A QUASI-OPTIMALLY EFFICIENT ALGORITHM FOR INDEPENDENT COMPONENT ANALYSIS

John J. Weng and Nan Zhang

Michigan State University
Department of Computer Science and Engineering
East Lansing, MI, 48823
email:weng@cse.msu.edu, nanzhang@cse.msu.edu

ABSTRACT

We propose an incremental algorithm for independent component analysis (ICA), that is guided by the statistical efficiency. Starting from a ℓ^∞ norm sparseness measure contrast function, we derive the learning algorithm based on a winner-take-all learning mechanism. It avoids the optimization of high order non-linear function or density estimation, which have been used by other ICA methods, such as negentropy approximation, infomax, and maximum likelihood estimation based methods. We show that when the latent independent random variables are super-Gaussian distributions, the network efficiently extracts the independent components. We observed a much faster convergence than other ICA methods.

1. INTRODUCTION

Recently, there has been a resurgence of interest in independent component analysis (ICA) for Blind Source Separation (BSS). ICA has a wide application in signal and image processing, telecommunication, and medical data processing.

Independent Component Analysis (ICA) [1] is a technique to derive statistically independent components from random signals. The standard linear data model used in ICA is as follows. There is an unknown m-dimensional random signal source $\mathbf{s}$, whose components are mutually statistically independent. For every time instance $t = 1, 2, ...$, an unknown random sample vector $\mathbf{s}(t) = [s_1(t), s_2(t), ..., s_n(t)]^\top$ is generated from the signal source. There is an unknown $m \times m$ constant, full-rank, mixing matrix $\mathbf{A}$, which transforms each column vector $\mathbf{s}(t)$ into an observable vector $\mathbf{x}(t)$: $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$, where $\mathbf{x}(t) = [x_1(t), x_2(t), ..., x_n(t)]^\top$ is the m-dimensional vector of observed signals. The goal of ICA is to find a linear transformation $\mathbf{W}$ for the dependent sensory signals $\mathbf{x}(t)$ that makes the recovered signal $\mathbf{u}(t)$ as independent as possible: $\mathbf{u}(t) = \mathbf{W}\mathbf{x}(t) = \mathbf{W}\mathbf{A}\mathbf{s}(t)$, where $\mathbf{u}(t)$ is an estimation of the source signals. The source signals $\mathbf{s}(t)$ will be fully recovered when $\mathbf{W}$ is the inverse of $\mathbf{A}$. However, in practise, it is only possible that $\mathbf{u}(t)$ is recovered up to a permutation and scaling factors.

Many existing ICA methods aim to optimize certain contrast function with respect to the component matrix $\mathbf{W}$. The contrast function can be kurtosis or negentropy in maximizing non-Gaussianality methods, mutual information in infomax method, Maximum Likelihood Estimation (MLE), or sparseness measure (see [2] for a survey). For example, a typical method maximize or minimize the expectation of a non-linear and non-quadratic function G, i.e. $E\{G(\mathbf{W}^\top \mathbf{x})\}$ with respect to $\mathbf{W}$, where $G(\mathbf{W}^\top \mathbf{x})$ gives the information of high order statistics. The choice of G is crucial, since the asymptotic variance, robustness and convergence speed of the estimation depends on it.

In this study, we propose a novel sparseness measure based on ℓ^p norm, and use the sparseness measure as the contrast function to derive a fast ICA algorithm. To obtain a fast convergence, we utilize the concept of statistical efficiency, based on which uses amnesic mean instead of conventional learning rate, which results in the fast convergence.

The paper is organized as follows: In Section 2 we propose the contrast function. In Section 3 we discuss the efficiency of a estimator. We give the proposed algorithm in Section 4, the experiment results in Section 5, and comparison with existing ICA algorithms in Section 6. Section 7 provides conclusions.

2. ℓ^∞ NORM SPARSENESS MEASURE

Sparseness is an important property of super-Gaussian random variable. Maximizing sparseness is a desirable strategy that is used by different ICA algorithms. But the measurement of sparseness is not unique. Many heuristic measurements of sparseness were proposed. For example, Olshausen & Field [3] proposed the following form:

$$\text{Sparse}\,(\mathbf{u}) = -\sum_i S\,(u_i), \tag{1}$$

where Sparse $(\mathbf{u})$ is defined as the sparseness of random vector $\mathbf{u}$; $S(u_i)$ is a component-wise nonlinear function, and u_i are the recovered signals. Olshausen & Field suggest $S(u_i)$ to be an even nonnegative function, e.g. $-e^{-u^2}$, $\log\left(1+u^2\right)$, or $|u|$. Intuitively, when there are more neurons (elements in random vector $\mathbf{u}$) firing at the same time, the larger the function $\sum_i S\left(u_i\right)$ would be. Therefore, maximizing Eq. 1 will maximize the sparseness.

Karvanen et al. [4] generalized the measure of sparseness to ℓ^p norm criteria:

$$\text{Sparse}\left(\mathbf{u}\right) = E\left\{\left(\sum_i |u_i^p|\right)^{\frac{1}{p}}\right\}. \tag{2}$$

Olshausen & Filed's sparseness measure is just a special case of ℓ^p norm, where p is equal to 1 and $|u|$ being as the S function. When $p=4$, the ℓ^p norm is related to the widely used non-Gaussianality measure kurtosis, which is defined as $\text{kurt}\left(u\right) = E\left\{u^4\right\} - 3$, where u has unit variance.

Karvanen & Cichocki suggested the range of p should be in $(0,1]$, and particularly, a smaller p, such as $p=0.1$ or $p=0.01$, should be used. *However we found when $p \to \infty$, it gives a good sparseness measure.* Suppose we have the joint distribution of two independent Laplace random variables. Each sample is rotated by a rotation matrix. The mean of ℓ^p norms of the rotated samples can be computed. Figure 1 displays the mean of ℓ^p norm of different rotation angles with different p values. It is obvious that the ℓ^2 norm is a straight line, because rotation will not change the Euclidean distance. The extrema of ℓ^p norm curve are inverted on the two side of the ℓ^2 norm. Then, to find the independent component (in this case it is along the 20 degree and 110 degree directions), one needs to find the minima of the ℓ^p norm with $p < 2$ or maxima of ℓ^p norm with $p > 2$. Of course, this is only true for super-Gaussian symmetrical distributions.

Another issue is the robustness of the estimation when noise presents. From Figure 1, we can see almost all the norm curves agree on the same optima position (except ℓ^2 norm), but the norm curves with greater difference between the maxima and minima will be more robust when in the presence of noise. So the choice of p can be either close to zero or close to infinity. Karvanen & Cichocki suggested smaller p, such as 0.1 or 0.01.

On the other hand, let $p \to \infty$ we have:

$$\lim_{p\to\infty}\left(\sum_i |u_i|^p\right)^{\frac{1}{p}} = \max\{|u_i|\}.$$

So it leads to a simple computation. It can be shown that the gradient of infinity norm also has a simple form. Thus the optimization process is relatively easy for the ℓ^∞ norm

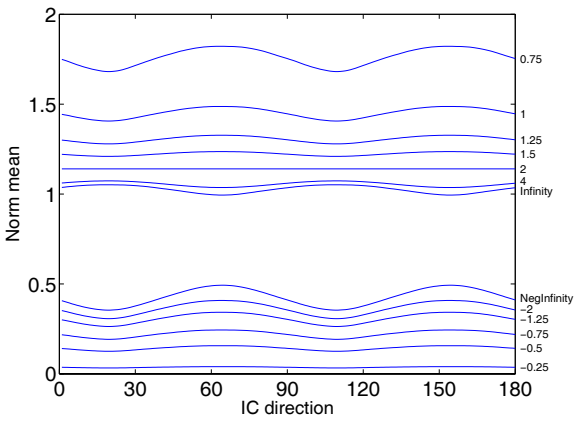

Fig. 1. The ℓ^p norm criteria functions under different p. The data set is projected onto a series of orthogonal basis, and then the expectation of ℓ^p norms of all the projected samples are computed.

sparseness measure. We can define the following contrast function

$$J(\mathbf{u}) = E\left\{\lim_{p\to\infty}\left(\sum_i |u_i|^p\right)^{\frac{1}{p}}\right\} = E\left\{\max_i |u_i|\right\}. \tag{3}$$

Maximizing this function solves the maximizing sparseness problem. Alternatively, the contrast function is written as

$$J(\mathbf{W}) = \int \max_j \left[\left(\mathbf{w}_j^\top \mathbf{x}\right)^2\right] p\left(\mathbf{x}\right) d\mathbf{x}, \tag{4}$$

where we replace the $|u_j|$ with $|u_j|^2 = (\mathbf{w}_j^\top \mathbf{x})^2$ for mathematical convenience, since the absolute function is not differentiable. Since both functions $|x|$ and x^2 are monotone of the same sign around the origin $x=0$, Eqs. 3 and 4 are equivalent in terms of maximized solution. So our goal is to maximize the contrast function with respect to $\mathbf{W}$. It can be shown that the gradient is given by

$$\partial J(\mathbf{W})/\partial \mathbf{w}_j = 2\delta_{cj}(\mathbf{w}_c^\top \mathbf{x})\mathbf{x}, \tag{5}$$

where $c = \arg\max_j[(\mathbf{w}_j^\top \mathbf{x})^2]$, δ_{cj} is the Kronecker delta: $\delta_{cj} = \{1 \text{ if } c=j, 0 \text{ otherwise}\}$.

3. ESTIMATOR EFFICIENCY

Suppose there are two statistical estimators Γ_1 and Γ_2 for estimating parameter θ. If $E\left\|\Gamma_1 - \theta\right\|^2 < E\left\|\Gamma_2 - \theta\right\|^2$, Γ_1 is said to be more statistically efficient than Γ_2.

We consider $(\mathbf{w}_i^\top \mathbf{x})\mathbf{x}$ with $\|\mathbf{w}_i\| = 1$ as an "observation." The goal is to get the mean of this observation, while $\mathbf{w}_i$ is estimated incrementally. It is known that for many distributions, the sample mean is the most efficient estimator for the mean of the random variable. When the distri-

bution is unknown, the sample mean is the best linear estimator, which results in the minimum error variance. For many distributions, the sample mean reaches of approaches the Cramér-Rao bound (CRB).

Then an efficient estimator is one that has the least variance from the real parameter $\mathbf{W}$, and its variance is bounded below by the CRB. Thus, we estimate an independent component vectors $\mathbf{w}_i$ by the sample mean of the observation $(\mathbf{w}_i^\top \mathbf{x})\mathbf{x}$.

The sample mean uses a batch method. For incremental estimation, during which $\mathbf{W}$ is continuously improved, we use what is called an amnesic mean [5].

$$\bar{x}^{(n)} = \alpha\,(n)\,\bar{x}^{(n-1)} + \beta\,(n)\,x_n, \tag{6}$$

where $\bar{x}^{(n)}$ is the mean at the n-th iteration, x_n is the n-th sample, and $\alpha\,(n)$ and $\beta\,(n)$ are defined by

$$\alpha\,(n) = \frac{n-1-\mu(n)}{n}, \tag{7}$$

and

$$\beta\,(n) = \frac{1+\mu(n)}{n}. \tag{8}$$

$\mu(n)$ is a non-negative small function that discounts old estimate and gives more weight to the new observation $\mathbf{x}_n$ at time n. When $\mu(n) \equiv 0$, $\bar{\mathbf{x}}^{(n)}$ is exactly the sample mean.

The algorithm is guided by the statistical efficiency, but it is not absolutely the most efficient one, because

1. the true distribution of the observation is unknown;
2. the distribution changes with $\mathbf{W}$ being incrementally estimated and therefore,
3. amnesic average is used to gradually discount "old" observations, which reduces the statistical efficiency moderately.

4. THE QUASI-OPTIMAL ICA ALGORITHM

The quasi-optimal ICA Algorithm is base on Eq. 5 and also considers the statistically efficient estimation. The algorithm computes the separating matrix $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_k]$, $k \leq m$, from whitened samples $\mathbf{x}_1, \mathbf{x}_2, ...$, where k is the number of principal component vectors determined by the pre-whitening step. The algorithm is shown in Algorithm 1.

5. RESULTS

We have tested the algorithm on a simulation of the cocktail party problem. Nine sound sources are mixed by an randomly chosen full rank matrix. Each sound source is 6.25 seconds long and the sampling rate is 8.0KHz in 8 bits mono format. Therefore, each sound source contains 50,000 values.

Algorithm 1 Quasi-optimal ICA

1: $\mathbf{w}_i = \mathbf{x}_i, i = 0, 1, 2, ..., k.$
2: $n_i = 1, i = 1, 2, ..., k.$
3: **for** $t = 1, 2, ...$ **do**
4: $\quad j = \arg\max_i \left\{ \frac{|\mathbf{w}_i(n_i)^\top \cdot \mathbf{x}_t|}{\|\mathbf{w}_i(n_i)\|} \right\}$
5: Update the independent component vector $\mathbf{w}_j(n_j)$ by the following updating rule:

$$\mathbf{w}_j(n_j{+}1) = \alpha(n_j)\mathbf{w}_j(n_j) + \beta(n_j)\frac{\mathbf{w}_j(n_j)^\top \cdot \mathbf{x}_t}{\|\mathbf{w}_j(n_j)\|}\mathbf{x}_t, \tag{9}$$

where $\mathbf{w}_j(n_j)$ is the component vector $\mathbf{w}_j$ after the n_j-th updating, $\alpha(n_j)$ and $\beta(n_j)$ are given in Eqs. 7 and 8, respectively.
6: $\quad n_j = n_j + 1.$
7: **end for**

Fig. 2(a) shows one of the nine original source signals. Fig. 2(b) displays one of the nine mixed sound signals. The mixed signals are first whitened, then we applied the proposed algorithm to the mixed sound signals. It is worth noting that the proposed algorithm is an incremental method. Therefore, unlike other batch ICA method doing iterations on the date set, we have used data only once and then discarded them. The result is shown in Fig. 2(c). The independent components quickly converge to the true ones, with a good approximation as early as 1.5 second.

6. COMPARISON

We have also tested the efficiency of the proposed algorithm for high dimensional data. It is known that the ICA algorithms are "data grizzlies". Typically, even for a low dimensional simulation task, ICA algorithms need thousands of samples to evaluate the independent components. Convergence speed in the number of samples used in training is a good evaluation of the efficiency of ICA algorithms.

In this experiment, we proceed to compare different ICA algorithms in terms of number of training samples required. We have chosen FastICA [6] algorithm and Extended Infomax [7] as the comparison benchmark due to their popularity and superior convergence properties. The original signals were drawn from a i.i.d Laplace (double exponential) random vector with dimensionality of 100. The original signals are mixed with a randomly chosen full rank matrix, whose dimensionality is 100×100. The data were then pre-whitened, and thus the mixing matrix is only a rotation transformation. We then applied the proposed algorithm to the mixed data. The result is shown in Fig. 3, where X axis marks the number of training samples and Y axis indicates the Basis Distance Index (BDI) between the current independent components (ICs) and the ground truth. The BDI is

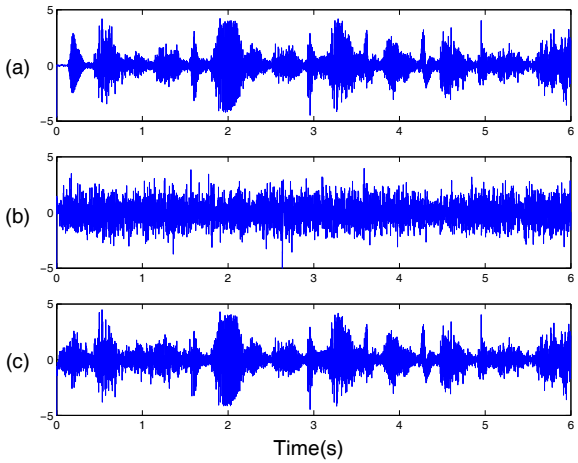

Fig. 2. Cocktail party problem. (a) A music sound clip in its original form. It is one of the nine sound sources. (b) One of the nine mixed sound signals. (c) The recovered music sound wave. Comparing to (a), the sound signal is recovered well after approximately 1.5 second.

the average angle (in radian) between the ground truth vectors and the corresponding estimated ICs. Thus, the lower BDI the better.

Apparently, the proposed method, which is referred as "LCA" in Figure 3, converges much faster than the benchmark methods, where "LCA with fixed m" and "LCA with dynamic m" are proposed algorithm with two different ways to compute the amnesic function $\mu(n)$. FastICA method gains good accuracy, better than two aforementioned LCA algorithms, but only after a large number of inputs. But another variate of the proposed algorithm referred as "LCA eliminating cells", which involves dynamic eliminating unnecessary components, outperforms the FastICA algorithm both in speed of convergence and the final accuracy. Extended Infomax algorithm needs much more samples, therefore it did not get near the true value in our high dimensional tests.

7. CONCLUSION

The proposed quasi-optimal ICA algorithm is simple and fast under multiple demanding computational requirements: high dimensional, incremental and free of higher order statistics computation. This is due to the simplicity of the ℓ^∞ norm sparseness measure and the most efficient estimation concept used in the algorithm design.

8. REFERENCES

[1] P. Comon, "Independent component analysis—a new concept?," *Signal Processing*, vol. 36, pp. 287–314, 1994.

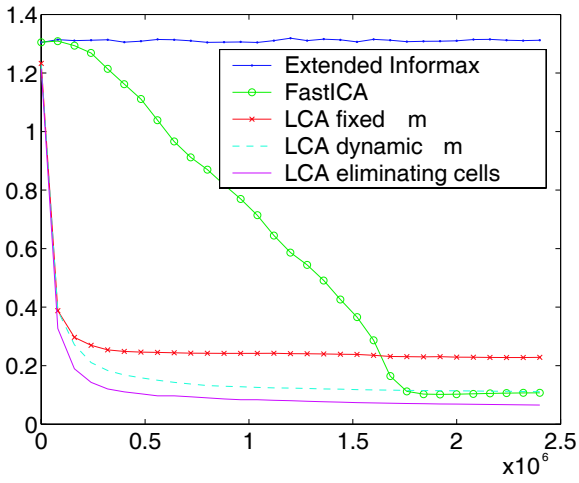

Fig. 3. Comparison between proposed ICA method and its variate with FastICA and Extended Infomax algorithm.

[2] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and application," *Neural Networks*, vol. 13, pp. 411–430, 2000.

[3] B.A. Olshausen and D.J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607–609, 1996.

[4] J. Karvanen and A. Cichocki, "Measuring sparseness of noisy signals," in *Proceedings of 4th International symposium on Independent Component Analysis and Blind Signal Seperation (ICA2003)*, 2003, pp. 125–130.

[5] J. Weng, Y. Zhang, and W.-S. Hwang, "Candid covariance-free incremental principal component analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 1034–1040, 2003.

[6] A. Hyvärinen, "Fast independent component analysis," in *Independent Component Analysis: Principles and Practice*, S. Roberts and R. Everson, Eds. Cambridge University Press, 2001, in press.

[7] T.-W. Lee, M. Girolami, and T. J. Sejnowski, "Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources," *Neural Computation*, vol. 11, no. 2, pp. 417–441, 1999.