Survey on Independent Component Analysis

Aapo Hyvärinen
Helsinki University of Technology, Finland

Abstract

A common problem encountered in such disciplines as statistics, data analysis, signal processing, and neural network research, is finding a suitable representation of multivariate data. For computational and conceptual simplicity, such a representation is often sought as a linear transformation of the original data. Well-known linear transformation methods include, for example, principal component analysis, factor analysis, and projection pursuit. A recently developed linear transformation method is independent component analysis (ICA), in which the desired representation is the one that minimizes the statistical dependence of the components of the representation. Such a representation seems to capture the essential structure of the data in many applications. In this paper, we survey the existing theory and methods for ICA.

1 Introduction

A central problem in neural network research, as well as in statistics and signal processing, is finding a suitable representation of the data, by means of a suitable transformation. It is important for subsequent analysis of the data, whether it be pattern recognition, data compression, de-noising, visualization or anything else, that the data is represented in a manner that facilitates the analysis. As a trivial example, consider speech recognition by a human being. The task is clearly simpler if the speech is represented as audible sound, and not as a sequence of numbers on a paper.

In this paper, we shall concentrate on the problem of representing continuous-valued multidimensional variables. Let us denote by $\mathbf{x}$ an m-dimensional random variable; the problem is then to find a function $\mathbf{f}$ so that the n-dimensional transform $\mathbf{s} = (s_1, s_2, ..., s_n)^T$ defined by

$$\mathbf{s} = \mathbf{f}(\mathbf{x}) \tag{1}$$

has some desirable properties. (Note that we shall use in this paper the same notation for the random variables and their realizations: the context should make the distinction clear.) In most cases, the representation is sought as a linear transform of the observed variables, i.e.,

$$\mathbf{s} = \mathbf{W}\mathbf{x} \tag{2}$$

where $\mathbf{W}$ is a matrix to be determined. Using linear transformations makes the problem computationally and conceptually simpler, and facilitates the interpretation of the results. Thus we treat *only linear* transformations in this paper. Most of the methods described in this paper can be extended for the non-linear case. Such extensions are, however, outside the scope of this paper.

Several principles and methods have been developed to find a suitable linear transformation. These include principal component analysis, factor analysis, projection pursuit, independent component analysis, and many more. Usually, these methods define a principle that tells which transform is optimal. The

[0]. Updates, corrections, and comments should be sent to Aapo Hyvärinen at aapo.hyvarinen@hut.fi.

Neural Computing Surveys 2, 94-128, 1999, http://www.icsi.berkeley.edu/~jagota/NCS

optimality may be defined in the sense of optimal dimension reduction, statistical 'interestingness' of the resulting components s_i, simplicity of the transformation $\mathbf{W}$, or other criteria, including application-oriented ones.

Recently, a particular method for finding a linear transformation, called independent component analysis (ICA), has gained wide-spread attention. As the name implies, the basic goal is to find a transformation in which the components s_i are statistically as independent from each other as possible. ICA can be applied, for example, for blind source separation, in which the observed values of $\mathbf{x}$ correspond to a realization of an m-dimensional discrete-time signal $\mathbf{x}(t)$, $t = 1, 2, ...$ Then the components $s_i(t)$ are called source signals, which are usually original, uncorrupted signals or noise sources. Often such sources are statistically independent from each other, and thus the signals can be recovered from linear mixtures x_i by finding a transformation in which the transformed signals are as independent as possible, as in ICA. Another promising application is feature extraction, in which s_i is the coefficient of the i-th feature in the observed data vector $\mathbf{x}$. The use of ICA for feature extraction is motivated by results in neurosciences that suggest that the similar principle of redundancy reduction explains some aspects of the early processing of sensory data by the brain. ICA has also applications in exploratory data analysis in the same way as the closely related method of projection pursuit.

In this paper, we review the theory and methods for ICA. First, we discuss relevant classical representation methods in Section 2. In Section 3, we define ICA, and show its connections to the classical methods as well as some of its applications. In Section 4, different contrast (objective) functions for ICA are reviewed. Next, corresponding algorithms are given in Section 5. The noisy version of ICA is treated in Section 6. Section 7 concludes the paper. For other reviews on ICA, see e.g. [3, 24, 95].

2 Classical Linear Transformations

2.1 Introduction

Several principles have been developed in statistics, neural computing, and signal processing to find a suitable linear representation of a random variable. In this Section, we discuss classical methods for determining the linear transformation as in (2). All the methods discussed in this paper are based on using centered variables. In other words, the mean of the random vector is subtracted. To simplify the discussion, it is henceforth assumed that the variable $\mathbf{x}$ is centered, which means that it has already been transformed by $\mathbf{x} = \mathbf{x}_0 - E\{\mathbf{x}_0\}$, where $\mathbf{x}_0$ is the original non-centered variable.

2.2 Second-order methods

The most popular methods for finding a linear transform as in Eq. (2) are second-order methods. This means methods that find the representation using only the information contained in the covariance matrix of the data vector $\mathbf{x}$. Of course, the mean is also used in the initial centering. The use of second-order techniques is to be understood in the context of the classical assumption of Gaussianity. If the variable $\mathbf{x}$ has a normal, or Gaussian distribution, its distribution is completely determined by this second-order information. Thus it is useless to include any other information. Another reason for the popularity of the second-order methods is that they are computationally simple, often requiring only classical matrix manipulations.

The two classical second-order methods are principal component analysis and factor analysis, see [51, 77, 87]. One might roughly characterize the second-order methods by saying that their purpose is to find a *faithful* representation of the data, in the sense of reconstruction (mean-square) error. This is in contrast to most higher-order methods (see next Section) which try to find a *meaningful* representation. Of course, meaningfulness is a task-dependent property, but these higher-order methods seem to be able to find meaningful representations in a wide variety of applications [36, 46, 80, 81].

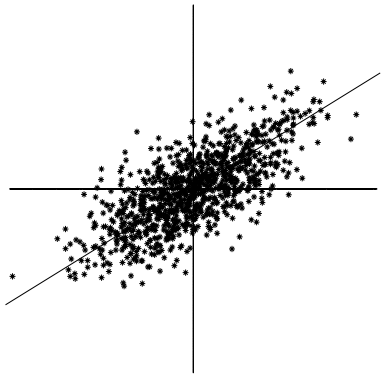

Figure 1: Principal component analysis of a two-dimensional data cloud. The line shown is the direction of the first principal component, which gives an optimal (in the mean-square sense) linear reduction of dimension from 2 to 1 dimensions.

2.2.1 Principal component analysis

Principal Component Analysis, or PCA (see [77, 87]), is widely used in signal processing, statistics, and neural computing. In some application areas, this is also called the (discrete) Karhunen-Loève transform, or the Hotelling transform.

The basic idea in PCA is to find the components $s_1, s_2, ..., s_n$ so that they explain the maximum amount of variance possible by n linearly transformed components. PCA can be defined in an intuitive way using a recursive formulation. Define the direction of the first principal component, say $\mathbf{w}_1$, by

$$\mathbf{w}_1 = \arg \max_{\|\mathbf{w}\|=1} E\{(\mathbf{w}^T\mathbf{x})^2\} \tag{3}$$

where $\mathbf{w}_1$ is of the same dimension m as the random data vector $\mathbf{x}$. (All the vectors in this paper are column vectors). Thus the first principal component is the projection on the direction in which the variance of the projection is maximized. Having determined the first $k-1$ principal components, the k-th principal component is determined as the principal component of the residual:

$$\mathbf{w}_k = \arg \max_{\|\mathbf{w}\|=1} E\{[\mathbf{w}^T(\mathbf{x} - \sum_{i=1}^{k-1} \mathbf{w}_i\mathbf{w}_i^T\mathbf{x})]^2\} \tag{4}$$

The principal components are then given by $s_i = \mathbf{w}_i^T\mathbf{x}$. In practice, the computation of the $\mathbf{w}_i$ can be simply accomplished using the (sample) covariance matrix $E\{\mathbf{x}\mathbf{x}^T\} = \mathbf{C}$. The $\mathbf{w}_i$ are the eigenvectors of $\mathbf{C}$ that correspond to the n largest eigenvalues of $\mathbf{C}$.

The basic goal in PCA is to reduce the dimension of the data. Thus one usually chooses $n << m$. Indeed, it can be proven that the representation given by PCA is an optimal linear dimension reduction technique in the mean-square sense [77]. Such a reduction in dimension has important benefits. First, the computational overhead of the subsequent processing stages is reduced. Second, noise may be reduced, as the data not contained in the n first components may be mostly due to noise. Third, a projection into a subspace of a very low dimension, for example two, is useful for visualizing the data. Note that often it is not necessary to use the n principal components themselves, since any other orthonormal basis of the subspace spanned by the principal components (called the PCA subspace) has the same data compression or denoising capabilities.

A simple illustration of PCA is found in Fig. 1, in which the first principal component of a two-dimensional data set is shown.

2.2.2 Factor analysis

A method that is closely related to PCA is factor analysis [51, 87]. In factor analysis, the following generative model for the data is postulated:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n} \tag{5}$$

where $\mathbf{x}$ is the vector of the observed variables, $\mathbf{s}$ is the vector of the latent variables (factors) that cannot be observed, $\mathbf{A}$ is a constant $m \times n$ matrix, and the vector $\mathbf{n}$ is noise, of the same dimension, m, as $\mathbf{x}$. All the variables in $\mathbf{s}$ and $\mathbf{n}$ are assumed to be Gaussian. In addition, it is usually assumed that $\mathbf{s}$ has a lower dimension than $\mathbf{x}$. Thus, factor analysis is basically a method of reducing the dimension of the data, in a way similar to PCA.

There are two main methods for estimating the factor analytic model [87]. The first method is the method of principal factors. As the name implies, this is basically a modification of PCA. The idea is here to apply PCA on the data $\mathbf{x}$ in such a way that the effect of noise is taken into account. In the simplest form, one assumes that the covariance matrix of the noise $\mathbf{\Sigma} = E\{\mathbf{n}\mathbf{n}^T\}$ is known. Then one finds the factors by performing PCA using the modified covariance matrix $\mathbf{C} - \mathbf{\Sigma}$, where $\mathbf{C}$ is the covariance matrix of $\mathbf{x}$. Thus the vector $\mathbf{s}$ is simply the vector of the principal components of $\mathbf{x}$ with noise removed. A second popular method, based on maximum likelihood estimation, can also be reduced to finding the principal components of a modified covariance matrix. For the general case where the noise covariance matrix is not known, different methods for estimating it are described in [51, 87].

Nevertheless, there is an important difference between factor analysis and PCA, though this difference has little to do with the formal definitions of the methods. Equation (5) does not define the factors uniquely (i.e. they are not identifiable), but only up to a rotation [51, 87]. This indeterminacy should be compared with the possibility of choosing an arbitrary basis for the PCA subspace, i.e., the subspace spanned by the first n principal components. Therefore, in factor analysis, it is conventional to search for a 'rotation' of the factors that gives a basis with some interesting properties. The classical criterion is *parsimony* of representation, which roughly means that the matrix $\mathbf{A}$ has few significantly non-zero entries. This principle has given rise to such techniques as the varimax, quartimax, and oblimin rotations [51]. Such a rotation has the benefit of facilitating the interpretation of the results, as the relations between the factors and the observed variables become simpler.

2.3 Higher-order methods

Higher-order methods use information on the distribution of $\mathbf{x}$ that is not contained in the covariance matrix. In order for this to be meaningful, the distribution of $\mathbf{x}$ must not be assumed to be Gaussian, because all the information of (zero-mean) Gaussian variables is contained in the covariance matrix. For more general families of density functions, however, the representation problem has more degrees of freedom. Thus much more sophisticated techniques may be constructed for non-Gaussian random variables.

Indeed, the transform defined by second-order methods like PCA is not useful for many purposes where optimal reduction of dimension in the mean-square sense is not needed. This is because PCA neglects such aspects of non-Gaussian data as clustering and independence of the components (which, for non-Gaussian data, is not the same as uncorrelatedness), as illustrated in Fig. 2. We shall here review three conventional methods based on higher-order statistics: projection pursuit, redundancy reduction, and blind deconvolution.

2.3.1 Projection pursuit

Projection pursuit [45, 46, 57, 78, 132, 37] is a technique developed in statistics for finding 'interesting' projections of multidimensional data. Such projections can then be used for optimal visualization of the clustering structure of the data, and for such purposes as density estimation and regression. Reduction of dimension is also an important objective here, especially if the aim is visualization of the data.

In basic (1-D) projection pursuit, we try to find directions $\mathbf{w}$ such that the projection of the data in that direction, $\mathbf{w}^T\mathbf{x}$, has an 'interesting' distribution, i.e., displays some structure. It has been argued by Huber [57] and by Jones and Sibson [78] that the Gaussian distribution is the least interesting one, and that the most interesting directions are those that show the least Gaussian distribution.

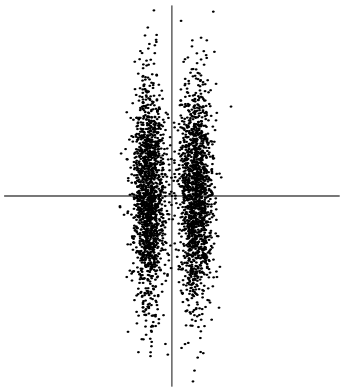

Figure 2: A classical illustration of the problems of variance-based methods like PCA. The data in this figure is clearly divided into two clusters. However, the principal component, i.e. the direction of maximum variance, would be vertical. Projections on the principal component would thus not produce any separation of clusters. In contrast, the projection pursuit direction is horizontal, providing optimal separation of the clusters.

The usefulness of finding such projections can be seen in Fig. 2, where the projection on the projection pursuit direction, which is horizontal, clearly shows the clustered structure of the data. The projection on the first principal component (vertical), on the other hand, fails to show this structure.

In projection pursuit, one thus wants to reduce the dimension in such a way that some of the 'interesting' features of the data are preserved. This is in contrast to PCA where the objective is to reduce the dimension so that the representation is as faithful as possible in the mean-square sense.

The central theoretical problem in projection pursuit is the definition of the projection pursuit index that defines the 'interestingness' of a direction. Usually, the index is some measure of non-Gaussianity. A most natural choice is using differential entropy [57, 78]. The differential entropy H of a random vector $\mathbf{y}$ whose density is $f(.)$, is defined as:

$$H(\mathbf{y}) = -\int f(\mathbf{y}) \log f(\mathbf{y}) \mathrm{d}\mathbf{y} \tag{6}$$

Now, consider zero-mean variables $\mathbf{y}$ of different densities f, and constrain the covariance of $\mathbf{y}$ to be fixed. Then the differential entropy $H(\mathbf{y})$ is maximized with respect to f when f is a Gaussian density. For any other distribution, entropy is strictly smaller. Thus one might try to find projection pursuit directions by minimizing $H(\mathbf{w}^T\mathbf{x})$ with respect to $\mathbf{w}$, constraining the variance of $\mathbf{w}^T\mathbf{x}$ to be constant.

The problem with differential entropy is that the estimation of entropy according to definition (6) requires estimation of the density of $\mathbf{w}^T\mathbf{x}$, which is difficult both in theory and in practice. Therefore, other measures of non-normality have been proposed [37, 46]. These are based on weighted L^2 distances between the density of $\mathbf{x}$ and the multivariate Gaussian density. Another possibility is to use cumulant-based approximations of differential entropy [78]. Furthermore, in [64], approximations of negentropy based on the maximum entropy principle were introduced. More details can be found in Section 4.4.1.

2.3.2 Redundancy reduction

According to Barlow [6, 7, 8, 9, 10] and several other authors [39, 4, 44, 128], an important characteristic of sensory processing in the brain is 'redundancy reduction'. One aspect of redundancy reduction is that the input data is represented using components (features) that are as independent from each other as possible. Such a representation seems to be very useful for later processing stages. Theoretically, the values of the components are given by the activities of the neurons, and $\mathbf{x}$ is represented as a sum of the weight vectors of the neurons, weighted by their activations. This leads to a linear encoding like the other methods in this Section.

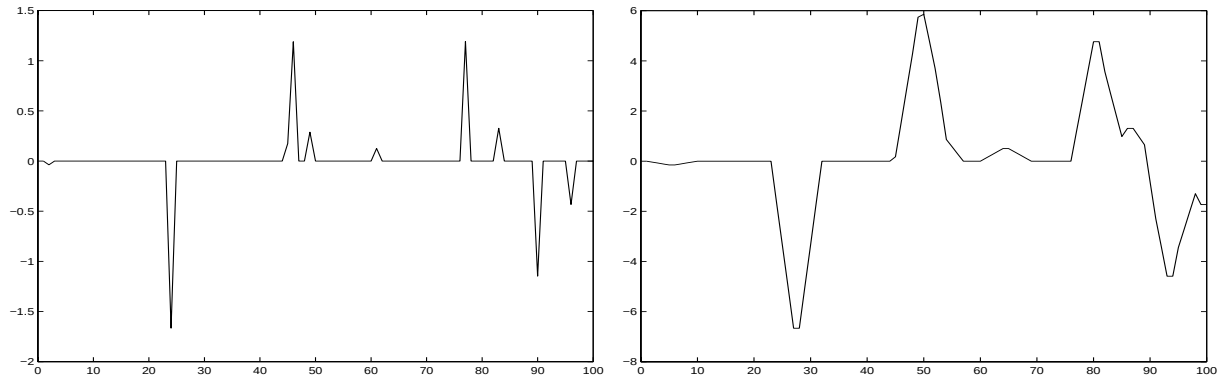

Figure 3: An illustration of blind deconvolution. The original signal is depicted on the left. On the right, a convolved version of the signal is shown. The problem is then to recover the signal on the left, observing only the signal on the right.

One method for performing redundancy reduction is sparse coding [7, 9, 44]. Here the idea is to represent the data $\mathbf{x}$ using a set of neurons so that only a small number of neurons is activated at the same time. Equivalently, this means that a given neuron is activated only rarely. If the data has certain statistical properties (it is 'sparse'), this kind of coding leads to approximate redundancy reduction [44]. A second method for redundancy reduction is predictability minimization [128]. This is based on the observation that if two random variables are independent, they provide no information that could be used to predict one variable using the other one.

2.3.3 Blind deconvolution

Blind deconvolution[1] is different from the other techniques discussed in this Section in the sense that (in the very simplest case) were are dealing with one-dimensional time signals (or time series) instead of multidimensional data, though blind deconvolution can also be extended to the multidimensional case. Blind deconvolution is an important research topic with a vast literature. We shall here describe only a special case of the problem that is closely connected to ours.

In blind deconvolution, a convolved version $x(t)$ of a scalar signal $s(t)$ is observed, without knowing the signal $s(t)$ or the convolution kernel [126, 42, 54, 53, 92, 129, 130, 148]. The problem is then to find a separating filter h so that $s(t) = h(t) * x(t)$. An illustration can be found in Fig. 3.

The equalizer $h(t)$ is assumed to be a FIR filter of sufficient length, so that the truncation effects can be ignored. A special case of blind deconvolution that is especially interesting in our context is the case where it is assumed that the values of the signal $s(t)$ at two different points of time are statistically independent. Under certain assumptions, this problem can be solved by simply whitening the signal $x(t)$. However, to solve the problem in full generality, one must assume that the signal $s(t)$ is non-Gaussian, and use higher-order information [53, 129]. Thus the techniques used for solving this (special case of the) problem are very similar to the techniques used in other higher-order methods discussed in this Section.

[1] Often the term 'blind equalization' is used in the same sense.

3 INDEPENDENT COMPONENT ANALYSIS

3.1 Statistical independence

To begin with, we shall recall some basic definitions needed. Denote by $y_1, y_2, ..., y_m$ some random variables with joint density $f(y_1, ..., y_m)$. For simplicity, assume that the variables are zero-mean. The variables y_i are (mutually) independent, if the density function can be factorized [122]:

$$f(y_1, ..., y_m) = f_1(y_1) f_2(y_2) ... f_m(y_m) \tag{7}$$

where $f_i(y_i)$ denotes the marginal density of y_i. To distinguish this form of independence from other concepts of independence, for example, linear independence, this property is sometimes called statistical independence.

Independence must be distinguished from uncorrelatedness, which means that

$$E\{y_i y_j\} - E\{y_i\}E\{y_j\} = 0, \text{ for } i \neq j. \tag{8}$$

Independence is in general a much stronger requirement than uncorrelatedness. Indeed, if the y_i are independent, one has

$$E\{g_1(y_i)g_2(y_j)\} - E\{g_1(y_i)\}E\{g_2(y_j)\} = 0, \text{ for } i \neq j. \tag{9}$$

for any[2] functions g_1 and g_2 [122]. This is clearly a stricter condition than the condition of uncorrelatedness. There is, however, an important special case where independence and uncorrelatedness are equivalent. This is the case when $y_1, ..., y_m$ have a joint Gaussian distribution (see [36]). Due to this property, independent component analysis is not interesting (or possible) for Gaussian variables, as will be seen below.

3.2 Definitions of linear independent component analysis

Now we shall define the problem of independent components analysis, or ICA. We shall only consider the linear case here, though non-linear forms of ICA also exist. In the literature, at least three different basic definitions for linear ICA can be found [36, 80], though the differences between the definitions are usually not emphasized. This is probably due to the fact that ICA is such a new research topic: most research has concentrated on the simplest one of these definitions. In the definitions, the observed m-dimensional random vector is denoted by $\mathbf{x} = (x_1, ..., x_m)^T$.

The first and most general definition is as follows:

Definition 1 *(General definition) ICA of the random vector* $\mathbf{x}$ *consists of finding a linear transform* $\mathbf{s} = \mathbf{W}\mathbf{x}$ *so that the components* s_i *are as independent as possible, in the sense of maximizing some function* $F(s_1, ..., s_m)$ *that measures independence.*

This definition is the most general in the sense that no assumptions on the data are made, which is in contrast to the definitions below. Of course, this definition is also quite vague as one must also define a measure of independence for the s_i. One cannot use the definition of independence in Eq. (7), because it is not possible, in general, to find a linear transformation that gives strictly independent components. The problem of defining a measure of independence will be treated in the next Section. A different approach is taken by the following more estimation-theoretically oriented definition:

Definition 2 *(Noisy ICA model) ICA of a random vector* $\mathbf{x}$ *consists of estimating the following generative model for the data:*

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n} \tag{10}$$

where the latent variables (components) s_i *in the vector* $\mathbf{s} = (s_1, ..., s_n)^T$ *are assumed independent. The matrix* $\mathbf{A}$ *is a constant* $m \times n$ *'mixing' matrix, and* $\mathbf{n}$ *is a* m*-dimensional random noise vector.*

[2] The functions must be assumed measurable. We shall, however, omit any questions of measurability in this paper.

This definition reduces the ICA problem to ordinary estimation of a latent variable model. However, this estimation problem is not very simple, and therefore the great majority of ICA research has concentrated on the following simplified definition:

Definition 3 *(Noise-free ICA model) ICA of a random vector* $\mathbf{x}$ *consists of estimating the following generative model for the data:*

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{11}$$

where $\mathbf{A}$ *and* $\mathbf{s}$ *are as in Definition 2.*

Here the noise vector has been omitted. This is also the model introduced by Jutten and Hérault in their seminal paper [80] (and earlier by Jutten in his PhD [79] in French), which was probably the earliest explicit formulation of ICA (but see [5]).

In this paper, we shall concentrate on this noise-free ICA model definition. This choice can be partially justified by the fact that most of the research on ICA has also concentrated on this simple definition. Even the estimation of the noise-free model has proved to be a task difficult enough. The noise-free model may be thus considered a tractable approximation of the more realistic noisy model. The justification for this approximation is that methods using the simpler model seem to work for certain kinds of real data, as will be seen below. The estimation of the noisy ICA model is treated in Section 6.

It can be shown [36] that if the data does follow the generative model in Eq. (11), Definitions 1 and 3 become asymptotically equivalent, if certain measures of independence are used in Definition 1, and the natural relation $\mathbf{W} = \mathbf{A}^{-1}$ is used with $n = m$.

3.3 Identifiability of the ICA model

The identifiability of the noise-free ICA model has been treated in [36]. By imposing the following fundamental restrictions (in addition to the basic assumption of statistical independence), the identifiability of the model can be assured.

1. All the independent components s_i, with the possible exception of one component, must be non-Gaussian.
2. The number of observed linear mixtures m must be at least as large as the number of independent components n, i.e., $m \geq n$.
3. The matrix $\mathbf{A}$ must be of full column rank.

Usually, it is also assumed that $\mathbf{x}$ and $\mathbf{s}$ are centered, which is in practice no restriction, as this can always be accomplished by subtracting the mean from the random vector $\mathbf{x}$. If $\mathbf{x}$ and $\mathbf{s}$ are interpreted as stochastic processes instead of simply random variables, additional restrictions are necessary. At the minimum, one has to assume that the stochastic processes are stationary in the strict sense. Some restrictions of ergodicity with respect to the quantities estimated are also necessary [122]. These assumptions are fulfilled, for example, if the process is i.i.d. over time. After such assumptions, one can consider the stochastic process as a random variable, as we do here.

A basic, but rather insignificant indeterminacy in the model is that the independent components and the columns of $\mathbf{A}$ can only be estimated up to a multiplicative constant, because any constant multiplying an independent component in Eq. (11) could be canceled by dividing the corresponding column of the mixing matrix $\mathbf{A}$ by the same constant. For mathematical convenience, one usually defines that the independent components s_i have unit variance. This makes the independent components unique, up to a multiplicative sign (which may be different for each component) [36].

The definitions of ICA given above imply no ordering of the independent components, which is in contrast to, e.g., PCA. It is possible, however, to introduce an order between the independent components. One way is to use the norms of the columns of the mixing matrix, which give the contributions of the independent components to the variances of the x_i. Ordering the s_i according to descending norm of the corresponding

columns of $\mathbf{A}$, for example, gives an ordering reminiscent of PCA. A second way is to use the non-Gaussianity of the independent components. Non-Gaussianity may be measured, for example, using one of the projection pursuit indexes in Section 2.3.1 or the contrast functions to be introduced in Section 4.4.3. Ordering the s_i according to non-Gaussianity gives an ordering related to projection pursuit.

The first restriction (non-Gaussianity) in the list above, is necessary for the identifiability of the ICA model [36]. Indeed, for Gaussian random variables mere uncorrelatedness implies independence, and thus any decorrelating representation would give independent components. Nevertheless, if more than one of the components s_i are Gaussian, it is still possible to identify the non-Gaussian independent components, as well as the corresponding columns of the mixing matrix.

On the other hand, the second restriction, $m \geq n$, is not completely necessary. Even in the case where $m < n$, the mixing matrix $\mathbf{A}$ seems to be identifiable [21] (though no rigorous proofs exist to our knowledge), whereas the realizations of the independent components are not identifiable, because of the noninvertibility of $\mathbf{A}$. However, most of the existing theory for ICA is not valid in this case, and therefore we have to make the second assumption in this paper. Recent work on the case $m < n$, often called ICA with overcomplete bases, can be found in [21, 118, 98, 99, 69].

Some rank restriction on the mixing matrix, like the third restriction given above, is also necessary, though the form given here is probably not the weakest possible.

As regards the identifiability of the *noisy* ICA model, the same three restrictions seem to guarantee partial identifiability, if the noise is assumed to be independent from the components s_i [35, 93, 107]. In fact, the noisy ICA model is a special case of the noise-free ICA model with $m < n$, because the noise variables could be considered as additional independent components. In particular, the mixing matrix $\mathbf{A}$ is still identifiable. In contrast, the realizations of the independent components s_i can no longer be identified, because they cannot be completely separated from noise. It would seem that the noise covariance matrix is also identifiable [107].

In this paper, we shall assume that the assumptions 1–3 announced above are valid, and we shall treat only the noiseless ICA model, except for some comments on the estimation of the noisy model. We also make the conventional assumption that the dimension of the observed data equals the number of the independent components, i.e., $n = m$. This simplification is justified the fact that if $m > n$, the dimension of the observed vector can always be reduced so that $m = n$. Such a reduction of dimension can be achieved by existing methods such as PCA.

3.4 Relations to classical methods

ICA is closely related to several of the methods described in Section 2.

1. By definition, ICA can be considered a method for achieving *redundancy reduction*. Indeed, there is experimental evidence that for certain kinds of sensory data, the conventional ICA algorithms do find directions that are compatible with existing neurophysiological data, assumed to reflect redundancy reduction [14, 58, 116]. See Section 3.5.2.

2. In the noise-free case, the estimation of the ICA model means simply finding certain 'interesting' projections, which give estimates of the independent components. Thus ICA can be considered, at least using Definitions 1 and 3, a special case of *projection pursuit*. Indeed, as will be explained in Section 4, the conventional criteria used for finding the 'interesting' directions in projections pursuit coincide essentially with the criteria used for estimating the independent components.

3. Another close affinity can be found between ICA and *blind deconvolution* (more precisely, the special case of blind deconvolution where the original signal is i.i.d. over time). Due to the assumption that the values of the original signal $s(t)$ are independent for different t, this problem is formally closely related to the problem of independent component analysis. Indeed, many ideas developed for blind deconvolution can be directly applied for ICA, and vice versa. Blind deconvolution, and especially the elegant and powerful framework developed in [42], can thus be considered an intellectual ancestor of ICA.

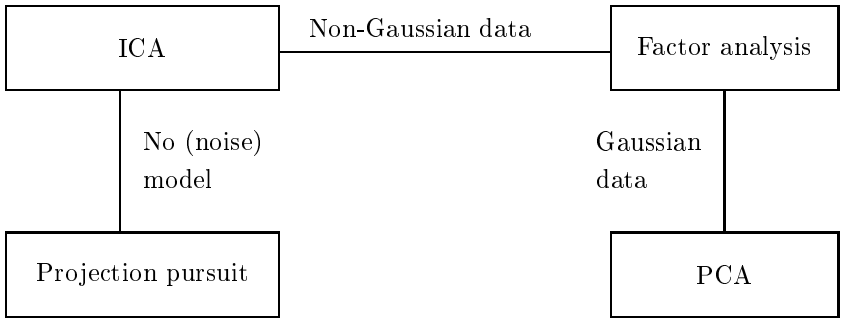

Figure 4: The relations between ICA and some other methods. The lines show close connections, and the texts next to the lines show the assumptions needed for the connection.

4. Comparing Eq. (10) in Definition 2 with the definition of factor analysis in Eq. (5), the connection between factor analysis and ICA becomes clear. Indeed, ICA may be considered a *non-Gaussian factor analysis*. The main difference is that usually in ICA, reduction of dimension is considered only as a secondary objective, but this need not be the case. Indeed, a simple combination of factor analysis and ICA can be obtained using factor rotations. Above we saw that after finding the factor subspace, a suitable rotation is usually performed. ICA could also be conceived as such a rotation, where the criterion depends on the higher-order statistics of the factors, instead of the structure of the matrix $\mathbf{A}$. Such a method is roughly equivalent to the method advocated in [71, 61, 84], which consists of first reducing the dimension by PCA, and then performing ICA without further dimension reduction.

5. Using Definition 1, the relation to *principal component analysis* is also evident. Both methods formulate a general objective function that define the 'interestingness' of a linear representation, and then maximize that function. A second relation between PCA and ICA is that both are related to factor analysis, though under the contradictory assumptions of Gaussianity and non-Gaussianity, respectively. The affinity between PCA and ICA may be, however, less important than the affinity between ICA and the other methods discussed above. This is because PCA and ICA define their objective functions in quite different ways. PCA uses only second-order statistics, while ICA is impossible using only second-order statistics. PCA emphasizes dimension reduction, while ICA may reduce the dimension, increase it or leave it unchanged. However, the relation between ICA and nonlinear versions of the PCA criteria, as defined in [82, 112], is quite strong, as will be seen in the next Section.

The connections between ICA and some other methods are illustrated in Fig. 4. The lines in the diagram indicate very close connections, under the assumptions given next to the lines. First, if no assumptions on the data are made, and in particular no noise is postulated in the data, ICA can be considered a method of exploratory data analysis, as projection pursuit. Indeed, using Definition 1, ICA means simply finding some interesting projections of the data, and the measures of interestingness are essentially equivalent in the two methods, as will be seen in the next Section. On the other hand, if one assumes a noisy data model, as in Definition 2, ICA can be considered a variation of factor analysis for non-Gaussian data. Thus one has two possible approaches to ICA that are quite different as they stem from two clearly distinct classical methods. ICA according to Definition 3, or the noise-free ICA data model, is something between these two approaches. As for PCA, its connection to ICA can be considered indirect, since it can be used to perform factor analysis for Gaussian data.

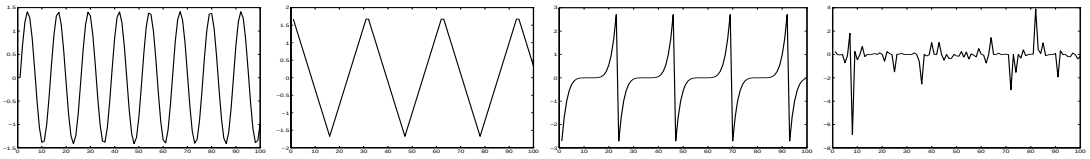

Figure 5: An illustration of blind source separation. This figure shows four source signals, or independent components.

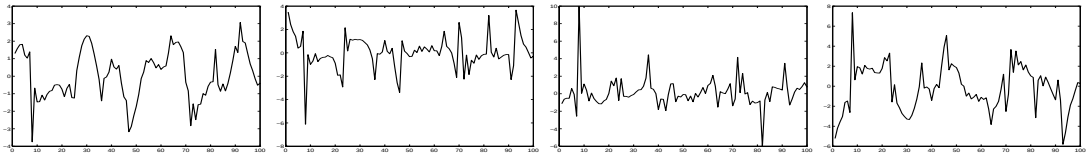

Figure 6: Due to some external circumstances, only linear mixtures of the source signals in Fig. 5, as depicted here, can be observed.

3.5 Applications of ICA

3.5.1 Blind source separation

The classical application of the ICA model is blind source separation [80]. In blind source separation, the observed values of $\mathbf{x}$ correspond to a realization of an m-dimensional discrete-time signal $\mathbf{x}(t)$, $t = 1, 2, ...$ Then the independent components $s_i(t)$ are called source signals, which are usually original, uncorrupted signals or noise sources. A classical example of blind source separation is the cocktail party problem. Assume that several people are speaking simultaneously in the same room, as in a cocktail party. Then the problem is to separate the voices of the different speakers, using recordings of several microphones in the room. In principle, this corresponds to the ICA data model, where $x_i(t)$ is the recording of the i-th microphone, and the $s_i(t)$ are the waveforms of the voices[3]. A more practical application is noise reduction. If one of the sources is the original, uncorrupted source and the others are noise sources, estimation of the uncorrupted source is in fact a denoising operation.

A simple artificial illustration of blind source separation is given in Figures 5–7. In this illustration, deterministic signals were used for purposes of illustration. However, the spectral properties of the signals are not used in the ICA framework, and thus the results would remain unchanged if the signals were simply (non-Gaussian) white noise.

In [140, 141, 101], results on applying ICA for blind separation of electroencephalographic (EEG) and magnetoencephalographic (MEG) data were reported. The EEG data consisted of recordings of brain activity obtained using electrodes attached to the scalp. Thus a 23-dimensional signal vector was observed. The MEG data was obtained with a more sophisticated measuring method, giving rise to a 122-dimensional signal vector. The ICA algorithms succeeded in separating certain source signals that were so-called artifacts, or noise sources not corresponding to brain activity [140, 141]. Canceling these noise sources is a central, and as yet unsolved problem in EEG and MEG signal processing. ICA offers a very promising method. Similarly, ICA can be used for decomposition of evoked field potentials measured by EEG or MEG [142, 143], which is an application of considerable interest in the neurosciences. Application on further brain imaging data, this time obtained by functional magnetic resonance imaging (fMRI), is reported in [104].

Another application area is on economic time series. Some work is reported in [89]. Very recently, applications on telecommunications have also been published [125].

Since most of the research on ICA has been done with the application of source separation in mind, many authors treating the ICA problem do not use the term ICA, but speak simply of blind source separation

[3] This application is to be taken rather as an illustrative example than a real application. In practice, the situation is much more complicated than described here due to echos and, above all, time delays, see Section 7.

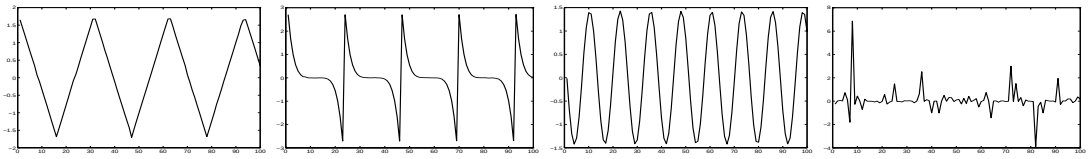

Figure 7: Using only the linear mixtures in Fig. 6, the source signals in Fig. 5 can be estimated, up to some multiplying factors. This figure shows the estimates of the source signals.

(BSS). We make, however, a clear distinction between ICA, which is a theoretical problem or data model with different applications, and blind source separation, which is an application that can be solved using various theoretical approaches, including but not limited to ICA. In fact, the blind source separation problem can be solved using methods very different from ICA. In particular, methods using frequency information, or spectral properties, are prominent (see [16, 18, 136, 147, 135, 105]). These methods can be used for time-correlated signals, which is of course the usual case in blind source separation, but not in many other applications of ICA (see below). Using such frequency methods, it is also possible to separate Gaussian source signals.

3.5.2 Feature extraction

Another application of ICA is feature extraction [14, 13, 58, 74, 81, 116]. Then the columns of $\mathbf{A}$ represent features, and s_i is the coefficient of the i-th feature in an observed data vector $\mathbf{x}$. The use of ICA for feature extraction is motivated by the theory of redundancy reduction, see Section 2.3.2.

In [116], an essentially equivalent method based on sparse coding was applied for extraction of low-level features of natural image data. The results show that the extracted features correspond closely to those observed in the primary visual cortex [116, 118]. These results seem to be very robust, and have been later replicated by several other authors and methods [14, 58, 74, 81, 116]. A systematical comparison between the ICA features and the properties of the simple cells in the macaque primary visual cortex was conducted in [139, 138], where the authors found a good match for most of the parameters, especially if video sequences were used instead of still images. The obtained features are also closely connected to those offered by wavelet theory and Gabor analysis [38, 102]. In fact, in [68, 74] it was shown how to derive a completely adaptive version of wavelet shrinkage from estimation of the noisy ICA model. Application of these features on data compression and pattern recognition are important research topics.

3.5.3 Blind deconvolution

A less direct application of the ICA methods can be found in blind deconvolution (see Section 2). Due to the fact that the values of the original signal $s(t)$ are independent for different t, this problem can be solved using essentially the same formalism as used in ICA, as noted above. Indeed this problem can also be represented (though only approximately) by Eq. (11); then the realizations of $\mathbf{x}$ and $\mathbf{s}$ are vectors containing $n = m$ subsequent observations of the signals $x(t)$ and $s(t)$, beginning at different points of time. In other words, a sequence of observations $\mathbf{x}(t)$ is such that $\mathbf{x}(t) = (x(t+n-1), x(t+n-2), ..., x(t))^T$ for $t = 1, 2, ...$. The square matrix $\mathbf{A}$ is determined by the convolving filter. Though this formulation is only approximative, the exact formulation using linear filters would lead to essentially the same algorithms and convergence proofs. Also blind separation of several convolved signals ('multi-channel deconvolution') can be represented combining these two approaches, see, for example, [41, 123, 137, 149, 150, 134, 92].

3.5.4 Other applications

Due to the close connection between ICA and projection pursuit on the one hand, and between ICA and factor analysis on the other, it should be possible to use ICA on many of the applications where projection pursuit and factor analysis are used. These include (exploratory) data analysis in such areas as economics, psychology, and other social sciences, as well as density estimation, and regression [46, 57].

4 OBJECTIVE (CONTRAST) FUNCTIONS FOR ICA

4.1 Introduction

The estimation of the data model of independent component analysis is usually performed by formulating an objective function and then minimizing or maximizing it. Often such a function is called a contrast function, but some authors reserve this term for a certain class of objective functions [36]. Also the terms loss function or cost function are used. We shall here use the term contrast function rather loosely, meaning any function whose optimization enables the estimation of the independent components. We shall also restrict ourselves to the estimation of the *noise-free* ICA model (for noisy ICA, see Section 6), and assume that the three restrictions in Section 3.3 that are sufficient for the identifiability of the model, are imposed.

4.2 Objective functions vs. algorithms

We draw here a distinction between the formulation of the objective function, and the algorithm used to optimize it. One might express this in the following 'equation':

$$\text{ICA method} = \text{Objective function} + \text{Optimization algorithm}. \tag{12}$$

In the case of explicitly formulated objective functions, one can use any of the classical methods of optimization for optimizing the objective function, like (stochastic) gradient methods, Newton-like methods, etc. In some cases, however, the algorithm and the estimation principle may be difficult to separate.

The properties of the ICA method depend on both of the elements on the right-hand side of (12). In particular,

- the statistical properties (e.g., consistency, asymptotic variance, robustness) of the ICA method depend on the choice of the objective function, and
- the algorithmic properties (e.g., convergence speed, memory requirements, numerical stability) depend on the optimization algorithm.

These two classes of properties are independent in the sense that different optimization methods can be used to optimize a single objective function, and a single optimization method may be used to optimize different objective functions. In some cases, however, the distinction may not be so clear. In this Section, we shall treat only the choice of the objective function. The optimization of the objective function is treated in Section 5. Therefore, in this Section, we shall compare the objective functions mainly in terms of their statistical properties.

4.3 Multi-unit contrast functions

First, we treat the problem of estimating all the independent components, or the whole data model, at the same time.

4.3.1 Likelihood and network entropy

It is possible to formulate the likelihood in the noise-free ICA model (11), which was done in [124], and then estimate the model by a maximum likelihood method. Denoting by $\mathbf{W} = (\mathbf{w}_1, ..., \mathbf{w}_m)^T$ the matrix $\mathbf{A}^{-1}$, the log-likelihood takes the form [124]:

$$L = \sum_{t=1}^{T} \sum_{i=1}^{m} \log f_i(\mathbf{w}_i^T \mathbf{x}(t)) + T \ln |\det \mathbf{W}| \tag{13}$$

where the f_i are the density functions of the s_i (here assumed to be known), and the $\mathbf{x}(t), t = 1, ..., T$ are the realizations of $\mathbf{x}$.

Another related contrast function was derived from a neural network viewpoint in [12, 108]. This was based on maximizing the output entropy (or information flow) of a neural network with non-linear outputs. Assume that $\mathbf{x}$ is the input to the neural network whose outputs are of the form $g_i(\mathbf{w}_i^T\mathbf{x})$, where the g_i are some non-linear scalar functions, and the $\mathbf{w}_i$ are the weight vectors of the neurons. One then wants to maximize the entropy of the outputs:

$$L_2 = H(g_1(\mathbf{w}_1^T\mathbf{x}), ..., g_m(\mathbf{w}_m^T\mathbf{x})). \tag{14}$$

If the g_i are well chosen, this framework also enables the estimation of the ICA model. Indeed, several authors, e.g., [23, 123], proved the surprising result that the principle of network entropy maximization, or 'infomax', is equivalent to maximum likelihood estimation. This equivalence requires that the non-linearities g_i used in the neural network are chosen as the cumulative distribution functions corresponding to the densities f_i, i.e., $g_i'(.) = f_i(.)$.

The advantage of the maximum likelihood approach is that under some regularity conditions, it is asymptotically efficient; this is a well-known result in estimation theory [127]. However, there are also some drawbacks. First, this approach requires the knowledge of the probability densities of the independent components. These could also be estimated [124, 96], but this complicates the method considerably. A second drawback is that the maximum likelihood solution may be very sensitive to outliers, if the pdf's of the independent components have certain shapes (see [62]), while robustness against outliers is an important property of any estimator [50, 56][4].

4.3.2 Mutual information and Kullback-Leibler divergence

Theoretically the most satisfying contrast function in the multi-unit case is, in our view, mutual information.

Using the concept of differential entropy defined in Eq. (6), one defines the mutual information I between m (scalar) random variables $y_i, i = 1...m$, as follows

$$I(y_1, y_2, ..., y_m) = \sum_i H(y_i) - H(\mathbf{y}). \tag{15}$$

where H denotes differential entropy. The mutual information is a natural measure of the dependence between random variables. It is always non-negative, and zero if and only if the variables are statistically independent. Thus the mutual information takes into account the whole dependence structure of the variables. Finding a transform that minimizes the mutual information between the components s_i is a very natural way of estimating the ICA model [36]. This approach gives at the same time a method of performing ICA according to the general Definition 1 in Section 3. For future use, note that by the properties of mutual information, we have for an invertible linear transformation $\mathbf{y} = \mathbf{W}\mathbf{x}$:

$$I(y_1, y_2, ..., y_m) = \sum_i H(y_i) - H(\mathbf{x}) - \log|\det \mathbf{W}|. \tag{16}$$

The use of mutual information can also be motivated using the Kullback-Leibler divergence, defined for two probability densities f_1 and f_2 as

$$\delta(f_1, f_2) = \int f_1(\mathbf{y}) \log \frac{f_1(\mathbf{y})}{f_2(\mathbf{y})} d\mathbf{y}. \tag{17}$$

The Kullback-Leibler divergence can be considered as a kind of a distance between the two probability densities, though it is not a real distance measure because it is not symmetric. Now, if the y_i in (15) were independent, their joint probability density could be factorized as in the definition of independence in

[4] Note that it is possible, at least in theory, to consider the robustness of estimators even in the case of the ICA model without any noise. This is because the distribution of $\mathbf{s}$ may be ϵ-contaminated, or contain outliers, even if $\mathbf{x}$ were generated from $\mathbf{s}$ exactly according to the noise-free model. A more realistic analysis of robustness would require, however, the introduction of noise in the model.

Eq. (7). Thus one might measure the independence of the y_i as the Kullback-Leibler divergence between the real density $f(\mathbf{y})$ and the factorized density $\tilde{f}(\mathbf{y}) = f_1(y_1)f_2(y_2)...f_m(y_m)$, where the $f_i(.)$ are the marginal densities of the y_i. In fact, this quantity equals the mutual information of the y_i.

The connection to the Kullback-Leibler divergence also shows the close connection between minimizing mutual information and maximizing likelihood. In fact, the likelihood can be represented as a Kullback-Leibler distance between the observed density and the factorized density assumed in the model [26]. So both of these methods are minimizing the Kullback-Leibler distance between the observed density and a factorized density; actually the two factorized densities are asymptotically equivalent, if the density is accurately estimated as part of the ML estimation method.

The problem with mutual information is that it is difficult to estimate. As was mentioned in Section 2, to use the definition of entropy, one needs an estimate of the density. This problem has severely restricted the use of mutual information in ICA estimation. Some authors have used approximations of mutual information based on polynomial density expansions [36, 1], which lead to the use of higher-order cumulants (for definition of cumulants, see Appendix A).

The polynomial density expansions are related to the Taylor expansion. They give an approximation of a probability density $f(.)$ of a scalar random variable y using its higher-order cumulants. For example, the first terms of the Edgeworth expansion give, for a scalar random variable y of zero mean and unit variance [88]:

$$f(\xi) \approx \varphi(\xi)(1 + \kappa_3(y)h_3(\xi)/6 + \kappa_4(y)h_4(\xi)/24 + ...) \tag{18}$$

where φ is the density function of a standardized Gaussian random variable, the $\kappa_i(y)$ are the cumulants of the random variable y (see Appendix A), and $h_i(.)$ are certain polynomial functions (Hermite polynomials). Using such expansions, one obtains for example the following approximation for mutual information

$$I(\mathbf{y}) \approx C + \frac{1}{48}\sum_{i=1}^{m}[4\kappa_3(y_i)^2 + \kappa_4(y_i)^2 + 7\kappa_4(y_i)^4 - 6\kappa_3(y_i)^2\kappa_4(y_i)] \tag{19}$$

where C is constant; the y_i are here constrained to be uncorrelated. A very similar approximation was derived in [1], and also earlier in the context of projection pursuit in [78].

Cumulant-based approximations such as the one in (19) simplify the use of mutual information considerably. The approximation is valid, however, only when $f(.)$ is not far from the Gaussian density function, and may produce poor results when this is not the case. More sophisticated approximations of mutual information can be constructed by using the approximations of differential entropy that were introduced in [64], based on the maximum entropy principle. In these approximations, the cumulants are replaced by more general measures of nongaussianity, see Section 4.4.3 and Section 4.4.1. Minimization of such an approximation of mutual information was introduced in [65].

4.3.3 Non-linear cross-correlations

Ever since the seminal paper by Jutten and Hérault [80] (and Jutten's PhD paper [79]), several authors have used the principle of canceling non-linear cross-correlations to obtain the independent components [80, 28, 33, 32]. Such non-linear cross-correlations are of the form $E\{g_1(y_i)g_2(y_j)\}$, where g_1 and g_2 are some suitably chosen odd non-linearities. If y_i and y_j are independent, these cross-correlations are zero, under the assumption that the y_i and y_j have symmetric densities. Often the objective function is here formulated only implicitly, and an exact objective function may not even exist. The non-linearities must be chosen according to the pdf's of the independent components; some guidelines are given in [92, 100, 131].

4.3.4 Non-linear PCA criteria

A related approach is offered by the non-linear PCA criteria [82, 83, 84, 112]. These objective functions are based on the idea of introducing a non-linearity in well-known objective functions used in PCA. Take, for example, the recursive definition of PCA in Eq. (3). Introducing a non-linearity $g(.)$ in the formula, one obtains

$$\mathbf{w}_1 = \arg\max_{\|\mathbf{w}\|=1} E\{g(\mathbf{w}^T\mathbf{x})^2\} \tag{20}$$

as a modified principal component of the data $\mathbf{x}$. If the non-linearity is suitably chosen as a function of the pdf's of the independent components, and the data is preprocessed by sphering (see Section 5.2), optimizing such non-linear PCA criteria enables estimation of the ICA model. Other definitions of PCA may also be modified by introducing a non-linearity.

Sometimes in connection with the non-linear PCA criteria, a 'bottom-up' approach is used, as with non-linear cross-correlations. Explicit contrast functions may not be formulated, and the algorithms are often obtained directly from existing PCA algorithms by introducing a suitable non-linearity [112, 84].

4.3.5 Higher-order cumulant tensors

A principle of ICA estimation that is less directly connected with the objective function framework, is the eigenmatrix decomposition of higher-order cumulant tensors (defined in Appendix A). Most solutions use the fourth-order cumulant tensor, whose properties and relation to the estimation of ICA have been studied extensively [20, 21, 22, 29, 27, 36]. Related methods were also introduced in [106]. The fourth-order cumulant tensor can be defined as the following linear operator T from the space of $m \times m$ matrices to itself:

$$T(\mathbf{K})_{ij} = \sum_{k,l} \text{cum}\,(x_i, x_j, x_k, x_l)\mathbf{K}_{kl} \tag{21}$$

where the subscript ij means the (i,j)-th element of a matrix, and $\mathbf{K}$ is a $m \times m$ matrix. This is a linear operator, and thus has m^2 eigenvalues that correspond to eigenmatrices. Solving for the eigenvectors of such eigenmatrices, the ICA model can be estimated [20].

The advantage of this approach is that it requires no knowledge of the probability densities of the independent components. Moreover, cumulants can be used to approximate mutual information [36, 1], as shown above, though the approximation is often very crude. The main drawback of this approach seems to be that the statistical properties of estimators based on cumulants are not very good, see Section 4.4.2.

4.3.6 Weighted covariance matrix

Another interesting approach was developed in [19], in which ICA estimation was performed by ordinary eigenvalue decomposition of a quadratically weighted covariance matrix, i.e., $E\{\|\mathbf{x}\|^2\mathbf{x}\mathbf{x}^T\}$. This approach needs the assumption that all the independent components have different distributions (in particular, different kurtoses). It was realized later [20] that this approach is in fact a special case of the cumulant tensor approach of Section 4.3.5.

4.4 One-unit contrast functions

We use the expression 'one-unit contrast function' to designate any function whose optimization enables estimation of a *single* independent component. Thus, instead of estimating the whole ICA model, we try to find here simply one vector, say $\mathbf{w}$, so that the linear combination $\mathbf{w}^T\mathbf{x}$ equals one of the independent components s_i. This procedure can be iterated to find several independent components. The use of one-unit contrast functions can be motivated by the following:

- The one-unit approach shows a direct connection to projection pursuit. Indeed, all the one-unit contrast functions discussed below can be considered as measures of non-Gaussianity, and therefore this approach gives a unifying framework for these two techniques. The same contrast functions and algorithms can be interpreted in two different ways.

- In many applications, one does not need to estimate all the independent components. Finding only some of them is enough. In the ideal case where the one-unit contrast functions are optimized globally, the independent components are obtained in the order of (descending) non-Gaussianity. In the light of the basic principles of projection pursuit, this means that the most interesting independent components are obtained first. This reduces the computational complexity of the method considerably, if the input data has a high dimension.

- Prior knowledge of the number of independent components is not needed, since the independent components can be estimated one-by-one.
- This approach also shows clearly the connection to neural networks. One can construct a neural network whose units learn so that every neuron optimizes its own contrast function. Thus the approach tends to lead to computationally simple solutions.

After estimating one independent component, one can use simple decorrelation to find a different independent component, since the independent components are by definition uncorrelated. Thus, maximizing the one-unit contrast function under the constraint of decorrelation (with respect to the independent components already found), a new independent component can be found, and this procedure can be iterated to find all the independent components. Symmetric (parallel) decorrelation can also be used, see [71, 60, 65, 84].

4.4.1 Negentropy

A most natural information-theoretic one-unit contrast function is negentropy. From Eq. (15), one is tempted to conclude that the independent components correspond to directions in which the differential entropy of $\mathbf{w}^T\mathbf{x}$ is minimized. This turns out to be roughly the case. However, a modification has to be made, since differential entropy is not invariant for scale transformations. To obtain a linearly (and, in fact, affinely) invariant version of entropy, one defines the negentropy J as follows

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \tag{22}$$

where $\mathbf{y}_{gauss}$ is a Gaussian random vector of the same covariance matrix as $\mathbf{y}$. Negentropy, or negative normalized entropy, is always non-negative, and is zero if and only if $\mathbf{y}$ has a Gaussian distribution [36].

The usefulness of this definition can be seen when mutual information is expressed using negentropy, giving

$$I(y_1, y_2, ..., y_n) = J(\mathbf{y}) - \sum_i J(y_i) + \frac{1}{2}\log\frac{\prod \mathbf{C}^y_{ii}}{\det \mathbf{C}^y} \tag{23}$$

where $\mathbf{C}^y$ is the covariance matrix of $\mathbf{y}$, and the $\mathbf{C}^y_{ii}$ are its diagonal elements. If the y_i are uncorrelated, the third term is 0, and we thus obtain

$$I(y_1, y_2, ..., y_n) = J(\mathbf{y}) - \sum_i J(y_i) \tag{24}$$

Because negentropy is invariant for linear transformations [36], it is now obvious that finding maximum negentropy directions, i.e., directions where the elements of the sum $J(y_i)$ are maximized, is equivalent to finding a representation in which mutual information is minimized. The use of negentropy shows clearly the connection between ICA and projection pursuit. Using differential entropy as a projection pursuit index, as has been suggested in [57, 78], amounts to finding directions in which negentropy is maximized.

Unfortunately, the reservations made with respect to mutual information are also valid here. The estimation of negentropy is difficult, and therefore this contrast function remains mainly a theoretical one. As in the multi-unit case, negentropy can be approximated by higher-order cumulants, for example as follows [78]:

$$J(y) \approx \frac{1}{12}\kappa_3(y)^2 + \frac{1}{48}\kappa_4(y)^2 \tag{25}$$

where $\kappa_i(y)$ is the i-th order cumulant of y. The random variable y is assumed to be of zero mean and unit variance. However, the validity of such approximations may be rather limited. In [64], it was argued that cumulant-based approximations of negentropy are inaccurate, and in many cases too sensitive to outliers. New approximations of negentropy were therefore introduced. In the simplest case, these new approximations are of the form:

$$J(y) \approx c[E\{G(y)\} - E\{G(\nu)\}]^2 \tag{26}$$

where G is practically any non-quadratic function, c is an irrelevant constant, and ν is a Gaussian variable of zero mean and unit variance (i.e., standardized). For the practical choice of G, see below. In [64], these approximations were shown to be better than the cumulant-based ones in several respects.

Actually, the two approximations of negentropy discussed above are interesting as one-unit contrast functions in their own right, as will be discussed next.

4.4.2 Higher-order cumulants

Mathematically the simplest one-unit contrast functions are provided by higher-order cumulants like kurtosis (see Appendix A for definition). Denote by $\mathbf{x}$ the observed data vector, assumed to follow the ICA data model (11). Now, let us search for a linear combination of the observations x_i, say $\mathbf{w}^T\mathbf{x}$, such that its kurtosis is maximized or minimized. Obviously, this optimization problem is meaningful only if $\mathbf{w}$ is somehow bounded; let us assume $E\{(\mathbf{w}^T\mathbf{x})^2\} = 1$. Using the (unknown) mixing matrix $\mathbf{A}$, let us define $\mathbf{z} = \mathbf{A}^T\mathbf{w}$. Then, using the data model $\mathbf{x} = \mathbf{A}\mathbf{s}$ one obtains $E\{(\mathbf{w}^T\mathbf{x})^2\} = \mathbf{w}^T\mathbf{A}\mathbf{A}^T\mathbf{w} = \|\mathbf{z}\|^2 = 1$ (recall that $E\{\mathbf{s}\mathbf{s}^T\} = \mathbf{I}$), and the well-known properties of kurtosis (see Appendix A) give

$$\text{kurt}\,(\mathbf{w}^T\mathbf{x}) = \text{kurt}\,(\mathbf{w}^T\mathbf{A}\mathbf{s}) = \text{kurt}\,(\mathbf{z}^T\mathbf{s}) = \sum_{i=1}^{m} z_i^4\,\text{kurt}\,(s_i). \tag{27}$$

Under the constraint $\|\mathbf{z}\|^2 = 1$, the function (27) has a number of local minima and maxima. To make the argument clearer, let us assume for the moment that in the mixture (11) there is at least one independent component s_j whose kurtosis is negative, and at least one whose kurtosis is positive. Then, as was shown in [40], the extremal points of (27) are the canonical base vectors $\mathbf{z} = \pm\mathbf{e}_j$, i.e., vectors whose all components are zero except one component which is ± 1. The corresponding weight vectors are $\mathbf{w} = \pm(\mathbf{A}^{-1})^T\mathbf{e}_j$, i.e., the rows of the inverse of the mixing matrix $\mathbf{A}$, up to a multiplicative sign. So, by minimizing or maximizing the kurtosis in Eq. (27) under the given constraint, one obtains one of the independent components as $\mathbf{w}^T\mathbf{x} = \pm s_j$. These two optimization modes can also be combined into a single one, because the independent components correspond always to maxima of the *modulus* of the kurtosis.

Kurtosis has been widely used for one-unit ICA (see, for example, [40, 103, 71, 72]), as well as for projection pursuit [78]. The mathematical simplicity of the cumulants, and especially the possibility of proving global convergence results, as in [40, 71, 72], has contributed largely to the popularity of cumulant-based (one-unit) contrast functions in ICA, projection pursuit and related fields. However, it has been shown, for example in [62], that kurtosis often provides a rather poor objective function for the estimation of ICA, if the statistical properties of the resulting estimators are considered. Note that despite the fact that there is no noise in the ICA model (11), neither the independent components nor the mixing matrix can be computed accurately because the independent components s_i are random variables, and, in practice, one only has a finite sample of $\mathbf{x}$. Therefore, the statistical properties of the estimators of $\mathbf{A}$ and the realizations of $\mathbf{s}$ can be analyzed just as the properties of any estimator. Such an analysis was conducted in [62] (see also [28]), and the results show that in terms of robustness and asymptotic variance, the cumulant-based estimators tend to be far from optimal[5]. Intuitively, there are two main reasons for this. Firstly, higher-order cumulants measure mainly the tails of a distribution, and are largely unaffected by structure in the middle of the distribution [46]. Secondly, estimators of higher-order cumulants are highly sensitive to outliers [57]. Their value may depend on only a few observations in the tails of the distribution, which may be outliers.

4.4.3 General contrast functions

To avoid the problems encountered with the preceding objective functions, new one-unit contrast functions for ICA were developed in [60, 64, 65]. Such contrast functions try to combine the positive properties of the preceding contrast functions, i.e. have statistically appealing properties (in contrast to cumulants), require no prior knowledge of the densities of the independent components (in contrast to basic maximum likelihood estimation), allow a simple algorithmic implementation (in contrast to maximum likelihood approach with

[5] Here we consider robustness to be one form of optimality, in particular, minimax-optimality in the neighborhood of the assumed model in a space of statistical models [56].

simultaneous estimation of the densities), and be simple to analyze (in contrast to non-linear cross-correlation and non-linear PCA approaches). The generalized contrast functions (introduced in [60]), which can be considered generalizations of kurtosis, seem to fulfill these requirements.

To begin with, note that one intuitive interpretation of contrast functions is that they are measures of non-normality [36]. A family of such measures of non-normality could be constructed using practically any functions G, and considering the difference of the expectation of G for the actual data and the expectation of G for Gaussian data. In other words, we can define a contrast function J that measures the non-normality of a zero-mean random variable y using any even, non-quadratic, sufficiently smooth function G as follows

$$J_G(y) = |E_y\{G(y)\} - E_\nu\{G(\nu)\}|^p \tag{28}$$

where ν is a standardized Gaussian random variable, y is assumed to be normalized to unit variance, and the exponent $p = 1, 2$ typically. The subscripts denote expectation with respect to y and ν. (The notation J_G should not be confused with the notation for negentropy, J.)

Clearly, J_G can be considered a generalization of (the modulus of) kurtosis. For $G(y) = y^4$, J_G becomes simply the modulus of kurtosis of y. Note that G must not be quadratic, because then J_G would be trivially zero for all distributions. Thus, it seems plausible that J_G in (28) could be a contrast function in the same way as kurtosis. The fact that J_G is indeed a contrast function in a suitable sense (locally), is shown in [61, 73]. In fact, for $p = 2$, J_G coincides with the approximation of negentropy given in (26).

In [62], the finite-sample statistical properties of the estimators based on optimizing such a general contrast function were analyzed. It was found that for a suitable choice of G, the statistical properties of the estimator (asymptotic variance and robustness) are considerably better than the properties of the cumulant-based estimators. The following choices of G were proposed:

$$G_1(u) = \log\cosh a_1 u, \quad G_2(u) = \exp(-a_2 u^2/2) \tag{29}$$

where $a_1, a_2 \geq 1$ are some suitable constants. In the lack of precise knowledge on the distributions of the independent components or on the outliers, these two functions seem to approximate reasonably well the optimal contrast function in most cases. Experimentally, it was found that especially the values $1 \leq a_1 \leq 2, a_2 = 1$ for the constants give good approximations. One reason for this is that G_1 above corresponds to the log-density of a super-gaussian distribution [123], and is therefore closely related to maximum likelihood estimation.

4.5 A unifying view on contrast functions

It is possible to give a unifying view that encompasses most of the important contrast functions for ICA.

First of all, we saw above that the principles of mutual information and maximum likelihood are essentially equivalent [26]. Second, as already discussed above, the infomax principle is equivalent to maximum likelihood estimation [23, 123]. The nonlinear PCA criteria can be shown to be equivalent to maximum likelihood estimation as well [86]. On the other hand, it was discussed above how some of the cumulant-based contrasts can be considered as approximations of mutual information. Thus it can be seen that most of the multi-unit contrast functions are, if not strictly equivalent, at least very closely related.

However, an important reservation is necessary here: for these equivalencies to be at all valid, the densities f_i used in the likelihood must be a sufficiently good approximations of the true densities of the independent components. At the minimum, we must have one bit of information on each independent component: whether it is sub- or super-Gaussian [28, 26, 73]. This information must be either available a priori or estimated from the data, see [28, 26, 96, 124]. This situation is quite different with most contrast functions based on cumulants, and the general contrast functions in Section 4.4.3, which estimate directly independent components of almost any non-Gaussian distribution.

As for the one-unit contrast functions, we have a very similar situation. Negentropy can be approximated by cumulants, or the general contrast functions in Section 4.4.3, which shows that the considered contrast functions are very closely related.

In fact, looking at the formulas for likelihood and mutual information in (13) and (16), one sees that they can be considered as sums of one-unit contrast functions plus a penalizing term that prevents the vectors $\mathbf{w}_i$ from converging to the same directions. This could be called a 'soft' form of decorrelation. Thus we see that almost all the contrast functions could be described by the single intuitive principle: *Find the most nongaussian projections, and use some (soft) decorrelation* to make sure that different independent components are found.

So, the choice of contrast function is essentially reduced the simple *choice between estimating all the independent components in parallel, or just estimating a few of them (possibly one-by-one).* This corresponds approximately to the choosing between symmetric and hierachical decorrelation, which is a choice familiar in PCA learning [114, 111].

One must also make the less important choice between cumulant-based and robust contrast functions (i.e. those based on nonquadratic functions as in (29)), but it seems that the robust contrast functions are to be preferred in most applications.

5 ALGORITHMS FOR ICA

5.1 Introduction

After choosing one of the principles of estimation for ICA discussed in Section 4, one needs a practical method for its implementation. Usually, this means that after choosing an objective function for ICA, we need to decide how to optimize it. In this Section, we shall discuss the optimization problem.

The statistical properties of the ICA method depend only on the objective functions used. Thus, the statistical properties of the objective functions were treated in Section 4. In this Section, we shall compare the different algorithms mainly on the basis of the stability, convergence speed, and memory requirements.

5.2 Preprocessing of the data

Some ICA algorithms require a preliminary sphering or whitening of the data $\mathbf{x}$, and even those algorithms that do not necessarily need sphering, often converge better with sphered data. (Recall that the data has also been assumed to be centered, i.e., made zero-mean.) Sphering means that the observed variable $\mathbf{x}$ of Eq. (11) is linearly transformed to a variable $\mathbf{v}$

$$\mathbf{v} = \mathbf{Q}\mathbf{x} \tag{30}$$

such that the covariance matrix of $\mathbf{v}$ equals unity: $E\{\mathbf{v}\mathbf{v}^T\} = \mathbf{I}$. This transformation is always possible. Indeed, it can be accomplished by classical PCA [36, 112, 110, 114]. In addition to sphering, PCA may allow us to determine the number of independent components (if $m > n$): if noise level is low, the energy of $\mathbf{x}$ is essentially concentrated on the subspace spanned by the n first principal components, with n the number of independent components in the model (11). Several methods exist for estimating the number of signals (here, independent components), see [146, 94, 144, 11]. Thus this reduction of dimension partially justifies the assumption $m = n$ that was made in Section 3, and will be retained here.

After sphering we have from (11) and (30):

$$\mathbf{v} = \mathbf{B}\mathbf{s} \tag{31}$$

where $\mathbf{B} = \mathbf{Q}\mathbf{A}$ is an *orthogonal* matrix, because

$$E\{\mathbf{v}\mathbf{v}^T\} = \mathbf{B}E\{\mathbf{s}\mathbf{s}^T\}\mathbf{B}^T = \mathbf{B}\mathbf{B}^T = \mathbf{I}$$

Recall that we assumed that the independent components s_i have unit variance. We have thus reduced the problem of finding an arbitrary matrix $\mathbf{A}$ in model (11) to the simpler problem of finding an orthogonal matrix $\mathbf{B}$. Once $\mathbf{B}$ is found, Eq. (31) is used to solve the independent components from the observed $\mathbf{v}$ by

$$\hat{\mathbf{s}} = \mathbf{B}^T\mathbf{v} \tag{32}$$

It is also worthwhile to reflect why sphering alone does not solve the separation problem. This is because sphering is only defined up to an additional rotation: if $\mathbf{Q}_1$ is a sphering matrix, then $\mathbf{Q}_2 = \mathbf{U}\mathbf{Q}_1$ is also a sphering matrix if and only if $\mathbf{U}$ is an orthogonal matrix. Therefore, we have to find the *correct* sphering matrix that equally separates the independent components. This is done by first finding any sphering matrix $\mathbf{Q}$, and later determining the appropriate orthogonal transformation from a suitable non-quadratic criterion.

In the following, we shall thus assume in certain sections that the data is sphered. For simplicity, the sphered data will be denoted by $\mathbf{x}$, and the transformed mixing matrix by $\mathbf{A}$, as in the definitions of Section 3. If an algorithm needs preliminary sphering, this is mentioned in the corresponding section. If no mention of sphering is made, none is needed.

5.3 Jutten-Hérault algorithm

The pioneering work in [80] was inspired by neural networks. Their algorithm was based on canceling the non-linear cross-correlations, see Section 4.3.3. The non-diagonal terms of the matrix $\mathbf{W}$ are updated according to

$$\Delta \mathbf{W}_{ij} \propto g_1(y_i) g_2(y_j), \text{for } i \neq j \tag{33}$$

where g_1 and g_2 are some odd non-linear functions, and the y_i are computed at every iteration as $\mathbf{y} = (\mathbf{I} + \mathbf{W})^{-1}\mathbf{x}$. The diagonal terms $\mathbf{W}_{ii}$ are set to zero. The y_i then give, after convergence, estimates of the independent components. Unfortunately, the algorithm converges only under rather severe restrictions (see [40]).

5.4 Non-linear decorrelation algorithms

Further algorithms for canceling non-linear cross-correlations were introduced independently in [34, 33, 30] and [91, 28]. Compared to the Jutten-Hérault algorithm, these algorithms reduce the computational overhead by avoiding any matrix inversions, and improve its stability. For example, the following algorithm was given in [34, 33]:

$$\Delta \mathbf{W} \propto (\mathbf{I} - g_1(\mathbf{y}) g_2(\mathbf{y}^T))\mathbf{W}, \tag{34}$$

where $\mathbf{y} = \mathbf{W}\mathbf{x}$, the non-linearities $g_1(.)$ and $g_2(.)$ are applied separately on every component of the vector $\mathbf{y}$, and the identity matrix could be replaced by any positive definite diagonal matrix. In [91, 28], the following algorithm called the EASI algorithm was introduced:

$$\Delta \mathbf{W} \propto (\mathbf{I} - \mathbf{y}\mathbf{y}^T - g(\mathbf{y})\mathbf{y}^T + \mathbf{y}g(\mathbf{y}^T))\mathbf{W}, \tag{35}$$

A principled way of choosing the non-linearities used in these learning rules is provided by the maximum likelihood (or infomax) approach as described in the next subsection.

5.5 Algorithms for maximum likelihood or infomax estimation

An important class of algorithms consists of those based on maximization of network entropy (infomax) [12], which is, under some conditions, equivalent to the maximum likelihood approach [124] (see Section 4.3.1). Usually these algorithms are based on (stochastic) gradient ascent of the objective function. For example, the following algorithm was derived in [12]:

$$\Delta \mathbf{W} \propto [\mathbf{W}^T]^{-1} - 2\tanh(\mathbf{W}\mathbf{x})\mathbf{x}^T \tag{36}$$

where the tanh function is applied separately on every component of the vector $\mathbf{W}\mathbf{x}$, as above. The tanh function is used here because it is the derivative of the log-density of the 'logistic' distribution [12]. This function works for estimation of most super-Gaussian (sparse) independent components; for sub-Gaussian independent components, other functions must be used, see e.g. [124, 26, 96]. The algorithm in Eq. (36) converges, however, very slowly, as has been noted by several researchers. The convergence may be improved by whitening the data, and especially by using the natural gradient.

The natural (or relative) gradient method simplifies the gradient method considerably, and makes it better conditioned. The principle of the natural gradient [1, 2] is based on the geometrical structure of the parameter space, and is related to the principle of relative gradient [28] that uses the Lie group structure of the ICA problem. In the case of basic ICA, both of these principles amount to multiplying the right-hand side of (36) by $\mathbf{W}^T\mathbf{W}$. Thus we obtain

$$\Delta\mathbf{W} \propto (\mathbf{I} - 2\tanh(\mathbf{y})\mathbf{y}^T)\mathbf{W} \tag{37}$$

with $\mathbf{y} = \mathbf{W}\mathbf{x}$. After this modification, the algorithm does not need sphering. Interestingly, this algorithm is a special case of the non-linear decorrelation algorithm in (34), and is closely related to the algorithm in (35).

Finally, in [124], a Newton method for maximizing the likelihood was introduced. The Newton method converges in fewer iterations, but has the drawback that a matrix inversion (at least approximate) is needed in every iteration.

5.6 Non-linear PCA algorithms

Nonlinear extensions of the well-known neural PCA algorithms [110, 114, 111] were developed in [115]. For example, in [115], the following non-linear version of a hierarchical PCA learning rule was introduced:

$$\Delta\mathbf{w}_i \propto g(y_i)\mathbf{x} - g(y_i)\sum_{j=1}^{i} g(y_j)\mathbf{w}_j \tag{38}$$

where g is a suitable non-linear scalar function. The symmetric versions of the learning rules in [114, 111] can be extended for the non-linear case in the same manner. In [82], a connection between these algorithms and non-linear versions of PCA criteria (see Section 4.3.4) were proven. In general, the introduction of non-linearities means that the learning rule uses higher-order information in the learning. Thus, the learning rules may perform something more related to the higher-order representation techniques (projection pursuit, blind deconvolution, ICA). In [84, 112], it was proven that for well-chosen non-linearities, the learning rule in (38) does indeed perform ICA, if the data is sphered (whitened). Algorithms for exactly maximizing the nonlinear PCA criteria were introduced in [113].

An interesting simplification of the non-linear PCA algorithms is the bigradient algorithm [145]. The feedback term in the learning rule (38) is here replaced by a much simpler one, giving

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \mu(t)g(\mathbf{W}(t)\mathbf{x}(t))\mathbf{x}(t)^T + \alpha(\mathbf{I} - \mathbf{W}(t)\mathbf{W}(t)^T)\mathbf{W}(t) \tag{39}$$

where $\mu(t)$ is the learning rate (step size) sequence, α is a constant on the range $[.5, 1]$, the function g is applied separately on every component of the vector $\mathbf{y} = \mathbf{W}\mathbf{x}$, and the data is assumed to be sphered. A hierarchical version of the bigradient algorithm is also possible. Due to the simplicity of the bigradient algorithm, its properties can be analyzed in more detail, as in [145] and [73].

5.7 Neural one-unit learning rules

Using the principle of stochastic gradient descent, one can derive simple algorithms from the one-unit contrast functions explained above. Let us consider first whitened data. For example, taking the instantaneous gradient of the generalized contrast function in (28) with respect to $\mathbf{w}$, and taking the normalization $\|\mathbf{w}\|^2 = 1$ into account, one obtains the following Hebbian-like learning rule

$$\Delta\mathbf{w} \propto r\mathbf{x}g(\mathbf{w}^T\mathbf{x}), \text{ normalize } \mathbf{w} \tag{40}$$

where the constant may be defined, e.g. as $r = E\{G(\mathbf{w}^T\mathbf{x})\} - E\{G(\nu)\}$. The nonlinearity g can thus be almost any nonlinear function; the important point is to estimate the multiplicative constant r in a suitable manner [73, 65]. In fact, it is enough to estimate the sign of r correctly, as shown in [73]. Such one-unit

algorithms were first introduced in [40] using kurtosis, which corresponds to taking $g(u) = u^3$. Algorithms for non-whitened data were introduced in [71, 103].

To estimate several independent components, one needs a system of several units, each of which learns according to a one-unit learning rule. The system must also contain some feedback mechanisms between the units, see e.g. [71, 73]. In [59], a special kind of feedback was developed to solve some problems of non-locality encountered with the other learning rules.

5.8 Other neural (adaptive) algorithms

Other relevant neural algorithms include

- the exploratory projection pursuit algorithms, which have also been applied to ICA [47, 49]. Due to the close connection between ICA and projection pursuit, it should not be surprising that projection pursuit algorithms can be used directly to solve the ICA problem,
- the least-squares type algorithms in [85], which are based on the non-linear PCA criteria, and
- the algorithm in [106], which is an adaptive algorithm related to higher-order cumulant tensors (see Section 4.3.5).

5.9 The FastICA algorithm

Adaptive algorithms based on stochastic gradient descent may be problematic when used in an environment where no adaptation is needed. This is the case in many practical situations. The convergence is often slow, and depends crucially on the choice of the learning rate sequence. As a remedy for this problem, one can use batch (block) algorithms based on fixed-point iteration [72, 60, 65]. In [72], a fixed-point algorithm, named FastICA, was introduced using kurtosis, and in [60, 65], the FastICA algorithm was generalized for general contrast functions. For sphered data, the one-unit FastICA algorithm has the following form:

$$\mathbf{w}(k) = E\{\mathbf{x}g(\mathbf{w}(k-1)^T\mathbf{x})\} - E\{g'(\mathbf{w}(k-1)^T\mathbf{x})\}\mathbf{w}(k-1). \tag{41}$$

where the weight vector $\mathbf{w}$ is also normalized to unit norm after every iteration, and the function g is the derivative of the function G used in the general contrast function in Eq. (28). The expectations are estimated, in practice, using sample averages over a sufficiently large sample of the input data. Units using this FastICA algorithm can then be combined, just as in the case of neural learning rules, into systems that estimate several independent components. Such systems may either estimate the independent component one-by-one using hierarchical decorrelation (deflation), or they may estimate all the independent components in parallel, with symmetric decorrelation [72, 65]. Versions of (41) that need no preliminary sphering were also introduced in [60].

The FastICA algorithm is neural in that it is parallel and distributed, but it is not adaptive. Instead of using every data point immediately for learning, FastICA uses sample averages computed over larger samples of the data. The convergence speed of the fixed-point algorithms is clearly superior to those of the more neural algorithms. Speed-up factors in the range from 10 to 100 are usually observed [48].

An interesting point proven in [67] is that when FastICA is used with symmetric decorrelation, it is essentially equivalent to a Newton method for maximum likelihood estimation. This means that FastICA is a general algorithm that can be used to optimize both one-unit and multi-unit contrast functions.

5.10 Tensor-based algorithms

A large amount of research has been done on algorithms utilizing the fourth-order cumulant tensor (see Section 4.3.5) for estimation of ICA [20, 21, 22, 29, 27, 36]. These are typically batch algorithms (non-adaptive), using such tensorial techniques as eigenmatrix decomposition, which is a generalization of eigenvalue decomposition for higher-order tensors. Such a decomposition can be performed using ordinary algorithms for eigen-value decomposition of matrices, but this requires matrices of size $m^2 \times m^2$. Since such matrices is

often too large, specialized Lanczos type algorithms of lower complexity have also been developed [20]. These algorithms often perform very efficiently on small dimensions. However, in large dimensions, the memory requirements may be prohibitive, because often the coefficients of the 4-th order tensor must be stored in memory, which requires $O(m^4)$ units of memory. The algorithms also tend to be quite complicated to program, requiring sophisticated matrix manipulations.

5.11 Weighted covariance methods

The eigenvalue decomposition of the weighted covariance matrix data, as explained in Section 4.3.6, allows the computation of the ICA estimates using standard methods of linear algebra [19] on matrices of reasonable complexity ($m \times m$). Here, the data must be sphered. This method is computationally highly efficient, but, unfortunately, it works only under the rather severe restriction that the kurtoses of the independent components are all different.

5.12 Choice of algorithm

To summarize, the choice of the ICA algorithm is basically a choice between adaptive and batch-mode (block) algorithms.

In the adaptive case, the algorithms are obtained by stochastic gradient methods. In the case where all the independent components are estimated at the same time, the most popular algorithm in this category is natural gradient ascent of likelihood, or related contrast functions, like infomax [1, 2, 12, 33, 28, 26] In the one-unit case, straightforward stochastic gradient methods give adaptive algorithms that maximize negentropy or its approximations [40, 71, 103, 73].

In the case where the computations are made in batch-mode, much more efficient algorithms are available. The tensor-based methods [29, 36] are efficient in small dimensions, but they cannot be used in larger dimensions. The FastICA algorithm, based on a fixed-point iteration, is a very efficient batch algorithm that can be used to maximize both one-unit contrast functions [72, 60, 65] and multi-unit contrast functions, including likelihood [67].

6 Noisy ICA

Finally, in this Section, we shall treat the estimation of the noisy ICA model, as in Definition 2 in Section 3. Not many such methods exist. The estimation of the noiseless model seems to be a challenging task in itself, and thus the noise is usually neglected in order to obtain tractable and simple results. Moreover, it may be unrealistic in many cases to assume that the data could be divided into signals and noise in any meaningful way.

Practically all methods taking noise explicitly into account assume that the noise is Gaussian. Thus one might use only higher-order cumulants (for example, 4th and 6th-order cumulants), which are unaffected by Gaussian noise, and then use methods not unlike those presented above. This approach was taken in [93, 150]. Note that the cumulant-based methods above used both second and fourth-order cumulants. Second-order cumulants are *not* immune to Gaussian noise. Most of the cumulant-based methods could still be modified to work in the noisy case. Their lack of robustness may be very problematic in a noisy environment, though.

Maximum likelihood estimation of the noisy ICA model has also been treated. First, one could maximize the joint likelihood of the mixing matrix and the realizations of the independent components, as in [117, 63, 31]. A more principled method would be to maximize the (marginal) likelihood of the mixing matrix, and possibly of the noise covariance, which was done in [107]. This was based on the idea of approximating the densities of the independent components as Gaussian mixture densities; the application of the EM algorithm then becomes feasible. In [15], the simpler case of discrete-valued independent components was treated. A problem with the EM algorithm is, however, that the computational complexity grows exponentially with the dimension of the data.

Perhaps the most promising approach to noisy ICA is given by bias removal techniques. This means that ordinary (noise-free) ICA methods are modified so that the bias due to noise is removed, or at least reduced. In [43], it was shown how to modify the natural gradient ascent for likelihood so as to reduce the bias. In [66], a new concept called gaussian moments was introduced to derive one-unit contrast functions and to obtain a version of the FastICA algorithm that has no asymptotic bias, i.e. is consistent even in the presence of noise. These methods can be used even in large dimensions.

7 CONCLUSIONS

This paper surveyed contrast functions and algorithms for ICA. ICA is a general concept with a wide range of applications in neural computing, signal processing, and statistics. ICA gives a representation, or transformation, of multidimensional data that seems to be well suited for subsequent information processing. This is because the components in the representation are 'as independent as possible' from each other, and at the same time 'as non-Gaussian as possible'. The transformation may also be interesting in its own right, as in blind source separation.

In the discussion of the many methods proposed for ICA, it was shown that the basic choice of the ICA method seems to reduce to two questions. First, the choice between estimating all the independent components at the same time, and estimating only a subset of them, possibly one-by-one. Most ICA research has concentrated on the first option, but in practice, it seems that the second option is very often more interesting, due to computational and other considerations. Second, one has the choice between adaptive algorithms and batch-mode (or block) algorithms. Again, most research has concentrated on the former option, although in many applications, the latter option seems to be preferable, again for computational reasons.

In spite of the large amount of research conducted on this basic problem by several authors, this area of research is by no means exhausted. It may be that the estimation methods of the very basic ICA model are so developed that the probability of new breakthroughs may not be very large. However, different extensions of the basic framework provide important directions for future research, for example:

1. Estimation of the noisy ICA model [31, 43, 63, 66, 107], as well as estimation of the model with overcomplete bases (more independent components than observed mixtures) [118, 99, 69], are basic problems that seem to require more research.

2. Methods that are tailor-made to the characteristics of a given practical application may be important in many areas. For example, in some cases it would be useful to be able to estimate a model which is similar to the ICA model, but the components are not necessarily all independent. Exact conditions that enable the estimation of the model in that case would be interesting to formulate. Steps in this direction can be found in [25, 70].

3. The problem of overlearning in ICA has been recently pointed out [76]. Avoiding and detecting overlearning is likely to be of great importance in practical applications.

4. If the $\mathbf{x}(t)$ come from a stochastic process, instead of being a sample of a random variable, blind source separation can also be accomplished by methods that use time-correlations [135, 16, 105]. Integrating this information in ICA methods may improve their performance. An important extension of ordinary ICA contrast functions for this case was introduced in [119], in which it was proposed that Kolmogorov complexity gives a meaningful extension of mutual information.

5. When ICA is used for blind separation of stochastic processes, there may also be time delays. This is the case if the signals propagate slowly from the physical sources to the sensors; because the distances between the sensors and the sources are not equal, the signals do not reach the sensors at the same time. This happens, e.g., in array processing of sound signals. A related problem in blind source separation is echos. Due to these phenomena, some kind of blind deconvolution must be made together with blind source separation, see e.g. [41, 123, 137, 149, 150, 134, 92].

6. Finally, non-linear ICA is a very important, though at the same time an almost intractable problem. In neural networks as well as in statistics, several non-linear methods have been developed [90, 52, 55, 17], and it may be possible to apply some of these to ICA. For example, in [120], the Self-Organizing Map [90] was used for non-linear ICA of sub-Gaussian independent components. This approach was generalized in [121] by using the generative topographic mapping [17]. Also the identifiability of non-linear ICA models needs further research; some results appeared in [75, 133]. Other work can be found in [39, 97].

A Definition of Cumulants

In this appendix, we present the definitions of cumulants. Consider a scalar random variable of zero mean, say x, whose characteristic function is denoted by $\hat{f}(t)$:

$$\hat{f}(t) = E\{\exp(itx)\}. \tag{42}$$

Expanding the logarithm of the characteristic function as a Taylor series, one obtains

$$\log \hat{f}(t) = \kappa_1(it) + \kappa_2(it)^2/2 + ... + \kappa(it)^r/r! + ... \tag{43}$$

where the κ_r are some constants. These constants are called the cumulants (of the distribution) of x. In particular, the first three cumulants (for zero-mean variables) have simple expressions:

$$\kappa_1 = E\{x\} = 0 \tag{44}$$
$$\kappa_2 = E\{x^2\} \tag{45}$$
$$\kappa_3 = E\{x^3\} \tag{46}$$
$$\tag{47}$$

Of particular interest for us is the fourth-order cumulant, called kurtosis, which can be expressed as [88, 109]

$$\text{kurt}\,(x) = E\{x^4\} - 3(E\{x^2\})^2 \tag{48}$$

Kurtosis can be considered a measure of the non-Gaussianity of x. For a Gaussian random variable, kurtosis is zero; it is typically positive for distributions with heavy tails and a peak at zero, and negative for flatter densities with lighter tails. Distributions of positive (resp. negative) kurtosis are thus called super-Gaussian (resp. sub-Gaussian).

Cumulants should be compared with (centered) moments. The r-th moment of x is defined as $E\{x^r\}$ [88, 109]. (For simplicity, x was here assumed to have zero mean, in which case the centered moments, or moments about the mean, equal the non-centered moments, or moments about zero). The moments may also be obtained from a Taylor expansion that is otherwise identical to the one in (43), but no logarithm is taken on the left side. Note that the first 3 moments equal the first 3 cumulants. For $r > 3$, however, this is no longer the case.

The mathematical simplicity of the cumulant-based approach in ICA is due to certain linearity properties of the cumulants [88]. For kurtosis, these can be formulated as follows. If x_1 and x_2 are two independent random variables, it holds $\text{kurt}\,(x_1 + x_2) = \text{kurt}\,(x_1) + \text{kurt}\,(x_2)$ and $\text{kurt}\,(\alpha x_1) = \alpha^4\,\text{kurt}\,(x_1)$, where α is a scalar.

Cumulants of several random variables $x_1, ..., x_m$ are defined similarly. The cross-cumulant $\text{cum}\,(x_{i_1}, x_{i_2}, .., x_{i_k})$ for any set of indices i_j is defined by the coefficient of the term $t_{i_1} t_{i_2} ... t_{i_k}$ in the Taylor expansion of the logarithm of the characteristic function $\hat{f}(t) = E\{\exp(i \sum t_j x_j)\}$ of the vector $\mathbf{x} = (x_1, ..., x_m)^T$ [88].

B Nomenclature:

PCA: Principal Component Analysis
ICA: Independent Component Analysis
pdf: probability density function
Variables and constants:
i: General-purpose index, also: imaginary unit
m: Dimension of the observed data
n: Dimension of the transformed component vector
t: Time or iteration index
x and y: General-purpose scalar random variables
y_i: Output of the i-th neuron in a neural network
α: A scalar constant
μ: Learning rate constant or sequence
All the vectors are printed in boldface lowercase letters, and are column vectors:
$\mathbf{x}$: Observed data, an m-dimensional random vector
Also: the input vector of a neural network
$\mathbf{s}$: n-dimensional random vector of transformed components s_i
$\mathbf{n}$: m-dimensional random noise vector
$\mathbf{w}$: m-dimensional constant vector
$\mathbf{w}_i$: Weight vectors of a neural network indexed by i
$\mathbf{y}$: m-dimensional general-purpose random vector
Also: the output vector of a neural network
All the matrices are printed in boldface capital letters:
$\mathbf{A}$: The constant $m \times n$ mixing matrix in the ICA model
$\mathbf{B}$: A transformed $m \times m$ mixing matrix
$\mathbf{C}$: Covariance matrix of $\mathbf{x}$, $\mathbf{C} = E\{\mathbf{x}\mathbf{x}^T\}$
$\mathbf{W}$: The weight matrix of an artificial neural network, with rows $\mathbf{w}_i^T$
Also: A general transformation matrix
Functions:
$E\{.\}$: Mathematical expectation
$f(.)$: A probability density function
$f_i(.)$: Marginal probability density functions
$\hat{f}(.)$: The characteristic function of a random variable
$g(.)$: A scalar non-linear function
$H(.)$: Differential entropy
$I(.)$: Mutual information
$J(.)$: Negentropy
$\delta(.)$: Kullback-Leibler divergence
$J_G(.)$: Generalized contrast function
$\mathbf{f}(.)$: A general transformation from R^m to R^n
$h(t)$: A FIR filter
$\kappa_i(.)$: The i-th order cumulant of a scalar random variable
kurt (.): Kurtosis, or fourth-order cumulant
cum (...): Cumulant (cross-cumulant) of several random variables
Other notation:
Δ: Change in parameter
$\propto$: Proportional to (proportionality constant may change with t)
f': First derivative of function f

References

[1] S. Amari, A. Cichocki, and H.H. Yang. A new learning algorithm for blind source separation. In *Advances in Neural Information Processing 8*, pages 757–763. MIT Press, Cambridge, MA, 1996.

[2] S.-I. Amari. Neural learning in structured parameter spaces — natural riemannian gradient. In *Advances in Neural Information Processing 9*, pages 127–133. MIT Press, Cambridge, MA, 1997.

[3] S.-I. Amari and A. Cichocki. Adaptive blind signal processing – neural network approaches. *Proceedings of the IEEE*, 9, 1998.

[4] J.J. Atick. Entropy minimization: A design principle for sensory perception? *International Journal of Neural Systems*, 3:81–90, 1992. Supp. 1992.

[5] Y. Bar-Ness. Bootstrapping adaptive interference cancellers: Some practical limitations. In *The Globecom Conf.*, pages 1251–1255, Miami, 1982. Paper F3.7.

[6] H. B. Barlow. Possible principles underlying the transformations of sensory messages. In W. A. Rosenblith, editor, *Sensory Communication*, pages 217–234. MIT Press, 1961.

[7] H. B. Barlow. Single units and sensation: A neuron doctrine for perceptual psychology? *Perception*, 1:371–394, 1972.

[8] H.B. Barlow. Unsupervised learning. *Neural Computation*, 1:295–311, 1989.

[9] H.B. Barlow. What is the computational goal of the neocortex ? In C. Koch and J.L. Davis, editors, *Large-scale neuronal theories of the brain*. MIT Press, Cambridge, MA, 1994.

[10] H.B. Barlow, T.P. Kaushal, and G.J. Mitchison. Finding minimum entropy codes. *Neural Computation*, 1:412–423, 1989.

[11] M. S. Bartlett. A note on the multiplying factors for various chi-square approximations. *J. Roy. Stat. Soc.*, 16 ser B:296–298, 1989.

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

[13] A.J. Bell and T.J. Sejnowski. Learning higher-order structure of a natural sound. *Network*, 7:261–266, 1996.

[14] A.J. Bell and T.J. Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37:3327–3338, 1997.

[15] A. Belouchrani and J.-F. Cardoso. Maximum likelihood source separation by the expectation-maximization technique: deterministic and stochastic implementation. In *Proc. NOLTA*, pages 49–53, 1995.

[16] A. Belouchrani, K. Abed Meraim, J.-F. Cardoso, and E. Moulines. A blind source separation technique based on second order statistics. *IEEE Trans. on S.P.*, 45(2):434–44, 1997.

[17] C. M. Bishop, M. Svensen, and C. K. I. Williams. GTM: The generative topographic mapping. *Neural Computation*, 10:215–234, 1998.

[18] V. Capdevielle, Ch. Serviere, and J.Lacoume. Blind separation of wide-band sources in the frequency domain. In *Proc. ICASSP-95*, volume 3, pages 2080–2083, Detroit, Michigan, USA, May 9–12 1995.

[19] J.-F. Cardoso. Source separation using higher order moments. In *Proc. ICASSP'89*, pages 2109–2112, 1989.

[20] J.-F. Cardoso. Eigen-structure of the fourth-order cumulant tensor with application to the blind source separation problem. In *Proc. ICASSP'90*, pages 2655–2658, Albuquerque, NM, USA, 1990.

[21] J.-F. Cardoso. Super-symmetric decomposition of the fourth-order cumulant tensor. blind identification of more sources than sensors. In *Proc. ICASSP'91*, pages 3109–3112, 1991.

[22] J.-F. Cardoso. Iterative techniques for blind source separation using only fourth-order cumulants. In *Proc. EUSIPCO*, pages 739–742, Brussels, Belgium, 1992.

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

[24] J.-F. Cardoso. Blind signal separation: statistical principles. *Proceedings of the IEEE*, 9(10):2009–2025, 1998.

[25] J. F. Cardoso. Multidimensional independent component analysis. In *Proc. ICASSP'98*, Seattle, WA, 1998.

[26] J. F. Cardoso. Entropic contrasts for source separation. In S. Haykin, editor, *Adaptive Unsupervised Learning*. 1999.

[27] J.-F. Cardoso and P. Comon. Independent component analysis, a survey of some algebraic methods. In *Proc. ISCAS'96*, volume 2, pages 93–96, 1996.

[28] J.-F. Cardoso and B. Hvam Laheld. Equivariant adaptive source separation. *IEEE Trans. on Signal Processing*, 44(12):3017–3030, 1996.

[29] J.-F. Cardoso and A. Souloumiac. Blind beamforming for non Gaussian signals. *IEE Proceedings-F*, 140(6):362–370, 1993.

[30] A. Cichocki, R.E. Bogner, L. Moszczynski, and K. Pope. Modified Herault-Jutten algorithms for blind separation of sources. *Digital Signal Processing*, 7:80 – 93, 1997.

[31] A. Cichocki, S. C. Douglas, and S.-I. Amari. Robust techniques for independent component analysis with noisy data. *Neurocomputing*, 22:113–129, 1998.

[32] A. Cichocki and R. Unbehauen. *Neural Networks for Signal Processing and Optimization*. Wiley, 1994.

[33] A. Cichocki and R. Unbehauen. Robust neural networks with on-line learning for blind identification and blind separation of sources. *IEEE Trans. on Circuits and Systems*, 43(11):894–906, 1996.

[34] A. Cichocki, R. Unbehauen, L. Moszczynski, and E. Rummert. A new on-line adaptive algorithm for blind separation of source signals. In *Proc. Int. Symposium on Artificial Neural Networks ISANN-94*, pages 406–411, Tainan, Taiwan, 1994.

[35] P. Comon. Blind identification in presence of noise. In *Signal Processing VI: Theories and Application (Proc. EUSIPCO)*, pages 835–838. Elsevier, 1992.

[36] P. Comon. Independent component analysis – a new concept? *Signal Processing*, 36:287–314, 1994.

[37] D. Cook, A. Buja, and J. Cabrera. Projection pursuit indexes based on orthonormal function expansions. *J. of Computational and Graphical Statistics*, 2(3):225–250, 1993.

[38] J. G. Daugman. Entropy reduction and decorrelation in visual coding by oriented neural receptive fields. *IEEE Trans. on Biomedical Engineering*, 36:107–114, 1989.

[39] G. Deco and D. Obradovic. Linear redundancy reduction learning. *Neural Networks*, 8(5):751–755, 1995.

[40] N. Delfosse and P. Loubaton. Adaptive blind separation of independent sources: a deflation approach. *Signal Processing*, 45:59–83, 1995.

[41] N. Delfosse and P. Loubaton. Adaptive blind separation of convolutive mixtures. In *Proc. ICASSP'96*, pages 2940–2943, 1996.

[42] D. Donoho. On minimum entropy deconvolution. In *Applied Time Series Analysis II*, pages 565–608. Academic Press, 1981.

[43] S.C. Douglas, A. Cichocki, , and S. Amari. A bias removal technique for blind source separation with noisy measurements. *Electronics Letters*, 34:1379–1380, 1998.

[44] D.J. Field. What is the goal of sensory coding? *Neural Computation*, 6:559–601, 1994.

[45] J. H. Friedman and J. W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Trans. of Computers*, c-23(9):881–890, 1974.

[46] J.H. Friedman. Exploratory projection pursuit. *J. of the American Statistical Association*, 82(397):249–266, 1987.

[47] C. Fyfe and R. Baddeley. Non-linear data structure extraction using simple Hebbian networks. *Biological Cybernetics*, 72:533–541, 1995.

[48] X. Giannakopoulos, J. Karhunen, and E. Oja. Experimental comparison of neural ICA algorithms. In *Proc. Int. Conf. on Artificial Neural Networks (ICANN'98)*, pages 651–656, Skövde, Sweden, 1998.

[49] M. Girolami and C. Fyfe. An extended exploratory projection pursuit network with linear and nonlinear anti-hebbian connections applied to the cocktail party problem. *Neural Networks*, 10:1607–1618, 1997.

[50] F.R. Hampel, E.M. Ronchetti, P.J. Rousseuw, and W.A. Stahel. *Robust Statistics*. Wiley, 1986.

[51] H. H. Harman. *Modern Factor Analysis*. University of Chicago Press, 2nd edition, 1967.

[52] T. Hastie and W. Stuetzle. Principal curves. *Journal of the American Statistical Association*, 84:502–516, 1989.

[53] S. Haykin, editor. *Blind Deconvolution*. Prentice-Hall, 1994.

[54] S. Haykin. *Adaptive Filter Theory*. Prentice-Hall International, 3rd edition, 1996.

[55] R. Hecht-Nielsen. Replicator neural networks for universal optimal source coding. *Science*, 269:1860–1863, 1995.

[56] P.J. Huber. *Robust Statistics*. Wiley, 1981.

[57] P.J. Huber. Projection pursuit. *The Annals of Statistics*, 13(2):435–475, 1985.

[58] J. Hurri, A. Hyvärinen, and E. Oja. Wavelets and natural image statistics. In *Proc. Scandinavian Conf. on Image Analysis '97*, Lappenranta, Finland, 1997.

[59] A. Hyvärinen. Purely local neural principal component and independent component learning. In *Proc. Int. Conf. on Artificial Neural Networks*, pages 139–144, Bochum, Germany, 1996.

[60] A. Hyvärinen. A family of fixed-point algorithms for independent component analysis. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'97)*, pages 3917–3920, Munich, Germany, 1997.

[61] A. Hyvärinen. Independent component analysis by minimization of mutual information. Technical Report A46, Helsinki University of Technology, Laboratory of Computer and Information Science, 1997.

[62] A. Hyvärinen. One-unit contrast functions for independent component analysis: A statistical analysis. In *Neural Networks for Signal Processing VII (Proc. IEEE Workshop on Neural Networks for Signal Processing)*, pages 388–397, Amelia Island, Florida, 1997.

[63] A. Hyvärinen. Independent component analysis in the presence of gaussian noise by maximizing joint likelihood. *Neurocomputing*, 22:49–67, 1998.

[64] A. Hyvärinen. New approximations of differential entropy for independent component analysis and projection pursuit. In *Advances in Neural Information Processing Systems 10*, pages 273–279. MIT Press, 1998.

[65] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 1999. To appear.

[66] A. Hyvärinen. Fast independent component analysis with noisy data using gaussian moments. In *Proc. Int. Symp. on Circuits and Systems*, Orlando, Florida, 1999. To appear.

[67] A. Hyvärinen. The fixed-point algorithm and maximum likelihood estimation for independent component analysis. *Neural Processing Letters*, 1999. To appear.

[68] A. Hyvärinen. Sparse code shrinkage: Denoising of nongaussian data by maximum likelihood estimation. *Neural Computation*, 1999. Submitted.

[69] A. Hyvärinen, R. Cristescu, and E. Oja. A fast algorithm for estimating overcomplete ICA bases for image windows. In *Proc. Int. Joint Conf. on Neural Networks*, Washington, D.C., 1999.

[70] A. Hyvärinen and P. Hoyer. Independent subspace analysis shows emergence of phase and shift invariant features from natural images. In *Proc. Int. Joint Conf. on Neural Networks*, Washington, D.C., 1999.

[71] A. Hyvärinen and E. Oja. Simple neuron models for independent component analysis. *Int. Journal of Neural Systems*, 7(6):671–687, 1996.

[72] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997.

[73] A. Hyvärinen and E. Oja. Independent component analysis by general nonlinear Hebbian-like learning rules. *Signal Processing*, 64(3):301–313, 1998.

[74] A. Hyvärinen, E. Oja, P. Hoyer, and J. Hurri. Image feature extraction by sparse coding and independent component analysis. In *Proc. Int. Conf. on Pattern Recognition (ICPR'98)*, pages 1268–1273, Brisbane, Australia, 1998.

[75] A. Hyvärinen and P. Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12(3):429–439, 1999.

[76] A. Hyvärinen, J. Särelä, and R. Vigário. Spikes and bumps: Artefacts generated by independent component analysis with insufficient sample size. In *Proc. Int. Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 425–429, Aussois, France, 1999.

[77] I.T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.

[78] M.C. Jones and R. Sibson. What is projection pursuit ? *J. of the Royal Statistical Society, ser. A*, 150:1–36, 1987.

[79] C. Jutten. *Calcul neuromimétique et traitement du signal, analyse en composantes indépendentes.* PhD thesis, INPG, Univ. Grenoble, 1987. (in French).

[80] C. Jutten and J. Herault. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24:1–10, 1991.

[81] J. Karhunen, A. Hyvärinen, R. Vigario, J. Hurri, and E. Oja. Applications of neural blind separation to signal and image processing. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'97)*, pages 131–134, Munich, Germany, 1997.

[82] J. Karhunen and J. Joutsensalo. Representation and separation of signals using nonlinear PCA type learning. *Neural Networks*, 7(1):113–127, 1994.

[83] J. Karhunen and J. Joutsensalo. Generalizations of principal component analysis, optimization problems, and neural networks. *Neural Networks*, 8(4):549–562, 1995.

[84] J. Karhunen, E. Oja, L. Wang, R. Vigario, and J. Joutsensalo. A class of neural networks for independent component analysis. *IEEE Trans. on Neural Networks*, 8(3):486–504, 1997.

[85] J. Karhunen and P. Pajunen. Blind source separation using least-squares type adaptive algorithms. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'97)*, pages 3048–3051, Munich, Germany, 1997.

[86] J. Karhunen, P. Pajunen, and E. Oja. The nonlinear PCA criterion in blind source separation: Relations with other approaches. *Neurocomputing*, 22:5–20, 1998.

[87] M. Kendall. *Multivariate Analysis.* Charles Griffin&Co., 1975.

[88] M. Kendall and A. Stuart. *The Advanced Theory of Statistics.* Charles Griffin & Company, 1958.

[89] K. Kiviluoto and E. Oja. Independent component analysis for parallel financial time series. In *Proc. ICONIP'98*, volume 2, pages 895–898, Tokyo, Japan, 1998.

[90] T. Kohonen. *Self-Organizing Maps.* Springer-Verlag, Berlin, Heidelberg, New York, 1995.

[91] Beate Laheld and Jean-François Cardoso. Adaptive source separation with uniform performance. In *Proc. EUSIPCO*, pages 183–186, Edinburgh, 1994.

[92] R. H. Lambert. *Multichannel Blind Deconvolution: FIR Matrix Algebra and Separation of Multipath Mixtures.* PhD thesis, Univ. of Southern California, 1996.

[93] L. De Lathauwer, B. De Moor, and J. Vandewalle. A technique for higher-order-only blind source separation. In *Proc. ICONIP*, Hong Kong, 1996.

[94] D. N. Lawley. Test of significance of the latent roots of the covariance and correlation matrices. *Biometrica*, 43:128–136, 1956.

[95] T.-W. Lee, M. Girolami, A.J. Bell, and T.J. Sejnowski. A unifying information-theoretic framework for independent component analysis. *International Journal on Mathematical and Computer Models*, 1999. To appear.

[96] T.-W. Lee, M. Girolami, and T. J. Sejnowski. Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources. *Neural Computation*, pages 609–633, 1998. 11.

[97] T-W. Lee, B.U. Koehler, and R. Orglmeister. Blind source separation of nonlinear mixing models. In *Neural networks for Signal Processing VII*, pages 406–415, 1997.

[98] M. Lewicki and B. Olshausen. Inferring sparse, overcomplete image codes using an efficient coding framework. In *Advances in Neural Information Processing 10 (Proc. NIPS*97)*, pages 815–821. MIT Press, 1998.

[99] M. Lewicki and T. J. Sejnowski. Learing overcomplete representations. In *Advances in Neural Information Processing 10 (Proc. NIPS*97)*, pages 556–562. MIT Press, 1998.

[100] U. Lindgren, T. Wigren, and H. Broman. On local convergence of a class of blind separation algorithms. *IEEE Trans. on Signal Processing*, 43:3054–3058, 1995.

[101] S. Makeig, A.J. Bell, T.-P. Jung, and T.-J. Sejnowski. Independent component analysis of electroencephalographic data. In *Advances in Neural Information PRocessing Systems 8*, pages 145–151. MIT Press, 1996.

[102] S. G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on PAMI*, 11:674–693, 1989.

[103] Z. Malouche and O. Macchi. Extended anti-Hebbian adaptation for unsupervised source extraction. In *Proc. ICASSP'96*, pages 1664–1667, Atlanta, Georgia, 1996.

[104] M. McKeown, S. Makeig, S. Brown, T.-P. Jung, S. Kindermann, A.J. Bell, V. Iragui, and T. Sejnowski. Blind separation of functional magnetic resonance imaging (fMRI) data. *Human Brain Mapping*, 6(5-6):368–372, 1998.

[105] L. Molgedey and H. G. Schuster. Separation of a mixture of independent signals using time delayed correlations. *Phys. Rev. Lett.*, 72:3634–3636, 1994.

[106] E. Moreau and O. Macchi. New self-adaptive algorithms for source separation based on contrast functions. In *Proc. IEEE Signal Processing Workshop on Higher Order Statistics*, pages 215–219, Lake Tahoe, USA, June 1993.

[107] E. Moulines, J.-F. Cardoso, and E. Gassiat. Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'97)*, pages 3617–3620, Munich, Germany, 1997.

[108] J.-P. Nadal and N. Parga. Non-linear neurons in the low noise limit: a factorial code maximizes information transfer. *Network*, 5:565–581, 1994.

[109] C. Nikias and J. Mendel. Signal processing with higher-order spectra. *IEEE Signal Processing Magazine*, pages 10–37, July 1993.

[110] E. Oja. A simplified neuron model as a principal component analyzer. *J. of Mathematical Biology*, 15:267–273, 1982.

[111] E. Oja. Neural networks, principal components, and subspaces. *Int. J. on Neural Systems*, 1:61–68, 1989.

[112] E. Oja. The nonlinear PCA learning rule in independent component analysis. *Neurocomputing*, 17(1):25–46, 1997.

[113] E. Oja. Nonlinear PCA criterion and maximum likelihood in independent component analysis. In *Proc. Int. Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 143–148, Aussois, France, 1999.

[114] E. Oja and J. Karhunen. On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix. *Journal of Math. Analysis and Applications*, 106:69–84, 1985.

[115] E. Oja, H. Ogawa, and J. Wangviwattana. Learning in nonlinear constrained Hebbian networks. In T. Kohonen et al., editor, *Artificial Neural Networks, Proc. ICANN'91*, pages 385–390, Espoo, Finland, 1991. North-Holland, Amsterdam.

[116] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[117] B. A. Olshausen and D. J. Field. Natural image statistics and efficient coding. *Network*, 7(2):333–340, May 1996.

[118] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37:3311–3325, 1997.

[119] P. Pajunen. Blind source separation using algorithmic information theory. *Neurocomputing*, 22:35–48, 1998.

[120] P. Pajunen, A. Hyvärinen, and J. Karhunen. Nonlinear blind source separation by self-organizing maps. In *Proc. Int. Conf. on Neural Information Processing*, pages 1207–1210, Hong Kong, 1996.

[121] P. Pajunen and J. Karhunen. A maximum likelihood approach to nonlinear blind source separation. In *Proceedings of the 1997 Int. Conf. on Artificial Neural Networks (ICANN'97)*, pages 541–546, Lausanne, Switzerland, 1997.

[122] Athanasios Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, 3rd edition, 1991.

[123] B.A. Pearlmutter and L.C. Parra. A context-sensitive generalization of ICA. In *Proc. ICONIP'96*, pages 151–157, Hong Kong, 1996.

[124] D.-T. Pham, P. Garrat, and C. Jutten. Separation of a mixture of independent sources through a maximum likelihood approach. In *Proc. EUSIPCO*, pages 771–774, 1992.

[125] T. Ristaniemi and J. Joutsensalo. On the performance of blind source separation in CDMA downlink. In *Proc. Int. Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 437–441, Aussois, France, 1999.

[126] Y. Sato. A method for self-recovering equalization for multilevel amplitude-modulation system. *IEEE Trans. on Communications*, 23:679–682, 1975.

[127] M. Schervish. *Theory of Statistics.* Springer, 1995.

[128] J. Schmidhuber, M. Eldracher, and B. Foltin. Semilinear predictability minimization produces well-known feature detectors. *Neural Computation*, 8:773–786, 1996.

[129] O. Shalvi and E. Weinstein. New criteria for blind deconvolution of nonminimum phase systems (channels). *IEEE Trans. on Information Theory*, 36(2):312–321, 1990.

[130] O. Shalvi and E. Weinstein. Super-exponential methods for blind deconvolution. *IEEE Trans. on Information Theory*, 39(2):504:519, 1993.

[131] E. Sorouchyari. Blind separation of sources, Part III: Stability analysis. *Signal Processing*, 24:21–29, 1991.

[132] J. Sun. Some practical aspects of exploratory projection pursuit. *SIAM J. of Sci. Comput.*, 14:68–80, 1993.

[133] A. Taleb and C. Jutten. Nonlinear source separation: The postlinear mixtures. In *Proc. European Symposium on Artificial Neural Networks*, pages 279–284, Bruges, Belgium, 1997.

[134] H.-L. Nguyen Thi and C. Jutten. Blind source separation for convolutive mixtures. *Signal Processing*, 45:209–229, 1995.

[135] L. Tong, R.-W. Liu, V.C. Soon, and Y.-F. Huang. Indeterminacy and identifiability of blind identification. *IEEE Trans. on Circuits and Systems*, 38, 1991.

[136] L. Tong, V. Soo, R. Liu, and Y. Huang. Amuse: a new blind identification algorithm. In *Proc. ISCAS*, New Orleans, USA, 1990.

[137] K. Torkkola. Blind separation of delayed sources based on information maximization. In *Proc. ICASSP'96*, pages 3509–3512, Atlanta, Georgia, 1996.

[138] J. H. van Hateren and D. L. Ruderman. Independent component analysis of natural image sequences yields spatiotemporal filters similar to simple cells in primary visual cortex. *Proc. Royal Society ser. B*, 265:2315–2320, 1998.

[139] J. H. van Hateren and A. van der Schaaf. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc. Royal Society ser. B*, 265:359–366, 1998.

[140] R. Vigário. Extraction of ocular artifacts from EEG using independent component analysis. *Electroenceph. clin. Neurophysiol.*, 103(3):395–404, 1997.

[141] R. Vigário, V. Jousmäki, M. Hämäläinen, R. Hari, and E. Oja. Independent component analysis for identification of artifacts in magnetoencephalographic recordings. In *Advances in Neural Information Processing 10 (Proc. NIPS'97)*, pages 229–235, Cambridge, MA, 1998. MIT Press.

[142] R. Vigário, J. Särelä, and E. Oja. Independent component analysis in wave decomposition of auditory evoked fields. In *Proc. Int. Conf. on Artificial Neural Networks (ICANN'98)*, pages 287–292, Skövde, Sweden, 1998.

[143] R. Vigário, J. Särelä, V. Jousmäki, and E. Oja. Independent component analysis in decomposition of auditory and somatosensory evoked fields. In *Proc. Int. Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 167–172, Aussois, France, 1999.

[144] H. Wang and M. Kaveh. Coherent signal-subspace processing for the detection and estimation of angles of arrival of multiple wide-band sources. *IEEE Trans. on ASSP*, 33:823–831, 1985.

[145] L.-Y. Wang and J. Karhunen. A unified neural bigradient algorithm for robust PCA and MCA. *Int. J. of Neural Systems*, 7(1):53–67, 1996.

[146] M. Wax and T. Kailath. Detection of signals by information-theoretic criteria. *IEEE Trans. on ASSP*, 33:387–392, 1985.

[147] E. Weinstein, M. Feder, and A. V. Oppenheim. Multi-channel signal separation by decorrelation. *IEEE Trans. on SAP*, 1:405–413, 1993.

[148] R. A. Wiggins. Minimum entropy deconvolution. *Geoexploration*, 16:12–35, 1978.

[149] D. Yellin and E. Weinstein. Criteria for multichannel signal separation. *IEEE Trans. on Signal Processing*, 42:2158–2167, 1994.

[150] D. Yellin and E. Weinstein. Multichannel signal separation: Methods and analysis. *IEEE Trans. on Signal Processing*, 44:106–118, 1996.