ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 205-210

How to generalize Geometric ICA to higher dimensions

Fabian J. Theis, Elmar W. Lang

Institute of Biophysics
University of Regensburg, D-93040 Regensburg, Germany
email: fabian.theis@mathematik.uni-regensburg.de

Abstract. Geometric algorithms for linear independent component analysis (ICA) have recently received some attention due to their pictorial description and their relative ease of implementation. The geometric approach to ICA has been proposed first by Puntonet and Prieto [6] in order to separate linear mixtures. One major drawback of geometric algorithms is, however, an exponentially rising number of samples and convergence times with increasing dimensiononality thus basically restricting geometric ICA to low-dimensional cases. We propose to apply overcomplete ICA to geometric ICA [7] to reduce high-dimensional problems to lower-dimensional ones, thus generalizing geometric ICA to higher dimensions.

1 Basics

For $m, n \in \mathbb{N}$ let $\mathrm{Mat}(m \times n)$ be the $\mathbb{R}-$vectorspace of real $m \times n$ matrices, and $\mathrm{Gl}(n) := \{W \in \mathrm{Mat}(n \times n) \mid \det(W) \neq 0\}$ be the general linear group of $\mathbb{R}^n$.

In linear blind source separation (BSS), a random vector $X : \Omega \to \mathbb{R}^m$ (**mixed vector**) originates from an independent random vector $S : \Omega \to \mathbb{R}^n$ (**source vector**) by mixing with a **mixing matrix** $A \in \mathrm{Mat}(m \times n)$, i.e. $X = AS$. Here Ω denotes a fixed probability space. Only the mixed vector is known, and the task is to recover both the mixing matrix A and the source signals S.

We will assume that the mixing matrix A has full rank. In the **quadratic** case $(m = n)$ A then is invertible ie. $A \in \mathrm{Gl}(n)$ and S can be recovered from A by $S = A^{-1}X$. This is not the true in the **overcomplete** case where less mixtures than sources are given $(m < n)$; then the BSS problem is ill-posed, hence further restrictions like source density assumptions have to be made.

In the following we denote two matrices $B, C \in \mathrm{Mat}(m \times n)$ to be **equivalent**, $B \sim C$, if C can be written as $C = BPL$ with an invertible diagonal matrix (scaling matrix) $L \in \mathrm{Gl}(n)$ and an invertible matrix with unit vectors in each row (permutation matrix) $P \in \mathrm{Gl}(n)$. Similarly, B is said to be

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 205-210

scaling-equivalent to C, $B \sim_s C$, if $C = BL$ holds, and B is **permutation-equivalent** to C, $B \sim_p C$, if $C = BP$. Therefore, if B is scaling- or permutation-equivalent to C, it is equivalent to C, but not vice-versa. If we write

$$B = (b_1| \dots |b_n)$$

where $b_i = Be_i$ are the columns of the matrix B, we have the following trivial lemmata:

Lemma 1.1. *$B \sim_s C$ if and only if $(c_1| \dots |c_n) = (\lambda_1 b_1| \dots |\lambda_n b_n)$ with $\lambda_i \in \mathbb{R} \setminus \{0\}$.*

Lemma 1.2. *$B \sim_p C$ if and only if $(c_1| \dots |c_n) = (b_{p(1)}| \dots |b_{p(n)})$ with $p \in S_n$ a permutation.*

Corollary 1.3. *$B \sim C$ if and only if $(c_1| \dots |c_n) = (\lambda_1 b_{p(1)}| \dots |\lambda_n b_{p(n)})$ with $\lambda_i \in \mathbb{R} \setminus \{0\}$ and $p \in S_n$ a permutation.*

If at most one of the source variables $S_i := \pi_i \circ S$ is Gaussian ($\pi_i : \mathbb{R}^n \to \mathbb{R}$ denotes the projection on the i-th coordinate) then for any **solution to the quadratic ($m = n$) BSS problem**, i.e. any $D \in \mathrm{Gl}(n)$ such that $D \circ X$ is independent, D^{-1} is equivalent to A [3]. Vice versa, any matrix $D \in \mathrm{Gl}(n)$ such that D^{-1} is equivalent to A solves the BSS problem, since we calculate for the transformed mutual information

$$I(D \circ X) = I(LPA^{-1} \circ X) = I(A^{-1} \circ X) = I(S) = 0,$$

taking into account that the information is invariant under scaling and permutation of coordinates.

For the overcomplete case no such uniqueness results exist. However it is easy to see that in this case an estimate of the unknown mixing matrix can only be obtained up to equivalence: If B is equivalent to A that is $A = BLP$, then set $S' := LPS$. S' is independent because the mutual information is invariant under scaling and permutation, and mixing S' gives again X because $X = AS = BLPS = BS'$.

2 Mixing dimension reduction

We consider quadratic ICA in n dimensions. The basic idea is to project the mixtures X onto different subspaces and then to estimate the original mixture matrix from the recovered projected matrices. With maximum likelihood source recovery algorithms within overcomplete ICA, it can be shown experimentally that the recovered sources are rather bad estimates of the original sources [7]. Therefore we use multiple recoveries to estimate the original *mixing matrix* A; we then only have to invert A to get the recovered sources.

One problem arises, however. As mentioned in the previous section, ICA algorithms can only find the mixing matrix up to equivalence (in the overcomplete case, it is not known if there is even more non-uniqueness to account for),

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 205-210

so we have to take that into account when comparing the various recovered matrices. We then eliminate permutation by comparing the correlation matrices of the sources estimated with overcomplete ICA form each recovered mixing matrix. Finally scaling will be accounted for by normalizing each recovered mixing matrix.

3 Equivalence after projections

Let $A \in \mathrm{Gl}(n)$, $n > 2$ be the invertible mixing matrix. Now let $m \in \mathbb{N}$ with $1 < m < n$, and let M denote the set of all subsets of $[1, n] := \{1, \ldots, n\}$ of size m. For an element $\tau \in M$ let $\tau = \{\tau_1, \ldots, \tau_m\}$ such that $\tau_1 < \ldots < \tau_m$. Let π_τ denote the (ordered) projection from $\mathbb{R}^n$ onto those coordinates given by τ, ie.

$$\begin{aligned} \pi_\tau : \mathbb{R}^n &\longrightarrow \mathbb{R}^m \\ (x_1, \ldots, x_n) &\longmapsto (x_{\tau_1}, \ldots, x_{\tau_m}). \end{aligned}$$

Obviously there are $\binom{n}{m}$ such projections for fixed m and n. For $\tau \in M$, we will consider the projected mixing matrix $A_\tau := \pi_\tau A : \mathbb{R}^n \longrightarrow \mathbb{R}^m$.

Lemma 3.1. *Let $A, B \in \mathrm{Gl}(n)$ and let $\tau^1, \ldots, \tau^k \in M$ such that $\bigcup_i \tau^i = [1, n]$. Then if $A_{\tau^i} \sim_s B_{\tau^i}$ for all i, $A \sim_s B$.*

Proof. From lemma 1.1 we know that for each $i = 1, \ldots, k$ there exist $\lambda_j^i \in \mathbb{R} \setminus \{0\}$ such that $b_{\tau_j^i} = \lambda_j^i a_{\tau_j^i}$ for $j = 1, \ldots, m$. As $\bigcup_i \tau^i = [1, n]$, we deduce that for each $l = 1, \ldots, n$ there exists i and j such that $b_l = \lambda_j^i a_l$, which again by lemma 1.1 means that B is scaling-equivalent to A. □

Now assume that the first row of A does not contain any zeros that is $a_{1j} \neq 0$ for $j = 1, \ldots, n$. This is a very mild assumption because A was assumed to be invertible, and the set with A's as above is dense in $\mathrm{Gl}(n)$.

As usual let $\lceil x \rceil$ denote the smallest integer larger or equal to $x \in \mathbb{R}$. Then let $k = \left\lceil \frac{n-1}{m-1} \right\rceil$, and define $\tau^i := \{1, 2 + (m-1)(i-1), \ldots, 2 + (m-1)i - 1\}$ for $i < k$ and $\tau^k := \{1, n-m+2, \ldots, n\}$. Then $\bigcup_i \tau^i = [1, n]$ and $\bigcap_i \tau^i \supset \{1\}$. Given any $B^1, \ldots, B^k \in \mathrm{Mat}(m \times n; \mathbb{R})$, define $A_{B^1,\ldots,B^k}$ by

$$A_{B^1,\ldots,B^k} := \begin{pmatrix} 1 & \ldots & 1 \\ (B^1)_{2,1}/(B^1)_{1,1} & \ldots & (B^1)_{2,n}/(B^1)_{1,n} \\ \vdots & \ddots & \vdots \\ (B^1)_{m,1}/(B^1)_{1,1} & \ldots & (B^1)_{m,n}/(B^1)_{1,n} \\ (B^2)_{2,1}/(B^2)_{1,1} & \ldots & (B^2)_{2,n}/(B^2)_{1,n} \\ \vdots & \ddots & \vdots \\ (B^{k-1})_{m,1}/(B^{k-1})_{1,1} & \ldots & (B^{k-1})_{m,n}/(B^{k-1})_{1,n} \\ (B^k)_{3+k(m-1)-n,1}/(B^k)_{1,1} & \ldots & (B^k)_{3+k(m-1)-n,n}/(B^k)_{1,n} \\ \vdots & \ddots & \vdots \\ (B^k)_{m,1}/(B^k)_{1,1} & \ldots & (B^k)_{m,n}/(B^k)_{1,n} \end{pmatrix}.$$

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 205-210

Lemma 3.2. *Let $A \in \mathrm{Gl}(n)$ and let $B^1, \ldots, B^k \in \mathrm{Mat}(m \times n; \mathbb{R})$ such that $A_{\tau^i} \sim_s B^i$ for $i = 1, \ldots k$. Then $A_{B^1,\ldots,B^k} \sim_s A$, in particular $A_{B^1,\ldots,B^k} \sim A$.*

Proof. From lemma 1.1 we know that for each $i = 1, \ldots, k$ there exist $\lambda_j^i \in \mathbb{R} \setminus \{0\}$ such that $(B^i)_{j,l} = \lambda_j^i (A_{\tau^i})_{j,l}$, hence $(B^i)_{j,l}/(B^i)_{1,l} = (A_{\tau^i})_{j,l}/(A_{\tau^i})_{1,l}$. One can check that by the choice of the τ^i's we then have $(A_{B^1,\ldots,B^k})_{j,l} = A_{j,l}/A_{j,1}$ and therefore $A_{B^1,\ldots,B^k} \sim_s A$. □

The above lemma can be used to cancel out scalings when putting together the recovered mixing matrices. In order to eliminate permutations however, we have to make the further assumption that the source data is sparse in the sense that its source distribution is supergaussian in every direction.

4 Reduction Algorithm

The reduction algorithm now is very simple. Pick k and $\tau^1, \ldots, \tau^k$ as in the previous section. Now perform overcomplete matrix-recovery [7] with the projected mixtures $\tau^i(X)$ for $i = 1, \ldots, k$ and get matrices B^i. We assume that this recovery has been carried out without any error; then every B^i is equivalent to A_{τ^i} in this ideal case.

The B^i however are not scaling-equivalent to the A_{τ^i}, so we will have to establish this in the next step of the algorithm. Therefore do the following iteratively for each $i = 1, \ldots, k$: Apply the overcomplete source-recovery [7] to $\tau^i(X)$ using B^i and get recovered sources S^i. For all $j < i$, consider the absolute correlation matrices $(|\operatorname{Cor}(S_r^i, S_s^j)|)_{(r,s)}$. Using the maxima in every column of this matrix, we claim that the row positions of these maxima are pairwise different because the original sources were chosen to be independent. Thereby we get a permutation matrix P^i indicating how to permute B^i, $C^i := B^i P^i$, so that it fits nicely in the above sense to the previously recovered C^j's, for $j < i$. Finally, we have constructed matrices C^i with $C^i \sim_p B^i$ such that there exists a permutation P with $C^i \sim_s A_{\tau^i} P$ for all $i = 1, \ldots, k$.

Now we can apply lemma 3.2 and get a matrix $A_{C^1,\ldots,C^k}$ with $A_{C^1,\ldots,C^k} \sim_s AP$ and therefore $A_{C^1,\ldots,C^k} \sim A$ as desired.

5 Experimental Results

In this section, we give some demonstration of the algorithm. The calculations have been performed on an AMD Athlon 1 GHz computer using Matlab and took no more than five seconds at most.

As example we consider a mixture of three high-kurtosis signals ($n = 3$) as pictured in figure 1, top left picture. They were mixed with the mixing matrix

$$A = \begin{pmatrix} 0.7071 & -0.2182 & -0.8018 \\ 0.7071 & 0.4364 & 0.5345 \\ 0 & 0.8729 & -0.2673 \end{pmatrix}$$

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 205-210

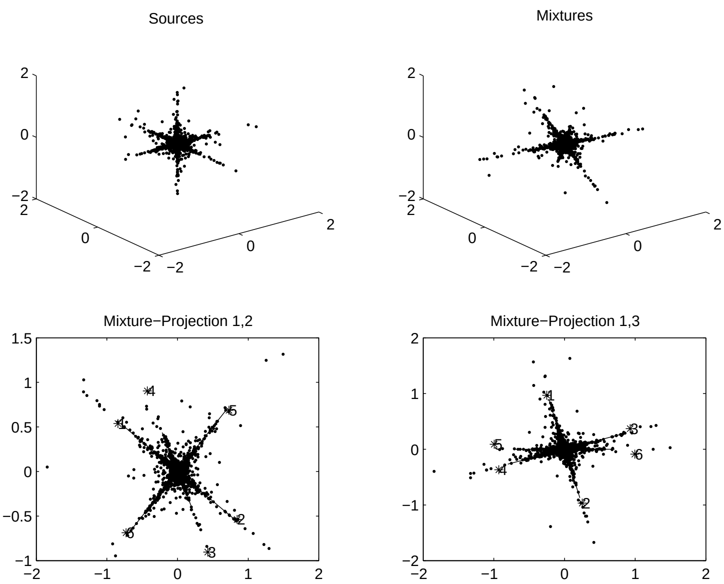

Figure 1: Example: Mixture of three high-kurtosis signals. The top left image shows a 3-dimensional scatterplot of the sources, the top right image shows the 3-dimensional scatterplot of the mixtures. The two lower pictures show the scatterplot of the mixtures after projecting onto the $1-2$- and the $1-3$-plane respectively. Furthermore, the stars and the numbers indicate the neurons that were used for the geometric overcomplete matrix-recovery algorithm.

The rather complicated general framework for the projections simplifies in the case $n = 3$, $m = 2$ to $k = 2$ and to two projections $\pi_{\{1,2\}}$ and $\pi_{\{1,3\}}$, which we also denote by $\pi_{1,2}$ and $\pi_{1,3}$.

We performed overcomplete ICA using the geometric algorithm from [7] with 4000 sweeps and initial learning rate of 1.0. The two recovered matrices were

$$B_{1,2} := B^1 = \begin{pmatrix} -0.8430 & 0.4283 & 0.7275 \\ 0.5379 & -0.9036 & 0.6862 \end{pmatrix}$$

and

$$B_{1,3} := B^2 = \begin{pmatrix} -0.2508 & 0.9298 & -0.9960 \\ 0.9680 & 0.3681 & 0.0896 \end{pmatrix};$$

so the recoveries were indeed very good (see figure 1). The correlation matrix between the recovered sources S^1 and S^2 is

$$\mathrm{Cor}(S^1, S^2) = \begin{pmatrix} 0.0544 & -0.9368 & 0.3244 \\ -0.8660 & 0.0535 & -0.0134 \\ 0.0289 & 0.2584 & -0.9022 \end{pmatrix},$$

so the first column of B^2 belongs to the second of B^1, the second of B^2 to the first of B^1 and the third of B^2 to the third of B^1. Hence, we get the recovered

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 205-210

matrix

$$A' = \begin{pmatrix} 1.0000 & 1.0000 & 1.0000 \\ -2.1099 & -0.6381 & 0.9432 \\ -3.8601 & 0.3959 & -0.0900 \end{pmatrix},$$

and the crosstalking error of A and A' is $E_1(A^{-1}A') = 0.4440$.

6 Conclusion

We have shown how to generalize geometric algorithms to higher dimensions. We propose to apply overcomplete ICA to various projections of the mixture data; then, taking into account the permutation equivalence of the recovered mixing matrices, we estimate the final mixing matrix from these recovered matrices. We proved that the estimated mixing matrix is equivalent to the original mixing matrix. This algorithm has been successfully applied to the case $n = 3$, and can now readily be generalized to higher dimensions.

For further research, we suggest two directions. First, the above algorithm has to be examined in more detail, especially how well it generalizes to large n, and we have to test if the high sample requirements for traditional geometric algorithms in high dimensions are indeed greatly reduced as indicated by first experiments. Furthermore, other overcomplete algorithms [5] [2] could be plugged into the above framework, and the quadratic 'FastGeo' algorithm [4], a histogram-based geometric algorithm, could be used to improve the speed of the geometric algorithm. Second, it would be interesting to know if the above algorithm can be reduced to the well-known quadratic ICA algorithm formula introduced by Bell and Sejnowski [1].

References

[1] A. J. Bell and T.J. Sejnowski. An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[2] P. Bofill and M. Zibulevsky. Blind separation of more sources than mixtures using sparsity of their short-time fourier transform. *P.Pajunen, J.Karhunen, eds., Independent Component Analysis and Blind Signal Separation (Proc. ICA'2000)*, pages 87–92, 2000.

[3] P. Comon. Independent component analysis - a new concept? *Signal Processing*, 36:287–314, 1994.

[4] A. Jung, F.J. Theis, C.G. Puntonet, and E.W. Lang. Fastgeo - a histogram based approach to linear geometric ica. *ICA2001 accepted*, 2001.

[5] M.S. Lewicki and T.J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 1998.

[6] C.G. Puntonet and A. Prieto. An adaptive geometrical procedure for blind separation of sources. *Neural Processing Letters*, 2, 1995.

[7] F.J. Theis and E.W. Lang. Geometric overcomplete ICA. *submitted to ESANN 2002*, 2002.