Universidad Carlos III de Madrid

TESIS DOCTORAL

Título de la tesis:
Independent Component Analysis for Time Series

Autor: Ester González Prieto

Directores: Antonio García Ferrer y Daniel Peña

Doctorado en Economía de la Empresa y Métodos Cuantitativos
Departamento de Estadística

Universidad Carlos III de Madrid
Getafe, Mayo 2011

A mis padres, a Rocío, y a Miguel

Agradecimientos

Sois muchas las personas que habéis estado conmigo durante esta etapa, gracias a todos por vuestro apoyo.

En primer lugar, quiero dar las gracias a mis directores, Antonio y Daniel. Gracias por iniciarme en este camino, por todo lo que me habéis enseñado en estos años, por vuestro tiempo y dedicación. Gracias por vuestros consejos, por vuestra paciencia, por esas inyecciones de optimismo cuando más lo he necesitado. Ha sido un placer trabajar a vuestro lado.

Quiero agradecer al Ministerio de Ciencia e Innovación, al Ministerio de Educación y a la Comunidad de Madrid por la financiación recibida a través de los proyectos CCG06-UC3M/HUM-0866, SEJ2006-04957, S2007-HUM-0413, SEJ2007-64500, y ECO2009-10287. En particular, quiero agradecer muy especialmente al proyecto BEC2002-00081 por la beca BES-2003-0836 con la que comencé este camino. También quiero agradecer al Departamento de Estadística de la Universidad Carlos III de Madrid por facilitarme los recursos necesarios durante el doctorado, y al Instituto Max Planck por su aportación en estos últimos meses.

Me gustaría mostrar mi gratitud al Prof. Erkki Oja, con el que tuve la suerte de entender los entresijos del algoritmo FastICA durante el verano de 2006 en Helsinki. Gracias también al Prof. David Brillinger por iniciarme en el mundo de los cumulantes y por hacerme sentir una más de sus estudiantes durante mi estancia en la Universidad de Berkeley en 2009.

No puedo olvidarme de mis compañeros, tanto de los estuvieron en mi etapa en la Autónoma, como los que me han acompañado en la Carlos III. A muchos de ellos tengo el honor de considerarles mis amigos. Alba, Maye, Santi, Ale, Ana Laura, Andrea, Pepa, Júlia, Adolfo, Bets, Jose, gracias por estar siempre ahí aunque yo a veces haya estado ausente.

Quiero dar las gracias a mi familia y amigos. Gracias a mis padres por su amor incondicional, por darme todo sin pedirme nada a cambio. Gracias por vuestro apoyo y generosidad, por estar siempre a mi lado y entender mis decisiones aunque a veces no os hayan gustado demasiado. Sin vosotros, no sería lo que soy. Gracias a mi hermana y a Fernando. Rocío, a ti especialmente te doy las gracias por ser mi cómplice, mi confidente, por animarme cuando lo he necesitado. Seguro que lo harás igual de bien en tu nuevo papel de madre. Iván, aunque aún no te hayamos visto la carita, no puedo olvidarme de ti, de toda la alegría que has traído a nuestras vidas sin ni siquiera haber nacido. En cierta manera, tu vida está ligada a la etapa final de esta tesis, vuestros 'nacimientos' siguen caminos paralelos. Ana, eres como una segunda madre para mí, gracias por todo lo bueno que me pasaste en la 'pila'. Marce, Mari Paz, tampoco me quiero olvidar de vosotros, gracias por todo vuestro cariño y por hacerme sentir como una hija más. Laura, Ignacio, Rosana, Elena, Noelia, Anita, Elisa, Óscar, Miguel, Lorena, gracias por vuestra amistad.

Miguel, para ti todas las palabras de agradecimiento son pocas. Siempre has estado a mi lado, pendiente de mí a pesar de la distancia, alentándome en los malos momentos y disfrutando conmigo de los momentos más dulces. Gracias por tu paciencia, por tu apoyo y por tu cariño, por hacerme reir y por darme serenidad. Gracias por ser el mejor compañero de viaje que pueda tener.

Contents

Resumen

El objetivo de esta tesis es aplicar el análisis de componentes independientes (ICA) sobre datos multivariantes de series temporales. También, se propone un nuevo procedimiento para predecir un vector de series temporales a partir de un número reducido de componentes independientes.

En el capítulo 1 analizamos la relación entre ICA y métodos clásicos de análisis multivariante: ICA es una extensión del análisis de componentes principales que calcula los componentes independientes (ICs) como la rotación que maximiza la independencia de los componentes principales; ICA puede definirse como un modelo de análisis factorial no-Gaussiano (Hyvärinen and Kano (2003)); ICA es un caso particular de método de búsqueda de proyección cuando la independencia de los componentes se mide en términos de su no-Gaussianidad; ICA, al igual que el algoritmo de Peña and Prieto (2001), detecta valores atípicos al proyectar los datos en las direcciones de máxima kurtosis. Además, en este capítulo, tratamos el problema de la reducción de la dimensionalidad en series temporales, describiendo brevemente algunos modelos multivariantes como el análisis canónico y el modelo factorial dinámico. Finalmente, comentamos los trabajos que se han propuesto en la literatura para aplicar ICA sobre datos con estructura temporal.

En el capítulo 2 proponemos un nuevo modelo de factores con heterocedasticidad condicionada, el modelo GICA-GARCH. Este modelo asume que las observaciones están generadas por una combinación lineal de factores no observados, que son independientes y condicionalmente heterocedásticos. El modelo GICA-GARCH supone que existe un número reducido de factores que explican los movimientos comunes de los datos observados y que tienen heterocedasticidad condicionada. Además, asume que la matriz de covarianzas condicionada de las observaciones es diagonal, y propone aproximarla mediante la combinación lineal de las varianzas condicionadas de los factores comunes. La ventaja del modelo GICA-GARCH con respecto a otros modelos de factores GARCH reside en el uso de ICA para la estimación de los

componentes comunes. Primero, tal y como muestran los resultados de las simulaciones, ICA reproduce bastante bien el exceso de kurtosis y obtiene 'buenos' estimadores de los componentes no-Gaussianos. Segundo, como los ICs son estadísticamente independientes, se pueden modelar por separado, ajustando distintos modelos ARMA-GARCH a cada uno de ellos, y así se simplifica el problema de estimar un modelo GARCH multivariante, reduciéndolo a la estimación de unos pocos modelos ARMA-GARCH univariantes. Por último, tal y como se muestra en la aplicación empírica, las predicciones un paso adelante de los rendimientos del IBEX 35 dadas por el modelo GICA-GARCH mejoran las dadas por los modelos O-GARCH (Alexander (2001)) y CUC-GARCH (Fan et al. (2008)).

En el capítulo 2 también presentamos una comparativa entre el modelo GICA-GARCH y otros modelos de factores GARCH, distinguiendo entre los que suponen estructura de factores en la distribución no condicionada de los datos, como el modelo de Diebold and Nerlove (1989) y el modelo DF-GARCH (Alessi et al. (2006)), y los que la asumen en la distribución condicionada, como el modelo FACTOR-ARCH (Engle (1987)), la familia de modelos GARCH ortogonales (Alexander (2001), van der Weide (2002), Lanne and Saikkonen (2007)), y el modelo CUC-GARCH (Fan et al. (2008)).

En el capítulo 3, presentamos un nuevo procedimiento, llamado FOTBI, para aplicar ICA a series temporales. Dado un conjunto de series temporales multivariantes, FOTBI es un algoritmo diseñado para extraer los componentes independientes y no-Gaussianos que generan dichos datos. Para ello, FOTBI propone la diagonalización conjunta de varias matrices de cumulantes temporales de cuarto orden. Así, FOTBI utiliza tanto la no-Gaussianidad como la estructura temporal de los datos, y puede verse como una extensión del algoritmo JADE (Cardoso and Souloumiac (1993)) que sólo tiene en cuenta la no-Gaussianidad de los datos, y del algoritmo SOBI (Belouchrani et al. (1997)) que se basa en la estructura temporal de las observaciones. Los experimentos de Monte Carlo muestran la eficiencia del FOTBI para estimar componentes independientes que son series temporales no lineales.

El capítulo 4 trata el problema de predicción y extracción de señal en series temporales multivariantes. Se presenta ICA como un procedimiento automático de extracción de señal. Se aplica ICA al problem de descomposición de una serie temporal, y se estiman los componentes de interés, tendencia, ciclo y estacionalidad, sin asumir ninguna estructura a-priori. La ventaja de ICA es que los ICs son, por hipótesis, estadísticamente independientes, y por tanto, los estimadores ICA para la tendencia, la estacionalidad, y el ciclo, van a ser indepen-

dientes entre sí de modo natural. Los resultados de las simulaciones demuestran que FOTBI puede considerarse un primer paso para definir un procedimiento automático de extracción de señal. Este resultado se confirma en la aplicación empírica, al identificar los componentes de tendencia y estacionalidad de las series del IPI de Alemania, Italia, Francia, y España.

También en el capítulo 4 se propone un procedimiento para predecir un conjunto multivariante de series temporales utilizando sólo un número reducido de ICs. Nuestro método se basa en la independencia estadística de los ICs. La idea es predecir los ICs utilizando modelos univariantes y utilizar esas predicciones de modo que, combinándolas con los pesos de la matriz de carga, se obtenga las predicciones para las series originales. Para analizar la eficiencia de nuestro procedimiento, predecimos las cuatro series del IPI mencionadas anteriormente utilizando los componentes de tendencia y estacionalidad estimados con ICA. Los resultados muestran el buen comportamiento del FOTBI, especialmente en el medio ($h = 6$) y largo ($h = 12$) plazo. En el corto plazo ($h = 1, 3$) no hay diferencias significativas entre las predicciones dadas por FOTBI y las dadas por los modelos de referencia de los IPIs (modelos ARIMA univariantes identificados con la especificación automática del programa TRAMO/SEATS).

List of Acronyms

AMUSE	Algorithm for Multiple Unknown Signals Extraction
ARCH	AutoRegressive Conditional Heteroskedasticity
ARIMA	AutoRegressive Integrated Moving Average
ARMA	AutoRegressive Moving Average
BSS	Blind Source Separation
CUC-GARCH	Conditionally Uncorrelated Components GARCH
DF-GARCH	Dynamic Factor-GARCH
DFM	Dynamic Factor Model
FA	Factor Analysis
FastICA	Fast Independent Component Analysis
FF-GARCH	Full Factor GARCH
FOTBI	Fourth-Order Temporal Blind Identification
GARCH	Generalized AutoRegressive Conditional Heteroskedasticity
GED	Generalized Error Distribution
GICA-GARCH	Generalized Independent Component Analysis GARCH
GMAE	Geometric Mean Absolute Error
GOF-GARCH	Generalized Orthogonal Factor GARCH
GO-GARCH	Generalized Orthogonal GARCH
GRW	Generalized Random Walk
HOS	Higher-Order Statistics
ICs	Independent Components
ICA	Independent Component Analysis
IPI	Industrial Production Index
IRW	Integrated Random Walk
JADE	Joint Approximate Diagonalization of Eigen-matrices
LLT	Local Linear Trend
MAPE	Mean Absolute Percentage Error
MASE	Mean Absolute Scale Error
MdRAE	Median Relative Absolute Error
MGARCH	Multivariate GARCH
MMSE	Minimum Mean Square Error
MSE	Mean Squared Error
O-GARCH	Orthogonal GARCH
PCs	Principal Components
PCA	Principal Component Analysis
PP	Projection Pursuit
RMSE	Root Mean Squared Error
RV	Realized Variance
RW	Random Walk
SEATS	Signal Extraction in ARIMA Time Series
SOBI	Second-Order Blind Identification
SOS	Second-Order Statistics
SRW	Smoothed Random Walk
STAMP	Structural Time Series Analyser, Modeller and Predictor
SUTSE	Seemingly Unrelated Time Series Equations
TDSEP	Temporal Decorrelation Source SEParation
TRAMO	Time series Regression with ARIMA noise, Missing values and Outliers
VARIMA	Vectorial ARIMA
VARMA	Vectorial ARMA

Chapter 1

Introduction

The aim of this thesis is to analyze the performance of independent component analysis (ICA) when it is applied to a vector of non-Gaussian time series in order to find an 'interesting' representation of the observations. First, we give an introduction to the ICA methodology and how it performs on estimating a set of non-Gaussian and statistically independent latent factors. Second, we review some basic ideas of multivariate time series analysis, paying special attention to well known dimension reduction techniques previously proposed in the literature. Third, we give an overview of the existing research that links ICA and time series data. Finally we outline the thesis.

1.1 Motivation

In many fields, such as Medicine, Engineering, Finance, and Economics among others, the amount of available data is continuously growing, and the data sets used in their empirical applications become very large. In addition, large data sets usually contain redundant information and/or are observed with high level of noise which make hard their analysis. Then, an important task in multivariate data analysis is to find a meaningful representation of the data which describe the 'interesting' features of the observations.

Principal component analysis (PCA), factor analysis (FA), and projection pursuit (PP) are classical examples of linear transformation methods proposed for finding projections of the data that have 'interesting' structure. PCA (Hotelling (1933)) and FA (Spearman (1904)) can be seen as dimension reduction techniques that transform the original data (highly correlated) in a set of a few underlying components that are maximally uncorrelated. Both methods compute the components of interest by using only the information contained in the data

covariance matrix. Although PCA and FA are very related, they are not identical. PCA takes into account all variability in the variables and the principal components are computed by maximizing the amount of total explained variability. FA tries to explain all the common variability by a set of common factors. Both methods are equivalent if the covariance matrix of the specific components in the FA model can be written as $\sigma^2\mathbf{I}$.

Empirical applications show that, if the observations are Gaussian distributed, the projections of the data computed by PCA and FA will reveal interesting features of the data. However, in many situations where the Gaussianity assumption does not hold, the representation of the data given by either the PCs or the latent factors (estimated by FA) could not describe the data in a meaningful way. For example, they cannot capture higher-order independence and possibly they cannot split the data into clusters. For non-Gaussian data, the information that is contained in the covariance matrix is not enough to obtain the 'interesting' projections of the data, and higher-order statistics are required. PP (Friedman and Tuckey (1974)) is a classical higher-order method that identifies the meaningful projections of the data (in the sense of displaying some relevant structure) as those that are further away from the Gaussian distribution (Huber (1985) and Jones and Sibson (1987), among others, argue that the Gaussian distribution is the least interesting one). Peña and Prieto (2001) proposed a new PP algorithm to identify clusters in multivariate data sets by projecting the observations onto the directions of both maximum and minimum kurtosis. Thus, when the data are projected in the direction that either maximizes or minimizes the kurtosis coefficient of the projections, the distance between clusters becomes as large as possible. In particular, Peña and Prieto (2001) showed that projecting the data in the directions of maximum kurtosis detects groups of outliers in the observations.

More recently independent component analysis (ICA) has emerged as an alternative methodology that uses higher-order information to find a set of underlying components (called independent components (ICs)) which provide a meaningful description of the data. The goal of ICA (Jutten and Hérault (1991), Comon (1994)) is to look for the projections of the data that become as independent as possible. That is, ICA defines the most 'interesting' components as those that are maximally independent. On the one hand, ICA is related to PP in the sense that ICA looks for the maximum independence of the components by maximizing their non-Gaussianity. In particular, if the non-Gaussianity is measured by using the kurtosis coefficient, ICA can be related to the Peña and Prieto (2001) procedure in the sense that

ICA is able to detect the outliers of the data set (Baragona and Battaglia (2007)). On the other hand, ICA can be seen as an useful extension of PCA, since the ICs are estimated by using higher-order statistics and then, under non-Gaussinity assumption, they will reveal more useful information than the PCs.

Although ICA was not formally defined until 1994 (Comon (1994)), the concept of statistical independence to estimate a set of underlying components without any a-priori information was firstly used by Hérault and Jutten (1986) and Jutten and Hérault (1991) in the context of neural networks. ICA is an active research topic that has been applied to several disciplines. Some examples are pattern recognition (Hyvärinen (1999b), Bingham (2001), and Bell and Sejnowski (1997) among others), visual brain theories (Vigàrio et al. (1998)), astronomy (Funaro et al. (2001)), telecommunications (Ristaniemi and Joutsensalo (1999)), and finance (Kiviluoto and Oja (1998)) among others.

In most of the previous applications, the observations are simply random vectors or data that do not exhibit temporal dependencies. However, multivariate time series data usually have a pronounce autocorrelation structure. Of course we can apply any of the previous methods, PCA, FA, PP, or ICA, to find the 'interesting' projections of multivariate time series data, but we might not obtain the desirable components (that is, the ones which describe the meaningful structure of the data). On the one hand, as we have discussed previously, PCA and FA only work properly under Gaussianity assumption, and their estimated components will not be represent the most important features of non-Gaussian data. On the other hand, although we know that both PP and ICA will provide meaningful components under non-Gaussianity assumption, neither PP nor ICA exploit all the available information in the observations (that is, the non-Gaussianity and autocorrelation structure of the data). Thus, whereas PP looks for the maximally non-Gaussian projections of the data without taking into account their temporal structure, ICA exploits either the non-Gaussianity or the time dependencies of the data to estimate the components, but not both features together. Then, it seems that the results given by ICA will be likely to be improved if we incorporate the non-Gaussianity as well as the time-structure to find the projections of time series data. The aim of this thesis is to propose ICA procedures for multivariate time series by combining those two criteria.

The rest of this chapter is organized as follows. Section 1.2 defines the ICA methodology and describes some of the principles used to estimate the ICs. Section 1.3 reviews well known

procedures proposed in the literature to reduce the dimensionality in multivariate time series. Section 1.4 introduces the existing second-order procedures to apply ICA in time series data. Finally, Section 1.5 gives an overview of the structure of this thesis.

1.2 Independent components analysis (ICA)

Independent component analysis (ICA) is an active interdisciplinary research topic since the early 80s. However, it was not formally defined until middle 90s (Comon (1994)). ICA uses the notion of statistical independence to estimate the underlying components which linearly generate the set of observations. In the following, we give an overview of the general description of ICA and discuss the relationship between ICA and classical multivariate techniques.

1.2.1 ICA model

Let $\mathbf{x} = (x_1, \ldots, x_m)'$ be an m-dimensional vector of observations. ICA assumes that $\mathbf{x}$ is linearly generated by a set of r, with $r \leq m$, mutually independent and non-Gaussian distributed components which are unknown. The ICA model may be written as,

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{1.1}$$

where $\mathbf{A}$ is an $m \times r$ unknown matrix of constant parameters, that is called mixing matrix, and $\mathbf{s} = (s_1, \ldots, s_r)'$ is the vector of non-Gaussian and mutually independent underlying components, which are called independent components (ICs). In addition, it is assumed without loss of generality that $E\{\mathbf{x}\} = \mathbf{0}$. Given a random sample of $\mathbf{x}$, $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$, the aim of ICA is to estimate both, $\mathbf{A}$ and $\mathbf{s}$, only from the observations. Thus, ICA tries to find a linear transformation of the data,

$$\widehat{\mathbf{s}} = \mathbf{B}\mathbf{x} \tag{1.2}$$

where the separating matrix, $\mathbf{B}$, of size $r \times m$, is such that the components of $\widehat{\mathbf{s}}$ become as independent as possible. However, assuming statistical independence on $\mathbf{s}$ is not enough to guarantee the identifiability of model (1.1), and some additional assumptions should be made (Comon (1994)):

1. The number of observed variables cannot be greater than the number of ICs, $r \leq m$. Moreover, the mixing matrix is assumed to be a full rank matrix, $rg(\mathbf{A}) = r$.

2. Calling $\mathbf{a}_i$ to the columns of $\mathbf{A}$, we have from (1.1) that $\mathbf{x} = \sum_{i=1}^{r} \mathbf{a}_i s_i$ and since both $\mathbf{A}$ and $\mathbf{s}$ are unknown, for any $\alpha_i \neq 0$, we also have $\mathbf{x} = \sum_{i=1}^{r} \left(\frac{1}{\alpha_i}\mathbf{a}_i\right)(s_i\alpha_i)$. Then, both s_i as well as $s_i\alpha_i$ could be the ICs of $\mathbf{x}$, but their variances take different values. To avoid such indeterminacy, the variances of the ICs are fixed to be equal to one, $var(\mathbf{s}_t) = \mathbf{I}_r$. However, the ICs are still indeterminate with respect their sign.

3. No more than one IC could be Gaussian distributed. If there are two or more Gaussian components, the observations will become more and more Gaussian and then, the components cannot be separated (by the central limit theorem, the sum of a set of independent random variables tends to be Gaussian distributed). Under Gaussianity, ICA and PCA are equivalent, and the ICs will be indeterminate under rotations.

An important drawback of ICA is that despite of imposing those three identifiability conditions on (1.1), the ordering of the ICs is still ambiguous.

Most of the literature related to ICA deals with the basic ICA linear model. This model is a particular case (1.1) but assuming, by simplicity, that the dimension of $\mathbf{x}$ equals the dimension of $\mathbf{s}$, $r = m$. However, note that this need not necessarily be the case in many empirical applications, where the interest is to estimate only a small number of components. In that case, PCA is usually applied to reduce the dimension of the observations (from m to r) and then, the basic ICA model on the r-dimensional vector of observations is hold. In the next section, we review different approaches to solve the basic ICA model when it is applied to a set of random vectors. Proposals to apply ICA on multivariate time series data will be discussed in Section 1.4.

1.2.2 Independent components estimation

Most of the algorithms proposed to solve the basic ICA model usually incorporate additional constraints which yield further simplification of the procedure. One popular constraint consists on imposing orthogonality on the mixing matrix, $\mathbf{A}$, which, in the basic ICA model, is a square matrix of order m. In that way, the ICA solution is restricted to the space of orthogonal matrices and the number of parameters to be estimated is reduced from m^2 in $\mathbf{A}$ to $\frac{m(m+1)}{2}$ in the new orthogonal mixing matrix. The orthogonality constrain is naturally included in the model by the multivariate standardization of the observations that transforms the original

data into a set of new observations that have zero mean and identity covariance matrix. That is,

$$\mathbf{z} = \mathbf{V}\mathbf{x} \tag{1.3}$$

where $\mathbf{V}$ is an $m \times m$ matrix such that $E\{\mathbf{z}\} = \mathbf{0}$ and $\mathbf{V_z} = E\{\mathbf{z}\mathbf{z}'\} = \mathbf{I}_m$. ICA procedures usually applies PCA to perform the multivariate standardization of $\mathbf{x}$ as follows. Let $\mathbf{V_x} = E\{\mathbf{x}\mathbf{x}'\}$ be the covariance matrix of $\mathbf{x}$. The eigenvalue decomposition of $\mathbf{V_x}$ is given by $\mathbf{V_x} = \mathbf{Q}\mathbf{D}\mathbf{Q}'$, where $\mathbf{Q}_{m\times m}$ is the orthogonal matrix of eigenvectors (in columns) and $\mathbf{D}_{m\times m} = diag\,(d_1, ..., d_m)$, with $d_1 \geq \ldots \geq d_m$, is the matrix of eigenvalues. Taking $\mathbf{V} = \mathbf{D}^{-1/2}\mathbf{Q}'$ in (1.3), the original observations, $\mathbf{x}$, are multivariate standardized, and the ICA model (1.1) can be written in terms of the standardized data, $\mathbf{z}$, as,

$$\mathbf{z} = \widetilde{\mathbf{A}}\mathbf{s} \tag{1.4}$$

where $\widetilde{\mathbf{A}} = \mathbf{V}\mathbf{A} = \mathbf{D}^{-1/2}\mathbf{Q}'\mathbf{A}$ is the new mixing matrix of size $m\times m$ that is clearly orthogonal. Moreover, this procedure can also be applied to reduce the dimensionality of the data from m to r by discarding the $m - r$ smallest eigenvalues of $\mathbf{V_x}$ (then, $\mathbf{z}$ and $\widetilde{\mathbf{A}}$ will be, respectively, the $r \times 1$ vector of observations and the $r \times r$ orthogonal mixing matrix).

The estimates of the ICs will be given by those linear combination of the standardized data,

$$\mathbf{y} = \mathbf{W}\mathbf{z}, \tag{1.5}$$

where $\mathbf{W}$ is an orthogonal matrix of size $m \times m$ (or $r \times r$ if the dimension is reduced), that are maximally independent. Depending on how the statistical independence is measured, Hyvärinen et al. (2001) distingue three ICA estimation principles: maximizing the non-Gaussianity (Delfosse and Loubaton (1995), Hyvärinen and Oja (1997), Hyvärinen (1999a), and Cardoso and Souloumiac (1993) among others), minimizing the mutual information (Bell and Sejnowski (1995) and Amari et al. (1996) among others), or maximizing the likelihood of the components (Gaeta and Lacoume (1990) and Pham et al. (1992) among others).

Maximization of non-Gaussianity

Many ICA procedures estimate the ICs by focusing on their non-Gaussianity. They maximize the independence of the ICs by maximizing their non-Gaussianity. There are several ways to measure non-Gaussianity and each of them leads to different ICA algorithms. The first alternative measures the non-Gaussianity by using the kurtosis coefficient. The kurtosis of the i-th IC, $y_i = \mathbf{w}_i'\mathbf{z}$ (see (1.5)), is given by: $kurt(y_i) = E\{y_i^4\} - 3E\{y_i^2\}^2$. $kurt(y_i)$ can take

positive or negative values and it is equal to zero if y_i is Gaussian. The idea of measuring the non-Gaussianity with the kurtosis was firstly implemented by Delfosse and Loubaton (1995). However, since the kurtosis is highly sensitive to outliers, it will not be a robust measure of non-Gaussianity.

As an alternative to the kurtosis coefficient to solve the lack of robustness, the second approach proposes to use the entropy to measure the non-Gaussianity. Since, for a given covariance matrix, the distribution that has highest entropy is the Gaussian distribution (Cover and Thomas (2001)), the principle of maximizing the non-Gaussianity is equivalent to minimizing the entropy. The entropy of a random vector, $\mathbf{y}$, is defined as $H(\mathbf{y}) = -E\{logp_{\mathbf{y}}(\xi)\}$, where $p_{\mathbf{y}}(\xi)$ is the probability that $\mathbf{y}$ is in the state ξ. However, since the entropy is not invariant to linear transformations ($H(\mathbf{Wz}) = H(\mathbf{z}) + \log|\det \mathbf{W}|$), it is commonly accepted to use the negentropy, instead of the entropy, as a measure of non-Gaussianity. Negentropy can be seen as a measure of distance from Gaussianity that is defined by:

$$J(\mathbf{y}) = H(\mathbf{y}_{Gauss}) - H(\mathbf{y}), \tag{1.6}$$

where $\mathbf{y}$ is a random vector (Gaussian or non-Gaussian) and $\mathbf{y}_{Gauss}$ is a random vector whose covariance matrix is equal to that of $\mathbf{y}$. Since negentropy is invariant to linear transformations, is always non-negative, and is zero iff $\mathbf{y}$ is a Gaussian vector, negentropy can be seen as a 'good' index to measure non-Gaussianity. However, computing the negentropy as in (1.6) is quite difficult and it is usual to take into account different approximations. For example, Comon (1994) proposes an approximation based on polynomial functions, $J(\mathbf{w}_i'\mathbf{z}) \approx \frac{1}{12}E\{(\mathbf{w}_i'\mathbf{z})^3\}^2 + \frac{1}{48}kurt(\mathbf{w}_i'\mathbf{z})^2$, and Hyvärinen (1998b) uses a non-quadratic function, $G(\cdot)$ (usually $G(\cdot) = tanh(\cdot)$), and approximates the negentropy by $J(\mathbf{w}_i'\mathbf{z}) \approx E\{G(\mathbf{w}_i'\mathbf{z})\} - E\{G(\mathbf{y_{Gauss}})\}^2$. The FastICA algorithm (Hyvärinen and Oja (1997), Hyvärinen (1999a)), that will be introduced in the next chapter, uses the kurtosis coefficient and/or the negentropy to measure the non-Gaussianity of the components.

The third approach to measure the non-Gaussianity of the components is based on higher-order cumulants. Higher-order cumulants are closely related to higher-order moments and both provide the same statistical information. However, higher-order cumulants have some useful statistical properties that make them preferable to solve the ICA problem. Thus, for instance, the cumulants of order higher than two of Gaussian random vectors are equal to zero. Furthermore, it is easy to see that two (or more) random vectors are independent

if their cross-cumulants of order higher than two are equal to zero (for more details about cumulants and their properties, see Chapter 3). JADE (Cardoso and Souloumiac (1993)), that will will be presented in Chapter 2, is a well known ICA algorithm that estimates the ICs by maximizing their non-Gaussianity using fourth-order cumulants.

Minimization of mutual information

In information theory, the mutual information of two or more random variables is commonly used to measure the statistical dependence among them. The mutual information of an r-dimensional random vector, $\mathbf{y} = (y_1, \ldots, y_r)'$, is defined by (Cover and Thomas (2001)):

$$I(y_1, \ldots, y_r) = \sum_{i=1}^{r} H(y_i) - H(\mathbf{y}). \tag{1.7}$$

Moreover, mutual information can also be expressed as the Kullback-Leibler (KL) distance between the product of the marginal distributions of the two (or more) random variables and the random variables' joint distribution. That is,

$$I(y_1, \ldots, y_r) = KL\left(p_{\mathbf{y}}(\xi), \prod_{i=1}^{r} p_i(y_i)\right) \tag{1.8}$$

where $p_{\mathbf{y}}(\xi)$ is the joint distribution of $\mathbf{y}$ and $p_i(y_i)$ are the marginal distributions of $\{y_i\}_{i=1}^{m}$. Thus, using the concept of KL distance, it is easy to see that the mutual information is $\mathbf{y}$ is always a non-negative measure of independence and $I(y_1, \ldots, y_r) = 0 \; iff$ $\mathbf{y}$ is a vector of independent random variables. Therefore the ICA procedures which try to find the directions, $\{\mathbf{w}_i\}_{i=1,\ldots,r}$, that minimize $I(\mathbf{w}_1'\mathbf{z}, \ldots, \mathbf{w}_r'\mathbf{z})$, are actually looking for the components, $\mathbf{w}_i'\mathbf{z}$, that are maximally independent. Examples of algorithms that follow this approach are given by Bell and Sejnowski (1995) and Amari et al. (1996) (InfoMax) among others. Since the mutual information is defined from the entropy, as FastICA does, those algorithms computes the mutual information by using different approximations for the entropy.

Maximization of likelihood

The classical estimation principle of maximum likelihood was also applied to estimate the ICs (see Gaeta and Lacoume (1990) and Pham et al. (1992) among others). In particular, Pham et al. (1992) showed that, from model (1.4),

$$p_{\mathbf{z}}(z) = p_{\mathbf{s}}(s) \left|\frac{\partial \mathbf{s}}{\partial \mathbf{z}}\right| = p_{\mathbf{s}}(s) \left|\det \mathbf{W}\right|,$$

i.e.,

$$p_{\mathbf{z}}(z) \equiv L(\mathbf{W}) = \prod_{i,j=1}^{r} p_i(\mathbf{w}_i'\mathbf{z}_{ij}) \left|\det \mathbf{W}\right| \tag{1.9}$$

where $p_i(\cdot)$ is the marginal distribution of the i-th IC. Taking logs in (1.9), we have the contrast function that is used to estimate the ICs by the maximum likelihood principle:

$$\log L(\mathbf{W}) = \sum_{i,j=1}^{r} \log p_i(\mathbf{w}_i' \mathbf{z}_{ij}) + n \log |\det \mathbf{W}|$$

The main drawback of this approach is that the marginal distributions of the ICs are unknown and non-parametric estimation methods are required to estimate

Despite that the properties and the optimality criteria of the three estimation principles are quite different, many theoretical links are established among them. For example, Cardoso (1997) presented the mathematical equivalence between the mutual information and maximum likelihood approaches. Moreover, Lee et al. (2000) showed that negentropy maximization (that is, maximizing non-Gaussianity) also has equivalent properties. Therefore, since the three approaches are mathematically equivalent, it seems that there exist a unifying framework for ICA, where the ICs are obtained as the solution to the following optimization problem:

$$\max_{\|\mathbf{w}_i\|^2=1} E\{\pm G(\mathbf{w}_i' \mathbf{z})\} \tag{1.10}$$

where $G(\cdot)$ is a non-quadratic function. Then, choosing the estimation principle is equivalent to choose $G(\cdot)$, and the statistical properties of the ICs estimates will depend on that election.

In this thesis, we will focus on the estimation of the ICs by the maximization of the non-Gaussianity of the components.

1.2.3 Relation to other multivariate methods

In this section, we analyze the relationship between ICA and classical methods previously used in the literature to simplify the structure of large data sets: principal components analysis (PCA), factor analysis (FA), and projection pursuit (PP). ICA, as well as PCA, FA, and PP, is based on the idea of finding an 'interesting representation' of the data by the projection of the observations, but the concept of 'interesting representation' differs from one method to another.

In PCA, the 'interesting representation' of the data set is given by those projections of the data that are mutually uncorrelated and explain as much of the variability in the observations as possible. These projections, called principal components, will be statistically independent only if the observations are Gaussian distributed. Thus, ICA can be seen as a generalization

of PCA in the sense that the ICs will be estimated by the rotation of the principal components that makes them as independent as possible.

ICA is related to FA in the sense that model (1.1) is quite similar to the model used in classical FA. However, FA assumes uncorrelated underlying components, whereas ICA assumes non-Gaussian factors in addition to their statistical independence. Then, ICA can be seen as a non-Gaussian FA (Hyvärinen and Kano (2003)) that uses higher-order information to uniquely identify the model (1.1). By the non-Gaussianity assumption on the underlying components, ICA will be very helpful to determine the optimal factor rotation, without using traditional methods like varimax. That is, estimating the ICs is equivalent to find the optimal rotation of the latent factors in FA.

If ICA measures the statistical independence of the components in terms of non-Gaussianity, it could be seen as an special case of PP, which tries to find those projections of the data that are far away from Gaussianity. Therefore, measures of non-Gaussianity such as kurtosis or negentropy could be considered as projection indexes in PP. ICA differs from PP in the fact that ICA is based on a data generating model that makes easier the statistical inference and the prediction on the data.

1.3 Multivariate time series models

When we have several time series data, dynamic relationships usually appear among them, and in order to capture those interactions, the data should be modelled using a multivariate framework. Vector ARMA models and simultaneous equation econometric models are well known approaches for examining temporal relationships among multiple time series. However, since for both modelling approaches the number of parameters to estimate grows rapidly with the number of series considered, they are not very convenient for large dimensional data sets. Then, the so-called curse of dimensionality is an important problem in multivariate time series analysis. Next, we present some popular procedures for dimension reduction.

1.3.1 Dimension reduction techniques

The problem of dimensionality reduction have attracted great attention in multivariate time series literature since late 70s. Most of techniques proposed to achieve dimensionality reduction in time series data are extensions of multivariate classical methods such as PCA and FA

among others. Those methods are based on the idea that there exist a few underlying components, which are given by 'interesting' linear combinations of the data, that represent the relevant dynamic features of the observations. Tiao and Tsay (1989) have supported the usefulness of using linear combination of the time series data to achieve dimensionality reduction. Then, the general problem of dimension reduction in multivariate time series can be formulated as follows: let $\mathbf{x}_t = (x_{1t}, ..., x_{mt})'$ be an m-dimensional time series vector of observations. The aim to achieve dimensionality reduction in $\mathbf{x}_t$ is to look for a linear transformation,

$$\mathbf{y}_t = \mathbf{M}\mathbf{x}_t \tag{1.11}$$

where $\mathbf{M}$ is the parameter matrix of size $r \times m$, $r < m$, and $\mathbf{y}_t = (y_{1t}, ..., y_{rt})'$ are the r linear combinations of the observations that capture the dynamic relationships among them. On the one hand, if $\mathbf{x}_t$ is a stationary time series vector, it will interesting to find those linear combinations of $\mathbf{x}_t$ that are white noise processes. Then, the r remainder will represent the dynamic structure of $\mathbf{x}_t$. On the other hand, if $\mathbf{x}_t$ is non-stationary, its dynamics will be given by the linear combination that are non-stationary and the interest will be to estimate the stationary linear combinations (or cointegration relationships) of $\mathbf{x}_t$. Then the dimension of $\mathbf{x}_t$ is reduced to the space generated by the r non-stationary linear combinations.

Literature on dimensionality reduction in multivariate time series is vast (see, for example, Peña and Poncela (2006a) for a survey). Here we review three methods that come from standard multivariate techniques: principal components in time series data (Stock and Watson (2002)), canonical analysis (Box and Tiao (1977)), and dynamic factor model (Geweke and Singleton (1981), Brillinger (1981), Peña and Box (1987), Forni et al. (2000), and Peña and Poncela (2006b) among others).

Principal components in time series data

Principal component analysis (PCA) has been applied to build economic indicators from multivariate time series data (see, for example, Stock and Watson (2002)). This approach, that is an extension of classical PCA, uses second-order information to separate the stationary underlying components from those linear combinations of the data that are white noise processes. Let us assume that the observations, $\mathbf{x}_t$, is a vector of m stationary time series with zero mean. Let $\mathbf{\Gamma}_{\mathbf{x}}(k) = E\{\mathbf{x}_{t-k}\mathbf{x}_t'\}$ be the lagged k covariance matrices of $\mathbf{x}_t$, $\forall k$. The aim of PCA is to find the r orthogonal directions, $\mathbf{m}_i\mathbf{m}_i' = 1$ where $\mathbf{m}_i$ is the i-th column of $\mathbf{M}$,

such that the projection of the data,

$$y_{it} = \mathbf{m}_i' \mathbf{x}_t \quad \forall i = 1, ..., r, \tag{1.12}$$

explain maximum percentage of total variability. The linear combination of the data given by (1.12) are the principal components (PCs). The solution to the PCA problem is given by: $\mathbf{\Gamma_x}(0)\mathbf{m}_i = \lambda_i \mathbf{m}_i$. That is, the directions of projections, $\{\mathbf{m}_i\}_{i=1}^r$, which define the PCs as in (1.12), correspond to the eigenvectors of $\mathbf{\Gamma_x}(0)$. Moreover, the variance of the PCs defined as in (1.12) are the eigenvalues of $\mathbf{\Gamma_x}(0)$, $\lambda_i = \mathbf{m}_i' \mathbf{\Gamma_x}(0)\mathbf{m}_i$. Then, if the eigenvalues are sorted decreasingly, $\lambda_1 \geq \ldots \geq \lambda_r$, it is clear that the first PC, y_{1t}, will be the projection of $\mathbf{x}_t$ in the direction given by the eigenvector, $\mathbf{m}_1$, associated to the largest eigenvalue.

If the data have some dynamics, assuming that the PCs are generated by $y_{it} = \Psi(B)u_t$, we will have: $var(y_{it}) = \sigma_u^2 \sum_i \Psi_i^2$ where $\sigma_u^2 = var(u_t)$. According to this results, the PCs that are close to be non-stationary (they are the ones associated to the largest eigenvalues) could be separated to those PCs that are white noise processes (which correspond to the ones that come from the smallest λ_i). If there is any x_{it} that is given by a linear combination of some x_{jt}, with $j \neq i$, $\mathbf{\Gamma_x}(0)$ will have some null eigenvalues. In that case, those eigenvalues that are equal to zero will provide the linear combinations among the components of $\mathbf{x}_t$.

If $\mathbf{x}_t$ is a non-stationary time series vector, $\mathbf{x}_t \sim I(d)$, the matrices $\mathbf{\Gamma_x}(k)$ are not well-defined. Peña and Poncela (2006b) define the generalized sample covariance matrices as:

$$\mathbf{C_x}(k) = \frac{1}{T^{2d}} \sum (\mathbf{x}_{t-k} - \overline{\mathbf{x}})(\mathbf{x}_t - \overline{\mathbf{x}})', \tag{1.13}$$

where $\overline{\mathbf{x}} = \frac{1}{T} \sum \mathbf{x}_t$. These matrices play the same role than the sample covariance matrices in the stationary case for identification purposes. Thus, using the same optimality criterion as before, we will get analogous results: whereas the non-stationary components will be the PCs associated to the largest eigenvalues of $\mathbf{C_x}(0)$, the PCs which come from the smallest eigenvalues will correspond to white noise processes.

Canonical analysis

The canonical analysis, that was proposed as an extension of PCA, solves the problem of dimensionality reduction using the concept of predictability (Box and Tiao (1977)). The idea of canonical analysis is finding the linear combinations of $\mathbf{x}_t$, defined as in (1.12), which have maximum (or minimum) predictability. These components are called canonical variables.

Given a time series vector, $\mathbf{x}_t$, we can always write the orthogonal decomposition,

$$\mathbf{x}_t = \widehat{\mathbf{x}}_{t-1}(1) + \varepsilon_t \tag{1.14}$$

where $\widehat{\mathbf{x}}_{t-1}(1)$ is the one steap ahead prediction of $\mathbf{x}_t$, and ε_t is the one step ahead prediction error with zero mean and $\mathbf{\Sigma}$ covariance matrix. Since $\widehat{\mathbf{x}}_{t-1}(1)$ and ε_t are uncorrelated, $\mathbf{\Gamma}_{\mathbf{x}}(0)$ can be decomposed as: $\mathbf{\Gamma}_{\mathbf{x}}(0) = \mathbf{F}_{\mathbf{x}}(0) + \mathbf{\Sigma}$, where $\mathbf{F}_{\mathbf{x}}(0) = E\{\widehat{\mathbf{x}}_{t-1}(1)\widehat{\mathbf{x}}_{t-1}(1)'\}$. By (1.14), the canonical variables given by (1.12) can be written as $\mathbf{y}_t = \widehat{\mathbf{y}}_{t-1}(1) + \mathbf{u}_t$, where $\mathbf{u}_t = \mathbf{m}_i'\varepsilon_t$. Then, the predictability of $\mathbf{y}_t$, p, is defined as (Box and Tiao (1977)):

$$p = \frac{\mathbf{F}_{\mathbf{y}}(0)}{\mathbf{\Gamma}_{\mathbf{y}}(0)} = \frac{\mathbf{m}_i'\mathbf{F}_{\mathbf{x}}(0)\mathbf{m}_i}{\mathbf{m}_i'\mathbf{\Gamma}_{\mathbf{x}}(0)\mathbf{m}_i} = 1 - \frac{\mathbf{m}_i'\mathbf{\Sigma}\mathbf{m}_i}{\mathbf{m}_i'\mathbf{\Gamma}_{\mathbf{x}}(0)\mathbf{m}_i}, \;\; 0 \leq p \leq 1.$$

It can be shown that the direction of projection, $\mathbf{m}_i$, which maximizes the predictability of y_{it} satisfies: $\mathbf{Q}\mathbf{m}_i = \lambda_i\mathbf{m}_i$, where $\mathbf{Q} = \mathbf{\Gamma}_{\mathbf{x}}(0)^{-1}\mathbf{F}_{\mathbf{x}}(0) = \mathbf{I} - \mathbf{\Gamma}_{\mathbf{x}}(0)^{-1}\mathbf{\Sigma}$. That is, y_{it} will have maximum predictability if $\mathbf{m}_i$ is the eigenvector of $\mathbf{Q}$ that is associated to the largest eigenvalue. According to this result, we can separate the canonical variables that are stationary from the ones that are non-stationary. On the one hand, the stationary canonical variables will be those projections of $\mathbf{x}_t$ in the direction of the eigenvectors of $\mathbf{Q}$ that are associated to the smallest eigenvalues. On the other hand, the non-stationary canonical variables will be those linear combinations of maximum predictability (that is, those y_{it} where $\mathbf{m}_i$ are the eigenvectors of $\mathbf{Q}$ associated to the eigenvalues close to one). Then, since it is possible to find a stationary linear combination of non-stationary time series, it could be said that canonical analysis introduces the concept of cointegration.

Dynamic factor models

Despite that previous methods are very useful for understanding and simplifying the dynamic structure of a time series vector, none of them can be considered model-based methodologies and therefore, it is difficult to make statistical inference or prediction from them. As an alternative for those purposes, dynamic factor models (DFM) were introduced (see, for example, Geweke and Singleton (1981), Brillinger (1981), Peña and Box (1987), Forni et al. (2000), and Peña and Poncela (2006b) among others).

The DFM assumes that all the common dynamic structure of $\mathbf{x}_t$ comes from through a set of few common factors. The data is assumed to be generated by a set of r underlying factors, $r < m$, that represent the common dynamic structure of the m time series data, plus a noise

term that represents the specific component of each x_{it}. The DFM can be formulated as:

$$\mathbf{x}_t = \mathbf{\Lambda}\mathbf{f}_t + \mathbf{n}_t \tag{1.15}$$

where $\mathbf{f}_t$ is the r dimensional vector of common factors (that may be stationary or non-stationary) which are assumed to follow a VARIMA(p, d, q) model: $\mathbf{\Phi_f}(B)\Delta^d\mathbf{f}_t = \mathbf{\Theta_f}(B)\mathbf{e}_t$; $\mathbf{\Lambda}$ is an unknown $m \times r$ matrix of parameters, that is called loading matrix; $\mathbf{n}_t$ is the m dimensional vector of idiosyncratic components (some of them could be white noise process while others could have stationary dynamic structure). In general, it is assumed $\mathbf{n}_t \sim$ VARMA(p, q), i.e., $\mathbf{\Phi_n}(B)\mathbf{n}_t = \mathbf{\Theta_n}(B)\mathbf{a}_t$ where $\mathbf{\Phi_n}$ and $\mathbf{\Theta_n}$ are diagonal $m \times m$ matrices, and $\mathbf{a}_t \sim N(\mathbf{0}, \mathbf{\Sigma_a})$ with $\mathbf{\Sigma_a}$ diagonal. In addition, two extra assumptions are made: (1) the covariance matrices of $\mathbf{e}_t$ and $\mathbf{n}_t$ are diagonal; (2) $\mathbf{e}_t$ and $\mathbf{a}_t$ are mutually uncorrelated for all lags. However, previous assumptions are not enough to guarantee the identifiability in (1.15): the set of common factors that generate $\mathbf{x}_t$ will be either correlated or uncorrelated. To avoid identifiability problems, the loading matrix is assumed to be orthogonal, $\mathbf{\Lambda}'\mathbf{\Lambda} = \mathbf{I}_r$. In that way, the common factors, $\mathbf{f}_t$, will be uncorrelated but they are indeterminate under rotations.

If $\mathbf{x}_t$ is a vector of stationary time series, the r common factors $\mathbf{f}_t$ will be stationary, $\mathbf{n}_t$ will be a multivariate white noise process. In that case, model (1.15) is analogous to the model analyzed by Peña and Box (1987) that satisfies:

$$\begin{aligned} \mathbf{\Gamma_x}(0) &= E\{\mathbf{x}_t\mathbf{x}_t'\} = \mathbf{\Lambda}\mathbf{\Gamma_f}(0)\mathbf{\Lambda}' + \mathbf{\Sigma_n} \\ \mathbf{\Gamma_x}(k) &= E\{\mathbf{x}_{t-k}\mathbf{x}_t'\} = \mathbf{\Lambda}\mathbf{\Gamma_f}(k)\mathbf{\Lambda}', \forall k \geq 1, \end{aligned} \tag{1.16}$$

where $\mathbf{\Sigma_n} = E\{\mathbf{n}_t\mathbf{n}_t'\}$ is as in (1.15) and $\mathbf{\Gamma_f}(k) = E\{\mathbf{f}_{t-k}\mathbf{f}_t'\}$, $\forall k \geq 0$. From (1.16), we have the following remarks: (i) $\mathbf{\Gamma_x}(k)$, $\forall k \geq 1$, are symmetric matrices; (ii) $rank(\mathbf{\Gamma_x}(k)) = r =$number of common factors, $\forall k \geq 1$; (iii) the eigenvectors of $\mathbf{\Gamma_x}(k)$, $\forall k \geq 1$, are the estimates of the columns of $\mathbf{\Lambda}$; (iv) the eigenvalues of $\mathbf{\Gamma_x}(k)$, $\forall k \geq 1$, are the values for the variances of the common factors.

If $\mathbf{x}_t$ is a stationary time series vector, it has been shown that it is enough to take $k = 1$ to identify the model (1.15). That is, both the common factors, $\mathbf{f}_t$, and the loading matrix, $\mathbf{\Lambda}$, can be estimated by the eigenvalue decomposition of $\mathbf{\Gamma_x}(1)$ as follows: $\mathbf{\Gamma_x}(1)$ will have r eigenvalues different from zero whose associated eigenvectors will be the columns of $\mathbf{\Lambda}$, and the projections of the data in the directions of those eigenvectors will be the estimates of the common factors. The remainder $m - r$ eigenvalues of $\mathbf{\Gamma_x}(1)$ will be close to zero, and correspond to those linear combination of the data that are white noise processes.

If $\mathbf{x}_t \sim I(d)$, Peña and Poncela (2006b) propose to decompose $\mathbf{f}_t$ and $\mathbf{\Lambda}$ in such a way that the stationary factors are separated from the non-stationary ones as: $(\mathbf{f}_t)_{r\times 1} = \begin{bmatrix}(\mathbf{f}_{1t})_{r_1\times 1}\\(\mathbf{f}_{2t})_{r_2\times 1}\end{bmatrix}$ and $\mathbf{\Lambda}_{m\times r} = \begin{bmatrix}(\mathbf{\Lambda}_1)_{m\times r_1} & (\mathbf{\Lambda}_2)_{m\times r_2}\end{bmatrix}$. Thus, they reformulate model (1.15) as:

$$\mathbf{x}_t = \mathbf{\Lambda_1 f}_{1t} + \mathbf{\Lambda_2 f}_{2t} + \mathbf{n}_t$$

Peña and Poncela (2006b) show that, in the non-stationary DFM, the identification of the non-stationary factors is carried out from the generalized sample covariance matrices defined in (1.13), which play the same role as the sample covariance matrices in the stationary case. It is shown that $\mathbf{C_x}(k)$ converges to $\mathbf{\Gamma_x}(k)$, $\forall k \geq 1$, that is mainly driven by non-stationary components. Then, the r_1 non-stationary components, denoted by $\mathbf{f}_{1t}$, will be identify from $\mathbf{\Gamma_x}(1)$, following an analogous procedure as the one applied in the stationary case.

1.4 ICA and time series

The basic ICA model can be extended for a vector of m time series data as follows (see, for example, Hyvärinen (1998a)):

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t \tag{1.17}$$

where $\mathbf{A}$ is a parameter matrix of size $m \times r$ and $\{s_{it}\}_{i=1}^{r}$, with $r < m$, are the underlying components which are assumed to be statistically independent (they are the 'dynamic' ICs). To achieve the identifiability of the model (1.17), instead of assuming non-Gaussianity, $\mathbf{x}_t$ is assumed to be stationary and ergodic (Hyvärinen (1998a)).

As it is done in the static ICA model, $\mathbf{x}_t$ is standardized to have zero mean and identity covariance matrix, and model (1.17) can be rewritten as $\mathbf{z}_t = \widetilde{\mathbf{A}}\mathbf{s}_t$, where $\widetilde{\mathbf{A}} = \mathbf{\Gamma_x}(0)^{-\frac{1}{2}}\mathbf{A}$ is an $r \times r$ orthogonal matrix. Then, the r 'dynamic' ICs will be those linear combinations of the standardized data which become as independent as possible. That is,

$$\mathbf{s}_{t-k} = \mathbf{W}\mathbf{z}_{t-k},\ \forall k \geq 0$$

where $\mathbf{W}\left(\approx \widetilde{\mathbf{A}}^{-1}\right)$ is an $r \times r$ orthogonal matrix that maximizes the statistical independence $\mathbf{s}_t$. If $\mathbf{s}_t$ are statistically independent, then they will be uncorrelated: $E\{s_{i,t-k}s_{j,t}\} = 0$, $\forall i \neq j$, k. According to that, ICA time series literature proposes to estimate the 'dynamic' ICs making their time-delayed cross-correlations equal to zero. There are two different approaches, depending whether the optimality criterion proposes to diagonalize one or several covariance matrices, $\mathbf{\Gamma_s}(k)$, for $k \geq 1$, are presented in the literature.

The first approach, introduced by Tong et al. (1990), proposes to estimate the 'dynamic' ICs by the diagonalization of one of their time delayed covariance matrix, $\mathbf{\Gamma_s}(\cdot)$. This idea was implemented in the AMUSE algorithm (Tong et al. (1990)) that is based on the eigenvalue decomposition of $\mathbf{\Gamma_z}(k)$, for any $k \geq 1$ that is fixed a-priori. However, since the matrices $\mathbf{\Gamma_z}(k)$ are, in general, not symmetric, Tong et al. (1990) propose the following transformation,

$$\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k) = \frac{1}{2}\{\mathbf{\Gamma_z}(k) + \mathbf{\Gamma_z}(k)'\} \quad \forall k \geq 1 \tag{1.18}$$

where $\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)$ is now symmetric $\forall k \geq 1$. Since $\mathbf{W}$ is linear and orthogonal, (1.18) can be written as:

$$\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k) = \mathbf{W}'\overline{\mathbf{\Gamma}}_{\mathbf{s}}(k)\mathbf{W} \quad \forall k \geq 1 \tag{1.19}$$

where $\overline{\mathbf{\Gamma}}_{\mathbf{s}}(k)$ is a diagonal matrix $\forall k \geq 1$ (because of the diagonality of $\mathbf{\Gamma_s}(k)$). Therefore, from (1.19), $\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k) = \mathbf{W}'\mathbf{D}\mathbf{W}$, $\forall k \geq 1$, and once k is fixed, the eigenvalue decomposition is applied to $\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)$, and the eigenvectors of $\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)$ are the estimates of the rows of $\mathbf{W}$. Then, the 'dynamic' ICs are given by $\widehat{\mathbf{s}}_t = \widehat{\mathbf{W}}\mathbf{z}_t$, and the values of their k-th lag auto-covariances correspond to the eigenvalues of $\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)$. AMUSE is a quite fast algorithm, but it is efficient only when all the eigenvalues of $\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)$ are uniquely determined. Then, k should be chosen carefully to achieve that all the eigenvalues of $\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)$ are distinct. That is, in such a way that there will not be two 'dynamic' ICs with identical k-th lag auto-covariances.

The second approach, trying to avoid the problematic of choosing k in AMUSE, proposes to consider several time lags. Let $K = \{1, ..., k_T\}$ be a set of multiple time lags. Then, as an extension of the idea introduced by Tong et al. (1990), the aim of this approach is to estimate the 'dynamic' ICs by the joint diagonalization of several time delayed covariance matrices, $\mathbf{\Gamma_s}(k)$, $\forall k \in K$. That is, $\mathbf{W}$ will be the transformation that makes the k-th cross-correlation of $\mathbf{s}_t$, $\forall k \in K$, to be equal to zero. Depending on how the lack of diagonality of $\mathbf{\Gamma_s}(k)$ is measured, different implementations have been presented in the literature.

Belouchrani et al. (1997) introduced the SOBI (Second Order Blind Identification) algorithm, where the lack of diagonality of any squared matrix, $\mathbf{M}$, is measured by the sum of the squares of its off-diagonal elements, that is: $\text{off}(\mathbf{M}) = \sum_{i \neq j} m_{ij}^2$. Then, the matrix $\mathbf{M}$ will be as diagonal as possible when the 'off' criterion is minimized. In Chapter 2, we will describe SOBI in more detail.

Kawamoto et al. (1997) defined an alternative measure of lack of diagonality as: $F(\mathbf{M}) = \sum_i \log m_{ii} - \log|\det \mathbf{M}|$. This measure is based on the idea that any $m \times m$ positive-definite

matrix, $\mathbf{M}$, satisfies that $\sum_{i=1}^{m} \log m_{ii} \geq \log |det\mathbf{M}|$, and the equality holds iff $\mathbf{M}$ is a diagonal matrix. This property, in addition to the scale invariance of $F(\cdot)$ (Pham and Cardoso (2001)), makes it a 'good' index to measure lack of diagonality.

Since $\overline{\mathbf{\Gamma}}_{\mathbf{s}}(k)$, $\forall k \in K$, is a positive definite matrix, Kawamoto et al. (1997) applied previous criterion, and considered a new contrast function to estimate $\mathbf{W}$,

$$J(\mathbf{W}) = \frac{1}{2} \sum_{k \in K} F\left(\overline{\mathbf{\Gamma}}_{\mathbf{s}}(k)\right) = \frac{1}{2} \sum_{k \in K} F\left(\mathbf{W}\overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)\mathbf{W}'\right)$$

which, by the definition of $F(\cdot)$, can be written as:

$$J(\mathbf{W}) = \sum_{k \in K} \left\{ \sum_{i=1}^{m} \frac{1}{2} \log\left(\mathbf{w}_i' \overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)\mathbf{w}_i\right) - \log|\det \mathbf{W}| - \frac{1}{2} \log\left|\det \overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)\right| \right\} \tag{1.20}$$

Moreover, since $\mathbf{W}$ is an orthogonal matrix, (1.20) can be simplified:

$$J(\mathbf{W}) = \sum_{k \in K} \sum_{i=1}^{m} \frac{1}{2} \log\left(\mathbf{w}_i' \overline{\mathbf{\Gamma}}_{\mathbf{z}}(k)\mathbf{w}_i\right) + cte$$

The main drawback of these two approaches is that, under non-Gaussianity, the estimates of the 'dynamic' ICs will be temporally uncorrelated but not statistically independent.

Note that, if $\mathbf{x}_t$ is assumed to be Gaussian distributed, estimating the 'dynamic' ICs by any of previous separation principles (that exploit the temporal structure of the data) is equivalent to estimate the latent factors in the DFM, when $r = m$. Then, it could make sense to think about formulating ICA as a dimensionality reduction technique that extends the DFM under non-Gaussianity assumption.

1.5 Organization of the thesis

After introducing the background of ICA and giving an overview of the problem of dimensionality reduction in multivariate time series, the rest of this thesis is organized as follows.

Chapter 2 proposes a new multivariate conditionally heteroskedastic factor model, GICA-GARCH model, where the observations are assumed to be linearly generated by a set of underlying factors that are independent and evolve according to univariate ARMA-GARCH models. The GICA-GARCH model is an alternative procedure to explain the conditional covariance matrix of large financial data sets using a small number of factors with GARCH effects. The GICA-GARCH works as follows: first, it exploits the information provided by the

unconditional data covariance matrix and estimates the set of conditionally heteroskedastic independent components that explain the co-movements among the observations; second, it assumes factor structure in the conditional distribution of the data and approximates the conditional covariance matrix of large data sets by a linear combination of the conditional variances of the previous common factors. This chapter also analyzes the relationship between the GICA-GARCH model and popular alternatives that impose the factor structure either in the conditional distribution of the data (see Engle (1987) and the family of orthogonal GARCH models such as Alexander (2001), van der Weide (2002), Lanne and Saikkonen (2007), and Fan et al. (2008) among others) or in the unconditional distribution of the data (see Diebold and Nerlove (1989) and Alessi et al. (2006) among others). Some simulation experiments show the ability of ICA to extract the underlying components from financial observations (ICA seems to be an appropriate method to identify the latent factors of the financial data sets, since the conditional distribution of financial data is far away from Gaussianity and they exhibit some kind of non-linear dependence). Finally, Chapter 2 presents an empirical application where the GICA-GARCH model is applied to the Madrid stock market.

In Chapter 3, we present a new algorithm to find the projections of non-Gaussian time series data that will describe 'interesting' features of the observations. Our algorithm, called FOTBI, uses time-delayed fourth-order cumulants to estimate the components using all the available information of the data: the non-Gaussianity as well as the temporal dependencies of the data. Then, it can be as an extension of previous ICA algorithms such as JADE (Cardoso and Souloumiac (1993)), that estimates the components by using the non-Gaussianity of the data but not the autocorrelation structure, and SOBI (Belouchrani et al. (1997)), that neglects the non-Gaussianity and only takes into account the temporal structure to obtain the underlying components. We design three Monte Carlo experiments to analyze the performance of FOTBI to extract non-Gaussian and statistically independent time series components.

Chapter 4 explores the idea of applying ICA to economic multivariate time series data. Our interest is twofold: first, exploring how helpful is ICA to understand the dynamics relationship among the observed time series; second, analyzing the forecasting performance of some ICA procedures at different time horizons. Several simulation experiments analyze the ability of three different algorithms, JADE, SOBI, and FOTBI, to extract a set of underlying components which can be easily interpreted in terms of trend, cycle, and seasonality among others. Moreover, we consider industrial production index (IPI) time series of four European

countries to empirically test how those three ICA algorithms capture the dynamic relationships among the data. In addition, we compare the forecasting performance of each procedure with respect to some benchmark models. FOTBI provides the best results overall the ICA procedures.

Finally, in Chapter 5 we summarize the main conclusions of this thesis and introduce some ideas for future research.

Chapter 2

A conditionally heteroskedastic independent factor model with an application to financial stock returns

In this chapter we propose a new conditionally heteroskedastic factor model, the GICA-GARCH model, which combines independent component analysis (ICA) and multivariate GARCH (MGARCH) models. This model assumes that the data are generated by a set of underlying independent components (ICs) that capture the co-movements among the observations, which are assumed to be conditionally heteroskedastic. The GICA-GARCH model separates the estimation of the ICs from their fitting with a univariate ARMA-GARCH model. Here we will use two ICA approaches to find the ICs: the first one estimates the components maximizing their non-Gaussianity, and the second approach exploits the temporal structure of the data. After estimating and identifying the common ICs, we fit a univariate GARCH model to each of them in order to estimate their univariate conditional variances. Then, the GICA-GARCH model provides a new framework for modelling multivariate conditional heteroskedasticity in which we can explain and forecast the conditional covariances of the observations by modelling univariate conditional variances of a few common ICs. We report some simulation experiments to show the ability of ICA to discover leading factors in a multivariate vector of financial data. Finally, we present an empirical application to the Madrid stock market where we evaluate the forecasting performance of the GICA-GARCH, and two additional factor GARCH models: the orthogonal GARCH and the conditionally uncorrelated components GARCH.

2.1 Introduction

Since Engle (1982) introduced the ARCH model and Bollerslev (1986) generalized it to the GARCH representation the interest in modelling volatilities has grown considerably. In multivariate time series, researches are interested in understanding not only the co-movements of the volatilities of financial assets, but also the co-movements of financial returns. For these purposes a multivariate modelling approach is required. Multivariate GARCH (MGARCH) models should be able to explain the structure of the covariance matrix of a large financial datasets and to represent the dynamics of their conditional variances and covariances too. Depending on the parametrization of the conditional covariance matrix, different specifications for MGARCH have been proposed in the literature (see, for example, the survey of Bauwens et al. (2006)). Two popular MGARCH specifications are the VEC model (Bollerslev et al. (1988)) that is an extension of the univariate GARCH model (see Engle et al. (1984) for an ARCH version), and the BEKK model (Engle and Kroner (1995)) that can be seen as a restricted version of the VEC model. However, in most of these developments the number of parameters to estimate can be very large and the restrictions to guarantee the positive definiteness of the conditional covariance matrix are difficult to implement.

Factor models are an alternative to achieve dimensionality reduction in large datasets. They are based on the idea of the existence of few underlying components that are the driving forces for large datasets. In finance, many empirical applications motivate the use of factor models with conditional heteroskedasticity. For example, asset pricing models usually assume that the dynamics of prices of different assets are explained by a small number of underlying dynamic factors that are conditionally heteroskedastic.

There are two branches of literature about factor GARCH models depending whether the factor structure is referred to the conditional or the unconditional distribution of the data. On the one hand, the FACTOR-ARCH model (Engle (1987)) exploits the conditional distribution of the data applying common factors to model the conditional covariance matrix of the observations. The factors, that follow GARCH-type processes, are given by those linear combinations of the data that summarize the co-movements in their conditional variances. Some applications of the FACTOR-ARCH parametrization are: modelling the term structure of interest rates (Engle et al. (1990); Ng and Engle (1992)), investigating whether international stock markets have the same volatility process (Engle and Susmel (1993)), and modelling

common persistence in conditional variance (Bollerslev and Engle (1993)). Particular models related to the FACTOR-ARCH model are the orthogonal models. They assume that the data conditional covariance matrix is generated by some underlying factors that follow univariate GARCH processes. Examples of this class of models are the orthogonal GARCH (O-GARCH) model (Alexander (2001)), the generalized orthogonal GARCH (GO-GARCH) model (van der Weide (2002)), the generalized orthogonal factor GARCH (GOF-GARCH) model (Lanne and Saikkonen (2007)), and the conditional uncorrelated component GARCH (CUC-GARCH) model (Fan et al. (2008)). Additionally, the full factor GARCH (FF-GARCH) model proposed by Vrontos et al. (2003) and extended by K. and D. (2010), allowing for multivariate t-Student distributions, are also nested in the FACTOR-ARCH approach. On the other hand, the latent factor ARCH model (Diebold and Nerlove (1989)) applies factor structure in the unconditional distribution of the data, and can be seen as a traditional latent factor model where the factors display strong evidence of ARCH structure. In this model, the factors represent the co-movements among the observations, and it is assumed that the commonalities in the volatilities among observations are due to the ARCH effect of such common latent factors. Harvey et al. (1992) extended the Diebold and Nerlove model allowing for general dynamics in the mean and providing a modified version of the Kalman filter for unobserved components models with GARCH disturbances. King et al. (1994), who consider a multifactor model for aggregate stock returns, and Doz and Renault (2004), who present a conditionally heteroskedastic factor model where the common factors represent conditionally orthogonal influences, also extended the Diebold and Nerlove model. The dynamic factor GARCH (DF-GARCH) model (Alessi et al. (2006)) is another example of this branch of the literature. It can be seen as a generalized dynamic factor model where both the dynamic common factors as well as the idiosyncratic components are conditionally heteroskedastic.

In this paper we propose a multivariate conditionally heteroskedastic factor model denoted as GICA-GARCH model. The GICA-GARCH model is a new method for explaining the conditional covariance matrix of large datasets using a small number of factors with GARCH effects. It is based on the intuition that financial markets are driven by a few latent factors that represent the co-movements of financial variables. These factors are estimated by independent component analysis (ICA). ICA can be seen as a factor model (Hyvärinen and Kano (2003)) where the unobserved components are non-Gaussian and mutually independent. Previous researchers, Back and Weigend (1997), Kiviluoto and Oja (1998), Cha and Chan (2000),

and Malaroiu et al. (2000) among others, have applied ICA to financial data. Furthermore, ICA can be considered as a generalization of principal component analysis (PCA) (Hyvärinen et al. (2001)) and seems to be, a priori, more suitable than PCA to explain the non-Gaussian behavior of financial data (Wu and Yu (2005)).

The GICA-GARCH model assumes that observations are generated by a set of underlying factors that are independent and conditionally heteroskedastic. Once the ICs are estimated, they are sorted in terms of total explained variability to choose the few components which represent the co-movements of financial variables. Then, the GICA-GARCH model assumes factor structure in the unconditional distribution of the data. Furthermore, due to the statistical assumption on the ICs, the GICA-GARCH model fits a univariate ARMA-GARCH model to each of them. Then, the conditional covariance matrix of the ICs is allowed to be diagonal. Thus, the GICA-GARCH model transforms the complexity associated to the estimation of a multivariate ARMA-GARCH model into the estimation of a few number of univariate ARMA-GARCH models, and approximates the conditional covariance matrix of the data by the linear combination of a conditional variances of a few ICs. Therefore, the GICA-GARCH model also applies factor structure on the conditional distribution of the data.

This Chapter is organized as follows. Next section describes the ICA model, introduces the three ICA algorithms used to estimate the unobserved components and explains the procedure we have used to sort the ICA components in terms of their explained variability. Furthermore, the relationship between ICA and dynamic factor model (DFM) is analyzed. In Section 2.3 we introduce the GICA-GARCH model to explain and forecast the conditional covariance matrix of a vector of stock returns from the univariate conditional variances of a small number of components. Furthermore, we analyze the relationship between the GICA-GARCH model and other factor GARCH models proposed in the literature. Next, Section 2.4 presents some simulation experiments that illustrate the ability of the GICA-GARCH model to estimate the underlying components of conditionally heteroskedastic data. An empirical application to a real-time dataset is shown in Section 2.5. Finally, Section 2.6 gives some concluding remarks for this chapter.

2.2 The ICA model

In this section we introduce the concept of ICA. First, we present the basic ICA model according to the formal definition given by Comon (1994). Then, we briefly describe the three algorithms we use to estimate the ICA components. As the definition of ICA implies no ordering of the ICs a procedure to weight and sort them is next explained. Finally, we formulate the ICA model as a particular DFM and analyze the relationship between both models.

2.2.1 Definition of ICA

ICA assumes that the observed data are generated by a set of unobserved components that are independent. Let $\mathbf{x}_t = (x_{1t}, x_{2t}, \ldots, x_{mt})'$ be the $m-$dimensional vector of stationary time series, with $E[\mathbf{x}_t] = \mathbf{0}$ and $E[\mathbf{x}_t\mathbf{x}_t'] = \Gamma_{\mathbf{x}}(0)$ positive definite. It is assumed that $\mathbf{x}_t$ is generated by a linear combination of r $(r \leq m)$ latent factors. That is,

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t, \quad t = 1, 2, ..., T \tag{2.1}$$

where $\mathbf{A}$ is an unknown $m \times r$ full rank matrix, with elements a_{ij} that represent the effect of s_{jt} on x_{it}, for $i = 1, 2, ..., m$ and $j = 1, 2, ..., r$, and $\mathbf{s}_t = (s_{1t}, s_{2t}, \ldots, s_{rt})'$ is the vector of unobserved factors, which are called independent components (ICs). It is assumed that $E[\mathbf{s}_t] = \mathbf{0}$, $\mathbf{\Gamma}_{\mathbf{s}}(0) = E[\mathbf{s}_t\mathbf{s}_t'] = \mathbf{I}_r$, and the components of $\mathbf{s}_t$ are statistically independent. Let $(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T)$ be the observed multivariate time series. The problem is to estimate both $\mathbf{A}$ and $\mathbf{s}_t$ only from $(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T)$. That is, ICA looks for an $r \times m$ matrix, $\mathbf{W}$, such that the components given by

$$\widehat{\mathbf{s}}_t = \mathbf{W}\mathbf{x}_t, \quad t = 1, 2, ..., T, \tag{2.2}$$

are as independent as possible. However, previous assumptions are not sufficient to estimate $\mathbf{A}$ and $\mathbf{s}_t$ uniquely, and it is required that no more than one IC is normally distributed. By (3.21), we have:

$$\begin{aligned} \mathbf{\Gamma}_{\mathbf{x}}(0) &= E[\mathbf{x}_t\mathbf{x}_t'] = \mathbf{A}\mathbf{A}', \\ \mathbf{\Gamma}_{\mathbf{x}}(\tau) &= E[\mathbf{x}_t\mathbf{x}_{t-\tau}'] = \mathbf{A}\mathbf{\Gamma}_{\mathbf{s}}(\tau)\mathbf{A}', \quad \tau \geq 1. \end{aligned} \tag{2.3}$$

Therefore, all the dynamic structure of the data comes through the unobserved components, and if they are uncorrelated, then $\mathbf{\Gamma}_{\mathbf{s}}(\tau) = E[\mathbf{s}_t\mathbf{s}_{t-\tau}']$ is a diagonal matrix for all $\tau \geq 1$.

Note that, in spite of previous assumptions, ICA cannot determine either the sign or the order of the ICs. In the following, we focus on the most basic form of ICA, which considers that the number of observed variables is equal to the number of unobserved factors, i.e., $m = r$.

2.2.2 Procedures for estimating the ICs

Both ICA and PCA obtain the latent factors as linear combinations of the data. However their aims are slightly different. On one hand, PCA tries to get uncorrelated factors and, for this purpose, it requires the matrix $\mathbf{W}$ such that $\mathbf{W}\mathbf{W}' = \mathbf{I}$, and the rows of $\mathbf{W}$ are the projection vectors that maximize the variance of the estimated unobserved factors, $\widehat{\mathbf{s}}_t$. On the other hand, ICA tries to get independent factors, and the most often used methods for estimating the ICs impose the restriction that the rows of $\mathbf{W}$ are the directions that maximize the independence of $\widehat{\mathbf{s}}_t$.

Three main ICA algorithms have been proposed: JADE (Cardoso and Souloumiac (1993)) and FastICA (Hyvärinen (1999a); Hyvärinen and Oja (1997)) are based on the non-Gaussianity of the ICs, while SOBI (Belouchrani et al. (1997)) is based on the temporal uncorrelatedness between components. Before any of these algorithms are applied, it is useful to multivariate standardized the data (similarly as we explained in Chapter 1, but considering the eigenvalue decomposition of $\mathbf{\Gamma}_{\mathbf{x}}(0) = E[\mathbf{x}_t\mathbf{x}_t']$). After the standardization, the model (3.21) is written as

$$\mathbf{z}_t = \mathbf{U}\mathbf{s}_t, \tag{2.4}$$

where $\mathbf{U}$ is the new $m \times m$ orthogonal mixing matrix and $\mathbf{z}_t$ is the m-dimensional vector of standardized data.

Joint Approximate Diagonalization of Eigen-matrices: JADE

JADE (Cardoso and Souloumiac (1993)) estimates the ICs maximizing their non-Gaussianity. After whitening the observed data, JADE looks for a matrix $\mathbf{U}'$ such that the components given by

$$\widehat{\mathbf{s}}_t^J = \mathbf{U}'\mathbf{z}_t, \tag{2.5}$$

are maximally non-Gaussian distributed. Note that under the non-Gaussianity assumption, the information provided by the covariance matrix of the data, $\Gamma_{\mathbf{z}}(0) = \mathbf{I}$, is not sufficient to compute (2.5), and higher-order information is needed. Cardoso and Souloumiac (1993) use

cumulants, which are the coefficients of the Taylor series expansion of the logarithm of the characteristic function. In practice, it is enough to take into account fourth-order cumulants, which are defined as

$$\begin{aligned} cum_4(z_{it}, z_{jt}, z_{ht}, z_{lt}) &= E\{z_{it}z_{jt}z_{ht}z_{lt}\} - E\{z_{it}z_{jt}\}E\{z_{ht}z_{lt}\} - \\ &\quad -E\{z_{it}z_{ht}\}E\{z_{jt}z_{lt}\} - E\{z_{it}z_{lt}\}E\{z_{jt}z_{ht}\}, \end{aligned} \tag{2.6}$$

and the fourth-order cumulant tensor associated to $\mathbf{z}_t$ is a $m \times m$ matrix, that is given by

$$[Q_{\mathbf{z}}(\mathbf{Q})]_{ij} = \sum_{k,l=1}^{r} cum_4(z_{it}, z_{jt}, z_{kt}, z_{lt}) q_{kl},$$

where $\mathbf{Q} = (q_{kl})_{k,l=1}^{m}$ is an arbitrary $m \times m$ matrix, and $cum_4(z_{it}, z_{jt}, z_{kt}, z_{lt})$ is like in (2.6). It is easy to see that a set of random vectors are independent if all their cross-cumulants of order higher than two are equal to zero. In particular, $\widehat{\mathbf{s}}_t^J$ will be independent if its associated fourth order cumulant tensor, $Q_{\widehat{\mathbf{s}}_t^J}(\cdot)$, is diagonal. Cardoso and Souloumiac (1993) show that given a set of $m \times m$ matrices, $\Im = \{\mathbf{Q}_1, \cdots, \mathbf{Q}_q\}$, there exists an orthogonal transformation, $\mathbf{V}$, such that the matrices $\{\mathbf{V}'Q_{\mathbf{z}}(\mathbf{Q}_i)\mathbf{V}\}_{\mathbf{Q}_i \in \Im}$ are approximately diagonal. Then we can choose $\mathbf{V} = \mathbf{U}'$, and estimate the latent factors by (2.5). JADE uses an iterative process of Jacobi rotations to solve the joint diagonalization of several fourth-order cumulants matrices. It is a very efficient algorithm in low dimensional problems, but when the dimension increases, it requires high computational cost.

Fast Fixed-Point Algorithm: FastICA

FastICA is a fixed-point algorithm proposed by Hyvärinen and Oja (1997). It estimates

$$\widehat{\mathbf{s}}_t^F = \mathbf{U}'\mathbf{z}_t \tag{2.7}$$

by maximizing their univariate kurtosis. Thus, FastICA searches the directions of projection that maximize the absolute value of the kurtosis of the $\widehat{\mathbf{s}}_t^F$. As kurtosis is very sensitive to outliers, FastICA is not a robust algorithm. Hyvärinen (1999a) proposes a more robust version of FastICA that measures the non-Gaussianity of the ICs by using an approximation of negentropy (previously defined in Chapter 1) instead of the kurtosis coefficient. Therefore, the ICs, (2.7), are estimated as the projections of the data in the directions such that the negentropy of $\widehat{\mathbf{s}}_t^F$ is maximum. The main advantage of FastICA is that it converges in a few number of iterations.

Second-Order Blind Identification: SOBI

Belouchrani et al. (1997) extended the AMUSE algorithm Tong et al. (1990) (see Chapter 1) and proposed the SOBI algorithm. SOBI requires that the ICs, given by

$$\widehat{\mathbf{s}}_t^S = \mathbf{U}'\mathbf{z}_t, \tag{2.8}$$

will be mutually uncorrelated for a set of time lags. That is, SOBI looks for an orthogonal $m \times m$ matrix, $\mathbf{U}'$, that simultaneously diagonalizes a set of $\mathbf{K}$ time delayed covariance matrices of $\widehat{\mathbf{s}}_t^S$,

$$\Gamma_{\mathbf{s}}(\tau) = E\left\{\widehat{\mathbf{s}}_t^S \widehat{\mathbf{s}}_{t-\tau}^{S\prime}\right\}, \quad \tau \in \jmath = \{1, ..., K\}. \tag{2.9}$$

However, the matrix $\Gamma_{\mathbf{s}}(\tau)$, for any $\tau \in \jmath$, is in general not symmetric and, as we showed in Chapter 1, it is usual to take the transformation, $\overline{\mathbf{\Gamma}}_{\mathbf{s}}(\tau) = \frac{1}{2}\{\mathbf{\Gamma}_{\mathbf{s}}(\tau) + \mathbf{\Gamma}_{\mathbf{s}}(\tau)'\}$ that is always a symmetric matrix, $\forall \tau \in \jmath$. SOBI, as JADE and FastICA do, also applies whitening as a preprocessing procedure, and the covariance structure of the whitened data model (3.24) is given by:

$$\mathbf{\Gamma}_{\mathbf{z}}(\tau) = \mathbf{U}\mathbf{\Gamma}_{\mathbf{s}}(\tau)\mathbf{U}', \quad \tau \geq 1, \tag{2.10}$$

where $\mathbf{U}$ is an orthogonal matrix. Therefore,

$$\mathbf{\Gamma}_{\mathbf{s}}(\tau) = \mathbf{U}'\mathbf{\Gamma}_{\mathbf{z}}(\tau)\mathbf{U}, \quad \tau \geq 1. \tag{2.11}$$

Since $\mathbf{U}$ is linear an orthogonal, (2.10) and (2.11) can be rewritten as $\overline{\mathbf{\Gamma}}_{\mathbf{z}}(\tau) = \mathbf{U}\overline{\mathbf{\Gamma}}_{\mathbf{s}}(\tau)\mathbf{U}'$ and $\overline{\mathbf{\Gamma}}_{\mathbf{s}}(\tau) = \mathbf{U}'\overline{\mathbf{\Gamma}}_{\mathbf{z}}(\tau)\mathbf{U}$. Thus, SOBI searches for an orthogonal transformation that will be the joint diagonalizer of the set of time delayed covariance matrices, $\left\{\overline{\mathbf{\Gamma}}_{\mathbf{s}}(\tau_q)\right\}_{\tau_q \in \jmath}$. The optimization problem is therefore to minimize:

$$F(\mathbf{U}) = \sum_{\tau_q \in \jmath} \text{off}\left(\mathbf{U}'\overline{\mathbf{\Gamma}}_{\mathbf{z}}(\tau)\mathbf{U}\right), \tag{2.12}$$

where $\mathbf{U}$ is an orthogonal matrix and 'off' is a measure of the non-diagonality that was defined in the previous chapter. Note that the off $\left(\mathbf{U}'\overline{\mathbf{\Gamma}}_{\mathbf{z}}(\tau)\mathbf{U}\right)$ criterion can also be written as difference between the sum of the squares of all the elements of $\left(\mathbf{U}'\overline{\mathbf{\Gamma}}_{\mathbf{z}}(\tau)\mathbf{U}\right)$ and sum of the squares of its diagonal elements. Then, since $\mathbf{U}$ is an orthogonal matrix, the sum of the squares of all the elements of $\left(\mathbf{U}'\overline{\mathbf{\Gamma}}_{\mathbf{z}}(\tau)\mathbf{U}\right)$ is constant, and the minimization of (2.12) is equivalent to minimize

$$\widetilde{F}(\mathbf{U}) = -\sum_{\tau_q \in \jmath}\sum_{i=1}^{m}\left(\mathbf{u}_i\overline{\mathbf{\Gamma}}_{\mathbf{z}}(\tau)\mathbf{u}_i'\right)^2 \tag{2.13}$$

where $\mathbf{u}_i$ is the i-th row of $\mathbf{U}$. SOBI estimates the ICs using Jacobi rotation techniques. Belouchrani et al. (1997) show that this problem has unique solution: if there exists two different ICs that have different autocovariances for at least one time-lag, then the joint diagonalizer, $\mathbf{U}$, exists and it is unique. That is, if for all $1 \leq i \neq j \leq m$, there is any $q = 1, ..., K$ such that $\gamma_{\mathbf{s}_i}(\tau_q) \neq \gamma_{\mathbf{s}_j}(\tau_q)$, then the components of $\widehat{\mathbf{s}}_t^S$ can be separated, they are unique, and lagged uncorrelated. Note that SOBI cannot get the ICs if they have identical autocovariances for the lags considered.

Weighting the ICs

After estimating the components, we should decide which of them are more important to explain the underlying structure of the data. Note that the PCs are sorted in terms of variability, but the ICs are undetermined with respect to the order. Following Back and Weigend (1997), we will sort the ICs in terms of their explained variability. According to model (3.21), the ith observed variable is given by $x_{it} = \sum_{j=1}^{m} a_{ij} s_{jt}$, and its variance is

$$var(x_{it}) = \sum_{j=1}^{m} a_{ij}^2, \quad i = 1, ..., m. \tag{2.14}$$

For each x_{it}, with $i = 1, ..., m$, Back and Weigend (1997) define the weighted ICs in terms of the elements of the ith row of $\mathbf{A}$ as $\mathbf{s}_t^{w(i)} = diag(a_{i1}, a_{i2}, \ldots, a_{im})\, \mathbf{s}_t$. That is, for each x_{it}, the jth weighted IC is given by $s_{jt}^{w(i)} = a_{ij} s_{jt}$, for $j = 1, ..., m$, and its variance is

$$var\left(s_{jt}^{w(i)}\right) = a_{ij}^2, \quad i, j = 1, ..., m. \tag{2.15}$$

Therefore, from (2.14) and (2.15), the variance of x_{it} which is explained by $s_{jt}^{w(i)}$ is computed as:

$$\nu_j^i = \frac{a_{ij}^2}{\sum_{j=1}^{m} a_{ij}^2}, \quad i, j = 1, ..., m, \tag{2.16}$$

and the total variance of $\mathbf{x}_t$ explained by the jth IC is given by:

$$\vartheta_j = \frac{\sum_{i=1}^{m} \nu_j^i}{\sum_{j=1}^{m} \left(\sum_{i=1}^{m} \nu_j^i\right)}, \quad j = 1, ..., m.$$

Thus, after getting $\{\vartheta_1, \vartheta_2, ..., \vartheta_m\}$, we know how much of total variance is explained by each IC and we can sort them in terms of variability. Thus, the most important ICs will be those that explain the maximum variance of $\mathbf{x}_t$.

2.2.3 ICA and the Dynamic Factor Model

Suppose that, in the basic ICA model, there are r non Gaussian components, $\mathbf{s}_t^{(1)} = (s_{1t}, \ldots, s_{rt})'$ with $r < m$, representing the common dynamic of the time series, but the other $m - r$ components, $\mathbf{s}_t^{(2)} = (s_{r+1t}, \ldots, s_{mt})'$, are Gaussian. Then we can split the matrix $\mathbf{A} = [\mathbf{A}_1 : \mathbf{A}_2]$ accordingly and write

$$\mathbf{x}_t = \mathbf{A}_1 \mathbf{s}_t^{(1)} + \mathbf{A}_2 \mathbf{s}_t^{(2)}. \tag{2.17}$$

Calling $\mathbf{n}_t = \mathbf{A}_2 \mathbf{s}_t^{(2)}$ to the vector of Gaussian noise we have $\mathbf{x}_t = \mathbf{A}_1 \mathbf{s}_t^{(1)} + \mathbf{n}_t$, which is similar to the DFM studied by Peña and Box (1987) and generalized in Peña and Poncela (2006b). However, there are two main differences between these models. First, in the factor model, the r common factors, $\mathbf{s}_t^{(1)}$, are assumed Gaussian and linear, whereas here they are non Gaussian. Second, in the standard factor model the covariance matrix of the noise is of full rank, whereas here it will have rank equal to $m-r$. This last constraint can be relaxed by assuming that the ICA model is contaminated with some Gaussian error model, as in $\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \mathbf{u}_t$, where $\mathbf{u}_t$ is Gaussian. Note that the latent factors of the DFM can be estimated consistently by PCA when both the number of of series and the sample size (m and T respectively) go to infinity (see, for example, Stock and Watson (2002)).

2.3 The GICA-GARCH model

This section presents the GICA-GARCH model as a new multivariate conditionally heteroskedastic factor model. From now on, let $\mathbf{x}_t = (x_{1t}, x_{2t}, \ldots, x_{mt})'$ be the vector of m financial time series. First, we introduce the GICA-GARCH model, give its mathematical formulation, and describe the structure of the ICA components. Then, we explain how this model can be used to forecast the conditional variances of a vector of financial data from the univariate conditional variances of a set of common ICs. Finally, we relate the GICA-GARCH model to the factor GARCH models.

2.3.1 The model

Let us assume that $\mathbf{x}_t$ is a linear combination of a set of independent factors given by (3.21) . Because of series of stock returns are characterized by the presence of clusters of volatility, some of the underlying factors will follow conditionally heteroskedastic processes. In the literature, GARCH models are the most popular specifications for modelling the conditional

variance of the stock returns. Additionally, from empirical finance, it is common to admit that the stock returns could exhibit low order temporal dependencies on the conditional mean, that can be explained by an ARMA model. Therefore, as there could be temporal structure on the conditional mean of the latent factors too, it seems reasonable to propose an ARMA-GARCH specification to model the underlying factor given by (1). Then, we suppose that the vector of unobserved components, $\mathbf{s}_t$, follows an r-dimensional $ARMA(p,q)$ model with $GARCH\,(p',q')$ disturbances:

$$\mathbf{s}_t = \sum_{i=1}^{p} \mathbf{\Phi}_i \mathbf{s}_{t-i} + \sum_{l=0}^{q} \mathbf{\Theta}_l \mathbf{e}_{t-l}, \tag{2.18}$$

where $\mathbf{\Phi}_i = diag\left(\phi_i^{(1)}, ..., \phi_i^{(r)}\right)$ with $|\phi_i^{(j)}| < 1\ \forall j$, $\mathbf{\Theta}_l = diag\left(\theta_l^{(1)}, ..., \theta_l^{(r)}\right)$ with $\mathbf{\Theta}_0 = \mathbf{I}_r$ and $|\theta_l^{(j)}| < 1\ \forall j$, and $\mathbf{e}_t$ is an r-dimensional vector of conditionally heteroskedastic errors given by:

$$\mathbf{e}_t = \mathbf{H}_t^{1/2} \varepsilon_t, \tag{2.19}$$

where $\varepsilon_t \sim iid\,(\mathbf{0}, \mathbf{I}_r)$ and $\mathbf{H}_t^{1/2} = diag(\sqrt{h_{jt}})$ is an $r \times r$ positive definite diagonal matrix such that

$$h_{jt} = \alpha_0^{(j)} + \sum_{i=1}^{p'} \alpha_i^{(j)} e_{jt-i}^2 + \sum_{l=1}^{q'} \beta_l^{(j)} h_{jt-l}, \text{ for } j = 1, .., r, \tag{2.20}$$

where h_{jt} is a stationary process, independent of ε_{jt}, and represents the conditional variance of the jth IC: $h_{jt} = V\,(e_{jt}|\mathbf{I}_{t-1}) = V\,(s_{jt}|\mathbf{I}_{t-1})$, where $\mathbf{I}_{t-1}$ is the past information available until time $t-1$. In order to ensure a positive $h_{jt} > 0$, $\forall j$, it is assumed that $\alpha_0^{(j)} > 0$, $\alpha_i^{(j)} \geq 0$, $\beta_i^{(j)} \geq 0$, and $\sum_{i=1}^{\max(p',q')} \left(\alpha_i^{(j)} + \beta_i^{(j)}\right) < 1$ (see Bollerslev (1986)).

Focusing on forecasting the volatility of the observed financial data, from (3.21), we have that the conditional covariance matrix of $\mathbf{x}_t$ is:

$$\mathbf{\Omega}_t = V\,(\mathbf{x}_t|\mathbf{I}_{t-1}) = \mathbf{A}\mathbf{H_t}\mathbf{A}', \tag{2.21}$$

where $\mathbf{H}_t = diag(h_{1t}, ..., h_{rt})$ is the $r \times r$ conditional covariance matrix of $\mathbf{s}_t$ at time time t. In order to guarantee the diagonality of $\mathbf{H}_t$, we should assume that the conditional correlations of the ICs are zero. This assumption allows us to achieve our purpose: explaining and forecasting the conditional covariances of the observations from the univariate conditional variances of the set of conditionally heteroskedastic components that represents the co-movements of the stock returns. In the GICA-GARCH model, it is assumed that the number of conditionally heteroskedastic common ICs relative to the dimension of the dataset is small. Then, the

GICA-GARCH reduces considerably the number of parameters to be estimated but at the cost of obtaining conditional covariances matrices with a reduced rank. Furthermore, note that the GARCH structure of $\mathbf{x}_t$ is ensured because each IC is generated by an independent GARCH process, and the linear combination of r independent GARCH processes will be a weak GARCH process (see Nijman and Sentana (1996)).

2.3.2 Fitting the model

The model is fitted in two steps. First, we apply ICA to identify the underlying independent components and the loading matrix. Second, univariate GARCH models are fitted to the components. Next, we describe these two steps. Any of the previous ICA algorithms standardizes the data as a preprocessing step, and solves the basic ICA model for the normalized data, which is given by equation (3.24). Thus, JADE, FastICA, and SOBI will estimate the orthogonal loading matrix and the m ICs, defined by equation (2.5) , (2.7) , and (2.8) , respectively. After estimating the model, we should choose the common ICs that we will take into account to forecast the conditional variances of the financial variables. For this purpose, we weight the ICs according to the procedure explained in Section 2.2.4: we sort the ICs in terms of their explained total variability and we split the vector of ICs as $\mathbf{s}_t = [\mathbf{s}_t^{(1)}\mathbf{s}_t^{(2)}]$, where $\mathbf{s}_t^{(1)} = (s_{1t}, \ldots, s_{rt})'$ are the r ICs, with $r < m$, which we choose to represent the co-movements of the data, and $\mathbf{s}_t^{(2)} = (s_{r+1t}, \ldots, s_{mt})'$ are the $m - r$ ICs which we consider as noise. This splitting is done by testing that the $m - r$ ICs are white noise. As an alternative we can fit $ARMA\,(p, q)$ models to $\mathbf{s}_t$ and $\mathbf{s}_t^2$ and check that the order selected with the BIC criteria is in both cases $ARMA(0, 0)$. From now on, we focus on the r selected ICs, that are conditionally heteroskedastic, and fit a univariate $ARMA(p, q) - GARCH(p\prime, q\prime)$ to each one of them. According to the corresponding model, we estimate the univariate conditional variance of each IC and generate the conditional covariance matrix of $\mathbf{s}_t^{(1)}$, $\mathbf{H}_t$. Finally, we get the conditional covariance matrix of the observed data from (2.21) and its ith diagonal term, $\gamma_{it}^2 = \sum_{j=1}^{r} h_{jt}a_{ij}^2$, is the conditional variance of x_{it}, for $i = 1, 2, ..., m$.

Note that the performance of the GICA-GARCH model depends on the method applied to estimate the ICs. In the next sections, we will investigate the usefulness of the three algorithms presented in section 2. Since they use different estimation principles (JADE and FastICA non-Gaussianity, and SOBI dynamic uncorrelatedness) the performance of the algorithms is expected to depend on the features of the data. If the data have excess kurtosis and do not

have significant autocorrelation structure, FastICA and JADE would work better than SOBI. However, for data with large autocorrelation coefficients, SOBI may be the most appropiated algorithm to estimate the ICs.

2.3.3 The GICA-GARCH model and related factor GARCH models

In this section, we point out the relationship between the GICA-GARCH model and other factor GARCH models such as the latent factor ARCH model (Diebold and Nerlove (1989)), the dynamic factor GARCH (DF-GARCH) model (Alessi et al. (2006)), the factor GARCH model (Engle (1987); Engle et al. (1990)), and several orthogonal models.

The GICA-GARCH model assumes that the observations are given by a linear combination of a set of underlying components that are independent and conditionally heteroskedastic. Let's assume that r of these components, $\mathbf{s}_t^{(1)} = (s_{1t}, ..., s_{rt})'$, with $r < m$, explain the co-movements between the observations and the other $m-r$ components, $\mathbf{s}_t^{(2)} = (s_{r+1t}, \ldots, s_{mt})'$, are the noisy ones. Splitting the matrix $\mathbf{A} = [\mathbf{A}_1 : \mathbf{A}_2]$ properly, the GICA-GARCH is given by

$$\mathbf{x}_t = \mathbf{A}_1 \mathbf{s}_t^{(1)} + \mathbf{n}_t, \tag{2.22}$$

where $\mathbf{n}_t = \mathbf{A}_2 \mathbf{s}_t^{(2)}$ is the noise vector. By assumption, both the common and the noisy components are conditionally heteroskedastic and distributed as

$$\begin{pmatrix} \mathbf{s}_t^{(1)} \\ \mathbf{n}_t \end{pmatrix} | \mathbf{I}_{t-1} \quad \sim \quad D \left\{ \begin{pmatrix} \mu_t^{(1)} \\ \mu_t^{(n)} \end{pmatrix}, \begin{pmatrix} \mathbf{H}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{\Gamma}_t \end{pmatrix} \right\}$$

where $\mathbf{H}_t$ is a $r \times r$ conditional covariance matrix of the vector of common factors, and $\mathbf{\Gamma}_t$ is a $m \times m$ conditional covariance matrix of the noise vector with $\text{rank}(\mathbf{\Gamma}_t) = m - r$. Note that the GICA-GARCH model assumes that the vector of common components and the noise vector are conditionally uncorrelated, and allows for the possibility that the common factors and the noise have non-zero conditional mean (the GICA-GARCH model assumes that each IC could fit a univariate ARMA-GARCH model, see (2.18)-(2.20)). Furthermore, due to the independence assumption on the underlying components, both $\mathbf{H}_t$ and $\mathbf{\Gamma}_t$ are diagonal matrices: $\mathbf{H}_t = diag(h_{1t}, ..., h_{rt})$ and $\mathbf{\Gamma}_t = diag(0, ..., 0, h_{r+1t}, ..., h_{mt})$. According to these assumptions, the GICA-GARCH model assumes factor structure in both the unconditional distribution of the data,

$$\mathbf{\Gamma}_{\mathbf{x}}(0) = \mathbf{A}_1 \mathbf{\Gamma}_{\mathbf{s}^{(1)}}(0) \mathbf{A}_1^{'} + \mathbf{\Gamma}_{\mathbf{n}}(0), \tag{2.23}$$

as well as in the conditional distribution,

$$\mathbf{\Omega}_t = \mathbf{A}_1 \mathbf{H}_t \mathbf{A}_1^{'} + \mathbf{\Gamma}_t. \tag{2.24}$$

where $\mathbf{H}_t = diag(h_{1t}, ..., h_{rt})$ and h_{jt} is the conditional variance of the jth component of $\mathbf{s}_t^{(1)}$ given by (2.20).

In practice, the GICA-GARCH model approximates the data conditional covariance matrix as

$$\mathbf{\Omega}_t = \mathbf{A}_1 \mathbf{H}_t \mathbf{A}_1^{'} = \sum_{i=1}^{r} \mathbf{a}_{(1)i} \mathbf{a}_{(1)i}^{'} h_{it}, \tag{2.25}$$

with an accuracy that depends on the number of chosen common components, r, and where $\mathbf{a}_{(1)i} = (a_{1i}, ..., a_{mi})'$. Plugging (2.20) into (2.25), we have:

$$\mathbf{\Omega}_t = \sum_{i=1}^{r} \mathbf{a}_{(1)i} \mathbf{a}_{(1)i}^{'} (\alpha_0^{(i)} + \sum_{l=1}^{p'} \alpha_l^{(i)} e_{it-l}^2 + \sum_{l=1}^{q'} \beta_l^{(i)} h_{it-l}), \text{ for } i = 1, .., r, \tag{2.26}$$

where $e_{it} = s_{it}^{(1)} - \mu_{it}^{(1)}$ for $i = 1, ..., r$. Note that $s_{it}^{(1)} = \mathbf{w}_{(1)i}' \mathbf{x}_t$, where $\mathbf{w}_{(1)i}$ is the ith row vector of $\mathbf{W}_1$ ($\mathbf{W}' = [\mathbf{W}_1 : \mathbf{W}_2]'$ is such that $\mathbf{A}\mathbf{W}' = \mathbf{W}'\mathbf{A} = \mathbf{I}_m$). Then,

$$\mathbf{\Omega}_t = \sum_{i=1}^{r} \mathbf{a}_{(1)i} \mathbf{a}_{(1)i}^{'} (\alpha_0^{(i)} + \sum_{l=1}^{p'} \alpha_l^{(i)} (\mathbf{w}_{(1)i}' \mathbf{x}_{t-l} - \mu_{it-l}^{(1)})^2 + \sum_{l=1}^{q'} \beta_l^{(i)} (\mathbf{w}_{(1)i}' \mathbf{\Omega}_{t-l} \mathbf{w}_{(1)i})), \text{ for } i = 1, .., r, \tag{2.27}$$

and it is clear that the data conditional covariance matrix, estimated by the GICA-GARCH model, is measurable with respect to the information set that contains only past values of the observations.

In the following, we analyze the relationship between the GICA-GARCH model and other factor GARCH models. We distinguish between the two branches of the literature about factor GARCH models depending on whether the factor structure is referred to the unconditional or the conditional distribution of the data.

Factor structure in the unconditional distribution of the data

Here, we analyze the relationship between the GICA-GARCH model and the latent factor GARCH model (Diebold and Nerlove (1989)) and the DF-GARCH model (Alessi et al. (2006)).

First, the GICA-GARCH model can be seen as a latent factor model with GARCH effects (Diebold and Nerlove (1989)). As with the GICA-GARCH model, the latent factor ARCH model (Diebold and Nerlove (1989)) assumes that there are a few common latent factors

(in particular, $r = 1$ in Diebold and Nerlove model) that explain the co-movements among the observations and evolve according to univariate GARCH models ($\mathbf{H}_t = h_{1t}$). However, whereas the GICA-GARCH model assumes factor structure in the unconditional as well as the conditional distribution of the data, the Diebold and Nerlove model only assumes factor structure in the unconditional covariance matrix of the dataset. According to this, the latent factor GARCH model assumes that the commonalities in the volatilities among observations are due to the ARCH effect of the common factor. That is, in Diebold and Nerlove model, the conditional covariance matrix of the observations is given by:

$$\mathbf{\Omega}_t = \mathbf{a}_{(1)1}\mathbf{a}_{(1)1}^{'} h_{1t} + \mathbf{\Gamma}, \tag{2.28}$$

where $\mathbf{\Gamma}$ is a diagonal matrix whose elements correspond to the constant conditional variances of the noisy components. Furthermore, in (2.28), h_{1t} is the conditional variance of the common factor that is not unobservable. Then, $\mathbf{\Omega}_t$ is not measurable when the information set contains only past values of the observations (it should contain past values of the latent factor too).

Second, the GICA-GARCH model can be seen as a parsimonious version the DF-GARCH model (Alessi et al. (2006)). Both models exploit the unconditional information contained in the entire dataset to estimate the conditional covariance matrix of the observations. The main difference between the two models is the parametrization of the common factors conditional covariance matrix. While the GICA-GARCH model, due to the statistical independence of the unobserved components, fits a univariate $ARMA(p,q) - GARCH(p\prime, q\prime)$ to each of them, and assumes that $\mathbf{H}_t$ is diagonal, the DF-GARCH model assumes that the common factors have zero-conditional mean and evolve according to a MGARCH model that is parameterized as a BEKK model:

$$\mathbf{H}_t = \mathbf{C}_0\mathbf{C}_0' + \mathbf{C}_1'\mathbf{s}_{t-1}^{(1)}\mathbf{s}_{t-1}^{(1)\prime}\mathbf{C}_1 + \mathbf{C}_2'\mathbf{H}_{t-1}\mathbf{C}_2, \tag{2.29}$$

where $\mathbf{C}_i$ are matrices of constant parameters. Therefore, whereas in the GICA-GARCH model, the conditional covariance matrix of the dataset depends on the conditional variances of the r common components, the DF-GARCH model estimates the conditional covariance matrix of the observations taking into account both the conditional variances and covariances among the common latent factors. For both models, the GICA-GARCH and the DF-GARCH, the noise components, which in the DF-GARCH model represent the idiosyncratic part, follow univariate ARMA-GARCH models. Then, the conditional covariance matrix, $\mathbf{\Gamma}_t$, is diagonal for both models, but it is a full rank matrix in the DF-GARCH model, whereas in the GICA-GARCH model it will have rank equal to $m - r$.

Factor structure in the conditional distribution of the data

In this section, we analyze the relationship between the GICA-GARCH model, Engle's model and some orthogonal models.

From (2.25)-(2.27), it is clear that the GICA-GARCH model is related to the FACTOR-ARCH model (Engle (1987)). Both models assume that the data conditional covariance matrix is given by a linear combination of the conditional variances of some portfolios (factors) of the observations. Therefore, $\mathbf{\Omega}_t$ is measurable when the information set contains only past values of the observations. Engle's factor GARCH model assumes that $\mathbf{\Gamma}_t$ is a constant matrix that does not play any role in the model.

The GICA-GARCH model is also related to several orthogonal models, such as the O-GARCH (Alexander (2001)), the GO-GARCH (van der Weide (2002)), the GOF-GARCH (Lanne and Saikkonen (2007)), and the CUC-GARCH (Fan et al. (2008)). All these models assume that the data are generated by a linear combination of several factors that follow univariate GARCH models. The GICA-GARCH model can be seen as extension of the O-GARCH model where the estimates of the factors are given by the ICs instead of by the principal components (PCs). Both the GICA-GARCH and the O-GARCH models approximate the data conditional covariance matrix by the univariate conditional variances of few factors (the most risky factors), and transform the problem to estimate a MGARCH model into a small number of univariate volatility models. The cost of reducing dimensionality is that the factors conditional covariance matrices have reduced rank. Some extension of the O-GARCH model is the GO-GARCH model (van der Weide (2002)) that does not reduce dimension and considers $r = m$. A restricted version of the model where only a subset of the underlying factors has a time-varying conditional variance has been analyzed recently by Lanne and Saikkonen (2007). This model, called GOF-GARCH model (Lanne and Saikkonen (2007)), parameterizes the factors conditional covariance matrix as,

$$\mathbf{H}_t = diag(\mathbf{V}_t : \mathbf{I}_{m-r}) \tag{2.30}$$

where $\mathbf{V}_t = diag(v_{1t}, ..., v_{rt})$ is the conditional covariance matrix of the heteroskedastic components. Then, the GOF-GARCH model is similar to the GICA-GARCH model when the noisy components of the GICA-GARCH are homoskedastic ($\mathbf{\Gamma_t} \equiv \mathbf{\Gamma}$ is a constant matrix). Thus, the GOF-GARCH model estimates the data conditional covariance matrix as:

$$\mathbf{\Omega}_t = \mathbf{A}\mathbf{H}_t\mathbf{A}^{'} = \mathbf{A}_1\mathbf{V}_t\mathbf{A}_1^{'} + \mathbf{\Gamma}, \tag{2.31}$$

where $\mathbf{\Gamma} = \mathbf{A}_2\mathbf{A}_2'$. Therefore, the GOF-GARCH model is also related to Engle's model, but assuming that $\mathbf{\Gamma}$ plays a specific role, it is the conditional covariance matrix of the homoskedastic components. Finally, the GICA-GARCH model is related to the work proposed by Fan et al. (2008) that models multivariate volatilities through conditionally uncorrelated components. Both GICA-GARCH and CUC-GARCH models separate the estimation of the unobserved components from fitting a univariate GARCH model for each one of them, and they estimate the components looking for an orthogonal matrix that is the solution of a non-linear optimization problem. However, the GICA-GARCH model requires that the components are statistically independent, while the CUC-GARCH model imposes the weaker assumption of conditional uncorrelatedness.

Table 2.1 summarizes the main features of all the models considered in this section.

2.4 Simulation experiments

In this section we compare the performance of the GICA-GARCH, the O-GARCH, and the CUC-GARCH models. The main difference among the three models relies on the properties assumed for the latent factors: the O-GARCH model assumes unconditionally uncorrelated factors which are estimated by PCA; the CUC-GARCH assumes conditionally uncorrelated components which follow extended GARCH(1,1) models and are estimated by quasimaximum likelihood; and the GICA-GARCH model generalizes the previous models assuming independent underlying factors which are estimated by ICA. Then, it would be interesting to analyze the performance of the three models to identify conditionally heteroskedastic components.

We present three simulation experiments to show the effectiveness of ICA and CUC versus PCA to identify unobserved components that have the main features of financial assets: excess kurtosis and non-Gaussian conditional distributions. In all the experiments we generate six components of 1000 observations and standardize them to have zero mean and unit variance. Then we generate a 6×6 random loading matrix, $\mathbf{A}$, mix the components according to (3.21) , and apply the three procedures, the GICA-GARCH, the O-GARCH, and the CUC-GARCH, to the vector of observations, $\mathbf{x}_t$, and we obtain the ICs, PCs, and CUCs respectively.

In the first experiment, we consider the case where the excess kurtosis of the data comes from different standard ARMA-GARCH specifications and, in addition to Gaussian innovations, we include the Student's t distribution (Bollerslev (1987)), the Laplace distribution

Table 2.1: Taxonomy of some factor GARCH models.

Model	Factor Structure	Factors conditional covariance matrix $\mathbf{H}_t$	Data conditional covariance matrix $\mathbf{\Omega}_t$	Noise conditional covariance matrix $\mathbf{\Gamma}_t$
Latent factor ARCH *(Diebold and Nerlove, 1989)*	$r = 1$ common latent factor, s_{1t}, explains the co-movements of the data	$\mathbf{H}_t = h_{1t}$ $h_{1t} = \alpha_0 + \sum_{l=1}^{p} \alpha_l s_{1t-l}^2$	$\mathbf{\Omega}_t = \mathbf{a}_{(1)1}\mathbf{a}'_{(1)1} h_{1t} + \mathbf{\Gamma}$ not measurable when $I_{t-1} = \{\mathbf{x}_{t-j}\}_{j=1}^{t-1}$	$\mathbf{\Gamma} = diag(\gamma_1, ..., \gamma_m)$ full-rank matrix γ_i constant conditional variance of the noise
DF-GARCH *(Alessi et al., 2006)*	$r < m$ common dynamic latent factors, $\mathbf{s}_t^{(1)}$, are assumed in the UD of the data $\mathbf{s}_t^{(1)}$ are conditionally correlated	$\mathbf{H}_t$ non-diagonal and time-varying evolves according to a MGARCH (BEKK model) $\mathbf{H}_t = \mathbf{C}_0\mathbf{C}_0' + \mathbf{C}_1'\mathbf{s}_{t-1}^{(1)}\mathbf{s}_{t-1}^{(1)\prime}\mathbf{C}_1 + \mathbf{C}_2'\mathbf{H}_{t-1}\mathbf{C}_2$	$\mathbf{\Omega}_t = \mathbf{A}_1\mathbf{H}_t\mathbf{A}_1' + \mathbf{\Gamma}_t$ not measurable when $I_{t-1} = \{\mathbf{x}_{t-j}\}_{j=1}^{t-1}$	$\mathbf{\Gamma}_t = diag(\gamma_{1t}, ..., \gamma_{mt})$ full-rank matrix γ_{it} conditional variance of n_{it} $\mathrm{n}_{it} \sim$ ARMA-GARCH model
Factor ARCH model *(Engle, 1987; Engle et al., 1990)*	$r < m$ common factors ('portfolios'), $\mathbf{s}_t^{(1)} = \mathbf{W}_1'\mathbf{x}_t$, referred to the CD of the data	h_{it} conditional variance of the ith portfolio $h_{it} = \alpha_0^{(i)} + \sum_{l=1}^{p} \alpha_l^{(i)} (\mathbf{w}'_{(1)i}\mathbf{x}_{t-l})^2$	$\mathbf{\Omega}_t = \sum_{i=1}^{r} \mathbf{a}_{(1)i}\mathbf{a}'_{(1)i} h_{it} + \mathbf{\Gamma}$ measurable when $I_{t-1} = \{\mathbf{x}_{t-j}\}_{j=1}^{t-1}$	$\mathbf{\Gamma}$ full-rank positive definite non-specific role
O-GARCH *(Alexander, 2001)*	$r < m$ unconditionally uncorrelated factors, $\mathbf{s}_t^{(1)}$, referred to the CD of the data $\mathbf{s}_t^{(1)}$ are the first r PCs	$\mathbf{H}_t = diag(h_{1t}, ..., h_{rt})$ $h_{it} = \alpha_0^{(i)} + \sum_{l=1}^{p} \alpha_l^{(i)} (\mathbf{w}'_{(1)i}\mathbf{x}_{t-l})^2 + \sum_{l=1}^{q} \beta_l^{(i)} h_{it-l}$	$\mathbf{\Omega}_t = \mathbf{A}_1\mathbf{H}_t\mathbf{A}_1' + \mathbf{\Gamma}_t$ $\mathbf{\Omega}_t \approx \mathbf{A}_1\mathbf{H}_t\mathbf{A}_1'$ measurable when $I_{t-1} = \{\mathbf{x}_{t-j}\}_{j=1}^{t-1}$	$\mathbf{\Gamma}_t = diag(0, ..., 0, h_{r+1t}, ..., h_{mt})$ of rank $m - r$ $\{h_{it}\}_{i=r+1}^{m}$ conditional variances of the $m - r$ discarded PCs
GO-GARCH *(van der Weide, 2002)*	$r = m$ unconditionally uncorrelated factors, $\mathbf{s}_t = \mathbf{W}'\mathbf{x}_t$, explain the CD of the data	$\mathbf{H}_t = diag(h_{1t}, ..., h_{mt})$ $h_{it} = \alpha_0^{(i)} + \alpha_l^{(i)} (\mathbf{w}'_{(1)i}\mathbf{x}_{t-1})^2 + \beta_l^{(i)} h_{it-1}$	$\mathbf{\Omega}_t = \mathbf{A}\mathbf{H}_t\mathbf{A}'$ measurable when $I_{t-1} = \{\mathbf{x}_{t-j}\}_{j=1}^{t-1}$	no noise
GOF-GARCH *(Lanne and Saikkonen, 2007)*	m unconditionally uncorrelated factors, $\mathbf{s}_t$, referred to the CD of the data $r < m$ heteroskedastic factors, $\mathbf{s}_t^{(1)} = \mathbf{W}_1'\mathbf{x}_t$ $m - r$ homoskedastic factors, $\mathbf{s}_t^{(2)} = \mathbf{W}_2'\mathbf{x}_t$	$\mathbf{H}_t = diag(\mathbf{V}_t : \mathbf{I}_{m-r})$ $\mathbf{V}_t = diag(v_{1t}, ..., v_{rt})$ $v_{it} = \alpha_0^{(i)} + \alpha_l^{(i)} (\mathbf{w}'_{(1)i}\mathbf{x}_{t-1})^2 + \beta_l^{(i)} v_{it-1}$	$\mathbf{\Omega}_t = \mathbf{A}\mathbf{H}_t\mathbf{A}'$ $\mathbf{\Omega}_t = \mathbf{A}_1\mathbf{V}_t\mathbf{A}_1' + \mathbf{\Gamma}$ measurable when $I_{t-1} = \{\mathbf{x}_{t-j}\}_{j=1}^{t-1}$	$\mathbf{\Gamma} = \mathbf{A}_2\mathbf{A}_2'$ covariance matrix of the homoskedastic factors
CUC-GARCH *(Fan et al., 2008)*	$m = r$ conditionally uncorrelated factors, $\mathbf{s}_t = \mathbf{W}'\mathbf{x}_t$, explain the CD of the data	$\mathbf{H}_t = diag(h_{1t}, ..., h_{mt})$ $h_{it} = \alpha_0^{(i)} + \alpha_l^{(i)} (\mathbf{w}'_{(1)i}\mathbf{x}_{t-1})^2 + \beta_l^{(i)} h_{it-1}$	$\mathbf{\Omega}_t = \mathbf{A}\mathbf{H}_t\mathbf{A}'$ measurable when $I_{t-1} = \{\mathbf{x}_{t-j}\}_{j=1}^{t-1}$	no noise
GICA-GARCH	m statistically independent components, $\mathbf{s}_t$, explain the co-movements of the data and imply factor structure in the CD of the data $r < m$ are the most risky ICs, $\mathbf{s}_t^{(1)} = \mathbf{W}_1'\mathbf{x}_t$ $m - r$ noisy components, $\mathbf{s}_t^{(2)} = \mathbf{W}_2'\mathbf{x}_t$	$\mathbf{H}_t = diag(h_{1t}, ..., h_{rt})$ $h_{it} = \alpha_0^{(i)} + \sum_{l=1}^{p'} \alpha_l^{(i)} e_{it-l}^2 + \sum_{l=1}^{q'} \beta_l^{(i)} h_{it-l}$ e_{it} residuals of the ARMA model for s_{it} $h_{it} = \alpha_0^{(i)} + \sum_{l=1}^{p'} \alpha_l^{(i)} (\mathbf{w}'_{(1)i}\mathbf{x}_{t-l} - \mu_{it-l}^{(1)})^2 + \beta_l^{(i)} h_{it-l}$	$\mathbf{\Omega}_t = \mathbf{A}_1\mathbf{H}_t\mathbf{A}_1' + \mathbf{\Gamma}_t$ $\mathbf{\Omega}_t \approx \mathbf{A}_1\mathbf{H}_t\mathbf{A}_1'$ measurable when $I_{t-1} = \{\mathbf{x}_{t-j}\}_{j=1}^{t-1}$	$\mathbf{\Gamma}_t = diag(0, ..., 0, h_{r+1t}, ..., h_{mt})$ of rank $m - r$ $\{h_{it}\}_{i=r+1}^{m}$ conditional variances of the $m - r$ discarded ICs

NOTE: UD and CD denotes unconditional and conditional distribution, respectively

(Granger and Ding (1995)), and the generalized error distribution (GED) (Nelson (1991)). The second experiment considers conditionally heteroskedastic factors without temporal dependencies on the conditional mean. In the third experiment, we explore the case where the different excess kurtosis of the latent factors comes from different conditional distributions, and distinguish between two cases: Student's t distribution with different degrees of freedom and GED with different values for the shape parameter.

In order to analyze the performance of the three models we compute the correlation coefficient between each original component and its estimation. Moreover, we compute the mean square error (MSE) between the original and the estimated components as $MSE(s_j, \widehat{s}_j^{(\cdot)}) = 1/T(\sum_{t=1}^{T}\left(s_{jt} - \widehat{s}_{jt}^{(\cdot)}\right)^2)$, for $j = 1, \ldots, r$, where $\widehat{s}_{jt}^{(\cdot)}$ is the jth estimated component by the corresponding method.

Table 2.2: Definition of the original factors in the first experiment

$s_{1t} \sim AR(1) - GARCH(1,1)$	$s_{1t} = 0.0289 + 0.7112 s_{1t-1} + a_{1t}$
	$a_{1t} = \sqrt{h_{1t}}\varepsilon_{1t}$; $h_{1t} = 0.0152 + 0.2080 a_{1t-1}^2 + 0.7918 h_{1t-1}$
$s_{2t} \sim AR(2) - ARCH(1)$	$s_{2t} = 1.2 s_{2t-1} - 0.32 s_{2t-1} + a_{2t}$
	$a_{2t} = \sqrt{h_{2t}}\varepsilon_{2t}$; $h_{2t} = 0.2 + 0.7 a_{2t-1}^2$
$s_{3t} \sim ARMA(1,1) - GARCH(1,1)$	$s_{3t} = 5 + 0.9 s_{3t-1} + a_{3t} - 0.4 a_{3t-1}$
	$a_{3t} = \sqrt{h_{3t}}\varepsilon_{3t}$; $h_{3t} = 0.0079 + 0.0650 a_{3t-1}^2 + 0.9291 h_{3t-1}$
$s_{4t} \sim GARCH(1,3)$	$a_{4t} = \sqrt{h_{4t}}\varepsilon_{4t}$; $h_{4t} = 0.241 a_{4t-1}^2 + 0.077 h_{4t-1} + 0.430 h_{4t-2} + 0.203 h_{4t-3}$
$s_{5t} \sim U(0,1)$	$s_{6t} \sim GED(0, 1, 1.8)$

NOTE: ε_{jt} is a random noise with zero-mean and unit variance, and it is independent of h_{jt}, $\forall j = 1, 2, 3, 4$. We generate four sets of these components by changing the conditional distribution of ε_{jt}: Gaussian, Student's t (t_6), Laplace, and GED ($\kappa = 1.5$).

Table 2.3: Average values for the correlation coefficients and the MSE between the original and the estimated components in the first experiment

	Gaussian		Student's (t_6)		Laplace		GED	
	Correlation	MSE	Correlation	MSE	Correlation	MSE	Correlation	MSE
CUC	0,7903	0,4192	0,7824	0,4349	0,7663	0,4672	0,7360	0,5277
FAST	0,9617	0,0766	0,9634	0,0731	0,9571	0,0858	0,9586	0,0828
JADE	0,9591	0,0817	0,9408	0,1184	0,9158	0,1682	0,9554	0,0892
SOBI	0,8353	0,3292	0,7790	0,4419	0,8403	0,3192	0,8076	0,3846
PCA	0,6646	0,6700	0,7035	0,5925	0,6952	0,6091	0,7087	0,5820

In the first simulation experiment, we generate the components as defined in Table 2.2. Note that the conditional distribution of the ARMA-GARCH components depends on the conditional distribution of ε_{jt} $\forall j = 1, 2, 3, 4$. We consider four possible distributions for the innovations. First, we generate the factors defined in Table 2.2 assuming that ε_{jt} is conditionally Gaussian $\forall j = 1, 2, 3, 4$. We repeat this procedure three more times, assuming that the conditional distribution of ε_{jt}, $\forall j = 1, 2, 3, 4$, is Student's t (t_6), Laplace, and GED ($\kappa = 1.5$). Table 2.3 presents the average results for the correlation coefficients and the MSE between the original and the corresponding estimated components.

According to the results shown in Table 2.3, we see that the average of the correlation coefficients and the MSE take almost identical values along the four conditional distributions we have considered here. Independently of the conditional distribution we take into account, the GICA-GARCH model that estimates the ICs applying FastICA or JADE, provides the most reliable identification of the unobserved ARMA-GARCH components. On the other hand, PCA shows the worst performance for all distributions. SOBI is the ICA algorithms that has worse performance although slightly better than CUC. This is to be expected, as conditionally heteroskedastic components have excess kurtosis and small correlation coefficients.

In the second experiment we generate components which have constant conditional mean but are conditionally heteroskedastic as given in Table 2.4.

Table 2.4: Definition of the original factors in the second experiment

$s_{1t} \sim ARCH(1)$	$s_{1t} = \sqrt{h_{1t}}\varepsilon_{1t}; h_{1t} = 0.2 + 0.7s_{1t-1}^2$
$s_{2t} \sim GARCH(1,1)$	$s_{2t} = \sqrt{h_{2t}}\varepsilon_{2t}; h_{2t} = 0.021 + 0.073s_{2t-1}^2 + 0.906h_{2t-1}$
$s_{3t} \sim GARCH(1,2)$	$s_{3t} = \sqrt{h_{3t}}\varepsilon_{3t}; h_{3t} = 1.692 + 0.245s_{3t-1}^2 + 0.337h_{3t-1} + 0.310h_{3t-2}$
$s_{4t} \sim t_9$	$s_{5t} \sim N(0,1)$ $s_{6t} \sim U(0,1)$

NOTE: ε_{jt} is a random noise with zero-mean and unit variance, and it is independent of h_{jt}, $\forall j = 1,2,3$. We generate four sets of these components by changing the conditional distribution of ε_{jt}: Gaussian, Student's t (t_6), and GED ($\kappa = 1.3$).

As in the first experiment, we generate the factors defined in Table 2.4 assuming that ε_{jt} is conditionally Gaussian $\forall j = 1,2,3$. Then, we repeat the procedure twice, assuming Student's t (t_6) distribution, and the GED ($\kappa = 1.3$) for ε_{jt}, $\forall j = 1,2,3$. We compute the correlation coefficients and the MSEs between each original and the corresponding estimated component. The results (average measures) are shown in Table 2.5, and are very similar to those we got in the first experiment. Note that this result is not surprising and we conclude that imposing the ARMA structure on the conditional mean does not change the results at all.

Table 2.5: Average values for the correlation coefficients and the MSE between the original and the estimated components in the second experiment

	Gaussian		Student's t		GED	
	Correlation	MSE	Correlation	MSE	Correlation	MSE
CUC	0,7870	0,4257	0,8438	0,3123	0,8523	0,2953
FAST	0,9711	0,0578	0,9850	0,0300	0,9847	0,0306
JADE	0,9796	0,0408	0,9852	0,0296	0,9733	0,0533
SOBI	0,9218	0,1563	0,8495	0,3009	0,9392	0,1215
PCA	0,6994	0,6007	0,7037	0,5920	0,6964	0,6066

In the third experiment we analyze the situation in where all the components follow the same ARMA-GARCH specification, and the different excess kurtosis comes from different conditional distributions as defined in Table 2.2. The conditional distribution for ε_{jt}, $j = 1,2,3$

could be: (i) Student's t with different degrees of freedom for each $j = 1, 2, 3$, or (ii) GED with different values for the shape parameter for each $j = 1, 2, 3$. In the case (i) we generate $\varepsilon_{1t} \sim t_6$, $\varepsilon_{2t} \sim t_9$, and $\varepsilon_{3t} \sim t_{11}$. In the second case, (ii), we generate $\varepsilon_{1t} \sim GED(0, 1, 1.5)$, $\varepsilon_{2t} \sim GED(0, 1, 2)$, and $\varepsilon_{3t} \sim GED(0, 1, 1.01)$. The average results obtained for the correlation coefficients and the MSE, are given in Table 2.6. The results show that when excess kurtosis comes from different conditional distributions (or better said, from the same conditional distribution with different values for the parameters) any of the ICA methods performs better than PCA or CUC. If the innovations come from Student's t conditional distribution with different degrees of freedom, PCA and CUC have similar performance. However, if the conditional distribution is the GED with different shape parameters, PCA performs worse than CUC.

Table 2.6: Average values for the correlation coefficients and the MSE between the original and the estimated components in the third experiment

	Student's t		GED	
	Correlation	MSE	Correlation	MSE
CUC	0,7113	0,5771	0,7443	0,5111
FAST	0,8039	0,3920	0,9004	0,1991
JADE	0,8868	0,2263	0,8949	0,2101
SOBI	0,8191	0,3616	0,7926	0,4146
PCA	0,7297	0,5400	0,6705	0,6583

From these simulations, we conclude that the ICA algorithms, specially FastICA and JADE, provide the best performance to identify the unobserved conditionally heteroskedastic factors. The performance of the three ICA algorithms is as expected: as FastICA and JADE look for the independence of the ICs maximizing the non-Gaussianity, they capture better than SOBI the excess kurtosis of the conditionally heteroskedastic components. PCA has the worst performance, so it seems that the orthogonal GARCH models would not be good methods to forecast the conditional variance of large datasets. According to the results, the GICA-GARCH method seems to outperform the CUC- and the O-GARCH. We will investigate this contention in the next section.

2.5 Empirical application

In this section we apply our procedure to a data set of stock returns. First, we describe the data used; second, we explain the procedure to estimate the components; and third, we present the results of applying the GICA-GARCH, the CUC-GARCH, and the O-GARCH,

to forecast the conditional variances of the stock returns.

The data consist of daily closing prices of the 19 assets which were always included in the IBEX 35 from 2000 to 2004 (the 19 stocks are listed in Table 2.12). The IBEX 35 index is the main stock market index of the Madrid stock market. Its composition is revised twice a year and it comprises 35 companies with the largest trading volume of the Madrid stock exchange. We apply some preprocessing steps to the data. First of all, to achieve stationarity, we computed the daily stock returns by taking the first differences of the logarithm of daily closing prices: $\mathbf{r}_t = \log(\mathbf{p}_{t+1}) - log(\mathbf{p}_t),\ t = 1, ..., T = 1250$. Then, $\mathbf{r}_t$ is a 19×1250 multivariate vector of stock returns, whose columns are the value of these 19 stocks in the 1250 trading days in the period 2000-2004. There are some extreme observations that correspond to outliers, which are due to known changes such as stock splits or other legal changes, that have been removed. Finally, we also remove the mean from the stocks returns, and $\mathbf{x}_t = \mathbf{r}_t - \overline{\mathbf{r}}$ is the data that we analyze.

Table 2.7: Summary statistics for the standardized stock returns

	Summary statistics								
	Zero-mean stock returns $\mathbf{x_t}$							$\lvert\mathbf{x_t}\rvert$	$\mathbf{x_t^2}$
Stocks	Median	Maximum	Minimum	St.Dev	Kurtosis	JB	LB(50)	LB(50)	LB(50)
ACS	0,0003	0,0868	-0,0797	0,0182	5,3619	$295,64^*$	$119,23^{**}$	$1053,39^{**}$	$596,25^{**}$
ACX	-0,0004	0,0923	-0,0998	0,0206	4,8558	$182,63^*$	$84,03^{**}$	$718,48^{**}$	$425,03^{**}$
ALT	0,0003	0,0823	-0,0994	0,0189	5,9571	$493,04^*$	$82,19^{**}$	$1188,20^{**}$	$667,89^{**}$
AMS	-0,0001	0,1421	-0,1502	0,0298	5,3303	$282,95^*$	65,20	$985,56^{**}$	$463,07^{**}$
ANA	-0,0003	0,0720	-0,0731	0,0155	5,8949	$436,48^*$	42,75	$678,85^{**}$	$703,10^{**}$
BBVA	0,0002	0,0944	-0,0799	0,0217	4,7493	$165,20^*$	$87,12^{**}$	$2411,12^{**}$	$1871,64^{**}$
BKT	0,0003	0,0900	-0,0906	0,0192	5,9641	$458,73^*$	66,99	$1488,85^{**}$	$843,23^{**}$
ELE	0,0005	0,0831	-0,0747	0,0175	5,4053	$305,54^*$	75,20	$2430,05^{**}$	$1931,99^{**}$
FCC	-0,0005	0,0784	-0,0595	0,0173	5,0625	$245,37^*$	73,40	$1008,10^{**}$	$599,91^{**}$
FER	-0,0009	0,0836	-0,0800	0,0194	4,6196	$138,63^*$	62,73	$972,33^{**}$	$625,17^{**}$
IBE	-0,0001	0,0567	-0,0592	0,0121	5,3139	$282,92^*$	53,60	$655,78^{**}$	$352,19^{**}$
IDR	-0,0002	0,0903	-0,0921	0,0232	4,8257	$177,73^*$	66,29	$892,09^{**}$	$626,85^{**}$
NHH	0,0001	0,0872	-0,0845	0,0182	4,7760	$164,72^*$	64,54	$334,79^{**}$	$191,12^{**}$
POP	0,0000	0,0722	-0,0601	0,0157	4,9358	$199,95^*$	$84,47^{**}$	$786,48^{**}$	$498,84^{**}$
REP	0,0004	0,0879	-0,0814	0,0180	4,9307	$197,25^*$	$77,35^{**}$	$1927,56^{**}$	$1033,91^{**}$
SAN	0,0002	0,0964	-0,1135	0,0233	5,0546	$220,64^*$	67,92	$2524,01^{**}$	$1783,61^{**}$
SGC	0,0004	0,1414	-0,1394	0,0339	4,8686	$189,30^*$	69,27	$1394,44^{**}$	$850,22^{**}$
TEF	0,0002	0,1016	-0,0872	0,0235	4,0998	$72,24^*$	61,42	$1533,22^{**}$	$740,86^{**}$
TPI	0,0004	0,1402	-0,1305	0,0294	5,5625	$342,14^*$	67,48	$1334,54^{**}$	$682,21^{**}$

NOTE: JB denotes the Jarque-Bera test statistic for normality and LB is the Ljung-Box test statistic based on 50 lags for the autocorrelation of the rates of return, the absolute and the squared returns. * indicates that the null hypothesis of normality is rejected at 1% level of significance, while ** indicates that the null of no autocorrelation is rejected at 1% level of significance for the rates of returns, the absolute and the squared returns, respectively.

Table 2.7 presents a summary of the basic statistics of the data. This table includes the Jarque-Bera statistic and the Ljung-Box statistic computed based on 50 lags of the series as well as the absolute values and the squares of the stock returns. The standard deviation of the stock returns, varying from 0.0121 for IBE to 0.0339 for SGC, indicates that there are both, high and low volatile stock returns on our data set. The high values of the kurtosis coefficients (higher than 3 for all the stock returns) confirms the fat-tailed property of the

conditional stock returns distribution. Moreover, the Jarque-Bera test statistics are very high and we clearly reject the null hypothesis of Normality at 1% level of significance. Then, as the conditional distribution of stock returns is far away from Gaussianity, ICA may have the potential to identify the set of latent components that explain the co-movements of the stock returns. According to the Ljung-Box statistics for the stock returns, 13 of the 19 series do not present relevant autocorrelation (the other 6 series have some significant autocorrelation coefficients that can be removed by fitting autoregressive models to these series). For the squares and the absolute values of the stock returns, the high values of Ljung-Box statistics indicate strong autocorrelation in all series, and suggest the presence of non-linear dependence in the stock returns. These are the empirical results we expect when dealing with financial data.

We apply GICA-, CUC-, and O-GARCH model to the vector of zero-mean stock returns, $\mathbf{x}_t$, and we get the corresponding estimates of the 19 unobserved factors. We sort the ICs, the CUCs, and the PCs in terms of the explained total variance. From the results, which are displayed in Table 2.8, we can quantify how much of risk is associated with each component. This fact is crucial since we would like to calculate value at risk of a portfolio of the IBEX 35 index or any other risk management application.

Table 2.8: Sorted components in terms of their explained variability

CUC	$\%_{CUC}$	FAST	$\%_{Fast}$	JADE	$\%_{JADE}$	SOBI	$\%_{SOBI}$	PCA	$\%_{PCA}$
$\widehat{s}^C_{1t}$	18.10	$\widehat{s}^F_{1t}$	17.72	$\widehat{s}^J_{1t}$	11.75	$\widehat{s}^S_{1t}$	11.13	$\widehat{s}^P_{1t}$	35.30
$\widehat{s}^C_{2t}$	16.44	$\widehat{s}^F_{2t}$	10.22	$\widehat{s}^J_{2t}$	7.29	$\widehat{s}^S_{2t}$	9.65	$\widehat{s}^P_{2t}$	7.00
$\widehat{s}^C_{3t}$	8.04	$\widehat{s}^F_{3t}$	6.40	$\widehat{s}^J_{3t}$	6.48	$\widehat{s}^S_{3t}$	9.16	$\widehat{s}^P_{3t}$	5.91
$\widehat{s}^C_{4t}$	5.43	$\widehat{s}^F_{4t}$	5.92	$\widehat{s}^J_{4t}$	6.36	$\widehat{s}^S_{4t}$	8.15	$\widehat{s}^P_{4t}$	4.78
$\widehat{s}^C_{5t}$	5.20	$\widehat{s}^F_{5t}$	5.76	$\widehat{s}^J_{5t}$	6.17	$\widehat{s}^S_{5t}$	7.52	$\widehat{s}^P_{5t}$	4.73
$\widehat{s}^C_{6t}$	4.35	$\widehat{s}^F_{6t}$	4.89	$\widehat{s}^J_{6t}$	5.70	$\widehat{s}^S_{6t}$	5.42	$\widehat{s}^P_{6t}$	4.35
$\widehat{s}^C_{7t}$	4.27	$\widehat{s}^F_{7t}$	4.65	$\widehat{s}^J_{7t}$	5.61	$\widehat{s}^S_{7t}$	5.23	$\widehat{s}^P_{7t}$	4.24
$\widehat{s}^C_{8t}$	3.86	$\widehat{s}^F_{8t}$	4.62	$\widehat{s}^J_{8t}$	5.52	$\widehat{s}^S_{8t}$	4.37	$\widehat{s}^P_{8t}$	4.04
$\widehat{s}^C_{9t}$	3.58	$\widehat{s}^F_{9t}$	4.43	$\widehat{s}^J_{9t}$	5.20	$\widehat{s}^S_{9t}$	4.11	$\widehat{s}^P_{9t}$	3.62
$\widehat{s}^C_{10t}$	3.45	$\widehat{s}^F_{10t}$	4.15	$\widehat{s}^J_{10t}$	5.16	$\widehat{s}^S_{10t}$	4.00	$\widehat{s}^P_{10t}$	3.60
$\widehat{s}^C_{11t}$	3.37	$\widehat{s}^F_{11t}$	3.86	$\widehat{s}^J_{11t}$	5.12	$\widehat{s}^S_{11t}$	3.83	$\widehat{s}^P_{11t}$	3.31
$\widehat{s}^C_{12t}$	3.33	$\widehat{s}^F_{12t}$	3.85	$\widehat{s}^J_{12t}$	4.74	$\widehat{s}^S_{12t}$	3.73	$\widehat{s}^P_{12t}$	3.13
$\widehat{s}^C_{13t}$	3.30	$\widehat{s}^F_{13t}$	3.69	$\widehat{s}^J_{13t}$	4.01	$\widehat{s}^S_{13t}$	3.57	$\widehat{s}^P_{13t}$	3.03
$\widehat{s}^C_{14t}$	3.22	$\widehat{s}^F_{14t}$	3.67	$\widehat{s}^J_{14t}$	3.85	$\widehat{s}^S_{14t}$	3.57	$\widehat{s}^P_{14t}$	2.86
$\widehat{s}^C_{15t}$	3.05	$\widehat{s}^F_{15t}$	3.56	$\widehat{s}^J_{15t}$	3.84	$\widehat{s}^S_{15t}$	3.57	$\widehat{s}^P_{15t}$	2.66
$\widehat{s}^C_{16t}$	3.03	$\widehat{s}^F_{16t}$	3.47	$\widehat{s}^J_{16t}$	3.76	$\widehat{s}^S_{16t}$	3.42	$\widehat{s}^P_{16t}$	2.56
$\widehat{s}^C_{17t}$	2.90	$\widehat{s}^F_{17t}$	3.26	$\widehat{s}^J_{17t}$	3.51	$\widehat{s}^S_{17t}$	3.26	$\widehat{s}^P_{17t}$	2.21
$\widehat{s}^C_{18t}$	2.83	$\widehat{s}^F_{18t}$	2.97	$\widehat{s}^J_{18t}$	3.41	$\widehat{s}^S_{18t}$	3.22	$\widehat{s}^P_{18t}$	1.71
$\widehat{s}^C_{19t}$	2.29	$\widehat{s}^F_{19t}$	2.89	$\widehat{s}^J_{19t}$	2.51	$\widehat{s}^S_{19t}$	3.10	$\widehat{s}^P_{19t}$	0.93
	100.00		100.00		100.00		100.00		100.00

We use Figure 2.1, that shows the explained variability by the components estimated by the five algorithms, to decide the optimal number of components for each method. That is, we choose those components that are the most important sources of risk. The results are given in

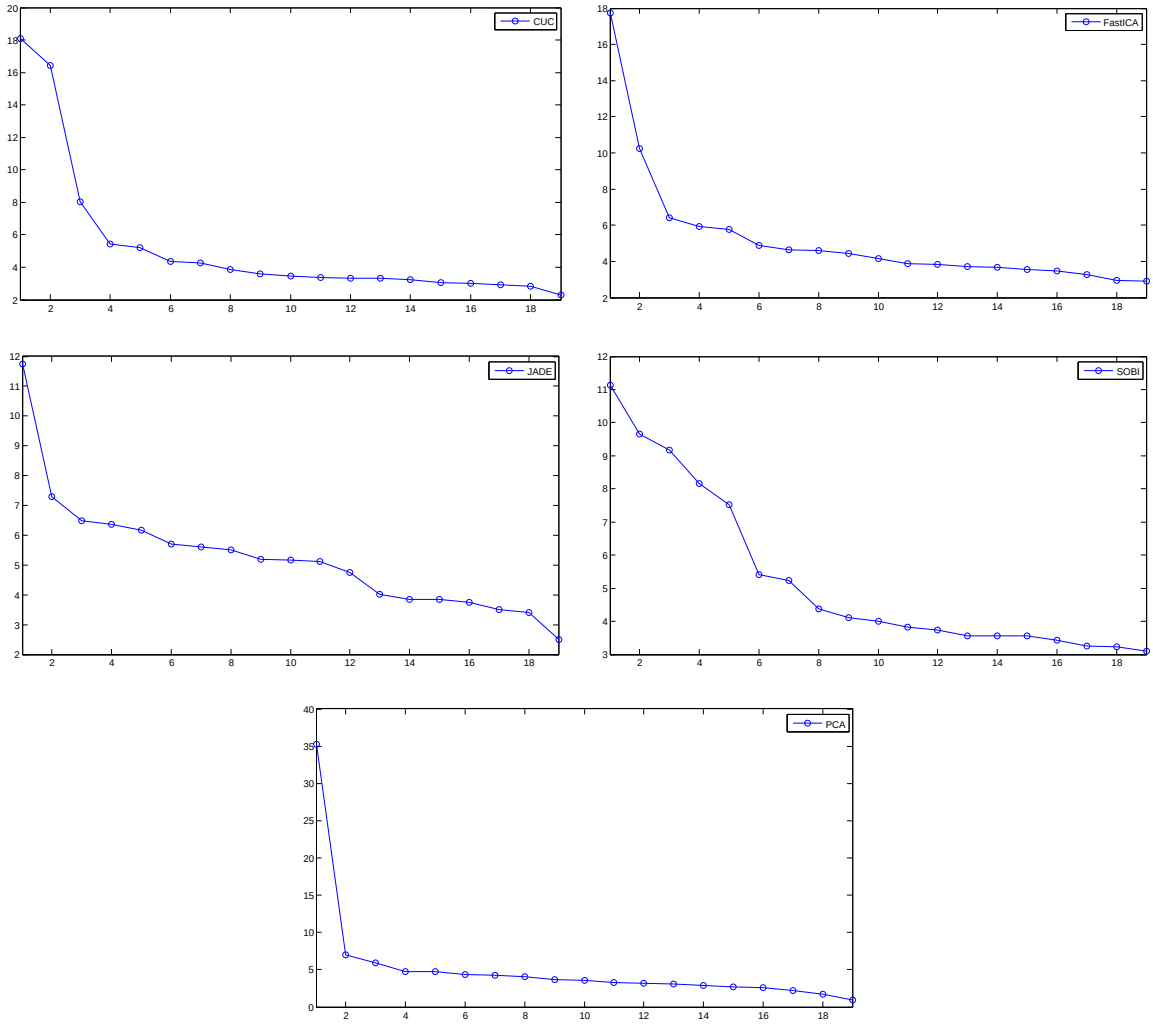


Figure 2.1: Explained total variability by the components

Table 2.9, that also includes the absolute explained variability by the r selected components.

Table 2.9: Number of unobserved components and percentage of total explained variability

	CUC	FAST	JADE	SOBI	PCA
r	4	2	2	5	1
% variability	47.97	27.95	19.04	45.62	35.30

We are interested in investigating which assets are most important to define each component. From (2.2), $\{\widehat{s}_{it}\}_{i=1}^{19}$ can be written as a linear combination of the stock returns, $\widehat{s}_{it} = \sum_{j=1}^{19} w_{ij}x_{jt}$, where w_{ij} represents the effect of the jth stock returns on the ith component, and the largest weights correspond to the most important assets. The ICs, the CUCs, and

the PCs have different interpretations. As an example, we analyze the first components. The first PC is given by a weighted mean of the 19 stock returns and it can be considered an index of the market. Indeed, if we plot the variation of variability of the first PC and the IBEX 35 index, considering groups of ten observations, it is clear that the first PC reflects the main movements of the index IBEX 35 (see Figure 2.2). Then, if we forecast the volatility of $\mathbf{x}_t$ from the volatility of the first PC, the 19 stock returns will tend to move together.

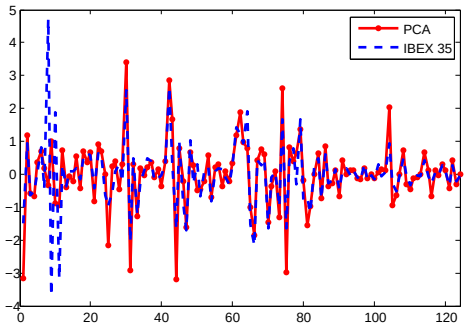


Figure 2.2: Variation of variability of $\widehat{s}_{1t}^{P}$ and the IBEX 35 index

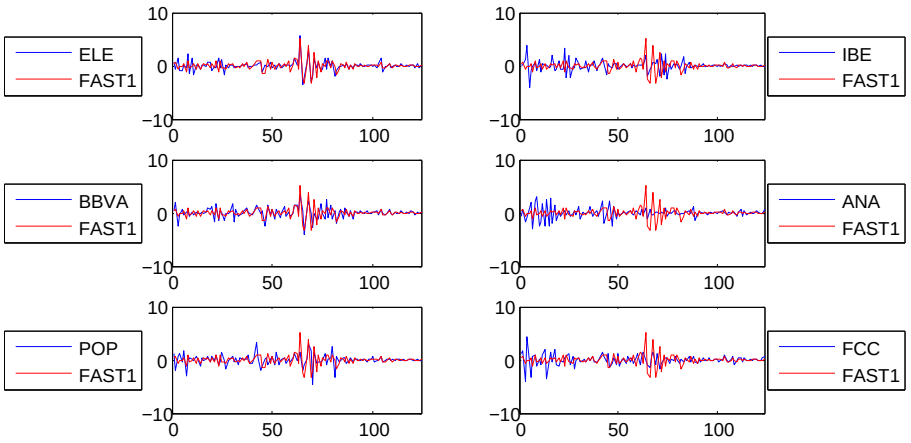


Figure 2.3: Variation of variability of $\widehat{s}_{1t}^{F}$ and the stock returns with the largest weights: a) on the left, the positive ones; b) on the right, the negative ones.

The results for the ICs are different: they cannot be seen as indexes of the market. The first ICs are mainly associated with electricity, building industries, and banking (the sectorial economic classification is detailed in the Appendix), and separate the stock returns in terms of the individual explained variability, $\{\nu_1^i\}_{i=1}^{19}$ (see (2.16)). As an example, we analyze

the first FastICA, $\widehat{s}_{1t}^F$. In Figure 2.3, that shows the variation of variability of $\widehat{s}_{1t}^F$ and the largest weighted assets on $\widehat{s}_{1t}^F$, we see that all assets present a cluster of high variability from observation 600 to 750. The assets which are positively weighted only show this period of higher variability, but the negative ones are also volatile at the beginning of the sample.

The forecasting performance of the GICA-,CUC-, and O-GARCH models is checked as follows:

1. We estimate $\mathbf{A}$ and the unobserved components, by each model, using the whole sample. Then, the components are sorted and r is fixed.

2. Using the whole sample, we fit an $ARMA(p,q)$ with $GARCH(p',q')$ disturbances for each component $\widehat{s}_{jt}$, with $j = 1, ..., r$.

3. The standard ARMA-GARCH processes assume conditionally Gaussian distributions. However, as the stock returns are far away from Gaussianity, the unobserved components should be non-Gaussian too and then, the standard ARMA-GARCH specification may not be adequate to fit the components. In this paper, we explore alternative conditional distributions, and estimate the parameters of the $ARMA(p,q)$-$GARCH(p',q')$ model, with a sample of 1000 observations, using Gaussian, Student's t, and GED as distributional models for innovations. Then, for each model, we generate the one-step-ahead forecast for the univariate conditional variance of each $\widehat{s}_{jt}$,

$$\widehat{h}_{j,1001|1000} = V\left[\widehat{s}_{j1001}|\mathbf{I}_{1000}\right], \quad j = 1, ..., r. \tag{2.32}$$

Thus, by rolling prediction for $t = 1001, ..., 1250$, we have:

$$\widehat{\mathbf{H}}_{t|t-1} = diag(\widehat{h}_{1,t|t-1}, ..., \widehat{h}_{r,t|t-1}), \quad t = 1001, ..., 1250, \tag{2.33}$$

which is the conditional covariance matrix of $\widehat{\mathbf{s}}_t = (\widehat{s}_{1t}, ..., \widehat{s}_{rt})'$ at time t.

4. The conditional variance of $\mathbf{x}_t$ at time t, $\mathbf{\Omega}_t$, is computed by (2.21) . Then, the conditional variance of the ith stock return at time t is given by the ith diagonal term of $\mathbf{\Omega}_t$:

$$\widehat{\gamma}_{i,t|t-1}^2 = \sum_{j=1}^{r} \widehat{h}_{j,t|t-1} a_{ij}^2, \ i = 1, 2, ..., 19, \ t = 1001, ..., 1250. \tag{2.34}$$

From this expression and (2.20) , we see that $\mathbf{x}_t$, which is generated by a linear combination of a set of ICs, possess a GARCH-type structure. This result is confirmed by the work of Nijman and Sentana (1996) in which they show that a linear combination of independent GARCH processes will be a weak GARCH process.

5. To evaluate the forecasting performance of the GICA-,CUC-, and O-GARCH models, we need to compare the predicted volatility and the real one. As population volatility is not observed, the literature proposes to substitute a proxy for the real volatility. Initially, the squares of the stock returns was used as a proxy for the conditional variance (see, for example, Franses and van Dijk (1996)). However, it is shown that the squared returns is a noisy proxy for the conditional variance and it performs very poorly (Andersen and Bollerslev (1998)). Furthermore, Hansen and Lunde (2006) show that an evaluation based on squared returns can induce an inconsistent ranking of volatility models, and may select an inferior model as the 'best' with a probability that goes to one when the sample size increases. To avoid such inconsistency, we follow the Hansen and Lunde's approach and estimate the conditional variance with the realized variance (RV), that is constructed by taking the sum of squared intraday returns (for more details, see Hansen and Lunde (2006)). Assuming that at day t we have f intraday observations of the ith stock return, the RV at time t is defined as:

$$RV_{it} = \sum_{l=1}^{f} x_{i,t,l}^2, \quad i = 1, ..., 19, \ t = 1, ..., T = 1250, \tag{2.35}$$

In our empirical analysis, we artificially construct the intraday stock returns as follows. For a given trading day, t, we use the part of the day that the Madrid stock market is open (9:00-17:30), and generate artificial five-minute returns per day ($f = 102$) by a linear interpolation method. Then, we have $x_{i,t,l}^2$ and we compute RV_{it}, for i=1,...,19, t=1001,...,T=1250 as in (2.35). Once we have computed the RV, we need to define the proxy for the true volatility. Following Hansen and Lunde (2006) we employ three different proxies for the conditional variance: $Proxy1_{it} = \widehat{c}RV_{it}$, where $\widehat{c} = T^{-1}\sum_{t=1}^{T} x_{it}^2/RV_{it}$, $Proxy2_{it} = RV_{it} + (p_t^{open} - p_{t-1}^{close})^2$, and $Proxy3_{it} = x_{it}^2$.

Then, substituting each proxy for the unobserved conditional variance, the one-step-ahead volatility forecast error is given by:

$$\epsilon_{it} = Proxy_{it} - \widehat{\gamma}_{i,t|t-1}^2, \ i = 1, 2, ..., 19, \ t = 1001, ..., 1250. \tag{2.36}$$

6. To evaluate the accuracy of the model we compare the prediction error (2.36) to a benchmark. This benchmark is obtained by predicting the volatility of the stock returns by their marginal variance. Then, we define the relative forecast error by:

$$RE_{it} = \frac{\epsilon_{it}}{\epsilon_{it}^*}, \quad i = 1, 2, ..., 19, \ t = 1001, ..., 1250, \tag{2.37}$$

where ϵ_{it}^{*} is the forecast error of the ith stock return obtained by the benchmark method computed by

$$\epsilon_{it}^{*} = Proxy_{it} - \widehat{\sigma}_{i}^{2},\ i = 1, 2, ..., 19,\ t = 1001, ..., 1250, \tag{2.38}$$

where $\widehat{\sigma}_{i}^{2}$ is the marginal variance of the ith stock return at time t. To minimize the impact of outliers when we analyze the volatility forecasting performance of GICA-, CUC-, and O-GARCH models, we use the Median Relative Absolute Error (MdRAE) criteria (see, for example, Hyndman and Koehler (2006) for a complete revision of measures of forecast accuracy):

$$MdRAE(RE_{it}) = median(|RE_{it}|)$$

In addition, we can use the ratio of the corresponding measure for the ICA and the CUC methods to respect the PCA one:

$$RelMdRAE = \frac{MdRAE_{ICA}}{MdRAE_{PCA}} \tag{2.39}$$

Our purpose is to compare the forecasting performance of the GICA-, CUC-, and O-GARCH models when the latent factors are conditionally Gaussian, Student's t, and GED distributed. We propose to make this comparison following two approaches. In the first approach, we fit a univariate ARMA-GARCH model for each component as we have explained above. In the second approach, even though the CUC-GARCH model assumes that all components follow GARCH(1,1) processes, and it is common to use this specification for modelling stock returns (see, for example, Hansen and Lunde (2005)), we decide to analyze the forecasting performance fitting univariate GARCH(1,1) processes to each IC, CUC, and PC.

The estimates of the parameters when we fit a univariate model to each component are shown in Table 2.13, for the GARCH(1,1) specifications), and Tables 2.14-2.16, for the ARMA-GARCH specifications. From these four tables, we see that the GARCH parameters are significant for both the GARCH(1,1) and the ARMA-GARCH approaches, and for the three conditional distributions. Then, it indicates the time-varying volatility phenomenon of the components. Moreover, from Tables 2.14-2.16, we have that the ARMA parameters are statistically significant too. Thus, it seems that fitting a univariate ARMA model to the conditional mean of the components is reasonable. This result is corroborated by the fact that the values

of the likelihood function for the ARMA-GARCH models are larger than the corresponding ones for the GARCH(1,1) specifications. Moreover, the values of the likelihood under the assumption of conditional Student's t innovations are the largest ones (and the GED distribution outperforms the Gaussian one). Under the Student's t distribution, the degrees of freedom parameter, ν, is very similar for the two approaches. According to the GARCH(1,1) specification as well as the ARMA-GARCH one, the estimates for ν vary from 5.12 to 32.73 indicating heavy tails and excess kurtosis. A similar experience happens with the shape parameter of the GED distribution, which varies from 1.29 to 1.93. According to previous conclusions, the ARMA-GARCH specifications with conditional Student's t innovations seems to provide the most appropriate approach to fitting the underlying conditionally heteroskedastic components.

To evaluate the forecasting performance of the GICA-,CUC-, and O-GARCH models we take into account the two modelling approaches mentioned before. Moreover, in order to analyze the effect of increasing the number of components when we evaluate the forecasting performance of the three models, we vary r from 1 to 5. The average results of the $RelMdRAE$ measured over the 19 stock returns are displayed in Table 2.10 (GARCH(1,1) specifications) and in Table 2.11 (ARMA-GARCH processes). To avoid the choice of the proxy affecting our evaluation, we compute the $RelMdRAE$ criterion using the three proxies proposed by Hansen and Lunde (2006). From Tables 2.10-2.11 we see that, due to the use of relative measures, $RelMdRAE$, the values of the criterion do not differ so much for the different proxies. For both the GARCH(1,1) and the ARMA-GARCH modelling approaches, we obtain robust results and JADE is chosen as the best method to estimate the underlying components, independently of the proxy and the conditional distribution we use.

Tables 2.10-2.11 also show that the values of the $RelMdRAE$ criterion are smaller when we adopt the ARMA-GARCH modelling approach assuming conditional GED innovations. Then, it seems that the GICA-GARCH model where the underlying components are estimated by JADE, and modelled according to univariate ARMA-GARCH models, produces the best forecasting performance. Furthermore, note that independently of the scenario we had, any of the ICA algorithms performs better than CUC and PCA. Therefore, the GICA-GARCH model seems to be a good method to forecast the conditional covariance matrix of large datsets[1].

[1]Evaluating the forecasting performance of the model using the Relative Geometric Mean Relative Absolute Error ($RelGMRAE$) gives similar results which are available from the authors upon request.

Table 2.10: Comparison of overall forecasting performance for the GARCH(1,1) modelling approach. The entries represent the average values of the *RelMdRAE* criterion, measured over the 19 stock returns, when a univariate GARCH(1,1) is fitted to each component

	Gaussian Distribution														
	RelMdRAE (Proxy 1)					RelMdRAE (Proxy 2)					RelMdRAE (Proxy 3)				
	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA
r=1	0,729	0,750	0,754	0,729	1,000	0,732	0,755	0,760	0,734	1,000	0,735	0,750	0,760	0,727	1,000
r=2	0,838	0,794	0,690	0,770	1,000	0,840	0,797	0,696	0,773	1,000	0,832	0,799	0,702	0,759	1,000
r=3	0,964	0,810	0,679	0,781	1,000	0,968	0,814	0,685	0,784	1,000	0,959	0,816	0,683	0,773	1,000
r=4	0,966	0,847	0,660	0,854	1,000	0,967	0,851	0,663	0,857	1,000	0,966	0,849	0,661	0,846	1,000
r=5	1,004	0,863	0,693	0,859	1,000	1,006	0,864	0,694	0,860	1,000	1,007	0,867	0,695	0,856	1,000
	Student's t Distribution														
	RelMdRAE (Proxy 1)					RelMdRAE (Proxy 2)					RelMdRAE (Proxy 3)				
	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA
r=1	0,735	0,766	0,769	0,741	1,000	0,740	0,772	0,777	0,747	1,000	0,743	0,768	0,776	0,741	1,000
r=2	0,831	0,821	0,701	0,768	1,000	0,831	0,823	0,708	0,775	1,000	0,830	0,826	0,712	0,764	1,000
r=3	0,944	0,828	0,683	0,782	1,000	0,946	0,833	0,689	0,785	1,000	0,942	0,837	0,688	0,777	1,000
r=4	0,939	0,861	0,667	0,835	1,000	0,940	0,864	0,670	0,838	1,000	0,935	0,858	0,669	0,830	1,000
r=5	0,992	0,885	0,704	0,845	1,000	0,992	0,886	0,706	0,846	1,000	0,988	0,885	0,708	0,840	1,000
	GED Distribution														
	RelMdRAE (Proxy 1)					RelMdRAE (Proxy 2)					RelMdRAE (Proxy 3)				
	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA
r=1	0,721	0,745	0,752	0,723	1,000	0,724	0,750	0,759	0,728	1,000	0,733	0,751	0,760	0,726	1,000
r=2	0,818	0,795	0,685	0,755	1,000	0,822	0,799	0,693	0,761	1,000	0,820	0,802	0,696	0,752	1,000
r=3	0,938	0,808	0,668	0,771	1,000	0,939	0,809	0,673	0,774	1,000	0,934	0,814	0,675	0,762	1,000
r=4	0,937	0,842	0,654	0,829	1,000	0,937	0,844	0,657	0,833	1,000	0,935	0,840	0,659	0,821	1,000
r=5	0,985	0,863	0,688	0,836	1,000	0,985	0,863	0,690	0,838	1,000	0,983	0,865	0,694	0,833	1,000

Table 2.11: Comparison of overall forecasting performance for the ARMA-GARCH modelling approach. The entries represent the average values of the *RelMdRAE* criterion, measured over the 19 stock returns, when a univariate ARMA-GARCH is fitted to each component

	Gaussian Distribution														
	RelMdRAE (Proxy 1)					RelMdRAE (Proxy 2)					RelMdRAE (Proxy 3)				
	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA
r=1	0,729	0,750	0,754	0,731	1,000	0,732	0,755	0,760	0,735	1,000	0,750	0,750	0,760	0,728	1,000
r=2	0,842	0,799	0,693	0,769	1,000	0,843	0,801	0,698	0,773	1,000	0,799	0,799	0,703	0,757	1,000
r=3	0,968	0,814	0,676	0,780	1,000	0,970	0,817	0,682	0,783	1,000	0,818	0,818	0,677	0,774	1,000
r=4	0,881	0,773	0,596	0,753	1,000	0,883	0,776	0,598	0,758	1,000	0,776	0,777	0,603	0,754	1,000
r=5	0,920	0,792	0,630	0,761	1,000	0,923	0,792	0,631	0,762	1,000	0,792	0,794	0,633	0,758	1,000
	Student's t Distribution														
	RelMdRAE (Proxy 1)					RelMdRAE (Proxy 2)					RelMdRAE (Proxy 3)				
	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA
r=1	0,735	0,766	0,769	0,743	1,000	0,740	0,772	0,777	0,748	1,000	0,743	0,768	0,776	0,743	1,000
r=2	0,831	0,822	0,702	0,767	1,000	0,831	0,824	0,709	0,773	1,000	0,830	0,826	0,713	0,763	1,000
r=3	0,944	0,830	0,685	0,780	1,000	0,946	0,835	0,691	0,785	1,000	0,942	0,838	0,688	0,774	1,000
r=4	0,939	0,786	0,610	0,753	1,000	0,940	0,788	0,613	0,755	1,000	0,935	0,788	0,614	0,752	1,000
r=5	0,992	0,813	0,647	0,764	1,000	0,992	0,815	0,650	0,769	1,000	0,988	0,813	0,651	0,762	1,000
	GED Distribution														
	RelMdRAE (Proxy 1)					RelMdRAE (Proxy 2)					RelMdRAE (Proxy 3)				
	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA	CUC	FAST	JADE	SOBI	PCA
r=1	0,721	0,745	0,752	0,724	1,000	0,724	0,750	0,759	0,729	1,000	0,733	0,751	0,760	0,727	1,000
r=2	0,823	0,799	0,689	0,756	1,000	0,825	0,802	0,696	0,761	1,000	0,822	0,803	0,699	0,751	1,000
r=3	0,942	0,809	0,671	0,770	1,000	0,942	0,811	0,675	0,771	1,000	0,938	0,815	0,677	0,763	1,000
r=4	0,862	0,774	0,603	0,749	1,000	0,862	0,775	0,604	0,752	1,000	0,865	0,775	0,608	0,748	1,000
r=5	0,910	0,797	0,635	0,756	1,000	0,910	0,798	0,637	0,758	1,000	0,906	0,795	0,639	0,754	1,000

2.6 Concluding remarks

We have proposed a new framework for modelling and forecasting large conditional covariance matrices of stock returns using a few underlying factors with conditionally heteroskedasticity. Our model, called GICA-GARCH model, assumes that the co-movements of a vector of financial data are driven by a few independent components which evolve according to univariate ARMA-GARCH models. In the our model, the conditional covariance matrix of the factors is assumed to be diagonal. Therefore the GICA-GARCH provides a parsimonious representation for the conditional covariance matrix of the data, and reduces the number of parameters to be estimated. Our estimation procedure consists of two parts: in the first step, we exploit the unconditional distribution of the data to estimate the ICs, we sort them in terms of variability and disentangle common and idiosyncratic components of the financial data; in the second step, we estimate the conditional covariance matrix of the data as a linear combination of the conditional variances of the common components, that are modelled according to univariate ARMA-GARCH models.

The advantage of the GICA-GARCH model with respect to the existing literature lies in the potentiality of ICA to identify the underlying components of a vector of financial data. In this paper, we have proposed three simulation experiments to test the potential of ICA (using three different algorithms), CUC, and PCA to identify the conditionally heteroskedastic components when they have different excess kurtosis. We have analyzed the performance of the three models both in terms of the correlation coefficients and in terms of the mean square errors between each original component and its estimation. The results show that, regardless of whether the excess kurtosis comes from different GARCH specifications or it comes from different conditional distributions, the ICA methods perform better than CUC and PCA to identify the conditionally heteroskedastic components. Furthermore, the results for the ICA algorithms are as expected: both FastICA and JADE, which estimate the ICs maximizing their non-Gaussianity, capture better than SOBI the excess kurtosis of the conditionally heteroskedastic factors. Therefore, the GICA-GARCH model seems to provide more reliable identification of the unobserved components than the O-GARCH and the CUC-GARCH do.

We have empirically tested the GICA-GARCH model on a vector of stock returns of the Madrid stock market. After applying the three ICA algorithms to identify the unobserved components and fitting a univariate model to each one of them, the empirical results show

that the most appropriate specification to fitting each IC is the ARMA-GARCH model with conditional Student's t innovations. Furthermore, as accurate volatility forecasts are a crucial issue, we have evaluated the forecasting performance of our model. We have implemented a rolling window scheme to compare the relative ability to predict one-step ahead volatility for the GICA-, CUC-, and O-GARCH models. In terms of the average results of the *RelMdRAE*, and independently of the proxy used to substitute the real volatility, our model provides more accurate volatility forecasts than the CUC- and O-GARCH models for the stock returns of the IBEX 35 index. In particular, according to the empirical results, the volatility forecasts obtained using the JADE algorithm are more accurate than those generated by using any other ICA algorithms.

Designing an alternative procedure to sort the ICs and to choose the optimal number of factors may be challenges for the future. Moreover, we are interested in comparing the performance of our model with other multivariate GARCH, such as the dynamic factor GARCH, and extending the GICA-GARCH model for other applications.

2.7 Appendix

Table 2.12: Components of the IBEX 35 from 2000 to 2004 classified by sectors.

Consumption		
Other goods of consumption	ALT	Altadis
Consumption services		
Leisure time / Tourism / Hotel industry	AMS	Amadeus
	NHH	NH Hoteles
Mass media / Publicity	SGC	Sogecable
	TPI	Telefónica Publicidad e Información
Financial Services / Estate Agencies		
Banking	BBVA	Banco Bilbao Vizcaya Argentaria
	BKT	Bankinter
	POP	Banco Popular
	SAN	Banco Santander Central Hispano(*)
Oil and Energy		
Oil	REP	Repsol
Electricity and Gas	ELE	Endesa
	IBE	Iberdrola
Materials / Industry / Building		
Minerals / Metals	ACX	Acerinos
Building	ACS	Grupo ACS
	ANA	Acciona
	FCC	Fomento de Construcciones y Contratas S.A.
	FER	Grupo Ferrovial
Technology / Telecommunications		
Telecommunications and others	TEF	Telefónica
Electronic and Software	TPI	Indra

(*) From 01/01/2000 to 31/10/2001, its name was SCH.

Table 2.13: Estimates for the parameters when a univariate GARCH(1,1) model is fitted to each component. The standard errors are between brackets.

		Estimates GARCH parameters (Gaussian)				Estimates GARCH parameters (Student's t)					Estimates GARCH parameters (GED)				
		α_0	α_1	β_1	L	α_0	α_1	β_1	ν	L	α_0	α_1	β_1	κ	L
CUC	1st	0,01227	0,07732	0,90871	-1285,70	0,00478	0,06068	0,93429	10,89320	-1277,02	0,00761	0,06785	0,92379	1,61184	-1279,61
		(0,00011)	(0,00111)	(0,00169)		(0,00002)	(0,00034)	(0,00045)	(1,25590)		(0,00005)	(0,00058)	(0,00081)	(0,01417)	
	2nd	0,01332	0,13065	0,85452	-1115,61	0,01010	0,11359	0,87552	9,41743	-1106,76	0,01172	0,12276	0,86422	1,51608	-1106,29
		(0,00008)	(0,00207)	(0,00245)		(0,00005)	(0,00158)	(0,00189)	(1,58004)		(0,00007)	(0,00180)	(0,00217)	(0,00996)	
	3rd	0,02650	0,10639	0,86589	-1298,55	0,01442	0,08448	0,90113	8,24801	-1285,39	0,02008	0,09770	0,88168	1,49041	-1286,95
		(0,00053)	(0,00255)	(0,00480)		(0,00026)	(0,00246)	(0,00390)	(1,36669)		(0,00041)	(0,00264)	(0,00463)	(0,00945)	
	4th	0,00533	0,05540	0,93840	-1237,02	0,00535	0,05506	0,93784	13,72002	-1225,95	0,00489	0,05557	0,93874	1,63661	-1230,79
		(0,00001)	(0,00014)	(0,00016)		(0,00001)	(0,00018)	(0,00020)	(1,48350)		(0,00001)	(0,00015)	(0,00016)	(0,03297)	
	5th	0,00680	0,04572	0,94533	-1242,28	0,00544	0,04100	0,95196	8,29869	-1227,34	0,00606	0,04411	0,94794	1,47294	-1229,22
		(0,00005)	(0,00046)	(0,00077)		(0,00006)	(0,00071)	(0,00120)	(3,85762)		(0,00005)	(0,00051)	(0,00085)	(0,01028)	
FAST	1st	0,00524	0,08521	0,91036	-1266,91	0,00643	0,08987	0,90446	18,07498	-1264,81	0,00603	0,08760	0,90700	1,72765	-1264,54
		(-0,00002)	(-0,00057)	(-0,00061)		(0,00002)	(0,00059)	(0,00064)	(0,67388)		(0,00002)	(0,00058)	(0,00063)	(0,01239)	
	2nd	0,01685	0,05415	0,91887	-1153,00	0,02416	0,06498	0,89605	9,14226	-1140,59	0,02094	0,05994	0,90615	1,53997	-1143,29
		(0,00039)	(0,00111)	(0,00393)		(0,00133)	(0,00277)	(0,01202)	(0,63534)		(0,00073)	(0,00174)	(0,00697)	(0,01250)	
	3rd	0,00360	0,04397	0,95215	-1251,49	0,00187	0,02219	0,97463	5,12131	-1212,83	0,00240	0,02994	0,96599	1,30333	-1219,60
		(0,00005)	(0,00118)	(0,00162)		(0,00001)	(0,00030)	(0,00038)	(0,57865)		(0,00003)	(0,00080)	(0,00110)	(0,00741)	
	4th	0,00279	0,04287	0,95329	-1273,14	0,00114	0,03507	0,96295	22,14731	-1271,33	0,00224	0,04022	0,95654	1,84167	-1272,33
		(0,00002)	(0,00051)	(0,00063)		(0,00001)	(0,00029)	(0,00034)	(1,33957)		(0,00001)	(0,00042)	(0,00050)	(0,01534)	
	5th	0,00401	0,04428	0,94883	-1081,87	0,00445	0,04691	0,94566	5,88473	-1050,94	0,00392	0,04389	0,94887	1,33466	-1055,15
		(0,00001)	(0,00043)	(0,00062)		(0,00002)	(0,00068)	(0,00090)	(1,09484)		(0,00001)	(0,00043)	(0,00060)	(0,00798)	
JADE	1st	0,00346	0,07403	0,92268	-1260,39	0,00288	0,07342	0,92418	32,73407	-1259,62	0,00329	0,07367	0,92328	1,92725	-1260,22
		(0,00001)	(0,00042)	(0,00046)		(0,00001)	(0,00042)	(0,00046)	(3,91472)		(0,00001)	(0,00042)	(0,00046)	(0,01528)	
	2nd	0,00136	0,02519	0,97268	-1327,85	0,00118	0,02878	0,96949	11,61326	-1320,81	0,00132	0,02677	0,97118	1,65921	-1323,07
		(0,00001)	(0,00010)	(0,00012)		(0,00001)	(0,00015)	(0,00018)	(3,32136)		(0,00001)	(0,00011)	(0,00014)	(0,01176)	
	3rd	0,00089	0,02107	0,97717	-1245,98	0,00342	0,03006	0,96380	6,88107	-1203,32	0,00273	0,02901	0,96645	1,37756	-1216,31
		(0,00001)	(0,00064)	(0,00084)		(0,00001)	(0,00011)	(0,00016)	(2,41987)		(0,00001)	(0,00023)	(0,00031)	(0,01811)	
	4th	0,00124	0,00791	0,98948	-1184,77	NUM!	0,01036	0,98901	8,83368	-1160,25	0,00017	0,00845	0,99066	1,49322	-1169,71
		(0,00001)	(0,00001)	(0,00003)		(0,00000)	(0,00001)	(0,00001)	(1,57673)		(0,00000)	(0,00001)	(0,00002)	(0,01734)	
	5th	0,00000	0,12415	0,87585	-1257,95	0,00827	0,08109	0,91028	12,65770	-1242,84	0,00912	0,09136	0,89997	1,68202	-1244,06
		(0,00000)	(0,00095)	(0,00098)		(0,00003)	(0,00097)	(0,00108)	(1,61714)		(0,00003)	(0,00091)	(0,00097)	(0,01074)	
SOBI	1st	0,00333	0,04761	0,94871	-1278,83	0,00259	0,04569	0,95154	12,43569	-1273,65	0,00292	0,04726	0,94957	1,61668	-1273,50
		(0,00001)	(0,00021)	(0,00024)		(0,00001)	(0,00018)	(0,00022)	(2,04489)		(0,00001)	(0,00019)	(0,00023)	(0,01247)	
	2nd	0,01170	0,09669	0,89109	-1248,43	0,00292	0,05750	0,93899	10,41371	-1234,26	0,00513	0,06952	0,92511	1,58657	-1240,27
		(0,00035)	(0,00796)	(0,01040)		(0,00001)	(0,00053)	(0,00059)	(1,93100)		(0,00005)	(0,00198)	(0,00231)	(0,01882)	
	3rd	0,01962	0,07834	0,89444	-1209,84	0,01224	0,07779	0,90655	13,00792	-1202,14	0,01676	0,07808	0,89916	1,67812	-1205,51
		(0,00014)	(0,00030)	(0,00068)		(0,00005)	(0,00033)	(0,00050)	(3,15802)		(0,00009)	(0,00031)	(0,00057)	(0,01911)	
	4th	0,00378	0,04269	0,95273	-1241,83	0,00351	0,03986	0,95514	7,43635	-1217,77	0,00314	0,03855	0,95707	1,38849	-1220,54
		(0,00013)	(0,00478)	(0,00629)		(0,00001)	(0,00045)	(0,00058)	(3,14261)		(0,00002)	(0,00074)	(0,00096)	(0,01457)	
	5th	0,00642	0,10249	0,88864	-1057,76	0,00570	0,09197	0,90013	9,35770	-1049,60	0,00639	0,09734	0,89325	1,49605	-1047,64
		(0,00002)	(0,00088)	(0,00097)		(0,00002)	(0,00099)	(0,00110)	(3,81094)		(0,00002)	(0,00091)	(0,00102)	(0,00864)	
PCST	1st	0,01532	0,11528	0,87109	-1281,88	0,00898	0,09332	0,89955	11,26723	-1275,69	0,01232	0,10488	0,88439	1,63892	-1277,09
		(0,00005)	(0,00086)	(0,00083)		(0,00003)	(0,00054)	(0,00054)	(1,22038)		(0,00004)	(0,00067)	(0,00065)	(0,01142)	
	2nd	0,01142	0,09827	0,88501	-1127,17	0,00983	0,08571	0,89938	12,46776	-1122,40	0,01079	0,09356	0,89044	1,60094	-1121,45
		(0,00005)	(0,00105)	(0,00142)		(0,00004)	(0,00089)	(0,00124)	(1,67765)		(0,00005)	(0,00099)	(0,00136)	(0,01073)	
	3rd	0,00249	0,05268	0,94404	-1167,42	0,00174	0,05147	0,94655	12,45832	-1162,00	0,00202	0,05324	0,94440	1,63752	-1162,63
		(0,00001)	(0,00035)	(0,00039)		(0,00001)	(0,00027)	(0,00028)	(1,68052)		(0,00001)	(0,00032)	(0,00034)	(0,01256)	
	4th	0,00498	0,06273	0,93065	-1172,73	0,00757	0,08404	0,90725	9,52976	-1163,11	0,00636	0,07343	0,91853	1,54978	-1164,24
		(0,00004)	(0,00122)	(0,00172)		(0,00005)	(0,00156)	(0,00209)	(0,65577)		(0,00005)	(0,00140)	(0,00197)	(0,01008)	
	5th	0,00756	0,08021	0,91151	-1174,15	0,00241	0,04144	0,95541	7,51165	-1149,85	0,00269	0,03848	0,95715	1,41907	-1156,06
		(0,00015)	(0,00845)	(0,00959)		(0,00001)	(0,00143)	(0,00150)	(0,40320)		(0,00001)	(0,00077)	(0,00086)	(0,01282)	

Table 2.14: Estimates for the parameters when a univariate ARMA-GARCH model, with conditionally Gaussian innovations, is fitted to each component. The standard errors are between brackets.

		Estimates conditional mean (Gaussian)			Estimates GARCH parameters (Gaussian)					
		ϕ_1	ϕ_2	θ_1	α_0	α_1	β_1	α_2	β_2	L
CUC	1st	-	-	-	0,01227	0,07732	0,90871	-	-	-1285,70
					(0,00011)	(0,00111)	(0,00169)			
	2nd	-	-	-	0,01332	0,13065	0,85452	-	-	-1115,61
					(0,00008)	(0,00207)	(0,00245)			
	3rd	-	-	-	0,02650	0,10639	0,86589	-	-	-1298,55
					(0,00053)	(0,00255)	(0,00480)			
	4th	-	-	-	0,00533	0,05540	0,93840	-	-	-1237,02
					(0,00001)	(0,00014)	(0,00016)			
	5th	-	-	-	0,00680	0,04572	0,94533	-	-	-1242,28
					(0,00005)	(0,00046)	(0,00077)			
FAST	1st	-	-	-	0,00524	0,08521	0,91036	-	-	-1266,91
					(-0,00002)	(-0,00057)	(-0,00061)			
	2nd	-0,17807	-0,10131	-	0,01387	0,04792	0,92942	-	-	-1137,32
		(0,03873)	(0,01395)		(0,00014)	(0,00048)	(0,00149)			
	3rd	-	-	-	0,00360	0,04397	0,95215	-	-	-1251,49
					(0,00005)	(0,00118)	(0,00162)			
	4th	-	-	-	0,00421	0,04991	0,02731	0,00000	0,91670	-1272,69
					(0,00006)	(0,00052)	(0,00063)	(0,00000)	(0,00222)	
	5th	-0,22140	-	-	0,00411	0,04469	0,94823	-	-	-1080,46
		(0,01912)			(0,00002)	(0,00043)	(0,00064)			
JADE	1st	-	-	-	0,00346	0,07403	0,92268	-	-	-1260,39
					(0,00001)	(0,00042)	(0,00046)			
	2nd	-	-	-	0,00136	0,02519	0,97268	-	-	-1327,85
					(0,00001)	(0,00010)	(0,00012)			
	3rd	0,26230	-	-0,3133	0,00095	0,02224	0,97604	-	-	-1239,33
		(0,00939)		(0,02586)	(0,00001)	(0,00054)	(0,00070)			
	4th	-	-	-	0,00124	0,00791	0,98948	-	-	-1184,77
					(0,00001)	(0,00001)	(0,00003)			
	5th	-	-	-	0,00000	0,12415	0,87585	-	-	-1257,95
					(0,00000)	(0,00095)	(0,00098)			
SOBI	1st	-	-0,19241	-	0,00340	0,04671	0,94937	-	-	-1275,42
			(0,00558)		(0,00001)	(0,00019)	(0,00024)			
	2nd	0,66035	-	-0,694639071	0,00988	0,08780	0,90194	-	-	-1242,97
		(0,10792)		(0,06644)	(0,00036)	(0,00854)	(0,01137)			
	3rd	-	-	-	0,01962	0,07834	0,89444	-	-	-1209,84
					(0,00014)	(0,00030)	(0,00068)			
	4th	-	-0,25338	-	0,00094	0,02297	0,97499	-	-	-1217,08
			(0,03571)		(0,00001)	(0,00086)	(0,00105)			
	5th	0,13886	-	-	0,00443	0,08276	0,90974	-	-	-1051,31
		(0,01712)			(0,00001)	(0,00064)	(0,00073)			
PCST	1st	-	-	-	0,01532	0,11528	0,87109	-	-	-1281,88
					(0,00005)	(0,00086)	(0,00083)			
	2nd	0,10370	-	-	0,01046	0,09091	0,89308	-	-	-1123,96
		(0,00849			(0,00005)	(0,00141)	(0,00185)			
	3rd	-0,27392	-	-	0,00248	0,05245	0,94428	-	-	-1165,67
		(0,04905			(0,00001)	(0,00035)	(0,00038)			
	4th	-0,50852)	-0,12906	-	0,54059	0,28527	-	-	-	-1235,24
		(0,07742)	(0,01737)		(0,00213)	(0,00425)	( )			
	5th	-	-	-	0,00756	0,08021	0,91151	-	-	-1174,15
					(0,00015)	(0,00845)	(0,00959)			

Table 2.15: Estimates for the parameters when a univariate ARMA-GARCH model, with conditionally Student's t innovations, is fitted to each component. The standard errors are between brackets.

		Estimates conditional mean (Student's t)			Estimates GARCH parameters (Student's t)						
		ϕ_1	ϕ_2	θ_1	α_0	α_1	β_1	α_2	β_2	ν	L
CUC	1st	-	-	-	0,00478	0,06068	0,93429	-	-	10,89320	-1277,02
					(0,00002)	(0,00034)	(0,00045)			(1,25590)	
	2nd	-	-	-	0,01010	0,11359	0,87552	-	-	9,41743	-1106,76
					(0,00005)	(0,00158)	(0,00189)			(1,58004)	
	3rd	-	-	-	0,01442	0,08448	0,90113	-	-	8,24801	-1285,39
					(0,00026)	(0,00246)	(0,00390)			(1,36669)	
	4th	-	-	-	0,00535	0,05506	0,93784	-	-	13,72002	-1225,95
					(0,00001)	(0,00018)	(0,00020)			(1,48350)	
	5th	-	-	-	0,00544	0,04100	0,95196	-	-	8,29869	-1227,34
					(0,00006)	(0,00071)	(0,00120)			(3,85762)	
FAST	1st	-	-	-	0,00643	0,08987	0,90446	-	-	18,07498	-1264,81
					(0,00002)	(0,00059)	(0,00064)			(0,67388)	
	2nd	-0,16450	-0,09279	-	0,01685	0,04987	0,92217	-	-	8,89697	-1124,68
		(0,02138)	(0,03832)		(0,00045)	(0,00105)	(0,00437)			(1,53093)	
	3rd	-	-	-	0,00187	0,02219	0,97463	-	-	5,12131	-1212,83
					(0,00001)	(0,00030)	(0,00038)			(0,57865)	
	4th	-	-	-	0,00146	0,05707	0,00000	0,16346	0,77648	20,89172	-1270,65
					(0,00002)	(0,00064)	(0,00000)	(0,01313)	(0,01492)	(1,31786)	
	5th	-0,20820	-	-	0,00467	0,04830	0,94390	-	-	5,90092	-1049,51
		(0,02066)			(0,00002)	(0,00075)	(0,00101)			(1,10583)	
JADE	1st	-	-	-	0,00288	0,07342	0,92418	-	-	32,73407	-1259,62
					(0,00001)	(0,00042)	(0,00046)			(3,91472)	
	2nd	-	-	-	0,00118	0,02878	0,96949	-	-	11,61326	-1320,81
					(0,00001)	(0,00015)	(0,00018)			(3,32136)	
	3rd	0,24290	-	-0,31926	0,00308	0,02822	0,96609	-	-	6,70582	-1195,76
		(0,01905)		(0,02896)	(0,00001)	(0,00010)	(0,00013)			(2,16043)	
	4th	-	-	-	NUM!	0,01036	0,98901	-	-	8,83368	-1160,25
					(0,00000)	(0,00001)	(0,00001)			(1,57673)	
	5th	-	-	-	0,00827	0,08109	0,91028	-	-	12,65770	-1242,84
					(0,00003)	(0,00097)	(0,00108)			(1,61714)	
SOBI	1st	-	-0,21951	-	0,00272	0,04557	0,95143	-	-	13,29084	-1271,02
			(0,06128)		(0,00001)	(0,00017)	(0,00021)			(2,51943)	
	2nd	0,66536	-	-0,68480	NUM!	0,07030	0,92970	-	-	29,23345	-1235,73
		(0,09307)		(0,05207)	(0,00000)	(0,00123)	(0,00108)			(3,06391)	
	3rd	-	-	-	0,01224	0,07779	0,90655	-	-	13,00792	-1202,14
					(0,00005)	(0,00033)	(0,00050)			(3,15802)	
	4th	-	-0,24597	-	0,00243	0,03053	0,96535	-	-	6,60787	-1189,80
			(0,04729)		(0,00001)	(0,00042)	(0,00057)			(1,90747)	
	5th	0,13256	-	-	0,00436	0,08017	0,91330	-	-	9,64494	-1040,07
		(0,01894)			(0,00001)	(0,00069)	(0,00078)			(2,37447)	
PCST	1st	-	-	-	0,00898	0,09332	0,89955	-	-	11,26723	-1275,69
					(0,00003)	(0,00054)	(0,00054)			(1,22038)	
	2nd	0,11600	-	-	0,00769	0,07146	0,91625	-	-	11,32251	-1116,85
		(0,00906			(0,00003)	(0,00060)	(0,00082)			(1,37506)	
	3rd	-0,27877	-	-	0,00175	0,05163	0,94643	-	-	12,37339	-1160,80
		(0,04874			(0,00001)	(0,00027)	(0,00029)			(1,65436)	
	4th	-0,51948)	-0,12175	-	0,52641	0,36565	-	-	-	5,23611	-1207,89
		(0,07128)	(0,01586)		(0,00223)	(0,00562)	( )			(0,68776)	
	5th	-	-	-	0,00241	0,04144	0,95541	-	-	7,51165	-1149,85

Table 2.16: Estimates for the parameters when a univariate ARMA-GARCH model, with conditionally GED innovations, is fitted to each component. The standard errors are between brackets.

		Estimates conditional mean (GED)			Estimates GARCH parameters (GED)						
		ϕ_1	ϕ_2	θ_1	α_0	α_1	β_1	α_2	β_2	κ	L
CUC	1st	-	-	-	0,00761	0,06785	0,92379	-	-	1,61184	-1279,61
					(0,00005)	(0,00058)	(0,00081)			(0,01417)	
	2nd	-	-	-	0,01172	0,12276	0,86422	-	-	1,51608	-1106,29
					(0,00007)	(0,00180)	(0,00217)			(0,00996)	
	3rd	-	-	-	0,02008	0,09770	0,88168	-	-	1,49041	-1286,95
					(0,00041)	(0,00264)	(0,00463)			(0,00945)	
	4th	-	-	-	0,00489	0,05557	0,93874	-	-	1,63661	-1230,79
					(0,00001)	(0,00015)	(0,00016)			(0,03297)	
	5th	-	-	-	0,00606	0,04411	0,94794	-	-	1,47294	-1229,22
					(0,00005)	(0,00051)	(0,00085)			(0,01028)	
FAST	1st	-	-	-	0,00603	0,08760	0,90700	-	-	1,72765	-1264,54
					(0,00002)	(0,00058)	(0,00063)			(0,01239)	
	2nd	-0,18356	-0,11151	-	0,01575	0,04940	0,92441	-	-	1,53683	-1127,43
		(0,05594)	(0,01395)		(0,00024)	(0,00064)	(0,00235)			(0,01162)	
	3rd	-	-	-	0,00240	0,02994	0,96599	-	-	1,30333	-1219,60
					(0,00003)	(0,00080)	(0,00110)			(0,00741)	
	4th	-	-	-	0,00360	0,06065	0,00602	0,16333	0,76453	1,83612	-1271,89
					(0,00004)	(0,00077)	(0,00065)	(0,01029)	(0,01134)	(0,01527)	
	5th	-0,21700	-	-	0,00410	0,04477	0,94767	-	-	1,33268	-1053,63
		(0,01551)			(0,00001)	(0,00046)	(0,00065)			(0,00807)	
JADE	1st	-	-	-	0,00329	0,07367	0,92328	-	-	1,92725	-1260,22
					(0,00001)	(0,00042)	(0,00046)			(0,01528)	
	2nd	-	-	-	0,00132	0,02677	0,97118	-	-	1,65921	-1323,07
					(0,00001)	(0,00011)	(0,00014)			(0,01176)	
	3rd	0,25026	-	-0,30598	0,00243	0,02743	0,96836	-	-	1,37199	-1208,88
		(0,02849)		(0,02275)	(0,00001)	(0,00017)	(0,00023)			(0,01695)	
	4th	-	-	-	0,00017	0,00845	0,99066	-	-	1,49322	-1169,71
					(0,00000)	(0,00001)	(0,00002)			(0,01734)	
	5th	-	-	-	0,00912	0,09136	0,89997	-	-	1,68202	-1244,06
					(0,00003)	(0,00091)	(0,00097)			(0,01074)	
SOBI	1st	-	-0,20611	-	0,00297	0,04644	0,95022	-	-	1,63470	-1270,72
			(0,06716)		(0,00001)	(0,00018)	(0,00023)			(0,01239)	
	2nd	0,64005	-	-0,68167	0,00459	0,06619	0,92914	-	-	1,60530	-1235,73
		(0,08910)		(0,05644)	(0,00004)	(0,00162)	(0,00188)			(0,02064)	
	3rd	-	-	-	0,01676	0,07808	0,89916	-	-	1,67812	-1205,51
					(0,00009)	(0,00031)	(0,00057)			(0,01911)	
	4th	-	-0,26120	-	0,00191	0,02850	0,96796	-	-	1,38617	-1194,25
			(0,06143)		(0,00002)	(0,00085)	(0,00112)			(0,01050)	
	5th	0,13034	-	-	0,00468	0,08160	0,91062	-	-	1,50754	-1040,56
		(0,01330)			(0,00001)	(0,00065)	(0,00075)			(0,01259)	
PCST	1st	-	-	-	0,01232	0,10488	0,88439	-	-	1,63892	-1277,09
					(0,00004)	(0,00067)	(0,00065)			(0,01142)	
	2nd	0,10278	-	-	0,00902	0,08209	0,90390	-	-	1,56961	-1116,80
		(0,00911			(0,00004)	(0,00098)	(0,00131)			(0,01197)	
	3rd	-0,27098	-	-	0,00201	0,05319	0,94448	-	-	1,63389	-1160,80
		(0,00509			(0,00001)	(0,00033)	(0,00035)			(0,01251)	
	4th	-0,52010)	-0,12053	-	0,51991	0,32528	-	-	-	1,29398	-1207,77
		(0,06942)	(0,01486)		(0,00189)	(0,00445)	( )			(0,00551)	
	5th	-	-	-	0,00269	0,03848	0,95715	-	-	1,41907	-1156,06
					(0,00001)	(0,00077)	(0,00086)			(0,01282)	

Chapter 3

Blind source separation for non-Gaussian time series using high-order statistics

This chapter introduces a new blind source separation approach that exploits both, the non-Gaussianity and the temporal structure, of the dataset. We propose a fourth-order temporal blind identification (FOTBI) algorithm which identifies the set of underlying independent components by the joint diagonalization of several time-delayed fourth-order cumulant matrices. Some Monte Carlo simulation experiments are carried out to investigate the performance of FOTBI. Moreover, the effectiveness of FOTBI is compared to the algorithms presented in Chapter 2. According to our results, FOTBI seems to be a good alternative for the separation of nonlinear time series independent components.

3.1 Introduction

Blind source separation (BSS) consists on identifying a set of underlying factors given only the vector of observations, which is assumed to be generated by a linear combination of those unobserved components. The term 'blind' usually refers to the fact that there is no previous knowledge about either the mixture process or the components. However it is not possible to perform BSS successfully without any a-priori information. In general, the BSS problem is formulated under the assumption of independent components (ICs) and, then it could be perform using independent component analysis (ICA). However, statistical independence could be not enough to guarantee the identifiability of the BSS problem as shown in the previous Chapter. Then it is usually assumed that the ICs satisfy at least one of the following properties: non-Gaussianity, non-stationarity, or having a pronounced autocorrelation structure.

The BSS algorithms proposed in the literature, depending on the specific properties for the ICs, could be categorized in three approaches: algorithms based on the non-Gaussianity of the components, algorithms based on their linear autocorrelations, and algorithms based on non-stationary components.

The first approach assumes that a necessary condition to solve the BSS problem is the non-Gaussianity: all the components, except at most one of them, must be non-Gaussian distributed. Thus, following this approach, BSS is performed by maximizing the non-Gaussianity of the components using higher-order statistics (HOS). As we have seen in previous chapters, the maximization of the non-Gaussianity of the components is one of the ICA estimation principles and then, FastICA and JADE, which were introduced in Chapter 2 as examples of ICA algorithms based on the non-Gaussianity, could be used to perform BSS. In addition to FastICA and JADE, the quasi-JADE algorithm (Bonhomme and Robin (2009)) extends JADE in the presence of noise.However, since none of the previous algorithms take into account the autocorrelation structure of the data, they could have bad BSS performance on time-dependent data sets, because they ignore important information for the separation.

The second category includes BSS algorithms that exploit the temporal structure of the data using second order statistics (SOS), as the AMUSE and the SOBI algorithms presented in Chapter 1 and 2, respectively. They solve the BSS problem under the assumption of mutually uncorrelated components (less restrictive assumption than the independence) that have a pronounced linear autocorrelation structure. The algorithms of this approach estimate the components making their cross-correlations equal to zero. Molgedey and Schuster (1994) was the first one that suggested the simultaneous diagonalization of time-delayed covariance matrices. In addition to AMUSE and SOBI, the TDSEP algorithm (Ziehe and Müller (1998)), which uses the same methodology as SOBI but it is restricted to the noiseless case (there are not essential differences between SOBI and TDSEP and both provide identical results) is a well-known example of this BSS approach. All these algorithms are computationally simple and are allowed to separate Gaussian components. However, as any SOS-based algorithm, they do not guarantee the independence of the non-Gaussian (or non-linear) components which will be only uncorrelated.

The last approach, firstly proposed by Matsuoka et al. (1995), comprises algorithms that allow the separation of both, Gaussian and non-Gaussian components, by exploiting the non-stationarity of the data. Here, the components are assumed to be second-order non-

stationarity, in the sense that their variances are not constant over time. Algorithms of this approach are based on a simple decorrelation technique that performs the BSS by minimizing the sum of several lagged cross-correlation of the squares observations. Some works on non-stationary BSS are given by Choi and Cichocki (2000), Pham and Cardoso (2001), and Hyvärinen (2001) among others.

So far BSS algorithms have been using either information from non-stationarity of the components or from the autocorrelation structure in the data. However, making only one of the three assumptions may lead us to ignore valuable information that could be useful for improving BSS performance. For example, let focus on the case of our interest: we observe a multidimensional non-Gaussian (non-linear) data set that exhibits a significant temporal structure and is second-order stationary. In this example, for identifying the ICs, we should choose between algorithms that exploiting either the temporal structure or the non-Gaussianity of the data. On the one hand, if we use the autocorrelation structure of the observations then, the separation can be based entirely on SOS. However, note that, strictly speaking, the SOS-based algorithms do not estimate ICs. They are merely uncorrelated components, and although the ICs are always uncorrelated, the inverse is not true. The SOS-based algorithms have the capability of separating Gaussian components, and in this situation, the independence of the estimates uncorrelated components is guarantee. However, under non-Gaussianity (or non-linearity) assumption, the SOS-algorithms provide uncorrelated components that are not independent. On the other hand, if we deal with the non-Gaussianity (or non-linearity) of the processes, then HOS are required for the BSS of the ICs. In that case, the statistical independence of the components is achieved but, since HOS-based algorithms do not deal with the temporal structure of the components, again essential information for the separation is lost. Then, it would be desirable to have a BSS procedure that combines both, HOS and temporal structure, to identify the ICs from a vector of non-Gaussian (or non-linear) time series.

In this chapter we present a new BSS approach that jointly exploits the non-Gaussianity (or non-linearity) and the temporal structure of the ICs. Then, it will not be necessary to choose the BSS estimation principle a-priori, any useful information will be ignored, and then, an improvement of the BSS performance is expected. We propose a new fourth-order temporal blind identification (FOTBI) algorithm that allows for the separation of the ICs based on HOS as well as their temporal structure. Since higher-order cumulants contain valuable

information for non-Gaussian components then, FOTBI incorporates the HOS using higher-order cumulants. Moreover, in order to introduce the time structure of the ICs, FOTBI is based on time-delayed higher-order cumulants instead of the instantaneous ones. Although it is well known that cumulants of any higher-order can be used to construct a sufficient criteria for BSS of non-Gaussian components, in practice, it is enough to consider fourth-order cumulants. According to this, FOTBI uses time-delayed fourth-order cumulants. Furthermore, as the cross-cumulants of the ICs vanish, taking into account the temporal structure of the ICs, FOTBI performs BSS by minimizing the time-delayed fourth-order cross-cumulants of the ICs (or equivalently, maximizing their time-delayed fourth-order autocumulants). Note that previous condition is equivalent to say that a set of cumulant matrices are maximally diagonal (Cardoso and Souloumiac (1993)). Then, FOTBI can be defined as a new BSS algorithm that is based on the joint diagonalization of a set of time-delayed fourth-order cumulant matrices.

The rest of the chapter is organized as follows. Section 3.2 briefly summarizes some definitions and fundamental properties of the cumulants, paying special attention to the case of time-delayed cumulants. Moreover, we review the existing approaches for joint diagonalization. In the next section, the BSS model and the relevant assumptions made throughout the chapter are presented. In particular, it is focused on describing the BSS problem for non-Gaussian (or non-linear) and temporally correlated components. Section 3.4 introduces a fourth-order temporal blind identification (FOTBI) method that jointly exploits the non-Gaussianity (or the non-linearity) and the temporal structure of the ICs. In addition, since FOTBI is based on the joint diagonalization of a set of time-delayed fourth-order cumulant matrices, we give the framework to formulate the joint diagonalization problem in our procedure. In Section 3.5 Monte Carlo experimental results show the high performance of FOTBI when the ICs are non-Gaussian (or non-linear) time series. Finally, conclusions are drawn in Section 3.6.

3.2 Preliminaries

In this section, we review some theoretical results that are useful for our blind identification approach. First, we focus on HOS: we review the definition of cumulants of both one- as well as multi-dimensional random variables and stochastic processes, and we include some essential properties for performing BSS. Next, since our BSS approach is based on the simultaneous diagonalization of several time-delayed fourth-order cumulant matrices, we review some joint

diagonalization approaches previously proposed in the literature.

3.2.1 Cumulants: definitions and properties

Here, we introduce the definition of cumulants and point out some important properties which will be useful for our approach. First we focus on cumulants of random variables, and then we move to define cumulants of time-dependent processes.

Let x be a zero-mean random variable with probability density function $p_x(\cdot)$. Let $\varphi_x(\cdot)$ and $\phi_x(\cdot)$ be, respectively, the first and the second characteristic functions of x, given by:

$$\begin{aligned} \varphi_x(\xi) = E\{\exp(i\xi x)\} = \textstyle\int_{\Re} \exp(i\xi x) dF(x) \ , \ \xi \in \Re \\ \phi_x(\xi) = \ln(\varphi_x(\xi)) \ , \ \xi \in \Re \end{aligned} \tag{3.1}$$

where $F(x)$ is the distribution function of x. The pth-order cumulant of x, denoted by $cum_{p,x}(x)$, is defined as the pth-order coefficient of the Taylor series expansion of $\phi_x(\cdot)$ about the origin. That is:

$$\phi_x(\xi) = 1 + \textstyle\sum_{p=1}^{\infty} cum_{p,x}(x) \frac{(i\xi)^p}{p!}, \ \xi \in \Re,$$

where $cum_{p,x}(x)$ is:

$$cum_{p,x}(x) = (-i)^p \frac{\delta^p \phi_x(\xi)}{\delta \xi^p}|_{\xi=0}, \ \xi \in \Re.$$

Cumulants and moments are very related: they are defined in a similar way (the pth-order moment of x is the pth-order coefficient of the Taylor series expansion of $\varphi_x(\cdot)$), and cumulants can be expressed in terms of moments (in fact, they are equal in some particular cases). Thus, for example, the first-, second-, and third-order cumulants of x, given by $cum_{1,x}(x) = E\{x\} = 0$, $cum_{2,x}(x) = E\{x^2\}$, $cum_{3,x}(x) = E\{x^3\}$, correspond to the respective first-, second-, and third-order moments of x. Moreover, the fourth-order cumulant of x is equal to its kurtosis coefficient, $cum_{4,x}(x) = E\{x^4\} - 3E\{x^2\}^2$, that is defined in terms of the second- and the fourth-order moments of x. Then, it is clear that both, cumulants and moments, give us the same statistical information.

Moving to the multivariate case, let $\mathbf{y} = (y_1, ..., y_m)'$ be an mth-dimensional vector of zero-mean random variables with probability density function $p_{\mathbf{y}}(\cdot)$. The first and the second characteristic functions of $\mathbf{y}$, $\varphi_{\mathbf{y}}(\cdot)$ and $\phi_{\mathbf{y}}(\cdot)$, are given straightforward from (3.1) as follows: $\varphi_{\mathbf{y}}(\mathbf{u}) = E\{\exp(i\mathbf{u}'\mathbf{y})\} = \int_{\Re^m} \exp(i\mathbf{u}'\mathbf{y}) dF(\mathbf{y})$ and $\phi_{\mathbf{y}}(\mathbf{u}) = log(\varphi_{\mathbf{y}}(\mathbf{u}))$, where $F(\mathbf{y})$ is the

distribution function of $\mathbf{y}$ and $\mathbf{u} = (u_1, \ldots, u_m)' \in \Re^m$. The pth-order cumulant of $\mathbf{y}$, denoted by $cum_{p,\mathbf{y}}(\overbrace{\mathbf{y}, \ldots, \mathbf{y}}^{p-times}) \equiv cum_{p,\mathbf{y}}(\mathbf{y})$, is defined as an m^p-dimensional vector that contains the pth-order coefficients of the Taylor series expansion of $\phi_{\mathbf{y}}(\cdot)$ about the origin. That is,

$$\phi_{\mathbf{y}}(\mathbf{u}) = 1 + \sum_{p=1}^{\infty} \frac{1}{p!} cum_{p,\mathbf{y}}(\mathbf{y}) \overbrace{(i\mathbf{u}) \otimes \ldots \otimes (i\mathbf{u})}^{p-times}, \tag{3.2}$$

where $\otimes$ is the Kronecker product and $cum_{p,\mathbf{y}}(\mathbf{y})$ is:

$$cum_{p,\mathbf{y}}(\mathbf{y}) = (-i)^p \overbrace{\left(\frac{\delta \phi_{\mathbf{y}}(\mathbf{u})}{\delta \mathbf{u}}\right) \otimes \ldots \otimes \left(\frac{\delta \phi_{\mathbf{y}}(\mathbf{u})}{\delta \mathbf{u}}\right)}^{p-times} |_{\mathbf{u}=\mathbf{0}}. \tag{3.3}$$

Therefore, $cum_{p,\mathbf{y}}(\mathbf{y})$ is an $m^p \times 1$ vector with $[(i_1 - 1)m^{p-1} + (i_2 - 1)m^{p-2} + \ldots + (i_{p-1} - 1)m + i_p]$th element given by

$$cum_{p,\mathbf{y}}(y_{i_1}, \ldots, y_{i_p}) = (-i)^p \frac{\delta^p \phi_{\mathbf{y}}(\mathbf{u})}{\delta u_{i_1} \ldots \delta u_{i_p}},$$

where $i_\nu \in \{1, \ldots, m\}$ for $\nu = 1, \ldots, p$, is a set of p indexes.

The following are examples of some pth-order cumulants:

- if $p = 1$, the first-order cumulant of $\mathbf{y}$, $cum_{1,\mathbf{y}}(\mathbf{y})$, is an $m \times 1$ vector whose ith element is: $cum_{1,\mathbf{y}}(y_i) = E\{y_i\} = 0$, $\forall i = 1, \ldots, m$. Then, $cum_{1,\mathbf{y}}(\mathbf{y}) = E\{\mathbf{y}\} = \mathbf{0}$.

- if $p = 2$, the second-order cumulant of $\mathbf{y}$, $cum_{2,\mathbf{y}}(\mathbf{y})$, is an $m^2 \times 1$ vector whose $[(i - 1)m + j]$th element is: $cum_{2,\mathbf{y}}(y_i, y_j) = E\{y_i y_j\}$, $\forall i, j = 1, \ldots, m$. Then, $cum_{2,\mathbf{y}}(\mathbf{y}) = vec(E\{\mathbf{y}\mathbf{y}'\})$, i.e., it is the vectorization of the covariance matrix of $\mathbf{y}$.

- if $p = 3$, the third-order cumulant of $\mathbf{y}$, $cum_{3,\mathbf{y}}(\mathbf{y})$, is an $m^3 \times 1$ vector whose $](i - 1)m^2 + (j - 1)m + k]$th element is: $cum_{3,\mathbf{y}}(y_i, y_j, y_k) = E\{y_i y_j y_k\}$.

- if $p = 4$, the fourth-order cumulant of $\mathbf{y}$, $cum_{4,\mathbf{y}}(\mathbf{y})$, is an $m^4 \times 1$ vector whose $[(i-1)m^3 + (j-1)m^2 + (k-1)m + l]$th element is:
  $cum_{4,\mathbf{y}}(y_i, y_j, y_k, y_l) = E\{y_i y_j y_k y_l\} - E\{y_i y_j\}E\{y_k y_l\} - E\{y_i y_k\}E\{y_j y_l\} - E\{y_i y_l\}E\{y_j y_k\}$.

Here we have that, as in the univariate case, the cumulants of a vector of random variables can be written in terms of moments. For example, the first- and the second-order cumulant of $\mathbf{y}$ are, respectively, the mean and the vectorization of the covariance matrix of $\mathbf{y}$. Moreover, the

third-order cumulants are equal to the third order moments, and the fourth-order cumulants differ from the fourth-order moments in some second-order moments.

Despite of higher-order cumulants and higher-order moments provide the same statistical information, there are some mathematical and practical reasons for which higher-order cumulants are preferable to perform BSS. In the following, we point out some examples of these properties (for a complete revision of cumulants see, for example, Mendel (1991)):

Property P1. *Invariance and equivariance.* If $\mathbf{u} = (u_1, \ldots, u_m)' \in \Re^m$ is a vector of constants, the first-order cumulant of $\mathbf{y}$ is shift-equivariant:

$$cum_{1,\mathbf{y}}(\mathbf{u} + \mathbf{y}) = \mathbf{u} + cum_{1,\mathbf{y}}(\mathbf{y}), \text{ i.e.,}$$
$$cum_{1,\mathbf{y}}(u_i + y_i) = u_i + cum_{1,\mathbf{y}}(y_i), \forall i = 1, \ldots, m$$

and all of the others are shift-invariant:

$$cum_{p,\mathbf{y}}(\mathbf{u} + \mathbf{y}, \overbrace{\mathbf{y}, \ldots, \mathbf{y}}^{(p-1)-times}) = cum_{p,\mathbf{y}}(\overbrace{\mathbf{y}, \ldots, \mathbf{y}}^{p-times}), \text{ i.e.,}$$
$$cum_{p,\mathbf{y}}(u_{i_1} + y_{i_1}, \ldots, y_{i_p}) = cum_{p,\mathbf{y}}(y_{i_1}, \ldots, y_{i_p}) \ , \ \forall p > 1,$$

where $i_\nu \in \{1, \ldots, m\}$ for $\nu = 1, \ldots, p$, is a set of p indexes.

Property P2. *Gaussian rejection.* If $\mathbf{y}$ is a vector of Gaussian random variables,

$$cum_{p,\mathbf{y}}(\overbrace{\mathbf{y}, \ldots, \mathbf{y}}^{p-times}) = \mathbf{0}, \forall p \geq 3, \text{ i.e.,}$$
$$cum_{p,\mathbf{y}}(y_{i_1}, \ldots, y_{i_p}) = 0 \ , \ \forall p \geq 3,$$

where $i_\nu \in \{1, \ldots, m\}$ for $\nu = 1, \ldots, p$, is a set of p indexes.

Property P3. *Additivity.* If $\mathbf{y}^{(1)} = (y_1^{(1)}, \ldots, y_m^{(1)})'$ and $\mathbf{y}^{(2)} = (y_1^{(2)}, \ldots, y_m^{(2)})'$ are independent random vectors,

$$\text{cum}_{p,\mathbf{y}^{(1)}+\mathbf{y}^{(2)}}(\overbrace{\mathbf{y}^{(1)} + \mathbf{y}^{(2)}, \ldots, \mathbf{y}^{(1)} + \mathbf{y}^{(2)}}^{p-times}) = cum_{p,\mathbf{y}^{(1)}}(\overbrace{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(1)}}^{p-times}) + cum_{p,\mathbf{y}^{(2)}}(\overbrace{\mathbf{y}^{(2)}, \ldots, \mathbf{y}^{(2)}}^{p-times}),$$

i.e.,

$$cum_{p,\mathbf{y}^{(1)}+\mathbf{y}^{(2)}}(y_{i_1}^{(1)}+y_{j_1}^{(2)},\ldots,y_{i_p}^{(1)}+y_{j_p}^{(2)}) = cum_{p,\mathbf{y}^{(1)}}(y_{i_1}^{(1)},\ldots,y_{i_p}^{(1)}) + cum_{p,\mathbf{y}^{(2)}}(y_{j_1}^{(2)},\ldots,y_{j_p}^{(2)}),$$

where $i_\nu, i_{\nu'} \in \{1,\ldots,m\}$ for $\nu,\nu' = 1,\ldots,p$. Note that higher-order moments do not satisfy this property.

Property P4. *Multilinearity.* If $\{\mathbf{U}_l\}_{l=1,\ldots,p}$, is a set of constant matrices of size $m_l \times m$,

$$cum_{p,(\mathbf{U}_1\mathbf{y},\ldots,\mathbf{U}_p\mathbf{y})}(\mathbf{U}_1\mathbf{y},\ldots,\mathbf{U}_p\mathbf{y}) = (\mathbf{U}_1\otimes\ldots\otimes\mathbf{U}_p)cum_{p,\mathbf{y}}(\overbrace{\mathbf{y},\ldots,\mathbf{y}}^{p-times})$$

and, in particular, for any mth dimensional vector of constants, $\mathbf{u} = (u_1,\ldots,u_m) \in \Re^m$,

$$cum_{p,\mathbf{u}'\mathbf{y}}(u_{i_1}y_{i_1},\ldots,u_{i_p}y_{i_p}) = (\textstyle\prod_{\nu=1}^{p} u_{i_\nu})cum_{p,\mathbf{y}}(y_{i_1},\ldots,y_{i_p}).$$

where $i_\nu \in \{1,\ldots,m\}$ for $\nu = 1,\ldots,p$.

Property P5. *Symmetry.* For any permutation $(i_1^{\wp},\ldots,i_p^{\wp})$ of the indexes $(i_1,\ldots,i_p)$, where $i_\nu \in \{1,\ldots,m\}$ for $\nu = 1,\ldots,p$,

$$cum_{p,\mathbf{y}}(y_{i_1},\ldots,y_{i_p}) = cum_{p,\mathbf{y}}(y_{i_1^{\wp}},\ldots,y_{i_p^{\wp}}).$$

Under statistical independence and non-Gaussianity assumptions, properties **P2.** and **P3.** are specially relevant. In particular, **P2.** will be very useful to extract the non-Gaussian part of the observed variables. According to that, it seems reasonable using higher-order cumulants to define BSS criteria and to separate the unobserved non-Gaussian ICs from the Gaussian ones (which could be considered as Gaussian noise). Despite of cumulants of any order could be theoretically used for performing BSS, in practice, the separation criteria are mainly based on fourth-order cumulants. But, why most of the BSS algorithms use fourth-order cumulants instead of third-order ones? The third-order cumulants of random (or stochastic) processes that are symmetric distributed (e.g., Gaussian, Uniform or Laplacian distributed) are equal to zero. Then, the third-order cumulants of non-Gaussian components that were symmetric distributed will be equal to zero, and these components could not be estimated because they would be considered as Gaussian noise. For that reason, our approach will be based on time-delayed fourth-order cumulants.

Given the difficulties of working with the algebraic tensorial nature of cumulants, as we have shown in Chpater 2, Cardoso and Souloumiac (1993) introduce the fourth-order cumulant matrices defined component-wise by,

$$[Q_{\mathbf{y}}(\mathbf{N})]_{(i,j)} = \sum_{k,l=1}^{m} cum_{4,\mathbf{y}}(y_i, y_j, y_k, y_l) n_{kl}, \ 1 \leq i, j \leq m. \tag{3.4}$$

where n_{kl} is the (k,l)-th element of the $m \times m$ matrix $\mathbf{N}$ (note that $Q_{\mathbf{y}}(\mathbf{N})$ is an $m \times m$ matrix). For example, if $\mathbf{N} = \mathbf{I}_m$, the fourth-order cumulant matrix associated to $\mathbf{y}$ is:

$$Q_{\mathbf{y}}(\mathbf{I}_m) = \left(\begin{array}{ccc} cum_{4,\mathbf{y}}(y_1, y_1, y_1, y_1) + \ldots + cum_{4,\mathbf{y}}(y_1, y_1, y_m, y_m) & \cdots & \\ \vdots & \ddots & \\ cum_{4,\mathbf{y}}(y_m, y_1, y_1, y_1) + \ldots + cum_{4,\mathbf{y}}(y_m, y_1, y_m, y_m) & \cdots & \end{array} \right.$$

$$\left. \begin{array}{ccc} \cdots & cum_{4,\mathbf{y}}(y_1, y_m, y_1, y_1) + \ldots + cum_{4,\mathbf{y}}(y_1, y_m, y_m, y_m) \\ \ddots & \vdots \\ \cdots & cum_{4,\mathbf{y}}(y_m, y_m, y_1, y_1) + \ldots + cum_{4,\mathbf{y}}(y_m, y_m, y_m, y_m) \end{array} \right) \tag{3.5}$$

Note that $Q_{\mathbf{y}}(\mathbf{I}_m)$ is equivalent to the 'quadratically weighted covariance' matrix (that is the covariance matrix of the variable $|\mathbf{y}|\mathbf{y}$) proposed by Cardoso (1989), but considering fourth-order cumulants instead fourth-order moments.

Cardoso and Souloumiac (1993) consider a particular case of (3.4) by choosing $(N) = \mathbf{N}^{kl} = \mathbf{e}_k \mathbf{e}_l'$ with $\mathbf{e}_k$ denoting the $m \times 1$ vector that takes value 1 in its kth-position and value 0 elsewhere. Thus, Cardoso and Souloumiac (1993) define the (k,l)-th parallel cumulant slice as the $m \times m$ matrix $Q_{\mathbf{y}}(\mathbf{N}^{kl})$ with (i,j)th-element given by $cum_{4,\mathbf{y}}(y_i, y_j, y_k, y_l)$. That is,

$$Q_{\mathbf{y}}(\mathbf{N}^{kl}) = \left(\begin{array}{cccc} cum_{4,\mathbf{y}}(y_1, y_1, y_k, y_l) & cum_{4,\mathbf{y}}(y_1, y_2, y_k, y_l) & \cdots & cum_{4,\mathbf{y}}(y_1, y_m, y_k, y_l) \\ cum_{4,\mathbf{y}}(y_2, y_1, y_k, y_l) & cum_{4,\mathbf{y}}(y_2, y_2, y_k, y_l) & \cdots & cum_{4,\mathbf{y}}(y_2, y_m, y_k, y_l) \\ \vdots & \vdots & \ddots & \vdots \\ cum_{4,\mathbf{y}}(y_m, y_1, y_k, y_l) & cum_{4,\mathbf{y}}(y_m, y_2, y_k, y_l) & \cdots & cum_{4,\mathbf{y}}(y_m, y_m, y_k, y_l) \end{array} \right) \tag{3.6}$$

Then, each entry in the matrix $Q_{\mathbf{y}}(\mathbf{N}^{kl})$ has one fourth-order cumulant, and therefore it only contains m^2 instead of all the m^4 fourth-order cumulants. The drawback of this approach is that $\mathbf{N}^{kl}$ should be chosen a-priori and there is no information for choosing the optimal one. However, since the set of m^2 matrices $\{\mathbf{N}^{11}, \ldots, \mathbf{N}^{1m}, \mathbf{N}^{21}, \ldots, \mathbf{N}^{2m}, \ldots, \mathbf{N}^{m1}, \ldots, \mathbf{N}^{mm}\}$ is an orthonormal basis for the space of $m \times m$ real matrices, any $m \times m$ matrix, $\mathbf{N}$, can be written as $\mathbf{N} = \sum_{k,l=1}^{m} \mathbf{N}^{kl}$. Then, the problem of choosing a particular matrix $\mathbf{N}^{kl}$ is avoided. Moreover, from (3.4) it is clear that $Q_{\mathbf{y}}(\mathbf{N})$ is given by a linear combination of the (k,l)-th parallel cumulant slices, where the elements of $\mathbf{N}$ are the weights of the linear combination.

That is,

$$Q_{\mathbf{y}}(\mathbf{N}) = \sum_{k,l=1}^{m} n_{kl} Q_{\mathbf{y}}(\mathbf{N}^{kl}). \tag{3.7}$$

An example to illustrate (3.7) is given in the appendix (Section 3.7.1). Thus, instead of computing directly the matrix $Q_{\mathbf{y}}(\mathbf{N})$ (that has the sum of several fourth-order cumulants in each cell), we can compute it more easily as the sum of several (k,l)-th parallel cumulants which only have one cumulant per entry. This approach was used by Cardoso and Souloumiac (1993) to propose the JADE algorithm that was presented in Chapter 2.

From now on, since we are interested in applying higher-order cumulants to perform BSS for time series data, we will move to review the definition of time-delayed cumulants. Focusing on the univariate case, let x_t be a pth-order stationary stochastic process. Then time-delayed pth-order cumulant of x_t is defined as,

$$C_{p,x}(0, \tau_1, \tau_2, \ldots, \tau_{p-1}) = cum(x_t, x_{t+\tau_1}, \ldots, x_{t+\tau_{p-1}}). \tag{3.8}$$

and, due to the stationarity of x_t, (3.8) only depends on the $p-1$ lags. As particular cases of (3.8), the second-, third-, and fourth-order cumulants of x_t are given by:

$$C_{2,x}(0, \tau) = E\{x_t x_{t+\tau}\} \tag{3.9}$$

$$C_{3,x}(0, \tau_1, \tau_2) = E\{x_t x_{t+\tau_1} x_{t+\tau_2}\} \tag{3.10}$$

$$\begin{aligned} C_{4,x}(0, \tau_1, \tau_2, \tau_3) &= E\{x_t x_{t+\tau_1} x_{t+\tau_2} x_{t+\tau_3}\} - C_{2,x}(0, \tau_1) C_{2,x}(0, \tau_3 - \tau_2) - \\ &\quad - C_{2,x}(0, \tau_2) C_{2,x}(0, \tau_3 - \tau_1) - C_{2,x}(0, \tau_3) C_{2,x}(0, \tau_2 - \tau_1) \end{aligned} \tag{3.11}$$

From (3.9) to (3.11) it is clear that, similarly to cumulants of random variables, cumulants of a stochastic process can be written in terms of its moments. In particular, the second-order cumulant given by (3.9) is just the autocovariance matrix of x_t at lag τ, for any $\tau > 0$.

In the multidimensional case, the definition of time-delayed cumulants for a vector of stochastic processes is easy extended. Let $\mathbf{z}_t = (z_{1t}, z_{2t}, \ldots, z_{mt})'$ be a zero-mean pth-order stationary vector of time series. Let $(i_1, \ldots, i_p)$ be a set of p indexes where $i_j = 1, \ldots, m$ for all $j = 1, \ldots, p$. The time-delayed pth-order cumulant of $\mathbf{z}_t$ is defined as the joint pth-order cumulant of the random processes $z_{i_1 t}, z_{i_2 t+\tau_1}, \ldots, z_{i_p t+\tau_{p-1}}$, i.e.,

$$C_{p,\mathbf{z}}^{(i_1,\ldots,i_p)}(0, \tau_1, \tau_2, \ldots, \tau_{p-1}) = cum(z_{i_1 t}, z_{i_2 t+\tau_1}, \ldots, z_{i_p t+\tau_{p-1}}). \tag{3.12}$$

Some particular cases of (3.12) are the time-delayed second-, third-, and fourth-order cumulants of $\mathbf{z}_t$, which follow from (3.9)-(3.11), are given by:

$$C_{2,\mathbf{z}}^{(i,j)}(0,\tau) = E\{z_{it}z_{jt+\tau}\} \tag{3.13}$$

$$C_{3,\mathbf{z}}^{(i,j,k)}(0,\tau_1,\tau_2) = E\{z_{it}z_{jt+\tau_1}z_{kt+\tau_2}\} \tag{3.14}$$

$$\begin{aligned} C_{4,\mathbf{z}}^{(i,j,k,l)}(0,\tau_1,\tau_2,\tau_3) &= E\{z_{it}z_{jt+\tau_1}z_{kt+\tau_2}z_{lt+\tau_3}\} - C_{2,\mathbf{z}}^{(i,j)}(0,\tau_1)C_{2,\mathbf{z}}^{(k,l)}(0,\tau_3-\tau_2) - \\ &\quad -C_{2,\mathbf{z}}^{(i,k)}(0,\tau_2)C_{2,\mathbf{z}}^{(j,l)}(0,\tau_3-\tau_1) - C_{2,\mathbf{z}}^{(i,l)}(0,\tau_3)C_{2,\mathbf{z}}^{(j,k)}(0,\tau_2-\tau_1) \end{aligned} \tag{3.15}$$

Note that here, as before, the relationship between cumulants and moments is clear. In fact, the time-delayed second-order cumulant of $\mathbf{z}_t$ is its τ-time-delayed covariance matrix, for any lag $\tau > 0$.

The properties **P1.-P5.** could be easily extended to the time-delayed cumulants. In particular, the symmetry in the arguments of the time-delayed fourth-order cumulants is stated as:

Property P5'. *Symmetry.* For any permutation $(i^\wp, j^\wp, k^\wp, l^\wp)$ of the indexes (i,j,k,l),

$$C_{4,\mathbf{z}}^{(i,j,k,l)}(\tau_i,\tau_j,\tau_k,\tau_l) = C_{4,\mathbf{z}}^{(i^\wp,j^\wp,k^\wp,l^\wp)}(\tau_{i^\wp},\tau_{j^\wp},\tau_{k^\wp},\tau_{l^\wp}),$$

where τ_i and $\tau_{(\cdot)^\wp}$, that is the corresponding permutation for τ_i, are equal to zero.

By the difficulty of working with time-delayed fourth order cumulants, it would be desirable to have a matrix based notation. For this purpose, we propose to extend the concept of fourth-order cumulant matrices defined by Cardoso and Souloumiac (1993) (see (3.4)). Let $\{(\tau_1,\tau_2,\tau_3)\}_{\tau_i=0,1,\ldots,K}$ any triple of time lags. We define the (τ_1,τ_2,τ_3)-time-delayed fourth-order cumulant matrix associated to $\mathbf{z}_t$, denoted by $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\cdot)$, as an $m \times m$ matrix whose (i,j)-th element is given by,

$$[Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})]_{(i,j)} = \sum_{k,l=1}^{m} cum(z_{it}, z_{jt+\tau_1}, z_{kt+\tau_2}, z_{lt+\tau_3})n_{kl},\ 1 \le i,j \le m, \tag{3.16}$$

where $\mathbf{N}$ is an arbitrary matrix of size $m \times m$. Analogous to the definition given by Cardoso and Souloumiac (1993), we define (k,l)-th (τ_1,τ_2,τ_3)-time-delayed parallel cumulant slice as the matrix whose (i,j)th-element is given by $Cum(z_{it}, z_{jt+\tau_1}, z_{kt+\tau_2}, z_{lt+\tau_3})$. Note that it corresponds to the (τ_1,τ_2,τ_3)-time-delayed fourth-order cumulant matrix $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})$, where $\mathbf{N}^{ij} = \mathbf{e}_i\mathbf{e}_j'$ and $\mathbf{e}_i$ is the $m \times 1$ vector that takes value 1 in its ith-position and value 0

elsewhere. Then, from (3.16), it is clear that $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})$ is given by a linear combination of the (k,l)-th (τ_1,τ_2,τ_3)-time-delayed parallel cumulant slice whose coefficients correspond to the elements of $\mathbf{N}$. For later use, we define the (τ_1,τ_2,τ_3)-time-delayed parallel set $N_p^{(\tau_1,\tau_2,\tau_3)}$ as,

$$N_p^{(0,\tau_1,\tau_2,\tau_3)} = \{Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij}) \mid \mathbf{N}^{ij} = \mathbf{e}_i\mathbf{e}_j', \, 1 \leq i,j \leq m\} \tag{3.17}$$

So far, we have summarized basic concepts about cumulants, and we have pointed out some properties that show the advantages of using higher-order cumulants instead of higher-order moments for BSS. However, working with cumulants has important drawbacks that should be took into account. First of all, working with higher-order cumulants increases the computational load of the problem. Fortunately, they have useful symmetry properties that can reduce it. Moreover, as we mention before, cumulants can be written in terms of moments so, in order to estimate cumulants, we must estimate moments first. We should be careful when we estimate moments because they are very sensitive to outliers. Finally, in order to reduce the variance associated to the sample estimates of the higher-order moments (and then, the higher-order cumulants), we should have large datasets (the lengths of the datasets for which HOS methods are applied should be larger than those where SOS methods are applied).

3.2.2 Joint diagonalization approaches

Joint diagonalization techniques play an important role to solve many statistical problems. For example, common principal components (Flury (1984)) applies joint diagonalization for testing whether or not several covariance matrices of some groups, that come from different populations, have the same eigenvectors. Furthermore, some ICA algorithms as JADE (Cardoso and Souloumiac (1993)) or SOBI (Belouchrani et al. (1997)), that have been used for performing BSS, estimate the ICs by the joint diagonalization of a set of fourth-order cumulant (JADE) or time-delayed covariance matrices (SOBI).

The joint diagonalization problem can be summarized as follows: let $M = \{\mathbf{M}_1, ..., \mathbf{M}_J\}$ be a set of J matrices of size $m \times m$. The aim of joint diagonalization is to find a transformation $\mathbf{V}$ of size $m \times m$ which makes the matrices $\mathbf{V}\mathbf{M}_j\mathbf{V}'$, for all $j = 1, \ldots, J$, as diagonal as possible. That is, it looks for the matrix $\mathbf{V}$ that minimizes the following cost function,

$$\sum_{j=1}^{J} \mathcal{F}(\mathbf{V}\mathbf{M}_j\mathbf{V}') \tag{3.18}$$

where $\digamma(\cdot)$ is a measure of diagonality. The matrix $\mathbf{V}$ is called joint diagonalizer of the set M. The exact diagonalization of more than two matrices simultaneously is not possible unless the matrices have a certain common structure. Otherwise, we can only speak of approximate joint diagonalization.

The way to formulate the joint diagonalization problem depends on the notion of diagonality that is used. Joint diagonalization techniques proposed in the literature can be categorized into three groups. First, we have joint diagonalization techniques that are based on the Frobenius norm formulation. They look for the transformation $\mathbf{V}$ that minimizes the sum of the squares of the off-diagonal elements in $\mathbf{V}\mathbf{M}_j\mathbf{V}'$. That is,

$$\digamma(\mathbf{V}\mathbf{M}_j\mathbf{V}') = \sum_{i \neq l} [(\mathbf{V}\mathbf{M}_j\mathbf{V}')_{il}]^2, \forall \mathbf{M}_j \in M. \tag{3.19}$$

Algorithms based on this approach (for example, JADE and SOBI) are very efficient and they converge very quickly to the optimal solution. However, it is obvious that the trivial solution, $\mathbf{V} = \mathbf{0}$, satisfies this criterion and it is not the optimal transformation we are looking for. In order to avoid that the algorithm converges to the trivial solution, it is usual to require the orthogonality of $\mathbf{V}$. The drawback of this approach is that the orthogonality assumption is too restrictive and may limit its applicability. The second category includes algorithms for simultaneous diagonalization that follow the positive definite formulation. These algorithms are based on the assumption that the J matrices of the set M are symmetric and positive-definited. This approach was Kawamoto et al. (1997) to simultaneously diagonalize several time-delayed covariance matrices (in Chapter 1, we explained this procedure and the measure of lack of diagonality that they used).

This approach is computationally efficient but it may fail when the positive-definiteness of the matrices $\{\mathbf{M}_j\}_{\mathbf{M}_j \in M}$ is not guaranteed. For example, it cannot be used in SOBI because, in general, the time-delayed covariance matrices of any time series vector are not positive-definite. Finally, within the third group there are joint diagonalization techniques that are based on subspace fitting formulation. Those methods formulate the approximate joint diagonalization problem as follows. Given the set of matrices $\{\mathbf{M}_j\}_{\mathbf{M}_j \in M}$, the idea behind this approach is to find an $m \times m$ matrix, $\widetilde{\mathbf{V}}$, and a set of J diagonal $m \times m$ matrices, $\{\mathbf{D}_j\}_{j=1}^{J}$, such that the following contrast function, $\digamma(\cdot)$, that is defined in terms of the Euclidean distance,

$$\digamma(\widetilde{\mathbf{V}}, \mathbf{D}_j) = \|\mathbf{M}_j - \widetilde{\mathbf{V}}\mathbf{D}_j\widetilde{\mathbf{V}}'\|_F^2, \forall \mathbf{M}_j \in M. \tag{3.20}$$

is minimized. Since this approach does not impose any restriction on the matrices $\widetilde{\mathbf{V}}$ and $\{\mathbf{M}_j\}_{j=1}^{J}$ (they are required to be neither orthogonal nor positive-definite matrices), it can be applied to most of practical problems. The problem is that the computational cost of these methods is too high (see van der Veen (2001) and Yeredor (2002) as examples).

Then, applying one of these three approaches to solve a joint diagonalization problem has an important cost: either the algorithm is computationally efficient but strong restrictive assumptions are assumed, or no a-priori assumptions are considered but the algorithm has high computational cost. Combining the 'advantages' of the three previous approaches, Ziehe et al. (2004) proposed a new algorithm for joint diagonalization called FFDIAG (fast Frobenius diagonalization). FFDIAG uses the Frobenius norm formulation (then, it is not too computationally intensive), but without assuming strong a-priori restrictions. Then, the FFDIAG algorithm looks for the transformation $\mathbf{V}$ that minimizes the cost function given by (3.18), where the measure of diagonality, $\digamma(\cdot)$, is defined as in (3.19). The FFDIAG, in order to avoid the convergence to the trivial solution, assumes the invertibility of $\mathbf{V}$, that is less restrictive than the orthogonality assumption.

3.3 Model and assumptions

In this chapter, we consider the ICA model presented in Chapter 2 (see Section 2.1 for more details), where the m-dimensional vector of observations is assumed to be generated by a linear combination of r underlying components that are statistically independent:

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t, \quad t = 1, 2, ..., T \tag{3.21}$$

To estimate the set of underlying components, $\widehat{\mathbf{s}}_t = \mathbf{W}\mathbf{x}_t$, such that they become as independent as possible, we will make the following assumptions:

Assumption A1. The components of $\mathbf{s}_t$ are fourth-order stationary random processes with zero-mean and identity covariance matrix: $E\{\mathbf{s}_t\} = \mathbf{0}$ and $\mathbf{\Gamma}_{\mathbf{s}}(0) = E\{\mathbf{s}_t\mathbf{s}_t'\} = \mathbf{I}_r$.

Assumption A2. $\mathbf{A}$ is a full rank matrix: $rank(\mathbf{A}) = r$.

Given a triple of time lags, (τ_1, τ_2, τ_3), let us define the (τ_1, τ_2, τ_3)-time-delayed fourth-order cumulant set of $\mathbf{s}_t$, denoted by $Q_{\mathbf{s}}^{(0,\tau_1,\tau_2,\tau_3)}$, as:

$$Q_{\mathbf{s}}^{(0,\tau_1,\tau_2,\tau_3)} = \{cum(s_{it}, s_{jt+\tau_1}, s_{kt+\tau_2}, s_{lt+\tau_3}) \mid 1 \leq i, j, k, l \leq r\}.$$

The elements of $Q_{\mathbf{s}}^{(0,\tau_1,\tau_2,\tau_3)}$ with identical indexes corresponds to the (τ_1,τ_2,τ_3)-time-delayed fourth-order autocumulants of $\mathbf{s}_t$,

$$\kappa_q^{\mathbf{s}}(\tau_1,\tau_2,\tau_3) = cum(s_{qt}, s_{qt+\tau_1}, s_{qt+\tau_2}, s_{qt+\tau_3}),\ \forall q = 1,\ldots,r$$

Assumption A3. No more than one component of $\mathbf{s}_t$ could be Gaussian distributed. That assumption implies that, for at least $r-1$ components, $\{s_{qt}\}_{q=1}^{r-1}$, there exists triples of time lags, (τ_1,τ_2,τ_3), such that,

$$\kappa_q^{\mathbf{s}}(\tau_1,\tau_2,\tau_3) \neq 0,\ \forall q = 1,\ldots,r-1.$$

Assumption A4. The components of $\mathbf{s}_t$ are mutually independent. Then, the (τ_1,τ_2,τ_3)-time-delayed fourth-order cross-cumulants of $\mathbf{s}_t$ vanish and the non-zero elements of $Q_{\mathbf{s}}^{(0,\tau_1,\tau_2,\tau_3)}$ are still $\kappa_q^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)$, for $q = 1,\ldots,r$.

Assumption A5. There exist consistent estimates for the matrices $\mathbf{\Gamma}_{\mathbf{x}}(\tau)$ and the sets $Q_{\mathbf{x}}^{(0,\tau_1,\tau_2,\tau_3)}$, $\forall\ \tau,\tau_1,\tau_2,\tau_3 \geq 0$.

Under the above assumptions, and according to the model (3.21), we have some structures on the observations. On the one hand, based on SOS, the (time-delayed) covariance matrices of the observations are given by:

$$\mathbf{\Gamma}_{\mathbf{x}}(0) = \mathbf{A}\mathbf{A}',\ \text{and}\ \mathbf{\Gamma}_{\mathbf{x}}(\tau) = \mathbf{A}\mathbf{\Gamma}_{\mathbf{s}}(\tau)\mathbf{A}',\ \forall\tau > 0 \tag{3.22}$$

where $\mathbf{\Gamma}_{\mathbf{s}}(\tau) = E\{\mathbf{s}_t\mathbf{s}'_{t+\tau}\} = Diag(\gamma_1(\tau),\ldots,\gamma_r(\tau))$ (as usual, $\gamma_q(\tau)$ denotes the autocovariance of s_{qt} at lag τ). On the other hand, based on higher-order cumulants and applying some of their properties (additivity, multilinearity, and Gaussian rejection), the (τ_1,τ_2,τ_3)-time-delayed fourth-order cumulant set of $\mathbf{x}_t$ satisfies:

$$Q_{\mathbf{x}}^{(0,\tau_1,\tau_2,\tau_3)} = (\mathbf{A}\otimes\mathbf{A}')Q_{\mathbf{s}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{A}\otimes\mathbf{A}')',\ \text{for any triple}(\tau_1,\tau_2,\tau_3), \tag{3.23}$$

where all the elements of $Q_{\mathbf{s}}^{(0,\tau_1,\tau_2,\tau_3)}$, with the exception of $\kappa_q^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)$, are equal to zero. Then, based on the above relations, it's clear that assumption **A5.** guarantees the existence of consistent estimates for $\mathbf{\Gamma}_{\mathbf{s}}(\tau)$ and $Q_{\mathbf{s}}^{(0,\tau_1,\tau_2,\tau_3)}$, for any $\tau,\tau_1,\tau_2,\tau_3 \geq 0$.

3.4 A BSS approach for non-Gaussian (non-linear) time series

This section describes our blind identification approach for dealing with non-Gaussian (non-linear) data that exhibit a significant temporal structure. Our proposal is a higher-order

cumulants-based approach that performs BSS by the simultaneous diagonalization of several time-delayed fourth-order cumulants matrices. It combines both SOS as well as HOS. First, it exploits second-order information to standardize the data and to restrict the estimation of the mixing matrix to the space of orthogonal matrices (see Chapter 1 for more details about the multivariate standardization procedure). After the standardization, model (3.21) can be written as,

$$\mathbf{z}_t = \mathbf{U}\mathbf{s}_t, \tag{3.24}$$

where $\mathbf{U}$ is an orthogonal matrix of size $r \times r$ and $\mathbf{z}_t$ is the r-dimensional vector of standardized observations: $E\{\mathbf{z}_t\} = \mathbf{0}$ and $E\{\mathbf{z}_t\mathbf{z}_t'\} = \mathbf{I}_r$ (Note that $r = m$ in the basic ICA model, or $r < m$ if the dimension of the data is reduced). Second, the FOTBI algorithm introduces HOS to determine the mixing matrix that guarantees the independence of the non-Gaussian (or non-linear) time series components.

In the next subsections, we will detail our procedure. Next, we will explain two different approaches to estimate the $r \times r$ orthogonal matrix $\mathbf{U}$ using time-delayed fourth-order cumulants. Second, we will show how to combine these two approaches and link them to the joint diagonalization problem in order to formulate our fourth-order temporal blind identification (FOTBI) technique. Finally, we will sketch the main steps to implement FOTBI.

3.4.1 Estimation of the orthogonal matrix using HOS

Here two approaches to determine the orthogonal matrix $\mathbf{U}$ are presented. On the one hand, $\mathbf{U}$ can be identify as the matrix of eigenvector of a set of some time-delayed fourth-order cumulant matrices. On the other hand, $\mathbf{U}$ can be the solution to the optimization criterion that consists on minimizing the sum of the squares of several time-delayed high-order cross-cumulants of the components. In the following, we point out how these approaches could be applied to our problem.

Approaches based on eigendecomposition

According to the model (3.24), and applying some properties of the cumulants (additivity, multilinearity, and Gaussian rejection), the (τ_1, τ_2, τ_3)-time-delayed fourth-order cumulant matrices, given by (3.16), can be written as

$$Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}) = \sum_{q=1}^{m} \kappa_q^{\mathbf{s}}(\tau_1, \tau_2, \tau_3)\ \mathbf{u}_q'\mathbf{N}\mathbf{u}_q\ \mathbf{u}_q\mathbf{u}_q', \text{ for any } r \times r \text{ matrix } \mathbf{N}, \tag{3.25}$$

where $\kappa_q^{\mathbf{s}}(\tau_1,\tau_2,\tau_3) = Cum(s_{qt}, s_{qt+\tau_1}, s_{qt+\tau_2}, s_{qt+\tau_3})$ and $\mathbf{u}_q = (u_{1q}, u_{2q}, \ldots, u_{rq})'$. Equivalently, (3.25) is given by:

$$Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}) = \mathbf{U}\mathbf{\Lambda}_{\mathbf{N}}^{(\tau_1,\tau_2,\tau_3)}\mathbf{U}', \text{ for any } r \times r \text{ matrix } \mathbf{N}, \tag{3.26}$$

where

$$\mathbf{\Lambda}_{\mathbf{N}}^{(\tau_1,\tau_2,\tau_3)} = Diag(\kappa_1^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)\ \mathbf{u}_1'\mathbf{N}\mathbf{u}_1, \ldots, \kappa_r^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)\ \mathbf{u}_r'\mathbf{N}\mathbf{u}_r), \text{ for } \tau_1,\tau_2,\tau_3 = 0,1,\ldots,K. \tag{3.27}$$

From expression (3.26), the orthogonal matrix $\mathbf{U}$ can be identified as the matrix of eigenvectors of $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})$, for any $r \times r$ matrix $\mathbf{N}$ and for any triple (τ_1,τ_2,τ_3) . However, due to the indetermination of the eigenvalue decomposition, $\mathbf{U}$ could not be identified if the (τ_1,τ_2,τ_3)-time-delayed fourth-order cumulant matrices has not different eigenvalues. According to (3.26), the eigenvalues of $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})$ are the elements of $\mathbf{\Lambda}_{\mathbf{N}}^{(\tau_1,\tau_2,\tau_3)}$ given by (3.27). If $\mathbf{N} = \mathbf{I}_r$, due to the orthogonality of $\mathbf{U}$, the eigenvalues of $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{I}_r)$ are the (τ_1,τ_2,τ_3)-time-delayed autocumulants of the ICs, $\kappa_q^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)$. Then, the matrix $\mathbf{U}$ could be identified only if all the (τ_1,τ_2,τ_3)-time-delayed autocumulants of the ICs are distinct. Since this situation is very likely when we consider a particular (τ_1,τ_2,τ_3)-time-delayed parallel set $N_p^{(0,\tau_1,\tau_2,\tau_3)}$, this approach seems to be feasible for our purpose.

Approaches based on optimization of cumulant criteria

Let $\mathbf{V}$ be an $r \times r$ orthogonal matrix such that,

$$\mathbf{b}_t = \mathbf{V}'\mathbf{z}_t. \tag{3.28}$$

Plugging (3.24) into (3.28), $\mathbf{b}_t = \mathbf{V}'\mathbf{U}\mathbf{s}_t$. Thus, if $\mathbf{V}$ is essentially equal to $\mathbf{U}$, then $\mathbf{b}_t = \mathbf{s}_t$. Therefore, since the components of $\mathbf{b}_t$ correspond to the ICs, their time-delayed high-order cross-cumulants should be equal to zero. According to that, the matrix $\mathbf{U}$ could be estimated as the $r \times r$ orthogonal transformation that minimizes the sum of the squares of the (τ_1,τ_2,τ_3)-time-delayed high-order cross-cumulants of $\mathbf{b}_t$, that is equivalent to maximize their (τ_1,τ_2,τ_3)-time-delayed high-order autocumulants. Following this approach, and focusing on the (τ_1,τ_2,τ_3)-time-delayed fourth-order cumulants of $\mathbf{b}_t$, for $\{(\tau_1,\tau_2,\tau_3)\}_{0\leq\tau_1,\tau_2,\tau_3\leq K}$, $\mathbf{U}$ could be estimated as the orthogonal matrix that maximizes,

$$f_1(\mathbf{V}) = \sum_{\tau_1,\tau_2,\tau_3=0}^{K} \sum_{i=1}^{r} \{cum(b_{it}, b_{it+\tau_1}, b_{it+\tau_2}, b_{it+\tau_3})\}^2. \tag{3.29}$$

We modify the contrast function given by (3.29), and following Cardoso and Souloumiac (1993), we propose to minimize the sum of the squares of the (τ_1, τ_2, τ_3)-time-delayed fourth-order cross-cumulants of $\mathbf{b}_t$, whose first and second indices are different. That is, $\mathbf{U}$ is the orthogonal matrix that maximizes

$$f_2(\mathbf{V}) = \sum_{\tau_1,\tau_2,\tau_3=0}^{K} \sum_{i,k,l=1}^{r} \{cum(b_{it}, b_{it+\tau_1}, b_{kt+\tau_2}, b_{lt+\tau_3})\}^2. \tag{3.30}$$

Note that, considering the contrast function (3.30) allows to perform BSS by the joint diagonalization of some (τ_1, τ_2, τ_3)-time-delayed fourth-order cumulant matrices, for any triple of time lags $\{(\tau_1, \tau_2, \tau_3)\}_{0 \leq \tau_1,\tau_2,\tau_3 \leq K}$. This fact will be shown next.

3.4.2 Joint diagonalization for our blind identification approach

Our aim is to identify a set of unknown ICs which generate the vector of non-Gaussian (non-linear) time series that we observe. The ICs will also be non-Gaussian (or non-linear) and have a significant autocorrelation structure. Then, we would like to estimate the ICs exploiting both the non-Gaussianity as well as the temporal structure of the observations. For this purpose, extending the concept of fourth-order cumulant matrices given by Cardoso and Souloumiac (1993), we combine HOS and temporal dependence to define the time-delayed fourth-order cumulant matrices (see (3.16)). Assuming that the model (3.21) is the real one, the (τ_1, τ_2, τ_3)-time-delayed fourth-order cumulant matrices of the standardized data and the ones of the ICs are related by equation (3.26). Since $\mathbf{U}$ is orthogonal, (3.26) is equivalent to,

$$\mathbf{\Lambda}_{\mathbf{N}}^{(0,\tau_1,\tau_2,\tau_3)} = \mathbf{U}' Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})\mathbf{U}, \text{ for any } r \times r \text{ matrix } \mathbf{N}, \tag{3.31}$$

that can be particularized for any (τ_1, τ_2, τ_3)-time-delayed parallel set $N_p^{(0,\tau_1,\tau_2,\tau_3)}$ defined as in (3.17). Note that equation (3.31) nests within the formulation of the joint diagonalization, and $\mathbf{U}$ can be obtained by the simultaneous diagonalization of some (τ_1, τ_2, τ_3)-time-delayed parallel sets $N_p^{(0,\tau_1,\tau_2,\tau_3)}$. We propose to follow this estimation principle and identify the orthogonal matrix $\mathbf{U}$ as the joint diagonalizer of a set of some (τ_1, τ_2, τ_3)-time-delayed parallel set $N_p^{(0,\tau_1,\tau_2,\tau_3)}$. That is, according to our proposal, $\mathbf{U}$ will be determined as the orthogonal transformation that minimizes

$$f_3(\mathbf{U}, N_p) = \sum_{\tau_1,\tau_2,\tau_3=0}^{K} \sum_{i,j=1}^{r} \text{off}(\mathbf{U}' Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})\mathbf{U}), \tag{3.32}$$

where $N_p = \{N_p^{(0,\tau_1,\tau_2,\tau_3)}, 0 \leq \tau_1, \tau_2, \tau_3 \leq K\}$ and off$(\cdot)$ is defined as the sum of the squares of the off-diagonal elements of a matrix. This is equivalent to identify $\mathbf{U}$ as the orthogonal

transformation that maximizes the following joint diagonalization criterion:

$$f_4(\mathbf{U}, N_p) = \sum_{\tau_1,\tau_2,\tau_3=0}^{K} \sum_{i,j=1}^{r} \left(\text{diag}(\mathbf{U}' Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})\mathbf{U})\right)^2, \tag{3.33}$$

where N_p is defined as before and diag$(\cdot)$ represent the sum of the squares of the diagonal elements of a matrix.

For convenience, we will follow the second joint diagonalization criterion and propose to determine $\mathbf{U}$ as the maximizer of (3.33). Thus, we link our joint diagonalization approach (given by the criterion (3.33)) to the two approaches introduced in the previous section (the eigendecomposition-based and the optimization-based approaches) as follows:

Proposition 1. For any $r \times r$ orthogonal matrix $\mathbf{V}$, $f_2(\mathbf{V}) = f_4(\mathbf{V}, N_p)$.

Proof. See the appendix (Section 3.7.2)

Proposition 1 shows the equivalence between maximizing $f_2(\mathbf{V})$ and the joint diagonalization of N_p. Then, our proposal to determine the orthogonal transformation, $\mathbf{U}$, as the joint diagonalizer of N_p can be used as a criterion for BSS. However, the identifiability of $\mathbf{U}$ should be guaranteed. We achieve that fact in two steps: first, we show the uniqueness of the joint diagonalizer of any (τ_1, τ_2, τ_3)-time-delayed parallel set, $N_p^{(0,\tau_1,\tau_2,\tau_3)}$ (first identifiability condition); then, we generalize the result to the case of several (τ_1, τ_2, τ_3)-time-delayed parallel sets, and guarantee the identifiability of $\mathbf{U}$ via the joint diagonalization of N_p (second identifiability condition). Formally, we have:

Theorem 1. (First identifiability condition) Let (τ_1, τ_2, τ_3) be a triple of time lags and $\mathbf{V}$ be an $r \times r$ orthogonal matrix such that:

H1. $\mathbf{V}$ is a joint diagonalizer of the (τ_1, τ_2, τ_3)-time-delayed parallel set $N_p^{(0,\tau_1,\tau_2,\tau_3)}$. That is:

$$\mathbf{V}' Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})\mathbf{V} = Diag(d_1, \ldots, d_r), \text{ for any } \{\mathbf{N}^{ij}\}_{1\leq i,j\leq r} \tag{3.34}$$

H2. for any $\{\mathbf{N}^{ij}\}_{1\leq i,j\leq r}$, the eigenvalues of $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})$ are all distinct,

$$\forall\ 1 \leq q_1 \neq q_2 \leq r,\ \kappa_{q_1}^{\mathbf{s}}(\tau_1, \tau_2, \tau_3)\mathbf{u}_{q_1}' \mathbf{N}^{ij} \mathbf{u}_{q_1} \neq \kappa_{q_2}^{\mathbf{s}}(\tau_1, \tau_2, \tau_3)\mathbf{u}_{q_2}' \mathbf{N}^{ij} \mathbf{u}_{q_2}. \tag{3.35}$$

Then,

1.1. $\mathbf{V}$ is essentially equal to $\mathbf{U}$: $\mathbf{V} = \mathbf{U}$,

1.2. it exists a permutation ϱ on $\{1,\ldots,r\}$, such that:

$$(\kappa_1^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)\mathbf{u}_1'\mathbf{N}\mathbf{u}_1,\ldots,\kappa_r^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)\mathbf{u}_r'\mathbf{N}\mathbf{u}_r)=(d_{\varrho(1)},\ldots,d_{\varrho(r)}). \tag{3.36}$$

The proof of this theorem is straightforward: applying the spectral theorem for normal matrices to $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})$, is guaranteed that there exists an $r\times r$ orthogonal matrix, $\mathbf{V}$, that satisfies (3.34) and then, **H1**. holds. Furthermore, by linearity of $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\cdot)$, the matrix $\mathbf{V}$ also diagonalizes any linear combination of matrices $\{\mathbf{N}^{ij}\}$, and then, $\widetilde{\mathbf{V}}'Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})\widetilde{V}'$ is diagonal for any matrix $\mathbf{N}$ of size $r\times r$. Then, to guarantee that $\mathbf{V}$ is essentially equal to $\mathbf{U}$, we need to prove that condition (3.35) holds. However, it is not trivial to show the existence of a triple of time lags, (τ_1,τ_2,τ_3), that satisfies (3.35). Then, the identification of the components of $\mathbf{s}_t$ is only possible if the eigenvalues of $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})$, for any triple (τ_1,τ_2,τ_3), are distinct. As it is not easy to determine a-priori the triple of time lags, (τ_1,τ_2,τ_3), that satisfies previous condition, we have proposed to diagonalize, simultaneously, a set of time-delayed fourth-order cumulant matrices, $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})$, for several triple of time lags, (τ_1,τ_2,τ_3), with $\tau_i=0,1,\ldots,K\ \forall\ i=1,2,3$. Then, we will estimate $\mathbf{U}$ as a joint diagonalizer of N_p, but the uniqueness of the orthogonal matrix $\mathbf{U}$ should be guaranteed. This is equivalent to show that the joint diagonalizer of N_p is unique, and is given by:

Theorem 2. (Second identifiability condition) Let $\Im=\{(\tau_1,\tau_2,\tau_3),0\leq\tau_1,\tau_2,\tau_3\leq K\}$ be a set of triples of time lags and let $\mathbf{N}$ be an arbitrary matrix of size $r\times r$. Furthermore, let $N_p=\{N_p^{(0,\tau_1,\tau_2,\tau_3)}\mid(\tau_1,\tau_2,\tau_3)\in\Im\}=\{Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})\mid(\tau_1,\tau_2,\tau_3)\in\Im\}$ be a set of time-delayed fourth-order cumulant matrices of size $r\times r$ such that:

$$Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})=\mathbf{U}\Lambda_{\mathbf{N}}^{(\tau_1,\tau_2,\tau_3)}\mathbf{U}',\ \forall(\tau_1,\tau_2,\tau_3)\in\Im \tag{3.37}$$

where $\Lambda_{\mathbf{N}}^{(\tau_1,\tau_2,\tau_3)}$ is an $r\times r$ diagonal matrix given by (3.27). Then, any joint diagonalizer of N_p is essentially equal to $\mathbf{U}$ if, and only if,

$$\forall 1\leq q_1\neq q_2\leq r,\ \exists(\tau_1,\tau_2,\tau_3)\in\Im\ :\kappa_{q_1}^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)\mathbf{u}_{q_1}'\mathbf{N}^{ij}\mathbf{u}_{q_1}\neq\kappa_{q_2}^{\mathbf{s}}(\tau_1,\tau_2,\tau_3)\mathbf{u}_{q_2}'\mathbf{N}^{ij}\mathbf{u}_{q_2}. \tag{3.38}$$

The proof of this theorem is a consequence of the essential uniqueness of joint diagonalization (see theorem 3 in Belouchrani et al. (1997)). From theorem 2, if (3.38) holds, then the uniqueness of the joint diagonalizer of N_p is guaranteed, and corresponds to the orthogonal matrix $\mathbf{U}$. Therefore, the key point is to find a triple of time lags, $(\tau_1,\tau_2,\tau_3)\in\Im$ that satisfies (3.38), and this is always possible due to the non-Gaussianity assumption. In effect: let's set

$N = \sum_{q=1}^{r} q[\kappa_q^{\mathbf{s}}(\tau_1, \tau_2, \tau_3)]^{-1}\mathbf{u}_q\mathbf{u}_q'$, for any $(\tau_1, \tau_2, \tau_3) \in \Im$. By assumption, no more than one IC can be Gaussian distributed. Then, either the r ICs are non-Gaussian or there is one IC that is Gaussian and the other are not. In the first case, if the r ICs are non-Gaussian distributed, $\kappa_q^{\mathbf{s}}(\tau_1, \tau_2, \tau_3) \neq 0$ for all $q = 1, \ldots, r$, and then, from (3.25), the eigenvalues of $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})$ are all distinct. In the second case, let consider, for example, that the Gaussian IC is the first one. Then, $\kappa_1^{\mathbf{s}}(\tau_1, \tau_2, \tau_3) = 0$ and $\kappa_q^{\mathbf{s}}(\tau_1, \tau_2, \tau_3) \neq 0$ for all $q > 1$, and $Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N})$ has r integer different eigenvalues (one is equal to zero and the others are different from zero and different each other too). Therefore, under the non-Gaussianity assumption, (3.38) holds, and the uniqueness condition for identifiability is guaranteed.

Note that an important fact of our approach is that it does not require the exact diagonalization of each individual time-delayed fourth-order cumulantmatrix. The orthogonal matrix that maximizes (3.33) can be seen as the 'approximate joint diagonalizer' of N_p. We will describe our new approach, that is called FOTBI (Fourth Order Temporal Blind Identification), in the next section.

3.4.3 Implementation of the FOTBI algorithm

Based on the previous sections, the fourth-order temporal blind identification (FOTBI) algorithm can be described as follows:

1. Remove the mean from the data and compute its sample covariance matrix as $\widehat{\mathbf{\Gamma}}_{\mathbf{x}}(0) = \frac{1}{T}\sum_{t=1}^{T}\mathbf{x}_t\mathbf{x}_t'$. Then, applying the EVD to $\widehat{\mathbf{\Gamma}}_{\mathbf{x}}(0)$, compute the whitening matrix, $\widehat{\mathbf{M}}$, and obtain the standardized data, $\mathbf{z}_t = \widehat{\mathbf{M}}\mathbf{x}_t$.

2. For a fixed set of triples of time lags, $\{(\tau_1, \tau_2, \tau_3)\}_{\tau_i=0,\ldots,K}$, form the sample estimates of the (τ_1, τ_2, τ_3)-time-delayed parallel set of fourth-order cumulant matrices associated to $\mathbf{z}_t$, $\widehat{N}_p^{(0,\tau_1,\tau_2,\tau_3)} = \{\widehat{Q}_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})\}_{1\leq i,j\leq r}$. Note that, applying definitions (3.9) and (3.11) in the equation (3.26)), the expression for $\widehat{Q}_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})$ is simplified and can be rewritten in terms of moments as:

$$\begin{aligned}\widehat{Q}_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij}) &= E\{\mathbf{z}_{t+\tau_2}'\mathbf{N}^{ij}\mathbf{z}_{t+\tau_3}\ \mathbf{z}_t\mathbf{z}_{t+\tau_1}'\} - \widehat{\mathbf{\Gamma}}_{\mathbf{z}}(\tau_1)\mathrm{Tr}(\mathbf{N}^{ij}\widehat{\mathbf{\Gamma}}_{\mathbf{z}}(\tau_2-\tau_3)) \\ &\quad -\widehat{\mathbf{\Gamma}}_{\mathbf{z}}(\tau_2)\mathbf{N}^{ij}\widehat{\mathbf{\Gamma}}_{\mathbf{z}}(\tau_1-\tau_3) - \widehat{\mathbf{\Gamma}}_{\mathbf{z}}(\tau_3)\mathbf{N}^{ji}\widehat{\mathbf{\Gamma}}_{\mathbf{z}}(\tau_1-\tau_2).\end{aligned} \quad (3.39)$$

3. Estimate the orthogonal matrix $\widehat{\mathbf{U}}$ of size $r \times r$ as the transformation that jointly diagonalizes the set $\widehat{N}_p^{(0,\tau_1,\tau_2,\tau_3)}$, for some triples $\{(\tau_1, \tau_2, \tau_3)\}_{\tau_i=0,\ldots,K}$. In order to obtain

the joint diagonalizer, $\widehat{\mathbf{U}}$, Jacobi rotation techniques are applied.

4. An estimate of the loading matrix is given by $\widehat{\mathbf{A}} = \widehat{\mathbf{M}}^{\sharp}\widehat{\mathbf{U}}$, and the estimates of the ICs are $\widehat{\mathbf{s}}_t = \widehat{\mathbf{W}}\mathbf{x}_t$, where $\widehat{\mathbf{W}} = \widehat{\mathbf{U}}'\widehat{\mathbf{M}}$.

3.5 Simulation Experiments

In this section, we illustrate the performance of the FOTBI algorithm by means of a Monte Carlo simulation. We report the results for three different experiments, which have been selected to validate the FOTBI algorithm under different scenarios. Furthermore, in order to test the FOTBI algorithm, we compare its performance with respect to JADE and SOBI algorithms, which were presented in Chapter 2.

For each experiment, the procedure is as follows. First, in order to satisfy assumption **A1.**, the r components of $\mathbf{s}_t$ are standardized. Then, they are mixed according to the (3.21) to get the vector of observations, $\mathbf{x}_t$. Finally, the aim is to estimate the ICs (and $\mathbf{A}$) only from the observations. We apply three ICA algorithms: JADE, SOBI, and FOTBI, and compare their performance by computing the correlation coefficients and the mean square errors (MSE) between each original components and its corresponding estimation.

For the three experiments, we generate $N = 1000$ realizations of each component, $\{s_{it}\}_{i=1,\ldots,r}^{t=1,\ldots,T}$, for different sample sizes, $T = 100, 500, 1000$. For each algorithm, we report the average values of both, the correlation coefficients and the MSEs between the original and the estimated components, computed over the N independent replicas for the different sample sizes. Furthermore, we provide the mean average values measured over the r ICs. Thus, we can analyze the global and the component by component performance of each method, and establish some ranking among the algorithms depending on the individual features of the ICs.

Experiment 1: The components are defined as univariate time series that are Gaussian distributed. In that scenario, where the components are Gaussian and exhibit a strong autocorrelation, SOBI seems to be the ICA algorithm that provides the most accurate ICs (the ICs given by SOBI will be temporal decorrelated and temporal independent too). Then, we are interested in testing the performance of FOTBI, and determining whether it is a competitive algorithm or not under this scenario.

In this experiment, we consider $r = 4$ components that follow different ARIMA models

(they are describe in Table 3.1). Here, the loading matrix is $\mathbf{A}_{4\times4} = [\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3\mathbf{a}_4]$ where $\mathbf{a}_1' = (2,1,1,1)$, $\mathbf{a}_2' = (1,2,1,1)$, $\mathbf{a}_3' = (1,1,2,1)$, and $\mathbf{a}_4' = (1,1,1,2)$.

Table 3.1: Experiment 1: Definition of the original components

Model specification		
$s_{1t} \sim AR(1)$	$s_{1t} = 0.68s_{1t-1} + n_{1t}$	$n_{1t} \sim N(0,1)$
$s_{2t} \sim ARMA(1,1)$	$s_{2t} = 0.5s_{2t-1} + n_{2t} - 0.3n_{2t-1}$	$n_{2t} \sim N(0,1)$
$s_{3t} \sim MA(1)$	$s_{3t} = n_{3t} - 0.8n_{3t-1}$	$n_{3t} \sim N(0,1)$
$s_{4t} \sim MA(2)$	$s_{4t} = n_{4t} + 0.6n_{4t-1} - 0.3n_{4t-1}$	$n_{4t} \sim N(0,1)$

NOTE: $\{n_{jt}\}_{j=1}^4$ are Gaussian white noise processes. Then, $\forall t_1 \neq t_2$, n_{jt_1} and n_{jt_2} are uncorrelated and, therefore, under Gaussianity, they are independent too.

Table 3.2: Experiment 1: Average values for the correlation coefficients and the MSE between the original and the estimated components

	s_{jt}	Correlation			MSE		
T	j	JADE	SOBI	FOTBI	JADE	SOBI	FOTBI
100	1	0.671	**0.913**	0.721	0.655	**0.172**	0.555
	2	0.665	**0.875**	0.734	0.667	**0.249**	0.529
	3	0.636	**0.924**	0.733	0.724	**0.151**	0.532
	4	0.598	**0.844**	0.706	0.800	**0.311**	0.585
	Average	0.642	**0.889**	0.723	0.712	**0.221**	0.550
500	1	0.748	**0.977**	0.826	0.504	**0.046**	0.347
	2	0.725	**0.979**	0.826	0.549	**0.041**	0.347
	3	0.707	**0.993**	0.844	0.585	**0.015**	0.312
	4	0.676	**0.965**	0.819	0.647	**0.070**	0.361
	Average	0.714	**0.979**	0.829	0.571	**0.043**	0.342
1000	1	0.770	**0.986**	0.854	0.459	**0.028**	0.292
	2	0.744	**0.990**	0.850	0.512	**0.020**	0.300
	3	0.716	**0.997**	0.865	0.569	**0.006**	0.269
	4	0.689	**0.983**	0.837	0.621	**0.033**	0.325
	Average	0.730	**0.989**	0.852	0.540	**0.022**	0.296

According to the average values, shown in Table 3.2, we see that the results are as we expected: SOBI and JADE present, respectively, the best and the worst performance, independently of the sample size. In this experiment, where the ICs are Gaussian (linear) and have a significant autocorrelation structure, SOBI provides more reliable identification of the unobserved components than JADE or FOTBI do. Comparing the results of JADE and FOTBI, we see that the ICs estimated by FOTBI are more correlated to the original ones and have lower MSEs than those estimated by JADE. This result is in accordance with the assumptions made for each algorithm: JADE identifies the ICs based only on their non-Gaussianity, but FOTBI exploits their temporal structure too. Then, as in this experiment the ICs are

time-dependent processes, FOTBI outperforms JADE.

Experiment 2: In this example, we define the components as univariate time series processes that are non-Gaussian distributed. We design this experiment because to the fact of dealing with temporal dependent components, we have added the fact that they are non-Gaussian distributed, since we have proposed the FOTBI algorithm to estimate the ICs under these conditions (note that the particular structure handle in this scenario is of course not exploited by the JADE and the SOBI algorithms).

We generate $r = 5$ components that fit to non-Gaussian ARIMA models (see Table 3.3 for the model specifications), and fix the parameters of the mixing matrix as: $\mathbf{A}_{5\times 5} = [\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3\mathbf{a}_4\mathbf{a}_5]$ where $\mathbf{a}_1' = (2,1,1,1,1)$, $\mathbf{a}_2' = (1,2,1,1,1)$, $\mathbf{a}_3' = (1,1,2,1,1)$, $\mathbf{a}_4' = (1,1,1,2,1)$, and $\mathbf{a}_5' = (1,1,1,1,2)$. We have designed non-linear (non-Gaussian) ICs that are temporal dependent. As in the first experiment, for comparing the performance of JADE, SOBI and FOTBI, we compute the average values for the correlation coefficients and the MSEs. From the results, that are provided in Table 3.4, we see that SOBI performs worse than JADE, and FOTBI clearly outperforms JADE and SOBI for any sample size. This is to be expected: if only either the non-Gaussianity (JADE) or the temporal structure (SOBI) of the data is exploited to identify the ICs then, useful information is ignored and makes the performance of the separation worse; FOTBI, that combines both the non-Gaussianity and the temporal structure of the data, seems to be the most reliable method to identify non-Gaussian (non-linear) time series components. Moreover, we would like to notice that the FOTBI performance improves when the sample size increases, and this is because longer data lengths reduce the variance associated with the sample time-delayed fourth-order cumulant matrices estimates.

Table 3.3: Experiment 2: Definition of the original components

Model specification		
$s_{1t} \sim ARMA(1,1)$	$s_{1t} = 0.9s_{1t-1} + n_{1t} - 0.8n_{1t-1}$	$n_{1t} \sim t_{15}$
$s_{2t} \sim ARMA(1,1)$	$s_{2t} = 0.72s_{2t-1} + n_{2t} - 0.5n_{2t-1}$	$n_{2t} \sim$ GED, $\kappa = 2$
$s_{3t} \sim ARMA(1,2)$	$s_{3t} = 0.75s_{3t-1} + n_{3t} - 0.2n_{3t-1} - 0.55n_{3t-2}$	$n_{3t} \sim t_9$
$s_{4t} \sim AR(1)$	$s_{4t} = 0.82s_{4t-1} + n_{4t}$	$n_{4t} \sim$ GED, $\kappa = 1.3$
$s_{5t} \sim AR(2)$	$s_{5t} = 0.11s_{5t-1} + 0.25s_{5t-2} + n_{5t}$	$n_{5t} \sim t_5$

NOTE: The processes $\{n_{jt}\}_{j=1}^4$ are non-Gaussian random noises, where n_{jt_1} and n_{jt_2} are uncorrelated $\forall t_1 \neq t_2$. However, n_{jt_1} and n_{jt_2}, with $t_1 \neq t_2$, are not independent (under non-Gaussianity, no-correlation does not imply independence).

Experiment 3: Here, the components are non-linear time series processes, and some of

them are allowed to be non-stationary. According to the assumption **A1.**, the components should be fourth-order stationary (in fact, stationarity is a general assumption of ICA). However, we would like to explore how standard ICA algorithms perform when stationary assumption does not hold.

Table 3.4: Experiment 2: Average values for the correlation coefficients and the MSE between the original and the estimated components

	s_{jt}	Correlation			MSE		
T	j	JADE	SOBI	FOTBI	JADE	SOBI	FOTBI
100	1	0.864	0.725	**0.914**	0.224	0.548	**0.171**
	2	0.845	0.725	**0.890**	0.264	0.548	**0.218**
	3	0.830	0.742	**0.875**	0.293	0.513	**0.248**
	4	0.900	0.776	**0.955**	0.154	0.446	**0.090**
	5	0.889	0.729	**0.938**	0.177	0.539	**0.123**
	Average	0.866	0.739	**0.914**	0.223	0.519	**0.170**
500	1	0.920	0.868	**0.981**	0.081	0.264	**0.038**
	2	0.916	0.856	**0.973**	0.088	0.288	**0.054**
	3	0.914	0.867	**0.968**	0.092	0.265	**0.064**
	4	0.929	0.922	**0.994**	0.061	0.156	**0.012**
	5	0.925	0.881	**0.989**	0.070	0.238	**0.021**
	Average	0.921	0.879	**0.981**	0.078	0.242	**0.038**
1000	1	0.927	0.907	**0.990**	0.066	0.185	**0.021**
	2	0.926	0.900	**0.987**	0.068	0.199	**0.025**
	3	0.925	0.901	**0.985**	0.069	0.199	**0.029**
	4	0.932	0.958	**0.997**	0.055	0.084	**0.005**
	5	0.930	0.929	**0.995**	0.059	0.143	**0.011**
	Average	0.928	0.919	**0.991**	0.064	0.162	**0.018**

For this purpose, we generate $r = 4$ non-Gaussian components which are defined in Table 3.5. Note that s_{1t}, s_{2t}, and s_{3t} represent, respectively, a deterministic polynomial trend, a seasonal component, and a cycle, which can be considered the basic components of a time series processes. Here, the loading matrix $\mathbf{A}$ is randomly generated from the $U(0,1)$ distribution.

Table 3.5: Experiment 3: Definition of the original components

Model specification	
$s_{1t} = (1 + 3t + t^2 + 2t^3)/15 + n_{1t}$	$n_{1t} \sim U(0,1)$
$s_{2t} = sin(3.3t/\pi) + 2cos(3.3t/20\pi) + n_{2t}$	$n_{2t} \sim U(0,1)$
$s_{3t} = 2sin((1/60)\pi t) + 2cos((1/60)\pi t) + n_{3t}$	$n_{3t} \sim U(0,1)$
$s_{4t} = n_{4t}$	$n_{4t} \sim U(0,1)$

For the third experiment, the average values for the correlation coefficients and the MSEs

are given in Table 3.6. Here, the results are mixed and depend on the sample size. Thus, for small sample size ($T = 100$), SOBI has the best separation performance for all the ICs except for the third one (the cyclical component). FOTBI and JADE performs similarly but slightly worse than SOBI. When the sample size increases, the three algorithms provide similar results although FOTBI is slightly better than the others. That is because using HOS, in order to reduce the variance associated with their estimates, requires longer data lengths that SOS do. In addition, from the results provided by the third experiment, it seems that any of the analyzed algorithms, JADE, SOBI and FOTBI, performs quite good to separate non-stationary time series components. Then, it hints at the possibility of applying these three BSS algorithms to explore datasets that are non-stationary and have significant temporal structure.

Table 3.6: Experiment 3: Average values for the correlation coefficients and the MSE between the original and the estimated components

	s_{jt}	Correlation			MSE		
T	j	JADE	SOBI	FOTBI	JADE	SOBI	FOTBI
100	1	0.770	**0.880**	0.764	0.458	**0.240**	0.470
	2	0.960	**0.990**	0.958	0.081	**0.020**	0.084
	3	0.901	0.777	**0.930**	0.196	0.445	**0.140**
	4	0.941	**0.977**	0.965	0.118	**0.045**	0.071
	Average	0.893	**0.906**	0.904	0.213	**0.187**	0.191
500	1	0.976	0.953	**0.995**	0.048	0.094	**0.011**
	2	0.988	**0.999**	**0.999**	0.024	**0.002**	**0.002**
	3	0.979	0.954	**0.994**	0.043	0.093	**0.011**
	4	0.987	0.990	**0.993**	0.025	**0.008**	0.013
	Average	0.983	0.974	**0.995**	0.035	0.049	**0.009**
1000	1	0.992	0.994	**1.000**	0.017	0.012	**0.000**
	2	0.996	0.999	**1.000**	0.008	0.002	**0.000**
	3	0.997	0.998	**0.999**	0.006	**0.002**	**0.002**
	4	0.993	0.997	**0.998**	0.014	0.007	**0.005**
	Average	0.994	0.997	**0.999**	0.011	0.006	**0.002**

From these simulations, we conclude that under the assumptions of non-Gaussian (non-linear) and temporal dependent ICs, FOTBI provides better performance than those algorithms that only exploit one of two previous assumptions and ignore useful information for the separation (such as JADE and SOBI do). Furthermore, we see that for autocorrelated data, the performance of FOTBI is acceptable even though the ICs were linear or non-stationary.

3.6 Concluding remarks

In this chapter we propose a new BSS blind separation approach for non-Gaussian (non-linear) data that have significant temporal structure. Our fourth-order temporal blind identification (FOTBI) algorithm is based on the joint diagonalization of several time-delayed fourth-order cumulant matrices. It firstly uses second order time structure to standardize the data, and then it introduces high-order information to exploit the non-Gaussianity and to achieve the independence of the components. The advantage of FOTBI are twofold: one, with FOTBI, it is not needed to make any a-priori assumption about the features of the data. Then, all the information is available for the identification of the ICs and the performance of the separation improves. Two, FOTBI guarantees the independence of the temporally correlated components in a non-Gaussian environment. Previous algorithms that deal with temporal dependent data (e.g., SOBI and TDSEP) are based on SOS and obtain uncorrelated components that are not independent under non-Gaussianity assumption.

The Monte Carlo simulation results show that FOTBI performs better than JADE and SOBI when the ICs are non-Gaussian and exhibit pronounced autocorrelation structure. This fact confirms our guess: choosing a-priori one of the two assumptions (either non-Gaussianity or temporal structure) about the data, deteriorates the separation performance of the method. Moreover, we see that FOTBI provides quite acceptable results when it is used for the separation of temporally decorrelated Gaussian components. Finally, the experiments provide some hints about the possibility of applying FOTBI for the separation of non-stationary time series components successfully.

Further research will be directed to the analysis of the asymptotic performance of our method. Furthermore, since the estimation of the sample time-delayed fourth-order cumulants is quite sensitive to the sample size, it would be interesting to explore additional techniques for getting more robust and better estimates.

3.7 Appendix

3.7.1 Example of fourth-order cumulant matrices:

Here we present an example to illustrate the equality given by (3.7). If $m = 2$ and $\mathbf{N} = (3, 2; -1, 4)$, we apply (3.7) to compute $Q_{\mathbf{y}}(\mathbf{N})$ as:

$$Q_{\mathbf{y}}(\mathbf{N}) = 3Q_{\mathbf{y}}(\mathbf{N}^{11}) + 2Q_{\mathbf{y}}(\mathbf{N}^{12}) - Q_{\mathbf{y}}(\mathbf{N}^{21}) + 4Q_{\mathbf{y}}(\mathbf{N}^{22}). \tag{3.40}$$

Using (3.6) in (3.40), we have:

$$\begin{aligned} Q_{\mathbf{y}}(\mathbf{N}) = 3 \begin{pmatrix} cum_{4,\mathbf{y}}(y_1,y_1,y_1,y_1) & cum_{4,\mathbf{y}}(y_1,y_2,y_1,y_1) \\ cum_{4,\mathbf{y}}(y_2,y_1,y_1,y_1) & cum_{4,\mathbf{y}}(y_2,y_2,y_1,y_1) \end{pmatrix} + 2 \begin{pmatrix} cum_{4,\mathbf{y}}(y_1,y_1,y_1,y_2) & cum_{4,\mathbf{y}}(y_1,y_2,y_1,y_2) \\ cum_{4,\mathbf{y}}(y_2,y_1,y_1,y_2) & cum_{4,\mathbf{y}}(y_2,y_2,y_1,y_2) \end{pmatrix} - \\ - \begin{pmatrix} cum_{4,\mathbf{y}}(y_1,y_1,y_2,y_1) & cum_{4,\mathbf{y}}(y_1,y_2,y_2,y_1) \\ cum_{4,\mathbf{y}}(y_2,y_1,y_2,y_1) & cum_{4,\mathbf{y}}(y_2,y_2,y_2,y_1) \end{pmatrix} + 4 \begin{pmatrix} cum_{4,\mathbf{y}}(y_1,y_1,y_2,y_2) & cum_{4,\mathbf{y}}(y_1,y_2,y_2,y_2) \\ cum_{4,\mathbf{y}}(y_2,y_1,y_2,y_2) & cum_{4,\mathbf{y}}(y_2,y_2,y_2,y_2) \end{pmatrix} \end{aligned} \tag{3.41}$$

3.7.2 Proof of proposition 1:

To proof that is enough to show the maximization of $\sum_{i,k,l=1}^{r}\{cum(b_{it}, b_{it+\tau_1}, b_{kt+\tau_2}, b_{lt+\tau_3})\}^2$ is equivalent to the joint diagonalization of the (τ_1, τ_2, τ_3)-time delayed parallel set $N_p^{(0,\tau_1,\tau_2,\tau_3)}$, for any triple of time lags (τ_1, τ_2, τ_3). That is, if we show that, for any triple of time lags, (τ_1, τ_2, τ_3),

$$f_4(\mathbf{V}, N_p^{(0,\tau_1,\tau_2,\tau_3)}) = \sum_{i,k,l=1}^{r}\{cum(b_{it}, b_{it+\tau_1}, b_{kt+\tau_2}, b_{lt+\tau_3})\}^2, \text{ for any } (\tau_1, \tau_2, \tau_3), \tag{3.42}$$

then, the proposition 1 holds. Let's prove (3.42): by definitions (3.17) and (3.33), for any (τ_1, τ_2, τ_3) triple of time lags, we have:

$$f_4(\mathbf{V}, N_p^{(0,\tau_1,\tau_2,\tau_3)}) = \sum_{i,j=1}^{r} |\mathbf{diag}(\mathbf{V}' Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})\mathbf{V})|^2 = \sum_{i,j,h=1}^{r} |\mathbf{v}_h' Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})\mathbf{v}_h|^2. \tag{3.43}$$

From definition (3.16), it easy to see that $\mathbf{v}_h' Q_{\mathbf{z}}^{(0,\tau_1,\tau_2,\tau_3)}(\mathbf{N}^{ij})\mathbf{v}_h = Trace(\mathbf{N}^{ij} Q_{\mathbf{z}}^{(\tau_2,\tau_3,0,\tau_1)}(\mathbf{v}_h\mathbf{v}_h'))$. Using this property in (3.43), we have:

$$f_4(\mathbf{V}, N_p^{(0,\tau_1,\tau_2,\tau_3)}) = \textstyle\sum_{i,j,h=1}^{r} |Trace(\mathbf{N}^{ij} Q_{\mathbf{z}}^{(\tau_2,\tau_3,0,\tau_1)}(\mathbf{v}_h\mathbf{v}_h'))|^2.$$

Since $\{\mathbf{N}^{ij} = \mathbf{e}_i\mathbf{e}_j', 1 \le i, j \le r\}$ and $\{\mathbf{v}_i\mathbf{v}_j', 1 \le i, j \le r\}$ are two sets of orthonormal basis for the space of matrices of size $r \times r$, the Frobenius norm of $Q_{\mathbf{z}}^{(\tau_2,\tau_3,0,\tau_1)}(\mathbf{v}_h\mathbf{v}_h')$ can be expressed

onto each of this basis and

$$\sum_{i,j,h=1}^{r} |Trace(\mathbf{N}^{ij} Q_{\mathbf{z}}^{(\tau_2,\tau_3,0,\tau_1)}(\mathbf{v}_h \mathbf{v}_h'))|^2 = \sum_{h=1}^{r} \|Q_{\mathbf{z}}^{(\tau_2,\tau_3,0,\tau_1)}(\mathbf{v}_h \mathbf{v}_h')\|_{FRO}^2 = \sum_{h,k,l=1}^{r} |\mathbf{v}_k' Q_{\mathbf{z}}^{(\tau_2,\tau_3,0,\tau_1)}(\mathbf{v}_h \mathbf{v}_h')\mathbf{v}_l|^2 \quad (3.44)$$

Using the definition (3.28) for $\mathbf{b}_t$ and applying the multilinearity of the cumulants (see property **P4.**) to $\mathbf{v}_k' Q_{\mathbf{z}}^{(\tau_2,\tau_3,0,\tau_1)}(\mathbf{v}_h \mathbf{v}_h')\mathbf{v}_l$, we have

$$\textstyle\sum_{h,k,l=1}^{r} |\mathbf{v}_k' Q_{\mathbf{z}}^{(\tau_2,\tau_3,0,\tau_1)}(\mathbf{v}_h \mathbf{v}_h')\mathbf{v}_l|^2 = \sum_{h,k,l=1}^{r} |cum(b_{kt+\tau_2}, b_{lt+\tau_3}, b_{it}, b_{it+\tau_1})|^2,$$

that, by symmetries of the cumulants (see property **P5'.**), is equal to

$$\textstyle\sum_{h,k,l=1}^{r} |cum(b_{it}, b_{it+\tau_1}, b_{kt+\tau_2}, b_{lt+\tau_3})|^2.$$

Then, (3.42) holds, and following the previous argument, the proposition 1 is proved.

Chapter 4

Exploring ICA for time series decomposition

In this chapter, we apply independent component analysis (ICA) to perform signal extraction in multivariate time series data. Moreover, we explore the idea of forecasting a set of multiple time series using the predictions of a small number of independent components. Some Monte Carlo simulation experiments are carried out to investigate the performance of three ICA algorithms presented in previous chapters, JADE, SOBI, and FOTBI, in order to extract components such as trend, cycle, and seasonal components. Moreover, we empirically test the performance of those three ICA procedures on capturing the dynamic relationships among the industrial production index (IPI) time series of four European countries. We also compare the accuracy of the IPI time series forecasts using a few JADE, SOBI, and FOTBI components, at different time horizons. According to the results, FOTBI seems to be a good starting point for automatic time series signal extraction procedures, and it also provides quite accurate forecasts for the IPIs.

4.1 Introduction

In many applications of empirical sciences such as Medicine, Engineering, and Economics, when the data are observed with a high level of noise, extracting the relevant patterns from the observations becomes an important task. The problem of estimating those underlying components (components of interest) from the observations is known as signal extraction or feature extraction problem. Thus, considering the additive decomposition,

$$x_t = \chi_t + \nu_t, \tag{4.1}$$

where x_t is the observed data, χ_t is the set of interesting components (signal), and ν_t is the noise process (not necessarily white) which is assumed to be independent of χ_t, the aim of signal extraction is to isolate the signal from the noise. The estimates of the signal will be obtained by filtering the observations, $\widehat{\chi}_t = \mathbf{F}x_t$, in such a way that the signal estimates satisfy the minimum mean square error (MMSE) criterion.

If x_t is a univariate time series process, model (4.1) might represent the decomposition of x_t as the sum of some underlying components of interest, which are usually interpreted in terms of trend, seasonality, and cycle, among others. Then, some economic applications such as seasonal adjustment, detrending, and analysis of the business cycles, can be seen as particular cases of signal extraction problems, where the interesting signals (χ_t) are, respectively, seasonally adjusted components, trends, and cycles.

Several approaches have been developed for solving the signal extraction problem in the univariate framework. The first one, called 'ad-hoc' filter design approach, includes methods that use moving-average smoothing filters to estimate the signal. These methods are supported by the main central statistical agencies for trend extraction and seasonal adjustment in time series. The X-11 filter (Shiskin et al. (1967)) for seasonal adjustment and the Beveridge-Nelson (Beveridge and Nelson (1981)), the Baxter and King (Baxter and King (1995)), and the Hodrick-Prescott (Hodrick and Prescott (1997)) filters, which were used to estimate the trend-cycle components, are some well-known examples of the 'ad-hoc' filter design approach. The main disadvantage of these filters is that they do not take into account the structure of the time series process and they could produce spurious results and over/under-estimated components. Trying to solve this important limitation, it has been developed the so-called model-based procedures, where the filter is derived from statistical models and it is adapted to the particular structure of the time series processes. Two directions emerge within the model-based procedures: the ARIMA-model-based approach and the structural modelling approach.

On the one hand, the ARIMA-model-based procedures (Box et al. (1978), Burman (1980), Bell and Hillmer (1984), Hillmer and Tiao (1982), among others) directly identify a parsimonious ARIMA model for the observations. Then, univariate models for the components are derived with the restriction that the aggregation of those models yields the ARIMA model identified for the data. Because there is not a unique admissible decomposition, these methods apply the 'canonical decomposition' (see Box et al. (1978)) to solve identifiability problems.

Within this approach, the most popular algorithm is the SEATS/TRAMO software (Gómez and Maravall (1996), Maravall (1993)) that is based on the filter developed by Burman (1980).

On the other hand, the structural modelling approach (Harvey (1989), Young et al. (1999), Bujosa et al. (2007), among others), instead of using a-priori information to specify a model for the observations, directly assumes different stochastic linear models for the unobserved components. These models are formulated within an stochastic state space setting, and the Kalman filter is used to estimate the parameters. STAMP (Koopman et al. (1995)) is a well known software that directly specifies structural models for the components of interest in the time domain framework. Another implementations of this approach, such as the CAPTAIN MatLab Toolbox program (Young and Pedregal (1999), Taylor et al. (2007)) and the linear dynamic harmonic regression algorithm (Bujosa et al. (2007)), are developed in the spectral framework assuming that the data are periodic time series.

When we move to the multivariate framework, where the issue of information redundancy in the observed data set is usually arising, capturing the most 'interesting' features of the data might be as important as (or even more than) it was in the univariate case. In particular, when we observe multiple time series data where dynamic relationships are involved, the components of interest might be common to different time series. Thus, extracting those underlying common components, which probably may have a useful interpretation in terms of common trends or common seasonality, becomes an important task in multivariate time series analysis. Dynamic factor models (see Forni et al. (2000) and Peña and Poncela (2006b), among others) and multivariate structural time series models (Harvey (1989)) have traditionally dealt with this topic. However, it is hard to develop 'automatic' (or quasi-automatic) procedures for signal extraction in the multivariate framework, and STAMP (Koopman et al. (1995)) is the only model-based procedure that can handle this problem.

As an alternative to model-based procedures, principal component analysis (PCA) is usually applied to multivariate data sets with the aim of noise and/or dimension reduction, and signal extraction. PCA can be seen as an 'automatic' procedure for signal extraction, where the relevant information is given by those components that explain the largest amount of variance in the data. PCA is quite successful in multivariate linear data but, when the data are non-Gaussian (non-linear), PCA has difficulty in separating the underlying components. Empirical applications show that, under non-Gaussianity assumption, the components extracted by PCA are quite far away from the real ones (see for example, Oja (1982) and Särelä and

Valpola (2005), among others). Moreover, these empirical results reveal that independent component analysis (ICA) estimates the underlying components better than PCA does.

In this chapter, we explore the performance of ICA in multivariate time series signal extraction, and analyze how the ICA components could be useful to predict the observations. ICA seems to be appropriate when we observed several economic time series data, where some components of interest, such as trend or seasonal variations, can be assumed to be fairly independent.

This chapter is organized as follows. Section 4.2 reviews the main approaches that have been presented in the literature for signal extraction. Then, we introduce the procedure to forecast the data using a set of ICA components. In Section 4.4, we carry out some simulation experiments to support the idea that ICA could be seen as the first step for automatic signal extraction procedures. Next, we apply ICA to extract the components of interest in the industrial production indexes of several European countries. In addition, we analyze how these data are forecasted using a few ICA components. Finally, Section 4.6 gives some concluding remarks.

4.2 Model-based methods for signal extraction

Most of the latest signal extraction algorithms are model-based procedures where the observations are decomposed as the sum of some components of interest. For example, for time series data, estimating the trend and the seasonality is important to analyze the main movements of the time series, and to obtain seasonal adjusted data, respectively. In general, since an infinite number of decompositions is possible, the identification of the components is not unique, and additional assumptions should be made.

An attractive feature of model based-approaches is that, since they are based on specific statistical models for the observations and/or the components, model-based approaches could facilitate analysis and inference. Next, we review the ARIMA-model based and the structural modelling approaches, paying attention to some of their well-known implementations.

4.2.1 ARIMA-model based methods

The ARIMA-model based methodology (Box et al. (1978), Hillmer and Tiao (1982), Burman (1980), Maravall and Pierce (1987), amongst others) came up as an alternative procedure

for seasonal adjustment of time series data. The ARIMA-model based approach starts by applying the Box and Jenkins methodology to specify an ARIMA model that describes the behavior of the time series data. Then, univariate models for the components are derived so that their aggregation should be consistent with the original ARIMA model. Two assumption are made to guarantee the unique identification of the components: first, it is assumed that the components of interest are mutually uncorrelated; second, it is applied the canonical principle (Box et al. (1978)) which maximizes the variance of the noise component and leads the 'interesting' components to be as stable as possible (Hillmer and Tiao (1982)). The underlying components are computed by the Wiener-Kolmogorov filter (Box et al. (1978)) that provides the MMSE estimators of the components, even for non-stationary time series (Bell (1984)).

Popular procedures that take the ARIMA-model based approach are the X-11-ARIMA (Dagum (1980)), the X-12-ARIMA (Findley et al. (1998)) and the SEATS/TRAMO software (Gómez and Maravall (1996); Maravall (1993)). These methods are commonly used by official statistical agencies to get seasonally adjusted data (for example, Statistics Canada, US Bureau of the Census, and Bank of Spain are well-known examples of official agencies that apply, respectively, X-11-ARIMA, X-12-ARIMA, and SEATS/TRAMO programs, to seasonal adjustment).

The first two procedures, the X-11-ARIMA and X-12-ARIMA, are based on moving averages filters and then, they are not ARIMA-model based procedures themselves. However, since at the first stage the two procedures identify an ARIMA model for the observations and the definitions of the signals are 'implicit', the X-11- and the X-12-ARIMA are considered as ARIMA-model based procedures. Both X-11- and X-12-ARIMA uses the X-11 filter (Cleveland and Tiao (1976)), that applies a set of centered moving averages to estimate the seasonal components. The problem is that when moving averages filters are used, many observations of the beginning and the end of the series are lost and the seasonal effect could be underestimated. The X-11-ARIMA, trying to avoid the loss of observations, uses the ARIMA model fitted to the original series for extending the length of the data set (forecasting and backcasting). The X-12-ARIMA follows the same idea that the X-11-ARIMA but introduces a pre-adjustment program, REGARIMA, that is applied to the original time series data (before the identification of the ARIMA model) to detect outliers and to estimate some deterministic effects (for example, the calendar effect).

The SEATS/TRAMO programs (Gómez and Maravall (1996); Maravall (1993)) are efficient and automatic procedures which are mainly applied for seasonal adjustment and trend-cycle estimation. First, TRAMO (Time series Regression with ARIMA noise, Missing values and Outliers) is a pre-adjustment program that is applied to the univariate time series data to pre-test for the log-level specification, to detect and correct outliers (additive outliers, transitory changes, and level shifts), to interpolate missing values, and to correct other deterministic effects such as Trading Day, Leap Year, and Easter effects. Then, TRAMO specifies a set of possible models for the pre-adjusted data, estimates them by maximum likelihood, and selects the 'optimal' one based on AIC and BIC criteria. Finally, according to the selected model, TRAMO forecasts the data to extend the time series and thus, it reduces the bias when a new observation enters to the model. Next, SEATS (Signal Extraction in ARIMA Time Series) derives univariate ARIMA models for the stochastic components so that they reflect the usual structures associated to trend, cyclical (or trend-cycle), and seasonal components. SEATS uses the canonical principle (Box et al. (1978)) to avoid identifiability problems and applies the Burman-Wilson algorithm (Burman (1980)) to estimate the components (MMSE estimators). The final estimates for the unobserved components are obtained by the aggregation of the deterministic effects (computed by TRAMO) of each individual component to the stochastic components given by SEATS.

ARIMA-model based procedures have two important drawbacks: first, since the models for the components are not directly specified (they are derived from the original ARIMA model for the observations and should be consistent with it) those components could not be easily interpretable, a-posteriori, in terms of trend or seasonality; second, since the ARIMA-model based procedures consider a common noise for all the components, the components' estimates could be correlated, and therefore, the assumption of uncorrelated components would not be satisfied. In structural modelling procedures, this problem is solved considering independent noises for each component.

4.2.2 Structural modelling approach

The structural modelling approach is an alternative model-based methodology for signal extraction that is based on unobserved components models. Contrary to the ARIMA-model based methodology the structural modelling procedures directly specify univariate stochastic models for the underlying components and then, their interpretability in terms of trends,

seasonalities and cycles is guaranteed.

We distinguish two structural modelling specifications: the structural time series approach (Harvey (1989)) that is implemented in the STAMP software (Koopman et al. (1995)), and the dynamic harmonic regression approach (Young et al. (1999)), that is implemented in the CAPTAIN Toolbox for Matlab (Young and Pedregal (1999), Taylor et al. (2007)) as well as in the new linear dynamic harmonic regression algorithm (Bujosa et al. (2007)). The main differences between the dynamic harmonic regression model (Young et al. (1999)) and Harvey's structural model (Harvey (1989)) rely on the model specification for the periodic components and the optimization method used to estimate the parameters. In the following, we discuss these two approaches.

Structural time series approach

Structural time series models (Harvey (1989)) are formulated in terms of unobserved components which have a direct interpretation. According to Harvey (1989), the structural time series models 'are not more than regression models in which explanatory variables are a function of time and the parameters change with time'. These explanatory variables represent dynamic features of the data (such as stochastic trends, cycles, and/or seasonalities). The starting point in structural time series models is to identify those features and model them in such a way that we can obtain useful predictions for the time series data. Structural time series models are usually formulated as state space models and the parameters of the unobserved components models are estimated using the Kalman filter and related algorithms (see Harvey (1989) for a detailed description of the state space and the Kalman filter methodologies).

STAMP (Structural Time Series Analyzer, Modeler and Predictor) (Koopman et al. (1995)) is a standard signal extraction procedure that is implemented according to structural time series models (as they are defined in Harvey (1989)). STAMP, contrary to alternative signal extraction procedures that are only developed in the univariate framework (e.g. SEATS/TRAMO), can be applied to extract the components of interest in both univariate as well as multivariate time series data.

The basic structural time series model assumes that univariate time series can be decomposed into additive stochastic components as

$$y_t = \mu_t + \psi_t + \gamma_t + \epsilon_t. \tag{4.2}$$

where μ_t represents the trend, ψ_t the cycle, γ_t the seasonality, and ϵ_t the irregular component (a structural time series model should not be necessarily defined in terms of these four UCs; it may be defined only by some of them). There are different specifications to formulate the stochastic process for each component. By default, for univariate time series data, STAMP considers a basic structural time series model which chooses the local linear trend (LLT) model for the trend, a stochastic cyclical component, a stochastic trigonometric model for the seasonality, and a white noise process for the irregular term, $\epsilon_t \sim NID(0, \sigma_\epsilon^2)$.

According to the LLT model, the stochastic trend is given by

$$\begin{array}{l} \mu_t = \mu_{t-1} + \beta_{t-1} + \eta_t,\ \eta_t \sim NID(0, \sigma_\eta^2), \\ \beta_t = \beta_{t-1} + \xi_t,\ \xi_t \sim NID(0, \sigma_\xi^2), \end{array} \tag{4.3}$$

where β_t is the stochastic slope of the trend. Here, the two noises, η_t and ξ_t, and the irregular component in (4.2), ϵ_t, are assumed to be mutually uncorrelated. Different specifications for the trend are possible: either the level (μ_t) or the slope (β_t) could be deterministic instead of stochastic, and the slope might not be included in the model (see Harvey (1989) for a complete revision of different specifications).

The stochastic cyclical component is given by

$$\begin{pmatrix} \psi_t \\ \psi_t^* \end{pmatrix} = \rho^\psi \begin{pmatrix} cos\lambda^c & sin\lambda^c \\ -sin\lambda^c & cos\lambda^c \end{pmatrix} \begin{pmatrix} \psi_{t-1} \\ \psi_{t-1}^* \end{pmatrix} + \begin{pmatrix} \kappa_t \\ \kappa_t^* \end{pmatrix} \tag{4.4}$$

where ρ^ψ and λ^c represent, respectively, the damping factor and the cyclical frequency (measured in radians) which take values $0 < \rho^\psi \leq 1$ and $0 \leq \lambda^c \leq \pi$, respectively. The period of the cycle is given by $2\pi/\lambda^c$. The cyclical disturbances, $\kappa_t \sim NID(0, \sigma_\kappa^2)$ and $\kappa_t^* \sim NID(0, \sigma_\kappa^2)$, are assumed to have the same variance and to be mutually uncorrelated.

The trigonometric formulation for the seasonal component is

$$\gamma_t = \sum_{j=1}^{[s/2]} \gamma_{j,t}, \tag{4.5}$$

where $[s/2] = \begin{cases} s/2, & \text{if } s \text{ is even} \\ (s-1)/2, & \text{if } s \text{ is odd} \end{cases}$ (s is the number of seasonal frequencies in a period), and $\gamma_{j,t}$ is defined as a non-stationary stochastic cycle, for each $j = 1, 2, ..., [s/2]$. That is, it is given by (4.4) where $\rho^\psi = 1$, and the frequency for $\gamma_{j,t}$, in radians, is $\lambda^c \equiv \lambda_j = 2j\pi/s$. As an alternative to the trigonometric form, the seasonality may be formulated using the dummy variable form (see Harvey (1989) for more details).

When we have more than one time series, dynamic interactions usually appear among most (or all) of them and capturing those relationships requires the joint estimation of the

multiple time series within a multivariate framework. Multivariate structural time series models are straightforward generalized from the univariate ones as follows: the data, that is now a vector of time series, $\mathbf{y}_t$, decompose as in (4.2), but considering vector components instead of scalars. The models that are specified for each vectorial component generalize the ones formulated in the univariate case (for instance, models (4.3), (4.4), and (4.5) for the trend, the cycle, and the seasonal components, respectively), replacing the scalar components with vectors. In the particular, for multivariate cycles, the damping factor, ρ^{ψ}, and the cyclical frequency, λ^{c}, are assumed to take the same value for all the series. This kind of models, called SUTSE (Seemingly Unrelated Time Series Equations), assumes that the disturbances of different components are multivariate normally distributed and mutually uncorrelated in all time periods.

In SUTSE models, the disturbance covariance matrices, in particular their ranks, play an important role to determine the presence of common factors. On the one hand, if the disturbance covariance matrices are of full-rank, then each individual time series of $\mathbf{y}_t$ will have its own components (trend, and/or cycle, and/or seasonality, and/or irregular components), and the interactions among the different time series are reflected as non-zero off-diagonal elements in the covariances matrices of the disturbances. On the other hand, if there is any disturbance covariance matrix with reduced rank, then the component associated to this disturbance term will be common to more than one series. Thus, multivariate structural time series models consider the possibility of dealing with cointegrated time series. The cointegration restrictions, that are interpreted as a lower rank of the disturbance covariance matrix, can be imposed a-priori, but it may also be given by the result of the model estimation. The general multivariate unobserved components model nests more specific models with a restricted number of common components. For instance, the non-stationary dynamic factor models (Peña and Poncela (2006b)), where the common factors can be formulated in terms of UC with a useful interpretation.

STAMP solves the signal extraction problem in both cases: general multivariate structural time series models (SUTSE) and multivariate structural time series models with common factors and cointegration. STAMP deals with common factor models writing them in terms of SUTSE models with reduced rank disturbance covariance matrices.

The problem of structural time series models (either univariate or multivariate) the a-priori structure imposed to the components (which makes easier their interpretation) may

not be appropriate for the particular series at hand, and wrong specifications could produce serious misleading errors.

Dynamic harmonic regression approach

As in Harvey's structural time series approach, the dynamic harmonic regression approach (Young et al. (1999)) directly specifies unobserved components models for the components within an stochastic state space setting. However, whereas structural time series models formulate the unobserved components models in the time domain (see previous section for more details), the whole process of identification and estimation for the dynamic harmonic regression model is formulated in the frequency domain.

The dynamic harmonic regression model assumes that the univariate time series, y_t, can be decomposed as in (4.2). According the dynamic harmonic regression approach, these additive unobserved components (trend, cycle, seasonal and irregular components) have a so-called dynamic harmonic representation. That is, each component is defined by a linear combination of sines and cosines with time varying coefficients, which are modelled as generalized random walk (GRW) stochastic processes (Young et al. (1999)). More formally, the general definition of the dynamic harmonic regression components is given by

$$s_t^{p_j} = a_{jt}cos(w_j t) + b_{jt}sin(w_j t) \tag{4.6}$$

where p_j and $w_j = 1/p_j$ are, respectively, the period and the frequency associated with the jth dynamic harmonic regression component, and $\{a_{jt}, b_{jt}\}$ follow generalized random walk (GRW) processes, that include the random walk (RW), integrated random walk (IRW), and smoothed random walk (SRW) processes as special examples. The trend component corresponds to the zero frequency component, s_t^{∞}, that is described by a GRW process of the form:

$$\begin{pmatrix} \mu_t \\ \beta_t \end{pmatrix} = \begin{pmatrix} \alpha & \beta \\ 0 & \gamma \end{pmatrix} \begin{pmatrix} \mu_{t-1} \\ \beta_{t-1} \end{pmatrix} + \begin{pmatrix} \delta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \eta_t \\ \xi_t \end{pmatrix}, \text{ where } \begin{pmatrix} \eta_t \\ \xi_t \end{pmatrix} \sim \text{WN}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_\eta^2 & 0 \\ 0 & \sigma_\xi^2 \end{pmatrix} \right) \tag{4.7}$$

where μ_t and β_t are, respectively, the changing level and the slope of the trend component.

The periodic components (cycle, ψ_t, and seasonality, γ_t) are given by

$$\psi_t \equiv \gamma_t = \sum_{j=1}^{R} s_t^{p_j}, \tag{4.8}$$

where $j = 1, 2, ..., R$ are the associated periodic frequencies and $s_t^{p_j}$ are defined as in (4.6). The time varying coefficients, $\{a_{jt}, b_{jt}\}$, that define the seasonal component, are usually assumed

to be random walk (RW) processes,

$$\begin{aligned} a_{jt} &= a_{jt-1} + \eta^a_{jt}, \text{ where } \eta^a_{jt} \sim N(0, \sigma^2_{\eta_a}), \\ b_{jt} &= b_{jt-1} + \eta^b_{jt}, \text{ where } \eta^b_{jt} \sim N(0, \sigma^2_{\eta_b}). \end{aligned} \tag{4.9}$$

From the state space formulation of the dynamic harmonic regression model, Young et al. (1999) derive an algorithm that combines the Kalman filter and the fixed interval smoothing to estimate the structural parameters (usually called hyper-parameters) of the unobserved components models. The dynamic harmonic regression algorithm estimates the autoregressive spectrum of the observed time series, and computes the hyper-parameters as the minimum non-linear least squares estimates of the difference between the logarithmic pseudo-spectrum of the dynamic harmonic regression model and the logarithmic autoregressive spectrum of the data (see Young et al. (1999) for more details). The dynamic harmonic regression algorithm is implemented in the CAPTAIN Toolbox for Matlab (see Young and Pedregal (1999), Taylor et al. (2007), among others). An alternative algorithm for the identification and estimation of dynamic harmonic regression models is the linear dynamic harmonic regression (Bujosa et al. (2007)) that simplifies and reduces the computational complexity of the basic dynamic harmonic regression algorithm by using an alternative cost function. The advantages of the linear dynamic harmonic regression algorithm are twofold: first, it eliminates the poles in the objective function of the dynamic harmonic regression algorithm by considering a quadratic cost function (that it is obtained by a linear algebraic transformation, using the ARIMA reduced-form representation of the components). Second, it requires less input information than other existing alternatives. In fact, the linear dynamic harmonic regression only needs the time series data (in a row) and the nature of its periodicity to extract the dynamic harmonic regression components (for a detailed description of the linear dynamic harmonic regression algorithm see Bujosa et al. (2007)).

4.3 ICA for prediction and signal extraction

In the literature, we can find many applications which use ICA to separate the components of interest in multivariate data sets (see, for example, Bingham (2001), Funaro et al. (2001) Hyvärinen (1999b), and Vigàrio et al. (1998), among others). However, ICA has never been applied to extract the basic components in time series data. In this chapter, we explore the performance of ICA for decomposing multivariate time series data in terms of trend, cycle, and seasonal components.

The ICA model defined in Chapter 1 assumes that the observations are linearly generated by a set of underlying components that are non-Gaussian and statistically independent. This model is quite realistic for being applied in many practical situations. In particular, the classical problem of time series signal extraction, where the time series data are given by the sum of some basic unobserved components, such as trend, cycle, and seasonality, fits to the ICA model formulation.

The motivation for applying ICA to multivariate time series signal extraction is twofold. First, since many multivariate time series data are non-Gaussian distributed, the additive underlying components that generate the data, will be further away from the Gaussian distribution than the original observations. Then, the signal components in time series decomposition, trend, cycle, and seasonal components, should be non-Gaussian (non-linear) distributed. Second, the theory of time series analysis is based on the idea of decomposing a time series into some components of interest, so that each of them has a certain characteristic or type of behavior: the trend reflects the long term behavior, the cycle describes repeated but not periodic fluctuations, and the seasonality is defined as the periodic variations of the time series. Based on this idea, the trend, cycle, and seasonal components should be as independent as possible (it is desirable that they do not share common information). Therefore, ICA seems to be a potential method for signal extraction, where we are looking for trend, cycle, and seasonal components that should be non-Gaussian and mutually independent components.

The main advantage of ICA with respect to existing signal extraction procedures is that it is 'automatic' in the sense that it is able to extract the components without assuming any a-priori structure either in the components nor in the loading matrix. ICA is only based on the assumption of statistical independence, and identifies the signal components as those linear combinations of the data that are maximally independent. In addition, it requires that each of the components explains the largest amount of variance in the data. Thus, if we apply ICA to extract the basic components in multivariate time series data, the estimates for the trend, cycle and seasonal components will be mutually independent. This fact is of great importance in time series signal extraction: on contrast to most of the well-known signal extraction procedures that, in practice, provide correlated components' estimates (despite that the components are assumed to be uncorrelated), ICA obtains mutually independent estimates for the trend, cycle, and seasonal components. Then, ICA can be seen as an 'automatic' procedure for time series decomposition where the ICA components do not share common

information and each of them represent different features of the data. Throughout this chapter, we will explore the idea of presenting ICA as an automatic method for multivariate time signal extraction.

Previous empirical applications proposed in the ICA literature assume that the ICs are stationary stochastic processes. However, our proposal applies ICA to extract the trend, cycle, and seasonal component in multivariate economic time series and some of the components could be non-stationary. Therefore, we propose applying ICA to perform the separation of possible non-stationary components but, does it make sense to think about non-stationary ICA? This is an open question that we will try to explore next.

One of the first approaches to deal with non-stationary unobserved components was proposed by Peña and Poncela (2006b). They present the non-stationary dynamic factor model (DFM) that extends the stationary factor model introduced by Peña and Box (1987) to the non-stationary case. The non-stationary DFM assumes that the dynamic structure of a vector of time series can be explained by a small number of stationary and/or non-stationary latent factors. Peña and Poncela (2006b) define the generalized covariance matrices, $\mathbf{C}_{\mathbf{x}}(k)$, that converges to a random matrix which can be diagonalized. Moreover, since ICA can be seen as dynamic factor model (DFM) with non-linear latent factors (see Section 2.2.3), it may have sense to think about non-stationary ICA. That is, ICA could be seen as a dynamic factor model with non-linear ICs that may be non-stationary. In the simulation experiments of the previous chapter (in particular, in the third experiment) we explore how ICA could deal with non-stationary components, and it seems that it performs quite well. However, from a theoretical point of view, non-stationary ICA is an open question that should be studied deeply.

4.3.1 Forecasting with ICA

In this section, we present the procedure that we will use to forecast multivariate time series data with some components of interest, that are estimated by ICA. This approach was firstly applied by Malaroiu et al. (2000) to forecast financial time series data. The idea is to make the forecasts in the space of the unobserved components, and then transforming back to the observed dataset. The main advantage of this methodology, in comparison to other procedures that also used a small number of factors to forecast large dataset, is that here the components are statistically independent. Then, they can be forecasted separately, fitting

different univariate models for each one of them. In the following, we summarize this three-steps procedure:

1. We apply any ICA algorithm to the observations (it is convenient to choose the algorithm which, a-priori, fits better to the features of the data), and we obtain estimates for both the ICs, $\widehat{\mathbf{s}}_t$, and the loading matrix, $\widehat{\mathbf{A}}$.

2. In this step, we make the ICs forecasts. Since the ICs are statistically independent, they can be modelled separately. Then, we fit a univariate ARIMA$(p,d,q)\times(P,D,Q)_s$ model for each $\widehat{s}_{jt}$, for $j=1,...,r$,

$$(1-\phi_1^{(j)}B-\ldots-\phi_p^{(j)}B^p)\Delta^d\Delta_s^D\widehat{s}_{jt}=(1-\theta_1^{(j)}B-\ldots-\theta_q^{(j)}B^q)a_{jt},\ t=1,...,T. \quad (4.10)$$

   For each ARIMA model, we estimate the parameters and, according to (4.10), the h-step-ahead forecasts for each IC are given by,

$$\widehat{s}_{jT}(h)=E[\widehat{s}_{j(T+h)}|\mathbf{I}_T].$$

3. The forecasts of the observed data set, $\widehat{\mathbf{x}}_T(h)$, are obtained by weighting the ICs forecasts, $\widehat{\mathbf{s}}_T(h)$, with the loading matrix. That is, according to model (**??**),

$$\widehat{\mathbf{x}}_T(h)=\widehat{\mathbf{A}}\widehat{\mathbf{s}}_T(h), \quad (4.11)$$

   or equivalently,

$$\widehat{x}_{it}(h)=\sum_{j=1}^{r}a_{ij}^2\widehat{s}_{jt}(h). \quad (4.12)$$

4.4 Simulation Study

In this section we present some simulation experiments to illustrate the performance of ICA as an automatic procedure in multivariate time series signal extraction. Since PCA is commonly used to estimate the components of interest in large data set (see Chapter 1), we will also apply PCA to the simulations in order to compare the performance of the two methodologies. We design four simulation experiments where the components are generated by the two different unobserved components formulations: whereas in two experiments the components are defined according to Harvey's structural model (Harvey (1989)), in the other two, they follow the dynamic harmonic regression specifications (Young et al. (1999)). For each experiment, we

generate $R = 1000$ realizations, and the components are generated with three different sample sizes, $T = 150, 300, 500$.

The procedure to design the Monte Carlo experiments is similar to the one presented in Chapters 2 and 3: once the m components are generated, they are mixed according to basic ICA model to obtain the observations, $\mathbf{x}_t$. Then, the unobserved components are estimated using PCA and the three ICA procedures considered in Chapter 3 (JADE, SOBI, and FOTBI). The performance of each procedure is analyzed by computing the correlation coefficient and the MSE between the original and the estimated components.

Table 4.1: Definition of the unobserved components-structural time series components (Harvey (1989)) in the Monte Carlo simulation experiments. The components are defined according to models 4.3, 4.5, and 4.4 for the trend, seasonal, and cyclical components, respectively.

Experiment 1: m=6 monthly time series	
$s_{1t} \sim$ LLT trend	$\sigma_\eta^2 = 7.49 \times 10^{-4}$, $\sigma_\xi^2 = 2.75 \times 10^{-6}$
$s_{2t} \sim$ seasonal component	s=12 (monthly seasonality), $\rho^\psi = 1$, $\sigma_\kappa^2 = 0.0109$
$s_{3t} \sim$ cyclical component	$\lambda^c = \frac{2\pi}{72}$ (7-years cycle), $\rho^\psi = 0.9$, $\sigma_\kappa^2 = 0.0278$
$s_{4t} \sim$ AR(1)	$\phi_1 = 0.7$, $n_{4t} \sim t_9$
$s_{5t} \sim$ AR(2)	$\phi_1 = 0.6$, $\phi_2 = -0.2$, $n_{5t} \sim U(0,1)$
$s_{6t} \sim$ irregular component	$s_{6t} \sim t_5$
Experiment 2: m=7 quarterly time series	
$s_{1t} \sim$ RW trend	$\sigma_\eta^2 = 0.0515$, $\sigma_\xi^2 = 0$
$s_{2t} \sim$ seasonal component	s=4 (quarterly seasonality), $\rho^\psi = 1$, $\sigma_\kappa^2 = 0.8$
$s_{3t} \sim I(1)_4$	$s_{3t} = s_{3t-4} + n_{3t}$, $n_{3t} \sim N(0,1)$
$s_{4t} \sim$ cyclical component	$\lambda^c = \frac{2\pi}{16}$ (4-years cycle), $\rho^\psi = 0.75$, $\sigma_\kappa^2 = 0.25$
$s_{5t} \sim$ AR(2)	$\phi_1 = 0.5$, $\phi_2 = 0.35$, $n_{5t} \sim t_9$
$s_{6t} \sim$ irregular component	$s_{6t} \sim U(0,1)$
$s_{7t} \sim$ irregular component	$s_{7t} \sim N(0,1)$

(*) s is the number of seasonal frequencies in a period.

First, we consider the two Monte Carlo experiments where the components follow the Harvey's structural time series approach. Experiments 1 and 2 are defined in Table 4.1 (see the loading matrix for each experiment in Table 4.10 in the appendix). Table 4.2 presents the average results (measured over the m components) for the correlation coefficients and the MSE between the original and the corresponding estimated component. (Table 4.11 in the appendix shows the results for each individual component). We see that the results for the two experiments are quite similar: independently of the sample size, PCA has the worst signal extraction performance overall the procedures. It is specially significant the

value of the MSE of PCA (around 0.52 and 0.73 in Experiments 1 and 2, respectively) that doubles, and sometimes triples, the values of the MSEs of the ICA procedures. Moving on the ICA procedures, independently of the sample size, FOTBI provides better unobserved components estimates than JADE and SOBI do (see Table 4.2). The performance of JADE and SOBI depends on T. Whereas SOBI performs better (or quite similar) than JADE for small sample sizes ($T = 150$), when the sample size increases ($T = 300, 500$) JADE estimates the components more accurately than SOBI. As we explained in Chapter 3, this is because higher-order methods (as JADE and FOTBI), in order to reduce the variance associated to their estimates, requires longer data sets than the second-order methods (as SOBI and PCA). Supporting this argument, whereas the values of correlation coefficients and the MSE for SOBI and PCA are quite similar for all T, the performance of JADE and FOTBI improves when the sample size increases (see Table 4.2).

Table 4.2: Unobserved components-Harvey's simulation experiments: comparison of the mean average of the correlation coefficients and the MSE between the original and the estimated components by PCA, JADE, FOTBI, and SOBI, measured over the m components. $\text{Corr}(\cdot) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{R}\sum_{r=1}^{R}\text{Corr}(s_{it}^{(\cdot)}, \widehat{s_{it}^{(\cdot)}})$
$\text{MSE}(\cdot) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{R}\sum_{r=1}^{R}\text{MSE}(s_{it}^{(\cdot)}, \widehat{s_{it}^{(\cdot)}})$

Experiment 1

	T=150		T=300		T=500	
	Corr	MSE	Corr	MSE	Corr	MSE
PCA	0.7264	0.5436	0.7407	0.5169	0.7492	0.5006
JADE	0.7798	0.4390	0.8433	0.3128	0.8681	0.2634
FOTBI	**0.8761**	**0.2471**	**0.9231**	**0.1535**	**0.9375**	**0.1248**
SOBI	0.8204	0.3579	0.8241	0.3513	0.8266	0.3465

Experiment 2

	T=150		T=300		T=500	
	Corr	MSE	Corr	MSE	Corr	MSE
PCA	0.6290	0.7371	0.6304	0.7367	0.6308	0.7370
JADE	0.7918	0.4151	0.8458	0.3080	0.8692	0.2612
FOTBI	**0.8537**	**0.2917**	**0.8903**	**0.2190**	**0.9105**	**0.1787**
SOBI	0.7818	0.4349	0.7895	0.4204	0.7970	0.4056

Next, we focus on the two Monte Carlo experiments where the components of interest are generated as dynamic harmonic regression components. These experiments are defined in Table 4.3. For the first dynamic harmonic regression experiment (Experiment 3), we generate the four basic components In Experiment 4, we would like to investigate how PCA and ICA procedures separate two periodic components with weekly and monthly periodicity.

Table 4.3: Definition of the dynamic harmonic regression components in the Monte Carlo simulation experiments. The components are defined according to models 4.7 for the trend, and 4.8 for the periodic components.

Experiment 3: m=4 monthly time series	
$s_{1t} \sim$ SRW trend	$0 < \alpha < 1$ randomly generated, $\beta = \gamma = 1$, $\delta = 0$, $\sigma^2_\xi = 0.0015$
$s_{2t} \sim$ periodic component	p=12 $a_t, b_t \sim$ RW, $\sigma^2_{\eta_a} = 0.01$, $\sigma^2_{\eta_b} = 0.0005$
$s_{3t} \sim$ periodic component	p=60 $a_t, b_t \sim$ RW, $\sigma^2_{\eta_a} = 3$, $\sigma^2_{\eta_b} = 12$
$s_{4t} \sim$ irregular component	$s_{4t} \sim U(0,1)$
Experiment 4: m=5 daily time series	
$s_{1t} \sim$ IRW trend	$\alpha = \beta = \gamma = 1$, $\delta = 0$, $\sigma^2_\xi = 0.00035$
$s_{2t} \sim$ periodic component	p=7 $a_{jt}, b_{jt} \sim$ RW, $\sigma^2_{\eta_a} = 0.1$, $\sigma^2_{\eta_b} = 0.05$
$s_{3t} \sim$ periodic component	p=30 $a_t, b_t \sim$ RW, $\sigma^2_{\eta_a} = 3$, $\sigma^2_{\eta_b} = 12$
$s_{4t} \sim$ AR(5)	$\phi_1 = 0.2$, $\phi_2 = 0.5$, $\phi_3 = -0.11$, $\phi_4 = 0.01$, $\phi_5 = 0.005$
$s_{5t} \sim$ irregular component	$s_{5t} \sim U(0,1)$

(*) p denotes the periodicity

The loading matrices of each experiment are in the appendix (see Table 4.10). Table 4.4 presents the average results for both measures, the correlation coefficients and the MSEs. The conclusions from the dynamic harmonic regression experiments are similar to those obtained from Experiments 1 and 2: PCA and FOTBI have, respectively, the worst and the best performance to extract the components of interest. Comparing Tables 4.2 and 4.4, we see that the three ICA procedures provide more accurate estimates for the dynamic harmonic regression components than for the structural time series Harvey's components.

According to the results, any of the three ICA procedures which have been considered here, provides better estimates of the trend, cycle and seasonal components than PCA does. Moreover, within the ICA procedures, we conclude that FOTBI outperforms JADE and SOBI algorithms. These results are as we expected. On the one hand, since PCA estimates the components by maximizing the total variance of the observations, the first PC will increase its percentage of explained variability by mixing the trend and the peaks of seasonality. Then, PCA cannot separate the trend, seasonal, and cyclical components from a vector of time series. On the other hand, since the signals in previous experiments are clearly non-linear and have a significant autocorrelation structure, FOTBI will provide more reliable component estimates than the other two ICA procedures do. In addition, as in previous experiments, the performance of PCA and SOBI does not depend on the sample size, whereas the performance of JADE and FOTBI improves when T increases.

Table 4.4: Unobserved components-dynamic harmonic regression simulation experiments: comparison of the mean average of the correlation coefficients and the MSE between the original and the estimated components by PCA, JADE, FOTBI, and SOBI, measured over the m components. $\text{Corr}(\cdot) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{R}\sum_{r=1}^{R}\text{Corr}(s_{it}^{(\cdot)}, \widehat{s_{it}^{(\cdot)}})$ $\text{MSE}(\cdot) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{R}\sum_{r=1}^{R}\text{MSE}(s_{it}^{(\cdot)}, \widehat{s_{it}^{(\cdot)}})$

Experiment 3

	T=150		T=300		T=500	
	Corr	MSE	Corr	MSE	Corr	MSE
PCA	0.6477	0.7000	0.6591	0.6795	0.6598	0.6791
JADE	0.9299	0.1397	0.9555	0.0889	0.9609	0.0782
FOTBI	**0.9721**	**0.0555**	**0.9859**	**0.0281**	**0.9879**	**0.0242**
SOBI	0.9083	0.1828	0.9139	0.1718	0.9167	0.1663

Experiment 4

	T=150		T=300		T=500	
	Corr	MSE	Corr	MSE	Corr	MSE
PCA	0.7355	0.5254	0.7286	0.5410	0.7231	0.5527
JADE	0.9470	0.1058	0.9661	0.0678	0.9716	0.0568
FOTBI	**0.9573**	**0.0851**	**0.9789**	**0.0422**	**0.9831**	**0.0338**
SOBI	0.8521	0.2948	0.8590	0.2816	0.8612	0.2773

4.5 Empirical application

In this section we apply the ICA methodology to extract the signal in a set of economic time series. First, we introduce the data and describe the estimates of the components obtained by different ICA algorithms. Then, we evaluate the forecasting performance of those different estimation procedures to predict the industrial production index of each country.

4.5.1 Data and components estimates

We consider the industrial production indexes (IPI) in four European countries: France, Germany, Spain, and Italy. They represent the four main economies of the Euro Area, and in all of them, the IPI is a highly quality indicator of their industrial activity. The data are monthly time series from the period January 1975 to October 2010 (430 monthly observations). Then, we have a 4×430 vector of time series, which is denoted by $\mathbf{y}_t$. We transform the dataset taking logs and subtracting the mean from the observations:

$$\mathbf{x}_t = \log(\mathbf{y}_t) - \overline{\log(\mathbf{y}_t)}.$$

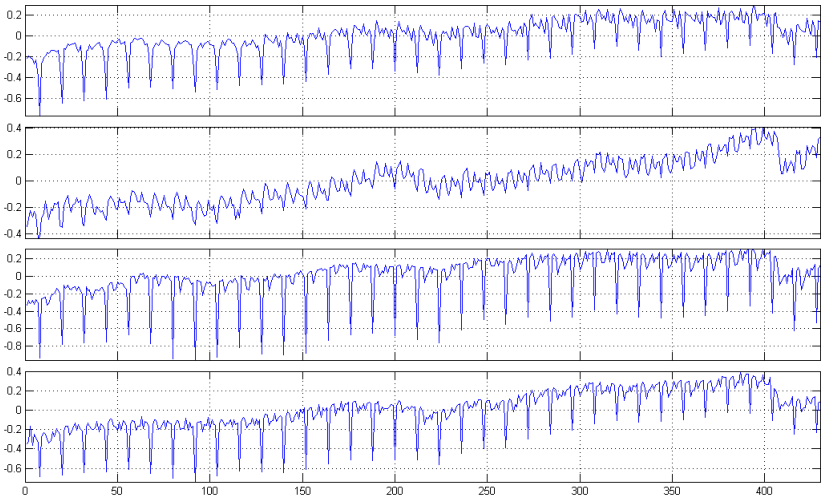


Figure 4.1: Series of 4 monthly IPI time series from 1975:01 to 2010:10 (France, Germany, Spain, and Italy)

The IPI time series (in logs) are shown in Figure 4.1. They are clearly non-stationary time series which are characterized by strong trend and seasonality patterns. Our aim is to extract those relevant features and isolate the less interesting ones. For this purpose, we will apply

PCA and ICA, which extract the underlying signals directly from the observations, without assuming any a-priori model for the components of interest. Thus, we could compare the PCs and the ICs components estimates.

To motivate the use of ICA in our data, we compute the Jarque-Bera skewness-kurtosis statistics of $\mathbf{x}_t$ to test for normality on each individual series. The results, which are displayed in the Table 4.5, show that the null hypothesis of normality is rejected at the 1% significance level for each time series. Therefore, since the dataset is non-Gaussian distributed, it is reasonable applying ICA to extract the interesting features from the data.

Table 4.5: Jarque-Bera skewness-kurtosis statistic of the IPI (in logs)

	France	**Germany**	**Italy**	**Spain**
Jarque-Bera	191.3013	7.9463	416.5863	53.9601
p-value	(0.0001)	(0.0233)	(0.0001)	(0.0001)

We apply JADE, SOBI and FOTBI, presented in Chapters 2 and 3, to extract the unobserved signal from the observations. These three ICA procedures decompose the multivariate time series data into a set of approximately independent components, but none of them provide a formal criterion to sort the ICs and to identify the more relevant ones. In this empirical application, the interest is to separate the trend (or trend-cycle) and the seasonal component of the IPIs time series. Since these patterns explain most of the variance of the observations and PCA sorts the components in terms of the total explained variability, we will use PCA as an intermediate step in the ICA signal extraction procedures. Thus, our proposal can be summarized by the following steps:

1. Applying PCA to the data and choose the optimal number of PCs, r, that depends on the percentage of the total variance that we would like to be explained. In time series signal extraction, usually no more that two or three components are selected.

2. Applying any ICA algorithm to the data to extract the m ICs.

3. Computing the correlation between the PCs and the ICs, and sorting the ICs according to the maximum correlation criterion. That is, for each $i = 1, ..., m$, the i-th IC satisfies:

$$\max_{1 \leq j \leq m} corr(\widehat{s}_{it}^{PCA}, \widehat{s}_{jt}^{ICA}). \tag{4.13}$$

Thus, the first IC will be the component that is maximally correlated to the first PC,

the second IC will have maximum correlation to the second PC, and so on. Once the ICs are sorted, we could select the r ICs that provides the estimates for the underlying signals.

Applying previous procedure for our data, $\mathbf{x}_t$, we firstly estimate the four PCs that are sorted in terms of the total explained variability. From Table 4.6 we have that the two first PCs explain almost the 98% of total variability, so we can fix r equal to two. Second, we estimate the four ICs using JADE, FOTBI, and SOBI. Then, we compute the correlation between the PCs and the different ICA components, and sort them according to the criterion (4.13). In Table 4.7 we report the value of the correlation coefficients between the two first PCs and the corresponding ICs. In this particular example, the two SOBI and FOTBI ICs that have been selected, correspond to the two first ICs which were given automatically by those ICA algorithms (for the JADE ICs the order is not preserved). However, this fact cannot be generalized to any empirical application.

Table 4.6: Individual and accumulate percentage of variability explained by the PCs

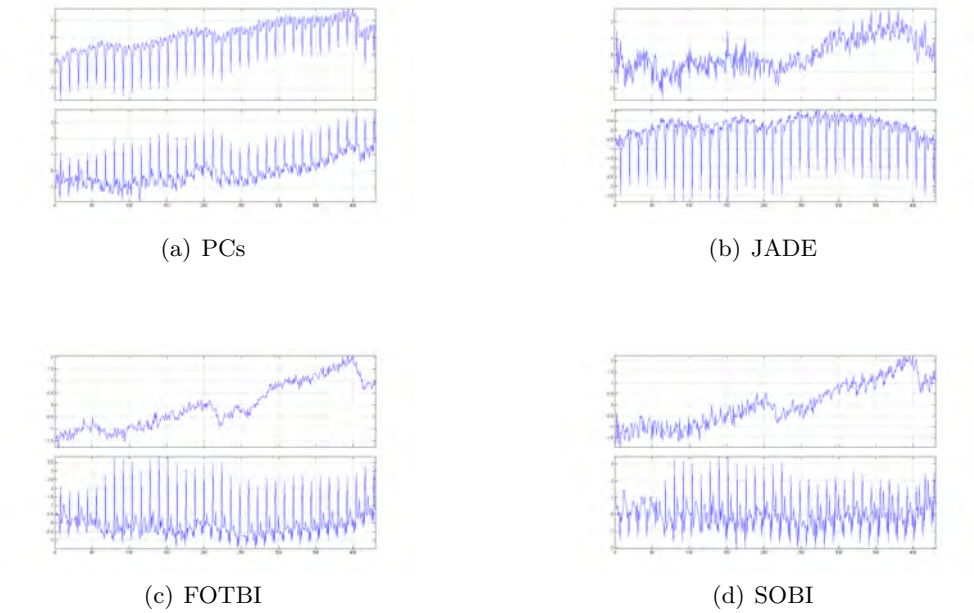
	$\widehat{s}_{1t}^{PCA}$	$\widehat{s}_{2t}^{PCA}$	$\widehat{s}_{3t}^{PCA}$	$\widehat{s}_{4t}^{PCA}$
% Variability	89.21	8.57	1.26	0.96
% Accumulate Variability	89.21	97.78	99.04	100

Table 4.7: Correlation coefficients between the two first PCs and the corresponding ICs which are maximally correlated to each of them.

	$\widehat{s}_{jt}^{JADE}$	$\widehat{s}_{jt}^{FOTBI}$	$\widehat{s}_{jt}^{SOBI}$
$\widehat{s}_{1t}^{PCA}$	0.4690	0.7346	0.7034
$\widehat{s}_{2t}^{PCA}$	0.6774	0.7398	0.5937

The PCs and the ICs that represent the relevant patterns of the IPI time series data are shown on Figure 4.2. The desirable results would provide estimates for the trend in the first component of interest, and estimates for the seasonal component in the second one. However, as we can see in Figure 4.2, the results are not very convincing, specially those corresponding to the first component estimates. Just by graphical inspection of the estimated components, it is clear that PCA is not able to separate the trend and the seasonal component. The first PC is a mixture of the trend and seasonality patterns; the second one is dominated by accentuated seasonality but some evidences of the trend component still remain. According

to the PCA optimization criterion, that looks for the components that maximize the total variability, those are the expected results.



(a) PCs (b) JADE

(c) FOTBI (d) SOBI

Figure 4.2: The two estimated components that have been selected for each procedure. We have the PCs in Figure 4.2(a), the JADE componentes in Figure 4.2(b), the FOTBI components in Figure 4.2(c), and the SOBI ones in Figure 4.2(d).

Although the results of the three ICA algorithms for our data are quite different, it seems that ICA provide more encouraging results than PCA for signal extraction purposes. The differences among the ICs extracted by JADE, FOTBI, and SOBI are due to the different estimation principle used by each procedure. On the one hand, JADE does not take into account the time structure of the data and it have the worse performance for the IPIs signal extraction: JADE cannot separate the trend and the seasonal patterns, and the two components are mixed in the first JADE IC (see Figure 4.2(b)). On the other hand, FOTBI and SOBI exploit the autocorrelation structure of the observations, and they would provide, a-priori, better estimates for the trend and the seasonal components than JADE and PCA do. In addition, since economic time series are usually non-Gaussian and the trend and seasonality are non-linear components, FOTBI seems to be more appropriate than SOBI for the IPI time series signal extraction. Figures 4.2(c)-4.2(d) confirm this fact: the first FOTBI IC seems to provide the most reliable estimate for the trend component overall the estimated ICs. The second best performance is given by SOBI, where the first SOBI IC still exhibits some evidence of seasonality, although it is less accentuated than in the first component given

by JADE and PC. Then, FOTBI can be seen as a first step for an automatic multivariate signal extraction procedure.

In the following subsection, we will analyze how the IPIs of four European countries can be forecasted using the underlying signals extracted by the previous procedures.

4.5.2 Forecasting results

The IPI is usually published with some significant delay, and this fact motivates the interest in providing accurate forecasts. Here, we analyze the forecasting performance of PCA, JADE, FOTBI, and SOBI, to predict the IPI of the four main European countries, using the first two components estimated by each procedure. We use simple univariate ARIMA models for the IPI of each country as benchmark models. We compute the forecasts at different time horizons, $h = 1, 3, 6, 12$ steps ahead. We apply a three-step iterative forecasting procedure: estimating the components of interest using the whole sample (as it is explained in previous section), making forecasts in the space of the components, and transforming back to the original data set.

Since the ICs are statistically independent, they can be forecasted separately fitting a different model for each IC. Then, to compute the forecasts for the components, we first apply the automatic procedure of TRAMO/SEATS program to fit univariate ARIMA$(p, d, q) \times (P, D, Q)_s$ models to the components (since ICA and PCA are automatic procedures, we decide to use the automatic specification given by the TRAMO/SEATS program). For each component, we estimate the univariate ARIMA model using observations from 1975:01 to 2007:10 (see Table 4.13 in the appendix for a detailed description of the models), and compute the $h = 1, 3, 6, 12$ steps ahead forecasts. This procedure is repeated following a rolling window approach. That is, after getting the first set of h-steps ahead forecasts ($h = 1, 3, 6, 12$) for each component, the estimation sample is extended by one further observation, the parameters of the corresponding ARIMA model are re-estimated each time (keeping constant the automatic specification for the ARIMA models thought the procedure), and the new $1-, 3-, 6-$, and 12-monthly steps ahead forecasts are built recursively until the end of the sample.

Then, we have computed the forecasts for the components of interest and we will use them to predict the IPIs time series. By (4.11), the h-steps ahead forecasts for the IPI of each country can be obtained just weighting the univariate forecasts of the components by the

corresponding loading matrix coefficients. That is, for the IPIs, a sequence of h-steps ahead forecasts for $h = 1, 3, 6, 12$ is performed by:

$$\widehat{\mathbf{x}}_{t_0}(h) = \widehat{\mathbf{A}}\widehat{\mathbf{s}}_{t_0}(h), \quad h = 1, 3, 6, 12,\ t_0 = 2007:10, \ldots, 2010:10 - h,$$

or equivalently, $\widehat{x}_{it_0}(h) = \sum_{j=1}^{2} a_{ij}^2 \widehat{s}_{jt_0}(h)$, for $i = 1, \ldots, 4$.

In order to evaluate the accuracy of each procedure to forecast the IPIs time series, we compare the $h - step - ahead$ prediction error associated to each method, given by:

$$e_{it_0} = x_{it_0+h} - \widehat{x}_{it_0}(h), \quad i = 1, \ldots, 4,\ h = 1, 3, 6, 12,\ t_0 = 2007:10, \ldots, 2010:10 - h,$$

to the one associated to some benchmark model. Here, the benchmark models will be the univariate ARIMA models fitted to each IPI time series using the automatic TRAMO/SEATS identification procedure. To compute the h-steps ahead forecasts and prediction errors associated to the benchmark models, we apply the same recursive procedure that we used to obtain the forecasts of the components.

To analyze the forecasting performance of PCA and ICA procedures with respect to the benchmark model, we propose to measure the forecasting accuracy of each procedure by the following criteria (see Hyndman and Koehler (2006) for a complete revision of measures of forecast accuracy). For each $i = 1, \ldots, 4$, and $h = 1, 3, 6, 12$,

1. Root Mean Squared Error: $RMSE_{ih} = \sqrt{\sum_{t=1}^{36-h+1} e_{it}^2}$.
2. Mean Absolute Percentage Error: $MAPE_{ih} = \sqrt{\sum_{t=1}^{36-h+1} |p_{it}|}$, where $p_{it} = \frac{e_{it}}{x_{it}}$.
3. Mean Absolute Scale Error: $MASE_{ih} = \sqrt{\sum_{t=1}^{36-h+1} |q_{it}|}$, where $q_{it} = \frac{e_{it}}{\frac{1}{t-1}\sum_{l=2}^{t-1} |x_{il} - x_{il-1}|}$.
4. Geometric Mean Absolute Error: $GMAE_{ih} = geomean(|e_{it}|)$.

We consider the relative values of the four criteria: RelRMSE, RelMAPE, RelMASE, and RelGMAE. That is, we use the ratios of the corresponding criterion for PCA, JADE, FOTBI, and SOBI, with respect to the corresponding one for the benchmark model (the value of each criterion for the univariate ARIMA models):

$$RelRMSE_{(\cdot)} = \frac{RMSE_{(\cdot)}}{RMSE_{benchmark}}; \quad RelMAPE_{(\cdot)} = \frac{MAPE_{(\cdot)}}{MAPE_{benchmark}}$$

$$RelMASE_{(\cdot)} = \frac{MASE_{(\cdot)}}{MASE_{benchmark}}; \quad RelGMAE_{(\cdot)} = \frac{GMAE_{(\cdot)}}{GMAE_{benchmark}}$$

Table 4.8 shows the average results for the relative criteria, measured overall the IPIs of the four European countries, at different time horizons, $h = 1, 3, 6, 12$. We obtain similar results, independently of the criterion used to evaluate the forecasting performance of the different procedures. The forecasting performance of the PCA and ICA procedures, with respect to the univariate one, depends on the time horizon, h. It is known that the univariate models produce quite accurate short-term forecasts ($h = 1, 3$), but not in the medium- and long-term. This fact is pointed out in our results, where the forecasting performance of the PCA and ICA procedures, relative to the univariate models performance, improves when h increases (Table 4.8).

Table 4.8: Relative values of the different criteria for each of the procedures (Univariate=1). The results represent the average values measured over the IPIs of the four main European countries: France, Germany, Italy, and Spain.

		RelRMSE	RelMAPE	RelMASE	RelGMAE
h=1	PCA	1.7982	1.1469	2.0040	1.7410
	JADE	3.7799	1.8392	4.6462	4.5830
	FOTBI	1.0439	**0.9561**	1.0462	**0.9934**
	SOBI	2.6876	2.0061	2.4041	2.5748
	UNIV	**1.0000**	1.0000	**1.0000**	1.0000
h=3	PCA	1.1509	0.9665	1.2660	1.1772
	JADE	2.1590	0.8465	2.5701	2.5803
	FOTBI	**0.9827**	**0.8261**	1.0176	**0.9834**
	SOBI	1.7867	1.3246	1.6353	1.8116
	UNIV	1.0000	1.0000	**1.0000**	1.0000
h=6	PCA	0.9992	0.9034	1.0033	0.9456
	JADE	1.2020	0.7009	1.3301	1.1830
	FOTBI	**0.8817**	**0.5039**	**0.8895**	**0.7982**
	SOBI	1.3271	1.0703	1.1608	1.0937
	UNIV	1.0000	1.0000	1.0000	1.0000
h=12	PCA	1.0924	1.0362	1.0628	1.0191
	JADE	0.7922	0.7209	0.7657	0.6155
	FOTBI	**0.7897**	**0.4502**	**0.7335**	**0.6145**
	SOBI	1.0757	0.9474	0.8759	0.6713
	UNIV	1.0000	1.0000	1.0000	1.0000

Within the ICA procedures, FOTBI performs better than JADE and SOBI at any time horizon, $h = 1, 3, 6, 12$. However, the forecasting performance of FOTBI in comparison to the univariate ARIMA models (benchmark models) depends on h. In the short-term ($h = 1, 3$), both procedures, FOTBI and univariate models, have similar forecasting performance. They provide more accurate short-term forecasts than PCA, JADE, and SOBI do. In addition,

note that, for $h = 1, 3$, PCA performs better than SOBI, and SOBI performs better than JADE (Table 4.8). In the medium-, and long- term, the results are slightly different. On the one hand, for $h = 6$, FOTBI has the best forecasting performance followed by PCA and the univariate models, which have similar performance and outperform JADE and SOBI. On the other hand, any of the ICA procedures (although the smallest values of the different criteria correspond to FOTBI) provide more accurate long-term forecasts ($h = 12$) than PCA and the benchmark models do. The results for each individual IPI time series are provided in the appendix (see Table 4.14). The conclusions are analogous to the ones explained above for the average results.

According to previous results, our main interest is to compare the forecasting performance of FOTBI and the univariate models. It seems that both procedures have similar forecasting performance in the short-term ($h = 1, 3$), but FOTBI outperforms the univariate models in medium- and long-term forecasting ($h = 6, 12$) (Table 4.8). However, we would like to investigate whether or not these differences are statistically significant applying the Diebold-Mariano test (Diebold and Mariano (1995)), that is used to compare the forecast accuracy of two competing models. Under the 'equal accuracy' null hypothesis of the Diebold-Mariano test, there are no-differences in the predictive accuracy of the two models. In this paper, we carry out the Diebold-Mariano test taking into account two different, squared and absolute error, loss functions. The outputs of the Diebold-Mariano test applied to the average results given in Table 4.8 are presented in Table 4.9. We applied the Diebold-Mariano test to all procedures, two by two, and we report the value of the Diebold-Mariano test statistic, the p-value (between brackets), and the procedure that produces better forecasts in each comparison (= means that the two procedures have equal predictive accuracy). The results of the Diebold-Mariano test to compare the forecast accuracy of the different procedures for each individual IPI time series are in the appendix (see Tables 4.15 and 4.16) These results are consistent to the previous ones.

Summarizing the results given by Tables 4.8 and 4.9, we cannot conclude that there is a procedure which outperforms the others for any time horizons. However, the FOTBI procedure seems to have quite promising performance: it provides similar forecasts than the univariate models do in the short-term ($h = 1, 3$), the best medium-term forecasts ($h = 6$) overall the procedures, and more accurate 12-steps ahead forecasts than PCA and the univariate models (the other ICA procedures, JADE and SOBI, perform equal than FOTBI in the long-run).

Table 4.9: Results of the Diebold-Mariano test carried out to evaluate the forecast accuracy (measured as an average over the four IPIs time series) of the different procedures

	Squared Error Loss Function				Absolute Error Loss Function			
MET A vs MET B	h=1	h=3	h=6	h=12	h=1	h=3	h=6	h=12
PCA vs UNIV	2.4316 (0.0075) **UNIV**	0.3024 (0.3812) =	-0.0021 (0.4992) =	0.0440 (0.4824) =	2.1510 (0.0157) **UNIV**	0.5897 (0.2777) =	-0.0013 (0.4995) =	0.1115 (0.4556) =
JADE vs UNIV	5.8395 (0.0000) **UNIV**	4.0193 (0.0000) **UNIV**	1.1808 (0.1188) =	-2.0155 (0.0347) **JADE**	7.5484 (0.0000) **UNIV**	4.2871 (0.0000) **UNIV**	1.0567 (0.1453) =	-1.9712 (0.0244) **JADE**
FOTBI vs UNIV	-0.0023 (0.4991) =	-0.0032 (0.4987) =	-2.1340 (0.0164) **FOTBI**	-4.4494 (0.0000) **FOTBI**	-0.0284 (0.4887) =	-0.0214 (0.4915) =	-1.9594 (0.0485) **FOTBI**	-3.9709 (0.0000) **FOTBI**
SOBI vs UNIV	3.1194 (0.0009) **UNIV**	2.7339 (0.0031) **UNIV**	1.7258 (0.0522) =	0.4367 (0.3312) =	4.0058 (0.0000) **UNIV**	2.7723 (0.0028) **UNIV**	1.1120 (0.1331) =	-0.9176 (0.1794) =
PCA vs SOBI	-2.2202 (0.0132) **PCA**	-2.4832 (0.0065) **PCA**	-1.8237 (0.0541) =	0.3216 (0.3739) =	-2.3833 (0.0086) **PCA**	-2.2104 (0.0135) **PCA**	-1.1275 (0.1298) =	1.2683 (0.1023) =
JADE vs SOBI	2.6451 (0.0041) **SOBI**	1.5609 (0.0593) =	-0.2912 (0.3854) =	-1.3380 (0.0904) =	3.1134 (0.0009) **SOBI**	1.9802 (0.0375) **SOBI**	0.1126 (0.4552) =	-0.8315 (0.2029) =
FOTBI vs SOBI	-3.2780 (0.0005) **FOTBI**	-3.1143 (0.0009) **FOTBI**	-2.5614 (0.0052) **FOTBI**	-1.9808 (0.0238) **FOTBI**	-4.4839 (0.0000) **FOTBI**	-3.3646 (0.0004) **FOTBI**	-2.0629 (0.0196) **FOTBI**	-1.2128 (0.1126) =
PCA vs FOTBI	2.9578 (0.0015) **FOTBI**	1.2520 (0.1053) =	2.2474 (0.0123) **FOTBI**	4.1526 (0.0000) **FOTBI**	2.8958 (0.0019) **FOTBI**	1.3144 (0.0944) =	1.9822 (0.0299) **FOTBI**	4.5209 (0.0000) **FOTBI**
JADE vs FOTBI	5.9747 (0.0000) **FOTBI**	4.3287 (0.0000) **FOTBI**	2.7698 (0.0028) **FOTBI**	0.0540 (0.4785) =	8.0409 (0.0000) **FOTBI**	4.6709 (0.0000) **FOTBI**	2.4767 (0.0066) **FOTBI**	-0.1718 (0.4318) =
PCA vs JADE	-6.0001 (0.0000) **PCA**	-4.7388 (0.0000) **PCA**	-1.4072 (0.0797) =	2.4965 (0.0063) **JADE**	-7.9074 (0.0000) **PCA**	-5.3897 (0.0000) **PCA**	-1.1535 (0.1243) =	2.4902 (0.0064) **JADE**

4.6 Concluding remarks

In this study we have explored how ICA performs for prediction and signal extraction in multiple non-stationary time series data.

ICA assumes that the observations are linearly generated by a set of underlying components which are statistically independent. It has been traditionally used in different areas of research, such as medical, biological, and engineering applications, where the data are observed with high level of noise. ICA is a powerful technique that is able to extract the underlying components only from the observations, and just by making the assumption of statistically independence on the components.

Here we have applied ICA to multivariate time series data in which the underlying components can be interpreted in terms of trends and seasonality patterns. Most of the procedures (e.g. TRAMO/SEATS, STAMP, and linear dynamic harmonic regression) found in the signal extraction literature, are model-based procedures, developed in the univariate case, that specify directly stochastic linear models either on the observations or on the underlying components. Despite that those procedures are, in general, quite successful, modelling the components a-priori could produce specification problems that culminate in crucial estimation errors.

We present ICA as an alternative methodology for multivariate time series signal extraction. The advantage of ICA with respect to the so called model-based signal extraction procedures relies on the fact that ICA is an automatic procedure that does not specify any a-priori structure either in the data nor in the components. ICA looks for the trend, cycle, and seasonal components by assuming only their statistical independence.

As different ICA algorithms provide different components estimates, we have implemented three different ICA algorithms, JADE, FOTBI, and SOBI, to analyze their performance as automatic signal extraction procedures. We have tested the three ICA procedures on four Monte Carlo simulation experiments, and the results show that FOTBI performs quite well. Then, it seems that the FOTBI procedure could be considered as a first-step for an automatic procedure in multivariate time series signal extraction.

We have empirically assess the ability of PCA and the three different ICA procedures to extract the dynamic relationships among the IPIs of the four main European countries. In this

analysis, the contribution of the paper are two fold. On the one hand, as it was expected, since these data were non-Gaussian and they had a pronounced autocorrelation structure, FOTBI provided the best estimates for the trend and the seasonal components. On the other hand, we have analyzed the forecasting performance of PCA and ICA, using the univariate ARIMA models for the IPIs as benchmark models. We have computed $h = 1, 3, 6, 12$ steps-ahead forecasts for the IPIs and the results are very promising. When we forecast the IPIs using the FOTBI ICs, we have: (i) short-term forecasts ($h = 1, 3$) given by the FOTBI components are similar to the ones obtained by the univariate models (we know that univariate models perform well in short-term forecasting); (ii) in medium-forecasting ($h = 6$), FOTBI outperforms overall the procedures; and (iii) any of the ICA procures (JADE, FOTBI, and SOBI have equally predictive power according to the Diebold-Mariano test) provide more accurate long-term forecasts of the IPIs ($h = 12$) than the benchmark models does.

Then, FOTBI seems to perform quite well for prediction and signal extraction in multivariate time series data, which may be non-stationary.

4.7 Appendix

Table 4.10: Mixing matrices using in the simulation experiments.

	Experiment 1	Experiment 2
A	$\begin{pmatrix} 2 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 1 & 1 & 2 \end{pmatrix}$	$\begin{pmatrix} -4 & 3 & -1 & 1 & -1 \\ 2 & -1 & 1 & 0 & 1 \\ 3 & 1 & -2 & 1 & 0 \\ -1 & -1 & 1 & 1 & 0 \\ -2 & -4 & 3 & 0 & -1 \end{pmatrix}$
	Experiment 3	Experiment 4
A	$\begin{pmatrix} 2 & 1 & -1 & 1 & 0 & 0 \\ 3 & 2 & 2 & 1 & 0 & 1 \\ -2 & 1 & -1 & 0 & 0 & -1 \\ 1 & -1 & 1 & -1 & 0 & 1 \\ 2 & -1 & -1 & 0 & -1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 4 & 3 & -2 & 1 & 1 & 0 & -1 \\ -2 & 1 & 1 & 1 & -1 & 0 & 0 \\ -1 & 1 & -1 & 1 & -1 & 0 & 1 \\ -3 & -2 & 4 & 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & 0 & 0 \\ 2 & 3 & 4 & 0 & -2 & 1 & -1 \\ 0 & 1 & 2 & 3 & -1 & 0 & 2 \end{pmatrix}$

Table 4.11: Unobserved components-Harvey's simulation experiments: comparison of the correlation coefficients and the MSE between the original and the estimated components by PCA, JADE, FOTBI, and SOBI. For each component, these values corresponds to the mean average values measured over the R realizations.
$\text{Corr}(\cdot) = \frac{1}{R}\sum_{r=1}^{R}\text{Corr}(s_{it}^{(\cdot)}, \widehat{s_{it}^{(\cdot)}})$; $\text{MSE}(\cdot) = \frac{1}{R}\sum_{r=1}^{R}\text{MSE}(s_{it}^{(\cdot)}, \widehat{s_{it}^{(\cdot)}})$

Experiment 1

		Correlation Coefficient				MSE			
T	$\mathbf{s}_t$	PCA	JADE	FOTBI	SOBI	PCA	JADE	FOTBI	SOBI
T=150	s_{1t}	0.8839	0.8614	0.9504	0.8374	0.2307	0.2762	0.0989	0.3240
	s_{2t}	0.7689	0.8366	0.9161	0.8419	0.4592	0.3257	0.1672	0.3151
	s_{3t}	0.7068	0.7212	0.8295	0.8150	0.5825	0.5558	0.3398	0.3687
	s_{4t}	0.6822	0.7110	0.8245	0.7940	0.6313	0.5760	0.3499	0.4106
	s_{5t}	0.6343	0.7026	0.8301	0.8009	0.7266	0.5928	0.3388	0.3969
	s_{6t}	0.6823	0.8457	0.9058	0.8334	0.6312	0.3077	0.1879	0.3321
T=300	s_{1t}	0.8813	0.9165	0.9627	0.8337	0.2366	0.1667	0.0745	0.3320
	s_{2t}	0.7785	0.8992	0.9769	0.8457	0.4416	0.2013	0.0462	0.3082
	s_{3t}	0.7391	0.7862	0.8743	0.8240	0.5201	0.4270	0.2509	0.3514
	s_{4t}	0.6937	0.7649	0.8708	0.7928	0.6106	0.4695	0.2580	0.4138
	s_{5t}	0.6361	0.7671	0.8988	0.8076	0.7253	0.4651	0.2021	0.3842
	s_{6t}	0.7154	0.9262	0.9554	0.8406	0.5673	0.1473	0.0891	0.3183
T=500	s_{1t}	0.8817	0.9407	0.9651	0.8405	0.2362	0.1185	0.0698	0.3187
	s_{2t}	0.7870	0.9217	0.9887	0.8498	0.4252	0.1565	0.0226	0.3002
	s_{3t}	0.7534	0.8075	0.8895	0.8221	0.4923	0.3847	0.2208	0.3554
	s_{4t}	0.6975	0.7878	0.8907	0.7985	0.6038	0.4241	0.2185	0.4027
	s_{5t}	0.6377	0.7932	0.9215	0.8087	0.7231	0.4131	0.1568	0.3822
	s_{6t}	0.7380	0.9581	0.9698	0.8398	0.5229	0.0837	0.0604	0.3200

Experiment 2

		Correlation Coefficient				MSE			
T	$\mathbf{s}_t$	PCA	JADE	FOTBI	SOBI	PCA	JADE	FOTBI	SOBI
T=150	s_{1t}	0.6670	0.8043	0.9112	0.8517	0.6615	0.3900	0.1770	0.2956
	s_{2t}	0.6651	0.8591	0.9026	0.7820	0.6653	0.2808	0.1941	0.4345
	s_{3t}	0.5047	0.6967	0.7735	0.7301	0.9841	0.6045	0.4514	0.5380
	s_{4t}	0.5431	0.7732	0.8462	0.8329	0.9077	0.4521	0.3066	0.3332
	s_{5t}	0.5718	0.7166	0.8153	0.7786	0.8508	0.5650	0.3682	0.4412
	s_{6t}	0.8290	0.8832	0.8411	0.7317	0.3398	0.2329	0.3168	0.5349
	s_{7t}	0.6222	0.8092	0.8857	0.7658	0.7506	0.3803	0.2279	0.4669
T=300	s_{1t}	0.6545	0.8432	0.9298	0.8502	0.6888	0.3130	0.1401	0.2991
	s_{2t}	0.6607	0.8786	0.9097	0.7889	0.6764	0.2423	0.1802	0.4215
	s_{3t}	0.4950	0.7192	0.7802	0.7410	1.0066	0.5607	0.4390	0.5172
	s_{4t}	0.5381	0.8373	0.8982	0.8320	0.9206	0.3248	0.2032	0.3355
	s_{5t}	0.5802	0.7772	0.8708	0.7852	0.8369	0.4448	0.2579	0.4289
	s_{6t}	0.8419	0.9700	0.9067	0.7498	0.3151	0.0599	0.1862	0.4995
	s_{7t}	0.6424	0.8947	0.9368	0.7792	0.7127	0.2103	0.1262	0.4408
T=500	s_{1t}	0.6448	0.8619	0.9443	0.8570	0.7089	0.2759	0.1113	0.2858
	s_{2t}	0.6513	0.8804	0.9115	0.7994	0.6959	0.2390	0.1768	0.4008
	s_{3t}	0.4896	0.7257	0.7886	0.7368	1.0188	0.5481	0.4224	0.5258
	s_{4t}	0.5353	0.8674	0.9292	0.8418	0.9274	0.2649	0.1414	0.3162
	s_{5t}	0.5878	0.8262	0.9015	0.7955	0.8227	0.3472	0.1968	0.4087
	s_{6t}	0.8572	0.9869	0.9395	0.7602	0.2850	0.0262	0.1210	0.4791
	s_{7t}	0.6493	0.9363	0.9593	0.7884	0.7000	0.1273	0.0814	0.4227

Table 4.12: Unobserved components-dynamic harmonic regression simulation experiments: comparison of the correlation coefficients and the MSE between the original and the estimated components by PCA, JADE, FOTBI, and SOBI. For each component, these values corresponds to the mean average values measured over the R realizations. $\text{Corr}(\cdot) = \frac{1}{R}\sum_{r=1}^{R}\text{Corr}(s_{it}^{(\cdot)}, \widehat{s_{it}^{(\cdot)}})$; $\text{MSE}(\cdot) = \frac{1}{R}\sum_{r=1}^{R}\text{MSE}(s_{it}^{(\cdot)}, \widehat{s_{it}^{(\cdot)}})$

Experiment 3

		Correlation Coefficient				MSE			
T	$\mathbf{s}_t$	PCA	JADE	FOTBI	SOBI	PCA	JADE	FOTBI	SOBI
T=150	s_{1t}	0.7129	0.9250	0.9602	0.9061	0.5703	0.1496	0.0794	0.1871
	s_{2t}	0.7235	0.9362	0.9807	0.9285	0.5493	0.1271	0.0385	0.1425
	s_{3t}	0.5923	0.9586	0.9766	0.8842	0.8100	0.0825	0.0467	0.2309
	s_{4t}	0.5620	0.8998	0.9711	0.9144	0.8702	0.1997	0.0576	0.1705
T=300	s_{1t}	0.6998	0.9517	0.9777	0.9112	0.5983	0.0965	0.0446	0.1774
	s_{2t}	0.7044	0.9558	0.9938	0.9289	0.5893	0.0882	0.0125	0.1420
	s_{3t}	0.6252	0.9768	0.9847	0.8970	0.7470	0.0464	0.0306	0.2056
	s_{4t}	0.6070	0.9375	0.9876	0.9187	0.7834	0.1247	0.0248	0.1623
T=500	s_{1t}	0.7065	0.9583	0.9807	0.9104	0.5859	0.0833	0.0385	0.1789
	s_{2t}	0.6711	0.9552	0.9937	0.9282	0.6565	0.0894	0.0127	0.1435
	s_{3t}	0.6547	0.9815	0.9854	0.9067	0.6891	0.0369	0.0292	0.1864
	s_{4t}	0.6067	0.9484	0.9917	0.9216	0.7850	0.1030	0.0165	0.1566

Experiment 4

		Correlation Coefficient				MSE			
T	$\mathbf{s}_t$	PCA	JADE	FOTBI	SOBI	PCA	JADE	FOTBI	SOBI
T=150	s_{1t}	0.7848	0.9568	0.9814	0.8814	0.4276	0.0861	0.0370	0.2365
	s_{2t}	0.6322	0.9423	0.9704	0.8691	0.7307	0.1151	0.0590	0.2609
	s_{3t}	0.5904	0.9786	0.9854	0.8457	0.8138	0.0426	0.0292	0.3075
	s_{4t}	0.9696	0.8818	0.9176	0.8317	0.0603	0.2356	0.1643	0.3356
	s_{5t}	0.7007	0.9753	0.9318	0.8327	0.5945	0.0493	0.1359	0.3335
T=300	s_{1t}	0.7656	0.9666	0.9862	0.8814	0.4672	0.0667	0.0276	0.2369
	s_{2t}	0.6173	0.9576	0.9873	0.8714	0.7629	0.0847	0.0254	0.2567
	s_{3t}	0.5919	0.9816	0.9899	0.8557	0.8134	0.0367	0.0202	0.2881
	s_{4t}	0.9810	0.9308	0.9554	0.8390	0.0378	0.1382	0.0890	0.3214
	s_{5t}	0.6871	0.9937	0.9756	0.8473	0.6236	0.0126	0.0486	0.3050
T=500	s_{1t}	0.7634	0.9680	0.9857	0.8746	0.4723	0.0639	0.0285	0.2505
	s_{2t}	0.6048	0.9605	0.9843	0.8701	0.7889	0.0790	0.0314	0.2596
	s_{3t}	0.5936	0.9820	0.9881	0.8710	0.8111	0.0359	0.0237	0.2578
	s_{4t}	0.9865	0.9504	0.9705	0.8391	0.0270	0.0991	0.0590	0.3214
	s_{5t}	0.6671	0.9969	0.9867	0.8513	0.6645	0.0062	0.0265	0.2971

Table 4.13: Univariate models for the different components and for the IPI time series data. The optimal specification for the $ARIMA(p,d,q) \times (P,D,Q)_s$ models as well as the estimation of the parameters are carried out by the TRAMO/SEATS automatic identification procedure.

	1^{st} component	2^{nd} component
PCA	ARIMA(3,1,1)×$(0,1,1)_{12}$	ARIMA(0,1,1)×$(1,1,1)_{12}$
	$(1 - \underset{(0.07)}{0.20}\,B^2 - \underset{(0.05)}{0.23}\,B^3)\Delta\Delta_{12}s_{1t}^{PCA} = (1 + \underset{(0.11)}{0.77}\,B)(1 + \underset{(0.05)}{0.41}\,B^{12})a_{1t}^{PCA}$	$(1 - \underset{(0.04)}{0.41}\,B^{12})\Delta\Delta_{12}s_{2t}^{PCA} = (1 + \underset{(0.05)}{0.67}\,B)(1 + \underset{(0.00)}{0.89}\,B^{12})a_{2t}^{PCA}$
JADE	ARIMA(0,1,1)×$(0,1,1)_{12}$	ARIMA(0,1,2)×$(0,1,1)_{12}$
	$\Delta\Delta_{12}s_{1t}^{JADE} = (1 + \underset{(0.04)}{0.69}\,B)(1 + \underset{(0.04)}{0.50}\,B^{12})a_{1t}^{JADE}$	$\Delta\Delta_{12}s_{2t}^{JADE} = (1 + \underset{(0.05)}{0.84}\,B - \underset{(0.05)}{0.18}\,B^2)(1 + \underset{(0.03)}{0.73}\,B^{12})a_{2t}^{JADE}$
FOTBI	ARIMA(0,1,1)×$(1,0,0)_{12}$	ARIMA(0,1,1)×$(1,1,1)_{12}$
	$(1 + \underset{(0.04)}{0.58}\,B^{12})\Delta s_{1t}^{FOTBI} = (1 - \underset{(0.05)}{0.16}\,B)a_{1t}^{FOTBI}$	$(1 + \underset{(0.03)}{0.81}\,B^{12})\Delta\Delta_{12}s_{2t}^{FOTBI} = (1 + \underset{(0.03)}{0.75}\,B)(1 - \underset{(0.03)}{0.85}\,B^{12})a_{2t}^{FOTBI}$
SOBI	ARIMA(0,1,1)×$(0,1,1)_{12}$	ARIMA(3,0,1)×$(0,1,1)_{12}$
	$\Delta\Delta_{12}s_{1t}^{SOBI} = (1 + \underset{(0.04)}{0.66}\,B)(1 + \underset{(0.03)}{0.70}\,B^{12})a_{1t}^{SOBI}$	$(1 - \underset{(0.16)}{0.36}\,B - \underset{(0.06)}{0.24}\,B^3)\Delta_{12}s_{2t}^{SOBI} = (1 + \underset{(0.15)}{0.31}\,B)(1 + \underset{(0.05)}{0.27}\,B^{12})a_{2t}^{SOBI}$
	Observed time series data (IPI)	
FRA	ARIMA(3,1,1)×$(0,1,1)_{12}$	
	$(1 - \underset{(0.06)}{0.16}\,B^2 - \underset{(0.05)}{0.44}\,B^3)\Delta\Delta_{12}x_{1t} = (1 + \underset{(0.04)}{0.89}\,B)(1 + \underset{(0.05)}{0.49}\,B^{12})a_{1t}$	
GER	ARIMA(2,1,0)×$(0,1,1)_{12}$	
	$(1 + \underset{(0.05)}{0.54}\,B + \underset{(0.05)}{0.21}\,B^2)\Delta\Delta_{12}x_{2t} = (1 + \underset{(0.04)}{0.59}\,B^{12})a_{2t}$	
ITA	ARIMA(0,1,1)×$(0,1,1)_{12}$	
	$\Delta\Delta_{12}x_{3t} = (1 + \underset{(0.04)}{0.60}\,B)(1 + \underset{(0.05)}{0.41}\,B^{12})a_{3t}$	
SPA	ARIMA(0,1,1)×$(0,1,1)_{12}$	
	$\Delta\Delta_{12}x_{4t} = (1 + \underset{(0.04)}{0.58}\,B)(1 + \underset{(0.04)}{0.51}\,B^{12})a_{4t}$	

The standard deviations for the estimates of the parameters are given between brackets

Table 4.14: Ratios of the four different evaluation criteria with respect to the benchmark (univariate ARIMA) model, $RelRMSE$, $RelMAPE$, $RelMASE$, and $RelGMAE$, measured over the IPI of each country

		RelRMSE				RelMAPE				RelMASE				RelGMAE			
		FRA	GER	ITA	SPA	FRA	GER	ITA	SPA	FRA	GER	ITA	SPA	FRA	GER	ITA	SPA
h=1	PCA	1.5296	2.6222	1.6941	1.4963	1.4479	1.7489	1.3800	1.2992	1.4345	3.0073	1.6438	1.4564	1.3506	3.4082	1.6978	1.0025
	JADE	2.7092	6.1680	3.2358	3.5241	1.6819	3.7333	2.1143	1.5590	2.7691	7.3742	3.3784	3.8814	2.7822	8.6476	3.5968	5.2167
	FOTBI	1.0786	1.1645	0.8055	1.1573	1.1830	1.1200	0.6876	0.8337	0.9438	1.2282	0.7864	1.1041	0.6978	1.3043	0.8182	1.4128
	SOBI	2.4338	1.8017	3.3811	3.0215	2.5811	1.6473	2.5078	1.5994	2.0687	2.0049	3.1210	2.9926	1.5812	2.3487	3.2827	3.3993
	UNIV	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
h=3	PCA	1.1682	1.3892	1.0792	0.9580	0.8442	1.0211	0.7301	1.1179	1.1112	1.5797	1.1045	0.9760	1.0292	1.8151	1.0473	0.9922
	JADE	1.7996	3.0036	1.8179	2.0294	0.8827	0.9970	0.8673	0.8054	1.7386	3.5878	1.9270	2.1996	1.6782	4.6292	2.3344	2.3937
	FOTBI	0.9550	0.9844	0.7654	1.2676	0.9064	1.8008	1.1037	0.5474	0.8903	1.0831	0.7485	1.2965	0.8074	1.2739	0.6918	1.3127
	SOBI	1.8172	1.1094	2.1030	2.1164	1.5383	1.0767	1.4430	1.2197	1.5605	1.2532	2.0290	2.1898	1.3391	1.4202	2.3602	2.1913
	UNIV	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
h=6	PCA	0.9198	0.9790	0.8908	1.2431	0.6400	0.9605	0.7078	1.1772	0.8954	0.9930	0.8582	1.3006	0.7870	0.8968	0.7138	1.4837
	JADE	1.0099	1.6256	1.0595	1.0989	0.5990	0.7166	0.6779	0.7695	0.9672	1.7407	1.0692	1.0867	0.8688	1.8461	0.9966	1.0125
	FOTBI	0.8589	0.9085	0.8467	0.9183	0.4419	0.9693	0.5747	0.4074	0.8291	0.9208	0.8640	0.9204	0.5952	0.8258	0.8814	0.9111
	SOBI	1.2670	0.8618	1.5423	1.6752	0.9781	0.8641	1.0054	1.2015	1.1397	0.8749	1.4205	1.6343	0.9689	0.8446	1.1621	1.4570
	UNIV	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
h=12	PCA	0.9914	1.0430	1.0619	1.2963	0.8345	0.9808	1.1730	1.0675	0.9700	1.0204	1.0326	1.3220	0.9059	0.8889	1.0366	1.3087
	JADE	0.7476	0.7625	0.7792	0.8913	0.4868	0.6830	0.6939	0.8814	0.6537	0.8633	0.6682	0.7441	0.5314	0.6877	0.5253	0.7312
	FOTBI	0.7125	0.9517	0.7178	0.7631	0.3713	0.5652	0.4875	0.4489	0.6580	0.7164	0.7446	0.8675	0.4348	0.5847	0.7028	0.7587
	SOBI	0.9887	0.8038	1.1965	1.3538	0.7165	0.7960	0.9448	1.1097	0.8402	0.7234	0.9990	1.2127	0.6338	0.5398	0.7030	0.8543
	UNIV	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Table 4.15: Results of the Diebold-Mariano test considering the squared error loss function (for each IPI time series)

	h=1				h=3				h=6				h=12			
Met A vs Met B	FRA	GER	ITA	SPA	FRA	GER	ITA	SPA	FRA	GER	ITA	SPA	FRA	GER	ITA	SPA
PCA vs UNIV	2.3497 0.0094 UNIV	4.0582 0.0000 UNIV	2.7108 0.0034 UNIV	3.0295 0.0012 UNIV	1.1791 0.1192 =	1.9856 0.0297 UNIV	0.5329 0.2971 =	-0.5160 0.3029 =	-0.9598 0.1686 =	-0.3329 0.3696 =	-1.2318 0.1090 =	3.9932 0.0000 UNIV	-0.1251 0.4502 =	0.6337 0.2631 =	0.7034 0.2409 =	3.4339 0.0003 UNIV
JADE vs UNIV	4.9888 0.0000 UNIV	6.0409 0.0000 UNIV	5.5669 0.0000 UNIV	5.8904 0.0000 UNIV	3.3830 0.0004 UNIV	4.6911 0.0000 UNIV	3.4075 0.0003 UNIV	3.6509 0.0001 UNIV	0.0703 0.4720 =	2.6840 0.0036 UNIV	0.3728 0.3547 =	0.5615 0.2872 =	-2.5792 0.0050 JADE	-0.3302 0.3706 =	-2.6525 0.0040 JADE	-2.0162 0.0431 JADE
FOTBI vs UNIV	0.5283 0.2986 =	1.9763 0.0478 UNIV	-2.0656 0.0428 FOTBI	1.1201 0.1313 =	-0.4561 0.3242 =	-0.2305 0.4088 =	-2.1041 0.0177 FOTBI	2.7656 0.0028 UNIV	-1.3319 0.0914 =	-0.6652 0.2529 =	-1.6072 0.0540 =	-1.5766 0.0574 =	-2.5218 0.0058 FOTBI	-2.8283 0.0023 FOTBI	-3.7047 0.0001 FOTBI	-3.4430 0.0003 FOTBI
SOBI vs UNIV	2.7632 0.0029 UNIV	2.9532 0.0016 UNIV	3.1009 0.0010 UNIV	4.1265 0.0000 UNIV	2.4961 0.0063 UNIV	0.8862 0.1878 =	2.7992 0.0026 UNIV	4.1411 0.0000 UNIV	1.3025 0.0964 =	-1.9860 0.0459 SOBI	2.2890 0.0110 UNIV	3.6752 0.0001 UNIV	-0.0684 0.4727 =	-4.7883 0.0000 SOBI	1.1814 0.1187 =	2.4961 0.0063 UNIV
PCA vs SOBI	-2.2734 0.0115 PCA	4.0572 0.0000 PCA	-2.4816 0.0065 PCA	-3.3368 0.0004 PCA	-2.4146 0.0079 PCA	2.4288 0.0076 SOBI	-2.6180 0.0044 PCA	-4.2146 0.0000 PCA	-1.9606 0.0250 PCA	1.6346 0.0511 =	-2.5658 0.0051 PCA	-2.4014 0.0082 PCA	0.0199 0.4921 =	3.0383 0.0012 SOBI	-0.6505 0.2577 =	-0.3403 0.3668 =
JADE vs SOBI	0.5795 0.2811 =	6.1554 0.0000 SOBI	-0.2352 0.4070 =	1.0548 0.1458 =	-0.0509 0.4797 =	5.0861 0.0000 SOBI	-0.7549 0.2251 =	-0.2679 0.3944 =	-1.0950 0.1368 =	3.7811 0.0001 SOBI	-1.9718 0.0382 JADE	-2.2744 0.0115 JADE	-1.4222 0.0775 =	1.0027 0.1580 =	-1.9706 0.0244 JADE	-2.6061 0.0046 JADE
FOTBI vs SOBI	-2.8171 0.0024 FOTBI	-2.8627 0.0021 FOTBI	-3.2383 0.0006 FOTBI	-3.9357 0.0000 FOTBI	-2.7606 0.0029 FOTBI	-1.2698 0.1021 =	-3.2179 0.0006 FOTBI	-3.2692 0.0005 FOTBI	-2.4519 0.0071 FOTBI	0.5558 0.2892 =	-2.7485 0.0030 FOTBI	-3.9552 0.0000 FOTBI	-2.0161 0.0277 FOTBI	-0.5534 0.2900 =	-2.1555 0.0156 FOTBI	-3.0296 0.0012 FOTBI
PCA vs FOTBI	2.5796 0.0049 FOTBI	4.1188 0.0000 FOTBI	3.4147 0.0003 FOTBI	1.9996 0.0469 FOTBI	2.0113 0.0221 FOTBI	2.3335 0.0098 FOTBI	2.2074 0.0136 FOTBI	-2.1988 0.0139 PCA	1.1261 0.1301 =	0.7519 0.2261 =	0.4969 0.3096 =	3.7811 0.0001 FOTBI	3.7934 0.0001 FOTBI	4.6473 0.0000 FOTBI	2.8276 0.0023 FOTBI	4.5043 0.0000 FOTBI
JADE vs FOTBI	4.6019 0.0000 FOTBI	6.0225 0.0000 FOTBI	6.0089 0.0000 FOTBI	5.6832 0.0000 FOTBI	3.3801 0.0004 FOTBI	4.7980 0.0000 FOTBI	4.4571 0.0000 FOTBI	3.0783 0.0010 FOTBI	1.1519 0.1247 =	4.3926 0.0000 FOTBI	2.0532 0.0254 FOTBI	1.1216 0.1310 =	-0.3513 0.3627 =	2.0363 0.0413 FOTBI	-0.7183 0.2363 =	-1.0066 0.1571 =
PCA vs JADE	-4.0967 0.0000 PCA	-6.3429 0.0000 PCA	-5.3449 0.0000 PCA	-5.7935 0.0000 PCA	-3.0023 0.0013 PCA	-5.3856 0.0000 PCA	-4.3005 0.0000 PCA	-3.9148 0.0000 PCA	-0.6330 0.2634 =	-2.9818 0.0014 PCA	-1.2330 0.1088 =	0.9099 0.1815 =	2.4825 0.0065 JADE	0.6789 0.2486 =	2.8513 0.0022 JADE	3.7156 0.0001 JADE

Table 4.16: Results of the Diebold-Mariano test considering the absolute error loss function (for each IPI time series)

	h=1				h=3				h=6				h=12			
Met A vs Met B	FRA	GER	ITA	SPA	FRA	GER	ITA	SPA	FRA	GER	ITA	SPA	FRA	GER	ITA	SPA
PCA vs UNIV	2.1373 0.0163 UNIV	5.3360 0.0000 UNIV	2.6293 0.0043 UNIV	2.7013 0.0035 UNIV	0.7404 0.2295 =	2.3572 0.0092 UNIV	0.6419 0.2605 =	-0.2934 0.3846 =	-1.0582 0.1450 =	-0.0942 0.4625 =	-1.4252 0.0771 =	3.3397 0.0004 UNIV	-0.4278 0.3344 =	0.2729 0.3925 =	0.3815 0.3514 =	3.5522 0.0002 UNIV
JADE vs UNIV	5.5998 0.0000 UNIV	8.2868 0.0000 UNIV	5.9620 0.0000 UNIV	7.6683 0.0000 UNIV	3.0917 0.0010 UNIV	5.7584 0.0000 UNIV	3.2256 0.0006 UNIV	3.8976 0.0000 UNIV	-0.1905 0.4244 =	2.8435 0.0022 UNIV	0.3358 0.3685 =	0.4073 0.3419 =	-2.8070 0.0025 JADE	-0.8230 0.2053 =	-2.6861 0.0036 JADE	-1.6274 0.0518 =
FOTBI vs UNIV	-0.3956 0.3462 =	1.8663 0.0310 UNIV	-1.6333 0.0512 =	0.8101 0.2089 =	-1.0205 0.1537 =	0.9736 0.1651 =	-2.0890 0.0184 FOTBI	2.2913 0.0110 UNIV	-1.4041 0.0801 =	-0.5308 0.2978 =	-1.2025 0.1146 =	-1.4893 0.0682 =	-3.1589 0.0008 FOTBI	-3.2807 0.0005 FOTBI	-3.6754 0.0001 FOTBI	-3.0610 0.0011 FOTBI
SOBI vs UNIV	3.0770 0.0010 UNIV	3.9510 0.0000 UNIV	4.5951 0.0000 UNIV	5.6577 0.0000 UNIV	2.2336 0.0128 UNIV	1.5892 0.0560 =	3.4326 0.0003 UNIV	4.8873 0.0000 UNIV	0.6714 0.2510 =	-1.2569 0.1044 =	1.8914 0.0293 UNIV	3.5057 0.0002 UNIV	-1.0163 0.1547 =	-5.4505 0.0000 SOBI	-0.0072 0.4971 =	1.5122 0.0652 =
PCA vs SOBI	-2.2328 0.0128 PCA	3.8989 0.0000 SOBI	-2.9377 0.0017 PCA	-3.8455 0.0001 PCA	-2.2240 0.0131 PCA	2.1032 0.0177 SOBI	-2.9232 0.0017 FOTBI	-4.6030 0.0000 FOTBI	-1.4307 0.0763 =	1.1933 0.1164 =	-2.4611 0.0069 PCA	-1.5995 0.0549 =	0.9528 0.1703 =	3.5269 0.0002 SOBI	0.1780 0.4294 =	0.5575 0.2886 =
JADE vs SOBI	1.3624 0.0865 =	7.9452 0.0000 SOBI	0.4053 0.3426 =	1.5365 0.0622 =	0.4894 0.3123 =	6.2437 0.0000 SOBI	-0.2497 0.4014 =	0.0233 0.4907 =	-0.6621 0.2539 =	3.9422 0.0000 SOBI	-1.2044 0.1142 =	-1.8415 0.0328 JADE	-0.9909 0.1609 =	0.8027 0.2111 =	-1.4930 0.0677 =	-1.9846 0.0236 JADE
FOTBI vs SOBI	-3.6466 0.0001 FOTBI	-3.7743 0.0001 FOTBI	-5.2761 0.0000 FOTBI	-4.9129 0.0000 FOTBI	-3.0846 0.0010 FOTBI	-1.3158 0.0941 =	-4.5742 0.0000 FOTBI	-3.2987 0.0005 FOTBI	-2.0922 0.0182 FOTBI	0.4087 0.3414 =	-2.4859 0.0065 FOTBI	-3.6825 0.0001 FOTBI	-1.6137 0.0533 =	-0.0875 0.4651 =	-1.6304 0.0515 =	-2.2050 0.0137 FOTBI
PCA vs FOTBI	2.9152 0.0018 FOTBI	5.5287 0.0000 FOTBI	3.6604 0.0001 FOTBI	1.9812 0.0493 FOTBI	1.9701 0.0244 FOTBI	2.4154 0.0079 FOTBI	2.3030 0.0106 FOTBI	-1.9289 0.0269 PCA	0.9025 0.1834 =	0.6548 0.2563 =	-0.0550 0.4781 =	3.7394 0.0001 FOTBI	4.6161 0.0000 FOTBI	4.6611 0.0000 FOTBI	3.1174 0.0009 FOTBI	4.5286 0.0000 FOTBI
JADE vs FOTBI	5.3447 0.0000 FOTBI	8.1754 0.0000 FOTBI	7.1663 0.0000 FOTBI	7.2710 0.0000 FOTBI	3.3415 0.0004 FOTBI	5.7275 0.0000 FOTBI	4.5488 0.0000 FOTBI	3.0097 0.0013 FOTBI	0.8434 0.1995 =	5.0193 0.0000 FOTBI	1.3695 0.0854 =	0.8491 0.1979 =	-0.0351 0.4860 =	1.1287 0.1295 =	-0.7052 0.2403 =	-0.8387 0.2008 =
PCA vs JADE	-4.4548 0.0000 PCA	-9.0368 0.0000 PCA	-5.2800 0.0000 PCA	-6.8152 0.0000 PCA	-2.8985 0.0019 PCA	-7.3121 0.0000 PCA	-3.9471 0.0000 PCA	-4.1657 0.0000 PCA	-0.4043 0.3430 =	-3.0683 0.0011 PCA	-1.1390 0.1273 =	1.1032 0.1350 =	2.4334 0.0075 JADE	0.9492 0.1712 =	2.8769 0.0020 JADE	3.3237 0.0004 JADE

Chapter 5

Conclusions and Future Research

Finding an 'interesting representation' of large data sets becomes an important task in multivariate data analysis. Classical methods such as PCA and FA have been proposed to obtain a meaningful representation of Gaussian data. However, in many practical situations, we are far away from Gaussianity and previous procedures fail. Recently ICA has emerged in the literature to get an 'interesting representation' of non-Gaussian data by using higher-order statistics. The idea of ICA is looking for the projections of the data which become as independent as possible. In Chapter 1, we investigate the relationship between ICA and other classical multivariate methods. We present ICA as an extension of PCA, in the sense that the ICs will be estimated as the rotation of the PCs that makes them maximally independent. Moreover, ICA can be seen as a non-linear factor model where the ICs are statistically independent instead of mutually uncorrelated. Moreover, moving to higher-order methods, if the statistical independence of the components is measured in terms of their non-Gaussianity, ICA is a special case of PP. In addition, ICA is related to the algorithm proposed by Peña and Prieto (2001): both procedures detect the outliers of the data set by projecting the observations onto the directions of maximum kurtosis.

We present in Chapter 2 a new multivariate conditionally heteroskedastic factor model, the GICA-GARCH model, where the observations are assumed to be linearly generated by a set of latent factors which are statistically independent and have GARCH effects. The GICA-GARCH model assumes factor structure in the unconditional distribution of the data and applies ICA to estimate the set of conditionally heteroskedastic components which explain the co-movements of the observations. In addition, the GICA-GARCH also assumes factor structure in the conditional distribution of the data and computes the conditional covariance matrix of the observations as a linear combination of the conditional variances of

those common factors. The advantages of the GICA-GARCH model over existing multivariate volatilities modelling approaches relies on the use of ICA to estimate the conditionally heteroskedastic factors. First, as the Monte Carlo experiments of Chapter 2 show, ICA is a powerful methodology to reproduce excess kurtosis: ICA procedures, specially FastICA and JADE, perform quite well to identify the non-Gaussian latent factors when their excess kurtosis comes from either different GARCH specifications or different conditional distributions. Second, since the ICs are assumed to be statistically independent, the latent factors in the GICA-GARCH model do not need to be jointly estimated and a univariate (ARMA)-GARCH model is fitted to each of them. Therefore, the GICA-GARCH model transform the complexity associated with the estimation of a multidimensional ARMA-GARCH model into the estimation of a few univariate (ARMA)-GARCH models. The third advantage of the GICA-GARCH model is illustrated in our empirical application to the Madrid stock market, where we compare the forecasting performance of our model with respect to existing models such as the orthogonal GARCH model (Alexander (2001)) and the CUC-GARCH (Fan et al. (2008)). We show that the GICA-GARCH model, where the underlying components are estimated by JADE and modelled as ARMA-GARCH processes with conditional Student's t innovations, provides the most accurate one-step ahead forecasts for the stocks of the IBEX 35 index.

In Chapter 2, we also investigate the relationship between the GICA-GARCH and alternative factor GARCH models, depending on whether the factor structure refers to the unconditional or conditional distribution of the data. On the other hand, since the GICA-GARCH assumes factor structure in the unconditional distribution of the data, it can be seen a latent factor model with GARCH effects (Diebold and Nerlove (1989)). Moreover, we present the GICA-GARCH model as a parsimonious version of the DF-GARCH (Alessi et al. (2006)): whereas the DF-GARCH assumes that the common factors evolve according to a MGARCH model, the GICA-GARCH model fits different univariate (ARMA)-GARCH models to each of them. On the other hand, since GICA-GARCH model assumes factor structure in the conditional distribution of the data, it is related to the FACTOR-ARCH model (Engle (1987)): for both models the data conditional covariance matrix is given by a linear combination of the conditional variances of some portfolios. The GICA-GARCH model is also related to some orthogonal GARCH models: it can be seen as a generalization of the O-GARCH (Alexander (2001)), where the estimates of the factors are given by the ICs instead

of the PCs, and it extends the GOF-GARCH (Lanne and Saikkonen (2007)) by allowing the noisy components to be conditionally heteroskedastic. In addition, we show that the GICA-GARCH model is quite similar to the the CUC-GARCH (Fan et al. (2008)) model but making the stronger assumption of mutually independence, instead of conditionally uncorrelatedness, on the components.

In Chapter 3, we present a new ICA procedure for multivariate time series data. Our procedure, called FOTBI, is designed to obtain the set of non-Gaussian and statistically independent components in a multivariate time series vector of observations. FOTBI is a fourth-order method that is based on the joint diagonalization of several time-delayed fourth-order cumulant matrices. In contrast to other ICA algorithms, FOTBI exploits the temporal structure as well as the non-Gaussianity of the data: on the one hand, FOTBI extend the JADE algorithm (Cardoso and Souloumiac (1993)) by taking into account the temporal dependencies among the observations; on the other hand, FOTBI broadens the SOBI procedure (Belouchrani et al. (1997)) by introducing higher-order statistics which guarantees the statistical independence of the components under non-Gaussianity assumption. The results of the simulation experiments show the advantage of FOTBI to extract non-linear time series independent components: for multivariate time series data sets that are non-Gaussian distributed, FOTBI outperforms the rest of the ICA procedures at different sample sizes. However, since FOTBI is a fourth-order method and requires large sample sizes to reduce the variance of the estimates, the performance of FOTBI improves when the sample size increases. Moreover, the Monte Carlo experiment show that, although SOBI has the best performance for Gaussian time series data, FOTBI seems to be a 'good' competitor, specially for large sample sizes.

Chapter 4 is concern with the problem of prediction and signal extraction in multivariate time series data by using ICA. We present ICA as an alternative multivariate time series signal extraction procedure. ICA is applied for finding the possible non-stationary components, such as the trend and the seasonal components, in a multivariate time series vector of observations. The advantage of ICA over the existing signal extraction procedures is that it is automatic and does not assumes any a-priori structure either in the observations nor in the underlying components of interest. ICA only assumes that the basic components are mutually independent and such that they explain as much of the total variability as possible. According to the results of some Monte Carlo experiments, FOTBI provides the most reliable estimates for the components and is able to separate the trend, cyclical, and seasonal components. Therefore,

FOTBI could be seen as the first-step for designing an automatic procedure in multivariate time series signal extraction. The empirical application to the industrial production index (IPI) time series of four European countries confirms the usefulness of FOTBI to identify the trend and seasonality patterns. However, the FOTBI components are a bit noisy and some smoothing techniques should be applied to consider FOTBI as a proper signal extraction procedure.

Moreover, in Chapter 4, we investigate the usefulness of forecasting multivariate time series data sets by using a small number of ICs. Since the ICs are statistically independent, they can be forecasted separately by using different univariate models to each of them. Based on this idea, our forecasting approach consist in making the forecasts in the space of the ICs, and then using these predictions and transforming them back to the space of the observations. We empirically test our approach for predicting the IPI time series of four European countries by using two ICs: the trend and the seasonal components. We analyze the forecasting performance of FOTBI and alternative ICA procedures with respect to some benchmark models, which are the univariate specifications given by the automatic TRAMO/SEATS identification procedure (Gómez and Maravall (1996)). The results show the potential of FOTBI to forecast the IPIs: the short-term IPIs forecasts ($h = 1, 3$) given by FOTBI and the univariate models (which performs well in the short-term) are quite similar; FOTBI outperforms overall procedures when $h = 6$; the most accurate long-term IPIs forecasts ($h = 12$) are provided by FOTBI, JADE, and SOBI (the three ICA procedures have equally forecasting performance).

In the following, we summarize several topics that have been arising while working on this thesis, and which will be the directions of the future research:

- To propose a formal criterion to sort the ICs. In Chapter 2, as PCA does, the ICs are sorted in terms of variability. However, since ICA uses higher-order information to identify the statistically independent underlying components, it could be interesting to define a new criterion based on the independence of the components.

- To extend the empirical application in Chapter 2 by analyzing the performance of the DF-GARCH model (Alessi et al. (2006)). The GICA-GARCH model assumes factor structure in the conditional as well as in the unconditional distribution of the data. In Chapter 2, we compare the performance of the GICA-GARCH model with the O- and the CUC-GARCH models, which assumes factor structure in the conditional distribution

of the data. However, there is no comparison of the GICA-GARCH model with models that assume factor structure in the unconditional distribution such as the DF-GARCH model.

- To investigate more about the statistical properties of the ICs estimates. Moreover, in Chapter 3, it would be interesting to analyze the asymptotic performance of FOTBI.
- To study more deeply the issue of non-stationarity in the ICA model. In Chapter 4, we apply ICA to a vector of non-stationary time series data, and the results are quite 'good'. However, additional theoretical research on non-stationary ICA is needed.
- To propose a procedure to test for the optimal number of ICs, stationary or not.

Bibliography

Alessi, L., M. Barigozzi, and M. Capasso (2006). Dynamic factor garch: Multivariate volatility forecast for a large number of series. *LEM Papers Series 2006/25. Laboratory of Economics and Management (LEM). Sant'Anna School of Advanced Studies*.

Alexander, C. (2001). *Market Models: A guide to financial data analysis*. Chichester, U.K.: John Wiley.

Amari, S.-I., A. Cichocki, and H. H. Yang (1996). A new learning algorithm for blind source separation. *Advances in Neural Information Processing Systems 8*, 757–763.

Andersen, T. G. and T. Bollerslev (1998). Answering the skeptics: yes, standard volatility models do provide accurate forecasts. *International Economic Review 39*, 885–905.

Back, A. D. and A. S. Weigend (1997). A first application of independent component analysis to extracting structure from stock returns. *International Journal of Neural Systems 8*, 473–484.

Baragona, R. and F. Battaglia (2007). Outliers detection in multivariate time series by independent component analysis. *Neural Computation 19*, 1962–1984.

Bauwens, L., S. Laurent, and J. V. K. Rombouts (2006). Multivariate garch models: a survey. *Journal of Applied Econometrics 21*, 79–109.

Baxter, M. and R. G. King (1995). Measuring business cycles: Approximate band-pass filters for economic times series. *NBER WP 5022*.

Bell, A. J. and T. J. Sejnowski (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation 7*, 1129–1159.

Bell, A. J. and T. J. Sejnowski (1997). The independent components of natural scenes are edge filters. *Vision Research 37*, 3327–3338.

Bell, W. R. (1984). Signal extraction for nonstationary time series. *The Annals of Statistics 12*.

Bell, W. R. and S. C. Hillmer (1984). Issues involved with the seasonal adjustment of economic time series. *Journal of Business and Economic Statistics 2*, 291–320.

Belouchrani, A., K. Abed Meraim, J.-F. Cardoso, and E. Moulines (1997). A blind source separation technique based on second order statistics. *IEEE Transactions on Signal Processing 45*, 434–444.

Beveridge, S. and C. R. Nelson (1981). A new approach to decomposition of economic time series into permanent and transitory components with particular attention to measurement of the business cycle. *Journal of Monetary Economics 7*, 151–174.

Bingham, E. (2001). Topic identification in dynamical text by extracting minimum complexity time components. *Proceedings of the 3rd International Workshop on Independent Component Analysis and Blind Source Separation*, 546–551.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroscedasticity. *Journal of Econometrics 31*, 307–327.

Bollerslev, T. (1987). A conditional heteroscedastic time series model for speculative prices and rates of return. *Review of Economics and Statistics 69*, 542–547.

Bollerslev, T. and R. F. Engle (1993). Common persistence in conditional variances. *Econometrica 61*, 167186.

Bollerslev, T., R. F. Engle, and J. W. Wooldridge (1988). A capital asset pricing model with time-varying covariances. *Journal of Political Economy 96*, 116–131.

Bonhomme, S. and J.-M. Robin (2009). Consistent noisy independent component analysis. *Journal of Econometrics 149*.

Box, G. and G. C. Tiao (1977). A canonical analysis of multiple time series. *Biometrika 64*, 355–365.

Box, G. E. P., S. C. Hillmer, and G. C. Tiao (1978). *Analysis and modelling of seasonal time series*. Washington, D.C.: U.S. Department of Commerce. Bureau of the Census.

Brillinger, D. R. (1981). *Time Series Data Analysis and Theory*. San Francisco: expanded edition. Holden Day.

Bujosa, M., A. García-Ferrer, and P. C. Young (2007). Linear dynamic harmonic regression. *Computational Statistics and Data Analysis 52*, 999–1024.

Burman, J. P. (1980). Seasonal adjustment by signal extraction. *Journal of the Royal Statistical Society series A 143*, 321–336.

Cardoso, J.-F. (1989). Source separation using higher order moments. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2109–2112.

Cardoso, J.-F. (1997). Infomax and maximum likelihood for source separation. *IEEE Letters on Signal Processing 4*, 112–114.

Cardoso, J.-F. and A. Souloumiac (1993). Blind beamforming for non-gaussian signals. *IEE-Proceedings-F 140*, 362–370.

Cha, S.-M. and L.-W. Chan (2000). Applying independent component analysis to factor model in finance. *Proceedings of the International Symposium on Intelligent Data Engineering and Learning*, 538–544.

Choi, S. and A. Cichocki (2000). Blind separation of nonstationary sources in noisy mixtures. *Electronics Letters 36*.

Cleveland, W. P. and G. C. Tiao (1976). Decomposition of seasonal time series: a model for the x-11 program. *Journal of the American Statistical Association 71*, 581–587.

Comon, P. (1994). Independent component analysis - a new concept? *Signal Processing 36*, 287–314.

Cover, T. M. and J. A. Thomas (2001). *Elements of Information Theory*. ????: John Wiley.

Dagum, E. B. (1980). The x-11 arima seasonal adjustment method. *Catalogue No. 12-564E, Statistics Canada*.

Delfosse, N. and P. Loubaton (1995). Adaptative blind separation of independent sources: a deflation approach. *Signal Processing 45*, 59–83.

Diebold, F. X. and R. S. Mariano (1995). Comparing predictive accuracy. *Journal of Business Economic Statistics 13*, 253–263.

Diebold, F. X. and M. Nerlove (1989). The dynamics of exchange rate volatility: a multivariate latent factor arch model. *Journal of Applied Econometrics 4*, 1–21.

Doz, C. and E. Renault (2004). Conditionally heteroskedastic factor models: Identification and instrumental variables estimation. *CIRANO working paper*.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica 50*, 987–1007.

Engle, R. F. (1987). Multivariate arch with factor structures - cointegration in variance. *Discussion Paper 87-2, UCSD, Department of Economics*.

Engle, R. F., C. W. J. Granger, and D. Kraft (1984). Combining competing forecasts of inflation using a bivariate arch model. *Journal of Economic Dynamics and Control 8*, 151165.

Engle, R. F. and K. Kroner (1995). Multivariate simultaneous garch. *Econometric Theory 11*, 122–150.

Engle, R. F., V. M. Ng, and M. Rothschild (1990). Asset pricing with a factor-arch structure: Empirical estimates for treasury bills. *Journal of Econometrics 45*, 213–237.

Engle, R. F. and R. Susmel (1993). Common volatility in international equity markets. *Journal of Business and Economic Statistics 11*, 167–176.

Fan, J., M. Wang, and Q. Yao (2008). Modelling multivariate volatilities via conditionally uncorrelated components. *Journal of the Royal Statistical Society B 70*, 679–702.

Findley, D. F., B. C. Monsell, W. R. Bell, M. C. Otto, and B.-C. Chen (1998). New capabilities and methods of the x-12 arima seasonal-adjustment program. *Journal of Business and Economic Statistics 16*.

Flury, B. (1984). Common principal components in k groups. *Journal of the American Statistical Association 79*, 892–898.

Forni, M., M. Hallin, M. Lippi, and L. Reichlin (2000). The generalized dynamic factor model: identification and estimation. *Review of Economics and Statistics 82*, 540–554.

Franses, P. H. and D. van Dijk (1996). Forecasting stock market volatility using nonlinear garch models. *Journal of Forecasting 15*, 229–235.

Friedman, J. H. and J. W. Tuckey (1974). A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on Computers*, 881–889.

Funaro, M., E. Oja, and H. Valpola (2001). Artefact detection in astrophysical image data using independent component analysis. *Proceedings of the 3rd International Workshop on Independent Component Analysis and Blind Source Separation*, 43–48.

Gaeta, M. and J.-L. Lacoume (1990). Source separation without prior knowledge: the maximum likelihood solution. *Proceedings of the EUSIPCO*, 621–624.

Geweke, J. F. and K. J. Singleton (1981). Maximum likelihood confirmatory analysis of economic time series. *International Economic Review 22*, 37–54.

Gómez, V. and A. Maravall (1996). Programs tramo and seats; instructions for the user. *Working Paper 9628, Servicio de Estudios, Banco de Espaa*.

Granger, C. and Z. Ding (1995). Some properties of absolute return, an alternative measure of risk. *Annales d'Economie et de Statistique 40*, 67–91.

Hansen, P. R. and A. Lunde (2005). A forecast comparison of volatility models: does anything beat a garch(1,1)? *Journal of Applied Econometrics 20*, 873–889.

Hansen, P. R. and A. Lunde (2006). Consistent ranking of volatility models. *Journal of Econometrics 131*, 97–121.

Harvey, A., E. Ruiz, and E. Sentana (1992). Unobserved component time series models with arch disturbances. *Journal of Econometrics 52*, 129–157.

Harvey, A. C. (1989). *Forecasting structural time series models and the Kalman filter*. Cambridge: Cambridge University Press.

Hérault, J. and C. Jutten (1986). Space or time adaptative signal processing by neural networks models. *International Conference on Neural Networks for Computing*, 206–211.

Hillmer, S. C. and G. C. Tiao (1982). An arima-model-based approach to seasonal adjustment. *Journal of the American Statistical Association 77*, 63–70.

Hodrick, J. R. and E. C. Prescott (1997). Postwar u.s. business cycles: an empirical investigation. *Journal of Money, Credit and Banking 29*.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology 24*.

Huber, P. J. (1985). Projection pursuit. *The Annals of Statistics 13*.

Hyndman, R. J. and A. B. Koehler (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting 22*, 679–688.

Hyvärinen, A. (1998a). Independent component analysis for time-dependent stochastic processes. *Proceedings International Conference on Artificial Neural Networks*, 135–140.

Hyvärinen, A. (1998b). New approximations of differential entropy for independent component analysis and projection pursuit. *Advances in Neural Information Processing Systems 10*, 273–279.

Hyvärinen, A. (1999a). Fast and robust fixed-point algorithms for independent component analysis. *IEEE. Transactions on Neural Networks 10*, 626–634.

Hyvärinen, A. (1999b). Sparse code shrinkage: Denoising of nongaussian data by maximum likelihood estimation. *Neural Computation 11*, 1739–1768.

Hyvärinen, A. (2001). Blind source separation by nonstationarity of variance: A cumulant-based approach. *IEEE Transactions on Neural Networks 12*.

Hyvärinen, A. and K. Kano (2003). *Independent component analysis for non-normal factor analysis*. Tokyo: Springer Verlag.

Hyvärinen, A., J. Karhunen, and E. Oja (2001). *Independent component analysis*. New York: John Wiley.

Hyvärinen, A. and E. Oja (1997). A fast fixed-point algorithm for independent component analysis. *Neural Computation 9*, 1483–1492.

Jones, M. C. and R. Sibson (1987). What is projection pursuit? *Journal of the Royal Statistical Society, Serie A 150*, 1–36.

Jutten, C. and J. Hérault (1991). Blind separation sources, part i: An adaptative algorithm based on neuromimetic architecture. *Signal Processing 24*, 1–10.

K., D. and V. I. D. (2010). A student-t full factor multivariate garch model. *Computational Economics 35*, 63–83.

Kawamoto, M., K. Matsuoka, and M. Oya (1997). Blind separation of sources using temporal correlation of the observed signals. *IEICE Transactions Fundamentals*, 695–704.

King, M. A., E. Sentana, and S. B. Wadhwani (1994). Volatility and links between national stock markets. *Econometrica 62*, 901933.

Kiviluoto, K. and E. Oja (1998). Independent component analysis for parallel financial time series. *Proceedings International Conference on Neural Information Processing 2*, 895–898.

Koopman, S. J., A. C. Harvey, J. A. Doornik, and N. Shephard (1995). *STAMP: Structural Time Series Analyser, Modeller and Predictor*. London.

Lanne, M. and P. Saikkonen (2007). A multivariate generalized orthogonal factor garch model. *Journal of Business and Economic Statistics 25*, 61–75.

Lee, T.-W., M. Girolami, A. J. Bell, and T. J. Sejnowski (2000). A unifying information-theoretic framework for independent component analysis. *Computers and Mathematics with Applications 11*, 1–21.

Malaroiu, S., K. Kiviluoto, and E. Oja (2000). Time series prediction with independent component analysis. Technical report, Helsinki University of Technology, Helsinki.

Maravall, A. (1993). Stochastic linear trends: models and estimators. *Journal of Econometrics 56*, 5–37.

Maravall, A. and D. A. Pierce (1987). A prototypical seasonal adjustment model. *Journal of Time Series Analysis 8*.

Matsuoka, K., M. Ohya, and M. Kawamoto (1995). A neural net for blind separation of nonstationary signals. *Neural Networks 8*.

Mendel, J.-M. (1991). Tutorial on higher-order statistics (spectra) in signal processing and system theory: theoretical results and some applications. *Proceeding of the IEEE 79*.

Molgedey, L. and H.-G. Schuster (1994). Separation of a mixture of independent signals using time delayed correlations. *Physical Review Letters 72*.

Nelson, D. B. (1991). Conditional heteroskedasticity in asset returns: a new approach. *Econometrica 59*, 347–370.

Ng, V. and M. Engle, R. F. Rothschild (1992). A multi-dynamic-factor model for stock returns. *Journal of Econometrics 52*, 245266.

Nijman, T. and E. Sentana (1996). Marginalization and contemporaneous aggregation in multivariate garch processes. *Journal of Econometrics 71*, 71–87.

Oja, E. (1982). A simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology 15*, 267–273.

Peña, D. and G. Box (1987). Identify a simplifying structure in time series. *Journal of the American Statistical Association 82*, 836–843.

Peña, D. and P. Poncela (2006a). *Dimension reduction in time series.* Advances in Distribution Theory, Order Statistics and Inference, Balakrishnan, N. et al. (Editors).

Peña, D. and P. Poncela (2006b). Nonstationary dynamic factor analysis. *Journal of Statistical Planning and Inference 136*, 1237–1257.

Peña, D. and F. J. Prieto (2001). Cluster identification using projections. *Journal of the American Statistical Association 96*, 1433–1445.

Pham, D.-T. and J.-F. Cardoso (2001). Blind separation of instantaneous mixtures of non-stationary sources. *IEEE Transactions on Signal Processing 49*.

Pham, D. T., P. Garrat, and C. Jutten (1992). Separation of a mixture of independent sources through a maximum likelihood approach. *Proceeding EUSIPCO*, 771–774.

Ristaniemi, T. and J. Joutsensalo (1999). On the performance of blind source separation in cdma downlink. *Proceeding on the International Workshop on Independent Component Analysis and Signal Separation*, 437–441.

Särelä, J. and H. Valpola (2005). Denoising source separation. *Journal of Machine Learning Research 6*, 233–272.

Shiskin, J., A. H. Young, and J. C. Musgrave (1967). The x11 variant of the census method ii seasonal adjustmen t program. *Technical Paper, 15, Bureau of the Census*.

Spearman, C. (1904). General intelligence, objectively determined and measured. *American Journal of Psycology 15*, 201–293.

Stock, J. H. and M. W. Watson (2002). Macroeconomic forecasting using diffusion indexes. *Journal of Business and Economic Statistics 20*, 147–162.

Taylor, C. J., D. Pedregal, P. C. Young, and W. Tych (2007). Time series analysis and forecasting with the captain toolbox. *Environmental Modelling and Software 22*, 797–814.

Tiao, G. C. and R. S. Tsay (1989). Model specification in multivariate time series. *Journal of the Royal Statistical Society B 51*, 157–213.

Tong, L., R.-W. Liu, V. C. Soon, Y.-F. Huang, and R. Liu (1990). Amuse: A new blind identification algorithm. *Proceedings of the IEEE. International Symposium on Circuits and Systems*, 1784–1787.

van der Veen, A.-J. (2001). Joint diagonalization via subspace fitting techniques. *Proceedings of the ICASSP 5*.

van der Weide, R. (2002). Go-garch: a multivariate generalized orthogonal garch model. *Journal of Applied Econometrics 17*, 549–564.

Vigàrio, R., V. Jousmäki, M. Hämäläinen, R. Hari, and E. Oja (1998). Independent component analysis for identification of artifacts in magnetoencephalographic recordings. *Advances in Neural Information Processing Systems 10*, 229–235.

Vrontos, I. D., P. Dellaportas, and D. N. Politis (2003). A full-factor multivariate garch model. *Econometrics Journal 6*, 311–333.

Wu, E. H. C. and P. L. H. Yu (2005). Volatility modelling of multivariate financial time series by using ica-garch models. *Proceedings of the International Symposium on Intelligent Data Engineering and Learning*, 571–579.

Yeredor, A. (2002). Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation. *IEEE Transactions on Signal Processing 50*, 1545–1553.

Young, P. C. and D. Pedregal (1999). Recursive and en-bloc approaches to signal extraction. *Journal of Applied Statistics 26*, 103–128.

Young, P. C., D. Pedregal, and W. Tych (1999). Dynamic harmonic regression. *Journal of Forecasting 18*, 369–394.

Ziehe, A., P. Laskov, G. Nolte, and K.-R. Müller (2004). A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation. *Journal of Machine Learning Research 5*, 777–800.

Ziehe, A. and K.-R. Müller (1998). Tdsep - an efficient algorithm for blind separation using time structure. *Proceedings International Conference on Artificial Neural Networks (ICANN'98)*, 675–680.