Independent Component Analysis: A Review with Emphasis on Commonly used Algorithms and Contrast Function

Rasmikanta Pati[1], Arun K. Pujari[2,4], Padmavati Gahan[3], Vikas Kumar[2]

[1] Sambalpur University Institute of Information Technology,
India

[2] Central University of Rajasthan,
India

[3] Sambalpur University Institute of Information Technology,
Department of Business Administration,
India

[4] University of Hyderabad,
India

rkpati@suiit.ac.in, {arun.k.pujari, pgahan7, vikas007bca}@gmail.com

Abstract. Independent Component Analysis (ICA) is an effective instrument for separating mixture signals from their blind sources that are specified or over-determined in the fields of signal processing, machine learning, data mining, finance, bio-medical, communications, artificial intelligence etc., ICA focuses primarily on finding an Objective Function (Contrast Function) and an appropriate optimization method to solve the problem. Different methods of ICA work out variously depending on how one models the contrast functions between themselves. ICA focuses mainly on finding components that are as independent as possible and as non-Gaussian as possible of an observed unexplained non-Gaussian Signal Mixture. ICA is an extremely important subject of great interest in numerous technological and scientific applications. In this article, we review a few different contrast functions in addition to the much earlier survey of Aapo Hyvarinen and widely used existing ICA algorithms in different scenarios for source separation. This article presents basic ideas on ICA, ICA algorithms and contrast functions.

Keywords. Independent component analysis, unsupervised learning, particle swarm optimization, higher order statistics, blind source separation.

1 Introduction

In data-driven era, data generation, data measurement and data processing are important steps of computation. With the advent of information technology that has successfully percolated to the bottom-most layers of our daily lives, enormous amount of data is being generated, effortlessly and inadvertently. In addition, to add to our misery, relevant and not-so-relevant data are generated indistinguishably and measured together, and one needs to put extra effort to separate the relevant data for the irrelevant one.

In that sense, the data that we access is of enormous volume, but might contain relatively less information than we might need. The problem persists in many aspects of life and in many disciplines of scientific research. Typical real-life situations are mixtures of simultaneous sounds or human voices that have been picked up by several microphones, brain signal measurements from multiple EEG sensors, several radio signals arriving at a portable phone, or multiple parallel time series obtained from some industrial process.

A well-known example of noisy room, known as Cocktail Party Problem, is appropriate to state here. Suppose in a cocktail party many people are talking at the same time and isolation of the individual signals is of interest. A guest at a cocktail party must focus on one person's voice in a room filled with competing voices and other noises. This 'cocktail-party problem' is solved effortlessly by humans with binaural hearing. Another example in the context of image processing can also be stated here. Consider the problem of removing blur from an image due to camera motion.

A photographer tries to take a photo, but their camera is not steady when the aperture is open. Each pixel in the sensor array records the combination of all lights within an integration period from the intended image along the camera motion trajectory. Thus, in blurred image each recorded pixel is the mixture of multiple image pixels. De-blurring an image requires recovering the original image as well as the underlying camera motion trajectory from the blurry image. Both the cocktail party and de-blurring problems are ill-posed and additional information must be employed to recover a solution.

The term Blind Source Separation (BSS) is coined to characterize this problem. Simply speaking, any real-life data measurement process measures the combined data of relevant and irrelevant data, which result from their respective independent sources and therefore, it is necessary to separate out the data of relevant source from the data of other sources. That said, progress has been made when the interactions between signals are simple in particular, linear interactions, as in both examples. When the combination of two signals results in the superposition of signals, we term this problem a linear mixture problem.

In mathematical terms, we need to find a suitable, proper multivariate description of random vectors. The representation is also referred, for simplicity, as linear transformation of initial data. To put it another way, each representative component is a linear blend of initial variables. There are renowned linear modes of transformation which includes Factor Analysis [1], Principal Component Analysis (PCA) [2, 3], and Projection Pursuit [4] etc.

Independent Component Analysis is a technique of data transformation that finds independent sources of activity in recorded mixtures of sources. Independent Component Analysis (ICA) is a computational technique for revealing hidden factors that underlie sets of measurements or signals. ICA assumes a statistical model whereby the observed multivariate data, typically given as a large database of samples, are assumed to be linear or nonlinear mixtures of some unknown latent variables.

Independent Component Analysis (ICA) was introduced in 1986 [5]. However, in that paper, there is no theoretical explanation was presented and the proposed algorithm was not applicable in several cases and in 1991 partial theoretical structure was laid down [6]. Thus, the ICA technique remained mostly unknown till 1994, where the name of ICA appeared and introduced as a new concept [7] where in it is suggested that signals of the sources are independent.

Several algorithms have been proposed since then for calculating ICA techniques, which differ among themselves in the handling of statistical independence, on estimation of the separation matrix, and use of statistics of higher order. For BSS, presumably, the source signals may be combined linearly or nonlinearly. ICA is ideal if the signals are supposed to be combined linearly. Several other methods with a nonlinear mixture assumption exist for BSS [8, 9, 10]. ICA's linear mixture model attempts to separate source signals according to certain assumptions:

1. The source vectors are statistically independent.
2. The mixing matrix (A, as defined in the next section) should be a square and full rank.
3. The source matrix (S, as defined in the next section) does not have any external noise.
4. The data are centered (zero mean).
5. The signals from the source should be non-Gaussian probability density function (pdf) with one source expected, which may be Gaussian.

Independent Component Analysis (ICA) has been employed for nearly 30 years for unmixing of complex signals.

Unmixing the signals without the use or even providing any background knowledge about signals of the source without mixing them up is generally identified as Blind Source Separation (BSS). ICA is the unique BSS technique developed with respect to signal processing. The key concern of the ICA is extraction of "source signals" from a set of observed signal mixtures and their mixing coefficients (proportion). That means, this way the derived information can be translated further straight.

Independent components analysis (ICA) is a probabilistic method, whose goal is to extract underlying component signals that are maximally independent and non-Gaussian, from mixed observed signals. The mixing coefficients are also unknown. The latent variables are non-Gaussian and mutually independent and they are called the independent components of the observed data. By ICA, these independent components, also called sources or factors, can be found.

Thus, ICA can be seen as an extension to Principal Component Analysis and Factor Analysis. ICA is a much richer technique, however, capable of finding the sources when these classical methods fail completely. In many cases, the measurements are given as a set of parallel signals or time series.

With the aim of maximizing non-Gaussianity or minimizing Gaussianity in order to reach source as independently as possible, ICA is therefore an optimization problem. The hypothesis for independence has to be approximated, thus turning the estimation of the sources into an optimization problem described by a contrast (cost) function that is minimal when the sources estimation are as far as independent.

Somehow or other, contrast function is an independent measure. This idea led to the concept of a contrast function: by definition, it is a criterion, which leads to an acceptable solution of the BSS problem by maximizing the separator, that is every row of the mixing-separating system is extracted one by one, that is, source signal is built component by component. This approach is called deflation. When the entire source signal (multi-unit) is simultaneously collected, it is called symmetric approach. ICA method in particular can be expressed as a sum of the Contrast Function and Optimization Algorithm.

More than 30 different ICA algorithms are already available so far [11]. ICA technique basically deals with two independent classes, i.e. Single method of optimization used for different contrast functions or different method of optimization used for single contrast functions.

The widespread and interdisciplinary applications of ICA in the context of image processing, text mining, data mining, audio signal processing, biomedical signal processing, and time series applications motivate us to present ICA theory and its most used methods in one article.

The goal of this review is to explain ICA and to present some of the widely used algorithms for ICA computation as well as some more contrast functions in addition to Aapo Hyvarinen's much earlier survey in 1999 [12].

The remainder of the survey is organized in the following way. We give introduction to ICA in section 2. Section 3 addresses different higher order statistical notations that are useful in ICA. Section 4 gives six different ICA algorithms. Section 5 describes the various ICA contrast functions. Section 6 gives applications of ICA in real world. Finally, section 7 of the survey is conclusion and references.

2. Independent Component Analysis (ICA)

Independent Component Analysis (ICA) [12, 13, 14, 15, 16] is a statistical tool for the transformation of an observed multidimensional random vector into statistically independent components. This approach is used to separate the mixed signals. PCA functions only in second-order statistics and provides optimal data for the Gaussian distribution sets. ICA is a PCA extension designed to optimize non-Gaussianity or minimize the Gaussianity of the datasets. ICA attempts to find independent components by assuming their statistical property of higher order.

The random vectors, x and s represent the data in ICA and the independent components respectively. ICA has many algorithms such as *FastICA* [17], *projection pursuit* [15], and *Infomax* [15, 18].

The main goal of these algorithms is to extract independent components by (1) maximizing the non-Gaussianity, (2) minimizing the mutual information, or (3) using maximum likelihood (ML) estimation method [19]. However, ICA suffers from a number of problems such as over-complete ICA and under-complete ICA.

Let us consider an observed m dimensional column vector $x(m) = [x_1, x_2, ..., x_m]^T$ which represents a linear combinations of n elements $(n \leq m$) n -dimensional elements $s(n) = [s_1, s_2, ..., s_n]^T$ those are different from the statistics (or are as independent as possible). So the ICA model is:

$$x(m) = As(n), \qquad (1)$$

where A is a linear matrix mixture of order $m \times n$. The input elements are usually statistically dependent due to the mixing phase although the elements were not original. Both the mixing matrix and independent components (ICs) - $s_i, i = 1, 2, ..., n$ are unknown. If a demixing matrix W, that produces $y(m)$ can be found then that will give components which are statistically independent:

$$y(m) = Wx(m) = WAs(n). \qquad (2)$$

The model assumed that the data variables were linear or nonlinear mixtures of these latent variables, and the type of mixing was also unknown. The latent variables are not Gaussian and should be mutually independent. They are referred to as independent components of the data observed.

This approach is called blind because there is no much knowledge about both the mixing matrix A and the matrix of the source s. In addition to this, ICA method can be described as finding a linear transformation, which maximizes the $\hat{s}$ non-Gaussianity. The Matrix W is de-mixed by optimizing cost function. Specific cost functions such as negentropy, kurtosis, etc. can be used for ICA method. Therefore, various methods for computing W exist in ICA method.

That said, progress has been made when the interactions between signals are simple – in particular, linear interactions, as in both of these examples.

When the combination of two signals results in the superposition of signals, we term this problem a linear mixture problem. The goal of ICA is to solve BSS problems, which arise from a linear mixture.

Furthermore, the metrics of cumulants, likelihood function, negentropy, kurtosis, and mutual information have been developed to obtain a demixing matrix in different adaptations of ICA-based algorithms. FastICA [18], [16] was developed to maximize non-Gaussianity with relative speed and simplicity. Recently, Zarzoso and Comon [20, 21] proposed the Robust Independent Component Analysis (R-ICA) method for better convergence performance.

They used a truncated polynomial expansion, rather than the output marginal probability density functions, to simplify the estimation process. Moreover, in [19], the authors developed the rapid ICA algorithm which takes advantage of multi-step past information with respect to a fixed-point method in order to augment the non-Gaussianity among the estimated signals. In [7, 22, 23], the authors have presented ICA methods using mutual information. They constructed a formulation by minimizing the difference between the joint entropy and the marginal entropy among the estimated sources. Moreover, the Euclidean distance divergence (ED-DIV) and the Kullback divergence (Kl-DIV) were used as the measure functions for nonnegative matrix factorization (NMF) problems in [24].

3. Definition of Independence and Higher Order Statistics

The various ICA algorithms could be divided into two classes due to their independent descriptions: algorithms, which maximizes the non-Gaussian complexity of the components or minimizes mutual information. ICA makes sense when you look for components which are absolutely as non-Gaussian as possible.

In fact, when a Gaussian distribution fits a random variable, all of those moments and order cumulants above 2 are null [15, 25]. Locating the ICs therefore implies detecting signals of its moments and order cumulants above 2.

Therefore, different notations need to be introduced to present and define contrast functions used in ICA.

3.1 Moments

For a variable, the ith moment m_i is equal to:

$$m_i = E\{x^i\}, \tag{3}$$

where E is the expectation and for $i = 1, m_1 = mean(x)$.

A variable's moments define its function of probability density, that is, its distribution.

3.2 Central Moments

For a variable, the ith central moment μ_i is equal to the moment of the centered variable x, i.e.:

$$\mu_i = E\left\{(x - m_1)^i\right\}. \tag{4}$$

Hence:

The mean of x is $\mu_1 = 0$ and, the variance of x is $\mu_2 = \sigma^2$.

The third moment $\mu_3 = E\{(x - m_1)^3\}$ is classified as *skewness*, and is a measure of distribution asymmetry. Skewness may be positive, negative or null for Gaussian distribution.

For variable x the fourth moment is $\mu_4 = E\{(x - m_1)^4\}$. It's related to its kurtosis, which reflects the pointedness or flatness of the distribution of the value.

3.3 Kurtosis

Under Central Limit Theorem, that declares the linear combination of independent random variables on finite support probability density functions (pdfs) tend to a Gaussian distribution. In general, higher-ordered statistics such as fourth order cumulant or kurtosis are used for non Gaussian measurement. When the data is pre processed to show unit variance, the kurtosis shall be equal to the fourth moment in the data.

Kurtosis, defined for a centered variable x, as:

$$k = E\{x^4\} - 3 \cdot [E\{x^2\}]^2, \tag{5}$$

which is a non-Gaussianity measurement for distribution.

When the data is whitened and centered i.e., $E\{x^2\} = 1$, kurtosis is same as :

$$k(x) = E\{x^4\} - 3,$$

when $k(x) = 0$, the distribution declared as Gaussian, similarly when $k(x) > 0$ and $k(x) < 0$ the distribution is declared as super-Gaussian and sub-Gaussian respectively. The probability density function peak is very sharp for super-Gaussian and peak is rather flat in case of sub- Gaussian.

It is thus possible to calculate non-Gaussian components, maximizing their kurtosis absolute value. It is possible to optimize the components' independence by maximizing each individual kurtosis (maximum non-Gaussianity) while minimizing their mutual kurtosis (minimum non-Gaussianity), that may be described as the fourth – order cumulant function.

Thanks to its computational and mathematical simplicity, kurtosis has already been used in ICA as a measure of non-Gaussianity and in related fields. It has a linear structure, so mathematically:

$$kurt(z_1 \pm z_2) = kurt(z_1) \pm kurt(z_1),$$

and

$$kurt(\alpha z_1) = \alpha^4 kurt(z_1),$$

where α is constant.

Kurtosis is easy to calculate but is of poor statistical significance. Therefore, a better measure of non-Gaussianity is required for kurtosis.

3.4 Cumulants

Covariances applied in second-order statistics can be compared with cumulants. The first three cumulants have their moments equal, for centered variables, i.e.:

$$k_1 = 0,$$

$$k_2 = E\{x^2\} \text{ is the variance of } x,$$

$$k_3 = E\{x^3\}.$$

The cumulant of fourth-order is expressed as:

$$k_4 = E\{x^4\} - 3[E\{x^2\}]^2 = k(x), \qquad (6)$$

and hence is same with kurtosis.

Fourth-order auto- and cross-cumulants of 4 vectors u_i, u_j, u_k and u_l are specified as:

$$k_4\{u_i,u_j,u_k,u_l\}=E\{u_i u_j \cdot u_k u_l\}-E\{u_i u_j\}\cdot E\{u_k u_l\}$$
$$-E\{u_i u_k\}\cdot E\{u_j u_l\}-E\{u_i u_l\}\cdot E\{u_j u_k\} \qquad (7)$$

From the general viewpoint, one may note that the auto-cumulant fourth order of a centered variable is identical to its kurtosis. Cumulants may be represented as a tensor. Cumulant tensor is the simplification of the covariance matrix with diagonal auto-cumulants. One vector is characterized with auto-cumulants that corresponds to variable variance, whereas two vectors are characterized by cross-cumulants that corresponds to variance of two variables. The independent vectors statistically generate maximum auto-cumulant and cumulant tensor of null off-diagonal elements.

Kurtosis is highly sensible to outliers and is a reasonable approximation of non-Gaussianity; therefore it is non-robust estimation of non-Gaussianity. Negentropy is another measure of the (non-)Gaussianity variable and robust, hence preferred in place of kurtosis.

3.5 Negentropy

Negentropy, which is based on the information theoretic quantity of (differential) entropy, can calculate non-Gaussianity. The sum of the product of each observation's probability and their log probabilities is called as entropy of a discrete variable. As Hyvarien explains, "Random-variable entropy may be understood as the degree of information provided by variable observation.

The more the variable is 'random', the greater its entropy, i.e., unstructured and unpredictable. A Gaussian variable is the biggest entropy with equal variance of all the random variables" [26].

Hence, entropy may also be a valid criterion for estimating a variable 's non-Gaussianity. For a variable x, it is specified as:

$$H(x) = -\sum_i P(x = a_i)\log P(y = a_i) \qquad (8)$$

and hence, it carries a negative value. At the other hand, entropy is called differential entropy for a continuous function, as given by the function integral times of the function log i.e.:

$$H(x) = -\int p(x)\log p(s)ds \qquad (9)$$

Negentropy is never negative, so if x has a Gaussian distribution, it is zero. Negentropy has a valuable property and its invertible linear transformation is invariant. It is also a robust non-Gaussian measure. One downside of negentropy is very hard to measure. This is why it needs approximation. The approximation is expressed by:

$$J(x) = H(x_{Gauss}) - H(x), \qquad (10)$$

where x_{Gauss} is a random Gaussian variable x with the same matrix of covariances. The more the variable is "non-Gaussian", the higher the negentropy value. Hence, one should seek to maximize the negentropy of component when looking for ICs. In real cases, the value of the negentropy is nevertheless hard to estimate, so usually one needs to work with a more simple approximation. Hyvarinen [26] includes a number of such approximations as shown below:

$$J(x) = \frac{1}{12}E\{x^3\} + \frac{1}{48}kurt(x)^2 \qquad (11)$$

where x is the variable of zero mean and unit variance. This estimate however is based on kurtosis, that is not a dependable estimator. A further approximation may instead be used:

$$J(x) \approx k[E\{G(x)\} - E\{G(v)\}]^2, \qquad (12)$$

where v is a variable with mean 0 and unit variance, k is a constant and G is a non quadratic function. Again, Hyvarinen suggests two important choices of G:

$$G_1(u) = \frac{1}{a_1}\log\cosh a_1 u \qquad (13)$$

and

$$G_2(u) = -\exp\left(\frac{-u^2}{2}\right). \qquad (14)$$

3.6 Maximum Likelihood Estimation

Maximum Likelihood is a traditional method used for independent component estimations. This is built on the density of a linear transform using well-known results. Taking the basic ICA model into consideration, $x(m) = As(n)$, the density p_x of the mixture signal observed may be formulated as:

$$p_x[x(m)] = \frac{1}{|\det A|} p_s[s(n)] = |\det W| p_s[Wx(m)] \qquad (15)$$

where $W = A^{-1}$. As the source is considered to be statistically independent and the mixture signal density is the product of the sources' marginal densities, so eq. 15 can be expressed with a function of $W = (w_1, w_2,..., w_n)^T$ and x, giving:

$$p_x[x(m)] = |\det W| \prod_{i=1}^{N} p_s[w_i^T x(m)] \qquad (16)$$

Suppose we have $x(m)$ of T observations, so this likelihood can be obtained as the density product assessed at T points. The likelihood of matrix W is given by:

$$L[W] = p_x[x(m)|w_i] = |\det W| \prod_{m=1}^{T} \prod_{i=1}^{N} p_s[w_i^T x(m)] \qquad (17)$$

Very often, the use of the logarithm of likelihood is more practical, as it is algebraically simpler. This makes no difference here since the logarithm maximum is found at the same point as the maximum likelihood. Thus the log likelihood function regarding the parameter W is:

$$\log L[W] = T \log|\det W| + \sum_{m=1}^{T} \sum_{i=1}^{N} \log\{p_i[w_i^T x(m)]\} \qquad (18)$$

Simplifying the notation and dividing by T to the likelihood, to get the equation as:

$$\frac{1}{T} \log L[W] = \log|\det W| + E\left\{\sum_{i=1}^{N} \log\{p_i[w_i^T x(m)]\}\right\} \qquad (19)$$

The log probability here is the function of the separation matrix W and the marginal density of the estimated sources. The estimation of estimated source densities is a non-parametric problem. The non-Gaussianity is used for non-parametric problem solving.

4.   Different ICA Methods

4.1 FastICA

FastICA algorithm was first introduced by Hyvarien *et al.* [19]. FastICA is a fixed-point iterative algorithm to maximize non-Gaussianity, which is an alternative to gradient-based ways that illustrates rapid (cubic) convergence.

The approach is used to optimize various forms of contrast functions like kurtosis/negentropy. Unlike gradient-based methods, the Fast-ICA method lacks the learning rate or other personalized parameters. It is a major advantage as a poor learning rate choice destroys convergence in general.

The Hyvarinen algorithm quickly converges as it seeks component one by one. For independent component estimation, FastICA uses kurtosis [19]. Whitening is generally done on the data before the algorithm is executed. This ensures that all correlation inside the data is eliminated, i.e. the data has to be uncorrelated.

The information-theoretical amount of entropy, which is the base of negentropy is robust but computationally complicated than kurtosis.

Nevertheless, computationally simple negentropy approximations are available to relieve the complexity of negentropy computation.

The followings are two different algorithms to perform FastICA.

4.2 INFOMAX

This approach maximizes the entropy of a nonlinear output (information flow) of neural network and is called as InfoMax [18]. Infomax specializes in locating ICs by optimizing member joint entropy.

Bell and colleagues tried to formulate a method, which was based on the Linsker's Infomax principle [27] to create unsupervised neural network learning rules and this was successful in solving the Blind Source separation (BSS) problem.

Table. 1. Algorithm FastICA (single independent source component)

1. Data center to zero the mean and so whiten the outcome of giving x.
2. Select a primary version of the p-vector w with unit norm.
3. Consider G, be any of these non-quadratic density with partial derivatives g (first) and g' (second).
4. Let $w^+ \leftarrow E\left(x\,g((w^+)^T x)\right) - w^+ E(g'((w^+)^T x))$. In practice, expectations are estimated by using sample averages.
5. Let $w^+ \leftarrow \dfrac{w^+}{\|w^+\|}$.
6. Iterate among steps 4 and 5. End once convergence has been reached.

Table. 2. Two FastICA algorithms (extracting multiple independent source components)

Deflation algorithm

1. Data center to zero the mean and so whiten the outcome of giving x.
2. Choose a number,m, independent components to be extracted.
3. For $l = 1,2,..., m$:

    − Initialize (e.g., randomly) the p-vector W_l to have unit norm.
    − Let $w_l^+ \leftarrow E\left(x\,g((w_l^+)^T x)\right) - w_l E\left(g'((w_l^+)^T x)\right)$ be a single-component update to FastICA for w_l^+, where g and g' defined as earlier. In practice, expectations are estimated by using sample averages.
    − Use Gram-Schmidt to orthogonalize w_l^+ with regard to previously chosen $W_1^+,...,W_{l-1}^+$:

    $$w_l^+ \leftarrow w_l^+ - \sum_{j=1}^{l-1} ((w_l^+)^T w_j^+)w_j^+$$

    − Let $w_l^+ \leftarrow \dfrac{w_l^+}{\|w_l^+\|}$.

    − Iterate w_l^+ until convergence.

4. Set $l \leftarrow l + 1$. If $l \leq m$, return to step 3.

Parallel algorithm

1. Data center to zero the mean and so whiten the outcome of giving x.
2. Choose a number,m, independent components to be extracted.
3. Initialize (e.g., randomly) the p-vector $w_1^+,...,w_n^+$, every of them must have unit norm.
a. let $W = (w_1^+,...,w_n^+)^T$.
4. Conduct a symmetric orthogonalization of W by $W \leftarrow (WW^T)^{-1/2} W$.
5. For each $l = 1,2,...,n$, let $w_l^+ \leftarrow E\left(xg((w_l^+)^T x)\right) - w_l^+ E\left(g'((w_l^+)^T x)\right)$ be the FastICA single
6. Component update for w_l^+, where g and g' are defined earlier. In practice, expectations estimated by using sample average estimation.
7. Conduct another symmetric W orthogonalazation.
8. If there is to be convergence, return to step 5.

The non-linearities in the transform function can take input distribution higher-order moments and reduce redundancy. This helps the neural network identify components, which are statistically independent in the data input. This method is also shown to be equivalent to the

Table. 3. Infomax algorithm

1	Initialize $W^*(0)$ (e.g. random),
2	$W^*(t+1) = W^*(t) + \eta(t)(I - f(Y)Y^T)W^*(t)$,
3	If not converged, go back to step 2.

Table. 4. Jade algorithm

1	From sample covariance R_x and calculate a whitening matrix W^+.
2	From the sample 4th –order cumulant Q_z of the whitened process $z(m) = W^+ x(m)$; calculate the n most significant Eigen pairs $\{\lambda_r \| 1 \le r \le n\}$.
3	With a unitary matrix U, Jointly diagonalize the set $N = \{\lambda_r M_r \| 1 \le r \le n\}$.
4	$A^+ = W^+ U$ is the estimation of A.

Table 5. Kernel ICA-KGV algorithm

Input: Data Vectors $y^1, y^2,..., y^N$

Kernel $K(x, y)$

1 Whiten the data
2 Minimize the contrast function $C(W)$ (Regarding W) defined as:

a. Compute the centered Gram matrices $K_1, K_2,..., K_m$ of the estimated sources $\{x^1, x^2,..., x^N\}$, where $x^i = Wy^i$

b. Define

$\hat{\delta}_F^k(K_1,..., K_m) = \det \mathsf{K}_k / \det D_k$

c. Define

$C(W) = \hat{I}_{\delta F}(K_1,..., K_m) = -\frac{1}{2}\log \hat{\delta}_F^k(K_1,..., K_m)$

Output: W

Whereas $\hat{\delta}_F^k(K_1,..., K_m)$ the kernel generalized variance, $\hat{I}_{\delta F}(K_1,..., K_m)$ contrast function, D stands for block-diagonal matrix of covariance of the individual vectors and $(\mathsf{K}_k)_{ij} = K_i K_j$.

methods of maximum likelihood [15]. Amari et al. (1996) proposed the algorithm as follows to calculate the unmixing matrix W (called Infomax) [28].

Being $\eta(t)$ a learning-rate function and $f(\cdot)$ a function related to the distribution nature (i.e., super Gaussian or sub Gaussian). It is important to bear in mind that a W^* initial value is a random matrix usually [22]. For more detail on Infomax's procedure see [22, 28].

4.3 JADE

The Joint Approximated Diagonalization of Eigen matrices (JADE) [20, 29] is a joint diagonalization method of the cumulant matrices, especially with regard to signal treatments for application in chemometrics [30]. Cumulants orders two and four are involved, and Joint diagonalization is carried out with Jacobi technique. The JADE algorithm also has no customizable parameters, and hence it is robust. However, this approach is very computationally intensive, since all cumulant matrices are diagonalized at once.

This algorithm works well in small dimension but is poor in high dimensional spaces. The Matrix X is first converted to a reduced set of PCA loadings, and then these are centered and whitened to equal variances. The auto-and cross-cumulants of these loadings are then put to a dimension tensor of fourth order $n \times n \times n \times n$ (is the load count).

The tensor is projected into orthogonal Eigen matrices and is diagonalized into a rotation matrix (using Jacobi algorithm). In the pre-processing stage, the rotational matrix is applied to the whitening matrix. Providing the computation of matrix W.

Equations (2, 1) gives respectively independent components and mixing matrix. The description of algorithm could be found from [29, 30].

4.4 KERNEL ICA

Kernel ICA [31], which is a non-parametric approach, works by setting a contrast function to replicate kernel Hilbert space. Contrast function may be selected either as a canonical correlation (KCC) or as a generalized variance (KGV). Here mixtures are designed for a higher dimensional space, and then the mixing matrix W is obtained to minimize pair-wise correlations in that space. If it does, this could be seen that reproducing kernel Hilbert spaces based on Gaussian kernels guarantees that the source is independent. In

Table 6. RADICAL, two-dimensional method Algorithm

Input:	Data vectors X^1, X^2,..., X^N, assumed whitened.
Parameters:	m: Size of spacing equivalent to $\sqrt{N}$.
	σ_r^2: Noise variance for replicated points.
	R: Number of replicated points per original data point.
	K: Number of angles from which to determine cost function.
Procedure:	1. Create X' by replicating R points with Gaussian noise for each original point.
	2. For each θ, rotate the data to this angle $(Y = W(\theta) * X')$ and evaluate cost function.
	3. Output W correspond to the optimal θ.
Output:	W (demixing matrix).

All parameter and notation are taken from (Learned-Miller and Fisher III (2003) [32])

addition, this approach is argued to be more stable than previous ICA algorithms as for the existence of outliers.

Bach and Jordan presented the algorithm Kernel ICA-KGV [31] as follows.

4.5 RADICAL

RADICAL (Robust, Accurate and Direct ICA algorithm) is an effective entropy estimator based ICA algorithm [32].

The ICA approach is based on a direct minimization of the measurement of the departure from independence by estimated divergence of Kullback-Leibler between the joint distribution and the marginal distribution product. RADICAL's entropy estimator is a function of the order statistics.

The entropy estimator used in particular is consistent, and shows rapid convergence. This entropy estimate is reliable, fast converging in computational efficiency and pretends to be robust to outliers.

The RADICAL algorithm described as follows, was proposed by E.G. Learned-Miller and J. W. Fisher III [32].

4.6 ICA with PSO

In recent times, Particle Swarm optimization (PSO) is a familiar population based-search. Particle swarm optimization (PSO) is used to detect the search space of a particular problem to find the settings or parameters required to maximize/minimize a specific objective. The algorithm works by maintaining a few candidate solutions at the same time in the search space. PSO algorithm works in 3 steps; first it estimates the fitness of each moving particle, secondly, it updates individual and global best fitness and position, and finally, it updates the velocity and position of each particle.

There has been much work to solve Independent Component Analysis with Particle Swarm Optimization. One of the Algorithms was presented [33] for optimizing the objective function:

$$J(y) = \frac{1}{\mu}\sum_\alpha (C_{\alpha\alpha\alpha\alpha}^{(y)})^2 + \frac{1}{\tau}\sum_\alpha (C_{\alpha\alpha\alpha\alpha}^{(y)})^2, \qquad (14)$$

where the auto cumulant in fourth order is given by:

$$C_{ijkl}^{(y)} = <y_i y_j y_k y_l> - <y_i y_j><y_k y_l> \\ - <y_i y_k><y_j y_l><y_i y_l><y_j y_k> \qquad (15)$$

The objective function J is kurtosis [34, 35], so it can be written as a function of an orthogonal matrix U to be determined by the method of optimization. It is not directly easy to work with this kurtosis objective function, so later on this objective function is modified with the help of reference vector. Therefore, a reference-based contrast function is defined. Reference signals are merely signals artificially introduced to facilitate the maximization of contrast function.

Since reference signals are indirectly involved in the process of iterative optimization, these reference-based contrast functions have a common appealing feature and the respective optimization algorithms are quadratic regarding the parameters searched:

$$C_z\{y\} \triangleq Cum\{y, y, z, z\} = E\{y^2 z^2\} - E\{y^2\}E\{z^2\} - 2E^2\{yz\} \quad (16)$$

where $E\{\cdot\}$ denotes the expectation value and z is the reference signal. Considering another

Table. 7. Two ICA with PSO algorithms

Gradient- based PSO

Input : $x(m)$: Observed signal, S: swarm size and δ: Trade-off parameter

Output: U^{best} : Separation vector

Initialize U^0 and the corresponding reference signal $z_0^p(m) = U_0^p x(m), \forall 1 \leq p \leq S$.

    for $k = 0,1, \dots, k_{max} - 1$ do

      $I_p = I\left(U_k^p, U_k^p\right), \forall\, 1 \leq p \leq S$

      $best = \underset{p}{\arg\max}\ I_p$

      $d_k^p = \nabla I(U_k^p, U_k^p)$

      $\alpha^p = \underset{\alpha}{\arg\max}\ I(U_k^p + \alpha d_k^p, U_k^p)$

      $\tilde{U}_{k+1}^p \leftarrow U_k^p + \alpha^p(\delta d_k^p + (1-\delta)(U^{best} - U_k^p))$

      $\tilde{U}_{k+1}^p \leftarrow \dfrac{\tilde{U}_{k+1}^p}{(E\{|\tilde{U}_{k+1}^p x(n)|^2\})^{\frac{1}{2}}}$

      $U_{k+1}^p \leftarrow \tilde{U}_{k+1}^p$

    end

Gradient-based PSO with Fixed-point update

Input : $x(m)$: Observed signal, S: swarm size and δ: Trade-off parameter

Output: U^{best}: Separation vector

Initialize U^0 and the corresponding reference signal $z_0^p(m) = U_0^p x(m), \forall 1 \leq p \leq S$.

    for $k = 0,1, \dots, k_{max} - 1$ do

    $\tilde{U}_0^p = U_k^p, \forall\, 1 \leq p \leq S$

    for $l = 0,1, \dots, l_{max} - 1$ do

      $I_p = I\left(\tilde{U}_l^P, U_k^p\right), \forall\, 1 \leq p \leq S$

      $best = \underset{p}{\arg\max}\ I_p$

      $\tilde{d}_k^p = \nabla_1 I(\tilde{U}_l^P, U_k^p)$

      $\tilde{\alpha}^p = \underset{\alpha}{\arg\max}\ I(\tilde{U}_l^p + \alpha \tilde{d}_k^p, U_k^p)$

      $\tilde{U}_{l+1}^p \leftarrow \tilde{U}_l^p + \alpha^p(\delta \tilde{d}_k^p + (1-\delta)(U^{best} - \tilde{U}_l^p))$

      $\tilde{U}_{l+1}^p \leftarrow \dfrac{\tilde{U}_{l+1}^p}{(E\{|\tilde{U}_{l+1}^p x(n)|^2\})^{\frac{1}{2}}}$

    end

    $U_{k+1}^p \leftarrow \tilde{U}_{l_{max}}^p$

    end

(reference) separation matrix V and $z(m) = Vx(m)$, now the contrasts function specifically in terms W and V as follows:

$$I(W,V) = \left| \frac{C_z\{y\}}{E\{y^2\}E\{z^2\}} \right|^2, \qquad (17)$$

where $y(m) = Wx(m)$ and $z(m) = Vx(m)$.

Some earlier suggestions have been made using PSO to solve ICA problem [36, 37]. The method of combining swarm-search with gradient-based optimization for ICA is different from PSO algorithm proposed in [33], where the particle velocity component is modified with gradient direction at each iteration, and the direction of global best. The two algorithms presented by Pati et al., are presented below [33].

5. Contrast Function for ICA

The model of the data estimation in independent component analysis is generally carried out by formulating an objective function and then minimizing or maximizing the function. The objective function is called as contrast function. Many researchers use the terms, 'loss function or cost function' in their researches. In simpler terms, this can be interpreted as any function whose optimization makes it possible to estimate independent components.

Some of the classical optimization methods can be used for explicitly formulated objective function to optimize the objective function. Some such methods are gradient method, Newton methods, Iterative method etc. For certain cases however, the theory of algorithm and estimation is difficult.

The initial phase of BSS comparison works focused on Shannon entropy and Kullback-Leibler divergence (KLD) based on theoretical definitions of information independence and its approximations through statistics of higher order. The other important group of contrasts came from the non-Gaussian definitions and their approximations of independence [15]. You will find more information on these commonly used, conventional contrast functions in [12, 20].

5.1 General Contrast Functions

This is a one-unit contrast function developed [38] that has statistically attractive properties (contrast to cumulant) without prior understanding of the

densities of the independent components which is required to allow simple algorithmic implementation to make it simple. This so-called one-unit contrast function as optimization makes it possible to estimate a single independent component rather than estimate the entire ICA model. A family of such non-normality measures could be virtually built using any function G and taking into account the gap between G's expectations regarding actual data and Gaussian Data Expectations. Put another way, a contrast function J can be defined, which measures the non-normality of a zero-mean random variable using any case, a non-quadratic, sufficiently smooth function G as follows:

$$J_G(y) = \left| E_y\{G(y)\} - E_v\{G(v)\} \right|^p, \qquad (18)$$

where v is a standardized Gaussian random variable, y is supposed to be normalized to the unit variance, and the exponent $p = 1,2$ usually. The subscripts indicates expectation regarding y and v. (The J_G notation not to be confused with the notion of negentropy, J.)

Clearly, J_G can be regarded a generalization of (the modulus of) kurtosis. For $G(y) = y^4$, J_G becomes simply the modulus of kurtosis of y. Note, G is not be quadratic, because J_G or all distribution would then be trivially zero. So, apparently J_G could be a contrast function just like kurtosis. The J_G is really a contrast functions in an appropriate sense (locally).

In [39], the estimators' finite-sample statistical properties were evaluated based on the optimization of such a general contrast function. It was found that for an acceptable choice of G, the estimator's statistical characteristics (asymptotic variance and robustness) are significantly better than those of the cumulant-based estimators. Varieties of G were suggested below:

$$G_1(u) = \log \cosh a_1 u, \ G_2(u) = \exp(-a_2 u^2 / 2),$$

where $a_1, a_2 \geq 1$ are certain adequate constants. Without the detail information about the distribution of independent components or outliers, both of these functions are the optimal contrast function, which seem to approximate fairly well in most cases. It was experimentally observed that the values in particular $1 \leq a_1 \leq 2$, $a_2 = 1$ bring good constant approximations. One explanation for this is that G_1 above matches to the log-density of a super-Gaussian distribution and is therefore closely connected to the estimation of maximum likelihood.

Since the BSS problem the linear combination of the observed mixtures is discussed $x(m)_j$, say $w^T x(m)$, where the weight vector w is constrained so that $E\{(w^T x(m))^2\} = 1$. So the algorithms are extreme based on the kurtosis square $K^2(w^T x(m)) = (E\{(w^T x(m))^4\} - 3)^2$ of such linear combinations [7, 23]. The kurtosis square could be presented as approximately to the negentropy of $w^T x(m)$. One may see that the kurtosis square of $w^T x(m)$ is maximized precisely where the linear combination is equal to, up to the sign, one of the ICs i.e., $w^T x(m) = \pm s_i$.

This can be used to create a contrast function instead of kurtosis, basically on any quadratic well behaving even function, say G. Such contrast function may generally be described as:

$$J_G(w) = [E_x\{G(w^T x(m))\} - E_v\{G(v)\}]^2, \qquad (19)$$

where v is a standardized Gaussian variable and J_G can be considered as a generalization of the kurtosis square, as for $G(u) = u^4$, J_G turn into simple kurtosis of $w^T x(m)$.

J_G is locally maximized when $w^T y(n) = \pm s_i$.

Thus J_G can be used as a contrast function just like the kurtosis square. Widely used one unit contrast functions are:

Skew: $g(x) = x^2$
Pow3: $g(x) = x^3$
$\quad\quad\quad g(x) = x^4/4, \quad x^2/2$
Gauss: $\quad g(x) = x \exp\left(-\frac{x^2}{2}\right), \ -\exp\left(-\frac{x^2}{2}\right),$
Tanh: $\quad g(x) = \tan h(x), \quad logcosh(x).$

The key advantage of the FastICA algorithm is its speed (superior to gradient based schemes), user-friendly (needs nonprobability distribution or collection of other parameters) and its flexibility for performance optimization by selection of the contrast function $G(x)$ or equivalently $g(x) = G'(x)$.

5.2 Contrast Function without Permutation Ambiguity

A linear combination of the fourth order marginal cumulants (kurtosis) for the separator output is a true contrast function for ICA under the pre-whiting assumption if the weights show the same sign of kurtosis as the source.

If the weights are equal to the source kurtosis then the contrast function is a cumulant criterion based on the principle of maximum likelihood.

If the source kurtosis is different from the linear weight combination (even not matched from the former) then the contrast eliminates the ambiguity of the permutation, since at the separator output the estimated source is shortened according to its kurtosis values in the same order as the weights. For more details, see [40].

5.3 Non-differentiable Contrast Functions

For ICA, the contrast function can be global (multi-unit) or component wise (single unit). When we speak of multi-unit, the function $C(y)$ summarizes the level of independence between all pairs of components in one scalar value. In single unit contrast function measures a quantity associated with the ith component of y which is typically higher for independence signals than for mixtures of signals.

The multi-unit algorithm is called symmetric approach (all source is extracted simultaneously) whereas the single-unit algorithm is called a deflation approach (source is extracted one after another). For the component wise contrast function as $C(y, i)$ is written as $C(w_i z)$, where w_i is the ith row of W knowing that w_i is orthogonal to any other row of w_j. Positive and negative angular variations of w_i that preserve unit norm defined and note as:

$$w_{i\uparrow j} = \cos(\alpha)\, w_i + \text{sing}(\alpha)\, w_j,$$

$$w_{i\downarrow j} = \cos(\alpha)\, w_i - \text{sing}(\alpha)\, w_j.$$

The relevant contrast values may be written as $C(w_{i\uparrow j}z)$ and $C(w_{i\downarrow j}z)$.

The maximization of the contrast function here is dependent on the assumptions:

1. The contrast function in relation to α should be continuous or at least almost continuous.

2. All maxima of the contrast function is differentiable with respect to α.

More information can be found in [41] on the Non-differentiability Contrast function.

5.4 Quadratic Contrast Function

The most enticing approach to the issue of blind equalization is the use of a suitable contrast function. A contrast function basically plays the role of an objective function in the sense that its (global) maximization makes it possible to solve problems. In [42] a contrast function for the i.i.d. source signals is defined as:

Definition 1:

Let $C(.)$ be a real function of the signal:

$$y(n) = \sum_{k \in Z} g(n-k) s(k) = \{g\} s(n),$$

where $g(n) = \sum_{k \in Z} w(k) M(n-k)$. Is called a contrast function when there exists $i_0 \in \{1,..., N\}$ such that:

P1. $\exists l \in Z$ such that for all possible output $y(n)$ of the equalizer:

$$C(y(n)) \le C(s_{i_0}(n-1)).$$

P2. If equality holds in the equation P1, then $g \in G_{1ed}^{i_0}$.

Definition 1 cannot be used for non i.i.d. source signals since the independence property for these signals only leads to one source being extracted up to a scale filter. Therefore, a generalization of definition 1 is required for non i.i.d. source signals.

Definition 2:

The real function $C(.)$ is called a Contrast Function when there exists $i_0 \in \{1,..., N\}$ such that:

P1. For all possible equalizer output:

$$C\big(\mathrm{y}(n)\big) \le \sup_{g \in G_{1e}^{i_0}} C\big(\{\mathrm{g}\}\mathrm{s}(n)\big).$$

P2. If equality holds in the equation P2, then $g \in G_{1ed}^{i_0}$.

5.5 Reference Based Contrast Function

A Singular Value Decomposition (SVD) based maximization algorithm is substantially faster than other maximization algorithms.

However, because of its sensitivity to a rank estimation, the method frequently suffers from the need to know the filter orders well. A method of gradient optimization with reference signals based on Kurtosis gets an optimal step size and requires no estimation of the level.

Thus, the SVD-based methods' drawback can be managed well. During the optimization process the reference signals involved in this method are fixed, which can result in poor separation output due to inappropriate initialization value of the corresponding reference signals. We usually inject reference signals artificially into an algorithm so that the contrast function can be maximized.

Since we consider linear separator, whose output is defined as: $y(m) = Wx(m)$, where W is the separation matrix of $n \times m$. $y(m)$ is the approximate estimation of $s(n)$. The "parameters searched" here are the W vectors in the row. with the obvious assumtion of independence in the source and the general defination of $Cum\{\cdot\}$. Signal may be apprised in real-valued or complex valued. Considering real valued signals for any jointly stationary signals $y(m)$ and $z(m)$, let:

$$\begin{aligned}C\{y\} &\triangleq Cum\{y(m), y(m), y(m), y(m)\}\\ &= E\{y(m)^4\}\\ &\quad - 3E\{y(m)^2\}^2,\end{aligned} \tag{20}$$

$$\begin{aligned}C_z\{y\} &\triangleq Cum\{y(m), y(m), z(m), z(m)\}\\ &\{y(m)^2 z(m)^2\} - E\{y(m)^2\}E\{z(m)^2\} -\\ &\quad 2E^2\{y(m)z(m)\},\end{aligned} \tag{21}$$

where $E\{\cdot\}$ denotes the expection value.

Introducing a "reference signals", one can consider similar like $y(m) = Wx(m)$ a

separating matrix of $n \times m$ denoted by V. The respective output can be denoted as:

$$z(m) = Vx(m), \tag{22}$$

where $z(m)$ components *are* the reference signals. The reference signals directly influence the reference-based contrast functions and their values affect the results on optimization, in particular the value of initialization.

With the criteria:

$$J(\mathrm{w}) = \left|\frac{C\{y(m)\}}{E\{(y(m))^2\}^2}\right|^2, \tag{23}$$

$$I(\mathrm{w}, \mathrm{v}) = \left|\frac{C_z\{y(m)\}}{E\{(y(m))^2\}E((z(m))^2)}\right|^2, \tag{24}$$

where J is the well known Kurtosis contrast function. I is the reference-based contrast function.

As described [24], ∇ refers to a gradient and partial gradient operator are denoted by ∇_1 and ∇_2 with correspond to the first and second parameters, respectively. More accurately, $\nabla J(\mathrm{w})$ is the vector containing all of the partial derivatives of $J(w)$, whereas $\nabla_1 I(\mathrm{w}, \mathrm{v})$ and $\nabla_2 I(\mathrm{w}, \mathrm{v})$ are Partial Derivatives Vectors of $I(w, v)$ with respect to w and v. Combined with (21, 24):

$$\begin{aligned}I(w,v) &= \left|\frac{C_z\{y(m)\}}{E\{(y(m))^2\}E\{(z(m))^2\}}\right|^2\\ &= \left|\frac{\begin{array}{c}E\left[(wx(m))^2(vx(m))^2\right] -\\ E\left[(wx(m))^2\right]E\left[(vx(m))^2\right] - 2\{E(wx(m))(vx(m))\}^2\end{array}}{E\{(wx(m))^2\}E\{(yx(m))^2\}}\right|^2.\end{aligned} \tag{25}$$

Because $x(m)$ in prewhitened and w, v are normalized so $E\left\{(wx(m))^2\right\} = E\left\{(vx(m))^2\right\} = 1$ and $ww^T = vv^T = 1$. Then equation (24) can be reduced to:

$$I(w, v) = \left|E\left[(wx(m))^2(vx(m))^2\right] - 3w^2\right|^2. \tag{26}$$

More details on the reference-based contrast can be found at [24, 42, 43].

6. Applications of ICA in Real World

There is several success of ICA application in a number of practical problems apart from the signal processing in telecommunications [7, 26]. Now the use of ICA has been expanded to a wide variety of domains. Some of these applications includes:

- Telecommunications [44],
- Machine fault detection [45, 46],
- Feature extraction [47, 48],
- Sensor Signal Processing [49],
- Audio signal processing [13, 15],
- Image processing [50, 51, 52],
- Text mining [53, 54],
- Analyzing financial time series [55, 56,57],
- Pattern recognition [58],
- Bio medical signal processing [59, 60, 61],
- In Astrophysics [62, 63],
- Petrochemical field [64, 65].

6.1 Face Recognition by ICA

Today, data security attacks remain a top concern, and the research on reliable recognition of humans' faces has seen a major research area of Computer Science, Artificial Intelligence and Machine Learning.

Building a face recognition system to replicate the human ability of face recognition is a non-trivial task and modeling the varying uncertain and imprecise condition has posed insurmountable challenges to researchers in the past few decades.

The problem of face recognition can be expelled as follows. Given a database of face images and a query face image, the goal is to find the most similar face images from the database. By Bartlett et al. (2002) [66] a method for face recognition is proposed based on ICA. Two architectures are presented for face recognition-spatial local images for the faces and factorial face code and it is shown that both ways of recognition are superior to PCA.

7. Conclusion

This review provides basic information about ICA and its methodology in addition to the earlier surveys; a few more contrast functions and advance algorithms for ICA have been clubbed together. ICA is a general term for the wide variety of applications in the fields of neural computation, signal processing and statistics. ICA provides a systematic transformation or representation of multidimensional data for the subsequent processing of information.

The transformation helps to examine and find its own interesting ways, rules and patterns. It was clear from the discussion that ICA operates mainly with two factors called contrast function and its optimization algorithm. Reference-based contrast functions are especially attractive, as the corresponding problem of maximization is quadratic in relation to the parameters searched. Non-differentiable contrast function is useful if source is extracted one by one (deflation approach).

Maximizing the non-differentiable contrast function is based on the assumptions that the contrast function is to be continuous or at least almost continuous and all contrast function maxima are differentiable. Evolutionary computing techniques are common methods of optimization based on population searches. Genetic algorithms and swarm intelligence are the most widely applied techniques of optimization based on evolutionary computation.

Particle swarm optimization (PSO) is used in ICA technology. The ICA method currently uses various biologically inspired optimization algorithms.

References

1. **Gopinath, R.A., Ramabhadran, B., Dharanipragada, S. (2001).** Factor analysis invariant to linear transformations of data. Fifth International Conference on Spoken Language Processing.

2. **Wold, S., Esbensen, K., Geladi, P. (1987).** Principal component analysis. Chemometrics and Intelligent Laboratory Systems, Vol. 2, No. 1-3, pp. 37−52. DOI: 10.1016/0169-7439(87)80084-9.

3. **Shlens, J. (2014).** A tutorial on principal component analysis. arXiv:1404.1100.

4. **Friedman, J.H., Tukey, J.W. (1974).** A projection pursuit algorithm for exploratory data analysis. IEEE Transactions on computers, Vol. 23, No. 9, pp. 881−890. DOI: 10.1109/T-C.1974.224051.

5. **Herault, J., Jutten, C. (1986).** Space or time adaptive signal processing by neural network models. AIP American Institute of Physics, Conference Proceedings, Vol. 151, No. 1, pp. 206−211. DOI: 10.1063/1.36258.

6. **Jutten, C., Herault, J. (1991).** Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. Signal Processing, Vol. 24, No. 1, pp. 1−10. DOI: 10.1016/0165-1684(91)90079-X.

7. **Comon, P. (1994).** Independent component analysis, a new concept?. Signal processing, Vol. 36, No. 3, pp. 287−314. DOI: 10.1016/0165-1684(94)90029-9ff.

8. **Hyvärinen, A., Pajunen, P. (1999).** Nonlinear independent component analysis: Existence and uniqueness results. Neural Networks, Vol. 12, No. 3, pp. 429−439. DOI: 10.1016/S0893-6080(98)00140-3.

9. **Sprekeler, H., Zito, T., Wiskott, L. (2014).** An extension of slow feature analysis for nonlinear blind source separation. The Journal of Machine Learning Research, Vol. 15, No. 1, pp. 921−947.

10. **Zheng, C.H., Huang, Z.K., Lyu, M.R., Lok, T.M. (2006).** Nonlinear blind source separation using hybrid neural networks. International Symposium on Neural Networks, Springer, pp. 1165−1170.

11. **Al-Saegh, A. (2015).** Independent component analysis for separation of speech mixtures: a comparison among thirty algorithms. Iraqi Journal for Electrical and Electronic Engineering, Vol. 11, No. 1, pp. 1−9. DOI: 10.37917/ijeee.11.1.1.

12. **Hyvärinen, A. (1999**). Survey on independent component analysis. Neural Computing Surveys, Vol. 2, No. 4, pp. 94−128.

13. **Bell, A.J., Sejnowski, T.J. (1997).** The "independent components" of natural scenes are edge filters. Vision Research, Vol. 37, No. 23, pp. 3327−3338. DOI: 10.1016/ S0042-6989(97)00121-1.

14. **Hyvärinen, A. (2013).** Independent component analysis: recent advances. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, Vol. 371, No. 1984. 20110534. DOI: 10.1098/rsta.2011.0534.

15. **Hyvärinen, A., Karhunen, J., Oja, E. (2001).** Independent component analysis. John Wiley & Sons.

16. **Jutten, C., Karhunen, J. (2004).** Advances in blind source separation (BSS) and independent component analysis (ICA) for nonlinear mixtures. International journal of neural systems, Vol. 14, No. 5, pp. 267−292. DOI: 10.1142/S012906570400208X.

17. **Hyvarinen, A. (1999).** Fast and robust fixed-point algorithms for independent component analysis. IEEE transactions on Neural Networks, Vol. 10, No. 3, pp. 626−634. DOI: 10.1109/72.761722.

18. **Bell, A.J., Sejnowski, T.J. (1995).** An information-maximization approach to blind separation and blind deconvolution. Neural Computation, Vol. 7, No. 6, pp. 1129−1159. DOI: 10.1162/neco.1995.7.6.1129.

19. **Hyvärinen, A., Oja, E. (1997).** A fast fixed-point algorithm for independent component analysis. Neural Computation, Vol. 9, No. 7, pp. 1483−1492. DOI: 10.1162/neco.1997.9. 1483.

20. **Cardoso, J.F. (1999).** High-order contrasts for independent component analysis. Neural computation, Vol. 11, No. 1, pp. 157−192. DOI: 10.1162/089976699300016863.

21. **Zarzoso, V., Comon, P. (2009).** Robust independent component analysis by iterative maximization of the kurtosis contrast with algebraic optimal step size. IEEE Transactions on neural networks, Vol. 21, No. 2, pp. 248−261. DOI: 10.1109/TNN. 2009.2035920.

22. **Langlois, D., Chartier, S., Gosselin, D. (2010).** An Introduction to Independent Component Analysis: InfoMax and FastICA algorithms. Tutorials in Quantitative Methods for Psychology, Vol. 6, No. 1, pp. 31–38. DOI: 10.20982/tqmp.06.1.p031.

23. **Delfosse, N., Loubaton, P. (1995).** Adaptive blind separation of independent sources: a deflation approach. Signal Processing, Vol. 45, No. 1, pp. 59–83. DOI: 10.1016/0165-1684(95)00042-C.

24. **Castella, M., Moreau, E. (2011).** New kurtosis optimization schemes for MISO equalization. IEEE Transactions on Signal Processing, Vol. 60, No. 3, pp. 1319–1330. DOI: 10.1109/TSP.2011.2177828.

25. **Ruckebusch, C. (2016).** Resolving spectral mixtures: with applications from ultrafast time-resolved spectroscopy to super-resolution imaging. Elsevier.

26. **Hyvärinen, A., Oja, E. (2000).** Independent component analysis: algorithms and applications. Neural networks, Vol. 13, No. 4-5, pp. 411−430. DOI: 10.1016/S0893-6080(00)00026-5.

27. **Linsker, R. (1992).** Local synaptic learning rules suffice to maximize mutual information in a linear network. Neural Computation, Vol. 4, No. 5, pp. 691−702. DOI: 10.1162/neco. 1992.4.5.691.

28. **Amari, S., Cichocki, A., Yang, H.H. (1995).** A new learning algorithm for blind signal separation. Advances in Neural Information Processing Systems, Vol. 8, pp. 757−763.

29. **Cardoso, J.F., Souloumiac, A. (1993).** Blind beamforming for non-Gaussian signals. IEE Proceedings Radar and Signal Processing, IET Digital Library, Vol. 140, No. 6, pp. 362−370. DOI: 10.1049/ip-f-2.1993. 0054.

30. **Rutledge, D.N., Bouveresse, D.J.R. (2013).** Independent components analysis with the JADE algorithm. TrAC Trends in Analytical Chemistry, Vol. 50, pp. 22−32. DOI: 10.1016/j.trac.2013.03.013.

31. **Bach, F.R., Jordan, M.I. (2002).** Kernel independent component analysis. Journal of machine learning research, Vol. 3, pp. 1−48. DOI: 10.1162/153244303768966085.

32. **Learned-Miller, E.G. (2003).** ICA using spacings estimates of entropy. Journal of Machine Learning Research, Vol. 4, pp. 1271−1295.

33. **Pati, R., Kumar, V., Pujari, A.K. (2019).** Gradient-based swarm optimization for ICA. Progress in Advanced Computing and Intelligent Engineering, pp. 225−235. DOI: 10.1007/978-981-13-1708-8_21.

34. **Simon, C., Loubaton, P., Jutten, C. (2001).** Separation of a class of convolutive mixtures: a contrast function approach. Acoustics, Speech, and Signal Processing, ICASSP´88, International Conference on, Vol. 81, No. 4, pp. 883−887. DOI: 10.1109/ICASSP.1999.756250.

35. **Tugnait, J.K. (1997).** Identification and deconvolution of multichannel linear non-Gaussian processes using higher order statistics and inverse filter criteria. IEEE Transactions on Signal Processing, Vol. 45, No. 3, pp. 658−672. DOI: 10.1109/78. 558482.

36. **Igual, J., Ababneh, J., Llinares, R., Miro-Borras, J., Zarzoso, V. (2010).** Solving independent component analysis contrast functions with particle swarm optimization. International Conference on Artificial Neural Networks, Springer, pp. 519−524. DOI: 10.1007/ 978-3-642-15822-3_63.

37. **Li, H., Li, Z., Li, H. (2016).** A blind source separation algorithm based on dynamic niching particle swarm optimization. EDP Sciences. MATEC Web of Conferences, Vol. 61. DOI: 10.1051/matecconf/20166103008.

38. **Hyvarinen, A. (1997).** A family of fixed-point algorithms for independent component analysis. IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 5, pp. 3917−3920. DOI: 10.1109/ICASSP.1997. 604766.

39. **Hyvarinen, A. (1997).** One-unit contrast functions for independent component analysis: A statistical analysis. Neural Networks for Signal Processing VII. Proceedings of the 1997 IEEE Signal Processing Society Workshop, pp. 388−397. DOI: 10.1109/NNSP.1997.622420.

40. **Zarzoso, V., Comon, P., Phlypo, R. (2010).** A contrast function for independent component analysis without permutation ambiguity. IEEE Transactions on Neural Networks, Vol. 21, No. 5, pp. 863−868. DOI: 10.1109/TNN.2010.2045128.

41. **Lee, J.A., Vrins, F., Verleysen, M. (2005).** A simple ICA algorithm for non-differentiable contrasts. 13th European Signal Processing Conference, pp. 1-4. DOI: 10.5281/zenodo. 39058.

42. **Castella, M., Moreau, E., Pesquet, J.C. (2004).** A quadratic MISO contrast function for blind equalization. IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 2, pp. 681. DOI: 10.1109/ICASSP.2004.1326349.

43. **Zhao, W., Wei, Y., Shen, Y., Cao, Y., Yuan, Z., Xu, P., Jian, W. (2015).** An efficient algorithm by kurtosis maximization in reference-based framework. Radio- engineering, Vol. 24, No. 2, pp. 544−551. DOI: 10.13164/re.2015.0544.

44. **Ristaniemi, T., Joutsensalo, J. (1999).** Independent component analysis with code information utilization in DS-CDMA signal separation. Seamless Interconnection for Universal Services, Global Telecommunications Conference, GLOBECOM'99, Vol. 1, pp. 320−324. DOI: 10.1109/GLOCOM.1999. 831657.

45. **Li, Z., He, Y., Chu, F., Han, J., Hao, W. (2006).** Fault recognition method for speed-up and speed-down process of rotating machinery based on independent component analysis and Factorial Hidden Markov Model. Journal of Sound and Vibration, Vol. 291, No. 1-2, pp. 60−71. DOI: 10.1016/j.jsv.2005.05.020.

46. **Zhonghai, L., Yan, Z., Liying, J., Xiaoguang, Q. (2009).** Application of independent component analysis to the aero-engine fault diagnosis.

Chinese Control and Decision Conference, pp. 5330−5333. DOI: 10.1109/CCDC.2009.5195066.

47. **Kwak, N., Choi, C.H., Ahuja, N. (2002).** Face recognition using feature extraction based on independent component analysis. Proceedings. International Conference on Image Processing, Vol. 2, pp. II–II. DOI: 10.1109/ICIP.2002.1039956.

48. **Kwak, N., Choi, C.H. (2003).** Feature extraction based on ICA for binary classification problems. IEEE Transactions on Knowledge and Data Engineering, Vol. 15, No. 6, pp. 1374−1388. DOI: 10.1109/ TKDE.2003.1245279.

49. **Cvejic, N., Bull, D., Canagarajah, N. (2007).** Multimodal image fusion in sensor networks using independent component analysis. 15th International Conference on Digital Signal Processing, pp. 260−263. DOI: 10.1109/ICDSP. 2007.4288568.

50. **Fiori, S. (2003).** Overview of independent component analysis technique with an application to synthetic aperture radar (SAR) imagery processing. Neural Networks, Vol. 16, No. 3-4, pp. 453−467. DOI: 10.1016/S0893-6080(03) 00016-9.

51. **Wang, H., Pi, Y., Liu, G., Chen, H. (2008).** Applications of ICA for the enhancement and classification of polarimetric SAR images. International Journal of Remote Sensing, Vol. 29, No. 6, pp. 1649−1663. DOI: 10.1080/01431160701395211.

52. **Karoui, M.S., Deville, Y., Hosseini, S., Ouamri, A., Ducrot, D. (2009).** Improvement of remote sensing multispectral image classification by using independent component analysis. First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, pp. 1−4. DOI: 10.1109/WHISPERS.2009.5289033.

53. **Bingham, E. (2001).** Topic identification in dynamical text by extracting minimum complexity time components. Proc. ICA´01, pp. 546-551.

54. **Yokoi, T., Yanagimoto, H., Omatu, S. (2009).** Information Filtering using Index Word Selection based on the Topics. International Journal of Computer and Information Engineering, Vol. 3, No. 2, pp. 276−282. DOI: 10.1007/978-3-540-30213-1_15.

55. **Baragona, R., Battaglia, F. (2007).** Outliers detection in multivariate time series by independent component analysis. Neural computation, Vol. 19, No. 7, pp. 1962−1984. DOI: 10.1162/neco.2007.19.7.1962.

56. **Lu, C.J., Lee, T.S., Chiu, C.C. (2009).** Financial time series forecasting using independent component analysis and support vector regression. Decision support systems, Vol. 47, No. 2, pp. 115−125. DOI: 10.1016/j.dss. 2009.02.001.

57. **Matilainen, M., Nordhausen, K., Oja, H. (2015).** New independent component analysis tools for time series. Statistics & Probability Letters, Vol. 105, pp. 80−87. DOI: 10.1016/j.spl.2015.04.033.

58. **Yang, J., Gao, X., Zhang, D., Yang, J.Y. (2005).** Kernel ICA: An alternative formulation and its application to face recognition. Pattern Recognition, Vol. 38, No. 10, pp. 1784−1787. DOI: 10.1016/j. patcog.2005.01.023.

59. **Safavi, H., Correa, N., Xiong, W., Roy, A., Adali, T., Korostyshevskiy, V.R., Seillier-Moiseiwitsch, F. (2008).** Independent component analysis of 2-D electrophoresis gels. Electrophoresis, Vol. 29, No. 19, pp. 4017−4026. DOI: 10.1002/elps.200800028.

60. **Llinares, R., Igual, J. (2009).** Application of constrained independent component analysis algorithms in electrocardiogram arrhythmias. Artificial Intelligence in Medicine, Vol. 47, No. 2, pp. 121−133.

61. **Rangayyan, R.M. (2015).** Biomedical signal analysis. John Wiley & Sons.

62. **Funaro, M., Oja, E., Valpola, H. (2001).** Artefact detection in astrophysical image data using independent component analysis. Proceedings of 3rd Int. Conf. on Independent Component Analysis and Signal Separation ICA´01, pp. 43−48.

63. **Maino, D., Farusi, A., Baccigalupi, C., Perrotta, F., Banday, A.J., Bedini, L., Salerno, E. (2002).** All-sky astrophysical component separation with Fast Independent Component Analysis FASTICA. Monthly Notices of the Royal Astronomical Society, Vol. 334, No. 1, pp. 53−68. DOI: 10.1046/j.1365-8711.2002. 05425.x.

64. **Pasadakis, N., Kardamakis, A.A. (2006).** Identifying constituents in commercial gasoline using Fourier transform-infrared spectroscopy and independent component analysis. Analytica Chimica Acta, Vol. 578, No. 2, pp. 250−255. DOI: 10.1016/j.aca. 2006.06.072.

65. **Kardamakis, A.A., Mouchtaris, A., Pasadakis, N. (2007).** Linear predictive spectral coding and independent component analysis in identifying gasoline constituents using infrared spectroscopy. Chemometrics and Intelligent Laboratory Systems, Vol. 89, No. 1, pp. 51−58. DOI: 10.1016/j.chemolab. 2007.05.008.

66. **Bartlett, M.S., Movellan, J.R., Sejnowski, T.J. (2002).** Face recognition by independent

component analysis. IEEE Transactions on neural networks, Vol. 13, No. 6, pp. 1450−1464. DOI: 10.1109/TNN.2002.804 287.