Jinwen Ma, Fei Ge, and Dengpan Gao

100871,

jwma@math.pku.edu.cn

Abstract.

1

The independent component analysis (ICA) [1,2] aims to blindly separate the independent sources $\mathbf{s}$ from their linear mixture $\mathbf{x} = \mathbf{A}\mathbf{s}$ via

$$\mathbf{y} = \mathbf{W}\mathbf{x}, \quad \mathbf{x} \in \mathbb{R}^m, \quad \mathbf{y} \in \mathbb{R}^n, \quad \mathbf{W} \in \mathbb{R}^{m \times n}, \tag{1}$$

where $\mathbf{A}$ is a mixing matrix, and $\mathbf{W}$ is the de-mixing matrix to be estimated. In the general case, the number of mixed signals equals to the number of source signals, i.e., $m = n$, and $\mathbf{A}$ is $n \times n$ nonsingular matrix. Although the ICA problem has been studied from di erent perspectives [3,4,5], it can be typically solved by minimizing the following objective function:

$$J = -H(\mathbf{y}) - \sum_{i=1}^{n} \int p(y_i; \mathbf{W}) \log p_i(y_i) dy_i, \tag{2}$$

where $H(\mathbf{y}) = -\int p(\mathbf{y}) \log p(\mathbf{y}) d\mathbf{y}$ is the entropy of $\mathbf{y}$, $p_i(y_i)$ is the predetermined model probability density function (pdf), and $p(y_i; \mathbf{W})$ is the probability distribution on $\mathbf{y} = \mathbf{W}\mathbf{x}$.

In the literature, how to choose the model pdfs $p_i(y_i)$ is still a key issue for the Eq.(2) based ICA algorithms [6,7]. In fact, there has not existed any

60471054.

Y. Hao et al. (Eds.): CIS 2005, Part I, LNAI 3801, pp. 915–920, 2005.

e cient method for the cases that sources of both super-Gaussian and sub-Gaussian coexist in an unknown manner. In order to solve this di cult problem, Xu, Cheung and Amari [7] summarized the one-bit-matching conjecture which states that “all the sources can be separated as long as there is a one-to-one same-sign-correspondence between the kurtosis signs of all source pdf’s and the kurtosis signs of all model pdf’s”. Clearly, this conjecture is important since, if it is true, the complicated task of learning the underlining distribution of each source can be greatly simplified to the task of learning only its kurtosis sign.

Since there have been many studies supporting the one-bit-matching conjecture, e.g. [8, 9], it is widely believed in the ICA community. Recently, Liu et al. [10] proved that under certain assumptions, the global minimum of the objective function with the one-bit-matching condition leads to a feasible solution of the ICA problem. Ma et al. [11] further proved that under the same assumptions, all the local minimums of the objective function on the two-source ICA problem with the one-bit-matching condition lead to the feasible solutions of the ICA problem. Moreover, many simulation experiments also showed that the ICA problem can be solved successfully via minimizing the objective function under the one-bit-matching condition. So, we can believe that the minimization of the objective function with the one-bit-matching condition can lead to a feasible solution of the ICA problem. On the other hand, if we can parametrize the model pdfs such that they can become super-Gaussian or sub-Gaussian adaptively and make them match the source pdfs according to the kurtosis signs during the learning process, the minimization of the objective function can also lead to a feasible solution of the ICA problem. Xu et al. [6] have designed a model pdf with mixer of Gaussians and have shown its capability to estimate the source distribution. However, their model is complicated.

where $p_{\rm super}$ is a super-Gaussian pdf, while $p_{\rm sub}$ is a sub-Gaussian pdf. α_i, β_i are parameters, with $\alpha_i, \beta_i \geq 0,\ \alpha_i + \beta_i = 1$. If α_i is greater than some constant value α_0 (determinded by the two fixed pdfs), $p_i(y_i)$ is super-Gaussian. Otherwise, if $\alpha_i < \alpha_0$, $p_i(y_i)$ is sub-Gaussian.

We select the fixed pdf as

$$p_{\rm super}(u) = \frac{1}{\pi}{\rm sech}(u), \qquad p_{\rm sub}(u) = \frac{1}{2}[p_{{\rm N}(1,1)}(u) + p_{{\rm N}(-1,1)}(u)],$$

where $p_{{\rm N}(\mu,\sigma^2)}$ denotes the Normal distribution.

In order to ensure that α_i, β_i satisfy the constraints, we use the following transformation:

$$\alpha_i = \frac{\exp(\gamma_{i1})}{\exp(\gamma_{i1}) + \exp(\gamma_{i2})}, \quad \beta_i = \frac{\exp(\gamma_{i2})}{\exp(\gamma_{i1}) + \exp(\gamma_{i2})},$$

so that α_i and β_j are equivalently expressed by free variables γ_{i1} and γ_{i2}. We can denote this flexible parametric mixture pdf by $p_i(y_i, \gamma_i)$ where $\gamma_i = (\gamma_{i1}, \gamma_{i2})$.

First, we must update $\mathbf{W}$ to learn a de-mixing matrix. We compute the derivatives of the objective function $J = J(\mathbf{W}, \gamma)$ with respect to $\mathbf{W}$, and apply the natural gradient algorithm to modify $\mathbf{W}$ in each step. The derivation is the same as in [4, 6, 8] and $\mathbf{W}$ is modified by

$$\Delta\mathbf{W} = \eta\left[\mathbf{I} + \Phi(\mathbf{y})\mathbf{y}^T\right]\mathbf{W}. \tag{4}$$

where η is the learning rate, $\Phi(\mathbf{y}) = [\phi_1(y_1), \cdots, \phi_n(y_n)]^T$, and

$$\phi_i(y_i) = \frac{p_i'(y_i, \gamma_i)}{p_i(y_i, \gamma_i)} = \frac{\alpha_i p'_{\rm super}(y_i) + \beta_i p'_{\rm sub}(y_i)}{\alpha_i p_{\rm super}(y_i) + \beta_i p_{\rm sub}(y_i)} \tag{5}$$

Meanwhile, we need to update the parameters of the model pdfs via the derivatives of $J(\mathbf{W}, \gamma)$ with respect to γ_{i1} and γ_{i2}. In fact, we have

$$\begin{aligned}\Delta\gamma_{i1} &= \eta\frac{\partial}{\partial\gamma_{i1}}\left(\sum_{l=1}^{n}\log p_l(y_l, \gamma_l)\right)\\ &= \eta\frac{p_{\rm super}(y_i) - p_{\rm sub}(y_i)}{\alpha_i p_{\rm super}(y_i) + \beta_i p_{\rm sub}(y_i)} \cdot \frac{\exp(\gamma_{i1})\exp(\gamma_{i2})}{(\exp(\gamma_{i1}) + \exp(\gamma_{i2}))^2}\\ &= \eta\frac{(p_{\rm super}(y_i) - p_{\rm sub}(y_i))\alpha_i\beta_i}{\alpha_i p_{\rm super}(y_i) + \beta_i p_{\rm sub}(y_i)}\end{aligned}$$

With the same derivation we can found out that $\Delta\gamma_{i2} = -\Delta\gamma_{i1}$.

Finally, we get to the following adaptive matching learning algorithm. At iteration k with an input $\mathbf{x}$, we can calculate $\mathbf{y}$ via $\mathbf{y} = \mathbf{W}\mathbf{x}$. Then, $\mathbf{W}$ and γ are modified by

$$\mathbf{W}^{(k+1)} = \mathbf{W}^{(k)} + \Delta\mathbf{W}, \qquad \gamma_{ij}^{(k+1)} = \gamma_{ij}^{(k)} + \Delta\gamma_{ij}. \tag{6}$$

2.2 Mixed Translated Super-Gaussian Model Pdf

Another flexible model pdf is constructed by two symmetrically translated pdfs:

$$p_i(y_i) = \frac{1}{4}\mathrm{sech}^2(y_i + \theta_i) + \frac{1}{4}\mathrm{sech}^2(y_i - \theta_i), \tag{7}$$

where $\theta_i \geq 0$ is the model parameter. As θ increase, the kurtosis decrease and the model pdf change from super-Gaussian to sub-Gaussian.

The derivation of the learning algorithm is quite similar to that in the previous subsection. We replace θ_i with e^{γ} in order to keep it positive. The procedure to update $\mathbf{W}$ is the same as Eq. (4) with

$$\phi_i(y_i) = \frac{-2\mathrm{sech}^2(y_i + \theta_i)\tanh(y_i + \theta_i) + 2\mathrm{sech}^2(-y_i + \theta_i)\tanh(-y_i + \theta_i)}{\mathrm{sech}^2(y_i + \theta_i) + \mathrm{sech}^2(y_i - \theta_i)}$$

Also, we need to update the parameters γ_i, and it turns out that

$$\begin{aligned} \Delta\gamma_i &= \eta\frac{\partial}{\partial\gamma_i}\left(\sum_{l=1}^{n}\log\left\{\frac{1}{4}\mathrm{sech}^2(y_i + \mathrm{e}^{\gamma}) + \frac{1}{4}\mathrm{sech}^2(y_i - \mathrm{e}^{\gamma})\right\}\right) \\ &= -2\eta\theta_i\frac{\mathrm{sech}^2(y_i + \theta_i)\tanh(y_i + \theta_i) + \mathrm{sech}^2(-y_i + \theta_i)\tanh(-y_i + \theta_i)}{\mathrm{sech}^2(y_i + \theta_i) + \mathrm{sech}^2(y_i - \theta_i)}. \end{aligned}$$

_ .. . a . . a . . a . . . . .

First, we consider the ICA problem of seven independent sources including four super-Gaussian sources (generated from the exponential distribution $E(0.5)$, the Chisquare distribution $\chi^2(6)$, the gamma distribution $\gamma(1,4)$ and the F distribution $F(10,50)$, respectively) and three sub-Gaussian sources (generated from the beta distributions $\beta(2,2)$, $\beta(0.5,0.5)$, and the uniform distribution $U([0,1])$, respectively). For each source, 100000 i.i.d. samples were generated and further normalized with zero mean and unit variance. The mixing matrix $\mathbf{A}$ was randomly chosen.

We set the learning rate $\eta = 0.001$. $\mathbf{W}$ was initially set as an identity matrix, and the initial model parameters were chosen such that the initial kurtosis of each mixture pdf $p_i(y_i, \gamma_i)$ is nearly zero.

The result of the adaptive matching learning algorithm using two model pdfs, respectively given by Eq. (3) and Eq. (7) are shown below, with $\mathbf{W}_1$ denoting the final $\mathbf{W}$ got using the first model pdf and $\mathbf{W}_2$ using the second one.

$$\mathbf{W}_1\mathbf{A} = \begin{bmatrix} -0.0125 & -0.0100 & -0.0283 & -0.0027 & -0.0041 & -0.0145 & \boxed{-1.4867} \\ -0.0143 & 0.0021 & 0.0087 & \boxed{-1.5540} & 0.0034 & -0.0366 & -0.0077 \\ \boxed{1.7166} & -0.0193 & 0.0159 & -0.0190 & -0.0074 & 0.0333 & 0.0066 \\ -0.0179 & 0.0006 & \boxed{1.6702} & -0.0203 & 0.0387 & 0.0075 & -0.0111 \\ 0.0149 & \boxed{1.5592} & -0.0202 & 0.0055 & 0.0024 & 0.0090 & 0.0168 \\ -0.0009 & -0.0204 & -0.0336 & 0.0018 & \boxed{-1.4433} & -0.0060 & 0.0083 \\ -0.0188 & -0.0048 & -0.0157 & -0.0015 & -0.0143 & \boxed{-1.4397} & -0.0162 \end{bmatrix}$$

$$
\mathbf{W}_2\mathbf{A} = \begin{bmatrix}
\boxed{-0.9769} & -0.0094 & 0.0078 & -0.0027 & 0.0303 & 0.0178 & 0.0193 \\
-0.0190 & -0.0021 & -0.0117 & -0.0161 & \boxed{-1.4425} & -0.0165 & -0.0045 \\
-0.0046 & -0.0027 & -0.0072 & 0.0052 & -0.0062 & -0.0150 & \boxed{-1.7547} \\
-0.0061 & -0.0002 & \boxed{0.9820} & -0.0131 & 0.0217 & 0.0241 & 0.0199 \\
0.0159 & -0.0045 & -0.0258 & \boxed{0.9444} & 0.0186 & 0.0213 & 0.0443 \\
-0.0203 & -0.0220 & 0.0010 & -0.0054 & -0.0098 & \boxed{1.2511} & -0.0095 \\
0.0271 & \boxed{0.9799} & -0.0113 & 0.0030 & -0.0008 & -0.0176 & -0.0510
\end{bmatrix}
$$

For a feasible solution of the ICA problem, the obtained $\mathbf{W}$ should make $\mathbf{WA} = \mathbf{\Lambda P}$ satisfied or approximately satisfied to a certain extent, where $\mathbf{\Lambda P} = diag[\lambda_1, \lambda_2, \cdots, \lambda_n]$ with each $\lambda_i \neq 0$, and P is a permutation matrix. We can see, that our adaptive matching learning algorithm can solve this ICA problem of both super- and sub-Gaussian sources e ciently.

Next, we use audio data to perform the tests. Eight sound clips[1], each containing 100000 samples (at 22050Hz sample rate), were normalized and then mixed using an 8×8 random matrix. We process the mixed signals with our adaptive matching learning algorithms. We rearrange the output signals so that each output y_i matches the recovered source s_i. Figure 2 shows the wave forms of four of the eight sources and their corresponding recovered signals obtained by the algorithm given in Section 2.1.

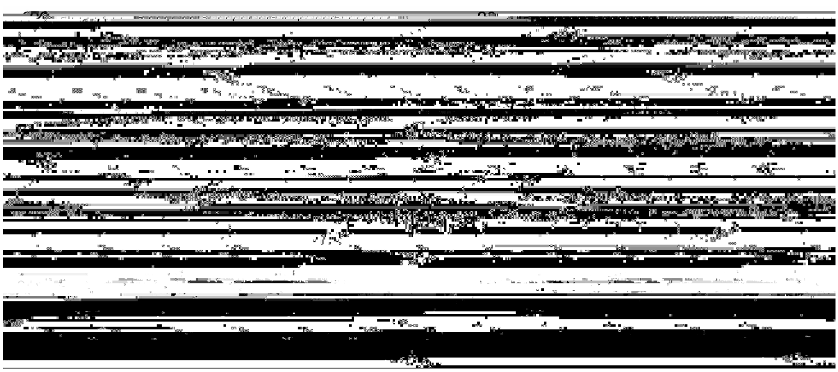

Fig. 1.

For comparision, we performed experiments using the Extended Infomax algorithm[8] and the Fast-ICA algorithm[5]. Then we calculate signal-to-noise ratio (SNR) to evaluate each recovered signal. The results are summarized in Table 1. We can find that on the average, our two adaptive matching algorithms perform better than the Extended Infomax and the Fast-ICA algorithms in this test.

---

1 http://www-bcl.cs.may.ie/~bap/demos.html

Table 1.

	[illegible] ( )							
A[illegible]	1	2	3	4	5	6	7	8