ESANN'2000 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 26-28 April 2000, D-Facto public., ISBN 2-930307-00-5, pp. 123-128

A Neural Network for Undercomplete Independent Component Analysis

Lu Wei, Jagath C. Rajapakse

School of Applied Science, Nanyang Technological University
Singapore 639798
asjagath@ntu.edu.sg

Abstract: The existing independent component neural networks (ICNNs) in the literature need same number of output neurons as the input nodes to achieve independence among output activations. We present a technique to learn the undercomplete ICNNs to produce an output with an lower dimension than the input by using joint entropy of a multidimensional Gaussian to approximate the mutual entropy of the output. Our approach is not restricted by the squared Jacobian matrix of outputs with respect to the inputs, and gives a general rule and some criteria to extract both super- and sub-Gaussianly distributed signals and remove the Gaussian distributed noise. Simulation results with simulated signals and audio signals are provided.

Keywords: Independent component analysis, non-complete independent component analysis, mutual independence, blind signal processing, stability analysis.

1. Introduction

Independent Component Analysis (ICA) transforms a multivariate random signal into components that are mutually independent in complete statistical sense. Recently researchers proposed many neural network learning mechanisms to perform ICA based maximum entropy [1], Kullback-Leibler divergence minimization [2] and maximum likelihood [3]. However, most researches only considered to extract the full space of independent components, or the case where the number of network outputs (M) equaled the number of input signals (N). Such networks $(M = N)$ are said to perform *complete* ICA, however, $M \neq N$ is the general case in practice. For instance, the cocktail-party problem tries to recover M individual speech voices from $N(\neq M)$ mixture signals from microphones. When we have more input signals or mixtures than the actual sources to be extracted then ICA is said to be *undercomplete* $(M < N)$, e.g. functional brain imaging is a good application that a large number of signals are accumulated from the brain where only a few signals come from activated brain voxels [4]. ICA is said to be *over complete* when the observed signals are less than the actual sources $(N < M)$.

This paper presents a gradient approach to undercomplete independent component neural network to extract independent components from signals with certain distribution while removing noises in the undercomplete ICA cases. Our method forms the contrast function based on component mutual information (CMI) between joint entropy and product of marginal entropies. By minimizing the component mutual information, we can find the linear demixing matrix for signal separation. The criteria for optimal learning rule and the algorithm's stability and convergence are discussed. All of Gaussian, sub- and super-Gaussian distributed signals can be estimated by our approach. Experiments show the applications of less dimensional signal separation from large amount data of observed signal and noise cancellation. We also state the reason of inapplicability of our approach to overcomplete ICA.

2. Problem and Basic Techniques

Let the time varying input signal be $\mathbf{x} = (x_1, x_2, \ldots, x_N)^{\mathrm{T}}$ and the interested signal consisting of independent components (ICs) or variables be $\mathbf{c} = (c_1, c_2, \ldots, c_M)^{\mathrm{T}}$, and generally $N \neq M$. The signal $\mathbf{x}$ is considered to be a linear mixture of independent components $\mathbf{c}$:

$$\mathbf{x} = \mathbf{A}\mathbf{c} \tag{1}$$

where $\mathbf{A}$ is an $N \times M$ mixing matrix.

The goal of general ICA is to obtain a linear $K \times N$ demixing matrix $\mathbf{W}$ to recover the independent components $\mathbf{c}$ or part of the interested components, i.e. $K \leq M$, with a minimal knowledge of $\mathbf{A}$ and $\mathbf{c}$. However, the exact components $\mathbf{c}$ are indeterminant because of possible permutation and dilation. Nevertheless, the source signals are identifiable in this sense [5].

A single-layer ICNN is defined with K neurons and N input nodes. The weight matrix of the network is denoted $\mathbf{W} = \{w_{ij}\}_{\mathrm{K}\times\mathrm{N}}$. K is generally not equal to N. The network output $\mathbf{u}$ to represent the source components that are to be extracted is given by

$$\mathbf{u} = \mathbf{W}\mathbf{x} \tag{2}$$

Our algorithm requires a preliminary sphering or whitening of the input $\mathbf{x}$. Whitening transforms the original observed variables $\mathbf{v}$ to signal $\mathbf{x}$ such that the correlation matrix of $\mathbf{x}$ becomes the identity matrix: $E\{\mathbf{x}\mathbf{x}^{\mathrm{T}}\} = \mathbf{I}$. This transform can always be done using any well-known PCA technique.

3. General Learning Algorithm

The contrast function, component mutual information (CMI) between the output and its components, is defined in the sense of random variable's entropy.

$$\mathbf{CMI} = \sum_{i=1}^{K} H(u_i) - H(\mathbf{u}) \tag{3}$$

where $H(u_i)$ is the marginal entropy of component u_i and $H(\mathbf{u})$ is the output joint entropy. CMI has non-negative value and equals to zero when components are completely independent.

The joint entropy calculation is not difficult under the conventional ICA in the case of $K = N$ [6]. But the simple linear relationship between the densities of network's input and output is not valid any more under the general case of $K \neq N$. As we have some minimal knowledge about the components needed to be recovered, it is possible to have both super-Gaussian and sub-Gaussian distributed independent components appearing at the network output. Because in practice the joint statistics of uniting all output signals have characteristics close to Gaussian distribution, we can approximate that the joint output distribution is a multidimensional Gaussian. Then entropy $H(\mathbf{u})$ is approximated as

$$H(\mathbf{u}) \approx K(1 + \log 2\pi)/2 + \log(|\mathrm{Cov}(\mathbf{u})|)/2 \tag{4}$$

where $|\cdot|$ indicates the determinant of the matrix and $\mathrm{Cov}(\mathbf{u}) = E\{\mathbf{u}\mathbf{u}^{\mathrm{T}}\} = \mathbf{W}\mathbf{W}^{\mathrm{T}}$, since the input variable $\mathbf{x}$ is preprocessed by whitening.

With multivariate Gaussian assumption of joint distribution, **CMI** with respect to demixing matrix $\mathbf{W}$ can be approximated as:

$$\mathbf{CMI}(\mathbf{W}) \approx -\frac{1}{2}\log(|\mathbf{W}\mathbf{W}^{\mathrm{T}}|) - \sum_{i=1}^{K} E\{\log p_{u_i}(u_i)\} \tag{5}$$

By using the gradient descent approach to minimize **CMI** and replacing the expectation values by their instantaneous values, we have the stochastic gradient descent algorithm:

$$\Delta\mathbf{W} = \eta\{(\mathbf{W}^{+})^{\mathrm{T}} - \mathbf{\Phi}(\mathbf{u})\mathbf{x}^{\mathrm{T}}\} \tag{6}$$

where $\mathbf{W}^{+}$ is the pseudo-inverse matrix of $\mathbf{W}$, equals to $\mathbf{W}^{\mathrm{T}}(\mathbf{W}\mathbf{W}^{\mathrm{T}})^{-1}$ when $\mathbf{W}$'s rank is K and

$$\phi_i(u_i) = -p_i'(u_i)/p_i(u_i) = -(\partial p_i(u_i)/\partial u_i)/p_i(u_i). \tag{7}$$

The learning rule (6) can generally yield a correct solution $\mathbf{W}$, and hence theoretically equation (7) can work on any source distribution. Nevertheless, this choice bears some implementation difficulty because $p_i(u_i)$ is not known in advance.

4. Performance Analysis

In our approach, we consider a family of density functions with exponential power, which is generally given as

$$p_i(u_i) = \alpha \exp(\beta |u_i|^{\gamma}) \tag{8}$$

where α and β are constants to ensure $\int p_i(u_i)du_i = 1$ and γ is a positive parameter. A super-Gaussian density of positive kurtosis is obtained in the range of $0 < \gamma < 2$ whereas $\gamma = 2$ gives the Gaussian distribution. A sub-Gaussian density of negative kurtosis is obtained for $\gamma > 2$. From equation (7), the nonlinear function becomes

$$\phi_i(u_i) = -\beta |u_i|^{\gamma - 1} \tag{9}$$

The above equation implies that the nonlinear function $\phi_i(u_i)$ should grow slower than linearly to extract super-Gaussian components and the function grow faster than linearly to extract sub-Gaussian components. With certain nonlinear function by fixing β and γ, the network tends to find the components with closest density functions to the nonlinear function's derivation. Here, several rules are proposed for selecting the nonlinear function $\phi_i(u_i)$:

1. $\phi_i(u_i)$ is an odd monotonic increasing function. It should have equivalent order of u_i within range of $(0, 1)$ to extract super-Gaussian sources, and the order greater than 1 for sub-Gaussian sources.
2. $\phi_i(u_i)$ can loosely match with the density function in the same category, e.g. sub-Gaussian or super-Gaussian, to well separate the components.
3. For computational simplicity, the function should be chosen to compute fast, e.g. polynomial functions tend to be faster than hyperbolic tangent.
4. Because any ordinary method of optimization tends to first find maxima that have large basins of attraction, the components with density closest matching to $\Phi(\mathbf{u})$ hold the global minimum of **CMI**. The measure of closeness gives the order of components to be extracted. The algorithm also converges if it is initialized at a nearby region of one local minimum for separation solution.
5. The sign value of component's kurtosis can be the way to determine the output's density type as super- or sub-Gaussian.

If row and column dimensions of demixing matrix $\mathbf{W}$ are equal, $K = N$, i.e. complete ICA case, our algorithm from above discussion can produce the algorithms same as existing learning rules with certain nonlinear functions given as odd sigmoidal function by infomax method [1] and 11th-order polynomial function by minimum mutual information approach [2] and so on. Here, we choose the following nonlinear function for super-Gaussian and sub-Gaussian class components:

$$\phi_{\text{super}}(u_i) = \tanh(au_i), \qquad \phi_{\text{sub}}(u_i) = bu_i^3. \tag{10}$$

where a and b are the parameters to choose the suitable shape of distribution.

Essentially, the demixing matrix $\mathbf{W}$ learned from the above algorithm is the left pseudo-inverse matrix of $\mathbf{A}$ with full column rank of M, which matches the case of complete and undercomplete ICA. For full row rank mixing matrix of overcomplete ICA, its left generalized inverse matrix is ambiguous, which is not unique like its right pseudo-inverse matrix. Therefore, according to the theory of matrices, our approach is not suitable for overcomplete ICA.

The stability of our learning algorithm is another important role to ensure the algorithm successful. The equilibrium point of our algorithm is

$$(\mathbf{W}^+)^{\mathrm{T}} - E[\Phi(\mathbf{u})\mathbf{x}^{\mathrm{T}}] = \mathbf{0} \tag{11}$$

By linearizing equation (6) at this equilibrium point, we have the Hessian matrix of component mutual information function $\bigtriangledown^2\mathbf{CMI(W)}$. This shows that, only when the Hessian matrix is positive definite at the minimum point of contrast function, or all the eigen values of the operator $\partial((\mathbf{W}^{+})^{\mathrm{T}} - E[\Phi(\mathbf{u})\mathbf{x}^{\mathrm{T}}])/\partial\mathbf{W}$ have negative real parts, the equilibrium is asymptotically stable. It can be proven that the convergence is stable if the neuron with super-Gaussian nonlinear function $\phi_i(u_i)$ is corresponding to extract super-Gaussianly distributed signal and same condition is held for sub-Gaussianly signal in our algorithm.

5. Experiments and Results

The algorithms were simulated in MATLAB version 5. All signals are preprocessed by a whitening process to have zero mean and uniform variance. The performances of the network separating the signals into ICs were measured by an individual performance index (IPI) of the permutation error ϵ_i for ith output:

$$\epsilon_i = (\sum\nolimits_{j=1}^{n} \frac{|p_{ij}|}{\max_k |p_{ik}|}) - 1 \tag{12}$$

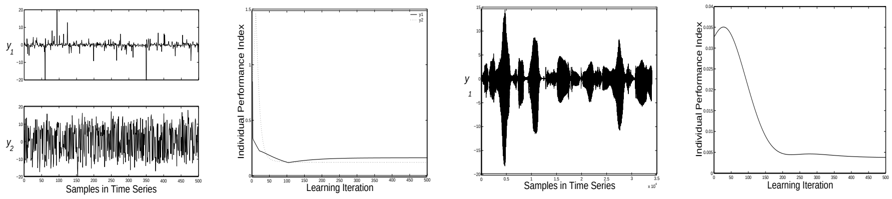
where p_{ij} are elements of the permutation matrix $\mathbf{P} = \mathbf{WA}$. IPI is close to zero when the corresponding output is close to an independent component.

Four independent random signals distributed in two sub- and two super-Gaussian manner were simulated. Their statistical configurations are similar as the simulation experiments in [6]. These source signals $\mathbf{c}$ were mixed with a random matrix to derive inputs to the network. The experiments trained the 2×4 matrix using a nonlinear function for super-Gaussianly signal in one neuron and sub-Gaussianly in another neuron. With the learnt network weight matrix, one super-Gaussian and another sub-Gaussian independent components were extracted at the network output. And the signal with the density function closer to nonlinear function ϕ_i is the one to be extracted at the output. The resulted output waveforms and IPI curves are shown in Figure 1(a) and (b) respectively. The super-Gaussian output has SNR 31.82dB and sub-Gaussian signal has 25.92dB. As seen, the extracted components were very close to the desired source signals.

Another experiment simulated that our neural network has the ability of the white noise cancellation for noisy speech signals. The simulation had original components of one real speech signal and a random generated Gaussian noise signal with normal distribution. Two components were mixed with a random matrix to derive input mixtures to the network. A two-to-one demixing matrix was trained by self-selecting the nonlinear function between sub-Gaussian and super-Gaussian based on the sign of the output signal's kurtosis. The result gave an output recovering the original speech signal with high SNR value of 38.75dB at the output neuron. It perfectly removed the white noise totally which was distorting the speech signal at the input of the neural network. Figure 1(c) and (d) illustrate the recovered speech signal waveform and the IPI curve respectively.

6. Conclusion

The above experiments show that our algorithm is successful in undercomplete ICA signal separation which is more generic than conventional ICA. Our



(a) (b) (c) (d)

Figure 1: (a) The output waveforms of one super-Gaussianly and another sub-Gaussianly signal from the separation. (b) The corresponding individual performance indices to these two outputs. (c) The output waveforms of recovered speech signal. (d) The network performance index for removing noise.

algorithm can recover both super-Gaussian and sub-Gaussian distributed signals from mixtures accurately. We also gave some general rules to choose the nonlinear functions in our algorithm to loosely match the distribution of components to be extracted. The noise can be discarded at the network output because usually the noise signal has Gaussian distribution which is not able to converge in our algorithm, because activation functions correspond to sub- and super-Gaussian signals.

The network output joint entropy can be further studied with more precise approximation. The stability and convergence of our algorithm need to be studied later. The learning of nonlinear function's parameters to match with the signal characteristics is also an interesting future research direction.

References

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

[2] S. Amari, A. Chchocki, and H. H. Yang. A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems 8*, 1996.

[3] T-W. Lee, M. Girolami, and T. Sejnowski. Independent component analysis using an extended informax algorithm for mixed sub-gaussian and super-gaussian sources. *Neural Computation*, 11(2):409–433, 1999.

[4] M. McKeown, S. Makeig, G. Brown, T-P Jung, S. Kindermann, A. Bell, and T. Sejnowski. Analysis of fMRI data by blind separation into independent spatial components. *Human Brain Mapping*, 6:160–188, 1998.

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

[6] Jagath C. Rajapakse and Lu Wei. Unified approach to independent component networks. In *Accepted for Second International ICSC Symposium on NEURAL COMPUTATION (NC'2000)*, 2000.