In this version a few typos have been corrected, relative to the printed version.

An Objective Function for Independence

Gonçalo Marques and Luís B. Almeida
INESC - Instituto de Engenharia de Sistemas e Computadores/IST
Rua Alves Redol, 9
1000 Lisboa, Portugal
Email: gcm@inesc.pt and lba@inesc.pt

ABSTRACT

The problem of separating a linear or nonlinear mixture of independent sources has been the focus of many studies in recent years. It is well known that the classical principal component analysis method, which is based on second order statistics, performs poorly even in the linear case, if the sources do not have Gaussian distributions. Based on this fact, several algorithms take in account higher than second order statistics in their approach to the problem. Other algorithms use the Kullback-Leibler divergence to find a transformation that can separate the independent signals. Nevertheless the great majority of these algorithms only take in account a finite number of statistics, usually up to the fourth order, or use some kind of smoothed approximations. In this paper we present a new class of objective functions for source separation. The objective functions use statistics of all orders simultaneously, and have the advantage of being continuous, differentiable functions that can be computed directly from the training data. A derivation of the class of functions for two dimensional data, some numerical examples illustrating its performance, and some implementation considerations are described.

1. Introduction

Real-life data often results from the influence of several independent sources. The influences of these sources can be combined in a linear way (e.g. the voices of several people, in the *cocktail party effect* [2]), or in a nonlinear way. The extraction of these sources form the data is an important problem, that has been receiving increasing attention in recent years. Depending on the settings it can appear under the names of *blind signal separation*, *independent component analysis* (ICA), *principal component analysis* (PCA), *nonlinear PCA*, or *factorial coding*.

The situations involving linear combinations of sources have been the object of most studies. A good overview if the basic methods can be found in Comon [4]. If the sources have Gaussian distributions, a simple principal component analysis, which performs a decorrelatiom, will yield independent signals. However, in most practical situations, the Gaussian assumption is not valid, and people have resorted to the cancellation of higher order moments, usually up to the fourth order [3,5,8,9,10,11,12].

In the case of nonlinear combinations of sources, the problem is ill posed: in general, there is an infinite number of sets of independent sources that, combined in an appropriate way, will yield the same data. Therefore, priors on the desirable solutions, and the consequent regularisation, will probably play an important role here.

Both for linear and nonlinear source separation, most authors have used either a small number of moments of the data distribution, or use the Kullback-Leibler (K-L) divergence between the joint density of the sources and the product of their marginal densities, taking into account that, for independent sources x_i,

$$p(x_1, x_2, \cdots, x_n) = \prod_{i=1}^{n} p(x_i) \tag{1}$$

Actual training sets are always finite, and thus have discrete distributions. For such distributions, the K-L divergence between the two members of (1) is a discontinuous, piecewise constant function of the x_i^k (the values of the sources for the different training patterns p^k). Another way to view this difficulty, is that the K-L divergence depends on the unknown continuous densities p, which must be estimated form the training data. This makes the K-L divergence somewhat inappropriate for the problem, and people have resorted to smoothed approximations, again typically involving truncation at the fourth order [1,4]. An indirect method that has also been used for nonlinear separation of discrete sources involves the use of the minimum description length principle [6,7,13].

It is apparent, from the foregoing discussion, that there is still a lack of good optimization criteria for achieving source independence. In this paper we propose a class of objective functions for this propose. Their main advantages are (1) that they are continuous, differentiable functions, and (2) that they can be computed directly from the training data.

This paper is organised as follows. Section 2 introduces the class of objective functions. Section 3 discusses some implementation issues. Section 4 gives examples of their use in linear and nonlinear signal separation. Section 5 concludes.

2. The Class of Objective Functions

We will develop the new class of objective functions for the case of two sources. Extension to more sources is straightforward. Let us consider two sources X and Y, which we view as jointly distributed random variables. If they are independent, we have

$$\left\langle X^m Y^n \right\rangle = \left\langle X^m \right\rangle \left\langle Y^n \right\rangle \tag{2}$$

for all m and n, where $\langle \cdot \rangle$ denotes expectation. As mentioned above, some authors have used versions of this property, with $m + n$ up to 4, for source separation. We wish to use this property for all orders simultaneously. We will therefore define our class of objective functions as

$$R_{XY} = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} a_{mn} \left(\left\langle X^m Y^n \right\rangle - \left\langle X^m \right\rangle \left\langle Y^n \right\rangle \right)^2 \tag{3}$$

where the a_{mn} are appropriately chosen, positive coefficients. Clearly, R_{XY} will be non-negative, and will be zero only if (2) is true for all orders m and n. We have

$$R_{XY} = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} a_{mn} \left[\left\langle X^m Y^n \right\rangle^2 - 2 \left\langle X^m Y^n \right\rangle \left\langle X^m \right\rangle \left\langle Y^n \right\rangle + \left\langle X^m \right\rangle^2 \left\langle Y^n \right\rangle^2 \right] \tag{4}$$

If we consider that we independently draw four patterns (which we denote by the subscripts 1 to 4), we can write

$$\left\langle X^m Y^n \right\rangle^2 = \left\langle X_1^m Y_1^n X_2^m Y_2^n \right\rangle \tag{5}$$

Making this and other similar substitutions in (4), we have

$$\begin{aligned} R_{XY} &= \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} a_{mn}\left(\left\langle X_1^m Y_1^n X_2^m Y_2^n\right\rangle - 2\left\langle X_1^m Y_1^n X_2^m Y_3^n\right\rangle + \left\langle X_1^m X_2^m Y_3^n Y_4^n\right\rangle\right) \\ &= \left\langle \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} a_{mn}\left(X_1^m Y_1^n X_2^m Y_2^n - 2X_1^m Y_1^n X_2^m Y_3^n + X_1^m X_2^m Y_3^n Y_4^n\right)\right\rangle \end{aligned} \tag{6}$$

With appropriate choices of the coefficients a_{mn} the infinite sums above can take a simple form. We make $a_{mn} = a_m a_n$. If we choose the a_m as the coefficients of the power series expansion of an entire function *f*,

$$f(x) = \sum_{m=0}^{\infty} a_m x^m \tag{7}$$

we have

$$R_{XY} = \left\langle f(X_1X_2)f(Y_1Y_2) - 2f(X_1X_2)f(Y_1Y_3) + f(X_1X_2)f(Y_3Y_4)\right\rangle \tag{8}$$

As an example, if we choose $a_m = 1/m!$ so that $f(x) = e^x$,

$$R_{XY} = \left\langle e^{X_1X_2+Y_1Y_2} - 2e^{X_1X_2+Y_1Y_3} + e^{X_1X_2+Y_3Y_4}\right\rangle \tag{9}$$

This is the objective function that we will use in the examples below.

3. Implementation Issues

The objective function in (9) is exponential in character, and is therefore strongly dependent on the scale of the variables *X* and *Y*. Clearly, if we scale *X* and *Y* down towards zero, the value of R_{XY} will tend towards zero, in the limit. Therefore, a minimization of R_{XY} will tend to decrease the range of variation of *X* and *Y*, if that is not countered by some means. For this reason, we have added to the objective function a term, tending to normalise the variance of the variables to 1:

$$\hat{R}_{XY} = R_{XY} + \lambda\left[\left(\sigma_X^2 - 1\right)^2 + \left(\sigma_Y^2 - 1\right)^2\right] \tag{10}$$

This is the function that was used in the examples of Section 4.

The scaling of the variables can be analysed in another way. Replacing *X* and *Y* by αX and by αY is equivalent to choosing $a_m = \alpha^{2m}/m!$. It is also equivalent to making

$$R_{XY} = \left\langle e^{\alpha^2(X_1X_2+Y_1Y_2)} - 2e^{\alpha^2(X_1X_2+Y_1Y_3)} + e^{\alpha^2(X_1X_2+Y_3Y_4)}\right\rangle \tag{11}$$

The use of this latter form of R_{XY} is therefore a simple way of having a global control on the relative weights of the various moments in the objective function. Another way to control these weights is through different choices of the function *f*. For example, $a_m = 1/(2m)!$ yields

$$f(x) = \begin{cases} \cosh\left(\sqrt{x}\right) & x \geq 0 \\ \cos\left(\sqrt{|x|}\right) & x < 0 \end{cases} \tag{12}$$

which, somewhat surprisingly, is a non-monotonic, non-convex function.

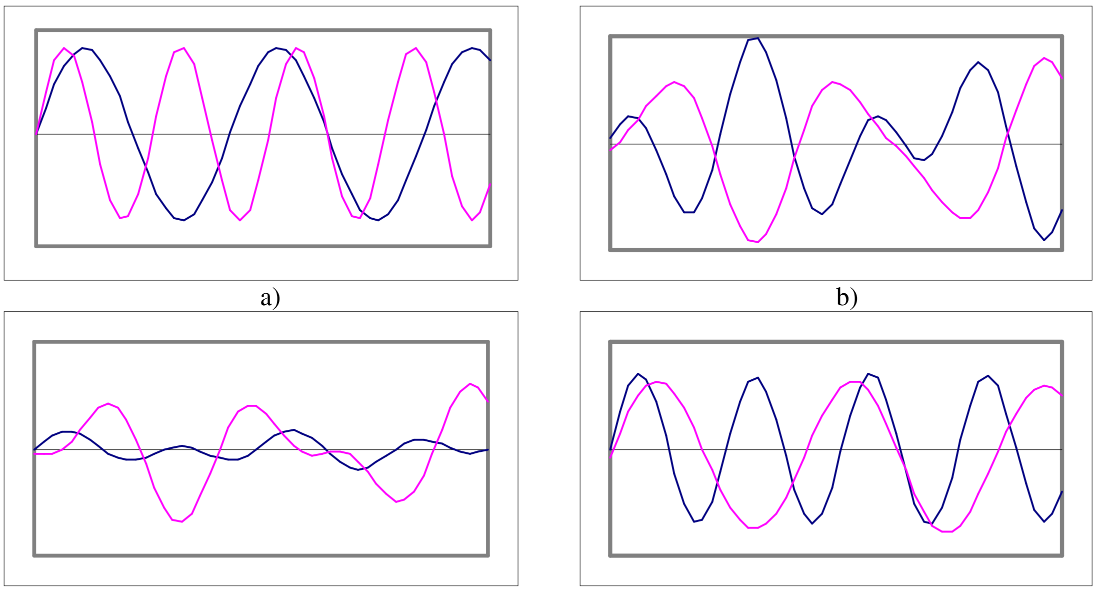
Another important issue has to do with the amount of computation required to evaluate R_{XY} Since its expression involves four independent patterns, a batch-mode training with a training set of size *N* would require the presentation of N^4 combinations of patterns, which becomes quickly impractical. For large training sets, a real-time (stochastic) training is therefore most appropriate.

4 Application Examples

The proposed objective function was tested in both linear and nonlinear situations. In the first test, 50 samples of two sinusoids, with frequencies $3\pi/2$ and $5\pi/2$ (Figure 1.a) were used as sources. Two mixed components were generated, using the matrix

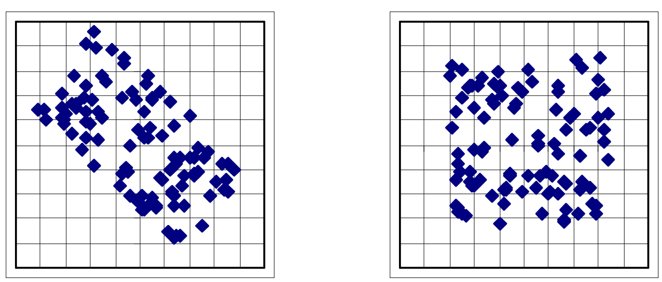
$$T = \begin{bmatrix} -4 & +5 \\ +3 & -1 \end{bmatrix}$$

(see Figure 1.b). The result of PCA is depicted in Figure 1.c which shows that PCA cannot separate the two components. The result of minimizing $\hat{R}_{XY}$ is shown in Figure 1.d, where the two components are well separated. The separation was achieved through a two unit linear network, trained by minimizing $\hat{R}_{XY}$ through gradient descent.



a) b) c) d)

Figure 1. a) Original signals b) mixed signals c) PCA separation d) output of the linear network

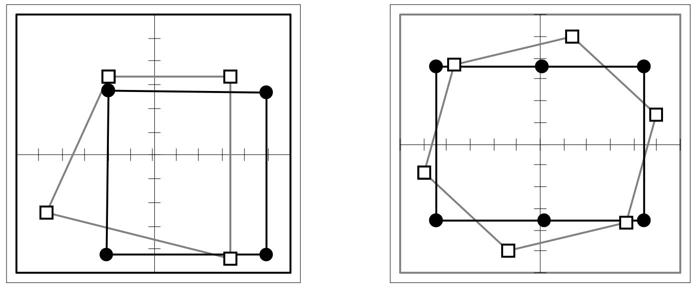


a) b)

Figure 2. a) Random set of 100 points used for testing the linear network. b) Output after adaptation

Another linear test is illustrated in Figure 2. A set of 100 random points, uniformly distributed inside a parallelogram, was used as training set (Figure 2.a). The output of a linear network, after minimizing $\hat{R}_{XY}$, is shown in Figure 2.b, where we see that the network arranged the data into a square with sides parallel to the axes.

Two small nonlinear tests were performed. In the first one (Figure 3.a), four training points were arranged in the position shown by the white squares. A nonlinear network with two hidden units (Figure 4) was trained by minimizing $\hat{R}_{XY}$ through gradient descent. The result, depicted by the black circles in Figure 3.a, shows that the network arranged the four patterns so that the horizontal and vertical coordinates are independent. A second test, with six points and a network with 3 hidden units, is shown in Figure 3.b, where once again the network placed the points into an independent arrangement.



a) b)

Figure 3.a) Nonlinear example with a 4 point set. b) with a 6 point set.

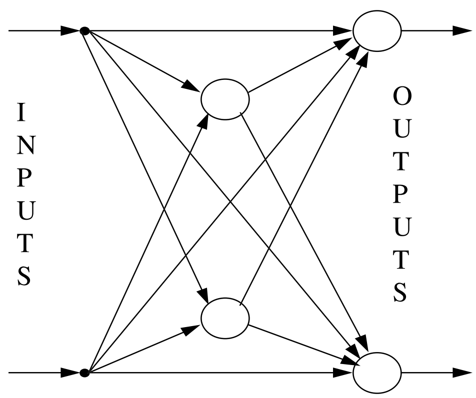


Figure 4. Non-linear network. The output units are linear

5. Conclusions

In this work we presented a new class of objective functions for blind signal separation. These functions use a weighted sum of statistics of all orders and are continuous and differentiable. With an appropriate choice of coefficients, the infinite sums in the objective function can be converted to a simple form. We tested a particular class of functions both in linear and nonlinear situations. In both cases we achieved good separation results. The algorithm was implemented through a neural network, trained by minimizing the objective function.

Acknowledgements

The first author holds a grant from PRAXIS XXI, and would like to acknowledge their support for this research.

References

[1] Amari, A., Cichoki, A., Yang, H. H. “A New Learning Algorithm for Blind Signal Seperation”. Proc. of the *1995 Conf. on Neural Information Processing Systems*, Denver CO, MIT Press, Nov-Dec 1995, in print.

[2] Bell, A. J., Sejnowski, T. J. “An Information-Maximization Approach to Blind Separation and Blind Deconvolution”. In *Neural Computation* Vol. 7, pp 1129-1159, 1995.

[3] Cardoso, J-F. “Source Separation Using Higher Order Moments”. In Int. Conf. on *Acoustic, Speech and Signal Processing*, Vol 4, pp 2109-2112, May 1989.

[4] Comon, P. “Independent Component Analysis - A New Concept?”. In *Signal Processing*, Vol. 36, pp 287-314, 1994.

[5] Deco, G., Brauer, W. “Nonlinear Higher-Order Statistical Decorrelation by Volume-Conserving Neural Architectures”. In *Neural Networks* Vol. 8, pp 525-535, 1995.

[6] Ghahramani, Z. “Factorial Learning and the *EM* Algorithm”. To appear in *Advances in Neural Information Processing Systems 7*, J. D. Cowan, G. Tesauro, and J. Alspector, eds., Morgan Kaufmann, San Mateo, CA, 1995.

[7] Hinton, G.E. , Zemel, R.S. “Autoencoders, Minimum Description Length and Helmholtz Free Energy”. In *Advances in Neural Information Processing Systems 6*, J. D. Cowan, G. Tesauro, and J. Alspector, eds., pp 3-10. Morgan Kaufmann, San Mateo, CA, 1994.

[8] Karhunen, J., Vigário, R., Wang, L., “Nonlinear PCA Type Approaches for Source Separation and Independent Component Analysis”. In *IEEE International Conference on Neural Networks*, Perth, Australia, 1995, in print.

[9] Oja, E. “PCA, ICA, and Nonlinear Hebbian Learning”. In *Int. Conf. on Artificial Neural Networks*, Paris, 1995, in print.

[10] Porat, B., Friedlander, B., “Direction Finding Algorithms Based on High-Order Statistics”. In *IEEE Transactions on Signal Processing*, Vol. 98, N 9, pp 2016-2023, September 1991.

[11] Swindlehurst, A. L., Kailath, T “Detection and Estimation Using the Third Moment Matrix”. In *Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 4, pp 2325-2328, May 1989.

[12] Wang, L., Karhunen, J., Oja, E., Vigário, R. “Blind Separation of Sources Using Nonlinear PCA Type Learning Algorithms”. In *Int. Conf. on Neural Networks and Signal Processing*, Nanjing, P.R. China, 1995, in print.

[13] Zemel, R. S. “A Mininum Description Length Framework for Unsupervised Learning”. *PhD. Thesis,* Department of Computer Science, University of Toronto, 1993.