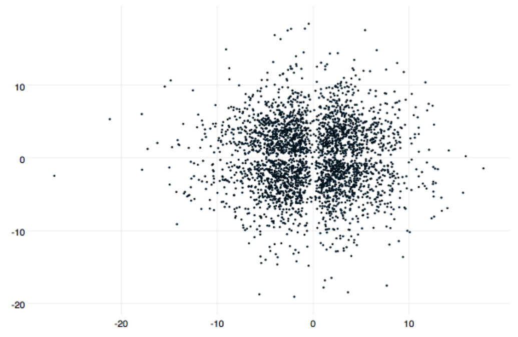
A.4. Comparing Kernel-ICA and Fast-ICA

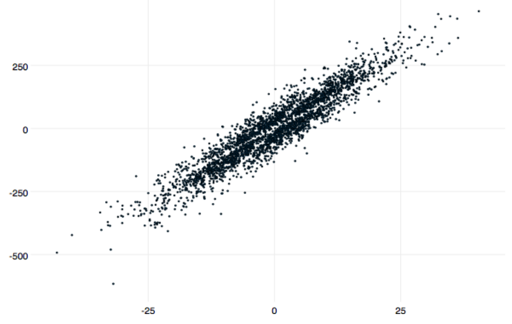
In the accompanying `Jupyter Notebook` I give an example of comparison of Fast-ICA and Kernel-ICA. For that we need a way of measuring the performance of the algorithms. If we know the real w we can use the Amari distance (which isn't a real distance) as it is invariant by scaling and permutation, the two indeterminate of our problem.

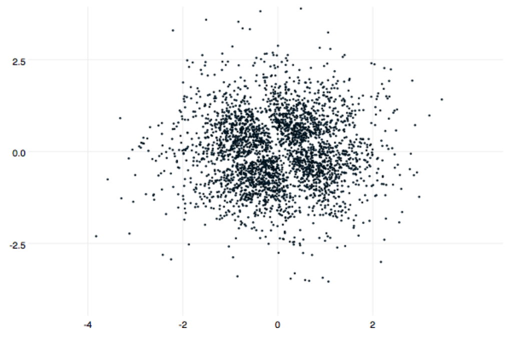
We give here the Amari distance:

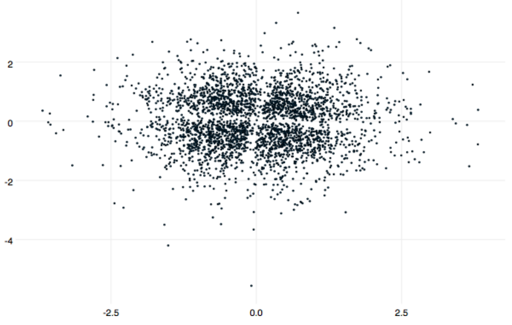
$$d(V,W) = \frac{1}{2m}\sum_{i=1}^{m}\left(\frac{\sum_{j=1}^{m}|a_{ij}|}{\max_j |a_{ij}|} - 1\right) + \frac{1}{2m}\sum_{j=1}^{m}\left(\frac{\sum_{i=1}^{m}|a_{ij}|}{\max_i |a_{ij}|} - 1\right)$$

We compare both algorithms on a hard distribution (with zero excess kurtosis).


(a) Original sources


(b) Mixed sources


(c) Fast-ICA


(d) Kernel-ICA

A.5. Cocktail party and more

I have other examples less suitable for a paper report in the accompanying notebook.

Bibliography

Bach, Francis R and Michael I Jordan

2002 "Kernel Independent Component Analysis", *Journal of Machine Learning Research*, 3, pp. 1-48, ISSN: 0003-6951, DOI: 10.1162/153244303768966085. (Cited on pp. 7, 9, 10.)

Bell, Anthony J. and Terrence J. Sejnowski

1997 "Edges are the 'Independent Components' of Natural Scenes", *Vision Research*, 37, 23, pp. 3327-3338, ISSN: 1098-6596, DOI: 10.1017/CBO9781107415324.004, arXiv: arXiv:1011.1669v3. (Cited on p. 12.)

Edelman, Alan, Tomás A. Arias, and Steven T. Smith

1998 "The Geometry of Algorithms with Orthogonality Constraints", *SIAM Journal on Matrix Analysis and Applications*, 20, 2, pp. 303-353, ISSN: 0895-4798, DOI: 10.1137/S0895479895290954, arXiv: 9806030 [physics], http://epubs.siam.org/doi/abs/10.1137/S0895479895290954. (Cited on pp. 9, 10.)

Gretton, A., R. Herbrich, A. J. Smola, O. Bousquet, and B. Schölkopf

2005 "Kernel Methods for Measuring Independence", *Journal of Machine Learning Research*, 6, pp. 2075-2129, ISSN: 1532-4435, http://eprints.pascal-network.org/archive/00001702/. (Cited on p. 10.)

Gretton, Arthur, Olivier Bousquet, Alex Smola, and Bernhard Sc̈hlkopf

2005 "Measuring statistical dependence with Hilbert-Schmidt norms", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3734 LNAI, pp. 63-77, ISSN: 03029743, DOI: 10.1007/11564089_7. (Cited on p. 10.)

Hyvärinen, Aapo

1998 "Independent component analysis in the presence of gaussian noise by maximising joint likelihood", *Neurocomputing*, pp. 1-. (Cited on p. 6.)

Hyvärinen, Aapo, Juha Karhunen, and Erkki Oja

2001 "Independent Component Analysis", *Analysis*, 26, 1, p. 481, ISSN: 10635203, DOI: 10.1016/j.acha.2006.03.003, http://linkinghub.elsevier.com/retrieve/pii/S1063520306000509. (Cited on p. 3.)

Shen, Hao, Stefanie Jegelka, and Arthur Gretton

2007 "Fast Kernel ICA using an Approximate Newton Method", p. 8, ISSN: 15324435, http://eprints.pascal-network.org/archive/00003141/. (Cited on p. 10.)

2009 "Fast Kernel-Based Independent Component Analysis", *IEEE Transactions on Signal Processing*, 57. (Cited on p. 10.)