

Document downloaded from:

<http://hdl.handle.net/10251/58589>

This paper must be cited as:

Safont Armero, G.; Salazar Afanador, A.; Vergara Domínguez, L. (2011). Nonlinear prediction based on independent component analysis mixture modelling. *Lecture Notes in Computer Science*. 6691(1):508-515. doi:10.1007/978-3-642-21498-1_64.



The final publication is available at

http://dx.doi.org/10.1007/978-3-642-21498-1_64

Copyright Springer Verlag (Germany)

Additional Information

Nonlinear Prediction based on Independent Component Analysis Mixture Modelling

Gonzalo Safont¹, Addisson Salazar¹, Luis Vergara¹

¹ Instituto de Telecomunicaciones y Aplicaciones Multimedia, Universidad Politécnica de Valencia, Camino de Vera
s/n, 46022, Valencia, Spain

gonsaar@upvnet.upv.es, asalazar@dcom.upv.es, lvergara@dcom.upv.es

Abstract. This paper presents a new algorithm for nonlinear prediction based on independent component analysis mixture modelling (ICAMM). The data are considered from several mutually-exclusive classes which are generated by different ICA models. This strategy allows linear local projections that can be adapted to partial segments of a data set while maintaining generalization (capability for nonlinear modelling) given the mixture of several ICAs. The resulting algorithm is a general purpose technique that could be applied to time series prediction, to recover missing data in images, etc. The performance of the proposed method is demonstrated by simulations in comparison with several classical linear and nonlinear methods.

Keywords: ICA, ICAMM, nonlinear prediction, Kriging, Wiener structure

1 Introduction

Independent component analysis (ICA) is an intensive area of research that is progressively finding more applications for both blind source separation (BSS) and for feature extraction/modelling. The goal of ICA is to perform a linear transformation of the observed sensor signals, such that the resulting transformed signals (the sources or prior generators of the observed data) are as statistically independent of each other as possible [1]. The linear ICA method is extended in independent component analysis mixture modelling (ICAMM) to a kind of nonlinear ICA model, i.e., multiple ICA models are learned and weighted in a probabilistic manner. Thus, the ICA mixture model is a conditional independence model, i.e., the independence assumption holds only within each class and there may be dependencies among the classes [2]. ICAMM has recently emerged as a flexible approach to model arbitrary data densities with non-gaussian distributions for the independent components (i.e., relaxing the restriction of modelling every component by a multivariate Gaussian probability density function).

ICA has been shown to improve the prediction of time series by considering certain realistic assumptions [3]. There is also a close relationship between ICA and the minimization of algorithmic complexity [4]. This has been used to improve the prediction of financial time series, see for instance [5]. A procedure that includes an ICA-based preprocessing step followed by prediction using neural networks has been applied for data prediction in several fields; see for instance [6]. Other applications benefit by using ICA for preprocessing since the independence among the variables is required [7].

This paper presents a novel procedure based on independent component analysis mixture modelling (ICAMM). The novelty of the procedure consists of the mixture of ICAs is not only proposed as a preprocessor to be applied before prediction, but the prediction itself is made by estimating the ICAMM parameters. The data are considered from several mutually-exclusive classes which are generated by different ICA models. This strategy allows linear local projections that can be adapted to partial segments of a data set while maintaining generalization (capability for nonlinear modelling) given the mixture of several ICAs.

The following sections of the paper include the following: the theoretical foundations of ICA and ICAMM; the proposed procedure for nonlinear prediction based on ICAMM; the results obtained for several simulations in comparison with two classical prediction methods (Kriging and a Wiener structure-based method); and the conclusions of the paper.

2 Independent Component Analysis Mixture Modelling

The standard noiseless instantaneous ICA formulates a $M \times 1$ random vector \mathbf{x} by linear mixtures of M random variables that are mutually independent (s_1, \dots, s_M) whose distributions are totally unknown. That is, for $\mathbf{s} = (s_1, \dots, s_M)^T$ and some matrix \mathbf{A} :

$$\mathbf{x} = \mathbf{A} \mathbf{s} . \quad (1)$$

The essential principle is to estimate the so-called mixing matrix \mathbf{A} , or equivalently $\mathbf{B} = \mathbf{A}^{-1}$ (the demixing matrix). The matrix \mathbf{A} contains the coefficients of the linear transformation that represents the transfer function from sources to observations. Thus, given N i.i.d. observations $(\mathbf{x}_1, \dots, \mathbf{x}_N)$ from the distribution of \mathbf{x} , \mathbf{A}^{-1} can be applied to separate each of the sources $s_i = \mathbf{B}_i \mathbf{x}$, where \mathbf{B}_i is the i th row of \mathbf{B} . This can be seen as a projection pursuit density estimation problem to find M directions such that the corresponding projections are the most mutually independent. For the sake of simplicity, we will assume the square problem (the same number of sources as mixtures, and thus the order of \mathbf{A} is $M \times M$).

ICAMM is proposed in the framework of pattern recognition, considering that the observed data come from a mixture model and they can be categorized into several mutually exclusive classes. ICAMM assumes the underlying process that generated observed data is composed by multiple ICA models (data of each class are modelled as an ICA, i.e., linear combinations of independent non-gaussian sources).

The general formulation of ICAMM is:

$$\mathbf{x}_t = \mathbf{A}_k \mathbf{s}_k + \mathbf{b}_k, \quad k = 1 \dots K . \quad (2)$$

where C_k denotes the class k , and each class is described by an ICA model with a mixing matrix \mathbf{A}_k and a bias vector \mathbf{b}_k . Essentially, \mathbf{b}_k determines the location of the cluster and

$\mathbf{A}_k \mathbf{s}_k$ its shape. The goal of an ICA mixture model algorithm is to determine the parameters for each class.

ICAMM was introduced in [2] considering a source model switching between Laplacian and bimodal densities. Recently, a generalization of the ICAMM framework called Mixca was proposed in [8], which includes non-parametric density estimation, semi-supervised learning, using any ICA algorithm for parameter updating, and correction of residual dependencies.

3 ICAMM-based Nonlinear Predictor

Let us consider data vector \mathbf{x} , of size $(N \times 1)$, which can be modelled through an ICAMM model with $k = 1 \dots K$ classes. The parameters $\mathbf{W}_k, \mathbf{s}_k, \mathbf{b}_k$ for this model have to be estimated from a training data set. Assuming that the last N_2 values of vector \mathbf{x} are unknown, we can group known and unknown values of \mathbf{x} into two smaller vectors, \mathbf{y} (known values) and \mathbf{z} (unknown values). Thus, we can write

$$\mathbf{x} = \begin{pmatrix} \mathbf{y} \\ \mathbf{z} \end{pmatrix}. \quad (3)$$

We would like to predict \mathbf{z} using the known values \mathbf{y} . Following the ICAMM model, the probability density function of data vector \mathbf{x} can be expressed as

$$p(\mathbf{x}) = p(\mathbf{y}, \mathbf{z}) = \sum_{k=1}^K p(\mathbf{y}, \mathbf{z} / C_k) p(C_k) = \sum_{k=1}^K |\det \mathbf{W}_k| p(\mathbf{s}_k) p(C_k), \quad (4)$$

where

$$\mathbf{s}_k = \mathbf{A}_k^{-1} \begin{pmatrix} \mathbf{y} \\ \mathbf{z} \end{pmatrix} - \mathbf{b}_k = \mathbf{W}_k \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} + \mathbf{W}_k \begin{pmatrix} \mathbf{0} \\ \mathbf{z} \end{pmatrix} - \mathbf{b}_k. \quad (5)$$

In order to maximize the joint probability density function, we propose a maximum a posteriori (MAP) estimator:

$$\mathbf{z}_{MAP} = \underset{\mathbf{z}}{\max} p(\mathbf{y}, \mathbf{z}). \quad (6)$$

Thus, we have to solve the following equations:

$$\frac{\delta p(\mathbf{y}, \mathbf{z})}{\delta \mathbf{z}} = \sum_{k=1}^K |\mathbf{W}_k| \frac{\delta p(\mathbf{s}_k)}{\delta \mathbf{z}} p(C_k) = 0 \quad (7)$$

$$\frac{\delta p(\mathbf{s}_k)}{\delta \mathbf{z}} = \sum_{n=1}^N \frac{\delta p(\mathbf{s}_k)}{\delta s_{kn}} \frac{\delta s_{kn}}{\delta \mathbf{z}}. \quad (8)$$

From (5), we can obtain the contribution of the sources corresponding to the variables \mathbf{z} :

$$\mathbf{W}_k \cdot \begin{pmatrix} \mathbf{0} \\ \mathbf{z} \end{pmatrix} = \begin{pmatrix} \mathbf{r}_{k1}^T \mathbf{z} \\ \vdots \\ \mathbf{r}_{kN}^T \mathbf{z} \end{pmatrix}, \quad (9)$$

where $\mathbf{W}_k = \begin{pmatrix} \cdots & \mathbf{r}_{k1}^T \\ \cdots & \vdots \\ \cdots & \mathbf{r}_{kN}^T \end{pmatrix}$. Thus, the sources can be expressed as

$$s_{kn} = \text{constant} + \mathbf{r}_{kn}^T \mathbf{z}. \quad (10)$$

The partial derivative of the sources with respect to the elements of vector \mathbf{z} is

$$\frac{\delta s_{kn}}{\delta \mathbf{z}} = \begin{bmatrix} \frac{\delta s_{kn}}{\delta z_1} \\ \vdots \\ \frac{\delta s_{kn}}{\delta z_{N_2}} \end{bmatrix} = \mathbf{r}_{kn}. \quad (11)$$

Assuming independence among the sources, and replacing (11) in (7), we obtain the objective function,

$$\frac{\delta p(\mathbf{y}, \mathbf{z})}{\delta \mathbf{z}} = \sum_{k=1}^K |\det \mathbf{W}_k| p(C_k) \sum_{n=1}^N c_{kn} \mathbf{r}_{kn} = \sum_{k=1}^K |\det \mathbf{W}_k| p(C_k) \mathbf{R}_k \mathbf{c}_k, \quad (12)$$

where $\mathbf{R}_k = [\mathbf{r}_{k1} \dots \mathbf{r}_{kN}]$ of dimension $N_2 \times 1$ and $\mathbf{c}_k = [c_{k1} \dots c_{kN}]^T$ of dimension $N \times 1$. We have called the proposed algorithm PREDICAMM (prediction based on ICAMM).

The optimization of the value of \mathbf{z} is done using a gradient method. Applying a steepest ascent optimization technique, we can write

$$\mathbf{z}_{(i+1)} = \mathbf{z}_{(i)} + \alpha_{(i)} \cdot \nabla f(\mathbf{z}_{(i)}), \quad (13)$$

where $\alpha_{(i)}$ is the stepsize and $\nabla f(\mathbf{z}_{(i)})$ is the gradient of the cost function. The stepsize α and the ascent direction $\nabla f(\mathbf{z})$ both set the convergence rate of the algorithm. There are several different combinations of stepsize and direction for gradient methods [9]. For simplicity, we used a steepest ascent method combined with a constant stepsize,

$$\mathbf{z}_{(i+1)} = \mathbf{z}_{(i)} + \alpha \cdot \left. \frac{\delta p(\mathbf{y}, \mathbf{z})}{\delta \mathbf{z}} \right|_{(i)}. \quad (14)$$

We selected a constant greater than 1 for α . This is due to the probability density values are lower in magnitude, and so its derivative has a much lower absolute value than \mathbf{z} .

4 Results

In order to test the proposed algorithm, a total of 13 different data sets were simulated, as shown in Table 1. The data sets consisted of ICA mixtures with up to three classes, which were obtained by using the Mixca algorithm [8] embedding the so-called JADE ICA algorithm [10] for parameter updating. This procedure has demonstrated flexibility for data modelling in several fields such as non-destructive testing [11][12] and biomedical problem diagnosis [13].

Table 1. Data sets used in simulations.

Dataset number	Known variables	Unknown variables	Number of classes	Density functions
01	1	1	1	Uniform, Laplacian
02	2	1	1	Uniform, Laplacian, K1
03	2	2	1	Uniform, Laplacian, K1, K10
04	2	2	1	Uniform
05	2	2	1	Laplacian
06	2	1	2	Uniform, Laplacian
07	2	2	2	Uniform, Laplacian
08	2	2	2	Laplacian, K1
09	2	2	2	K1, K10
10	2	1	3	Uniform, Laplacian, K1
11	2	2	3	Uniform, Laplacian, K1
12	2	1	3	Uniform, Laplacian, K1, K10
13	2	2	3	Uniform, Laplacian, K1, K10

From 2 to 4 variables were considered in simulations, defining 1 or 2 of them as unknowns for prediction purposes. The following source data densities were simulated: uniform (between ± 1), Laplacian distribution with a sharp peak at the bias and heavy tails, and two K distributions [14] with shape parameters $\nu = 1$ (named K1) and $\nu = 10$ (named K10). Examples of these densities are shown in Table 2. All densities had zero mean and unit variance. A total of 100 Montecarlo simulations were obtained for each of the data sets of Table 1.

Table 2. Probability density functions used in the simulations. All densities had zero mean and unit variance.


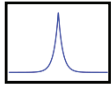
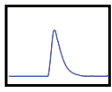
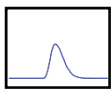
Density	Skewness	Kurtosis	Sample
Uniform	0	-1.2	
Laplacian	0	3	
K1	1.7	5.3	
K10	0.8	0.7	

Fig. 1 shows the results of a Montecarlo experiment to test the stability of the prediction algorithm. Both graphs in Fig. 1 show three different regions separated by different shadowing. The first region, named NC for Non-Convergence, is not shadowed. This region is composed by input values that do not converge at all and their end values are approximately equal to their starting values.

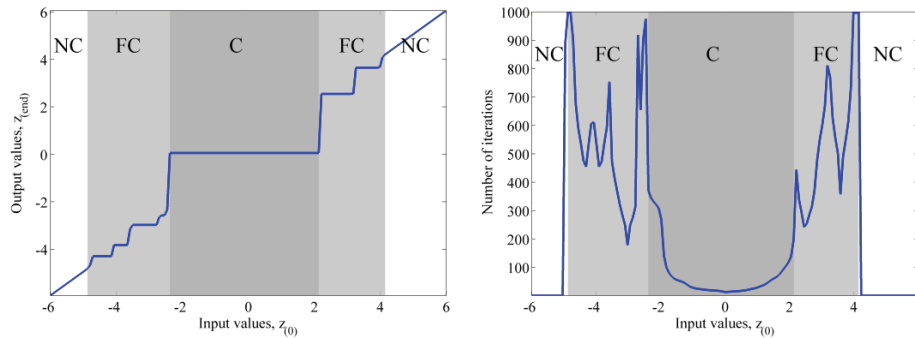


Fig. 1. Stability study for case number 1. Abbreviations (NC: nonconvergence, FC: failed convergence, convergence).

The second region, named FC for Failed Convergence, is shadowed in light gray. It is composed of input values that converge partially, but their value does not match the right prediction. The last region, named C for Convergence, is shadowed in darker gray. The values in this region converge to the right prediction and do so in the least amount of iterations. Note that the axis values in Fig. 1 (both for $z_{(end)}$ and for $z_{(0)}$) are normalized in such a way that 0 is the value of the right prediction. The size of the convergence zone (C) spans approximately from -2 to +2, which is about four times the standard deviation of the mixed signals. Thus, Fig. 1 shows that the algorithm was able to converge to the right value starting from relatively far values. We compared PREDICAMM with two classical predictors. The first predictor was ordinary kriging, a linear unbiased predictor. The second predictor was a Wiener structure composed by a linear step performed by ordinary kriging followed by a non-linear step performed by the conditional expectation of predicted to real data.

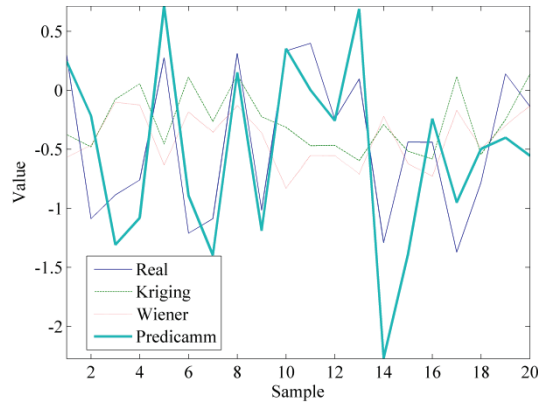


Fig. 2. Comparison of some predicted samples estimated by different methods.

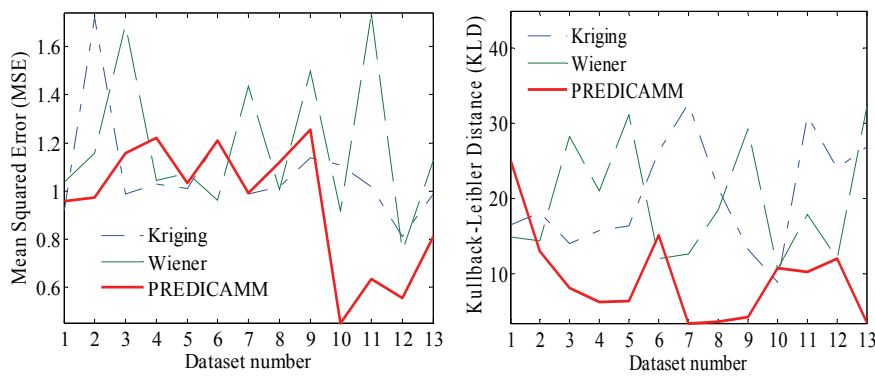


Fig. 3. Performance comparison of the proposed method with classical prediction methods.

Fig. 2 shows an example of 20 samples estimated by different prediction methods. The best results are obtained by the proposed method PREDICAMM. The curve estimated by the proposed method follows the changes in slope and magnitude of the real values. The values estimated by Kriging and the Wiener-based predictor are smoothed version of the real values.

Two figures of merit were defined in order to obtain a general evaluation of the method performance: (1) the mean squared error (MSE) and (2) the symmetric Kullback-Leibler divergence (KLD) between the probability density of the predicted data and the probability density of the real data. Fig. 3 shows the figures of merit estimated for the different simulated data sets. The PREDICAMM method obtained the lowest values for MSE and KLD demonstrating the best quality of the prediction. The difference of PREDICAMM with classical methods is higher for the datasets generated from models with multiple ICAs since the prediction is made using a method that is based on the generative model of the data.

5 Conclusion

A novel method for prediction has been presented. The method is based on ICAMM which allows complex data densities to be dealt with. The method is a general purpose technique that could be applied to several fields such as time series prediction, recovering of missing data in images, etc. The performance of the method has been demonstrated in several simulations

outperforming the results obtained by classical prediction methods such as kriging and Wiener structures.

Acknowledgments. This work has been supported by the *Generalitat Valenciana* under grant PROMETEO/2010/040, and the Spanish Administration and the FEDER Programme of the European Union under grant TEC 2008-02975/TEC.

References

1. Hyvärinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. John Wiley & Sons, New York (2001)
2. Lee, T.W., Lewicki, M.S., Sejnowski, T.J.: ICA mixture models for unsupervised classification of non-gaussian classes and automatic context switching in blind signal separation. *IEEE Trans. on Patt. Analysis and Mach. Intellig.* 22 (10), 1078-1089 (2000)
3. Malaroiu, S., Kiviluoto, K., Oja, E.: ICA Preprocessing for Time Series Prediction. In: 2nd International Workshop on ICA and BSS (ICA 2000), pp. 453-457 (2000)
4. Pajunen, P.: Extensions of Linear Independent Component Analysis: Neural and Information-Theoretic Methods. Ph.D. Thesis, Helsinki University of Technology (1998)
5. Gorriz, J.M., Puntonet, C.G., Salmeron, G., Lang, E.W.: Time Series Prediction using ICA Algorithms. In: Proceedings of the 2nd IEEE International Workshop on Intelligent Data Acquisit. and Advanc. Comput. Systems: Technology and Applications, pp. 226-230 (2003)
6. Wang, C.Z., Tan, X.F., Chen, Y.W., Han, X.H., Ito, M., Nishikawa, I.: Independent component analysis-based prediction of O-Linked glycosylation sites in protein using multi-layered neural networks. In: IEEE 10th Internat. Conf. on Signal Processing, pp.1-4 (2010)
7. Zhang, Y., Teng, Y., Zhang, Y.: Complex process quality prediction using modified kernel partial least squares. *Chemical Engineering Science* 65, 2153-2158 (2010)
8. Salazar A., Vergara L., Serrano A., Igual J.: A general procedure for learning mixtures of independent component analyzers. *Pattern Recognition* 43 (1), 69-85 (2010)
9. Bersektaş, D.: Nonlinear programming. Athena Scientific, Massachusetts (1999).
10. Cardoso, J.F., Souloumiac, A.: Blind beamforming for non gaussian signals. *IEE Proceedings-F* 140 (6), 362-370 (1993)
11. Salazar, A., Vergara, L., Llinares, R.: Learning material defect patterns by separating mixtures of independent component analyzers from NDT sonic signals. *Mechanical Systems and Signal processing* 24 (6), 1870-1886 (2010)
12. Salazar, A., Vergara, L.: ICA mixtures applied to ultrasonic nondestructive classification of archaeological ceramics. *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 125201, 11 pages, 2010. doi:10.1155/2010/125201
13. Salazar, A., Vergara, L., Miralles, R.: On including sequential dependence in ICA mixture models. *Signal Processing* 90 (7), 2314-2318 (2010)
14. Raghavan, R.S.: A Model for Spatially Correlated Radar Clutter. *IEEE Trans. on Aerospace and Electronic Systems* 27, 268-275 (1991)