Title
A Null-space Algorithm for Overcomplete Independent Component Analysis

Permalink
https://escholarship.org/uc/item/33z4b05d

Authors
Chen, Ray-Bing
Wu, Ying N

Publication Date
2002-01-01
Abstract

Independent component analysis (ICA) is an important method for blind source separation and unsupervised learning. Recently, the method has been extended to the overcomplete situation where the number of sources is greater than the number of receivers. Comparing complete ICA and overcomplete ICA in existing literature, one can notice that complete ICA does not assume noise in observations, and the sources can be explicitly solved from the receivers, whereas the overcomplete ICA in general assumes noise in observations and the sources are implicitly solved by gradient type algorithms. In this paper, we present an explicit null-space representation for overcomplete ICA in the noiseless situation based on singular value decomposition (SVD), and develop an algorithm for estimating mixing matrix and recovering the sources. The null-space representation makes the connection between complete ICA and overcomplete ICA more apparent, and leads to a mathematical explanation of lateral inhibition in the context of overcomplete linear model. It also appears to work well on the experimental examples. Moreover, it can be extended to the situation where there is noise in observations, and may lead to more efficient algorithms in this situation.

1 Introduction

Independent component analysis (ICA, e.g., Comon, 1994, Pece, 2001, and Hyvärinen, Karhunen & Oja, 2001) is one of the most exciting developments in signal processing and unsupervised learning in recent years. It has been applied quite successfully to blind source separation (Bell & Sejnowski, 1995), and statistical modeling and learning of natural images (Bell & Sejnowski, 1997), natural sounds (Bell & Sejnowski, 1996), and video sequences (van Hateren & Ruderman, 1998).

The idea of overcomplete linear representation and sparse coding was studied by the seminal paper of Olshausen & Field (1997). It was later connected to ICA by Lewicki & Sejnowski (2000), who developed the overcomplete ICA methodology. The method was then applied to blind source separation by Lee, Lewicki, Girolami & Sejnowski (1999). While the sparsity principle of Olshausen & Field (1997) is fundamental in its own right and does not require independence assumption, in this paper, we shall concentrate on the overcomplete representation with independent sources which may or may not be sparse.
If we compare complete ICA and overcomplete ICA, we may detect two differences. 1) The complete ICA usually does not assume noise in observations, whereas the overcomplete ICA often assumes noise. 2) In complete ICA, the sources can be explicitly solved from the receivers, whereas in overcomplete ICA, the sources are implicitly solved by gradient type algorithms. As to the observation noise, Bell & Sejnowski (1997) argued that: “in a general information theoretic framework, there is nothing to distinguish signal and noise a priori. … We considered ‘noisefree infomax’ to be the appropriate framework for making the first level of predictions based on information-theoretic reasoning.” While we have no objection to the assumption of Gaussian noise in overcomplete ICA, we feel it is interesting to investigate the noiseless situation. Intellectually, this will make the transition from complete ICA to overcomplete ICA more apparent, and consequently will lead to more understanding of the overcomplete ICA. Computationally, it will lead to general estimation algorithms for the noiseless situation, and may prove to be useful for situations with small noise.

The central idea of this paper is easy to convey. When we have a noiseless overcomplete system, i.e., the receivers is a linear mixing of sources, and there are more sources than the receivers, we can still solve this linear system explicitly, that is, representing the sources explicitly in terms of receivers, just as in the case of complete ICA, except that in the overcomplete situation, the solution is not unique, and the set of solutions can be identified explicitly via the null space of the mixing matrix. We call this representation the null-space representation, and the basis of the null-space can be identified by the singular value decomposition (SVD). Interestingly, this representation gives a mathematical explanation of the lateral inhibition in this particular context.

We then move on to develop an estimation algorithm, which consists of an inhibition algorithm for recovering the sources, and a Givens sampler for estimating the mixing matrix. We then apply the algorithm to the blind source separation problem with sparse sources as well as autoregressive sources. The experimental results appear to be encouraging. We shall also discuss how to deal with the situation with observation noise.

2 The Null-Space Representation via Singular Value Decomposition

To fix notation, let

$$x_t = As_t, \quad t = 1, \cdots, T,$$

where \(x_t = (x_{1t}, \cdots, x_{mt})^T\) is the receiver vector at time \(t\), \(s_t = (s_{1t}, \cdots, s_{Mt})^T\) is the source vector at time \(t\), and \(A\) is an \(m \times M\) mixing matrix with \(M > m\). Lewicki & Sejnowski (2000) considered this noiseless situation under the double exponential prior of \(s_t\), without elaborating on general situation.

The solutions to this overcomplete linear system can be easily identified. Let

$$\tilde{s}_t = A^{-1}x_t$$

be a solution, where \(A^{-1}\) is a generalized inverse of \(A\), and let

$$s_t = \tilde{s}_t + \Delta s_t$$

be another solution. Then clearly, \(\Delta s_t\) is in the null space of \(A\), that is, \(A\Delta s_t = 0\). To identify the basis for the null space, let’s consider the simplest situation first, where \(A = \begin{pmatrix} D & 0 \\ \end{pmatrix}\) and \(D = \text{diag}(d_i)\) is an \(m \times m\) diagonal matrix. Now the equation is

$$
\begin{pmatrix}
(x_{1t} \\
\vdots \\
(x_{mt})
\end{pmatrix}
= 
\begin{pmatrix}
d_1 & 0 & \cdots & 0 \\
0 & \ddots & \ddots & \vdots \\
\cdots & \ddots & \ddots & \ddots \\
0 & \cdots & d_m & 0 & \cdots & 0
\end{pmatrix}
\begin{pmatrix}
s_{1t} \\
\vdots \\
(s_{Mt})
\end{pmatrix}.
$$


Then all the solutions can be written as
\[ \mathbf{s}_t = \begin{pmatrix} D^{-1} & 0 \\ 0 & I_{M-m} \end{pmatrix} \begin{pmatrix} x_{1t} \\ \vdots \\ x_{mt} \end{pmatrix} + \begin{pmatrix} 0 \\ I_{M-m} \end{pmatrix} \begin{pmatrix} c_{1t} \\ \vdots \\ c_{(M-m)t} \end{pmatrix} = \begin{pmatrix} D^{-1} & 0 \\ 0 & I_{M-m} \end{pmatrix} \begin{pmatrix} x_t \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ I_{M-m} \end{pmatrix} c_t \] (1)
for an arbitrary coefficient vector \( c_t = (c_{1t}, \cdots, c_{(M-m)t})' \), and it is easy to see that \( c_t \) is a vector of coordinates of the null space.

Now let’s consider the general situation. Suppose we could rotate \( \mathbf{x}_t \) to \( U \mathbf{x}_t \), and rotate \( \mathbf{s}_t \) to \( V' \mathbf{s}_t \), where \( U \) is an \( m \times m \) rotation matrix, and \( V \) is an \( M \times M \) rotation matrix, so that the rotated vectors satisfy the simplest system studied above, i.e.,
\[ U \mathbf{x}_t = \begin{pmatrix} D & 0 \end{pmatrix} V' \mathbf{s}_t, \]
then according to equation (1),
\[ V' \mathbf{s}_t = \begin{pmatrix} D^{-1} & 0 \\ 0 & I_{M-m} \end{pmatrix} U' \mathbf{x}_t + \begin{pmatrix} 0 \\ I_{M-m} \end{pmatrix} c_t. \]

Fortunately, such rotations do exist, and it is given by singular value decomposition (SVD), more specifically for the mixing matrix \( A \), the SVD of \( A \) is
\[ A = U \begin{pmatrix} D & 0 \end{pmatrix} V', \]
Then we have the null-space representation for overcomplete ICA:
\[ \mathbf{s}_t = V \begin{pmatrix} \left( D^{-1} \right) U' \mathbf{x}_t + \begin{pmatrix} 0 \\ I_{M-m} \end{pmatrix} c_t \end{pmatrix} = V_1 D^{-1} U' \mathbf{x}_t + V_2 c_t, \] (2)
where \( V_1 = (v_1 \cdots v_m) \); \( V_2 = (v_{m+1} \cdots v_M) \), and \( v_i \) is the \( i \)th column of \( V \). Clearly, \( V_1 \) is a basis for the row space of \( A \), and \( V_2 \) is a basis for the null space of \( A \), and \( c_t \) is the vector of coordinates in the null space.

With this representation, we are ready to reformulate the overcomplete ICA problem as a standard latent variable/missing data problem. Assume the joint distribution of \( \mathbf{s}_1, \cdots, \mathbf{s}_T \) to be \( P(\mathbf{s}_1, \cdots, \mathbf{s}_T) \). Then under the linear mapping in equation (2), this distribution gives rise to a joint distribution of \( c_1, \cdots, c_T \) and \( \mathbf{x}_1, \cdots, \mathbf{x}_T \). To be more specific,
\[ P(\mathbf{s}_1, \cdots, \mathbf{s}_T) | ds_1 \cdots ds_T = P(c_1, \cdots, c_T; \mathbf{x}_1, \cdots, \mathbf{x}_T) dc_1 \cdots dc_T d\mathbf{x}_1 \cdots d\mathbf{x}_T, \]
and from (2), we have the Jacobian
\[ ds_t = |V \begin{pmatrix} D^{-1} U^T & 0 \\ 0 & I_{M-m} \end{pmatrix}| d\mathbf{x}_t dc_t = |D|^{-1} d\mathbf{x}_t dc_t. \]
Thus
\[ P(c_1, \cdots, c_T, \mathbf{x}_1, \cdots, \mathbf{x}_T | A) = P(\mathbf{s}_1, \cdots, \mathbf{s}_T | D)^{-T}. \]
Here \( \mathbf{x}_1, \cdots, \mathbf{x}_T \) are the observations; \( c_1, \cdots, c_T \) are the latent variables, and the matrix \( A = U \begin{pmatrix} D & 0 \end{pmatrix} V' \) is the unknown parameter. For technical convenience, we may treat this problem in a Bayesian framework by putting uniform priors on \( U \) and \( V \), and uniform priors on \( \log(D) \). The reason for working on the log scale of \( D \) is that \( D \) is a scaling matrix. This seems consistent with the natural gradient of Amari, Cichocki & Yang (1996) (see also Hyvärinen & Oja, 1997).

In this Bayesian latent variable/missing data framework, the inference is based on the posterior distribution of \( U, V \) and \( \log(D) \), and the conditional distribution of \( c_1, \cdots, c_T \), given \( \mathbf{x}_1, \cdots, \mathbf{x}_T \). The computation can be accomplished by the data augmentation algorithm of Tanner & Wong (1987), which is a stochastic version of the EM algorithm (Dempster, Laird, & Rubin, 1977), and which is also a special case of the Gibbs sampler or Markov chain Monte Carlo. Specifically, the algorithm iterates of the following two steps:
1. Recovering \( s_t \) by sampling from \( P(c_1, \cdots, c_T \mid x_1, \cdots, x_T, A) \), i.e., the predictive distribution of the latent variables given the observed data and the parameter.

2. Estimating \( A \) by sampling from \( P(U; V, \log(D) \mid e_1, \cdots, e_T, x_1, \cdots, x_T) \).

The two sampling steps can be changed to maximization steps which often give similar results. In the next section, we shall study the two steps in detail.

3 The Inhibition Algorithm and the Givens Sampler

For the first step in the algorithm outlined above, we have the following inhibition algorithm. Since the null-space representation is

\[
\mathbf{s}_t = V_1 D^{-1} U' \mathbf{x}_t + V_2 c_t,
\]

to recover the sources is the same as to find the coefficient vectors, \( c_1, \cdots, c_T \). Here we draw \( c_1, \cdots, c_T \) from its conditional distribution, \( P(c_1, \cdots, c_T \mid x_1, \cdots, x_T, A) \), which can be accomplished by a sequence of Langenvin-Euler moves. This algorithm essentially is a stochastic gradient descent. Specifically, assume the target distribution to be \( \pi(\mathbf{c}) \propto \exp(-H(\mathbf{c})) \), where \( \mathbf{c} = (c_1, \cdots, c_T) \). Suppose at step \( \tau \) of the Langenvin-Euler process, the value of \( \mathbf{c} \) is \( \mathbf{c}(\tau) \) (note that \( \mathbf{c}(\tau) \) should not be confused with \( c' \). \( c' \) is the null-space coordinate for \( \mathbf{s}_t \), whereas \( \mathbf{c}(\tau) \) collects the values of \( (c_1, \cdots, c_T) \) at \( \tau \)th step of Langenvin-Euler move.) Then the next step \( \mathbf{c}_{\tau+1} \) is

\[
\mathbf{c}_{\tau+1} = \mathbf{c}_\tau - \frac{1}{2} \frac{\partial H(\mathbf{c})}{\partial \mathbf{c}} h + \sqrt{2h} \mathbf{Z}_\tau,
\]

where \( \mathbf{Z}_\tau \) is white noise, and \( h \) is the small time increment.

When \( c_t \) moves in the null space spanned by \( V_2 \), the components of \( \mathbf{s}_t \) change correspondingly by inhibiting or exciting each other according to \( V_2 \). The null-space representation seems to provide a mathematical explanation of lateral inhibition in this particular context, and we therefore call this algorithm the inhibition algorithm.

For the second step of the data augmentation algorithm, we need to sample the orthogonal matrices \( U \) and \( V \), and the diagonal matrix \( D \). First, let’s consider sampling \( D \) conditioning on everything else. As we mentioned earlier, we work on the log scale with \( w_i = \log(d_i) \), and \( w_1, \cdots, w_m \) satisfy \( w_1 > \cdots > w_m \). The prior of \( w_1, \cdots, w_m \) is assumed to be uniform with the order constraint. Therefore the posterior of \( w_1, \cdots, w_m \) is

\[
P(w_1, \cdots, w_m \mid x_1, \cdots, x_T, c_1, \cdots, c_T, U, V) \\
\propto P(x_1, \cdots, x_T, c_1, \cdots, c_T \mid A) \\
= P(s_1, \cdots, s_T \mid D)^{-T} \\
= P(s_1, \cdots, s_T) \exp(-\sum_{i=1}^{m} w_i).
\]

Hence the MAP (maximum a posteriori) estimate, \( \hat{\mathbf{w}} \), of \( \mathbf{w} = (w_1, \cdots, w_m)' \) is found by solving the following equation:

\[
\frac{\partial \log P(s_1, \cdots, s_T)}{\partial \mathbf{w}} = T \frac{\partial \sum_{i=1}^{m} w_i}{\partial \mathbf{w}}.
\]

After locating the MAP, we can approximate the posterior distribution of \( \mathbf{w} \) by a multivariate normal distribution with covariance matrix \( \Sigma = \text{Var}(\frac{\partial \log P(s)}{\partial \mathbf{w}})^{-1} \). Hence we draw \( \mathbf{w} \) from the multivariate normal distribution with mean vector \( \hat{\mathbf{w}} \) and covariance matrix \( \Sigma \). Of course, we could apply the Metropolis-Hasting step to make the sampling exact. Given the number of observations, however, this seems unnecessary.
The sampling of the rotation matrices $U$ and $V$ conditioning on everything else is a bit trickier, because the columns of $U$ and $V$ are orthogonal to each other, and we have to maintain the orthogonality while updating $U$ and $V$. This can be accomplished as follows. Suppose we want to update $U$. We can randomly pick two columns, $u_i$ and $u_j$, of $U$, and then rotate the two column vectors by an angle $\theta$ on the plane spanned by the two vectors, i.e.

$$u_i \leftarrow u_i \cos(\theta) + u_j \sin(\theta),$$

$$u_j \leftarrow -u_i \sin(\theta) + u_j \cos(\theta).$$

The distribution of $\theta$ can be easily derived from the posterior distribution of $U$ given everything else. More specifically let $U(i, j, \theta)$ be the updated $U$ with above given notations. Then

$$P(\theta) = P(c_1, \cdots, c_T, x_1, \cdots, x_T | A = U(i, j, \theta) (D \ 0 \ V)).$$

Thus, we still maintain the orthogonality of all the column vectors, and the idea is very much like playing with the magic square. Mathematically, this corresponds to multiplying $U$ by a Givens rotation matrix. The algorithm is the same for $V$. We call this algorithm the Givens sampler.

### 4 Experiments on Blind Source Separation

Now we apply the methodology developed above to the blind separation algorithm. We shall investigate the unsupervised learning problem in our future work.

#### 4.1 Sparse sources

Lee et al. (1999) studied blind source separation based on the algorithm of Lewicki & Sejnowski (2000). In their work, three independent speech signals mixed into two receivers could be separated under double exponential prior of $s_d$ where there is noise in observations. We studied the problem where there is no observation noise.

Here we assume the observations are mixed by the following equation (following Lee et al., 1999):

$$\begin{pmatrix}
  x_{1t} \\
  x_{2t}
\end{pmatrix} = \begin{pmatrix}
  0 & 1/\sqrt{2} & 1/\sqrt{2} \\
  1 & 1/\sqrt{2} & -1/\sqrt{2}
\end{pmatrix} \begin{pmatrix}
  s_{1t} \\
  s_{2t} \\
  s_{3t}
\end{pmatrix} = \begin{pmatrix}
  0 & 0.7071 & 0.7071 \\
  1 & 0.7071 & -0.7071
\end{pmatrix} \begin{pmatrix}
  s_{1t} \\
  s_{2t} \\
  s_{3t}
\end{pmatrix}.$$

(3)
Three independent speech signals with sample size 10000 are considered here (Top row of Figure 1), and they are mixed into two sequences of observations by equation (3). First we assume the distribution of $s_{it}$ to be double exponential distribution,

$$P(s_{it}) \propto \exp(-\alpha|s_{it}|),$$

and the parameter $\alpha$ is assumed to be known. We iterated the algorithm 300 times. The estimated mixing matrix is

$$
\begin{pmatrix}
-0.0396 & 0.6400 & 0.8631 \\
0.8272 & 0.8823 & -0.9080
\end{pmatrix},
$$

and the recovered sources are shown in middle row of Figure 1. Recoveries of the second and the third sources are fine, while the recovery of the first source only capture the overall structure.

Besides the double exponential distribution, mixture normal distribution is another possible choice (Olshausen & Milman, 2000). The density of a mixture normal distribution is

$$P(s_{it}) = pf_0(s_{it}) + (1 - p)f_1(s_{it}),$$

where $p$ and $1-p$ are probabilities of the two mixture components, $f_i$ is the density function of the normal distribution with mean zero and variance $\sigma_i^2$, and $\sigma_0^2 < \sigma_1^2$. Here we also assume that the parameters, $p$, $\sigma_0$ and $\sigma_1$ are known. After 300 iterations, the estimated mixing matrix is

$$
\begin{pmatrix}
-0.0328 & 0.6356 & 0.7828 \\
0.9890 & 0.8037 & -0.6669
\end{pmatrix},
$$

and the recovered sources are shown in Figure 1 (Bottom row).

### 4.2 Auto-regressive sources

In last subsection, we assume that $s_1, \cdots, s_T$ are statistically independent. This assumption is clearly not true for the real speech signals. In this section, we assume that signals $s_{i1}, \cdots, s_{it}$ come from an AR($d$) model, (e.g. Vermaak, Niranjan & Godsill, 1998, and Andrieu, & Godsill, 1999):

$$s_{it} = \phi_{i1} s_{i(t-1)} + \cdots + \phi_{id} s_{i(t-d)} + z_t,$$

where $\phi_{i1}, \cdots, \phi_{id}$ are the autoregressive coefficients, and $z_t$ are i.i.d. normal distribution with mean 0 and variance $\sigma^2$. We also assume all the autoregressive coefficients $\phi_{ij}$ and $\sigma^2$ are known here.
First we perform a simulation study with three sources generated from three different AR models with order $d = 3$ and 1000 samples (see Figure 2), and observations are mixed by equation (3). After 100 iterations, the estimated mixing matrix is

$$
W \approx \begin{pmatrix}
0.0064 & 0.7039 & 0.6301 \\
1.0489 & 0.6358 & -0.7412
\end{pmatrix},
$$

and the recoveries are all closed to the original sources. Then we perform an experiment on real signals, where the original sources are a speech signal, a rain sound and a wind sound signal. The observations are mixed by the equation (3). After 300 iterations, the estimated mixing matrix is

$$
W \approx \begin{pmatrix}
0.0318 & 0.6760 & 0.8639 \\
1.0247 & 0.7202 & -0.6320
\end{pmatrix},
$$

and the recoveries of sources are shown in Figure 3.

5 Discussion

5.1 When noise is not zero

When the observation noise is not zero, $x_t = A s_t + \varepsilon_t$, where $\varepsilon_t$ are i.i.d. $\mathcal{N}(0, \sigma^2)$. The steepest gradient method without identifying $V_1$ and $V_2$ would slow down the convergence speed when $\sigma^2$ is small. This problem could be solved by conjugate gradient method which adapts the geometry of $||x_t - As_t||^2$. But null-space representation accomplishes this explicitly, and stochastic sampling algorithm can be easily designed.

The null-space representation needs to be augmented by adding the basis of the row space of the mixing matrix $A$, and $V_1$ in the SVD of $A$ is such a basis. Namely, let $A = U (D \ 0) (V_1 \ V_2)$. Then we have

$$
s_t = A^{-1} x_t + V_1 e_t + V_2 c_t,
$$

where $c_t$ is a $m$ dimensional coefficient vector in the null space of $A$. The benefit of this representation is that $c_t$ traces out the ridge of $||x_t - As_t||^2$, while $e_t$ traces out the side profile of $||x_t - As_t||^2$. That is, $V_1$ and $V_2$ better adapt the geometry of the problem. The algorithm is similar to that for the noiseless situation.

5.2 Conclusion

In this paper, we present an explicit null-space representation for overcomplete ICA. Intellectually, this representation provides a natural transition from complete ICA to overcom-
plete ICA, and leads to a simple mathematical explanation of lateral inhibition. Computationally, it enables the estimation in the noiseless situation, and may lead to more efficient algorithm for situation with small observation noise.

References


