Mixture Matrix Identification of Underdetermined Blind Source Separation Based on Plane Clustering Algorithm*

Beihai Tan and Yuli Fu

College of Electronic and Communication Engineering, South China University of Technology 510640, China
tanpayhy@sohu.com, fuyuli@scut.edu.cn

Abstract. Underdetermined blind source separation and sparse component analysis aim at to recover the unknown source signals under the assumption that the observations are less than the source signals and the source signals can be sparse expressed. Many methods to deal with this problem related to clustering. For underdetermined blind source separation model, this paper gives a new plane clustering algorithm to estimate the mixture matrix based on sparse sources information. Good performance of our method is shown by simulations.

1 Introduction

Blind source separation (BSS) has been applied to many fields, such as, digital communication, image processing, array processing and biomedicine, and so on. Also, it has a lot of potential applications. Therefor, it has been a hot topic in signal processing and neural networks field [1-6].

Blind separation comes from cocktail problem [7], just to say, we only can restore source signals by gotten sensor signals, what's more, mixture channel and source signals' distributions are unknown. So the mathematics model of BSS is

$$X(t) = AS(t) + N(t)\ ,\ t = 1 \cdots T\ . \tag{1}$$

where $X(t) = [x_1(t), x_2(t) \cdots x_m(t)]^T$ is sensor signals, $A \in R^{m \times n}$ is mixture matrix, and $S(t) = [s_1(t), s_2(t) \cdots s_n(t)]^T$ is source signals, and $N(t) = [n_1(t), n_2(t) \cdots n_m(t)]^T$ is noise. BSS aims at restoring source signals only by known sensor signals, generally, we suppose noise doesn't exist.

In general, if m is more than n, that is, the number of sensor signals is more than that of source signals [8], it is overdetermined BSS. We consider the case that m is less than n in this paper, namely, underdetermined BSS. Although it is difficult to restore source signals, we can use some other information, such as, sparseness of

* The work is supported by the National Natural Science Foundation of China for Excellent Youth (Grant 60325310), the Guangdong Province Science Foundation for Program of Research Team (grant 04205783), the Natural Science Fund of Guangdong Province, China (Grant 05006508), the Specialized Prophasic Basic Research Projects of Ministry of Science and Technology, China (Grant 2005CCA04100).

D.-S. Huang, K. Li, and G.W. Irwin (Eds.): ICIC 2006, LNCIS 344, pp. 60 – 65, 2006.

source signals, to restore source signals, and if some source signals aren't sparse in time-domain, we can make them sparse through some transformation, such as, Fourier transformation or wavelet transformation, so BSS model is also written as

$$x(t) = a_1 s_1(t) + a_2 s_2(t) + \cdots a_n s_n(t), t = 1 \cdots T . \quad (2)$$

Where $x(t) = [x_1(t), \cdots x_m(t)]^T$, $a_i = [a_{1i}, \cdots a_{mi}]^T$.

2 Sparse Representation of Underdetermined Blind Separation

For underdetermined BSS, generally, some blind extraction algorithms [9], [10] are taken in past, but the algorithms can't realize to restore all source signals. In order to restore all source signals in underdetermined BSS, researchers make use of some characteristics of signals, for example, sparse analysis is adopted to make signals sparse representation, so some underdetermined BSS algorithms are successfully. Among the good algorithms there are Belouchrani's maximum likelihood algorithm [11] for discrete sources, Zibulevsky's sparse decomposition algorithm [3], Lee [12]□Lewicki [13] and Li' overcomplete representation algorithms [5] and Bofill' sparse representation in frequency domain [14].

Generally, sparse signal is that the one whose most sample points are zero or are near to zero, and a little sample points are far from zero. Here, we suppose that the source signal $s_i(t)$ is nonzero and the other source signals are zero or are near to zero at the time of t. So equation (2) can be written as

$$x(t) = a_i s_i(t) . \quad (3)$$

From above equation, we can known that a_i and $x(t)$ are collinear□so we can estimate mixture matrix $A = [a_1, a_2, \cdots a_n]$ by clustering $x(t)$ in all time. It is a very important algorithm for sparse component analysis solving underdetermined BSS, named by k-means clustering, and the algorithm includes two steps [5],[14], first, clustering centers are estimated by k-means clustering; second, source signals are estimated by known mixture matrix through linear programming.

Because the above algorithms require that source signals are very sparse, so there is a lot of restriction for application. Recently, Pando Georgiev puts forward a new sparse component analysis method for underdetermined BSS based the next conditions [15].

A1) the mixture matrix $A \in R^{m \times n}$ has the property that any square $m \times m$ submatrix of it is nonsingular.

A2) each column of the source matrix $S(t)$ has at most $m-1$ nonzero elements.

A3) the sources are sufficiently rich represented in the following sense: for any index set of $n-m+1$ elements $I = \{i_1, i_2 \cdots i_{n-m+1}\} \subset \{1,2,\cdots n\}$ there exist at least m column vectors of the matrix S such that each of them has zero elements in places with indexes in I and each $m-1$ of them are linearly independent.

For simplicity, we suppose $m=3, n=4$ to explain the paper's algorithm for the problem. If $m=3, n=4$, the equation (2) can be written as:

$$x(t)=a_1 s_1(t)+a_2 s_2(t)+a_3 s_3(t)+a_4 s_4(t), t=1\cdots T \ . \qquad (4)$$

where $x(t)=[x_1(t), x_2(t), x_3(t)]^T$ and $a_i=[a_{1i}, a_{2i}, a_{3i}]^T$, according to A2), if the i th source signal and the j th source signal are nonzero at the time of t, then

$$x(t)=a_i s_i(t)+a_j s_j(t), t=1\cdots T \ . \qquad (5)$$

From equation (5), we can know the sensor signal vector is in the same plane with vector a_i and vector a_j. Again, according to A1, every two columns in mixture matrix are independent, there are defined C_4^2 different planes by every two columns in mixture matrix. From equation (5), the mixture matrix $A=[a_1, a_2, a_3, a_4]$ can be estimated through plane clustering of sensor signals in no noise or little noise. Next, the plane clustering algorithm is given in detail and source signals are restored by it.

3 Mixture Matrix Identification Based on Plane Clustering

Pando Georgiev has proved that the mixture matrix is identifiable when the conditions A1 ,A2 ,A3 are met. Because the mixture matrix is very important, but Pando Georgiev doesn't give substantial algorithm for it, so this paper gives the substantial novel algorithm for estimating mixture matrix.

For simplicity, we still suppose $m=3, n=4$ to explain the algorithm. To identify $C_4^2=6$ planes, we turn to identify their six normal lines , and if their normal lines are identified, then we identify their planes.

In order to begin plane clustering, we initialize the sensor signals $x(t), \quad t=1\cdots T$, which are normalized. If $m=3$, a sensor signal correspond to one point in the spherical surface, and the points of the below half spherical surface need to turn them to above half spherical surface symmetrically. Then, the new sensor signals are

$$\hat{x}(t)=\begin{cases} \dfrac{x(t)}{\|x(t)\|} & if \;\; x_3(t)\geq 0. \\ -\dfrac{x(t)}{\|x(t)\|} & if \;\; x_3(t)<0. \end{cases} \quad , \; t=1\cdots T \ . \qquad (6)$$

Clustering $\hat{x}(t)$ is correspond to clustering $x(t)$, and the points will locate in the above half spherical surface which are in the same planes with the planes by every two columns of the mixture matrix respectively.

Similar to k-means cluster, normal lines clustering is to get their normal lines and modify them in clustering algorithm. For example, there are some initialized points $y(t)=[y_1(t), y_2(t), y_3(t)]^T$, $t=1,2,\cdots N_0$ in a plane.To identify its plane, we suppose its normal line is $\vec{n}_0=[n_{01}, n_{02}, n_{03}]^T$, According to inner-product' s definition,

$$(\vec{n}_0, y(t))=n_{01}\cdot y_1(t)+n_{02}\cdot y_2(t)+n_{03}\cdot y_3(t)=\|\vec{n}_0\|\cdot\|y(t)\|\times\cos\theta_{\vec{n}_0 y(t)} \ , \qquad (7)$$

where $\theta_{\vec{n}_0 y(t)}$ is the angle between the normal line $\vec{n}_0$ and the point $y(t)$, so $0 \le \theta_{\vec{n}_0 y(t)} \le \pi$, and $-1 \le \cos\theta_{\vec{n}_0 y(t)} \le 1$.

From equation (7), if we need to identify the plane composed of the points $y(t)$, $t = 1,2,\cdots N_0$, the normal line $\vec{n}_0 = [n_{01}, n_{02}, n_{03}]^T$ must be found to let $\theta_{\vec{n}_0 y(t)}$ tend to $\frac{\pi}{2}$ for any $t \in \{1,2,\cdots N_0\}$, because $\|\vec{n}_0\| = 1, \|y(t)\| = 1$, so just to say

$$\begin{aligned} \vec{n}_0 &= \arg\min_{\vec{n}_0} \sum_{t=1}^{N_0} \left|(\vec{n}_0, y(t))\right| \\ s.t. &\quad (n_{01})^2 + (n_{02})^2 + (n_{03})^2 = 1. \end{aligned} \tag{8}$$

Based on equation (8), the plane clustering algorithm is followed in detail.

1) Initialize the sensor signals $x(t),\ t = 1\cdots T$ using equation (6) to get new sensor signals $\hat{x}(t),\ t = 1\cdots T$.
2) Bring six initialized normal lines randomly, $\vec{n}_1, \vec{n}_2, \vec{n}_3, \vec{n}_4, \vec{n}_5, \vec{n}_6$.
3) Compute the inner-products of $\hat{x}(t),\ t = 1\cdots T$ and $\vec{n}_i, i = 1\cdots 6$ respectively, and take their absolute values, let $X_i = \{\hat{x}(t) \mid \left|(\hat{x}(t), \vec{n}_i)\right| < \left|(\hat{x}(t), \vec{n}_j)\right|, j \ne i\}$.
4) Modify the initialized normal lines, let $\vec{n} = [\sin\theta\cos\varphi, \sin\theta\sin\varphi, \cos\theta]$, $0 \le \theta \le \frac{\pi}{2},\quad 0 \le \varphi \le \pi$. For the sake of simplicity, the algorithm is shown by the following Matlab programme.

```
for i = 1:6
    n̂_i = n_i;
  for θ = 0:η_1:π/2
    for φ = 0:η_2:π
      if |(X_i, n)| < |(X_i, n_i)|
        n_i = n;
      end
    end
  end
end
```

Where η_1, η_2 denote step sizes respectively, $\left|(X_i, \vec{n})\right|, \left|(X_i, \vec{n}_i)\right|$ respectively denote the sums of inner-product's absolute value between all the elements of the set X_i and normal lines $\vec{n}$, and $\vec{n}_i$.

5) If $\left|\hat{\vec{n}}_i - \vec{n}_i\right| < \varepsilon_i,\quad i = 1\cdots 6$, the algorithm stops and ε_i is a given little value, otherwise, continue the step 3).

Because each column vector a_i in the mixture matrix compose a plane with other column $a_j (j \ne i)$, so a_i must be orthogonal with three normal lines among $\vec{n}_1, \vec{n}_2, \vec{n}_3, \vec{n}_4, \vec{n}_5, \vec{n}_6$ and the three normal lines must be in the same plane. That is to say, if we find any three coplanar normal lines, the columns $a_i (i = 1,\cdots 4)$ will be estimated.

4 Restoring Source Signals

Now, we suppose that the normal line is $\vec{n}_k\,(k \in \{1,\cdots 6\})$ of the plane composed of $a_i, a_j (i \neq j)$, and the set of the sensor signals is $X_l\,(l \in \{1,\cdots 6\})$ which is coplanar with $a_i, a_j (i \neq j)$. For any $x(t) \in X_l$, so

$$x(t) = a_i s_i(t) + a_j s_j(t)\,, \tag{9}$$

or

$$x(t) = A_{ij} s_{ij}(t)\,, \tag{10}$$

where $A_{ij} = [a_i, a_j], s_{ij}(t) = [s_i(t), s_j(t)]^T$, so

$$s_{ij}(t) = A_{ij}{}^{\#} x(t)\,. \tag{11}$$

Where $A_{ij}{}^{\#}$ denotes the generalized inverse matrix of A_{ij}. So only the i th source signal and the j th source signal have nonzero values gotten by equation (11) at the time of t, but zero for the other source signals at the time of t.

5 Simulations Results

In the experiment, a random 3×4 matrix brings for the simulation but meets the condition A1), and take $N = 1000$, four source signals are denoted in fig 1, The initialized mixture matrix is $\begin{bmatrix} -0.27574 & 0.18977 & -0.67493 & 0.86583 \\ 0.59016 & 0.28866 & -0.72862 & -0.12535 \\ 0.75874 & -0.93844 & 0.11652 & 0.48439 \end{bmatrix}$, and the estimated mixture matrix by the above algorithm is $\begin{bmatrix} 0.67479 & 0.86533 & -0.27554 & -0.19024 \\ 0.7288 & -0.12444 & 0.59033 & -0.28914 \\ -0.11622 & 0.48551 & 0.75867 & 0.93819 \end{bmatrix}$.

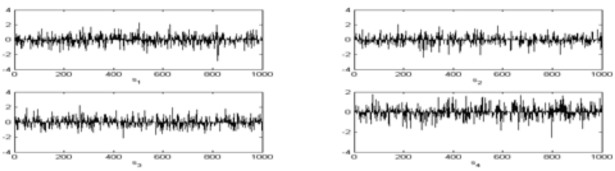

Fig. 1. Four source signals

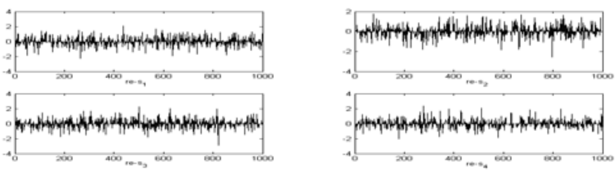

Fig. 2. Restored source signals

From the estimated mixture matrix and the above figures of restored source signals, the algorithm is successful except that the first and the fourth restored signals have sign difference from the third and the second source signals, which is allowed in BSS.

6 Conclusions

This paper gives a novel and substantial algorithm for estimating the mixture matrix and restoring the sparse source signals in underdetermined BSS. The algorithm is feasible and its good performance is shown in the simulation results, and it also easy to expand the algorithm to high dimension underdetermined BSS by sparse component analysis.

References

1. Hyvarinen, A., Oja, E.: Independent Component Analysis: Algorithms and Applications. Neural Networks, 13 (2000) 411-430
2. Xie, S. L., Zhang, J. L.: Blind Separation Algorithm of Minimal Mutual Information Based on Rotating Transform. Acta Electronic Sinica, 30 (5) (2002) 628-631
3. Zibulevsky, M., Pearlmutter, B.A.: Blind Source Separation by Sparse Decomposition in a Signal Dictionary. Neural computation, 13 (4) (2001) 863-882
4. Xie, S. L., He, Z. S., Gao, Y.: Adaptive Theory of Signal Processing. 1st ed. Chinese Science Press, Beijing (2006) 130-223
5. Li, Y., Cichocki, A., Amari, S.: Analysis of Sparse Representation and Blind Source Separation. Neural Computation 16 (2004) 1193–1234
6. Zhang, J. L., Xie, S. L., He, Z.S.: Separability Theory for Blind Signal Separation. Zidonghua Xuebao/Acta Automatica Sinica, 30 (3) (2004) 337-344
7. Jutten, C., Herault, J.: Blind Separation of Sources, Part I: An Adaptive Algorithm Based on Neuromimetic. Signal Processing, 24 (1991) 1-10
8. Zhang, J. L., Xie, S. L.: Multi-input Signal-output Neural Network Blind Separation Algorithm Based on Penalty Function. Intelligent and Complex Systems, 2 (2003) 353-362
9. Li, Y., Wang, J., Zurada, J. M.: Blind Extraction of Singularly Mixed Source Signals. IEEE Trans on Neural Networks, 11 (2000) 1413-1422
10. Li, Y., Wang, J.:Sequential Blind Extraction of Instantaneously Mixed Sources. IEEE Trans. Signal Processing, 50 (5) (2002) 997-1006
11. Belouchrani, A., Cardoso, J. F.: Maximum Likelihood Source Separation for Discrete Sources. In Proc. EUSIPCO, Edinburgh, Scotland (1994) 768-771
12. Lee, T. W., Lewicki, M.S., Girolami, M., Sejnowski, T. J.: Blind Source Separation of More Sources Than Mixtures Using Overcomplete Representation. IEEE Signal Processing Letter, 6 (1999) 87-90
13. Lewicki, M. S., Sejnowski, T. J.: Learning Overcomplete Representations. Neural computation, 12 (2000) 337-365
14. Bofill, P., Zibulevsky, M.: Underdetermined Source Separation Using Sparse Representation. Signal processing, 81 (2001) 2353-2362
15. Georiev, P., Theis, F., Cichocki, A.: Sparse Component Analysis and Blind Separation of Underdetermined Mixtures. IEEE Transactions On Neural Networks, 16 (4) (2005) 992-996