Stat Comput (2015) 25:273–288
DOI 10.1007/s11222-013-9431-3

COPICA—independent component analysis via copula techniques

Ray-Bing Chen · Meihui Guo · Wolfgang K. Härdle · Shih-Feng Huang

Received: 7 January 2012 / Accepted: 18 October 2013 / Published online: 22 March 2014
© Springer Science+Business Media New York 2014

Abstract Independent component analysis (ICA) is a modern computational method developed in the last two decades. The main goal of ICA is to recover the original independent variables by linear transformations of the observations. In this study, a copula-based method, called COPICA, is proposed to solve the ICA problem. The proposed COPICA method is a semiparametric approach, the marginals are estimated by nonparametric empirical distributions and the joint distributions are modeled by parametric copula functions. The COPICA method utilizes the estimated copula parameter as a dependence measure to search the optimal rotation matrix that achieves the ICA goal. Both simulation and empirical studies are performed to compare the COPICA method with the state-of-art methods of ICA. The results indicate that the COPICA attains higher signal-to-noise ratio (SNR) than several other ICA methods in recovering signals. In particular, the COPICA usually leads to higher SNRs than FastICA for near-Gaussian-tailed sources and is competitive with a nonparametric ICA method for two dimensional sources. For higher dimensional ICA problem, the advantage of using the COPICA is its less storage and less computational effort.

Keywords Blind source separation · Canonical maximum likelihood method · Givens rotation matrix · Signal/noise ratio · Simulated annealing algorithm

R.-B. Chen
Dept. of Statistics, National Cheng Kung University, Tainan 701, Taiwan
e-mail: rbchen@stat.ncku.edu.tw

M. Guo (✉)
Dept. of Applied Math., National Sun Yat-sen University, Kaohsiung 804, Taiwan
e-mail: guomh@math.nsysu.edu.tw

W.K. Härdle
Center for Applied Statistics and Economics, Humboldt-Universität zu Berlin, Berlin, Germany
e-mail: haerdle@wiwi.hu-berlin.de

W.K. Härdle
Lee Kong Chian School of Business, Singapore Management University, Singapore, Singapore

S.-F. Huang
Dept. of Applied Math., National University of Kaohsiung, Kaohsiung 811, Taiwan
e-mail: huangsf@nuk.edu.tw

1 Introduction

Independent component analysis (ICA) is a recently developed multivariate statistical method, and can be treated as a generalization of principal component analysis (PCA). PCA is based on the eigenvalue decomposition of the covariance matrix, and projects data onto the eigenvectors of the covariance matrix. Although an eigenvalue decomposition of covariance yields only uncorrelated factors, together with Gaussian distributional assumption, the principal components are independent. However, the "independent" property will not hold if Gaussianity is violated. Non-Gaussianity of the independent components is a fundamental restriction of ICA, since one can only estimate the ICA model of Gaussian data up to an orthogonal transformation and the mixing matrix is not identifiable if there are more than two Gaussian independent components. Thus ICA targets on non-Gaussian samples. The main goal of ICA is to find linear transformations that map the observed multivariate time series into independent components (ICs). To accomplish the ICA goal, unlike the eigenvalue decomposition approach in PCA, ICs are estimated via an optimization problem, in which the statistical cross dependency among the extracted ICs is minimized. In practice, ICA has been successfully applied in blind source separation (Comon 1994), image denoising (Hyvärinen 1999b), natural image patch (Bell and Sejnowski 1995), single-trial EEG records (Tsai et al. 2006) and many other applications (see for example Lee 1998; Hyvärinen and Oja 2000; Abayomi et al. 2011).

There has been a wide development of interest in the computational technique of ICA in the past two decades. The ICA method can be formulated as optimization of an objective function which minimizes the cross-dependency among the components. The performance of the ICA method depends on the choice of objective function and the algorithm used for implementation of the optimization problem determines the speed of the ICA method. Various objective functions used in ICA include maximum likelihood, negentropy, higher order cumulants, kurtosis and mutual information. Several procedures and algorithms were proposed to search the independent components based on different objective functions and searching algorithms. The well known FastICA proposed by Hyvärinen and Oja (1997) was based on maximization of non-Gaussianity via measurements such as kurtosis and negentropy. Since the negentropy is always nonnegative and vanishes if and only if the signal is Gaussian, it can be used as a measure of distance to normality. And an approximative Newton iteration fixed-point algorithm is used to improve the computational efficiency of the FastICA which is faster than the gradient based methods. The details of FastICA can be found in Hyvärinen and Oja (2000). Bell and Sejnowski (1995) proposed a natural gradient ICA algorithm by minimizing the mutual information among the outputs, which can be considered as the Kullback-Leibler divergence (KL divergence) between the current joint density and the product of marginal densities. Their approach can also be treated as a maximum likelihood approach. Comon (1994) gave a contrast function for ICA by approximating the mutual information in terms of third-order and fourth-order cumulants. CuBICA, proposed by Blaschke and Wiskott (2004), improved Comon's algorithm by simplifying the corresponding contrast function. Bach and Jordan (2002) proposed the kernel independent component analysis which uses flexible kernels to model the dependence between the variables. Gretton et al. (2005) proposed another kernel independent criterion, the Hilbert-Schmidt Independent Criterion (HSIC), and the HSIC-based ICA contrast has a diagonal Hessian at independence. Then Shen et al. (2009) introduce an optimization method for HSIC, named FastKICA, and RADICAL (Learned-Miller and Fisher 2003) used an estimate of univariate entropies to find Jacobi rotations that make pairs of signals as independent as possible. Kirshner and Póczos (2008) used Schweizer-Wolff measure of dependence to search the independent components.

In this research, a new procedure called COPICA is proposed for ICA. In the COPICA procedure, the joint distribution of the components is modeled by copula functions. For better modeling of non-Gaussianity and other empirical facts such as heavy tail behavior of financial data, copulae have been introduced into the quantitative finance practice. The copula technique is based on the 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 also provide a flexible family for modeling dependencies and include the product copula as the family element representing independence. An important property of the copula parameters is that in some cases they are also the tail dependence parameters. Hence, the estimates of the copula parameters provide direct parametric estimates of the tail dependence. In the proposed COPICA approach, we use the deviation between the fitted copula parameters and the copula parameters at independence as a measure of dependence, then define the corresponding divergence function used as the objective function in ICA. Thus the COPICA procedure combines ICA ideas from the engineering literature with the copula based research in quantitative finance. For parameter estimation, we use the historical empirical distribution in the estimation of marginal distributions then use the canonical maximum likelihood (CML) to estimate the copula parameters. A simulated annealing algorithm is used to minimize our divergence function to find the best recovered matrix. Since the marginal distribution is estimated by a nonparametric empirical estimate and the joint distribution is modeled by a parametric copula function, the proposed COPICA method can be viewed as a semiparametric ICA approach.

We took the advantage of copula to separate the parameter space of the full likelihood function into the copula parameter space and the marginal parameter space. If the margins are well fitted, then an estimator on the joint part (i.e. the copula parameters) can recover independence. In COPICA, we estimate the marginal distribution by the nonparametric empirical distribution. An advantage of estimating marginals using empirical distributions is that this procedure is relatively free of assumptions. And the empirical distribution has nice asymptotic properties including consistency and asymptotic normality. Since marginal distributions are estimated nonparametrically, the copula parameters are the only unknown parameters in COPICA. Based on the whitening data, our goal is to find the proper rotation matrix to recover the independent sources. To accomplish this goal, the divergence function is defined via the copula parameters. Given a rotation matrix, R, the estimations of copula parameters in divergence function are obtained via CML approach based on the current empirical marginal distributions of the rotated data. Thus the copula parameter estimators and our divergence function are function of the rotation matrix, R. However, it is difficult to express the divergence function explicitly in terms of the rotation matrix (or rotation angles). Hence to solve our optimization problem w.r.t. rotation angles, the gradient based optimization approach cannot be used. Simulated annealing algorithm is a stochastic optimization method which does not need the gradient information. Of course, SA is not the only optimization approach to solve our target problem. Other possible approaches are pattern search, Gold search and other stochastic optimization approach, for example, genetic algorithm.

In addition to our COPICA method, copula based independent component analysis approach has also been proposed in Ma and Sun (2007), Abayomi et al. (2008, 2011). Abayomi et al. (2008, 2011) considered the objective function based on the mutual information via copula, which measures a norm between the estimator and the oracular value. Specifically, Abayomi et al. (2008) provided a theoretical foundation of mutual information based approach and a version of their norm was utilized in Abayomi et al. (2011). Their rotation matrix is obtained by minimizing the mutual information (distance) between parametric copula and independent marginals. In addition to the full parametric approach, they also proposed a semiparametric approach by using the empirical distributions for marginals. Two numerical approaches were introduced to obtain their rotation matrix. In their full model method, the mutual information is used as the objective function and the gradient type approach is applied to obtain the rotation matrix numerically. In their partite model approach, they use Singular Value Decomposition of the bivariate mutual information matrix, which is constructed via pairwise copula, to find the orthogonal transformation matrix.

Although, the COPICA method and Abayomi et al.'s approach both use copula to model the joint distributions of the components, the objective function and optimization algorithm are different, which are the two major components determining the performance and speed of the ICA method. In Abayomi et al. (2008, 2011), mutual information was used as the dependent measurement. For the ICA problem when independent signals are obtained, the joint density function is equal to the product of the marginal densities and the mutual information is zero. And the copula parameter (no matter which copula is fitted) equals to its independent parameter, consequently our COPICA objective function equals to zero which is the same as the mutual information. Hence although our norm is not generated directly from the mutual information, yet it achieves the same optimal point when independence are obtained.

In the next section, the detail procedure of COPICA is introduced. In Sect. 3, blind source separation examples are demonstrated to illustrate the performance of our method. In Sect. 4, we compare the performance of COPICA with FastICA in terms of their signal to noise ratios (SNR) on the recovered signals for blind source separation problems. In Sect. 5, we compare COPICA method with nonparametric rank-based approaches. Both simulation and empirical studies will be performed to compare the COPICA method with the state-of-art methods of ICA. Our numerical results and empirical study also support the applicability of the proposed COPICA method. In summary, the comparison results show that:

(1) The computational burden in determining the ICA transformation are the same for the COPICA and the FastICA.
(2) The COPICA method attains higher SNR than the FastICA for near-Gaussian-tail sources on the recovered signals for the blind source separation problems. We also noted that the FastICA method sometimes fails to converge for near-Gaussian-tail sources.
(3) The COPICA method is competitive with the ICA method via a nonparametric measure, Schweizer-Wolff σ_{SW} for bivariate sources. For higher dimensional case, the COPICA method attains higher SNR than the ICA method via Schweizer-Wolff σ_{SW} on the average and reduces significantly the storage space.

Finally conclusion is given in Sect. 6.

2 COPICA procedure

Assume we observe the n linear mixtures

$$X = (x_1, x_2, \ldots, x_n)^\top$$

of the n independent components $S = (s_1, s_2, \ldots, s_n)^\top$, that is $X = AS$, where $A = (a_{ij})$ is the $n \times n$ mixing matrix. Here we assume that A is full rank. The independent components s_j's are latent random variables with zero mean which cannot be observed directly and the mixing matrix A is unknown. The goal of ICA is to find linear combination of the observed data X, $Y = BX$ such that the components of Y, y_i's, are as independent as possible. Here unlike PCA to obtain uncorrelated linear combination of x_i, to achieve the independence among y_i's, the possible measurements are related to nonlinear transformations of y_i, for example, nonlinear correlation, $E(f(y_i)g(y_j))$, where f and g are two function and at least one is nonlinear (Hyvärinen and Oja 2000). Thus ICA can be treated as to remove the nonlinear dependence by using the linear transformation of data.

In addition to centralize the observed data, most of the ICA procedures, such as FastICA, whiten the observations first by the matrix $W = \Sigma^{-1/2}$, where Σ is the covariance matrix of X. That is, the components of $Z = WX$ are uncorrelated with unit norm, i.e. $\mathrm{Cov}(Z) = I_n$. The independent components are obtained by multiplying the pre-whitened observations with an orthogonal matrix R such that the outputs $Y = RZ$ are nearly statistically independent.

In this section, we first introduce the copula modeling of the joint dependence structure of the transformed components, then define the copula parameters as a measure of dependence. A rotation matrix representation of the orthogonal matrix R is also given. Finally, the COPICA procedure is introduced.

2.1 Copula model

According to Nelsen (2006), an n-dimensional copula is defined as follows.

Definition 1 An n-dimensional copula $C(\mathbf{u})$, where $\mathbf{u} = (u_1, \ldots, u_n)$, is a function from $[0, 1]^n \to [0, 1]$ with the following properties:

1. $C(\mathbf{u})$ is grounded, that is,

   $$C(u_1, \ldots, u_{i-1}, 0, u_{i+1}, \ldots, u_n) = 0,$$

   which means that the copula is zero if one of the arguments is zero, and $C(1, \ldots, 1, u, 1, \ldots, 1) = u$, which means that the copula is equal to u if one argument is u and all others are 1.
2. $C(\mathbf{u})$ is n-increasing, that is, for each hyperrectangle $B = \prod_{i=1}^{n}[x_i, y_i] \subseteq [0, 1]^n$,

   $$\int_B dC(\mathbf{u}) = \sum_{\mathbf{z} \in \times_{i=1}^{n}\{x_i, y_i\}} (-1)^{N(\mathbf{z})} C(\mathbf{z}) \geq 0,$$

   where $\mathbf{z} = (z_1, \ldots, z_n)$, $\times_{i=1}^{n}\{x_i, y_i\}$ denotes the set of the vertices of B, and $N(\mathbf{z})$ is the number of $\{k : z_k = x_k\}$.

Copula has recently become the most significant new tool to handle co-movement between markets in the field of finance and the analysis of current status data in biostatistics, because it provides a flexible way to connect the marginal distributions of individual component to their multivariate joint distribution. Sklar's theorem provides the theoretical foundation for the application of copulae. Let $F_{X_j}(x_j)$ denote the marginal distribution of X_j, $j = 1, \ldots, n$. Based on the work of Sklar (1959), there exists a copula function C such that

$$F_X(x_1, \ldots, x_n) = C\big\{F_{X_1}(x_1), \ldots, F_{X_n}(x_n); \theta\big\}, \tag{1}$$

where $F_X(x_1, \ldots, x_n)$ is the joint distribution of $X = (X_1, \ldots, X_n)$ and $\theta = (\theta_1, \ldots, \theta_d)$ denotes the copula parameters. In the case of independence, the joint distribution is the product of the marginal distributions, that is $F_X(x_1, \ldots, x_n) = F_{X_1}(x_1) \cdots F_{X_n}(x_n)$. This corresponds to the product (independence) copula $C(\mathbf{u}) = u_1 \cdots u_n$. In the following, we introduce some well-known copula families which will be considered in this work as an illustration.

(1) Gumbel copula:

$$C(u_1, \ldots, u_n; \theta) = \exp\left[-\left\{\sum_{j=1}^{n}(-\log u_j)^{\theta}\right\}^{\frac{1}{\theta}}\right],$$

$\theta \geq 1$.

When the Gumbel parameter $\theta = 1$, it is the independence copula. The Gumbel copula is motivated by limit theorems for joint extremes (Kotz and Nadarajah 2000) and has for long played an important role in modeling distributions of extremes. The Gumbel copula can model upper tail dependence. For instance the bivariate Gumbel copula, the upper tail dependence of two random variables X_1 and X_2 is defined as

$$\begin{aligned}\lambda_U &= \lim_{v \to 1^-} P\big(F_{X_2}(X_2) > v \mid F_{X_1}(X_1) > v\big)\\ &= \lim_{v \to 1^-} \big(1 - 2v + C(v, v)\big)/(1 - v) = 2 - 2^{1/\theta},\end{aligned} \tag{2}$$

which is always positive for $\theta > 1$. In addition, the Gumbel copula can be rotated to change the direction of the tail dependence. For example, in the 2-dimensional case, the survival Gumbel copula, denoted by $\hat{C}(u_1, u_2; \theta)$, can be obtained by rotating a Gumbel copula by 180 degrees, that is,

$$\hat{C}(u_1, u_2; \theta) = u_1 + u_2 - 1 + C(1 - u_1, 1 - u_2; \theta),$$

where $C(u_1, u_2; \theta)$ is the 2-dimensional Gumbel copula. Thus, the survival Gumbel copula can be used to model lower tail dependence.

(2) Clayton copula:

$$C(u_1, \ldots, u_n; \theta) = \left(\sum_{j=1}^{n} u_j^{-\theta} - n + 1\right)^{-1/\theta}, \quad \theta > 0.$$

As the copula parameter $\theta \to 0$, the Clayton copula approaches to the independence copula. The Clayton copula can model multivariate lower tail dependence. For instance the bivariate Clayton copula, the lower tail dependence of two random variables X_1 and X_2 is defined as

$$\begin{aligned}\lambda_L &= \lim_{v \to 0^+} P\big(F_{X_2}(X_2) \leq v \mid F_{X_1}(X_1) \leq v\big)\\ &= \lim_{v \to 0^+} C(v, v)/v = 2^{-1/\theta},\end{aligned} \tag{3}$$

which is positive for all $\theta > 0$. Similar to the Gumbel copula, the Clayton copula can also be used to depict the upper tail dependence by rotation.

(3) Gaussian copula: for a given correlation matrix $\Sigma \in R^{n \times n}$, the Gaussian copula with parameter matrix Σ can be written as

$$C(u_1, \ldots, u_n; \Sigma) = \boldsymbol{\Phi}_{\Sigma}\big\{\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_n)\big\},$$

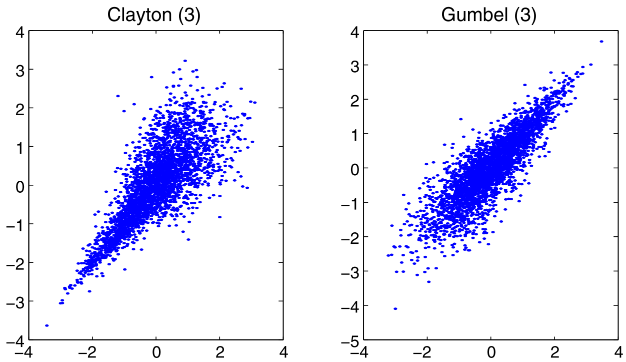

Fig. 1 Bivariate plots of Clayton and Gumbel copulae with $\theta = 3$ and $N(0, 1)$ marginals (Color figure online)

where $\boldsymbol{\Phi}_\Sigma$ is the joint cumulative distribution function of a multivariate normal distribution with mean vector zero and covariance matrix equal to the correlation matrix Σ and Φ^{-1} is the inverse cumulative distribution function of $N(0, 1)$. In particular, if the correlation matrix is the identity matrix, then Gaussian copula is the independence copula. Furthermore, if the X_j's are normally distributed, then the Gaussian copula is correspond to the multivariate normal distribution. Gaussian copula is a popular and convenient type of copula, especially when the dimension is large. Since Gaussian copula depends only on the pairwise rank correlations between the marginals when the marginal are continuous (Mardia 1970), it continues to capture the dependence structure of the Normal-To-Anything (NORTA) distribution with arbitrary continuous marginal distributions (Ghosh and Henderson 2003). The bivariate Gaussian copula can model neither upper nor lower tail dependence, unless the correlation coefficient $\rho = 1$, since $\lambda_U = \lambda_L = 0$ for $\rho < 1$ and $\lambda_U = \lambda_L = 1$ for $\rho = 1$.

In Fig. 1, we give the bivariate plots of random samples generated from Clayton and Gumbel copulae with parameter $\theta = 3$ and $N(0, 1)$ marginal distributions, respectively. Although the marginal distributions of the two cases are the same, different tail dependencies are displayed. Yet there are modeling limitations for the Gumbel and Clayton copulae (in general the family of Archimedean copulae) in higher-dimensions, as they imply exchangeability and hence equicorrelated ranks, which is obviously untenable in real application. For general reference of copulae, please refer to Nelsen (2006). For generalization of Archimedean copula models, see for example, McNeil and Nešlehová (2010) and Genest et al. (2011). For applications of copula in data mining, see Yu et al. (2011). For more discussion of the tail dependence parameters and the Gaussian copula, we refer to Schweizer and Wolff (1981) and Genest et al. (2011).

Let $\{\phi(\cdot|\theta)\}_{\theta\in\Theta}$ be a family of copula densities, where $\Theta \subset \Re^q$ is the parameter space. In this work, we use the canonical maximum likelihood (CML) estimator to estimate the copula parameter θ defined as below. Let $\{X_t = (x_{1t}, x_{2t}, \ldots, x_{nt})^\top\}_{t=1}^T$ is a realization of length T of the linear mixture X.

Step 1: Obtain $\hat{F}_{X_i}(\cdot)$, $i = 1, \ldots, n$ are the empirical marginal distributions, and then

Step 2:

$$\hat{\theta} = \arg\max_{\theta\in\Theta} \sum_{t=1}^{T} \ln\big(\phi\big(\big(\hat{F}_{X_i}(x_{it}), i = 1, 2, \ldots, n\big)\big|\theta\big)\big), \quad (4)$$

Therefore, the fitted copula $\phi(.|\hat{\theta})$ with the CML estimator can be treated as the best approximation for the true copula ψ in the copula family $\{\phi(\cdot|\theta)\}_{\theta\in\Theta}$ based on $\{X_t = (x_{1t}, x_{2t}, \ldots, x_{nt})^\top\}$, $t = 1, \ldots, T$. In the proposed COP-ICA method, we consider the best approximations of the transformed data in the three copula families, Gumbel, Clayton and Gaussian copulae to capture its different dependence structure.

2.2 Representation of orthogonal matrices

In an ICA model, the following two ambiguities are well known to hold. Firstly because we can freely change the order of the components s_i's, 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 though. Secondly we cannot determine the variances of the independent components. Thus, without loss of generality, we assume that each component of $Y = RZ$ has unit variance. Then by independence assumption of Y, we have $\mathrm{Cov}(Y) = I_n$. Therefore the transformation matrix R satisfies

$$RR^\top = R\,\mathrm{Cov}(Z)R^\top = \mathrm{Cov}(RZ) = \mathrm{Cov}(Y) = I_n.$$

That is, the transformation matrix R is an orthogonal matrix which can be represented as the following product of the Givens rotation matrices,

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

The matrix $G_{ij}(\beta_{ij})$ is an n-dimensional Givens rotation matrix which represents a rotation in the plane 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 $\beta_{ij} \in [0, 2\pi)$. This Givens matrix representation of R has been used in ICA algorithms, such as Comon (1994), Blaschke and Wiskott (2004), Kirshner and Póczos (2008) and so on. The product of the orthogonal matrix R and the whitening matrix W, $B = RW$, is our objective transformation matrix of the observed data X to achieve independence.

The major task is to search the rotation angles, β_{ij}, to make the components of

$$Y = RZ = (RW)X = (RW)(AS)$$

nearly independent. In the bivariate case, $n = 2$, the Givens matrix is derived in the following proposition.

Theorem 1 *Assume $S = (s_1, s_2)^\top$ is a random vector of two independent random variables with unit variance. Let $X = AS$ where*

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

is a non-degenerated mixing matrix. Let $Z = WX$ and $W = (AA^\top)^{-1/2}$ is the whitening matrix of X. Then the following Givens matrix of order 2,

$$G(\beta) = \begin{pmatrix} \cos\beta_{12} & \sin\beta_{12} \\ -\sin\beta_{12} & \cos\beta_{12} \end{pmatrix} \tag{5}$$

is the objective rotation matrix, that is

$$G(\beta)Z = \begin{cases} (s_1, s_2)^\top, & \text{if } ad - bc > 0, \\ (s_2, s_1)^\top, & \text{if } ad - bc < 0, \end{cases}$$

where

$$\begin{cases} \cos\beta_{12} = \frac{(a+d)\,\mathrm{sign}(ac+bd)}{\sqrt{(a+d)^2+(b-c)^2}}, \\ \sin\beta_{12} = \frac{(-b+c)\,\mathrm{sign}(ac+bd)}{\sqrt{(a+d)^2+(b-c)^2}}, \end{cases} \quad \text{if } ad - bc > 0,$$

or

$$\begin{cases} \cos\beta_{12} = \frac{(b+c)\,\mathrm{sign}(ac+bd)}{\sqrt{(a-d)^2+(b+c)^2}}, \\ \sin\beta_{12} = \frac{(-a+d)\,\mathrm{sign}(ac+bd)}{\sqrt{(a-d)^2+(b+c)^2}}, \end{cases} \quad \text{if } ad - bc < 0.$$

Proof First, consider the case $ad - bc > 0$. Since

$$G(\beta)Z = G(\beta)WX = G(\beta)\big(AA^\top\big)^{-1/2}AS = S,$$

it implies $G(\beta) = A^{-1}(AA^\top)^{1/2}$. Let $U = \begin{pmatrix} u_{11} & u_{12} \\ u_{12} & u_{22} \end{pmatrix}$ be the positive definite matrix satisfying $U^2 = AA^\top$. We have

$$\begin{cases} u_{11}^2 + u_{12}^2 = a^2 + b^2, \\ u_{11}u_{12} + u_{12}u_{22} = ac + bd, \\ u_{12}^2 + u_{22}^2 = c^2 + d^2. \end{cases} \tag{6}$$

Combining with the constrains of $ad - bc > 0$ and U be the positive definite matrix, the solutions of (6) are

$$u_{11} = \frac{(a^2 + b^2 + ad - bc)\,\mathrm{sign}(ac + bd)}{\sqrt{(a + d)^2 + (b - c)^2}},$$

$$u_{12} = \frac{|ac + bd|}{\sqrt{(a + d)^2 + (b - c)^2}},$$

$$u_{22} = \frac{(c^2 + d^2 + ad - bc)\,\mathrm{sign}(ac + bd)}{\sqrt{(a + d)^2 + (b - c)^2}}.$$

Thus,

$$G(\beta) = A^{-1}\big(AA^\top\big)^{1/2} = A^{-1}U = \begin{pmatrix} \frac{(a+d)\,\mathrm{sign}(ac+bd)}{\sqrt{(a+d)^2+(b-c)^2}} & \frac{(-b+c)\,\mathrm{sign}(ac+bd)}{\sqrt{(a+d)^2+(b-c)^2}} \\ \frac{(b-c)\,\mathrm{sign}(ac+bd)}{\sqrt{(a+d)^2+(b-c)^2}} & \frac{(a+d)\,\mathrm{sign}(ac+bd)}{\sqrt{(a+d)^2+(b-c)^2}} \end{pmatrix}.$$

Similarly, if $ad - bc < 0$, then $G(\beta)Z = JS$, where $J = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. The solutions of (6) are

$$u_{11} = \frac{(a^2 + b^2 + bc - ad)\,\mathrm{sign}(ac + bd)}{\sqrt{(a - d)^2 + (b + c)^2}},$$

$$u_{12} = \frac{|ac + bd|}{\sqrt{(a - d)^2 + (b + c)^2}},$$

$$u_{22} = \frac{(c^2 + d^2 + bc - ad)\,\mathrm{sign}(ac + bd)}{\sqrt{(a - d)^2 + (b + c)^2}}.$$

Thus,

$$G(\beta) = JA^{-1}\big(AA^\top\big)^{1/2} = JA^{-1}U = \begin{pmatrix} \frac{(b+c)\,\mathrm{sign}(ac+bd)}{\sqrt{(a-d)^2+(b+c)^2}} & \frac{(-a+d)\,\mathrm{sign}(ac+bd)}{\sqrt{(a-d)^2+(b+c)^2}} \\ \frac{(a-d)\,\mathrm{sign}(ac+bd)}{\sqrt{(a-d)^2+(b+c)^2}} & \frac{(b+c)\,\mathrm{sign}(ac+bd)}{\sqrt{(a-d)^2+(b+c)^2}} \end{pmatrix}.$$

This completes the proof. □

Geometrically speaking, the rotation angle β_{12} represents the angle between one of the column vectors in the matrix $(AA^\top)^{-1/2}A$ and the x_1-axis. In general for higher dimensional case, we have $G(\beta) = A^\top(AA^\top)^{-1/2}$, where β is the vector of the Givens rotation angles, β_{ij}. However the formula is not practically applicable, due to the fact that the matrix A is unknown in real applications. In order to determine the rotation angles of the Givens matrix, we will adopt a criterion based on copula parameter.

2.3 Divergence function based on copula parameter

Suppose $X = (X_1, \ldots, X_n)$ comes from the joint distribution, F_X. Then according to Eq. (1), we have that $dC(x) = \frac{dF_X(x)}{\prod_i \{dF_{X_i}(x_i)\}}$, where F_{X_i} is the marginal distribution of X_i. That is that the derivative of the copula is the ration of the joint density function and the product of the marginal density functions. Therefore, the copula parameters contain the information of the dependence among X. Furthermore, the mutual information for X can be re-presented via given copula C and its copula density ϕ by

$$\begin{aligned} MI(X) &= \int \log \frac{dF_X(x)}{\prod_i \{dF_{X_i}(x_i)\}} dF_X(x) \\ &= \int_{I^n} \log\big(dC(u)\big) dC(u) \\ &= \int_{I^n} \phi(u|\theta) \log\big(\phi(u|\theta)\big) du, \end{aligned}$$

where $I^n = [0, 1]^n$. Once the independent copula parameters are obtained, the value of the mutual information is zero. Thus the copula parameter θ could be used as the measurement of the dependency. The similar idea was also mentioned in Abayomi et al. (2008). Another point comes from the relation of the tail dependence and copula parameters. As shown by (2) and (3), both the upper tail dependence λ_U of the bivariate Gumbel copula and the lower tail dependence λ_L of the bivariate Clayton copula are monotonic function of their copula parameters θ. Therefore the copula parameters of the Gumbel copula and of the Clayton copula are also their tail dependence parameters. This is also supported us to use the copula parameters as the measure of dependence.

In ICA approach, we need to define an objective function for the source separation such that if the minimal value of this objective function is attended, then the recovered sources are independent. Since the copula parameters are used as the dependence measurement, we illustrate by the bivariate case in the following about how to choose the corresponding objective function for ICA problem. For a given (demean) realization $\{X_t = (x_{1t}, x_{2t})^\top\}_{t=1}^T$, select a copula function and a rotation angle β_{12}. Transform the whitening data $Z = WX$ by the Givens rotation matrix $R = G$ of the form given in (5), then compute the CML estimator $\hat{\theta}$ based on transformed data. Let θ_0 denote the copula parameter at independence of the selected copula, for example $\theta_0 = 1$ for the Gumbel copula. The magnitude

$$o(\hat{\theta}|\beta) = \|\hat{\theta} - \theta_0\|$$

is used as a measure of deviation from independence between x_{1t} and x_{2t} for this rotation angle β_{12}. Then search the angle β_{12} to minimize $o(\hat{\theta}|\beta)$ which is regarded as the optimal solution of the Givens rotation matrix to make $RZ = R(WX)$ nearly independent. In brief, we first find the best approximation of the true copula of the transformed data in a copula family, then measure the deviation from independence by the fitted copula (dependence) parameter. The objective rotation angle is obtained by minimizing the deviation from independence defined via copula parameters. In this study, we consider the best approximations of the true copula in the three copula families: Gumbel, Clayton and Gaussian. The three families are used to model upper tail, lower tail dependence structure and pairwise rank correlation between the marginals of the transformed components, respectively. Accordingly, the divergence function based on the three copula families is defined by the following weighted sum,

$$O(\hat{\theta}_1, \hat{\theta}_2, \hat{\theta}_3|\beta) = \sum_{i=1}^{3} w_i o_i(\hat{\theta}_i|\beta), \tag{7}$$

where $o_i(\hat{\theta}_i) = \|\hat{\theta} - \theta_{i0}\|$, $\hat{\theta}_i$ and θ_{i0} are respectively the fitted copula parameter and the independent parameter value of the i-th copula model, and w_i's are the positive weights. In our simulation and empirical studies, we set the weights w_i's to be inverse proportional to the standard deviations of the CML estimators. In the implementation we will rotate the transformed components by the angles $i\pi/2, i = 1, 2, 3$ to identify possible dependent structure, and include the corresponding measure of deviation from independence in the divergence function O.

The idea can be extended to higher dimensional case. For n-dimensional random variables $Y_1, \ldots, Y_n$ ($n \geq 2$), mutual independence implies that any subset random variables of $Y_1, \ldots, Y_n$ are also mutually independent. Therefore, the divergence function measuring multivariate dependence of a selected copula function can be defined as

$$O(\hat{\boldsymbol{\theta}}|\beta) = \sum_{\{i,j\}\subset\mathcal{N}} w_{ij} o(\hat{\theta}_{ij}|\beta) + \sum_{\{i,j,k\}\subset\mathcal{N}} w_{ijk} o(\hat{\theta}_{ijk}|\beta) + \cdots + w_{\mathcal{N}} o(\hat{\theta}_{\mathcal{N}}|\beta), \tag{8}$$

where the vector parameter

$$\hat{\boldsymbol{\theta}} = (\hat{\theta}_{12}, \ldots \hat{\theta}_{(n-1)n}, \hat{\theta}_{123}, \ldots, \hat{\theta}_{\mathcal{N}}),$$

denote the copula parameter estimations of the transformed (pre-whitening and rotated via the rotation angle vector β) data of

$$(Y_1, Y_2), \ldots, (Y_{n-1}, Y_n), (Y_1, Y_2, Y_3), \ldots, (Y_1, \ldots, Y_n),$$

respectively. The divergence function $O(\hat{\theta}|\beta)$ measure all multivariate dependence of dimensionality greater then or equal to 2. The components of the data are deemed nearly independent when $O(\hat{\theta}|\beta)$ is close to zero. Similarly, multiple copula families can be included in the divergence function (8) as in (7).

Based on the chosen copulae and the pre-whitening data Z, the magnitude of the divergence function $O(\hat{\theta}|\beta)$ given the rotation angle vector β, is computed in the following steps:

(1) Rotate the data according to the rotation angle vector β;
(2) Find the currently empirical marginal distributions, $\hat{F}_{Y_i}$;
(3) Obtain CML estimator $\hat{\theta}$ by minimizing Eq. (4) based on $\hat{F}_{Y_i}$;
(4) Compute $O(\hat{\theta}|\beta)$.

Thus the CML estimator $\hat{\theta}$ is the function of the rotation angles β_{ij}, $1 \leq i < j \leq n$. And the independent components are identified once $O(\hat{\theta}|\beta)$ attains its minimum value in the rotation angle vector β. For brevity, we use $O(\beta_{12}, \ldots, \beta_{(n-1)n})$ to denote the objective function $O(\hat{\theta}|\beta)$, and then our ICA problem is equivalent to the minimization problem

$$\min_{\beta_{ij}, 1\leq i<j\leq n} O(\beta_{12}, \ldots, \beta_{(n-1)n}), \tag{9}$$

which means that we find the rotation angles $\beta'_{ij}s$ to minimize the divergence function $O(\hat{\theta}|\beta)$ at the CML estimator $\hat{\theta}$ with respect to β.

Algorithm 1 COPICA by simulated annealing algorithm

(I) **[Initialization]**

(1) Center the data X to make its mean zero, and obtain its sample covariance matrix $\hat{\Sigma}$.

(2) Whiten the data by setting $Z = WX$ where $W = \hat{\Sigma}^{-1/2}$.

(3) Choose the copula families and define the objective function $O(\boldsymbol{\beta})$.

(II) **[Optimization by simulated annealing algorithm]**

(1) Select initial angles, $\beta_i^{(0)}$; Choose the decreasing function $T(t)$ and set $t = 1$.

(2) Repeat until t is large enough

(2.1) Run N_t iterations of the Gibbs sampler with $\pi_{T(t)}(\boldsymbol{\beta})$ as its target distribution. Pass the final sample as $\boldsymbol{\beta}^{(t)}$.

(2.2) $t = t + 1$.

(3) Identify the optimal angle vector β^* for $O(\beta)$.

(III) **[Transformation matrix]**

(1) Compute $R = \prod_{1 \le i < j \le n} G_{ij}(\beta_{ij}^*)$

(2) The optimal transformation matrix $B = RW$.

2.4 The details of the COPICA procedure

The COPICA procedure used to find the independent components of a given data set X is given in Algorithm 1, in which the optimization step is by the simulated annealing method.

There are two crucial steps in Algorithm 1: selection of the copulae models, and estimation of the copula parameter and search the best rotation angles. Discussion of the two steps are given below.

Copula selection It is well known that copulae are invariant with respect to strictly increasing transformations but not necessary to general linear transformation. Hence even if we can find the “true” copula family of the whiten random vector Z, the “true” copula after rotation might still fall in another family. Therefore, the key is not to find the “true” copula but to choose the proper copulae whose best approximations are useful in providing dependence measure under mis-specification situation. In general, prior information of the data or selection criteria are helpful to choose the copulae, for example, the Copula Information Criterion (CIC in short) proposed by Grønneberg and Hjort (2008). Due to the characteristic of the signals/sources in engineering or finance applications, heavy-tailed source is a widely used assumption in blind source separation (BSS) problems, for example, Kidmose (2001) and Chen and Wu (2007). In addition to BSS, the heavy-tailed assumption is also popular in the analysis of EEG signal (Tsai et al. 2006), natural image representation (Olshausen and Field 1996) and so on. The existence of tail dependence is a special feature of heavy-tailed signals/sources as well as an indication of non-independence. In this study, we utilize the tail dependence feature of the Gumbel and Clayton copulae to estimate the dependency of the transformed data. The Gaussian copula is also considered to capture pairwise rank correlations between the marginals. In an extensive simulation study, we compare the copula based dependence measure with the nonparametric Kendall’s τ in various settings of misspecified models. For the reason of concision, we only report the summary results here without detailed description. The results show that the dependence measured by the copula parameters is in good accordance with the Kendall’s τ. And under misspecified models, the copula-based measure still provide valuable dependence information. The advantages of the copula based criterion over the Kendall’s τ is its faster convergence rate and higher SNR values in ICA application. In addition, we will demonstrate the effectiveness of the three copula based dependency measure by blind source separation (BSS) examples in next section.

Optimization procedure for identifying best rotation angles Recall from Eq. (4), we estimate the copula parameters by the CML estimator which gives the best approximation to the “true” copula in its family based on the transformed data. Due to the constraints of copula parameter estimation method, it is in general difficult to have closed form solution of the CML estimators. As a result there is no explicit form of the objective function $O(\beta_{12}, \ldots, \beta_{(n-1)n})$ 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 the minimization problem defined in Eq. (9). Herein, the simulated annealing (SA) algorithm, proposed by Metropolis et al. (1953) and introduced as an optimization technique by Kirkpatrick et al. (1983), is used as an illustration to search these optimal angles. For simplicity of notation, we denote the rotation angels $\beta_{12}, \ldots, \beta_{(n-1)n}$ by $\beta_1, \ldots, \beta_q$, where $q = n(n-1)/2$. First define a density

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

where $O(\boldsymbol{\beta})$ is the objective function, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_q)^\top$ and $T(t)$ is the “temperature” at time t which is a decreasing function from initial temperature, $T(0) > 0$, to 0^+. The key step of the SA algorithm is that for t, we run N_t iterations of the Gibbs sampler with $\pi_{T(t)}(\boldsymbol{\beta})$ as its target distribution, and then choose the final sample as $\boldsymbol{\beta}^{(t)} = (\beta_1^{(t)}, \ldots, \beta_q^{(t)})^\top$ that denotes $\boldsymbol{\beta}$ at time t. In order to speed up our optimization process, we use $\exp(O(\boldsymbol{\beta}))$ in the SA algorithm instead of $O(\boldsymbol{\beta})$ directly. For more details about the SA algorithm, please refer to Liu (2001).

In implementing the COPICA with SA algorithm, the data is rotated by each sampled angles β_{ij}, and the copula parameter vector θ are re-estimated based on the rotated data to compute the magnitude of the divergency function. And in the Gibbs sampler, the simple inversion method is employed by using discretization of the continuous cumulative distribution function. Note here for this discretization method, suppose that we approximate the cumulative distribution function of $\pi_{T(t)}(\beta_i)$ by K points, β_i^j, $j = 1, \ldots, K$. Then for each point, β_i^j, CML approach is used to obtain the copula parameter estimator based on the rotated data with respect to β_i^j, and evaluate the corresponding objective function values. Finally we can have the approximated cumulative distribution function for β_i.

When we implement SA algorithm, we need to set the initial values of rotation angles, β, and the temperature, $T(t)$. For the initial β, we can simply set the $\beta = \mathbf{0}$ for the initial angles or we can set β based on our prior information, for example, the angles obtained by the FastICA. Consider the temperature $T(t)$. In order to get the global optimal point, the temperature $T(t)$ of the SA should decrease slowly such as $O(\log(t)^{-1})$, for details we refer to Liu (2001). However in practice, it is too slow to get the global optimal point and instead the linear or exponential temperature decreasing is used. In our COPICA, the temperature is chosen as $O(t^{-1/4})$ which would lead to a reasonable convergent area quickly. From our simulations and real example results, it seems that this $T(t)$ works well in our approach.

The complexity of Algorithm 1 can be analyzed as follows. First we consider the 2-dimensional situation and there is only one angle β_1 needed to identify via SA algorithm. Then in each iteration of Gibbs sampler in SA algorithm, the complexity of the inversion method is $O(KSCT\log(T))$, where K is the number of points to obtain the approximation cumulative distribution function, $T\log(T)$ is the complexity to sort each marginal source, C is the cost for maximization in CML method, and S is the number of the copula parameters used in our divergence function. Finally for general n-dimensional problem, the complexity of sweeping $q = n(n-1)/2$ angles in each iteration of the Gibbs sample is $O(n^2KSCT\log(T))$.

3 COPICA for blind source separation

We illustrate the performance of the proposed COPICA method by solving blind source separation (BSS) problems. Recently, blind source separation by ICA has received lots of attention because of its potential applications in signal processing such as in speech recognition systems, telecommunications and medical signal processing. In BSS problems, the observations $\mathbf{x}_t = (x_{1t}, \ldots, x_{nt})^\top$ are assumed to be mixtures of n mutually independent sources, $\mathbf{s}_t = (s_{1t}, \ldots, s_{nt})^\top$ at time t, that is

$$\mathbf{x}_t = A\mathbf{s}_t, \quad t = 1, \ldots, T, \tag{10}$$

where A is an $n \times n$ invertible mixing matrix. The goal of the BSS problem is to estimate the mixing matrix A and recover the original sources $\mathbf{s}_t$, for given mixtures, $\mathbf{x}_t$, $t = 1, \ldots, T$ simultaneously. If the matrix A is invertible and known, then the independent sources can be recovered by $A^{-1}\mathbf{x}_t$, $t = 1, \ldots, T$. While applying ICA methods to solve the BSS problems, the optimal transformation matrix will be the inverse matrix, A^{-1}, multiplying by a permutation matrix or a scaler.

In the following examples, we generate the sources $\{\mathbf{s}_t\}_{t=1}^T$ independently from a mixture normal distribution or from natural sound signals. The observation vectors are then generated by Eq. (10) for a given mixing matrix, A. In order to measure the performance of the COPICA method, we consider the following signal/noise ratio (SNR) value

$$\mathrm{SNR}_{s_i}(\hat{s}_i)[B] = 10\log_{10}\left(\|s_i\|^2/\|s_i - \hat{s}_i\|^2\right) \tag{11}$$

where $s_i = (s_{it}, t = 1, \ldots, T)$, $i = 1, \ldots, n$, are the original signals from the sources, $\hat{s}_i = (\hat{s}_{it}, t = 1, \ldots, T)$, $i = 1, \ldots, n$, are the recovered signals transformed by the matrix B found by the COPICA method, and $\|\cdot\|$ denote the L^2-norm. Note that the columns of the inverse of the transformation matrix, B, will be proportional to the true mixing matrix A, and the signals are normalized for SNR computation. By the definition of SNR in Eq. (11), larger value of SNR indicates better performance. We consider $\mathrm{SNR} \geq 10$ as a threshold of high SNR value, see also Sodoyer et al. (2003). By its definition, $\mathrm{SNR} \geq 10$ is equivalent to $\|s_i - \hat{s}_i\|^2/\|s_i\|^2 \leq 10\ \%$, which implies approximately at least 90 % of the signals are recovered by $\hat{s}_i$. Statistical reasoning of using 10 as high SNR value is also given below. If under independent and normal assumptions, we have roughly $\|s_i\|^2 \sim \chi^2_T$ and $\|s_i - \hat{s}_i\|^2 \sim \chi^2_{T-m}$, where $T \gg m$, hence $\frac{\|s_i\|^2/T}{\|s_i - \hat{s}_i\|^2/(T-m)} \sim F_{T,T-m}$. Since the probability of the event $\{\|s_i - \hat{s}_i\|^2/\|s_i\|^2 \leq 10\ \%\}(\equiv \mathrm{SNR} \geq 10)$ is very small for large T, it is reasonable to consider 10 as a large SNR value.

Example 1 Three sources are generated independently from the following mixture normal density,

$$f(s_i) = 0.7 f_{N(0,1)}(s_i) + 0.3 f_{N(0,3^2)}(s_i),$$

where $f_{N(\mu,\sigma^2)}$ is the density function of the normal distribution with mean μ and variance σ^2. That is each sample is generated from $N(0, 1)$ with probability 0.7 and from $N(0, 3^2)$ with probability 0.3. The mixing matrix A is set to be

$$\begin{pmatrix} 1.0000 & -2.0000 & 1.0000 \\ -1.0000 & 1.0000 & 2.0000 \\ -1.0000 & 1.0000 & 1.0000 \end{pmatrix}. \tag{12}$$

Two copulae, Gumbel and Clayton are used to measure the tail dependence. The objective function is set to be

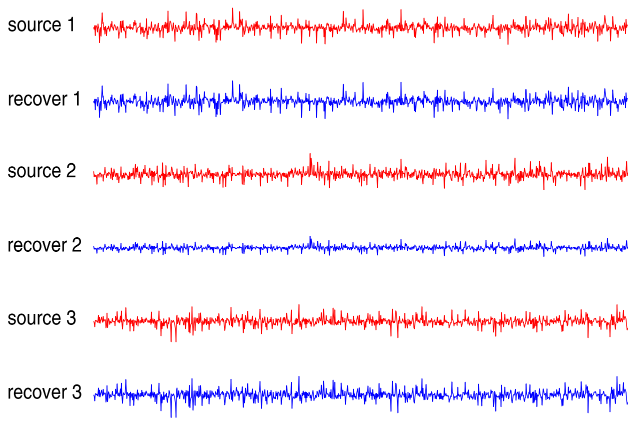

Fig. 2 The simulation results for a three dimensional blind source separation problem with mixture normal sources. The *red lines* are the original sources, and the *blue lines* are the recovered signals (Color figure online)

$$
\begin{aligned}
O(\hat{\theta}) = \omega_1 * |\hat{\theta}_{123,Gumbel} - 1| \\
+ \omega_2 * \sum_{i<j} |\hat{\theta}_{ij,Gumbel} - 1| \\
+ \omega_3 * \sum_{i<j} |\hat{\theta}_{ij,Clayton}| \\
+ \omega_4 * \sum_{i<j} |\hat{\theta}_{ij,Gaussian}|, \qquad (13)
\end{aligned}
$$

where the weights $(\omega_1, \omega_2, \omega_3, \omega_4) = (200, 300, 200, 500)$ are chosen to be inverse proportional to the standard deviations of the CML estimators of the copula parameters. After 100 iterations of the COPICA algorithm, the inverse transformation matrix found by the COPICA procedure is

$$
B^{-1} = \begin{pmatrix} 1.9974 & -3.5685 & 1.7266 \\ -1.9178 & 1.8719 & 3.6426 \\ -1.9304 & 1.8337 & 1.8368 \end{pmatrix}.
$$

Note that each column of this matrix is approximately proportional to the corresponding column of the genuine mixing matrix A, and the three recovered signals give high SNR values, 25.3238, 26.0529 and 32.8434. Figure 2 shows the original source signals and the recovered signals, which also illustrates high similarity between the two signals. The results show that the COPICA method successfully solve this simulated BSS problem.

Example 2 In this example we demonstrate a real case with one near-Gaussian-tail signal. Three natural sounds of thunder, water and fire each containing 5000 sample points are used as the original signals. The sample kurtosises of these three natural sounds are 3.5323, 29.4978 and 16.6685 respectively. Note that the p-values of the Jarque-Bera test for these natural sounds are all less than 10^{-3}, which indicates non-Gaussianity. The first source (thunder sound) is a near-Gaussian-tail sample since its sample kurtosis is close to 3,

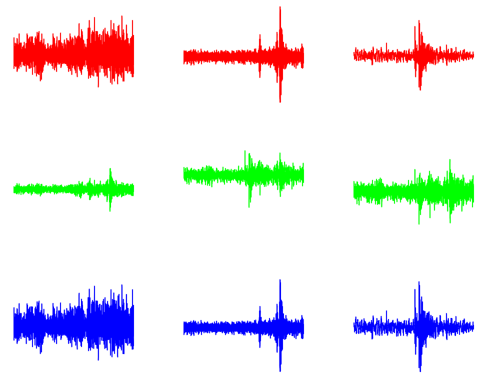

Fig. 3 The numerical results for three dimensional blind source separation problem with three natural sounds (thunder, water and fire). The *red lines* are the original sounds, *green lines* are their mixtures, and the *blue lines* are the recovered signals (Color figure online)

while the other two sources (water and fire sounds) are of heavy-tailed distributions. Using the same mixing matrix in Eq. (12) and the objective function defined in Eq. (13), after 100 iterations of the COPICA method, we obtained

$$
B^{-1} = \begin{pmatrix} 1.8946 & -2.0933 & 0.7112 \\ -1.8875 & 0.9933 & 1.5947 \\ -1.8888 & 1.0118 & 0.8202 \end{pmatrix},
$$

and the corresponding SNR values are 35.6150, 35.2765 and 32.3912. We also found great similarity in the original natural sounds and the recovered signals shown in Fig. 3.

Example 3 In this example we demonstrate a real case with three near-Gaussian-tail signals. Three sounds with 10000 sample points, boat engine, rain and wind, are used as the original signals. The values of their sample kurtosis are 3.20, 3.23 and 3.71, respectively. Note that the p-values of the Jarque-Bera test for the signals are all less than 10^{-3}, which indicates non-Gaussianity.

The mixing matrix A and the objective function are the same as in Example 2. After 100 iterations of the COPICA method, we obtain

$$
B^{-1} = \begin{pmatrix} 5.6205 & -3.3402 & 2.1439 \\ -6.0068 & 1.6872 & 3.4948 \\ -5.8871 & 1.7050 & 1.6141 \end{pmatrix},
$$

and the SNR values are 25.8270, 35.3203 and 23.8432 respectively. The time plots of the original natural sounds and the recovered signals are given in Fig. 4, again the result show that the COPICA method successfully separate the original natural sounds from their mixtures.

4 Comparisons with the FastICA

The FastICA (Hyvärinen and Oja 1997; Hyvärinen 1999a) is one widely used and efficient method for identifying independent components. The FastICA is a two-step method.

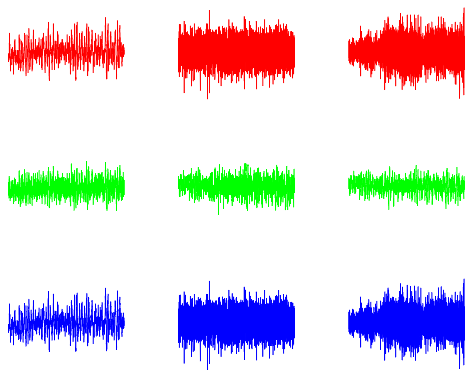

Fig. 4 The numerical results for three dimensional blind source separation problem with three natural sounds (boat engine, rain and wind). The *red lines* are the original sounds, *green lines* are their mixtures, and the *blue lines* are the recovered signals (Color figure online)

Table 1 The kurtoses of the mixture normal distributions with $\sigma_1 = 1$ and $\sigma_2 = 3$

	Near-Gaussian-tailed		Heavy-tailed		
	$p = 0.1$	$p = 0.2$	$p = 0.4$	$p = 0.6$	$p = 0.7$
Kur.	3.2570	3.5610	4.3698	5.6122	6.4879

After whitening the data at the first step, the FastICA find the independent components based on a fixed-point iteration scheme for finding a maximum of the non-Gaussianity of a linear projection. And the kurtosis or negentropy is used as the measure of non-Gaussianity. The computer program of the FastICA is available at the web-site,

http://www.cis.hut.fi/projects/ica/fastica/.

In this section, we compare the performance of COPICA and FastICA for the BSS problems via simulation study. The original independent sources are generated from mixture normal distributions with the pre-specified parameters σ_1, σ_2 and p. The corresponding kurtosis is $3\{p\sigma_1^4 + (1-p)\sigma_2^4\}/\{p\sigma_1^2 + (1-p)\sigma_2^2\}^2$. Thus we can generate samples with different kurtosis by choosing proper values of p, σ_1 and σ_2. In the following, (σ_1, σ_2) is set to be $(1, 3)$ and $p = 0.1, 0.2, 0.4, 0.6, 0.7$ respectively, and the corresponding kurtoses are given in Table 1.

Three dimensional BSS problem is considered for comparison. At each replication, three original sources are generated independently from the same mixture normal distribution with sample size T, and the observations are obtained by mixing the original sources with the matrix A given in Eq. (12). Two sample sizes $T = 1000$ and $T = 5000$ are considered and 100 replications are performed. Since there are three signals, 300 SNR values are obtained for each sample size. For each value of $p = 0.1, 0.2, 0.4, 0.6, 0.7$, we report the medians and standard deviations of the 300 SNR values obtained respectively by the two methods. The results are given in the first two columns of Table 2. The reason for reporting the medians instead of the means is to avoid the case of non-recovery (the FastICA method sometimes cannot recover the original sources for near-Gaussian-tailed case). We also compute the number of sources whose COPICA SNR values are larger than the FastICA SNR values denoted by N_1. And let N_2 be a 3×1 vector whose components denote the non-recovery numbers of FastICA for each source, and let N_3 (or N_4) be a 3×1 vector with each component representing the number that the COPICA (or FastICA) SNR values are less than 10 (including the number of non-recovery). The results of N_1–N_4 are given in the third and fourth columns of Table 2.

Table 2 Medians and standard deviations of the SNR for the BBS problem with three mixture-normal sources mixed by a fixed matrix, where N_1 denotes the number of sources whose COPICA SNR values are larger than the FastICA SNR values, N_2 is a 3×1 vector whose components denote the non-recovery numbers of FastICA for each source, and N_3 (or N_4) is a 3×1 vector with each component representing the number that the COPICA (or FastICA) SNR values are less than 10 (including the number of non-recovery)

		Median	std.	N_1 / N_2	N_3 / N_4
$T = 1000$	$p = 0.1$	28.05	4.78	297	(0,0,0)
		(7.56)	(6.14)	(15, 16, 11)	(74, 64, 75)
	$p = 0.2$	27.67	4.44	290	(0, 0, 0)
		(14.62)	(6.59)	(5, 4, 2)	(19, 19, 26)
	$p = 0.4$	27.73	4.66	274	(0, 0, 0)
		(18.63)	(5.77)	(0, 0, 0)	(1, 3, 5)
	$p = 0.6$	27.39	4.19	219	(0, 0, 0)
		(22.62)	(6.98)	(0, 0, 0)	(1, 0, 1)
	$p = 0.7$	27.05	4.08	225	(0, 0, 0)
		(22.82)	(6.25)	(0, 0, 0)	(1, 1, 0)
$T = 5000$	$p = 0.1$	31.81	4.51	290	(0, 0, 0)
		(15.09)	(6.52)	(1, 1, 2)	(19, 13, 23)
	$p = 0.2$	32.30	4.15	280	(0, 0, 0)
		(21.44)	(6.34)	(0, 0, 0)	(0, 0, 1)
	$p = 0.4$	31.43	4.90	225	(0, 0, 0)
		(26.27)	(6.11)	(0, 0, 0)	(0, 0, 0)
	$p = 0.6$	30.45	4.21	162	(0, 0, 0)
		(29.35)	(5.54)	(0, 0, 0)	(0, 0, 0)
	$p = 0.7$	30.98	4.93	184	(0, 0, 0)
		(29.38)	(5.69)	(0, 0, 0)	(0, 0, 0)

We summarized the results by the tail type of the original sources. The distribution is referred to "near-Gaussian-tailed" if the kurtosis is less than 4, to "heavy-tailed" if the kurtosis is greater than or equal to 4. In all cases, the COPICA method gives larger SNR medians and smaller standard deviations than the FastICA method. Note that there are 300 original sources for each pair (p, T), since all the values of $N_1 > 150$, the COPICA method attains higher SNR values more than half of all time. Significant dominance in the SNR medians and N_1 of the COPICA over the FastICA is apparent for smaller sample size ($T = 1000$) and near-Gaussian-tail case $p = 0.1, 0.2$. The non-recovered number of the FastICA method, N_2, are noted when $p = 0.1, 0.2$, $T = 1000$ and $p = 0.1$, $T = 5000$, which indicates the FastICA method might fail to recover the near-Gaussian-tailed signals. All the values of N_3 are equal to zero, implies the SNR values obtained by the COPICA are greater than 10 for all cases. Moreover, there are significant times (N_4) that the FastICA attains small SNR (≤ 10) values for the near-Gaussian-tailed case $p = 0.1, 0.2$, $T = 1000$ and $p = 0.1$, $T = 5000$. Based on the above, we conclude that for all generated sources, COPICA successfully identifies the three independent components, while FastICA works well for heavy-tailed sources, but may fail for the near-Gaussian-tailed sources. The reason might be due to the criterion of the FastICA is based on the kurtosis and negentropy which is not sensitive to near Gaussian-tailed distributions. However, the signals with kurtosis close to 3 do exist in real application. Recall the sample kurtosis of thunder, boat engine, rain and wind sounds in Examples 2 and 3 are all close to 3. We further applied the FastICA method to these two real sound examples. For the case with one near-Gaussian-tailed and two heavy-tailed signals (Example 2), the inverse matrix found by the FastICA is

$$B^{-1}_{\text{FastICA}} = \begin{pmatrix} 1.8889 & -2.0828 & 0.7548 \\ -1.8882 & 1.0274 & 1.5716 \\ -1.8880 & 1.0311 & 0.7970 \end{pmatrix},$$

and the SNR's are 37.0375, 51.6663 and 46.4812 which are all larger than those obtained by the COPICA method. While for the case with three near-Gaussian-tail signals (Example 3), the inverse transformation matrix found by the FastICA of is

$$B^{-1}_{\text{FastICA}} = \begin{pmatrix} 5.7296 & -3.3709 & 1.7740 \\ -5.5537 & 2.1166 & 3.9765 \\ -5.6351 & 2.0033 & 2.0973 \end{pmatrix},$$

and the SNR's are 18.3674, 18.3564 and 22.6044 which are all smaller than those found by the COPICA method. The results of the real sound examples also support the aforementioned simulation findings. Finally from Table 2, one can see that both methods improve their SNR median values when the sample size increases from $T = 1000$ to $T = 5000$.

In addition, we also compare the performance of the two methods by using random mixing matrix. The original sources are generated independently from a mixture-normal distribution with $\sigma_1 = 1$, $\sigma_2 = 3$ and $p \in \{0.1, 0.2, 0.4, 0.6, 0.7\}$. The size of each source is set to be 1000. However, in each replication, each component of the mixing matrix, A, is generated from $[-5, 5]$ uniformly such that A is invertible. That is the mixing matrix is different for each replication. The 100 simulation results are shown in Table 3. From Table 3 similar conclusions are obtained as from Table 2. That is COPICA recovers all original sources from their mixtures but FastICA might be fail for some near-Gaussian-tail sources, and overall COPICA attains higher SNR than FastICA, especially for the cases of near-Gaussian-tail sources.

Table 3 Medians and standard deviations of the SNR for the BBS problem with three mixture-normal sources mixed by a random matrix, where $N_1, \ldots, N_4$ are defined the same as in Table 2

	Median	std.	N_1 N_2	N_3 N_4
$p = 0.1$	27.97	4.82	298	(0, 0, 0)
	(7.07)	(6.32)	(20, 18, 12)	(72, 63, 66)
$p = 0.2$	27.95	4.31	288	(0, 0, 0)
	(13.97)	(6.15)	(0, 1, 1)	(19, 21, 23)
$p = 0.4$	27.51	4.31	260	(0, 0, 0)
	(19.64)	(6.34)	(0, 0, 0)	(4, 1, 4)
$p = 0.6$	27.90	4.16	236	(0, 0, 0)
	(21.64)	(6.37)	(0, 0, 0)	(0, 0, 0)
$p = 0.7$	27.67	4.46	241	(0, 0, 0)
	(22.64)	(6.39)	(0, 0, 0)	(1, 1, 0)

The infomax principle, maximizing the output entropy of a neural network with nonlinear outputs, has been applied to develop ICA algorithm in Bell and Sejnowski (1995), and this principle is closely related to the maximum likelihood approach. Hyvärinen (1999a) pointed out that the fixed-point scheme in FastICA can be directly applied to infomax type ICA algorithm by choosing the corresponding nonlinearity g, for example, $g(y) = -2\tanh(y)$ for heavy-tailed sources. We also studied the performance of the FastICA using the infomax principle with $g(y) = -2\tanh(y)$ for the three dimensional BSS problem with different mixture normal sources and the mixing matrix A given by Eq. (12). Since the results are similar to Table 2, thus details are omitted here.

5 COPICA vs. nonparametric rank-based approach

In this section, we compare the COPICA method with several nonparametric rank-based ICA approaches via simulation studies. Many non-linear dependence measures for a pair of continuous random variables (X, Y) are based on ranks. Among most commonly used are Kendall's τ and Spearman's ρ. Kendall's τ, is defined as the difference between probability of concordance and probability of discordance. Spearman's ρ is defined as the Pearson's correlation coefficient between the ranks of the two samples and for a given copula model. More details of these two measures can be found in Nelsen (2006). Another nonparametric approach for measuring the dependence is the Blomqvist's β (Schmid and Schmidt 2007). Recently, Kirshner and Póczos (2008) suggested using the Schweizer-Wolff σ_{SW}, defined as

$$\sigma_{SW} = 12 \int_{[0,1]^2} \left| C(u, v) - uv \right| du dv, \quad (14)$$

to measure the pairwise dependence (Schweizer and Wolff 1981). They proposed an algorithm for ICA by replacing the copula function in (14) with empirical copula.

Most of the ICA algorithms use an approximation to mutual dependence as their objective functions. And the performance of an ICA algorithm depends on how accurate the approximate dependence measure is. The above four nonparametric dependence measures are all zero if X and Y are independent. However, the converse is not necessary true. Kirshner and Póczos (2008) showed that σ_{SW} is more robust than Kendall's τ and Spearman's ρ with added outliers and noise. However, to obtain the nonparametric multivariate empirical distribution requires an intensive computational effort when sample size is large or dimensionality is high. It might even collapse when dimension is too high say higher than 4. A semiparametric approach such as COPICA, which estimates the joint distribution via copula function and one dimensional empirical distribution, can provide an alternative to greatly relieve the computational burden.

We conduct several simulation studies to compare the performance of COPICA with nonparametric ICA methods based on Kendall's τ, Spearman's ρ, Blomqvist's β, and Schweizer-Wolff σ_{SW}. Basically COPICA method attains higher SNRs than the ICA methods based on Kendall's τ, Spearman's ρ or Blomqvist's β, and is competitive with the ICA method via Schweizer-Wolff σ_{SW}. To save the space, we only show comparison between COPICA and ICA method via Schweizer-Wolff σ_{SW} (ICA_SW). Note that the ICA-SW used here is similar to a ICA algorithm proposed by Kirshner and Póczos (2008). In the simulation study to compare the ICA performance of COPICA and ICA-SW, various types of heavy-tailed sources are used to generating original independent sources. Similar to the experimental setting of Bach and Jordan (2002), we consider 12 different one-dimensional densities with kurtosis greater than 3, shown in Fig. 5, including those densities commonly used in finance (a)–(d), in reliability and lifetime modeling (e)–(h) and (k) and in communications (i), (j) and (l).

For the bivariate case, we generate two independent sources each of size 1,000 from the same density, normalize the sources, and then mix them by a matrix whose elements are randomly sampled from $[-5, 5]$. We compute the SNR's of the COPICA and ICA_SW for the 12 heavy-tailed sources, respectively. Figure 6 plots the medians of the differences in the SNRs of COPICA and ICA_SW (COPICA-ICA_SW) based on 100 replications. Since most of the medians in Fig. 6 are around zero, the results indicate the ICA performance of the semiparametric COPICA method is competitive with the nonparametric ICA_SW method, Also,

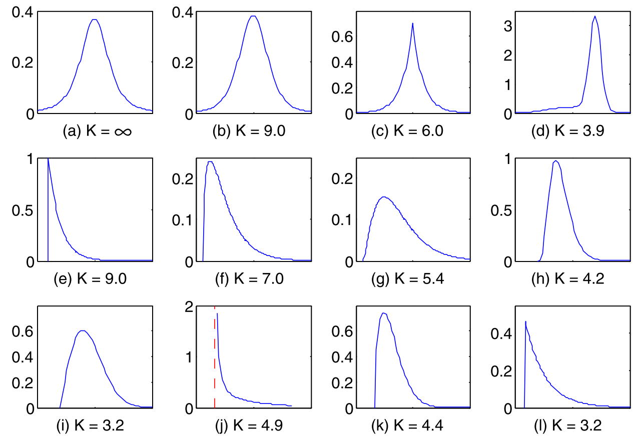

Fig. 5 Probability density functions of heavy-tailed sources. (**a**) Student t with 3 degrees of freedom (d.f.); (**b**) Student t with 5 d.f.; (**c**) double exponential distribution; (**d**) mixture of two Gaussians, where the density is $f(x) = 0.5\phi(x+0.5) + \phi(2x-1)$; (**e**) exponential distribution; (**f**) Chi-square distribution with 3 d.f.; (**g**) Chi-square distribution with 5 d.f.; (**h**) gamma distribution; (**i**) Rayleigh distribution; (**j**) Nakagami distribution; (**k**) Weibull distribution; (**l**) Rician distribution (Color figure online)

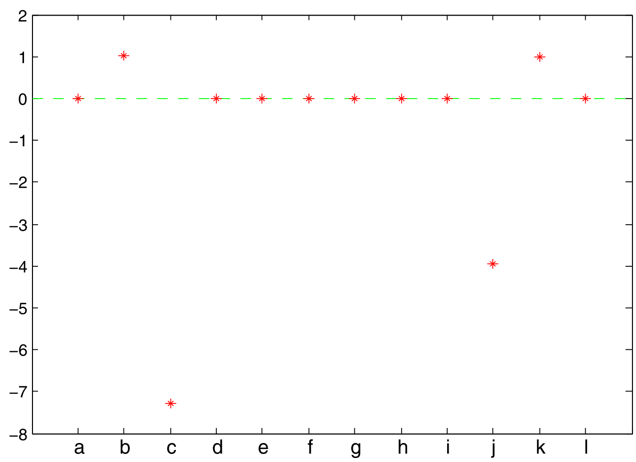

Fig. 6 The medians of SNR(COPICA)-SNR(ICA_SW) absed on 100 replications, where the two original independent sources are generated from the same density in (**a**)–(**l**) with sample size 1,000 (Color figure online)

in Table 4 we list the numbers of SNRs greater than 10 (or 15) in recovering the 200 mixed signals (of two sources and 100 replications) for COPICA and ICA_SW. The results show both methods recover almost all the mixed signals with SNRs greater 10. The SNRs of both methods increase as the sample size increases from 1,000 to 5,000, and the result based on 5,000 samples are similar to the ones shown in Fig. 6 and Table 4. However, the computational time of ICA_SW increases quadratically (i.e., 25 times) while the computational time of COPICA only increases linearly (i.e., 5 times). In the semiparametric COPICA approach, sorting is only needed for one dimensional marginal empirical distributions for each source and the joint distribution is

Table 4 The number of SNRs greater than 10 (or 15) of COPICA and ICA_SW in the 2-dimensional cases

Case	#{SNR ≥ 10}		#{SNR ≥ 15}	
	COPICA	ICA_SW	COPICA	ICA_SW
(a)	200	200	197	198
(b)	200	193	188	176
(c)	199	200	191	200
(d)	200	200	200	200
(e)	200	200	200	200
(f)	200	200	200	200
(g)	200	200	200	200
(h)	200	200	198	200
(i)	200	200	195	200
(j)	200	200	200	200
(k)	200	200	200	200
(l)	200	200	197	200

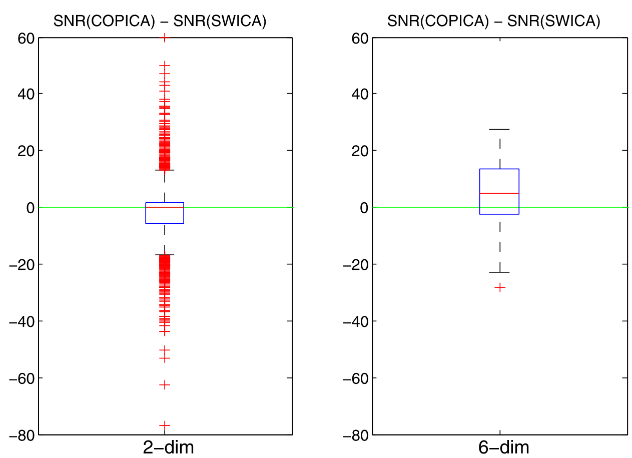

Fig. 7 Box-plots of the SNR difference (COPICA - ICA_SW) for 2-dimensional (*left*) and 6-dimensional (*right*) sources generated randomly from the 12 heavy tailed densities (a)–(l) (Color figure online)

linked by parametric copulae. While for the nonparametric approach ICA_SW, sorting is required for both one dimensional and two dimensional joint copulae. In sum this means that our method is computationally lighter than their technique.

Another scenario of the experiments is to generate each independent source randomly from the 12 heavy tailed densities (a)–(l) (therefore the sources are not necessary of the same distribution) and then mix the sources by a matrix with elements sampled randomly from $[-5, 5]$. We show the SNR differences (COPICA-ICA_SW) of 2-dimensional and 6-dimensional cases in Fig. 7. The SNR's are obtained after 90 iterations, the sample size of each source is 1,000, and the numbers of replication are 1,000 and 100 for the 2-dimensional and 6-dimensional cases, respectively. The results show that the COPICA method is still competitive with the ICA_SW method for the random mixing bivariate cases. Nevertheless, in the 6-dimensional case the COPICA method attains higher SNR than the ICA_SW method on the average.

6 Conclusions and discussions

In this article, a new ICA method, COPICA, is proposed. Similar to the FastICA, the COPICA method is also a two-step procedure. After whitening the data, COPICA projects the whiten data into the n-dimensional plane simultaneously, and this projection is chosen in terms of the parameters of the pre-specified copulae. Thus in COPICA, ICA problem is transformed to a minimization problem whose objective function is defined by the weighted combination of the divergence functions of copula parameters. The weights in the objective function are chosen to be inverse proportional to the standard deviations of the parameter estimators. Here the copula parameters are estimated via CML approach. Thus given a rotation matrix, R and the current recovered data, $Y = R(WX)$, the empirical marginal distributions of Y_i are obtained first and then the copula parameter vector, θ, is found by maximizing Eq. (4). Hence we only have parameterized copula model assumption and do not have other assumptions on the marginal distribution. That is why we treat our COPICA as a semiparametric approach.

By comparing COPICA with the commonly used FastICA method and the nonparametric ICA methods, we find that the copula parameter based divergence function of the three copulae Gumbel, Clayton and Gaussian provide useful dependency measures when the observations come from a linear mixing model. The simulation and real data studies indicate that COPICA attains higher SNR than FastICA in BSS problems, especially when the original sources come from near-Gaussian-tailed distributions. Also, the COPICA is shown to have higher SNRs than the ICA_SW on the average in the 6-dimensional case. Another interesting problem is to study the COPICA method for multi-modes densities, which is referred to our future work.

We investigate the sensitivity of COPICA w.r.t. weights via the BSS problem. A preliminary study is conducted here. In addition to $(\omega_1, \omega_2, \omega_3, \omega_4) = (200, 300, 200, 500)$ in Example 1, six more weight combinations $(\omega_1, \omega_2, \omega_3, \omega_4)$ are considered and five mixture-normal distributions with $p = 0.1, 0.2, 0.4, 0.6, 0.7$ are considered. In each replication three independent sources of length $T = 1000$, generated from a mixture-normal distribution, are mixed by the matrix A defined in (12). For each p, the average SNRs of 3×100 independent copies are obtained for each weight combination. The highest average SNR among the seven weight combinations is taken as the benchmark value. The ratios of the average SNR of each weight combination to the benchmark SNR are reported in Table 5. The

Table 5 The average SNR ratios of COPICA for different weight combinations of the objective function (13)

Weights	p				
	0.1	0.2	0.4	0.6	0.7
(200, 300, 200, 500)	0.84	0.87	0.82	0.86	0.83
(200, 200, 200, 200)	1.00	1.00	1.00	1.00	1.00
(0, 200, 200, 200)	0.97	0.95	0.98	0.89	0.99
(200, 200, 0, 200)	0.94	0.89	0.75	0.74	0.79
(200, 200, 0, 0)	0.84	1.00	0.84	0.89	0.96
(200, 0, 200, 0)	0.94	0.87	0.76	0.78	0.83
(200, 0, 0, 200)	0.78	0.61	0.31	0.25	0.30
Highest ave. SNR	6.25	7.56	14.83	17.79	17.98

initial rotation angles are set to be zero and the number of iterations in the SA algorithm are set to be 100. The SNR ratios of the first six weight combinations range from 0.74 to 1, which indicates the COPICA is only slightly sensitive to these six weight combinations. The weight combinations (200, 300, 200, 500) has relatively robust performance among 7 weight cases. The weight combinations (200, 200, 200, 200) and (0, 200, 200, 200) are the best two obtaining the high SNRs, while the combination (200, 0, 0, 200) has the poorest performance in this scenario. This suggest the necessity of including the Gumbel and Clayton copulae in the objective function. Moreover, the highest average SNR of the COPICA increases as p increases (equivalently the kurtosis increases, see Table 1) after 100 iterations in the SA algorithm. To find general rules for weight selection of high SNR further studies are still needed.

Acknowledgements The authors gratefully acknowledge the National Science Council in Taiwan, National Center for Theoretical Sciences (South), Tainan, Taiwan and the Deutsche Forschungsgemeinschaft through the SFB 649 "Economic Risk", Humboldt-Universitat zu Berlin. This work was supported in part by National Science Council under grants NSC 96-2118-M-390-002- (Chen), NSC 100-2118-M-110-001-003- (Guo) and NSC 101-2118-M-390-002- (Huang).

References

Abayomi, K., Lall, U., de la Pena, V.: Copula Based Independent Component Analysis. Working paper (2008)

Abayomi, K., de la Pena, V., Lall, U., Levy, M.: Quantifying sustainability: methodology for and determinants of an environmental sustainability index. In: Luo, Z.W. (ed.) Green Finance and Sustainability: Environmentally-Aware Business Models and Technologies, pp. 74–89 (2011)

Bach, F.R., Jordan, M.I.: Kernel independent component analysis. J. Mach. Learn. Res. **3**, 1–38 (2002)

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

Blaschke, T., Wiskott, L.: CuBICA: independent component analysis by simultaneous third- and fourth-order cumulant diagonalization. IEEE Trans. Signal Process. **52**, 1250–1256 (2004)

Chen, R.-B., Wu, Y.N.: A null space method for over-complete blind source separation. Comput. Stat. Data Anal. **51**, 5519–5536 (2007)

Comon, P.: Independent component analysis—a new concept? Signal Process. **36**(3), 287–314 (1994)

Genest, C., Nešlehová, J., Ziegel, J.: Inference in multivariate Archimedean copula models. Test **20**, 223–256 (2011)

Ghosh, S., Henderson, S.: Behaviour of the NORTA method for correlated random vector generation as the dimension increases. ACM Trans. Model. Comput. Simul. **13**, 276V294 (2003)

Gretton, A., Bousquet, O., Smola, A.J., Schölkopf, B.: Measuring statistical dependence with Hilbert-Schmidt norms. In: ALT, pp. 63–78. Springer, Heidelberg (2005)

Grønneberg, S., Hjort, N.L.: The Copula Information Criterion. Technical report, Department of Math., University of Oslo, Norway (2008)

Hyvärinen, A.: Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw. **10**, 626–634 (1999a)

Hyvärinnen, A.: Sparse code shrinkage: denoising of nongaussian data by maximum likelihood estimation. Neural Comput. **11**, 1739–1768 (1999b)

Hyvärinen, A., Oja, E.: A fast fixed-point algorithm for independent component analysis. Neural Comput. **9**, 1483–1492 (1997)

Hyvärinen, A., Oja, E.: Independent component analysis: algorithms and application. Neural Netw. **13**, 411–430 (2000)

Kidmose, P.: Blind Separation of Heavy Tail Signals. Ph.D. Thesis, Technical University of Denmark, Lyngby (2001)

Kirkpatrick, S., Gelatt, C.D. Jr., Vechhi, M.P.: Optimization by simulated annealing. Science **220**, 671–680 (1983)

Kirshner, S., Póczos, B.: ICA and ISA using Schweizer-Wolff measure of dependence. In: International Conference on Machine Learning (ICML-2008) (2008). http://icml2008.cs.helsinki.fi/papers/551.pdf

Kotz, S., Nadarajah, S.: Extreme Value Distributions. Theory and Applications. Imperial College Press, London (2000)

Learned-Miller, E.G., Fisher, J.W.: ICA using spacings estimates of entropy. J. Mach. Learn. Res. **4**, 1271–1295 (2003)

Lee, T.-W.: Independent Component Analysis. Theory and Applications. Kluwer Academic, Dordrecht (1998)

Liu, J.S.: Monte Carlo Strategies in Scientific Computing. Springer, New York (2001)

Ma, J., Sun, Z.: Copula component analysis. In: Proceedings of the 7th International Conference on Independent Component Analysis and Signal Separation, pp. 73–80. ACM, London (2007)

Mardia, K.V.: A translation family of bivariate distributions and Fréchet's bounds. Sankhya A **32**, 119–122 (1970)

McNeil, A.J., Nešlehová, J.: From Archimedean to Liouville copulas. J. Multivar. Anal. **101**, 1772–1790 (2010)

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H.: Equation of state calculation by fast computing machines. J. Chem. Phys. **21**, 1087–1092 (1953)

Nelsen, R.B.: An Introduction to Copulas. Springer, Berlin (2006)

Olshausen, B.A., Field, D.J.: Natural image statistics and efficient coding. Network **7**, 333–339 (1996)

Schmid, F., Schmidt, R.: Nonparametric inference on multivariate versions of Blomqvist's beta and related measures of tail dependence. Metrika **66**, 323–354 (2007)

Schweizer, B., Wolff, E.F.: On nonparametric measures of dependence for random variables. Ann. Stat. **9**, 879–885 (1981)

Shen, H., Jegelka, S., Gretton, A.: Fast kernel-based independent component analysis. IEEE Trans. Signal Process. **57**, 3498–3511 (2009)

Sklar, A.: Fonctions de Repartition à n Dimensions et Leurs Marges. Publ. Inst. Stat. Univ. Paris **8**, 229–231 (1959)

Sklar, A.: Random variables, distribution functions, and copulas—a personal look backward and forward. In: Ruschendorf, L., Schweizer, B., Taylor, M.D. (eds.) Distributions with Fixed Marginals and Related Topics, pp. 1–14. Institute of Mathematical Statistics, Hayward (1996)

Sodoyer, D., Girin, L., Jutten, C., Schwartz, J.-L.: Speech extraction based on ICA and audio-visual coherence. In: Proceedings of the 7th International Symposium on Signal Processing and Its Applications (ISSPA'03), vol. 2, pp. 65–68 (2003)

Tsai, A.C., Liou, M., Jung, T.-P., Onton, J.A., Cheng, P.E., Huang, C.-C., Duann, J.-R., Makeig, S.: Mapping single-trail EEG records on the cortical surface through a spatiotemporal modality. NeuroImage **32**, 195–207 (2006)

Yu, L., Verducci, J.S., Blower, P.E.: The tau-path test for monotone association in an unspecified subpopulation. In: Application to Chemogenomic Data Mining Statistical Methodology, vol. 8, pp. 97–111 (2011)