Independent Component Analysis Via Copula Techniques

Ray-Bing Chen*
Meihui Guo*
Wolfgang Härdle**
Shih-Feng Huang*

* University of Kaohsiung, Taiwan
** Humboldt-Universität zu Berlin, Germany

BERLIN

ECONOMIC RISK

SFB 649

Independent Component Analysis Via Copula Techniques

Ray-Bing Chen[1], Meihui Guo[2], Wolfgang K. Härdle[3]
and Shih-Feng Huang[2]

[1] *Institute of Statistics, National University of Kaohsiung,*

Kaohsiung, Taiwan, R.O.C.

[2] *Department of Applied Mathematics, National Sun Yet-sen University,*

Kaohsiung, Taiwan, R.O.C.

[3] *Center for Applied Statistics and Economics, Humboldt-Universität zu Berlin,*

Berlin, Germany

November 26, 2007

Abstract

Independent component analysis (ICA) is a modern factor analysis tool developed in the last two decades. Given p-dimensional data, we search for that linear combination of data which creates (almost) independent components. Here copulae are used to model the p-dimensional data and then independent components are found by optimizing the copula parameters. Based on this idea, we propose the COPICA method for searching independent components. We illustrate this method using several blind source separation examples, which are mathematically equivalent to ICA problems. Finally performances of our method and FastICA are compared to explore the advantages of this method.

Key words: Blind source separation, Canonical maximum likelihood method, Givens rotation matrix, Signal/noise ratio, Simulated annealing algorithm.
JEL-Codes: C01, C13, C14, C63

1 Introduction

Independent component analysis (ICA) solves an important and fundamental practical problem: it represents the joint co-movements of multivariate data

series as a linear combination of stochastically independent sources. Unlike Principal Component Analysis, it is based on non-Gaussian stochastic structure of the involved data signals and in fact makes only sense for non-Gaussian multivariate distributions. ICA has been successfully applied in blind source separation (Comon, 1994), image de-noising (Hyvärinnen, 1999) and many other applications. The computational technique to identify the ICA transformation is based on the principle of finding the "best" non-Gaussian directions. This principle can be implemented with different measures of non-Gaussianity. One characteristic of the Gaussian distribution is that its excess kurtosis is zero. Kurtosis based algorithms therefore try to find linear combinations of the original data signals that deviate most from zero excess kurtosis.

Hyvärinnen et al (2001) argued that such a kurtosis based method is necessarily outlier sensitive and may therefore yield practically less realizable independent signals. Their proposed alternative is based on negative entropy and mutual information criteria. Again the principle is to find that projection that is the best non-Gaussian one. The effect of outliers is reduced in that technique since the weight functions in the so-called negentropy criterion have bounded tail behavior.

Non-Gaussianity is not only an important issue in signal processing and other engineering applications but also an important element of financial risk management techniques. The normal distribution lacks the heavy tail behavior that is typical for financial return distributions and cannot model joint tail dependence of several stock returns (Schmidt, 2002). For better modelling of these and other empirical facts of financial data copulae have been introduced into the quantitative finance practice. The copula technique is based on the simple thought that every multivariate distribution can be seen as a coupling of a distribution function (on the unit cube) operating on the marginal distribution functions of each variable. This coupling function has been coined the name "copula", (Sklar, 1959 and 1996). Copulae can be parameterized with low dimensional parameters and fitted to multivariate data by a variety of optimization techniques (Nelsen, 2006). Copulae provide a flexible family for modelling dependencies and include the product copula as the family element representing independence. This property makes them an interesting alternative in ICA.

The idea of this research is to investigate the use of copulae in determining the ICA transformations. The proposed procedure can be summarized in the following. First, one pre-whitens the data as is done for ICA estimation. Second, one fits a copula (with low dimensional parameter space) to a linear transformation of this pre-processed data. Third, one optimizes with respect to the linear transformation the copula parameter so that the current copula

corresponds to the product (independence) copula. This proposed procedure combines ICA ideas from the engineering literature with the empirical copula based research in quantitative finance. We therefore call our technique the COPICA method. Our results are:

(1) The numerical burden to determine the ICA transformation of the COPICA is equivalent to the FastICA method, the current standard method for estimating ICA transformations.

(2) The signal/noise ratios of blind source separation problems with near-Gaussian sources are better reflected by the COPICA method, and sometimes the FastICA method fails to converge in near-Gaussian sources.

(3) The performance of the COPICA is superior to the FastICA for blind source separation with more than one Gaussian sources.

In next section, the ICA method combined with copula techniques, COPICA, is introduced. The 2-dimensional blind source separation examples are demonstrated to illustrate the performance of our method and to start the path to the higher dimensional analysis which we deal with in Section 3. In Section 4, based on blind source separation problems with different kinds of source samples, we compare the performances of COPICA with FastICA in terms of the signal/noise ratio. We discuss the connection and difference of COPICA and FastICA in Section 5. Finally we present a conclusion and outline the perspective of future work in Section 6.

2   COPICA Procedure

Given p-dimensional data, X, our goal is to find linear combinations of the data such that the resulting components are nearly independent. Herein, we propose a novel method to find the coefficients of the linear combination of independent component via copula technique. The proposed procedure, called COPICA, contains two main steps. The first step is to whiten the p-dimensional data, X, which is the same as in the FastICA procedure. That is to find a whitening matrix W such that the components of $Z = WX$ are uncorrelated and $Cov(Z) = I_p$. One popular method to get the whitening matrix W is to set $W = S^{-1/2}$, where S is the covariance matrix of the data. In the second step, we seek for a linear transformation, R, of the whitened data, Z, such that the outputs $Y = RZ$ are as statistically independent as possible. The proposed method is outlined as following:

1. Whiten the observations by W first, then

2. Rotate the whitened observations by multiplying R to achieve independence.

In an ICA model, the following two ambiguities are well known to hold. (1) Since we can freely change the order of the terms, and call any of the independent components the first one, we cannot determine the order of the independent components. This ambiguity is insignificant in most applications. (2) We cannot determine the variances of the independent components. Thus, without loss of generality we assume that each component of Y has unit variance, i.e. $Cov(Y) = I_p$. Since the matrix R satisfies

$$RR^\top = RCov(Z)R^\top = Cov(RZ) = Cov(Y) = I_p,$$

it is an orthogonal matrix and therefore can be represented as the product of the following Givens rotation matrices, for example see Hastings (1970),

$$R = \prod_{1 \le i < j \le p} G_{ij}(\beta_{ij}). \tag{1}$$

The matrix $G_{ij}(\beta_{ij})$ is a p-dimensional Givens rotation matrix which represents a rotation in the plan spanned by the axes x_i and x_j, $i < j$, with angle β_{ij}. Specifically, $G_{ij}(\beta_{ij})$ is obtained by modifying the identity matrix so that the (i, i), (i, j), (j, i) and (j, j) elements of this matrix are respectively $\cos \beta_{ij}$, $\sin \beta_{ij}$, $-\sin \beta_{ij}$, and $\cos \beta_{ij}$, where $\beta_{ij} \in [0, 2\pi)$. Finally the product of R and W, $B = RW$, is the objective transformation matrix. Clearly, the matrix R is determined by the rotation angles, β_{ij}, and the main problem now is to choose the angles such that the components of $Y = BX = RWX$ are nearly independent.

Dependencies among the components is used as the criterion to find the "best" rotation angles. Unlike the conventional ICA methods using mutual information, kurtosis or negentropy to measure dependence, we adopt copula parameters as measure of dependence. Copula has recently become the most significant new tool to handle the co-movement between markets in the field of finance, since it provides a flexible way to connect the marginal distributions of individual component to their multivariate joint distribution. For example, if assume the marginal distribution of $X = (X_1, \ldots, X_p)^\top$ is $F_{X_j}(x_j; \delta_j)$, where δ_j is the marginal parameter, $j = 1, \ldots, p$, then we can use a copula function C to model the joint distribution $F_X(x_1, \cdots, x_p)$ of $(X_1, \cdots, X_p)$, by setting

$$F_X(x_1, \cdots, x_p) = C(F_{X_1}(x_1; \delta_1), \ldots, F_{X_p}(x_p; \delta_p); \theta), \tag{2}$$

where $\theta = (\theta_1, \ldots, \theta_d)$ is the copula parameter. Then the joint density of $(X_1, \cdots, X_p)$ is given by

$$f(x_1, \cdots, x_p; \delta_1, \ldots, \delta_p, \theta) = c(F_{X_1}(x_1; \delta_1), \ldots, F_{X_p}(x_p; \delta_p); \theta) \prod_{j=1}^{d} f_j(x_j; \delta_j),$$

4

where c is a copula density and f_j's are the marginal densities of X_i's. A special copula is the product copula, defined as $C(u_1, \cdots, u_p) = u_1 \cdots u_p$. By Eq.(2), we have $F_X(x_1, \cdots, x_p) = F_{X_1}(x_1; \delta_1) \cdots F_{X_p}(x_p; \delta_p)$ under the product copula assumption, which is correspond to the case when X_j's are independent. This explains why the product copula is also called the independence copula. In the following, we introduce the copulae considered in this work.

(1) Gaussian copula:

$$C(u_1, \cdots, u_p) = \Phi_R(\Phi^{-1}(u_1), \cdots, \Phi^{-1}(u_p)),$$

where Φ_R is the standardized multivariate normal distribution with correlation matrix R and Φ is the distribution of $N(0, 1)$. In particular, if the correlation matrix is an identity matrix, then Gaussian copula is the independence copula. Furthermore, if the marginals of X_j's are also normal distributed, then Gaussian copula is correspond to the multivariate normal distribution.

(2) Clayton copula:

$$C(u_1, \cdots, u_p) = \Big(\sum_{j=1}^{p} u_j^{-\theta} - p + 1 \Big)^{\frac{-1}{\theta}}, \quad \theta > 0.$$

As $\theta \to 0$, the Clayton copula approaches to the independence copula. Furthermore, the Clayton copula can model for multivariate lower tail dependence. For bivariate case, the lower tail dependence of two variables X_1 and X_2 is defined as $\lambda_L = \lim_{v \to 0^+} P(F_{X_2}(X_2) \leq v \mid F_{X_1}(X_1) \leq v) = \lim_{v \to 0^+} \frac{C(v, v)}{v}$. If $\lambda_L > 0$, then X_1 and X_2 are said to have lower tail dependence. For bivariate Clayton copula, we have $\lambda_L = 2^{-1/\theta} > 0$, for $\theta > 0$, and thus one can use Clayton copula to model the data with lower tail dependence property.

(3) Gumbel copula:

$$C(u_1, \cdots, u_p) = \exp\Big\{ - \Big[\sum_{j=1}^{p} (-\ln u_j)^\theta \Big]^{\frac{1}{\theta}} \Big\}, \quad \theta \geq 1.$$

If $\theta = 1$, we obtain the independence copula as a special case. In contrast to the Clayton copula, the Gumbel copula can model upper tail dependence. For bivariate case, the upper tail dependence is defined as $\lambda_U = \lim_{v \to 1^-} P(F_{X_2}(X_2) > v \mid F_{X_1}(X_1) > v) = \lim_{v \to 1^-} \frac{1 - 2v + C(v, v)}{1 - v}$. If $\lambda_U > 0$, then there exists upper tail dependence. For bivariate Gumbel copula, we have $\lambda_U = 2 - 2^{1/\theta} > 0$, for $\theta > 1$, and thus one can use Gumbel copula to model upper tail dependent data.

Figure 1: Bivariate plots of Clayton and Gumbel copulae with $\theta = 3$ and N(0,1) marginals.
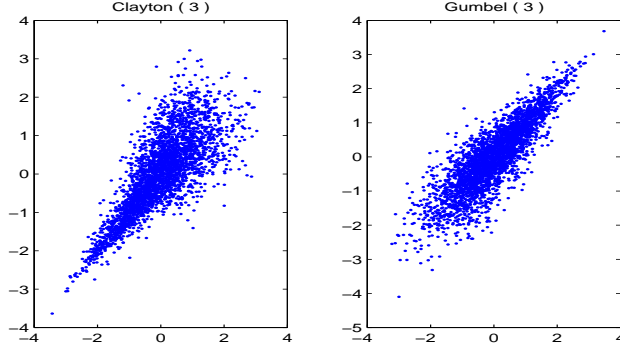
Further note that the bivariate Gaussian copula can neither model upper nor lower dependence, unless the correlation coefficient $\rho = 1$. Specifically, $\lambda_U = \lambda_L = 0$ for $\rho < 1$ and $\lambda_U = \lambda_L = 1$ for $\rho = 1$. Since the Gaussian, Clayton and Gumbel copulae model different tail dependence structures, we adopt the parameters of these three copulae in the criterion of independence to enhance the adaptability of our method. In Figure 1, we give the bivariate plots of random samples generated from Clayton and Gumbel copulae with N(0,1) marginals, respectively. Although the marginal distributions of the two cases are the same, different tail dependencies are demonstrated.

If the multivariate dependence is modeled by a single copula, then the magnitude of the divergence function $o(\theta) = \|\theta - \theta_0\|$ can be used to measure the strength of the dependency among the variables, where θ and θ_0 denote the copula parameters of the data and the independence case, respectively. For example, $o(\theta) = \|\theta - 1\|$ for the Gumbel copula. If we adopt d copula models for the given data set, then the independency among the given data can be measured by a weighted sum of the individually independent measures, i.e.

$$O(\theta_1, \ldots, \theta_d) = \sum_{i=1}^{d} w_i o_i(\theta_i), \tag{3}$$

where $o_i(\theta_i) = \|\theta_i - \theta_{i0}\|$, θ_i and θ_{i0} are respectively the fitted parameter and the independent parameter value of the ith copula model, and w_i's are the positive weights. Apply to the ICA problem, the goal is to find the rotation angles β_{ij} minimizing $O(\hat{\theta}_1, \ldots, \hat{\theta}_d) = O(\beta_{12}, \ldots, \beta_{(p-1)p})$, where $\hat{\theta}_i$ is the estimate of the ith copula parameter, $i = 1, \cdots, d$. Therefore, our ICA problem is transformed into the following constrained minimization problem,

$$\min_{\beta_{ij}, 1 \le i < j \le p} O(\beta_{12}, \ldots, \beta_{(p-1)p}).$$

6

Due to constraints of copula parameter estimation method, it is in general difficult to have close form expression of $O(\beta_{12}, \ldots, \beta_{(p-1)p})$ even for low dimensional case. Derivative-free optimization methods, such as genetic algorithm, simulated annealing algorithm, direct search method and so on, can aid to solve this minimization problem. Herein, the simulated annealing algorithm, proposed by Metropolis et al. (1953) and introduced as an optimization technique by Kirkpatrick et al. (1983), is used to search the optimal angles. For simplicity of notation, we denote $\beta_{12}, \ldots, \beta_{(p-1)p}$ by $\beta_1, \ldots, \beta_q$, where $q = p(p-1)/2$. Since the target is to minimize $O(\beta)$, first define a density

$$\pi_{T(t)}(\beta) \propto \exp(-O(\beta)/T(t)),$$

where $\beta = (\beta_1, \ldots, \beta_q)$, and $T(t)$ is the "temperature" at time t and is a decreasing function from initial temperature, $T(0) > 0$, to 0^+. Following Liu (2001), our simulated annealing (SA) algorithm is as follows.

1. Select the initial angles, $\beta_i^{(0)}$, $i = 1, \ldots, q$.

2. Run N_t iterations of the Gibbs sampler to sample β from $\pi_{T(t)}(\beta)$, and at each iteration of the Gibbs sampler, draw β_i, $i = 1, \ldots, q$, from $\pi_{T(t)}(\beta_i|\beta_{-i})$.

3. Set $t = t + 1$, go to step 1 until t is large enough.

Here β_{-i} is the set of all angles β_j except the ith angle, i.e. $\beta_{-i} = (\beta_1, \ldots, \beta_{i-1}, \beta_{i+1}, \ldots, \beta_q)$. For more details about the simulated annealing algorithm, see Liu (2001).

Here the COPICA procedure used to find the independent components of a given data set, X, is summarized in the following:

1. Center the data to make its mean zero and compute the sample covariance matrix, S.

2. Whiten the data to give Z by multiplying X by $W = S^{-1/2}$.

3. Choose d copulae and define the objective function, $O(\theta_1, \ldots, \theta_d)$.

4. Apply SA algorithm to find the rotation matrix $R = \prod_{1 \leq i < j \leq p} G_{ij}(\beta_{ij})$ by minimizing $O(\hat{\theta}_1, \ldots, \hat{\theta}_d)$ with respect to the rotation angles, β_{ij}.

5. Multiple R with W, i.e. $B = RW$.

3   Simulations and Real Examples on Blind Source Separation

Now we illustrate the performance of our COPICA by solving blind source separation (BSS) problems. In BSS problems, the observations are assumed

to be the mixtures of the p independent sources. Let A be a $p \times p$ invertible mixing matrix. Then we have the following linear mixing equation,

$$\mathbf{x}_t = A\mathbf{s}_t, t = 1, \ldots, T, \tag{4}$$

where $\mathbf{x}_t = (x_{1t}, \ldots, x_{pt})^\top$ is the observation vector at time t, and $\mathbf{s}_t = (s_{1t}, \ldots, s_{pt})^\top$ is the source vector at time t whose components, $s_{1t}, \ldots, s_{pt}$, are mutually independent. The goal of the BSS problem is that only given the mixtures, $\mathbf{x}_t, t = 1, \ldots, T$, we want to estimate the mixing matrix A and recover the original sources $\mathbf{s}_t, t = 1, \ldots, T$, simultaneously. Since A is invertible, then we focus on the inverse of A, because the independent sources can be recovered by $A^{-1}\mathbf{x}_t, t = 1, \ldots, T$. Thus BSS problems is closely related to ICA problems. In fact, when we apply ICA methods to BSS problems, the true inverse matrix, A^{-1}, might not be obtained directly, however, the transformation matrix, B, is the inverse of the true mixing matrix, A, up to a permutation and scale change. Here in our simulations, we assume that the sources $\mathbf{s}_t$ are at each time instant mutually and come from an identical heavy-tail distribution, for example, double exponential distribution or mixture normal distribution. Then given the true mixing matrix, A, the observations, $\mathbf{x}_t, t = 1, \ldots, T$, are mixed by ourselves.

Applying COPICA to BSS, we need to pre-specify copulae, and then to estimate these corresponding copula parameters, θ_i for divergence functions, $o_i(\hat{\theta}_i)$. Here canonical maximum likelihood (CML) method is adopted to estimate the copula parameter. The first step of CML method is to estimate the marginals via the empirical distributions, denoted by $\hat{F}_{X_i}(x_{it})$, $i = 1, \cdots, p$. Then estimate the copula parameters via

$$\hat{\theta}_{CML} = \arg\max_{\theta} \sum_{t=1}^{T} \ln c(\hat{F}_{X_1}(x_{1t}), \cdots, \hat{F}_{X_p}(x_{pt}); \theta)),$$

where c is the copula density.

In order to measure the performance of our COPICA, we compute the inverse of the transformation matrix, B, whose columns should be proportional to the true mixing matrix, and in addition to, we also calculate the signal/noise ratio (SNR) value by

$$SNR_{s_i}(\hat{s}_i)[dB] = 10 \log_{10} \frac{\|s_i\|^2}{\|s_i - \hat{s}_i\|^2}, \tag{5}$$

where $s_i = (s_{it}, t = 1, ..., T)$, $i = 1, ..., M$, are original signals from the sources; $\hat{s}_i = (\hat{s}_{it}, t = 1, ..., T)$, $i = 1, ..., M$, are recovered signals, and $\|\mathbf{x}\|$ is the two-norm of the vector $\mathbf{x}$. According to this definition of SNR value, the larger SNR values are, the better performance we have.

Figure 2: The tendency of the objective function for the simulation.
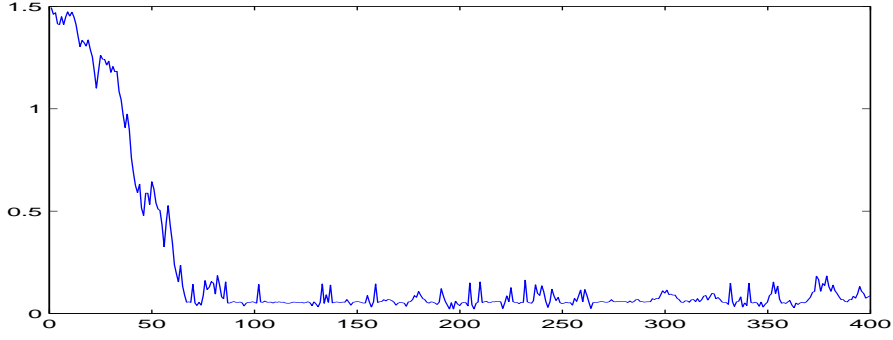
3.1 Blind source separation with two sources

First a two-dimensional simulation is illustrated. We independently generate two double exponential samples with 2000 points for each source, and the mixing matrix is set to be

$$\begin{pmatrix} 0.8000 & 1.0000 \\ -0.6000 & 1.0000 \end{pmatrix}. \tag{6}$$

In this simulation, two copulae are chosen, one is Gumbel copula, and another one is Clayton copula. The objective function is defined as

$$O(\theta) = 10(\theta_G - 1) + |\theta_C|,$$

where $\theta = (\theta_G, \theta_C)$, θ_G is the parameter of Gumbel copula, and θ_C is the parameter of Clayton copula. Both parameters are all estimated by canonical maximum likelihood method. As mentioned before, we need to white the data first. After whitening, the estimations of these two parameters are $\hat{\theta}_G = 1.0639$ and $\hat{\theta}_C = 0.0850$. Then we apply COPICA to search the independent components. After 400 steps, the trend of the objective function is shown in Figure 2, and the minimum value of the objective function is 0.0221. At this time, the corresponding parameter estimations are $\hat{\theta}_G = 1.0022$ and $\hat{\theta}_C = 1e^{-6}$, and the inverse of the transformation matrix, B, is

$$B^{-1} = \begin{pmatrix} 0.2841 & 0.3600 \\ -0.2057 & 0.3601 \end{pmatrix},$$

whose two columns are all proportional to the corresponding columns in the true matrix, A. To compare the estimation of the mixing matrix, we suggest to use the ratios, $A_{k1}/A_{k2}, k = 1, 2$, to compare B^{-1} and A. In this case, the ratios of B^{-1} are 1.3812 and -0.9997, which are close to the ratios in A, 1.3333 and -1.0000. The original source samples and recovered signals are shown in

9

Figure 3: The simulation results for blind source separation problem. The green lines are the mixtures; the red lines are the original double exponential sources, and the blue lines are the corresponding recovered signals.
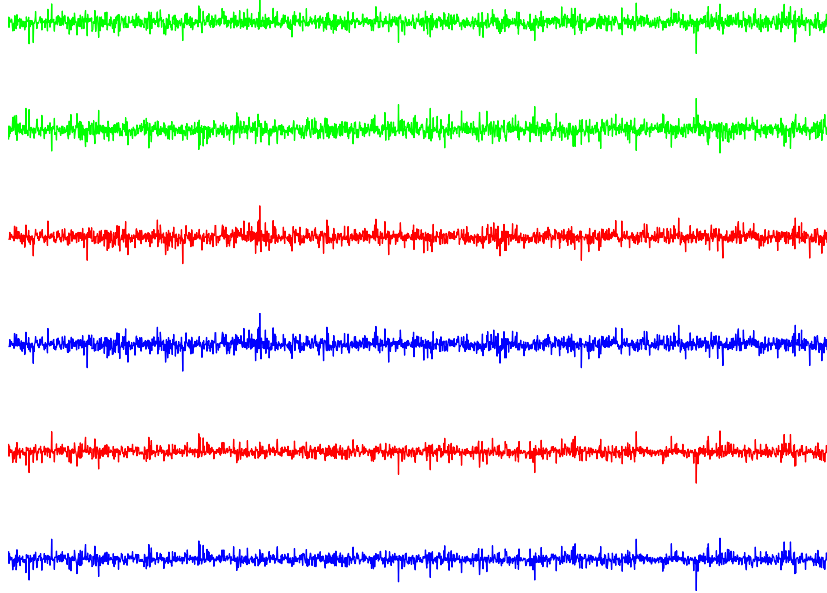

Figure 3. The SNR values for each recovered signals are also computed here, and the values are 98.4980 and 83.8026 for each source respectively.

Next two human speeches with 10000 samples are used as our original sources, and in this experiment, our observations are mixed by the same mixing matrix, Eq. (6). Here to speed up the convergence of the objective function, $O(\theta)$ is modified as

$$O(\theta) = \exp\{100(\theta_G - 1) + |\theta_C|\}.$$

After 100 steps of our COPICA, the objective function is stable, and we find that

$$B^{-1} = \begin{pmatrix} 0.0970 & 0.1186 \\ -0.0736 & 0.1187 \end{pmatrix},$$

and the SNR values are 156.2844 and 108.9266. Figure 4 shows the original human speeches and the recovered sounds by COPICA.

3.2 Blind source separation with more sources

Here blind source separation problems with more sources are considered. In two dimensional cases, the objective function is constructed by mixing two

Figure 4: The numerical results for blind source separation problem with two human speeches. The green lines are the mixtures; the red lines are the original human speeches, and the blue lines are the recovered signals.
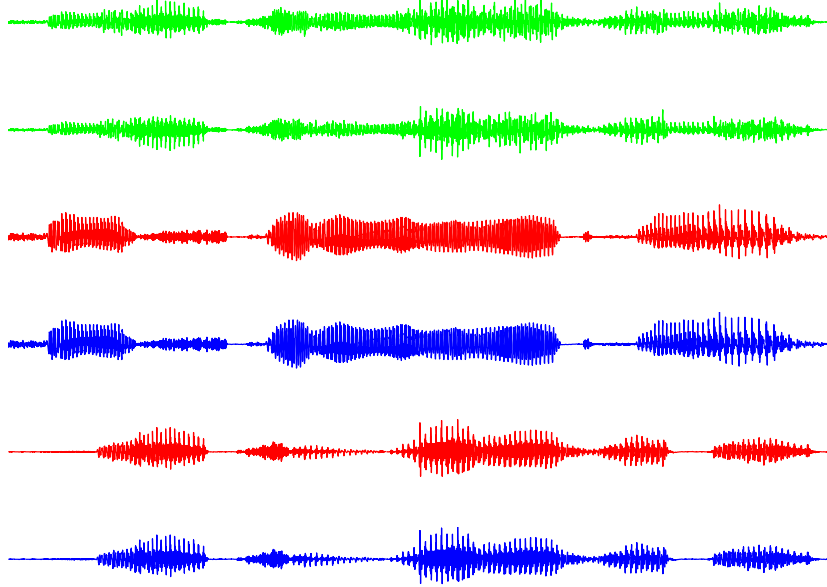
divergence functions of the parameters of two-dimensional copulae. Hence a natural extension of COPICA for p-dimensional BSS problem is that our objective function is also defined as the mixture of several divergence functions of p-dimensional copula parameters. However, this approach fails for the high dimensional situation. The reason might be due to the weakness of the copula and the estimation method of the copula parameter. In order to successfully extend COPICA for solving higher dimensional BSS problem, we suggest that our objective function should not only contain the divergence functions of p-dimensional copula parameters but also need to mix with lower dimensional copula parameters. For example, when three dimensional case is considered, besides the three dimensional copulae, two dimensional copulae are also chosen for fitting any two marginal samples. Here three dimensional BSS problems are used to illustrate our approach. Two simulations are shown first with different source generators, and then an example with three natural sounds is displayed.

In the first simulation, three original sources are independently generated from double-exponential distribution with the corresponding parameter, $\lambda = 5$.

Figure 5: The simulation results for three dimensional blind source separation problem with independent double exponential sources. The red lines are the original double exponential sources, and the blue lines are the recovered signals.
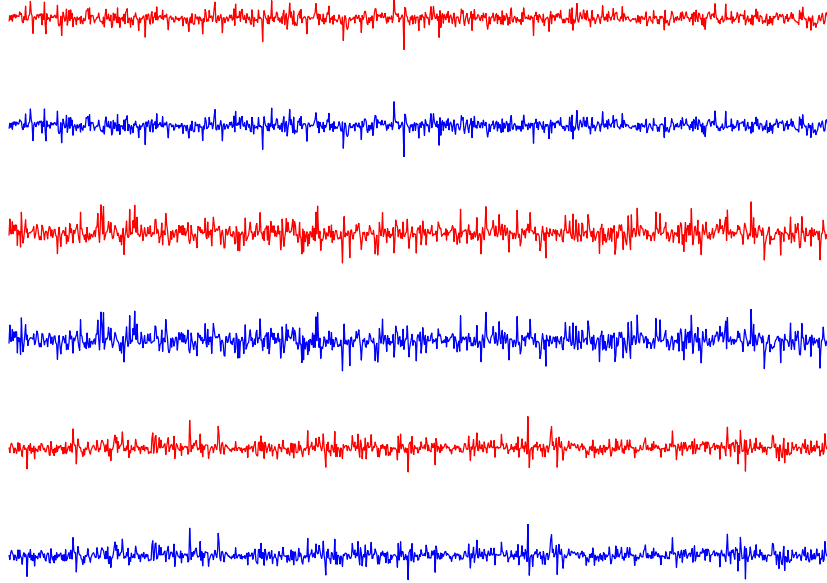

The mixing matrix is set to be

$$A = \begin{pmatrix} 1.0000 & -2.0000 & -1.0000 \\ -1.0000 & 1.0000 & 2.0000 \\ -1.0000 & 1.0000 & 1.0000 \end{pmatrix}. \tag{7}$$

Applying our COPICA to this case, three different copulae are used for modeling the data, Gumbel copula, Clayton copula and Gaussina copula. Here one three-dimensional Gumbel copula is adopted, and two-dimensional Gumbel, Clayton and Gaussian copulae are also used to measure the dependency among any two of three marginal samples. The notation $\theta_{123,C}$ is used to denote the parameter of the three-dimensional copula, C, and $\theta_{ij,C}$ is the parameter of the two-dimensional copula, C, for the ith and the jth current marginal samples. In this simulation, our objective function is defined to be

$$
\begin{aligned}
O(\theta) \;=\; & \exp[200 * (\theta_{123,Gumbel} - 1)] + \exp[200 * \sum_{i<j}(\theta_{ij,Gumbel} - 1)] \\
& + \exp[200 * \sum_{i<j}(\theta_{ij,Clayton})] + \exp[200 * \sum_{i<j}|\theta_{ij,Gaussian}|], \quad (8)
\end{aligned}
$$

where $\theta = \{\theta_{123,Gumbel}, \theta_{ij,Gumbel}, \theta_{ij,Clayton}, \theta_{ij,Gaussian}, 1 \leq i < j \leq 3\}$. Thus

12

Figure 6: The simulation results for a three dimensional blind source separation problem with mixture normal sources. The red lines are the original sources, and the blue lines are the recovered signals.
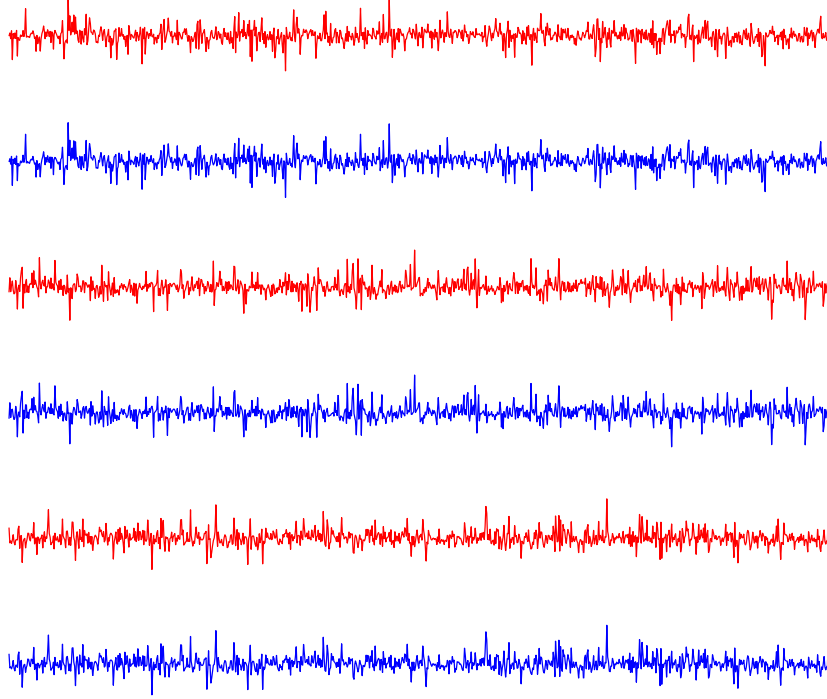
our goal is to minimize this objective function with respect to the rotation angles, β_{12}, β_{13} and β_{23}. Then after 400 steps of our COPICA algorithm, the inverse matrix of B is

$$B^{-1} = \begin{pmatrix} 0.3799 & -0.5365 & -0.2617 \\ -0.3739 & 0.2439 & 0.5301 \\ -0.3484 & 0.2524 & 0.2542 \end{pmatrix}.$$

Since each column of this matrix is proportional the corresponding column of the true mixing matrix, we believe COPICA successfully solve this simulated BSS problem. Figure 5 shows the original source signals and our recovered signals, and the corresponding SNR values are $40.7656, 42.1369$ and 46.6162.

Three original sources in the second simulation are generated independently from a mixture normal distribution,

$$s_i \sim 0.7N(0,1) + 0.3N(0,3^2).$$

That is for each sample point, it is sampled from standard normal distribution with probability 0.7 and from the normal distribution with variance 9 with probability 0.3. The true mixing matrix is also set to be the same as Eq. (7).

Applying our COPICA to this simulation, three different copulae are also used for modeling the data, Gumbel copula, Clayton copula and Gaussina copula. The our objective function is set to be

$$
\begin{aligned}
O(\theta) \;=\; & \exp[200 * (\theta_{123,Gumbel} - 1) + 300 * \sum_{i<j}(\theta_{ij,Gumbel} - 1) \\
& + 200 * \sum_{i<j}(\theta_{ij,Clayton}) + 500 * \sum_{i<j}|\theta_{ij,Gaussian}|].
\end{aligned}
\tag{9}
$$

In this objective function, unlike the equal wights in previous objective function, Eq. (8), the weights are proportional to the inverse of the standard deviations of the CML estimations for the corresponding copula parameters. Then after 100 steps of our COPICA algorithm,

$$
B^{-1} = \begin{pmatrix} 2.3435 & -3.5881 & -1.9690 \\ -2.1897 & 1.6172 & 3.6652 \\ -2.1347 & 1.7093 & 1.8532 \end{pmatrix}.
$$

Since each column of this matrix is proportional the corresponding column of the true mixing matrix, we believe COPICA successfully solve this simulated BSS problem. Figure 6 shows the original source signals and our recovered signals, and the corresponding SNR values are $57.1508, 41.3515$ and 55.8527.

Finally three natural sounds of thunder, water and fire are used as our original signals, and there are 5000 sample points for each sound data. With the same mixing matrix, Eq. (7), and Eq. (9) as our objective function, after 100 steps of COPICA, we find that

$$
B^{-1} = \begin{pmatrix} 1.8152 & -2.1149 & -0.8616 \\ -1.8085 & 1.0665 & 1.6384 \\ -1.8360 & 1.0706 & 0.8641 \end{pmatrix},
$$

and the corresponding SNR values are $57.0622, 79.6792$ and 67.3711. Figure 7 shows the original natural sounds and our recovered signals.

4   Comparisons

The FastICA (Hyvarinen and Oja, 1997) is a widely used and efficient method for identifying independent components. The FastICA is also a two-step method. After whitening the data as the first step, the FastICA is based on a fixed-point iteration scheme for finding the components with maximum of the nongaussianity, which are the independent components. Here kurtosis or negentropy is used as the measure of nongaussianity. The computer program of the FastICA is available at the web-site, `http://www.cis.hut.fi/projects/ica/fastica/`.

Figure 7: The numerical results for three dimensional blind source separation problem with three natural sounds. The red lines are the original sounds; green lines are their mixtures, and the blue lines are the recovered signals.
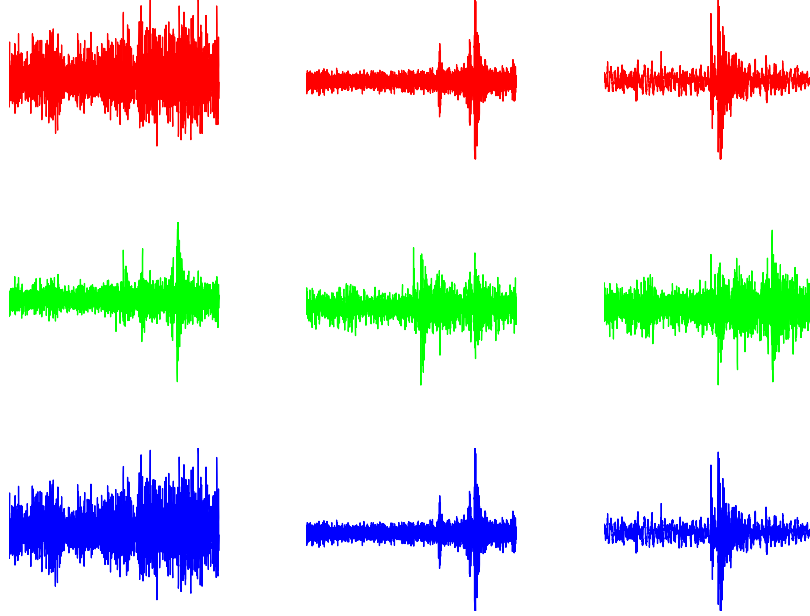
In this section, we compare COPICA with FastICA through BSS problems that original independent sources are generated from mixture normal distributions with different values of kurtosis. Let X follow a mixture normal distribution, i.e. $X \sim pN(0, \sigma_1^2) + (1-p)N(0, \sigma_2^2)$. Then the kurtosis of X is

$$3\{p\sigma_1^4 + (1-p)\sigma_2^4\}/\{p\sigma_1^2 + (1-p)\sigma_2^2\}^2.$$

Therefore, here we call a sample is heavy-tail if the value of kurtosis is larger than 5, and the near-Gaussian sample if the kurtosis is less than 4. Thus a heavy-tail sample or a near-Gaussian samples can be drawn from the mixture normal distribution by choosing the different values of p, σ_1 and σ_2. In the following comparison, (p, σ_1, σ_2) is set to be either $(0.7, 1, 3)$, $(0.6, 1, 3)$, $(0.2, 1, 3)$, $(0.15, 1, 3)$ and $(0.2, 1, 2)$, and the corresponding value of kurtosis is shown in Table 1.

Here the mixing matrix is fixed as Eq. (7), and then after generating the original sources with 1000 sample points for each, the observations are mixed according to Eq. (4). The SNR values are used as the criterion for the comparison. That is we count the number of sources that the SNR values

15

Table 1: The values of kurtosis of our source generators

Heavy-tail		Near-Gaussian	
(p, σ_1, σ_2)	Kurtosis	(p, σ_1, σ_2)	Kurtosis
(0.7,1,3)	6.4879	(0.2,1,3)	3.5610
(0.6,1,3)	5.6122	(0.15,1,3)	3.4024
		(0.2,1,2)	3.3737

Table 2: The SNR ratios of COPICA/FastICA for the heavy-tail sources. The ratios greater than 1 are in bold font.

Cases	$(p, \sigma_1, \sigma_2) = (0.7, 1, 3)$			$(p, \sigma_1, \sigma_2) = (0.7, 1, 3)$		
	Source 1	Source 2	Source 3	Source 1	Source 2	Source 3
1	**58.8/45.2**	**40.9/35.3**	**51.8/46.3**	**48.5/34.8**	**54.6/30.3**	57.8/80.0
2	48.2/50.5	**63.7/49.7**	**65.6/43.4**	**56.0/45.4**	44.9/47.5	**48.8/48.0**
3	36.5/43.9	50.5/73.0	**64.9/55.8**	36.7/44.3	45.1/48.2	**63.9/30.0**
4	**63.3/59.3**	**62.8/41.0**	**59.1/42.6**	**71.3/44.7**	38.0/39.5	37.4/39.1
5	**53.2/41.8**	**50.6/43.1**	54.8/70.1	39.8/56.5	**53.5/24.7**	**58.6/24.4**
6	**54.5/38.6**	49.8/55.3	49.9/50.9	41.1/71.2	46.9/55.4	50.7/59.3
7	**57.3/51.1**	49.4/53.0	**51.6/49.1**	**46.1/40.4**	51.1/93.1	**44.8/35.8**
8	42.8/57.1	52.9/62.4	45.4/46.9	**43.1/40.9**	**69.6/51.9**	**56.2/33.8**
9	**56.7/50.9**	**59.0/56.3**	62.6/71.0	**45.4/35.5**	**52.9/27.9**	**54.6/35.3**
10	**57.2/40.0**	41.4/57.6	**55.9/40.2**	**51.6/38.2**	**49.3/48.0**	**46.4/46.2**

produced by COPICA are better than those done by FastICA. The results are shown as follows:

Heavy-tail sources. Two different heavy-tail samples are generated from mixture normal distribution with $\sigma_1 = 1$; $\sigma_2 = 3$, and $p = 0.7$ or 0.6. For each type of original sources, we repeat the same simulation ten times, and for each replication, we re-generate the source samples according to the same mixture normal distribution. The COPICA and the FastICA are then applied to separate the source signals from their mixtures. As mentioned before, for COPICA, we need to pre-specified the copulae used here and defined the corresponding objective function, $O(\theta)$. Here $O(\theta)$ is set to be the same as Eq. (9). The SNR ratios of COPICA/FastICA for the heavy-tail sources are shown in Table 2. When $p = 0.7$, we see from the left panel of Table 2 that COPICA provides the better recovered results in cases 1,2,4,5,7,9 and 10, and there are 18 sources out of total 30 sources that SNR values produced by COPICA are larger than those produced by FastICA. When $p = 0.6$, the similar results are displayed in the right panel of Table 2. Thus it seems that the performance of COPICA

Table 3: The SNR ratios of COPICA/FastICA for the near-Gaussian sources. The ratios in favor of COPICA are in bold font.

	$(p, \sigma_1, \sigma_2) = (0.2, 1, 3)$			$(p, \sigma_1, \sigma_2) = (0.15, 1, 3)$		
Cases	Source 1	Source 2	Source 3	Source 1	Source 2	Source 3
1	**79.5/29.8**	**36.7/35.4**	**32.3/26.3**	**21.9/21.4**	**21.2/19.1**	**23.9/15.2**
2	35.1/38.6	**45.9/31.6**	**29.0/25.8**	**33.7/(-10.9)**†	**57.4/ 2.2**	**37.3/15.9**
3	**36.7/34.8**	37.6/45.7	**49.5/38.1**	23.4/31.7	**52.8/36.1**	25.1/33.1
4	26.5/57.0	34.8/61.9	25.5/77.6	25.9/33.2	**33.7/26.1**	**28.2/27.1**
5	32.4/37.3	**35.5/33.2**	**62.7/49.0**	**18.6/12.2**	**34.2/20.1**	**27.3/23.5**

	$(p, \sigma_1, \sigma_2) = (0.2, 1, 2)$		
Cases	Source 1	Source 2	Source 3
1	**24.2/NA**‡	**43.1/NA**‡	**29.4/NA**‡
2	**41.5/21.8**	**44.9/12.5**	**60.9/13.0**
3	40.8/45.5	28.5/34.5	31.6/39.3
4	**26.3/NA**‡	**21.9/NA**‡	**28.1/NA**‡
5	**44.9/30.9**	**29.0/28.2**	**26.3/24.9**

$\dagger$ Negative SNR value means that FastICA cannot identify the correct independent component.

$\ddagger$ "NA" means that at that case, FastICA could not find the independent components.

is similar as that of FastICA in our replications with heavy-tail samples.

Near-Gaussian sources. Three types of near-Gaussian sources are generated from the independent mixture normal distribution with different setting of parameters. For each type of source, we repeat the simulation five times and we also re-generate the independent sources at each replication. Then the COPICA and the FastICA are used to find the independent components for every replications. Here Eq. (8) is the objective function of our COPICA. The SNR ratios of COPICA/FastICA for the each case are shown in Table 3. From Table 3, regardless of what type of source we generate, COPICA always outperforms than FastICA, because COPICA provides better SNR values for more than half of sources. When $(p, \sigma_1, \sigma_2) = (0.15, 1, 3)$, in the second case, the negative SNR value means that FastICA fails to recover the signals from their mixtures. In the bottom panel of Table 3, there are two cases that FastICA cannot separate any one original sources from the observations. However, COPICA successfully solve these BSS problems with near-Gaussian samples.

We summery the comparison results here. No matter what kind of sources we generate, COPICA can successfully identify all three independent components. FastICA works well for heavy-tail sources, but may fail for the cases of near-Gaussian sources. The reason should be due to the limitation of the

kurtosis and negentropy. However, the signals whose kurtosis is close to 3 does exist naturally. For example, the sample kurtosis of thunder sound in Section 3.2 is 3.5323. Thus, we would suggest to use our COPICA to solve the BSS problems.

5 COPICA and FastICA

In Section 4, we numerically compare the performances of COPICA and FastICA through the blind sources separation with different independent sources. Here, we would like to note the connection and difference between FastICA and COPICA from the point of view of their methods.

Search approach: FastICA is closely related to projection pursuit (Jones and Sibson, 1987). Basically FastICA is performed by finding the most non-Gaussian projections of the data. According to the algorithm of FastICA, the independent components are found "one by one". That is the FastICA algorithm is a one-unit algorithm. When we have $p-1$ independent components, we can run FastICA algorithm for the pth independent component. However, we must orthogonalize this current component with the previous components at each iteration of the algorithm. Of course, we can use FastICA to search the whole independent components in parallel, but at each iteration of the FastICA algorithm, it is still necessary to orthogonalize all the components by a symmetric orthogonalization method.

In COPICA, we have shown the matrix R is an orthogonal matrix. Here we set R as the product of the Givens rotation matrices with different angles, and then these angles are determined by minimizing an objective function which is constructed by parameters of copulae. Thus we search these independent components "simultaneously" without orthogonalizing the vectors at each iteration of the COPICA algorithm. In fact, our search approach can be easy modified into FastICA algorithm by changing the corresponding objective function to be as the measure of kurtosis or negentropy of data.

The measure of independence: The fundamental principle of the FastICA is "Non-Gaussian is independent". Then independent components are found by maximizing non-Gaussianity, and the possible non-Gaussian measures are kurtosis and negentropy. However, the successfulness of this principle is based on the linear mixing model assumption, Eq (4), and the central limit theory. Thus, when this mixing model holds, FastICA can be used to identify independent components. Otherwise, what

18

FastICA provide are only the "projection pursuit directions".

In COPICA, it is not necessary to have the mixing model assumption for observations. After whitening, COPICA is to find a rotated orthogonal coordinate system such that the elements in new coordinates are as independent as possible, and here the independencies among the components are measured directly by the parameters of the corresponding copulae. Thus COPICA can be treated as a direct generalization of the principal component analysis (PCA).

More than one Gaussian source: A limitation of FastICA is that there is at most one Gaussian source. When there are more than one Gaussian source, it is hard to separate original sources from their mixtures by only maximizing the non-Gaussianity. However, in COPICA, since the CML method is used here to estimate copula parameters, we do not need the explicit density assumptions of original sources. Thus we still have chance to separate sources from their mixtures even though there are more than one Gaussian source.

One BSS example with more than one Gaussian source is illustrated here. Three original sources are generated as follows: The first source signals are the normal sample with $\sigma^2 = 4$; the second source signals are generated from a double exponential distribution with $\lambda = 1$, and the third source are the normal sample with $\sigma^2 = 2$. The observation is mixed according to Eq. (4) with the same mixing matrix A in Eq. (7). The objective function is defined as Eq. (9). Applying COPICA, we obtain that the inverse matrix of B is

$$B^{-1} = \begin{pmatrix} 1.8536 & -3.0923 & -1.4786 \\ -1.7672 & 1.5700 & 2.9192 \\ -1.8587 & 1.5696 & 1.5093 \end{pmatrix}.$$

Since each column of this matrix is proportional the corresponding column of the true mixing matrix, we believe COPICA successfully solve this BSS problem. The corresponding SNR values are $52.7718, 77.6114$ and 54.1027. However, when FastICA is applied to this example, the result is

$$B_F^{-1} = \begin{pmatrix} 1.3250 & -2.9654 & -2.1496 \\ -0.8216 & 1.3606 & 3.4014 \\ -1.3251 & 1.4444 & 2.0847 \end{pmatrix},$$

and the SNR values of each recovered signals are $20.4576, 62.7093, 18.9869$. That is FastICA identify that non-Guassian source but fail to separate the other two Gaussian sources.

6   Conclusions

In this article, a new ICA method, COPICA, is proposed. COPICA is a two-steps procedure which is similar to FastICA. After whitening the data, COPICA project the whiten data into the p-dimensional plan simultaneously, and this projection is chosen in terms of the parameters of the pre-specified copulae. Thus in COPICA, ICA problem is transformed to be a minimization problem whose objective function is constructed by the linear combination of the divergence functions of copula parameters, and here the minimization is accomplished by a simulated annealing algorithm.

When dimensionality of data, p, is large than 2, only p-dimensional copula with one corresponding parameter could not be used to find independent components. Besides p-dimensional copulae, we suggest to cooperate with the other low dimensional copulae. This approach is reasonable, because any subset of independent components are still statistically independent. For example, when $p = 3$, our objective function is contained a three dimensional copula and three two dimensional copulae. Here we demonstrate that COPICA successfully solve BSS problems for $p = 2$ and 3. Using BSS problems as examples, we compare COPICA with FastICA. When original sources come form heavy-tail distributions, the performance of COPICA is similar as that of FastICA. However, for the cases that sources are generated from near-Gaussian random variables, COPICA outperforms than FastICA. Besides the numerical comparison, the differences between COPICA and FastICA are that there is no linear mixing model assumption in COPICA, and we do not have the restriction with at most one Gaussian source. Since we only need to pre-specify the copulae without any marginal distributed assumptions, COPICA can be treated as a semi-parametric ICA method.

In COPICA, we need to pre-specify copulae first. Three different types of copulae with one corresponding parameter are chosen for demonstration here. When p is large, more and more copula parameters need to be estimated resulting in a high computational burden. Thus how to choose the proper copulae is an important issue. One possibility is the hierarchical Archimedean copulas (Okhrin, Okhrin and Schmid, 2007), because fewer parameters are in this kind of copulae. After choosing copulae, the corresponding weights in the objective function are also need to be determined. Two possible types of weights are shown in Eqs. (8) and (9). Intuitively we would suggest to treat every divergence functions equally likely. Otherwise the weights might be proportional to the inverse of the standard deviations of the parameter estimations. Hence one of future works would be on the selection of the weights in the objective function. Simulated annealing algorithm is adopted here for minimization due to its simplicity. However, the initial status will affect the

overall performance of SA algorithm, and the convergence of SA algorithm is quit slow. Therefore, an efficient optimization procedure would be necessary to improve the efficiency of COPICA.

Acknowledgments

References

[1] Comon, P. (1994). "Independent Component Analysis - a New Concept?" *Signal Processing*, **36**(3), 287-314.

[2] Hastings, W. (1970). "Monte Carlo Sampling Methods Using Markov Chains and Their Applications," *Biometrika*, **57**, 97-109.

[3] Hyvärinnen, A. (1999). "Sparse Code Shrinkage: Denosing of Nongaussian Data by Maximum Likelihood Estimation." *Neural Computation*, **11**(7), 1739 - 1768.

[4] Hyvärinen, A., Karhunen, J. and Oja, E. (2001). *Independent Component Analysis*, Wiley and Sons.

[5] Hyvärinen, A. and Oja, E. (1997). "A Fast Fixed-point Algorithm for Independent Component Analysis." *Neural Computation*, **9**(7), 1483 - 1492.

[6] Jones, M. C. and R. Sibson (1987). "What Is Projection Pursuit?" *J. of the Royal Statistical Society, ser. A*, 150:1-36.

[7] Kirkpatrick, S., Gelatt Jr. C. D. and Vechhi, M. P. (1983). "Optimization by Simulated Annealing." *Science*, **220**, 671-680.

[8] Liu, J. S. (2001). *Monte Carlo Strategies in Scientific Computing*. Springer-Verlag. New York.

[9] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N. and Teller, A. H. (1953). "Equation of State Calculation by Fast Computing Machines." *J. Chem. Phys.* **21**, 1087-1092.

[10] Nelsen, R. B.(2006). *An Introduction to Copulas*, Springer, Berlin.

[11] Schmidt, R. (2002). "Tail Dependence." In *Applied Quantitative Finance* (Edited by W. Härdle, T. Kleinow and G. Stahl), Springer Verlag, Berlin.

[12] Sklar, A. (1959). "Fonctions de Repartition á n Dimensions et Leurs Marges." *Pub. Inst. Statist. Univ. Paris*, **8**, 229-231.

[13] Sklar, A. (1996). "Random Variables, Distribution Functions, and Copulas - a Personal Look Backward and Forward." In *Distributions with Fixed Marginals and Related Topics* (Edited by L. Ruschendorf, B. Schweizer and M. D. Taylor), Institute of Mathematical Statistics, Hayward, CA, pages 1-14.

SFB 649 Discussion Paper Series 2008

001   "Testing Monotonicity of Pricing Kernels" by Yuri Golubev, Wolfgang Härdle and Roman Timonfeev, January 2008.
002   "Adaptive pointwise estimation in time-inhomogeneous time-series models" by Pavel Cizek, Wolfgang Härdle and Vladimir Spokoiny, January 2008.
003   "The Bayesian Additive Classification Tree Applied to Credit Risk Modelling" by Junni L. Zhang and Wolfgang Härdle, January 2008.
004   "Independent Component Analysis Via Copula Techniques" by Ray-Bing Chen, Meihui Guo, Wolfgang Härdle and Shih-Feng Huang, January 2008.