Egyptian Computer Science Journal (ISSN-1110-2586)
Volume 41– Issue 2, May 2017

Ordering of Dominant Independent Components in Time Series Analysis Using Fast ICA Algorithm

Abear Kamel, Amr Goneid and Daniah Mokhtar
Department of Computer Science & Engineering, the American University in Cairo, Cairo, Egypt

Abstract

Among the various methods used for time series analysis, Independent Component Analysis (ICA) has proven to be an effective tool to reconstruct and represent time series generated by overlapping independent sources. In spite of its success in obtaining independent components, there remains the need to order such components according to their contribution in data reconstruction. In this paper we present a method for ordering the independent components according to their dominance. The present method develops on the work done in [13] to reconstruct a time series of stock market exchange rates by using a modified Fast ICA (FICA) algorithm instead of the formerly used Learned Parametric Mixture algorithm (LPM). Compared to LPM, the present experimental results show that our Fast ICA gives better reconstruction results when applied to the dataset of stock market exchange rates time series. The present study also adds another reconstruction error measure, the Canberra measure, for the reconstruction of observations from estimated independent components. Further improvement is also obtained by adding such error measure for the reconstruction process.

Keywords: *Independent component analysis; Independent component ordering; Data Reconstruction*

1. Introduction

In time series analysis, Principle Component Analysis (PCA) was considered to be a prevalent analysis tool whose advantages are generally two-fold, first the order of uncorrelated principle components is explicitly given in terms of their variances, and second, the underlying structure of series can be revealed in using the first few principle components. However, the PCA technique only uses second order statistics information, which makes the principle components de-correlated but not really independent.

On the other hand, Independent Component Analysis (ICA) has shown promising results in not only removing correlations among the data but also in generating *independent* features. It has become an increasingly important tool for analyzing large data sets in search for patterns and has been applied in a wide range of problems in which the observed signals may be considered as results of linearly mixed instantaneous source signals [1]. There is no prior knowledge about the linear generative model or the source signals except that they are statistically independent. For this reason, ICA is in most cases associated with the problem of Blind Source Separation (BSS).

ICA has been applied in a wide range of problems [1]. In particular, this method has demonstrated to be successful in various speech recognition problems [2], three dimensional (3-D) object recognition [3], natural images [4], unsupervised classification [5], bioinformatics [6], texture segmentation [7], electroencephalograms (EEG) [8], functional

Magnetic Resonance Imaging (fMRI) [9], face recognition [10], the prediction of stock market prices [11] , and texture classification [12]. In a large number of problems of such nature, the observed signals may be considered as results of linearly mixed instantaneous source signals. There is no prior knowledge about the linear generative model or the source signals except that they are statistically independent. In time series analysis, it has been realized that ICA , rather than PCA , has the advantage of involving higher order statistics, which makes the components reveal more useful information than PCA [13]. This advantage comes from the ability of ICA to reconstruct occluded information from important independent and non-Gaussian data, whether it's due to loss or noise, using the principles of Blind Source Separation.

Given the independent components (IC's) recovered by ICA, there is also a great value in the reconstruction of data within its same original order as it has a great effect on the degree of dominance of such IC's. Although this problem of ordering of the IC's had some attention in the literature, the methods suggested vary considerably. For example, the components are sorted according to their non-Gaussianity [14], or by selecting a subset of the components based on the mutual information between the observations and the individual components [15]. Also, in the work [16], the L_∞ norm of each individual component is used to decide on the component ordering where the order is determined based on each individual component only. More recently, component ordering is suggested to be based on component power [17]. On the other hand, the work of Cheung [13] approaches the effect of interactions of individual components on the observed series by considering their joint contributions in data reconstruction, which naturally leads the component ordering to a typical combinatorial optimization problem. In that work, the extraction of independent components is done using the Learned Parametric Mixture Based ICA algorithm (LPM) [18, 19]. For IC's ordering, it also uses the process of minimizing the reconstruction error based on the Relative Hamming Distance (RHD).

In the present paper, we propose using a modified Fast ICA (FICA) algorithm [20] instead of the Learned Parametric Mixture algorithm (LPM) for the extraction of independent components. Such FICA algorithm provides higher performance and utilizes a more precise convergence measure. For independent component ordering, we also compare the process of minimizing the reconstruction error based on the Relative Hamming Distance (RHD), the Mean Square Error (MSE) and Canberra Distance (CD) measure.

The paper is organized as follows: section 2 introduces the ICA model , and the modified Fast ICA algorithm used in the present work; section 3 describes Independent Component Ordering under Data reconstruction criteria ; section 4 presents the determination of Dominant Independent Components ; section 5 gives results of experimentation ; and finally section 6 is the conclusion of our work.

2. ICA of time series

2. 1 The ICA model

We consider the observed k time series $\boldsymbol{X} = x(t) = [x_1(t), \ldots x_k(t)]^T$, $1 \leq t \leq N$ to be the instantaneous linear mixture of unknown statistically independent components $\boldsymbol{Y} = y(t) = [y_1(t), \ldots . y_k(t)]^T$. In order to generate Independent Components (IC's) from the observed time series $\boldsymbol{X}$, we consider the ICA instantaneous linear noiseless mixing model (Figure (1)) represented by:

$$\boldsymbol{X} = \boldsymbol{A}\ \boldsymbol{Y}, \qquad (1)$$

where $\boldsymbol{Y}$ is a random matrix of hidden sources with mutually independent components, and $\boldsymbol{A}$ is an unknown $k \times k$ nonsingular mixing matrix.

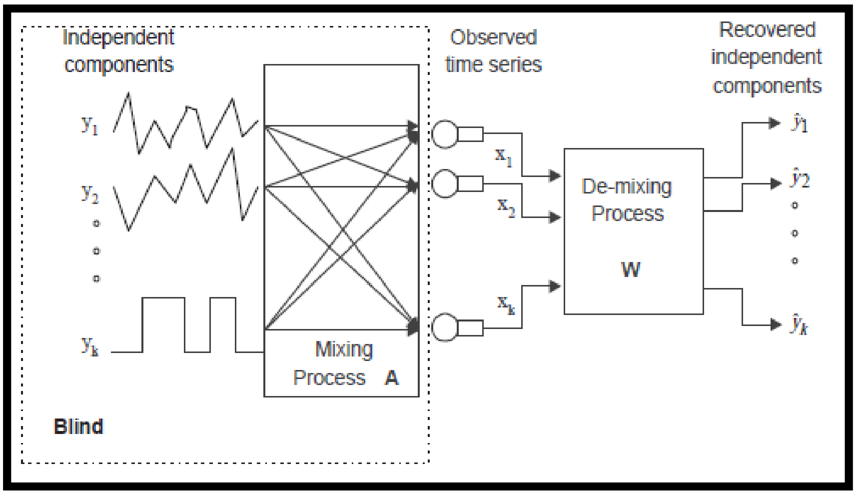

Figure 1. The ICA Model

Given $\boldsymbol{X}$, the basic problem is to find an estimate $\hat{\boldsymbol{y}}$ of $\boldsymbol{Y}$ and the mixing matrix $\boldsymbol{A}$ such that:

$$\hat{\boldsymbol{y}} = \boldsymbol{W}\,\boldsymbol{X} = \boldsymbol{W}\,\boldsymbol{A}\;\boldsymbol{Y} = \boldsymbol{G}\;\boldsymbol{Y} \approx \boldsymbol{Y} \qquad (2)$$

where $\boldsymbol{W} = \boldsymbol{A}^{-1}$ is the *unmixing* matrix, and $\boldsymbol{G} = \boldsymbol{W}\,\boldsymbol{A}$ is usually called the Global Transfer Function or Global Separating-Mixing (GSM) Matrix. The linear mapping $\boldsymbol{W}$ is such that the unmixed signals $\hat{\boldsymbol{y}}$ are statistically independent. However, the sources are recovered only up to scaling and permutation. In practice, the estimate of the unmixing matrix $\boldsymbol{W}$ is not exactly the inverse of the mixing matrix $\boldsymbol{A}$. Hence, the departure of $\boldsymbol{G}$ from the identity matrix $\boldsymbol{I}$ can be a measure of the error in achieving complete separation of sources.

The estimation of the unmixing matrix $\boldsymbol{W}$ cannot be done in closed form. Instead, solution methods are based on finding maxima or minima of some objective function. The most two famous methods seek an estimate of $\boldsymbol{W}$ either based on maximizing the negentropy (negative entropy) or by using Maximum Likelihood Estimation (MLE). Such approaches require that the solution advances iteratively in steps starting from some initial estimate until it converges to the final solution. Learning from the data is required in each of these steps leading to essentially neural unsupervised learning algorithms.

2. 2 The modified FICA neural learning algorithm

For computing the independent components (IC's) from the observed time series, we have adopted the modified algorithm given by [20] which is based on the Fast ICA algorithm originally given by [21]. Basically, the algorithm uses a fixed-point iteration method to maximize the negentropy using a Newton iteration method. We assume that the observation matrix $\boldsymbol{X}$ of k time series and N samples has been preprocessed by centering followed by whitening or sphering to remove correlations. Centering removes means via the transformation $\boldsymbol{X} \leftarrow \boldsymbol{X} - E\{\boldsymbol{X}\}$ and whitening is done using a linear transform (PCA like) $\boldsymbol{Z} = \boldsymbol{V}\boldsymbol{X}$ where $\boldsymbol{V}$ is a whitening matrix. A popular whitening matrix is $\boldsymbol{V} = \boldsymbol{D}^{-1/2}\,\boldsymbol{E}^{T}$, where $\boldsymbol{E}$ and $\boldsymbol{D}$ are the eigenvector and eigenvalue matrices of the covariance matrix of $\boldsymbol{X}$, respectively. The resulting new matrix $\boldsymbol{Z}$ is therefor characterized by $E\{\boldsymbol{Z}\boldsymbol{Z}^{T}\} = \boldsymbol{I}$ and $E\{\boldsymbol{Z}\} = 0$. After obtaining the unmixing matrix $\boldsymbol{W}$ from whitened data, the total unmixing matrix is then $\boldsymbol{W} \leftarrow \boldsymbol{W}\,\boldsymbol{V}$. The algorithm estimates several or all components in parallel using symmetric orthogonalization by setting $\boldsymbol{W} \leftarrow (\boldsymbol{W}\,\boldsymbol{W}^T)^{-1/2}\,\boldsymbol{W}$ in every iteration.

In this modified version of the algorithm, the performance during the iterative learning process is measured using the matrix $\boldsymbol{G} = \boldsymbol{W}\,\boldsymbol{A}$, which is supposed to converge to a permutation of the scaled identity matrix at complete separation of the IC's. This is done by decomposing $\boldsymbol{G} = \boldsymbol{Q}\,\boldsymbol{P}$, where $\boldsymbol{P}$ is a positive definite *stretching* matrix and $\boldsymbol{Q}$ is an orthogonal *rotational* matrix. The cosine of the rotation angle is to be found on the diagonal of $\boldsymbol{Q}$ so that a convergence criterion is taken as $\Delta\ |diag(\boldsymbol{Q})|_{min} < \varepsilon$, where ε is a threshold value. Also, In this algorithm, we use the performance (error) measure, *E3* introduced in [20]:

$$\boldsymbol{E}3 = \frac{1}{2\boldsymbol{k}(\boldsymbol{k}-1)}\sum_{i=1}^{k}\{\sum_{j=1}^{k}|\boldsymbol{g}_{ij}| - \boldsymbol{M}_i + |\boldsymbol{M}_i - 1|\} + \sum_{j=1}^{k}\{\sum_{i=1}^{k}|\boldsymbol{g}_{ij}| - \boldsymbol{M}_j + |\boldsymbol{M}_j - 1|\} \qquad (3)$$

where g_{ij} is the ijth element of the matrix $\boldsymbol{G}$ of dimensions $k \times k$, $M_i = max_k \mid g_{ik} \mid$ is the absolute value of the maximum element in row (i) and $M_j = max_k \mid g_{kj} \mid$ is the corresponding quantity for column (j). It is shown in [20] that the index *E3* is more precise than the commonly used *E1* and *E2* indices [e.g., 22] and is independent of the matrix dimensions. It is also normalized to the interval {0,1}, the greater the value of *E3*, the worse is the performance.

The algorithm is summarized in the following steps:

- Preprocess observation matrix $\boldsymbol{X}$ to get $\boldsymbol{Z}$
- Choose random initial orthonormal vectors $\boldsymbol{w_i}$ to form intial $\boldsymbol{W}$ and random $\boldsymbol{A}$
- Set $\boldsymbol{W_{old}} \leftarrow \boldsymbol{W}$
- Iterate:
    1. Do Symmetric orthogonalization of $\boldsymbol{W}$ by setting $\boldsymbol{W} \leftarrow (\boldsymbol{W}\,\boldsymbol{W}^T)^{-1/2}\,\boldsymbol{W}$
    2. Compute dewhitened matrix $\boldsymbol{A}$ and new $\boldsymbol{G} = \boldsymbol{W}\,\boldsymbol{A}$ and do polar decomposition of $\boldsymbol{G} = \boldsymbol{Q}\,\boldsymbol{P}$
    3. Compute error *E3*
    4. If not the first iteration, test for convergence: $\Delta \mid diag(\boldsymbol{Q}) \mid_{min} < \varepsilon$
    5. If converged, break.
    6. Set $\boldsymbol{W_{old}} \leftarrow \boldsymbol{W}$
    7. For each component $\boldsymbol{w_i}$ of $\boldsymbol{W}$, update using learning rule
       $\boldsymbol{w_i} \leftarrow E\{z\,f\,(w_i\,z)\} - E\{f\,'\,(w_i\,z)\}\,\boldsymbol{w_i}$
- After convergence, dewhiten using $\boldsymbol{W} \leftarrow \boldsymbol{W}\,\boldsymbol{V}$
- Compute independent components $\hat{\boldsymbol{y}} = \boldsymbol{W}\,\boldsymbol{X}$

In step 7 in the iteration loop, z is a column vector representing one sample from the whitened matrix $\boldsymbol{Z}$, $f\,(y)$ is a non-linearity function, $f\,'(y)$ is its derivative and the expectation is taken as the average over the N samples in $\boldsymbol{Z}$. The non-linearity $f\,(y)$ is essential in the optimization process and for the learning rule that updates the estimates of the unmixing matrix $\boldsymbol{W}$ and, overall, it is important for the stability and robustness of the convergence process. A general purpose non-linearity is $f(y) = tanh(ay)$.

3. Independent component ordering under data reconstruction criteria

3. 1 Contribution of independent components to observed time series

In ordinary ICA the components are assumed to be completely independent and they don't necessarily have any meaningful order relationship, but in practice, the estimated "Independent" components are often not all independent. Under such conditions, we might consider the process of reconstructing time series x_i from an estimated independent component $\hat{y}_j$, $1 \le j \le k$. Following [13], the contribution may be expressed by the 3-D space:

$$u(i, j, t) = W^{-1}(i, j)\hat{y}_j(t) , \quad 1 \le j \le k \tag{4}$$

where $W^{-1}(i, j)$ is the *(i,j)th* element in the inverse of the *W* matrix.

We may assume the presence of a list L_i of independent components indices expressing a specific component ordering. In this case, the reconstruction of x_i by the first m independent components in the list L_i is the sum of the contributions of each individual component given by equation (4), i.e.,

$$\hat{x}_{L_i}{}^m = \hat{x}_i{}^m(L_i, t) = \sum_{s=1}^{m} u(i, s, t) \tag{5}$$

where s denotes the s^th^ element of $L_{i..}$

3. 2 Reconstruction error and optimal order list

The reconstruction error for x_i can be computed under a certain error measure (for example, MSE, RHD or CD). At a given time in the time series x_i , this may be denoted by the quantity $q(x_i(t), \hat{x}_i{}^m(L_i, t))$. For the whole series, the average is given by:

$$Q(x_i, \hat{x}_i{}^m(L_i)) = \underset{1 \le t \le N}{aver}\left\{q(x_i(t), \hat{x}_i{}^m(L_i, t))\right\} \tag{6}$$

Hence, the cumulative error for the *Q*- measure over all possible values of m (1 ≤ m ≤ k) is given by:

$$\varepsilon(L_i) = \sum_{m=1}^{k} Q(x_i, \hat{x}_i{}^m(L_i)) \tag{7}$$

Under such measure, an optimal ordering list may be obtained as:

$$L_i^{opt} = \arg\min_{L_i}(\varepsilon(L_i)) \tag{8}$$

3. 3 Present approach for optimal order lists

We are using an iterative approach to calculate the cumulative summation on a defined number of independent components denoted by m. The reason we are using m independent components is to calculate the difference between cumulative errors to identify the optimum order of components according to their joint contributions in data reconstruction instead of ordering depending on single independent components. The steps for this approach are as follows:

1. We use the infinity norm to obtain an initial ordering list (L_∞).
2. The initial ordering list is used in the reconstruction of the summation of contributions to time series x_i using the quantity $\hat{x}_{L_i}^m$ given by equation (5). Notice that such quantity is computed as a matrix with k rows and N columns, and the mth row represents the sum of contributions of the first m components in the list. For example, for $L_\infty = [5\ 1\ 3\ 2\ 4\ 6]$, the first row of $\hat{x}_{L_i}^m$ is the contribution of component 5 to x_i , while the second row represents the sum of contributions of components 5 and 1, and so on till the kth row which represents the cumulative contribution of all available independent components.
   This step results in $\hat{x}_{L_i}^m$ for the infinity norm ordering list.
3. Given a certain error measure (RHD, MSE or CD), the above value of $\hat{x}_{L_i}^m$ is used to calculate the first iteration of reconstruction error using equations (6, 7).
4. Ordering the error rate for each of the difference measures in ascending order, we produce a new ordering list $L*$
5. We then repeat the previous steps 2 through 4 to calculate the new reconstruction upon from which an optimal ordering list L^{opt} is obtained.

4. Determination of dominant independent components

Currently, there is no systematic method to determine a sub-list of the dominant components (except if it is done manually). However, it was suggested in [13] to use a number of selection criteria to determine the set of m* dominant components from the entire ordered independent components time series. One method is to follow a successive exclusion process using a cost function:

$$C(m) = Q(x_i, \hat{x}_i^m(L_i)) - Q(x_i, \hat{x}_i^{m-1}(L_i)) \tag{9}$$

That would test the mixture variation of the independent components to find m*, which represents the appropriate number of the first ordered indices to be the dominant components. If a dominant component is removed from the calculation, the data reconstruction error will have an obvious high effect on the error rate.

However, if a non-dominant component is removed, it will have a minor input to that representation. The resulting set would still highly dominate the trend of the entire time series as non-dominant components slightly affect the data.

5. Experimentation Data and Results

5. 1 Time series dataset

We have chosen foreign exchange rates for experimenting with the present methodology of independent components ordering. Time series of 6 foreign exchange rate series were selected representing USD versus Australian Dollar, French Franc, Swiss Franc, German Mark, British Pound and Japanese Yen in the period from November 1991 till August 1995. The dataset size was 6 time series over 1,354 days collected from different historical exchange rates data sources such as [23, 24, 25].

5. 2 Error measures

For evaluating reconstruction errors, we have selected 3 error measures as follows:

A. Relative Hamming Distance (RHD):

The Q-measure using RHD is given as:

$$Q\left(x_i, \hat{x}_{L_i}^m\right) = RHD\left(x_i, \hat{x}_{L_i}^m\right) = \frac{1}{N-1}\sum_{t=1}^{N-1}[R_i(t) - \hat{R}_{L_i}^m(t)]^2 \quad (10)$$

$$R_i(t) = sign[x_i(t+1) - x_i(t)]$$

$$\hat{R}_{L_i}^m(t) = sign[\hat{x}_{L_i}^m(t+1) - \hat{x}_{L_i}^m(t)]$$

$$sign(r) = \begin{cases} 1 & if\ r > 0, \\ 0 & if\ r = 0, \\ -1 & otherwise \end{cases}$$

B. Mean Square Error (MSE):

The Q-measure using MSE is given as:

$$Q\left(x_i, \hat{x}_{L_i}^m\right) = MSE\left(x_i, \hat{x}_{L_i}^m\right) = \frac{1}{N}\sum_{t=1}^{N}[\hat{x}_{L_i}^m(t) - x_i(t)]^2 \quad (11)$$

C. Canberra Distance (CD):

The Canberra distance is a weighted version of the Manhattan distance. The Q-measure using CD is given as:

$$Q(x_i, \hat{x}_{L_i}^m) = CD\left(x_i, \hat{x}_{L_i}^m\right) = \sum_{i=1}^{n}\frac{|xi - \hat{x}_{L_i}^m|}{|xi| + |\hat{x}_{L_i}^m|} \quad (12)$$

5. 3 Experimental results

To simulate the component ordering and reconstruction processes, the dataset of 6 time series ***Y*** and a random mixing matrix ***A*** were used to obtain the simulated mixed time series ***X***. The present FICA algorithm was then used to obtain the demixing matrix ***W*** and the $\hat{\boldsymbol{y}}$ estimated independent components**.** These were then used to construct the 3-D space *u(i,j,t)* from which reconstruction can be made. To serve as an example, we choose to compare between the USD-AUD series and the reconstructed signals.

The present results for ordering lists obtained for L_∞ norm, after 1st iteration and after 2nd iteration for the different Q-measures are as follows:

L_∞ norm: (6, 3, 5, 2, 4, 1)
RHD1: (1, 3, 2, 4, 5, 6)   MSE1: (2, 1, 3, 4, 6, 5)   CD1: (1, 2, 3, 6, 4, 5)
RHD2: (1, 3, 2, 4, 5, 6)   MSE2: (1. 2, 3, 4, 6, 5)   CD2: (1, 2, 3, 6, 4, 5)

For comparison with the work [13], Figure (2) shows the reconstructed series using the estimated y_5 component in that work using LPM algorithm and RHD-measure together with the USD-AUD series. We have computed reconstruction results with one independent component (y_5) using the present FICA algorithm and the 3 *Q*-measures of RHD, MSE and CD. For brevity, we only give here the results for the CD- measure as shown in Figure (3).

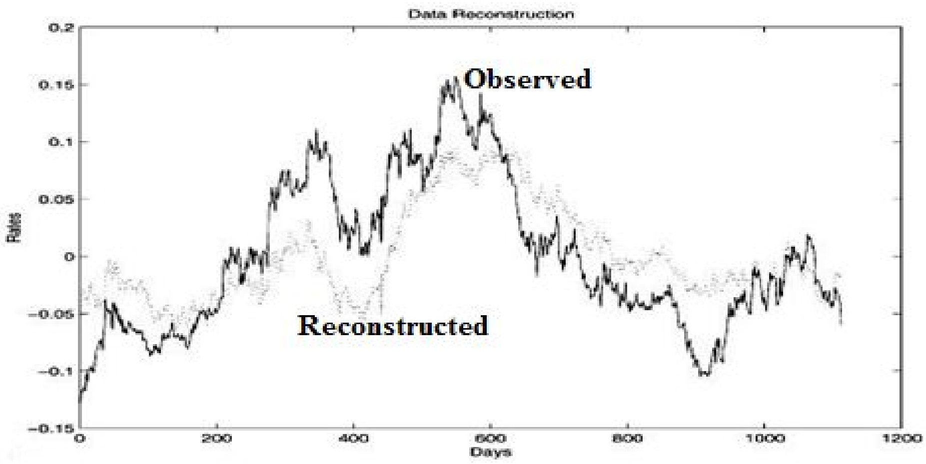

Figure 2. The observed series (solid lines) and reconstructed series from [13] with one independent component y_5 using LPM and RHD (dashed curve)

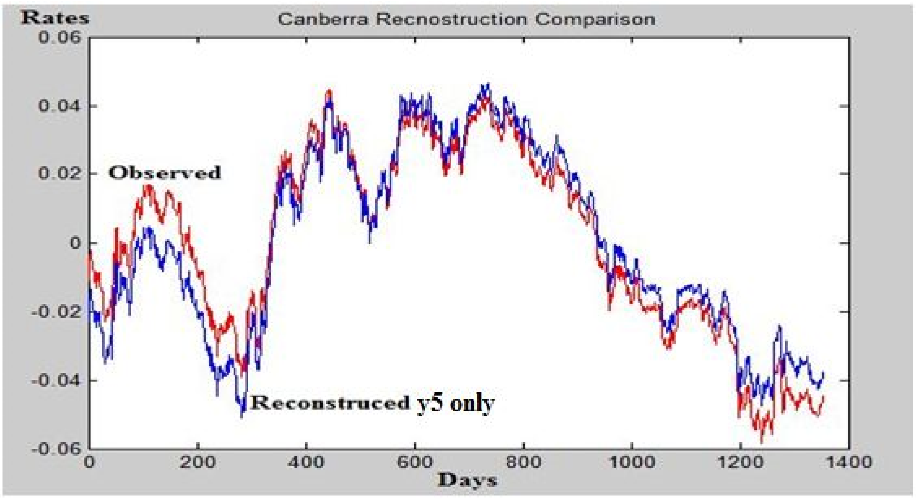

Figure 3. The observed USD-AUD series (Red lines) and present reconstructed series with one independent component y_5 using FICA algorithm and CD (Blue lines).

It can be seen from the above figures that the reconstruction using y_5 gives similar trends to observed series. However, the present use of the FICA algorithm and the CD measure has better similarity with observations. The average normalized difference between observed series and the present reconstructed series with one component y_5 is equal to 0.0226 for CD. Very similar results are obtained with the RHD, MSE measures.

Figure (4) shows the present results for the reconstruction using 3 independent components y_5, y_3 and y_1 as obtained by the FICA algorithm and the and CD-measure. The figure shows that such reconstruction gives better agreement with the observed series in comparison with results for only one independent component. Very similar results are obtained with the RHD, MSE measures (The average normalized difference between observed series and the present reconstructed series with 3 components is 0.0235, 0.0234 and 0.0235 for the RHD, MSE and CD measures, respectively).

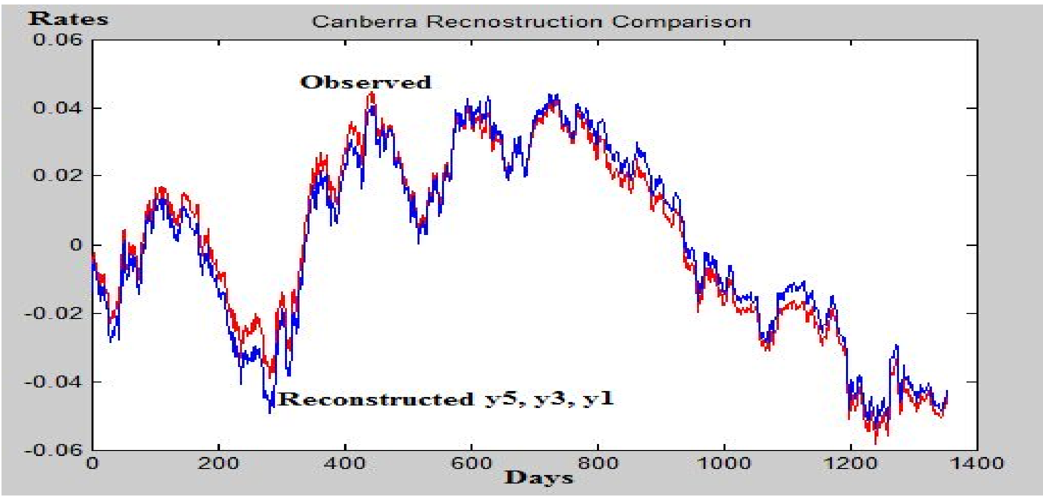

Figure 4. The observed USD-AUD series (Red lines) and present reconstructed series with 3 independent component y_5, y_3 and y_1 using FICA algorithm and CD (Blue lines).

6. Conclusion

The present study involves the implementation of an empirical ordering of independent components under reconstruction criteria by using a modified Fast ICA algorithm instead of the formerly used Learned Parametric Mixture algorithm (LPM). The present study also adds another reconstruction error measure, the Canberra measure, for the reconstruction of observations from estimated independent components. Present experimental results show that the Fast ICA gives better reconstruction results when applied to the dataset of stock market exchange rates time series. Further improvement is also obtained by adding the Canberra error measure for the reconstruction process.

References

[1]. Hyvarinen, A., Karhunen, J and Oja, E., "Independent Component Analysis", John Wiley and sons , New York , NT, 2001

[2]. Kwon, O.W and Lee, T.W., "Phoneme recognition using ICA-based feature extraction and transformation", *Signal Processing*, 84 (6), 1005 -1019, 2004

[3]. Sahambi, H.S. and Khorasani, K., "A Neural Network appearance-based 3-D object recognition using independent component analysis" , *IEEE Trans. on Neural Networks*, 14 (1), 138-148, 2003

[4]. Benlin, X., Fangfang, L., Xingliang, M. and Huazhong, J., “Study on Independent Component Analysis application in classification and change detection of multispectral images” *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* Vol.XXXVII , Part B7, Beijing 2008

[5]. Lee, T.W., Lewicki, M.S. and Sejnowski, T.J., "ICA mixture models for unsupervised classification of non-Gaussian classes and automatic context switching in blind signal separation", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22 (10) , 1078 –1089, 2000

[6]. Lee, S.I. and Batzoglou, S., "Application of independent component analysis to microarrays", *Genome Biol.*, 4 (11), R76, 2003

[7]. Chen, Y., and Runsheng, W., “Texture Segmentation Using Independent Component Analysis of Gabor Features”, *Proceedings of 18th ICPR,* Vol. 2, 147-150, 2006

[8]. Makeig, S., Bell, A.J., Jung, T.P. and Seinowski, T.J., " Independent component analysis of electroencephalographic data", *Advances in Neural Information Processing Systems,* Cambridge, MA : MIT Press, 8, 145 – 151, 1996

[9]. Mckeown, M.J., Makeig, S., Brown, G.G., Jung, T.P., Kindermann, S.S., Bell, A.J. and Sejnowski, T.J., " Analysis of fMRI by decomposition into independent spatial components", *Human Brain Mapping* , 6 (3), 160 – 188, 1998

[10]. Kwak, K.C., "Face Recognition Using an Enhanced Independent Component Analysis Approach", *IEEE Trans. on Neural Networks*, 18 (2), 530-541, 2007

[11]. Lai, Z. B., Y. M. Cheung, and Lei Xu. "Independent Component Ordering in ICA Analysis of Financial Data.", *Computational Finance,* 201-212, 1999

[12]. Goneid, A., and Ezzo, A., “Texture Classification using Gabor Filters and Independent Component Analysis”, *Egyptian Computer Science Journal*, 34(3), 1-16, 2010.

[13]. Cheung, Yiu-ming, and Lei Xu. "Independent component ordering in ICA time series analysis", *Neurocomputing* 41.1, 145-152, 2001

[14]. Hyvarinen, A., “Survey on independent component analysis”, *Neural Computing Surveys 2*, 94-128, 1999

[15]. Back, A.D., and Trappenberg, T.P., “Input variable selection using independent component analysis”**,** *Proceedings of International Joint Conference on Neural Networks*, Vol. 2, 989-992, 1999

[16]. Back, A.D., and Weigend, A.S., "A first application of independent component analysis to extracting structure from stock returns”, Int. J. Neural Systems, 8(4), 473-484, 1997

[17]. Hendrikse, A. J., Veldhuis, R. N. J., and Spreeuwers, L.J. "Component ordering in independent component analysis based on data power", *28th Symposium on Information Theory in the Benelux*, 211-218, 2007

[18]. Xu, L., Cheung, C.C., and Amari, S.I., “Learned parametric mixture based ICA algorithm”, *Neurocomputing*, 22, 69-80,1998

[19]. Xu, L., Cheung, C.C., Yang, H.H., and Amari, S.I., ”Information-theoretic approach with mixture of density”, *Proceedings of 1997 IEEE International Conference on Neural Networks (IJCNN'97)*, Vol. 3, 1821-1826, 1997

[20]. Goneid, A., Kamel, A., and Farag, I., “New convergence and performance measures for Blind Source Separation algorithms”, *Egyptian Computer Science Journal*, 31 (2), 13 - 24, 2009.

[21]. Hyvarinen, A., “Fast and robust fixed-point algorithms for independent component analysis”, *IEEE Trans. on Neural Networks*, 10(3), 626-634, 1999

[22]. Giannakopoulos, X., Karhunen, J., and Oja, E., “Experimental comparison of neural algorithms for independent component analysis and blind separation”, *Int. J. of Neural Systems*, 9 (2), 651 - 656, 1999

[23]. www.bankofengland.co.uk/boeapps/iadb/

[24]. www.oanda.com/solutions-for-business/historical-rates

[25]. www.xe.com/currencytables/