CHAPTER

Sparse component analysis 10

R. Gribonval and M. Zibulevsky

10.1 INTRODUCTION

Sparse decomposition techniques for signals and images underwent considerable development during the flourishing of wavelet-based compression and denoising methods [75] in the early 1990s. Some ten years elapsed before these techniques began to be fully exploited for blind source separation [72,106,66,70,116,61,15,111]. Their main impact is that they provide a relatively simple framework for separating a number of sources exceeding the number of observed mixtures. Also they greatly improve quality of separation in the case of square mixing matrix.

The underlying assumption is essentially geometrical [98], differing considerably from the traditional assumption of independent sources [19]. This can be illustrated using the simplified model of speech discussed by Van Hulle [106]. One wishes to identify the mixing matrix and extract the sources based on the observed mixture:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{b}(t), \qquad 1 \le t \le T. \tag{10.1}$$

For this purpose, one assumes that at each point t (variable designating a time instant for temporal signals, a pixel coordinate for images, etc.), a single source is significantly more active than the others. If $\Lambda_n \subset \{1, \ldots, T\}$ denotes the set of points where the source with index n is most active (we will refer to the temporal *support* of source n), then for all $t \in \Lambda_n$ one has by definition $|s_n(t)| \gg |s_m(t)|$ for $m \neq n$, and therefore:

$$\mathbf{x}(t) \approx s_n(t)\mathbf{A}_n, \qquad t \in \Lambda_n \tag{10.2}$$

where $\mathbf{A}_n = (A_{pn})_{1 \le p \le P}$ is the (unknown) n-th column of the mixing matrix. It follows that the set of points $\{\mathbf{x}(t) \in \mathbb{C}^P, t \in \Lambda_n\}$ is more or less aligned along the straight line passing through the origin and directed by vector $\mathbf{A}_n$. As shown in Fig. 10.1(a) for a stereophonic mixture ($P = 2$) of three audio sources ($N = 3$), this alignment can be observed in practice on the *scatter plot* $\{\mathbf{x}(t) \in \mathbb{C}^P, 1 \le t \le T\}$. In this figure in dimension $P = 2$, the scatter plot is the collection of points representing the pairs of values $(x_1(t), x_2(t))$ observed for all T samples of the mixture signal.

Handbook of Blind Source Separation: Independent Component Analysis and Applications

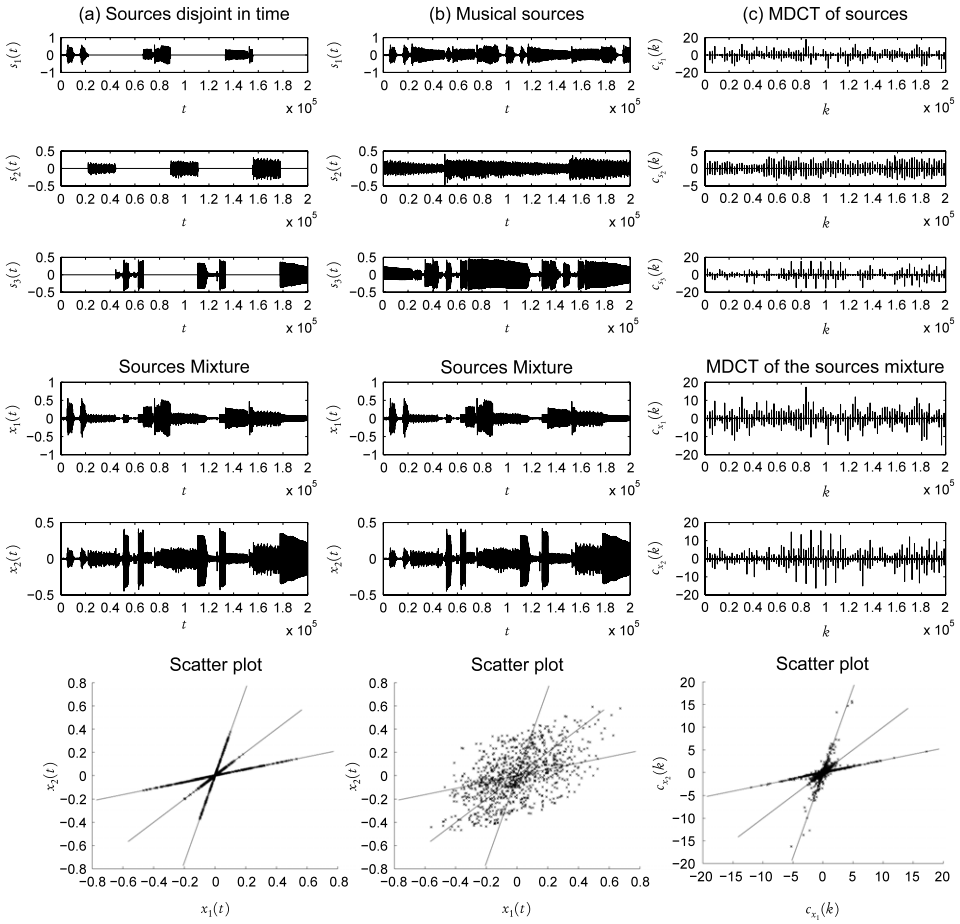


FIGURE 10.1

"Geometric" principle of source separation by sparse decomposition. Sources (top), the corresponding mixtures (middle) and the resulting scatter plots (bottom). (a) – For sources with disjoint temporal supports, the temporal scatter plot shows alignments along the columns of the mixing matrix. (b) – For more complex signals such as musical signals (center), the temporal scatter plot is not legible. (c) – An appropriate orthogonal time-frequency transform (MDCT) applied to the same musical signals (right) results in a scatter plot in which the column directions of the mixing matrix are clearly visible.

It is thus conceivable to estimate the mixing matrix $\widehat{\mathbf{A}}$ using a *clustering* algorithm [110] – the scatter plot is separated into N clusters of points $\{\mathbf{x}(t), t \in \widehat{\Lambda}_n\}$ – and to estimate the direction $\widehat{\mathbf{A}}_n$ of each cluster. If the mixture is (over)-determined ($N \leq P$), the sources can be estimated with the pseudo-inverse [47] of the estimated mixing matrix $\widehat{\mathbf{s}}(t) := \widehat{\mathbf{A}}^{\dagger}\mathbf{x}(t)$, but even in the case of an *under-determined* mixture ($N > P$), by assuming the sources

have disjoint supports, they can be estimated using least squares[1] from the obtained clusters:

$$\widehat{s}_n(t) := \begin{cases} \langle \mathbf{x}(t), \widehat{\mathbf{A}}_n \rangle / \|\widehat{\mathbf{A}}_n\|_2^2 & \text{if } t \in \widehat{\Lambda}_n, \\ 0 & \text{if not.} \end{cases}$$

Although simple and attractive, Van Hulle's technique generally cannot be applied in its original form, as illustrated in Fig. 10.1(b), which shows the temporal scatter plot of a stereophonic mixture ($P = 2$) of $N = 3$ musical sources. It is clear that the sources cannot be considered to have disjoint temporal supports, and the temporal scatter plot does not allow visual distinguishing of the column directions of the mixing matrix.

This is where sparse signal representations come into play. Figure 10.1(c) shows the coefficients $c_{s_n}(k)$ of an orthogonal time-frequency transform applied to the sources (MDCT [87]), the coefficients $c_{x_p}(k)$ of the mixture, and finally the time-frequency scatter plot of points $(c_{x_1}(k), c_{x_2}(k))$, $1 \leq k \leq T$. On the time-frequency scatter plot, one can once again observe the directions of columns of the mixing matrix and apply the strategy described previously for estimating $\mathbf{A}$. The MDCT coefficients for the original sources on the same figure explain this "miracle": the coefficients of the MDCTs for the different sources have (almost) disjoint supports, given that for each source only a small number of MDCT coefficients are of significant amplitude. The above-mentioned approach can once again be used to estimate $\widehat{\mathbf{A}}$ and the coefficients of the time-frequency transform applied to the sources. The sources themselves are estimated by applying an inverse transform.

Figure 10.2 shows the structure of sparsity-based source separation methods, for which we have just introduced the main ingredients:

1. A step for changing the representation of the observed mixture, to make the supports of the source coefficients in the new representation as disjoint as possible. For this purpose, *joint sparse representation* techniques are used.
2. The column directions of the mixing matrix are estimated using the mixture representation. In addition to classical ICA algorithms, *clustering algorithms* can be applied to the *scatter plot*.
3. Finally, *estimating the (sparse) source representations* makes it possible to *reconstruct* the sources.

Chapter outline

The rest of this chapter provides a detailed discussion of the main approaches for each step of sparse separation. Sparse signal representation and dictionaries are introduced in section 10.2 with relevant examples. The formalism and main algorithms for calculating joint sparse representations of mixtures $\mathbf{x}(t)$ are described in section 10.3, which

[1] The notation $\langle \cdot, \cdot \rangle$ is used for the scalar product between two vectors, e.g. for column vectors $\langle \mathbf{x}(t), \widehat{\mathbf{A}}_n \rangle := \mathbf{x}(t)\widehat{\mathbf{A}}_n^{\mathrm{H}} = \sum_{p=1}^{P} x_p(t)\widehat{A}_{pn}^*$ and for row vectors $\langle s_n, s_{n'} \rangle = \sum_{t=1}^{T} s_n(t)s_{n'}^*(t)$.

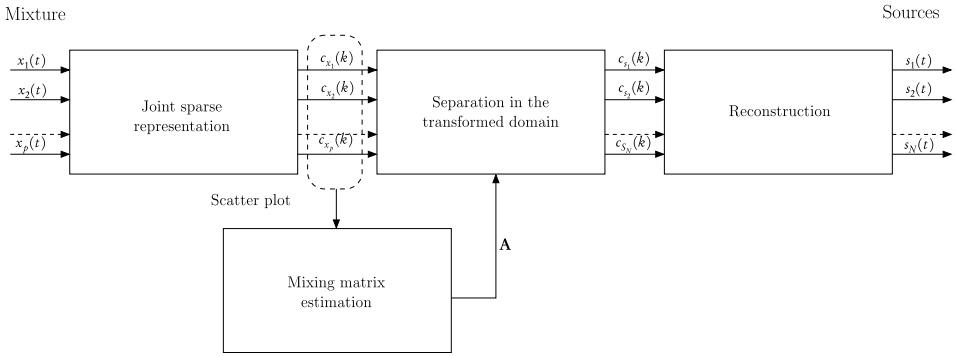


FIGURE 10.2

Block diagram of sparsity-based source separation methods.

concludes by reviewing the main characteristics, advantages and disadvantages of these algorithms, the scope of which extends well beyond source separation. Section 10.4 examines actually using the sparsity of joint mixture representations for the second step of source separation, in which the mixing matrix is estimated from scatter plots, while section 10.5 discusses the Relative Newton framework for the special case of a square mixing matrix. Finally, section 10.6 focuses on the algorithms for the actual separation step in the "transform" domain, to estimate the sources once the mixing matrix has been estimated. The possible choices for the different steps of sparse source separation are summarized in section 10.7, which also discusses the factors that may determine these choices. To conclude, in section 10.8 we assess the new directions and challenges opened up by sparse source separation, such as using new diversity factors between sources, blind estimation of suitable dictionaries and separation of under-determined convolutive mixtures.

10.2 SPARSE SIGNAL REPRESENTATIONS

The basic assumption underlying all sparse source separation methods is that each source can be (approximately) represented by a linear combination of a few elementary signals φ_k, known as *atoms*[2] that are often assumed to be of unit energy. Specifically it is assumed that each (unknown) source signal can be written:

$$s(t) = \sum_{k=1}^{K} c(k)\, \varphi_k(t) \tag{10.3}$$

[2]Although the notations φ (or ψ) and $\mathbf{\Phi}$ (or $\mathbf{\Psi}$) are used elsewhere in this book to denote the marginal score functions (or the joint score function) of the sources and the first (or second) characteristic function, they will be used in this chapter to denote atoms with regard to signal synthesis and analysis and the corresponding dictionaries/matrices.

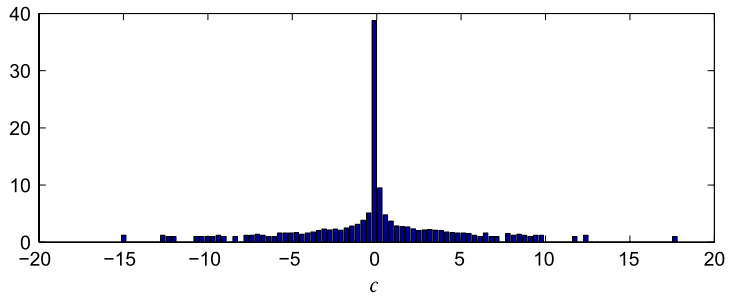


FIGURE 10.3

Histogram of MDCT coefficients of the first source in the mixture of musical signals in Fig. 10.1(c).

where only few entries $c(k)$ of the vector $\mathbf{c}$ are significant, most of them being negligible. In the case of Van Hulle's simplified speech model [106], the considered atoms are simply Dirac functions, $\varphi_k(t) = \delta(t-k)$, i.e. time samples. Many audio signal processing applications involve the short-term Fourier transform (STFT, also called Gabor transform or spectrogram) [75] where the index $k = (n, f)$ denotes a time-frequency point, $c(k)$ is the value of the STFT applied to the signal s at this point, and (10.3) corresponds to the reconstruction formula for s based on its STFT using the *overlap-add* (OLA) method [89]. It is possible to use other families of atoms known as *dictionaries*, which are better suited to representing a given type of signal. In our context, the sparseness of a signal's representation in a dictionary determines the dictionary's relevance.

10.2.1 Basic principles of sparsity

Informally, a set of coefficients $c(k)$ is considered sparse if most of the coefficients are zero or very small and only a few of them are significant. In statistical (or probabilistic) terms, coefficients $c(k)$ are said to have a sparse or super-Gaussian distribution if their histogram (or probability density) has a strong peak at the origin with heavy tails, as shown in Fig. 10.3.

The most common model of sparse distribution is the family of generalized Gaussian distributions:

$$p_c(c) \propto \exp(-\alpha |c|^\tau) \tag{10.4}$$

for $0 < \tau \leq 1$, where the $\tau = 1$ case corresponds to the Laplace distribution. Certain Bayesian source separation techniques that assume sparsity use Student-t distributions to exploit their analytical expression in terms of the Gaussian Mixture Model [40]. For the same reasons, the bi-Gaussian model, where one Gaussian distribution has low variance and the other high variance, is often used as well [30].

For the generalized Gaussian model, the set of coefficients $\mathbf{c} = \{c(k)\}_{k=1}^{K}$, assumed statistically independent, has a distribution of $p_{\mathbf{c}}(\mathbf{c}) \propto \exp(-\alpha ||\mathbf{c}||_\tau^\tau)$ where $||\mathbf{c}||_\tau$ denotes

the ℓ^τ "norm"[3]:

$$\|\mathbf{c}\|_\tau := \left(\sum_{k=1}^{K} |c(k)|^\tau\right)^{1/\tau}. \tag{10.5}$$

The likelihood of a set of coefficients thus increases as its ℓ^τ norm decreases. The norm $\|\mathbf{c}\|_\tau^\tau$ can then be used to quantify the sparsity of a representation $\mathbf{c}$, which is the approach taken in the rest of this chapter.

10.2.2 Dictionaries

In general, one considers complete dictionaries, that is to say dictionaries that allow the reconstruction of any signal $s \in \mathbb{C}^T$; in other words, the space generated by the atoms of the dictionary is $\mathbb{C}^T$ in its entirety. Limiting the approach presented in this chapter to the case of real signals poses no particular problems. With $\mathbf{\Phi}$ denoting the $K \times T$ matrix whose K rows consist of atoms $\varphi_k(t), 1 \leq t \leq T$, $k = 1, \ldots K$, the sparse model (10.3) of sources can be written in matrix form[4]:

$$s = c_s \cdot \mathbf{\Phi} \tag{10.6}$$

where $c_s = (c_s(k))_{k=1}^K$ is a row vector of K coefficients.

If the selected dictionary is a basis, $K = T$ and a one-to-one correspondence exists between the source signal s and its coefficients $c_s = s \cdot \mathbf{\Phi}^{-1}$ ($c_s = s \cdot \mathbf{\Phi}^{\mathrm{H}}$ for an orthonormal basis). This is the case for a dictionary corresponding to an orthogonal basis of discrete wavelets, or the orthogonal discrete Fourier basis [75], or a MDCT (*Modified Discrete Cosine Transform* [87]) type basis.

By contrast, if the dictionary is redundant ($K > T$), there is an infinite number of possible coefficient sets for reconstructing each source. This is the case for a Gabor dictionary, comprising atoms expressed as

$$\varphi_{n,f}(t) = w(t - nT)\exp(2j\pi f t) \tag{10.7}$$

– where w is called an analysis window – and used to compute the STFT of a signal as $STFT_s(n, f) = \langle s, \varphi_{n,f} \rangle$. The construction and study of redundant dictionaries for sparse signal representation is based on the idea that, among the infinitely many possibilities, one can choose a representation particularly well suited to a given task.

[3] Strictly speaking, this is a norm only for $1 \leq \tau \leq \infty$, with the usual modification $\|\mathbf{c}\|_\infty = \sup_k |c(k)|$. We will also use the term "norm", albeit improperly, for $0 < \tau < 1$. For $\tau = 0$, with the convention $c^0 = 1$ if $c > 0$ and $0^0 = 0$, the ℓ^0 norm is the number of non-zero coefficients of $\mathbf{c}$.

[4] With the usual source separation notation, the mixing matrix $\mathbf{A}$ acts to the left on the matrix of source signals $\mathbf{s}$, which is why the dictionary matrix $\mathbf{\Phi}$ must act *to the right* on the coefficients c_s for signal synthesis. This notation will be used throughout this chapter.

Since the early 1990s, building dictionaries has been the focus of several harmonic analysis studies, still being pursued today. This has led to the construction of several wavelet families as well as libraries of orthogonal wavelet packet bases or local trigonometric functions [75] and other more sophisticated constructions (curvelets, etc. [95]) too numerous to cite here. Sparse coding [41,70,69,1] is closer to the source separation problem and also aims at the learning of a dictionary (redundant or non-redundant) maximizing the sparse representation of a training data set.

10.2.3 Linear transforms

In a given dictionary $\mathbf{\Phi}$, the traditional approach for calculating a representation of s consists of calculating a linear *transform*:

$$c_s^{\mathbf{\Psi}} := s \cdot \mathbf{\Psi}, \tag{10.8}$$

and the calculated coefficients are thus the values of the correlations

$$c_s^{\mathbf{\Psi}}(k) = \langle s, \psi_k \rangle \tag{10.9}$$

of s with a set of dual atoms $\psi_k(t)$, which form the columns of the matrix $\mathbf{\Psi}$. One can thus consider the atoms $\varphi_k(t)$ as related to signal synthesis, as opposed to the atoms related to signal analysis $\psi_k(t)$.

For the transformation of a signal s (analysis, (10.8)), followed by its reconstruction (synthesis, (10.6)), the transform must satisfy $s \cdot \mathbf{\Psi\Phi} = s$ for every signal s to result in perfect reconstruction, i.e. $\mathbf{\Psi\Phi} = \mathbf{I}$ ($\mathbf{I}$ is the identity matrix, here $T \times T$ in size). For example, in the case of a Gabor dictionary (10.7), a set of dual atoms compatible with the reconstruction condition (10.8) takes the form:

$$\psi_{n,f}(t) = w'(t - nT)\exp(2j\pi f t), \tag{10.10}$$

where the analysis window $w'(t)$ and the reconstruction window $w(t)$ must satisfy the condition $\sum_n w'(t - nT)w(t - nT) = 1$ for every t. Therefore, analyzing a signal with the analysis atoms $\psi_{n,f}$ is akin to calculating its STFT with the window w', whereas reconstruction with the synthesis atoms $\varphi_{n,f}$ corresponds to *overlap-add* reconstruction based on the STFT [89].

When the dictionary $\mathbf{\Phi}$ is a basis for the signal space, as is the discrete orthonormal Fourier basis, or a bi-orthogonal wavelet basis [75], the only transform $\mathbf{\Psi}$ allowing perfect reconstruction is $\mathbf{\Psi} = \mathbf{\Phi}^{-1}$ (a fast Fourier transform or a fast wavelet transform correspondingly).

For these types of non-redundant dictionaries that correspond to "critically sampled" transforms, the relation $\mathbf{\Phi\Psi} = \mathbf{I}$ also holds; that is, synthesis of a signal based on a representation c_s, followed by analysis of the signal, makes it possible to find the exact coefficients used. This does not hold true in the case of redundant linear transforms, such as the STFT or the continuous wavelet transform.

In the case of any redundant dictionary, there is an infinite number of "dual" matrices $\mathbf{\Psi}$ that satisfy the reconstruction condition $\mathbf{\Psi\Phi} = \mathbf{I}$, but none satisfies the identity $\mathbf{\Phi\Psi} = \mathbf{I}$. The pseudo-inverse [47] $\mathbf{\Psi} = \mathbf{\Phi}^{\dagger} = (\mathbf{\Phi}^{\mathrm{H}}\mathbf{\Phi})^{-1}\mathbf{\Phi}^{\mathrm{H}}$ of $\mathbf{\Phi}$ offers a specific choice of linear transform, providing the least squares solution to the representation problem $s = c_s\mathbf{\Phi}$:

$$c_s^{\mathbf{\Phi}^{\dagger}} = \arg \min_{c_s | c_s\mathbf{\Phi}=s} \sum_{k=1}^{K} |c_s(k)|^2 . \tag{10.11}$$

10.2.4 Adaptive representations

The reason for using a redundant dictionary is to find a particularly sparse representation from among the infinite possibilities for a given signal. The dictionary's redundancy is aimed at offering a broad range of atoms likely to represent the typical signal structures in a suitable way, so that the signal can be approximated by a linear combination of a small, carefully selected atom set from the dictionary.

Paradoxically, the more redundant the dictionary, the higher the number of non-zero coefficients in a representation obtained by a linear transform. Thus, for the STFT associated with a Gabor dictionary, redundancy increases with overlap between adjacent windows (which increases the number of analysis frames) and with *zero-padding* (which increases the number of discrete analysis frequencies), but since a given atom's "neighboring" atoms in time-frequency are highly correlated to it, the number of non-negligible coefficients observed during analysis of a Gabor atom (10.7) also increases with the transform's redundancy.

To fully leverage the potential of redundant dictionaries and obtain truly sparse signal representation, linear transforms must be replaced by adaptive techniques. The principle of these techniques, which emerged in the 1990s, is to select a limited subset Λ of atoms from the dictionary based on the analyzed signal. Consequently, only the selected atoms are assigned a non-zero coefficient, resulting in a sparse representation or a sparse approximation of the signal. The main techniques to obtain adaptive signal representation are described in the following section, in the slightly more general context of joint sparse representation of mixtures, which is the first step in sparse source separation.

10.3 JOINT SPARSE REPRESENTATION OF MIXTURES

The first step in a system of sparse source separation (see Fig. 10.2) consists in calculating a joint sparse representation of the observed mixture $\mathbf{x}$, to obtain a representation that facilitates both the estimation of the mixing matrix and that of the sources by clustering algorithms. While this type of representation can be obtained with linear transforms (STFT, wavelet transforms, etc.), greater sparsity may be achieved by using algorithms for joint adaptive decomposition of the mixture.

10.3.1 Principle

An approximate sparse representation of a mixture $\mathbf{x}$ takes the form $\mathbf{x} \approx C_{\mathbf{x}}\mathbf{\Phi}$, where:

$$C_{\mathbf{x}} = \begin{bmatrix} c_{x_1} \\ \dots \\ c_{x_P} \end{bmatrix} = \begin{bmatrix} C_{\mathbf{x}}(1) & \dots & C_{\mathbf{x}}(K) \end{bmatrix} \tag{10.12}$$

is a matrix of dimensions $P \times K$ in which each row c_{x_p} is a row vector of K coefficients $c_{x_p}(k)$ representing one of the P channels of the mixture. Calculating $C_{\mathbf{x}}$ for a mixture $\mathbf{x} \approx \mathbf{A}\mathbf{s}$ is aimed at reducing the problem to a separation problem $C_{\mathbf{x}} \approx \mathbf{A}C_{\mathbf{s}}$, which is expected to be easier to solve if $C_{\mathbf{s}}$ is sparse.

By combining the sparse model of sources (see (10.3) or (10.6)) with the noisy instantaneous linear mixture model (10.1), one obtains a global sparse model of the mixture:

$$\mathbf{x} = \mathbf{A}C_{\mathbf{s}}\mathbf{\Phi} + \mathbf{b}, \qquad \text{where} \qquad C_{\mathbf{s}} = \begin{bmatrix} c_{s_1} \\ \dots \\ c_{s_N} \end{bmatrix} = \begin{bmatrix} C_{\mathbf{s}}(1) & \dots & C_{\mathbf{s}}(K) \end{bmatrix} \tag{10.13}$$

is an $N \times K$ matrix in which each row c_{s_n} is a sparse vector consisting of K coefficients $c_{s_n}(k)$ representing one of the N sources, and each column $C_{\mathbf{s}}(k)$, $1 \leq k \leq K$, indicates the degree of "activity" of the different sources for a given atom. If a representation $C_{\mathbf{s}}$ of the sources exists where they have disjoint supports (each atom only activated for one source at most), then $C_{\mathbf{x}} \coloneqq \mathbf{A}C_{\mathbf{s}}$ is an admissible (approximate) representation of the (noisy) mixture $\mathbf{x}$ and should enable its separation via the scatter plot.

When $\mathbf{\Phi}$ is an orthonormal basis such as the MDCT, a unique representation exists for the mixture (and the sources) and can thus be calculated simply by inverse linear transform, necessarily satisfying $C_{\mathbf{x}} \approx \mathbf{A}C_{\mathbf{s}}$. We saw an example of this with the musical sources, where the scatter plot of the columns of the representation $C_{\mathbf{x}}$ obtained by linear transform with an orthonormal MDCT-type basis $\mathbf{\Phi}$ clearly highlighted the column directions of the mixing matrix (see Fig. 10.1(c)), making it possible to pursue the separation. By contrast, when $\mathbf{\Phi}$ is a redundant dictionary, the calculated representation $C_{\mathbf{x}}$ depends on the chosen algorithm and does not necessarily satisfy the identity $C_{\mathbf{x}} \approx \mathbf{A}C_{\mathbf{s}}$, which is crucial for the subsequent steps of sparse separation.

Selecting a relevant joint sparse approximation algorithm for the mixture thus depends on the algorithm's ability to provide a representation that satisfies the approximation $C_{\mathbf{x}} \approx \mathbf{A}C_{\mathbf{s}}$ with sufficient accuracy, provided the sources allow a representation $C_{\mathbf{s}}$ with sufficiently disjoint supports. This important point will be discussed for each of the joint sparse mixture approximation algorithms described in this section together with their main properties.

10.3.2 Linear transforms

The simplest joint representation of a mixture involves applying the same linear transform (e.g. STFT) to all channels. Formally, by juxtaposing the transforms $c_{x_p}^{\mathbf{\Psi}} \coloneqq x_p\mathbf{\Psi}$ of

each of the mixture components x_p, $1 \leq p \leq P$, one obtains a matrix:

$$C_{\mathbf{x}}^{\Psi} := \mathbf{x}\Psi \tag{10.14}$$

called the transform of mixture $\mathbf{x}$. Its numerical calculation is particularly simple but, as discussed above, this type of transform does not necessarily take full advantage of the dictionary's redundancy to obtain representations with high sparseness.

Algorithms

The linear transforms Ψ applied to a mixture are often associated with fast algorithms: Fourier transform, STFT, orthogonal wavelet transform, etc. Another possibility [29] is that the dictionary Φ consists in the union of a certain number of orthonormal bases associated with fast transforms (wavelet, local cosine, etc.), $\Phi = [\Phi_1 \dots \Phi_J]$, in which case the pseudo-inverse $\Phi^{\dagger} = J^{-1}\Phi^H$ provides a dual transform $\Psi := \Phi^{\dagger}$ that is simple and quick to apply by calculating each of the component fast transforms and concatenating the coefficients obtained. It is also possible to simultaneously calculate two STFTs for the same mixture, one with a small analysis window and the other with a large window, to obtain a sort of multi-window transform [82].

Properties and limitations

By the linearity of the transform, the initial source separation problem (10.1) has an exact analog in the transform domain (see (10.8) and (10.14)):

$$C_{\mathbf{x}}^{\Psi} = \mathbf{A}C_{\mathbf{s}}^{\Psi} + C_{\mathbf{b}}^{\Psi}. \tag{10.15}$$

Separation in the transform domain, to estimate the source transforms $\widehat{C}_{\mathbf{s}}$ based on $C_{\mathbf{x}}$ and ultimately reconstruct $\widehat{\mathbf{s}} := \widehat{C}_{\mathbf{s}}\Phi$, is thus relevant if the supports of the sources in the transform domain $C_{\mathbf{s}}^{\Psi}$ are reasonably disjoint.

Despite the advantages of a simple, efficient numerical implementation, transforms are far from ideal in terms of sparse representation when the dictionary is redundant. In particular, the least squares representation $C_{\mathbf{x}}^{\Phi^{\dagger}} := \mathbf{x}\Phi^{\dagger}$ is generally not as sparse as representations minimizing other criteria more appropriate to sparsity, such as ℓ^{τ} criteria, with $0 \leq \tau \leq 1$. This is why adaptive approaches based on minimizing the ℓ^{τ} norm where $0 \leq \tau \leq 1$ have elicited intense interest since the early 1990s [48,23], in a context extending well beyond that of source separation [83,43,90,29,96]. Most early efforts have focused on adaptive methods for sparse representation of a given signal or image; interest in equivalent methods for joint representation of several signals has developed subsequently [49,68,50,27,104,102,57]. We will discuss joint representation below, first in the general case of ℓ^{τ} criteria where $0 \leq \tau \leq \infty$. We will then look more specifically at the corresponding algorithms for values of $\tau \leq 2$ which can be used to obtain sparse representations.

10.3.3 Principle of ℓ^τ minimization

Rather than using linear transforms, *joint sparse representations* can be defined that minimize an ℓ^τ criterion, for $0 \leq \tau \leq \infty$. Before defining them in detail, we will review the definition of the corresponding "single-channel" sparse representations: among all the admissible representations of a signal x_1 (i.e. satisfying $x_1 = c_{x_1}\mathbf{\Phi}$), the representation $c_{x_1}^\tau$ with the smallest ℓ^τ norm is selected, which amounts to selecting the most likely representation in terms of an assumed generalized Gaussian distribution of the coefficients (10.4).

In the multi-channel case, the optimized criterion takes the form:

$$C_{\mathbf{x}}^\tau \coloneqq \arg \min_{C_{\mathbf{x}} | C_{\mathbf{x}}\mathbf{\Phi} = \mathbf{x}} \sum_{k=1}^{K} \left(\sum_{p=1}^{P} \left| c_{x_p}(k) \right|^2 \right)^{\tau/2} \tag{10.16}$$

for $0 < \tau < \infty$, with an obvious modification for $\tau = \infty$. For $\tau = 0$, with the convention $c^0 = 1$ if $c > 0$ and $0^0 = 0$, the ℓ^0 norm affecting the criteria (10.16) is the number of non-zero columns $C_{\mathbf{x}}(k)$, i.e. the number of atoms used in the representation. Minimizing the ℓ^0 norm is simply a means of representing the mixture with as few atoms as possible, which intuitively corresponds to looking for a sparse representation.

This particular form of the optimized criterion defines a representation of x that actually takes into account the *joint* properties of the different channels x_p, in contrast to a criterion such as $\sum_k \sum_p |c_{x_p}|^\tau$, which may be more intuitive but decouples optimization over the different channels. In section 10.3.4, we will examine a Bayesian interpretation explaining the form taken by the criterion (10.16).

Due to the intrinsic noise $\mathbf{b}$ of the mixing model (10.1) and the possible inaccuracy of the sparse source model (10.6), it is often more useful to look for an approximate sparse representation – also known as a *joint sparse approximation* – via the optimization of the criterion:

$$C_{\mathbf{x}}^{\tau,\lambda} \coloneqq \arg \min_{C_{\mathbf{x}}} \left\{ \| \mathbf{x} - C_{\mathbf{x}}\mathbf{\Phi} \|_F^2 + \lambda \sum_{k=1}^{K} \left(\sum_{p=1}^{P} \left| c_{x_p}(k) \right|^2 \right)^{\tau/2} \right\} \tag{10.17}$$

where $\|\mathbf{y}\|_F^2 \coloneqq \sum_{pt} |y_p(t)|^2$ is the Frobenius norm of the $\mathbf{y}$ matrix, i.e. the sum of the energies of its rows. The first term of the criterion is a data fidelity term that uses least squares to measure the quality of the approximation obtained. The second term measures the "joint" sparsity of the multi-channel representation $C_{\mathbf{x}}$, relative to the criterion (10.16), and the λ parameter determines the respective importance attributed to the quality of the approximation and to sparsity. When λ tends to zero, the obtained solution, $C_{\mathbf{x}}^{\tau,\lambda}$, approaches $C_{\mathbf{x}}^\tau$; when λ is sufficiently large, $C_{\mathbf{x}}^{\tau,\lambda}$ tends to zero.

10.3.4 Bayesian interpretation of ℓ^τ criteria

The specific form of the ℓ^τ criterion used to select a multi-channel joint representation (10.16)–(10.17), which couples the selection of coefficients for the different channels,

is not chosen arbitrarily. The following paragraphs aim to provide an overview of its derivation in a Bayesian context. Readers more interested in the algorithmic and practical aspects can skip to section 10.3.6.

Whether an adaptive or transform approach is used, a Bayesian interpretation is possible for optimizing the criteria in (10.11), (10.16) or (10.17): (10.11) corresponds to calculating the most likely coefficients for reconstructing $\mathbf{x} = C_\mathbf{x}\mathbf{\Phi}$ under the assumption of an iid Gaussian distribution $p_{C_\mathbf{x}}(\cdot)$ with variance $\sigma^2_\mathbf{x}$. Under this assumption, the distribution of $C_\mathbf{x}$ is expressed (see Eq. (10.4) for $\tau = 2$):

$$p_{C_\mathbf{x}}(C_\mathbf{x}) \propto \exp\left(-\frac{\|C_\mathbf{x}\|_F^2}{2\sigma^2_\mathbf{x}}\right) = \exp\left(-\frac{\sum_{k=1}^{K}\|C_\mathbf{x}(k)\|_2^2}{2\sigma^2_\mathbf{x}}\right). \tag{10.18}$$

In the specific context of source separation, the distribution $p_{C_\mathbf{x}}(C_\mathbf{x})$ of $C_\mathbf{x} = \mathbf{A}C_\mathbf{s}$ depends on the distribution $p_{C_\mathbf{s}}(C_\mathbf{s})$ of the source coefficients $C_\mathbf{s}$ and on the distribution $p_\mathbf{A}(\mathbf{A})$ of the mixing matrix. In the absence of prior knowledge of the mixing matrix, for blind separation its distribution is naturally assumed invariant by arbitrary spatial rotations, i.e. $p_\mathbf{A}(\mathbf{A}) = p_\mathbf{A}(\mathbf{U}\mathbf{A})$ for any unitary matrix $\mathbf{U}$. This is the case if the columns $\mathbf{A}_n$ are normalized $\|\mathbf{A}_n\|_2 = 1$, independent and uniformly distributed over the unit sphere of $\mathbb{C}^P$. This assumption alone implies that the prior distribution of the coefficients $C_\mathbf{x}$ of the noiseless mixture for any atom k is radial; that is, for any unitary matrix $\mathbf{U}$:

$$p(C_\mathbf{x}(k)) = p(\mathbf{U}C_\mathbf{x}(k)) = r_k(\|C_\mathbf{x}(k)\|_2) \tag{10.19}$$

where $r_k(\cdot) \geq 0$ is a function of the distributions $p_{C_\mathbf{s}(k)}(C_\mathbf{s}(k))$ and $p_\mathbf{A}(\mathbf{A})$.

If the coefficients $C_\mathbf{s}(k)$ are iid for different k, then the form of the prior distribution of $C_\mathbf{x}$ can be deduced:

$$p(C_\mathbf{x}) = \prod_{k=1}^{K} r(\|C_\mathbf{x}(k)\|_2). \tag{10.20}$$

A *maximum-likelihood* (ML) approach makes it possible to define a decomposition of the noiseless mixture as:

$$C^r_\mathbf{x} := \arg\max_{C_\mathbf{x}|C_\mathbf{x}\mathbf{\Phi}=\mathbf{x}} p(C_\mathbf{x}) = \arg\min_{C_\mathbf{x}|C_\mathbf{x}\mathbf{\Phi}=\mathbf{x}} \sum_{k=1}^{K} -\log r(\|C_\mathbf{x}(k)\|_2). \tag{10.21}$$

In the noisy case, assuming the noise $\mathbf{b}$ is iid Gaussian of variance $\sigma^2_\mathbf{b}$, a *maximum a posteriori (MAP)* approach maximizes the posterior probability, i.e. it minimizes:

$$-\log p(C_\mathbf{x} \mid \mathbf{x}) = \frac{1}{2\sigma^2_\mathbf{b}}\|\mathbf{x} - C_\mathbf{x}\mathbf{\Phi}\|_F^2 + \sum_{k=1}^{K} -\log r(\|C_\mathbf{x}(k)\|_2). \tag{10.22}$$

The function r, which in theory can be calculated from the distributions $p_\mathbf{A}(\mathbf{A})$ of the mixing matrix and $p_c(c)$ of the sparse source coefficients, is sometimes difficult to use

in solving the corresponding optimization problem. Since the distributions on which this function depends are not necessarily well understood themselves, it is reasonable to solve the global optimization problem (10.21) or (10.22) with an assumed specific form of the r function, such as $r(z) \coloneqq \exp(-\alpha |z|^{\tau})$ for an exponent $0 \leq \tau \leq 2$, similarly to the adaptive approaches (10.16)–(10.17).

10.3.5 Effect of the chosen ℓ^{τ} criterion

The choice of the ℓ^{τ} norm is important *a priori*, since it can considerably influence the sparsity of the resulting solution. Figure 10.4 illustrates this point schematically. Figure 10.4(a)-(b)-(c) show, in dimension 2, the contour lines of ℓ^{τ} norms for $\tau = 2$, $\tau = 1$ and $\tau = 0.5$. In Fig. 10.4(d), a continuous line represents the set of admissible representations ($C_{\mathbf{x}}\mathbf{\Phi} = \mathbf{x}$) of a mixture $\mathbf{x}$, which forms an affine sub-space of the space $\mathbb{R}^K$ of all possible representations. The least ℓ^{τ} norm representation of $\mathbf{x}$ is the intersection of this line with the "smaller" contour line of the ℓ^{τ} norm that intersects it. These intersections have been indicated for $\tau = 2$, $\tau = 1$ and $\tau = 0$ (for $\tau = 0$ the intersections are with the axes). One can see that none of the coordinates of the least squares solution ($\tau = 2$) is non-zero, whereas the least ℓ^0 or ℓ^1 norm solutions are situated along the axes. Here, the least ℓ^1 norm solution, which is unique, also coincides with the least ℓ^{τ} norm solutions for $0 < \tau \leq 1$. Geometrically, this is explained by the fact that the contour lines of the ℓ^{τ} norms, for $0 < \tau \leq 1$ have "corners" on the axes, in contrast to the contour lines of the ℓ^2 norm (and of the ℓ^{τ} norms for $1 < \tau \leq 2$), which are convex and rounded.

As a result, in the single-channel context, any representation $C_{\mathbf{x}}^{\tau} \in \mathbb{R}^K$ (or $C_{\mathbf{x}}^{\tau,\lambda}$) that solves the optimization problem (10.16) (or (10.17)) for $0 \leq \tau \leq 1$, is *truly sparse* insofar as it has no more than T non-zero coefficients, where T is the dimension of the analyzed signal $\mathbf{x} \in \mathbb{R}^T$.[5] This property is due to the "concavity" of the ℓ^{τ} norm [64] for $0 \leq \tau \leq 1$, as shown in Fig. 10.4(c). We will refer to this property repeatedly throughout this chapter, for the multi-channel case as well. It can be defined as follows:

DEFINITION 10.1
The representation $C_{\mathbf{x}}^{\tau}$ (or $C_{\mathbf{x}}^{\tau,\lambda}$) is a truly sparse representation if it implies no more than T atoms of the dictionary (where T is the dimension of the signals), i.e. if there are no more than T indices k whereby the column $C_{\mathbf{x}}(k)$ is non-zero.

This definition is given here in the multi-channel case, but it generalizes an equivalent idea initially proposed for the single-channel case by Kreutz-Delgado *et al*. [64].

For $\tau > 1$, since ℓ^{τ} is a real norm in the mathematical sense of the term, it is convex, and its minimization does not lead to truly sparse representations, as shown in Fig. 10.4(a). However, the convexity of the corresponding criteria greatly facilitates their optimization and can favor their practical use.

[5] Here, the fact that we are considering real-valued signals and coefficients seems to be important, since examples with complex coefficients show that the ℓ^{τ} minimizer can be non "truly sparse" [108].

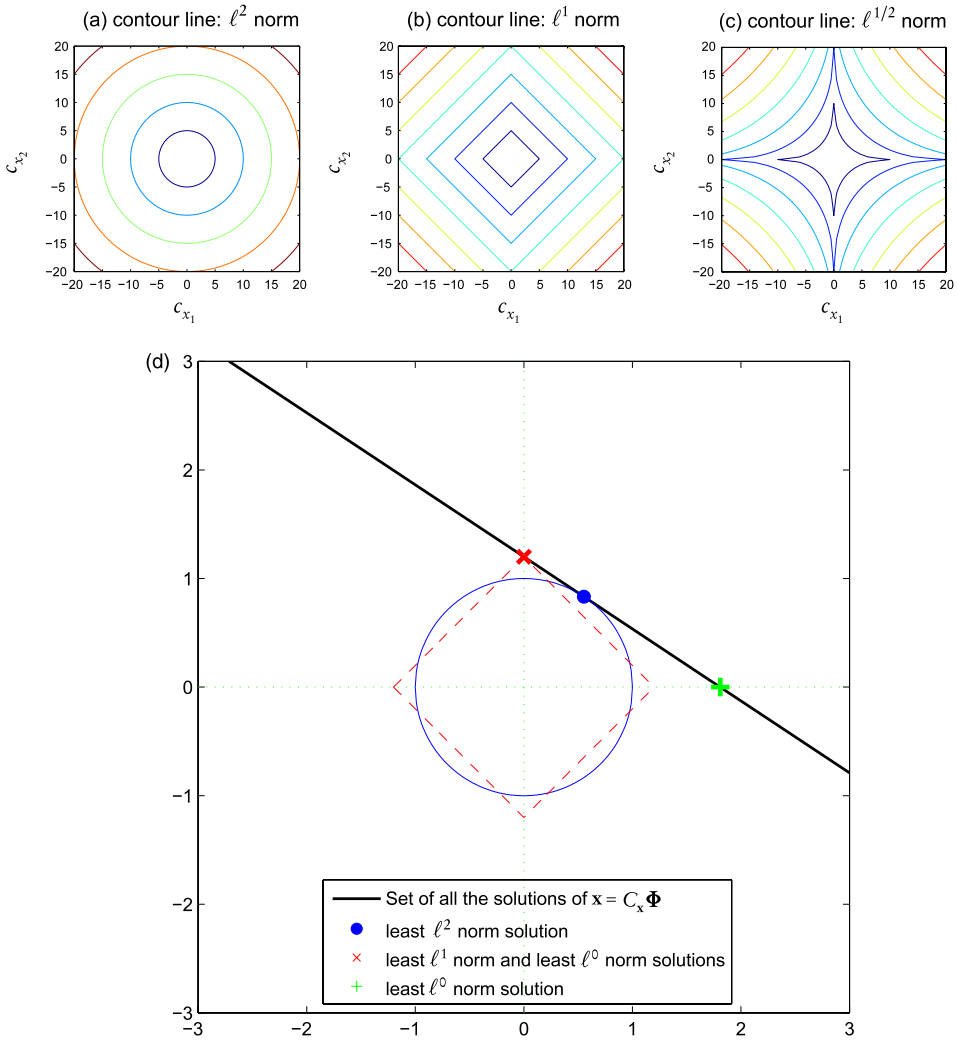


FIGURE 10.4

(a)-(b)-(c): contour lines of ℓ^τ norms for $\tau = 2, 1, 1/2$. (d) Schematic comparison of the sparsity of representations obtained using least squares (non-adaptive linear transform) and low-dimensional ℓ^1 or ℓ^0 norm minimization (adaptive representations). The continuous line symbolizes all the possible representations $C_{\mathbf{x}} = (c_{x_1}, c_{x_2})$ of a given mixture $\mathbf{x}$. The intersection of this line with the circle centered on the origin which is tangentine to it provides the least squares solution. Its intersections with the x-axis ("+" symbol) and the y-axis ("×" symbol) correspond to two solutions with a unique non-zero coefficient, and thus a minimal ℓ^0 norm. The y-axis solution is also the unique intersection between all the possible representations and the smallest "oblique" square centered on the origin (dotted line) that intersects it, making it the least ℓ^1 norm solution as well. One can see that the least squares solution has more non-zero coefficients than the least ℓ^0 or ℓ^1 norm solutions.

Intensive theoretical work, initiated by the results of Donoho and Huo [33], has shown in the single-channel case that the specific choice of the ℓ^τ norm optimized with $0 \leq \tau \leq 1$ has little (or no) influence on the representation obtained [55,54,52] when the analyzed signal $\mathbf{x}$ is adequately approximated with sufficiently sparse coefficients $C_\mathbf{x}$. Similar results for the multi-channel case [102] show that, when the same assumptions are made, the ℓ^0 and ℓ^1 criteria essentially lead to the same representations. Extending these results to the ℓ^τ criteria, $0 \leq \tau \leq 1$, will undoubtedly point to the same conclusion.

In light of these theoretical results, the choice of a specific ℓ^τ criterion to optimize for $0 \leq \tau \leq 1$ depends primarily on the characteristics of the algorithms available for optimization and their numerical properties. The description of optimization algorithms for the slightly broader range of $0 \leq \tau \leq 2$ is the subject of the following section.

10.3.6 Optimization algorithms for ℓ^τ criteria

Adaptive representation algorithms that optimize a criterion (10.17) or (10.16) for $0 \leq \tau \leq 2$ have received a great deal of attention, with regard to theoretical analysis as well as algorithms and applications.

For $\tau = 2$, both (10.16) and (10.17) can be solved with simple linear algebra, but for $\tau = 0$, the problem is a combinatorial one [31,78] and *a priori* untractable. A body of theoretical work initiated by Donoho and Huo [33] suggests that its solution is often equal or similar to that of the corresponding problems with $0 < \tau \leq 1$ [55]. The algorithms able to optimize these criteria are of particular interest, despite the specific difficulties of the non-convex optimization of $\tau < 1$, and in particular the risk of obtaining a local optimum. Finally, the algorithms for optimizing strictly convex ℓ^τ criteria with $1 < \tau \leq 2$ offer the advantage of simplicity, even though the theoretical elements linking the representations they provide to those obtained with ℓ^τ criteria for $0 \leq \tau \leq 1$ are currently lacking. Their main advantage, owing to the strict convexity of the optimized criterion, is the guaranteed convergence to the unique global optimum of the criterion.

For all the algorithms presented below, selecting an order of magnitude for the regularization parameter λ is also a complex issue in blind mode. An excessively high value for this parameter will over-emphasize the sparsity of the obtained representation $C_\mathbf{x}^{\tau,\lambda}$, compromising the quality of the reconstruction $||\mathbf{x} - C_\mathbf{x}^{\tau,\lambda}\mathbf{\Phi}||_2$. An excessively low value will amount to assuming that the sparse source model is accurate and the mixture is noiseless, which could result in artefacts. The selection of parameters τ and λ, which define the optimized criterion, should thus represent a compromise between the sparsity of the representation and the numerical complexity of the minimization. In typical applications, λ would be chosen to obtain a reconstruction error of the order of the noise, if the magnitude of the latter is known.

Algorithm for $\tau = 2$: regularized least squares

For $\tau = 2$, the exact representation $C_\mathbf{x}^2$ solves the least squares problem $\min ||C_\mathbf{x}||_F^2$ under the constraint $C_\mathbf{x}\mathbf{\Phi} = \mathbf{x}$ and is obtained analytically by linear transform, using

the pseudo-inverse of the dictionary, i.e. $C_{\mathbf{x}}^{2} = C_{\mathbf{x}}^{\mathbf{\Phi}^{\dagger}} = \mathbf{x}\mathbf{\Phi}^{\dagger}$. As shown in Fig. 10.4, $C_{\mathbf{x}}^{2}$ is generally not a "truly sparse" representation. The approximate representations $C_{\mathbf{x}}^{2,\lambda}$, which solve the regularized least squares problem $\min ||\mathbf{x} - C_{\mathbf{x}}\mathbf{\Phi}||_F^2 + \lambda ||C_{\mathbf{x}}||_F^2$ for $\lambda > 0$, are also obtained linearly as $C_{\mathbf{x}}^{2,\lambda} = \mathbf{x}\mathbf{\Psi}_{\lambda}$, where $\mathbf{\Psi}_{\lambda}$ has two equivalent expressions:

$$\mathbf{\Psi}_{\lambda} \coloneqq (\mathbf{\Phi}^{\mathrm{H}}\mathbf{\Phi} + \lambda \mathbf{I}_T)^{-1}\mathbf{\Phi}^{\mathrm{H}} = \mathbf{\Phi}^{\mathrm{H}}(\mathbf{\Phi}\mathbf{\Phi}^{\mathrm{H}} + \lambda \mathbf{I}_K)^{-1} \tag{10.23}$$

with $\mathbf{I}_M$ as the $M \times M$ identity matrix. When λ tends to zero, $\mathbf{\Psi}_{\lambda}$ tends to the pseudo-inverse $\mathbf{\Phi}^{\dagger}$ of the dictionary, and for any λ value the linearity of the representation guarantees that $C_{\mathbf{x}}^{2,\lambda} = \mathbf{A}C_{\mathbf{s}}^{2,\lambda} + C_{\mathbf{b}}^{2,\lambda}$.

Algorithm for $0 < \tau \leq 2$: M-FOCUSS

For $0 < \tau \leq 2$ the *M-FOCUSS* algorithm [27] (an *iterative reweighted least squares* or IRLS algorithm) and its regularized version are iterative techniques that converge extremely quickly towards a local minimum of the criterion to be optimized. This algorithm is derived from the FOCUSS algorithm initially defined for the single-channel case [48]. Given the non-convexity of the optimized criterion, it is difficult to predict whether the global optimum is reached, which in any case depends on the chosen initialization. It has not yet been demonstrated (except in the $P = 1$ single-channel case) that the algorithm always converges on a "truly sparse" solution, although this is indeed the case experimentally [27]. Starting with an initialization $C_{\mathbf{x}}^{(0)}$, the value of which can have a decisive impact on the algorithm's point of convergence, a weighted mean square criterion is minimized iteratively:

$$C_{\mathbf{x}}^{(m)} \coloneqq \arg\min_{C_{\mathbf{x}}} \left\{ ||\mathbf{x} - C_{\mathbf{x}}\mathbf{\Phi}||_F^2 + \frac{\lambda|\tau|}{2} ||C_{\mathbf{x}}(W^{(m)})^{-1}||_F^2 \right\} \tag{10.24}$$

where the diagonal weighting matrix:

$$W^{(m)} \coloneqq \mathrm{diag}\left(||C_{\mathbf{x}}^{(m-1)}(1)||_2^{1-\tau/2}, \ldots, ||C_{\mathbf{x}}^{(m-1)}(K)||_2^{1-\tau/2} \right) \tag{10.25}$$

tends to drive to zero those columns whose energy $||C_{\mathbf{x}}^{(m-1)}(k)||_2$ is low. In practice, as we saw in section 10.3.6, each iteration thus involves updating the diagonal weighting matrix $W^{(m)}$ and the weighted dictionary $\mathbf{\Phi}^{(m)} \coloneqq W^{(m)}\mathbf{\Phi}$ to obtain:

$$C_{\mathbf{x}}^{(m)} \coloneqq \mathbf{x}\left((\mathbf{\Phi}^{(m)})^{\mathrm{H}}\,\mathbf{\Phi}^{(m)} + \frac{\lambda|\tau|}{2}\mathbf{I}_T \right)^{-1} (\mathbf{\Phi}^{(m)})^{\mathrm{H}}\, W^{(m)}. \tag{10.26}$$

While the M-FOCUSS algorithm converges very rapidly towards a local minimum of the optimized criterion [27] and often produces very sparse representations, it is difficult to determine when convergence to the global minimum is reached, depending on the chosen initialization. The initialization most often used, and which in practice gives

satisfactory results, is the least squares representation, obtained using the pseudo-inverse $\mathbf{\Phi}^{\dagger}$ of the dictionary. Even if the number of iterations necessary before convergence is low, each iteration can be quite costly numerically because of the inversion of a square matrix of size at least T which is required, cf. (10.26).

Because the algorithm is globally nonlinear, it is difficult *a priori* to determine whether its representation $C_{\mathbf{x}}$ of the mixture satisfies the identity $C_{\mathbf{x}} \approx \mathbf{A}C_{\mathbf{s}}$ with reasonable accuracy. Theoretical results [52,102] on optimization criteria similar to (10.17) suggest this to be true when the representations of the sources are sufficiently sparse and disjoint, but actual proof is still lacking for the optimum $C_{\mathbf{x}}^{\tau,\lambda}$ of the criterion (10.17), and for the approximate numerical solution $\lim_{m\to\infty} C_{\mathbf{x}}^{(m)}$ calculated with M-FOCUSS.

Algorithms for $\tau = 1$ (single-channel case)

Optimizing (10.16) in the single-channel case ($P = 1$) for $\tau = 1$ is known as *Basis Pursuit* and optimizing (10.17) is known as *Basis Pursuit Denoising* [23]. Proposed by Donoho and his colleagues, Basis Pursuit and Basis Pursuit Denoising are closer to principles than actual algorithms, since the optimization method is not necessarily specified. Several experimental studies and the intense theoretical work initiated by Donoho and Huo [33] have shown the relevance of sparse representations obtained using the ℓ^1 criterion; for example, they guarantee the identity $C_{\mathbf{x}} \approx \mathbf{A}C_{\mathbf{s}}$ when $C_{\mathbf{s}}$ is sufficiently sparse [102].

However, the generic numerical optimization methods of *linear programming*, *quadratic programming*, *conic programming*, etc. [12] for calculating them (in both the real and complex cases and the single-channel and multi-channel cases) are often computationally costly. In order to reduce computations, Truncated Newton strategy was used by [23] in the framework of Primal-Dual Interior Point method. Algorithms such as LARS (*Least Angle Regression*) [35] that take more specific account of the optimization problem [105, 35,74,85,45] are able to further simplify and accelerate the calculations. The next challenge is to adapt these efficient algorithms to the multi-channel case.

Algorithms for $1 \leq \tau \leq 2$: iterative thresholding (single-channel case)

For the values $1 \leq \tau \leq 2$, the criterion to optimize (10.17) is convex. Efficient iterative numerical methods were proposed [28,36,42] for calculating the optimum $c_x^{\lambda,\tau}$, for the $P = 1$ single-channel case. Starting with an arbitrary initialization $c_x^{(0)}$ (typically $c_x^{(0)} = 0$), the principle is to iterate the following thresholding step:

$$c_x^{(m)}(k) = S_{\lambda,\tau}\left(c_x^{(m-1)}(k) + \langle x - c_x^{(m-1)}\mathbf{\Phi}, \varphi_k\rangle\right), \qquad 1 \leq k \leq K \tag{10.27}$$

where $S_{\lambda,\tau}$ is a thresholding function, plotted in Fig. 10.5 for $\lambda = 1$ and a few values of $1 \leq \tau \leq 2$.

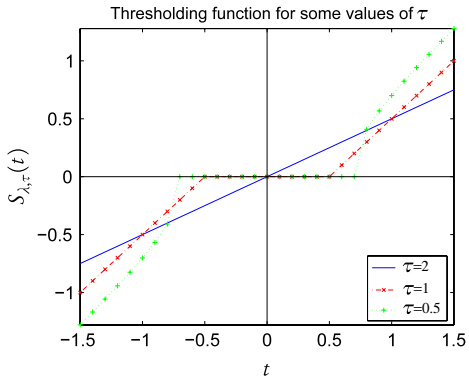


FIGURE 10.5

Curves of the thresholding function $S_{\lambda,\tau}(t)$ over the interval $t \in [-1.5,\ 1.5]$ for $\lambda = 1$ and a few values of τ.

For $\tau = 1$, the soft thresholding function is used:

$$S_{\lambda,1}(t) \coloneqq \begin{cases} t + \dfrac{\lambda}{2}, & \text{if } t \leq -\dfrac{\lambda}{2} \\ 0, & \text{if } |t| < \dfrac{\lambda}{2} \\ t - \dfrac{\lambda}{2}, & \text{if } t \geq \dfrac{\lambda}{2} \end{cases} \tag{10.28}$$

whereas for $\tau > 1$:

$$S_{\lambda,\tau}(t) \coloneqq u, \text{ where } u \text{ satisfies } u + \frac{\lambda \cdot \tau}{2} \cdot \text{sign}(u) \cdot |u|^{\tau-1} = t. \tag{10.29}$$

This thresholding function comes from the optimization of criterion (10.17) when $\mathbf{\Phi}$ is an orthonormal basis (recall that iterative thresholding is defined for the $P = 1$ single-channel case), because each coordinate must then be optimized independently:

$$\min_{c(k)} |\langle x, \varphi_k \rangle - c(k)|^2 + \lambda |c(k)|^\tau .$$

When the value of the optimum $c_x^{\lambda,\tau}(k)$ is non-zero, the derivative of the optimized criterion exists and cancels outs, which implies:

$$2(c_x^{\lambda,\tau}(k) - \langle x, \varphi_k \rangle) + \lambda \cdot \text{sign}(c_x^{\lambda,\tau}(k)) \cdot \left| c_x^{\lambda,\tau}(k) \right|^{\tau-1} = 0$$

that is, according to the expression of the thresholding function (10.29):

$$c_x^{\lambda,\tau}(k) = S_{\lambda,\tau}(\langle x, \varphi_k \rangle).$$

For $1 \leq \tau \leq 2$, Daubechies, Defrise and De Mol proved [28, theorem 3.1] that if the dictionary satisfies $\|\mathbf{\Phi} c\|_2^2 \leq \|c\|_2^2$ for all c, then $c^{(m)}$ converges strongly to a minimizer $c_x^{\lambda,\tau}$ of (10.17), regardless of the initialization. If $\mathbf{\Phi}$ is also a basis (not necessarily orthonormal), this minimizer is unique, and the same is true by strict convexity if $\tau > 1$. However, this minimizer is generally a "truly sparse" solution only for $\tau = 1$. In addition, when the dictionary consists of atoms of unit energy, then at best $\|\mathbf{\Phi} c\|_2^2 \leq \mu \|c\|_2^2$ for a certain number $\mu < \infty$. The algorithm must then be slightly modified to ensure convergence, by iterating the thresholding step below:

$$c_x^{(m)}(k) = S_{\frac{\lambda}{\mu},\tau}\left(c_x^{(m-1)}(k) + \frac{\langle x - c_x^{(m-1)} \mathbf{\Phi}, \varphi_k \rangle}{\mu} \right). \tag{10.30}$$

While there is currently no known complexity analysis of this algorithm, it often takes few iterations to obtain an excellent approximation of the solution. Additional significant acceleration can be achieved via sequential optimization over subspaces spanned by the current thresholding direction and directions of a few previous steps (SESOP) [37]. Many algorithmic variants exist and the approach is intimately related to the notion of proximal algorithm. Each iteration is relatively low-cost given that matrix inversion is not necessary, unlike for the algorithm M-FOCUSS, which has been examined previously. These thresholding algorithms can be extended to the multi-channel case [44]. To use them for sparse source separation, it remains to be verified whether they guarantee approximate identification $C_{\mathbf{x}} \approx \mathbf{A} C_{\mathbf{s}}$.

10.3.7 Matching pursuit

Matching Pursuit proposes a solution to the problem of joint sparse approximation $\mathbf{x} \approx C_{\mathbf{x}} \mathbf{\Phi}$ not through the global optimization of a criterion, but through an iterative *greedy algorithm*. Each step of the algorithm involves finding the atom in the dictionary that, when added to the set of already selected atoms, will reduce the approximation error to the largest extent. We assume here that the atoms in the dictionary are of unit energy. To approximate the multi-channel signal $\mathbf{x}$, a residual $\mathbf{r}^{(0)} := \mathbf{x}$ is initialized, and then the following steps are repeated, starting at $m = 1$:

1. Selection of most correlated atom:

$$k_m := \arg\max_k \sum_{p=1}^{P} |\langle \mathbf{r}_p^{(m-1)}, \varphi_k \rangle|^2. \tag{10.31}$$

2. Updating of residual:

$$\mathbf{r}_p^{(m)} := \mathbf{r}_p^{(m-1)} - \langle \mathbf{r}_p^{(m-1)}, \varphi_{k_m} \rangle \varphi_{k_m}, \qquad 1 \leq p \leq P. \tag{10.32}$$

After M iterations, one obtains the decomposition:

$$\mathbf{x} = \sum_{m=1}^{M} \begin{pmatrix} c_1(m)\varphi_{k_m} \\ \cdots \\ c_P(m)\varphi_{k_m} \end{pmatrix} + \mathbf{r}^{(M)} \tag{10.33}$$

with $c_p(m) := \langle \mathbf{r}_p^{(m-1)}, \varphi_{k_m} \rangle$.

This algorithm was initially introduced for single-channel processing by Mallat and Zhang [76]. The generalization to the multi-channel case [49,50,27,68] that we present here converges in the same manner as the original single-channel version [49,50] in that the residual norm tends to zero as the number of iterations tends to infinity. In contrast, the convergence rate, whether single-channel or multi-channel, is not fully understood [68]. In one of the variants of this algorithm, the residuals can be updated at each step by orthogonal projection of the analyzed signal on the linear manifold spanned by all the selected atoms (referred to as orthogonal *matching pursuit*). It is also possible to select the best atom at each step according to other criteria [68,27,104], or to select several atoms at a time [79,14].

The *matching pursuit* presented here, which involves no matrix inversion, can be implemented quite rapidly by using the structure of the dictionary and fast algorithms for calculating and updating $\mathbf{r}^{(m)}\mathbf{\Phi}^{\mathrm{H}}$ [53,65]. The main parameter is the choice of the stopping criterion; the two main approaches set either the number of iterations or the targeted approximation error $\|\mathbf{r}^{(m)}\|_F$ beforehand. A typical choice is to stop when the residual error is of the order of magnitude of the noise level, if the latter is known.

Like the approaches based on ℓ^1 norm minimization [102], *matching pursuit* has been the subject of several experimental and theoretical studies, which have shown [104] that it can guarantee the identity $C_{\mathbf{x}} \approx \mathbf{A} C_{\mathbf{s}}$ when $C_{\mathbf{s}}$ is sufficiently sparse [56].

10.3.8 Summary

Table 10.1 summarizes the main numerical characteristics of the algorithms for joint sparse signal representation presented in this section. Their field of application extends far beyond source separation, since the need to jointly represent several signals exists in a number of areas. For example, color images consist of three channels (red-green-blue) whose intensities are correlated, and the use of joint sparse representation techniques (based on *matching pursuit* [43]) has proven effective for low-bit rate image compression. Readers interested in further details on these algorithms will find their original definitions in the bibliographic references and, in some cases therein, an analysis of certain theoretical properties that have been proven.

Table 10.1 Comparative summary of algorithms for joint sparse mixture representation discussed in section 10.3

Method	**Transform**	**M-FOCUSS**	**Basis Pursuit**	**Iterative thresholding**	**Matching Pursuit**
Criterion	(10.17)	(10.17)	(10.17)	(10.17)	(10.31)–(10.32)
Parameters	$\tau = 2$ λ	$0 \leq \tau \leq 2$ λ	$\tau = 1$ λ	$1 \leq \tau \leq 2$ λ	M iterations
Advantages	–very fast	–high sparsity ($\tau \leq 1$)	–high sparsity	–fast –good sparsity	–fast –good sparsity
Issues and difficulties	–choice of λ –limited sparsity	–choice of λ –initialization –memory cost –computing time (inversion of large matrices)	–choice of λ –memory cost –computing time (linear, quadratic, conic progr.)	–choice of λ	–choice of M
References					
$P = 1$		[48] [55]	[23] [33]	[28]	[76] [101]
$P > 1$		[52] [27]	[103] [102]	[44]	[58] [49] [104]

In practice, the joint sparse representation of a mixture implies two choices:

1. selection of a dictionary $\mathbf{\Phi}$ in which the mixture channels are likely to allow a sufficiently sparse joint representation;
2. selection of an algorithm and its parameters for calculating a representation.

For example, for audio signals, the Gabor dictionary (10.7) is often used to take account of the frequency components that may appear at different times in the mixture, and the representation most often used is the STFT, a linear transform. While we have attempted in this section to present useful information for selecting an algorithm, selecting the dictionary currently mostly depends on expert knowledge of the analyzed signals in most cases. One can also learn a dictionary automatically from a set of signal realizations [70,71,80,4,92].

10.4 ESTIMATING THE MIXING MATRIX BY CLUSTERING

The algorithms discussed above can be used to analyze a mixture $\mathbf{x}$ in order to calculate a joint sparse representation (or approximation) $C_{\mathbf{x}}$, which is the first step in a source separation system based on sparsity, as described in Fig. 10.2. The second step of this system, for separating instantaneous linear mixtures, consists in estimating the mixing matrix $\mathbf{A}$ by means of the scatter plot of the coefficients $\{C_{\mathbf{x}}(k)\}_{1\leq k\leq K}$.

For determined mixtures, it is of course possible to use any ICA method to estimate $\mathbf{A}$, which means that the other techniques described in this book can be employed as well. Here we concentrate on approaches specifically based on the assumed sparsity of the sources.

The two main classes of approaches used make assumptions about the sparsity of the sources but not about the type of mixture, i.e. determined or undetermined. The first approach, based on Bayesian (or variational) interpretation, aims to maximize the likelihood of the estimated matrix and involves (approximate) optimization of the global criteria; the second is more geometric and relies on clustering algorithms.

Variational approaches

Variational approaches [70,115] generally involve joint iterative optimization of the matrix and the source representations. This task is very challenging computationally, and the quality of solution may suffer from the presence of spurious local minima. However, in the case of square mixing matrix, the problem is reduced to the optimization with respect to unmixing matrix only, which can be treated efficiently using the Relative Newton method [112,113] as described in section 10.5.

Geometric approach: clustering from scatter plots

The geometric approach is very intuitive and capable of estimating the number of sources [8]. In addition, certain clustering algorithm variants do not make the strong sparsity assumption in order to estimate the mixing matrix using simple algorithms, even when the source representations are almost non-disjoint [2,8].

Most of the approaches presented below are applicable in theory, regardless of the number of sensors $P \geq 2$ [107,98,2,8]. However, we will only explain these approaches where $P = 2$, corresponding for example to stereophonic audio mixtures, because the geometric intuitiveness is simpler and more natural in such cases. Since generalizing to the $P \geq 3$ case poses no particular conceptual problems, we leave this to the reader. However, the computing requirements become excessive for certain methods, particularly those based on calculating histograms.

10.4.1 Global clustering algorithms

Figure 10.6(a), an enlargement of Fig. 10.1(c), shows a typical scatter plot of the coefficients $\{C_{\mathbf{x}}(k)\} \subset \mathbb{R}^2$ for a stereophonic mixture ($P = 2$). Alignments of points are observed along the straight lines generated by the columns $\mathbf{A}_n$ of the mixing matrix. In this example, each point $C_{\mathbf{x}}(k) = \rho(k) \cdot [\cos\theta(k), \sin\theta(k)]^T \in \mathbb{R}^2$ can also be shown with polar coordinates:

$$\begin{cases} \rho(k) & := & (-1)^{\varepsilon} \sqrt{|c_{x_1}(k)|^2 + |c_{x_2}(k)|^2}, \\ \theta(k) & := & \arctan(c_{x_2}(k)/c_{x_1}(k)) + \varepsilon\,\pi, \end{cases} \tag{10.34}$$

where $\varepsilon \in \{0,1\}$ is selected such that $0 \leq \theta(k) < \pi$, as shown in Fig. 10.6(b). The points then accumulate around the angles θ_n corresponding to the directions of the columns $\mathbf{A}_n$. A natural idea [100,15] is to exploit the specific geometric structure of the scatter plot in order to find the directions $\widehat{\theta}_n$, making it possible to estimate $\widehat{\mathbf{A}}_n := [\cos\widehat{\theta}_n \sin\widehat{\theta}_n]^T$.

Raw histogram

A naive approach consists of calculating a histogram of $\theta(k)$ angles, assuming the accumulation of points corresponding to the directions of interest will result in visible peaks. The resulting raw histogram, shown in Fig. 10.7(a) and based on points from Fig. 10.6(b), does make it possible to see (and detect by thresholding) one of the directions of interest, but trying to find the other directions with it would be futile.

The scatter plot often contains a number of points of low amplitude $\rho(k)$, whose direction $\theta(k)$ is not representative of the column directions of the mixing matrix. These points, which correspond to the numerous atoms (time-frequency points in this example) where all the sources are approximately inactive, have a "flattening" effect on the histogram of directions, making it unsuitable for detecting the direction of the sources.

Weighted histograms

A weighted histogram can be calculated in a more efficient manner, and may be smoothed with "potential functions", also called Parzen windows. The weighting aims to obtain a histogram that depends on the small number of points with large amplitude rather than the majority of points with negligible amplitude. The following function is

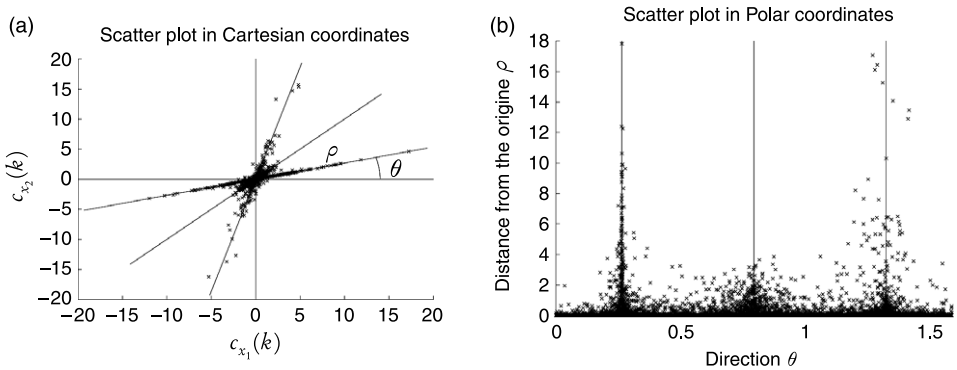


FIGURE 10.6

Scatter plot of MDCT coefficients for the mixture on the right of Fig. 10.1, with Cartesian coordinates (a) and polar coordinates (b). Accumulations of points, shown as dotted lines, are observed along the directions of the columns of the mixing matrix.

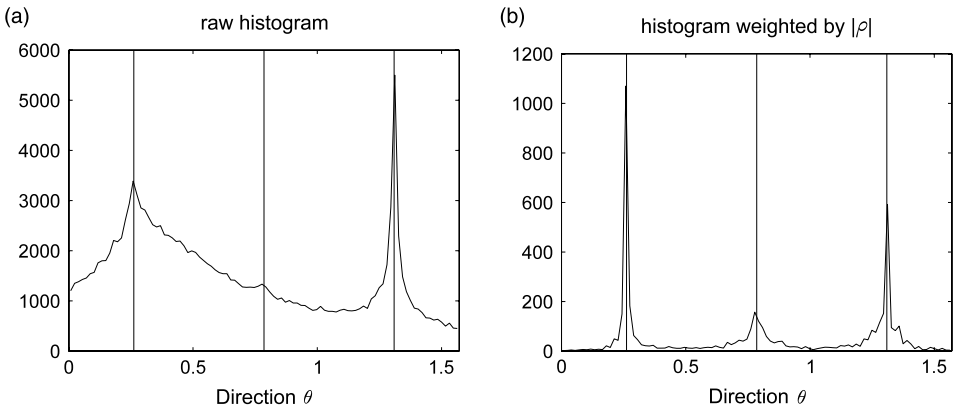


FIGURE 10.7

Histograms based on the scatter plot with polar coordinates in Fig. 10.6(b): (a) raw histogram measuring the frequencies of occurrence at a given θ angle; (b) histogram weighted by the distance from the origin $|\rho|$. Weighting based on distance from the origin results in strong peaks close to the directions of the actual sources (indicated by solid lines) whereas only one peak is visible on the raw histogram.

calculated (for a set of discrete angles on a grid $\theta_\ell = \ell\pi/L$, $0 \leq \ell \leq L$):

$$H(\theta_\ell) := \sum_{k=1}^{K} f(|\rho(k)|) \cdot w(\theta_\ell - \theta(k)) \tag{10.35}$$

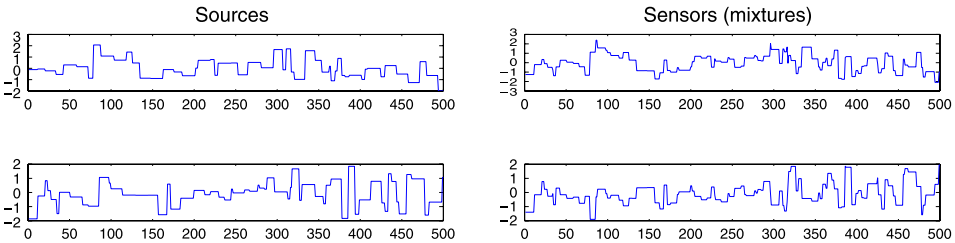


FIGURE 10.8

Time plots of block signals.

where $f(\rho)$ is a weighting function that gives a different role to the scatter plot points depending on their distance from the origin $|\rho(k)|$, and $w(\cdot)$ is a smoothing window, also known as a potential function [15] or Parzen window.

Figure 10.7(b) illustrates the histogram thus obtained with a rectangular smoothing window of size π/L (corresponding to a *non-smoothed* histogram) and respectively $f(|\rho|) = |\rho|$. Taking account of ρ to weight the histogram proves to be critical in this example for the success of histogram-based techniques, since it highlights peaks in the histogram close to angles corresponding to the actual directions of the sources in the mixture. Weighting according to the distance from the origin $|\rho(k)|$ is more satisfactory here than using a raw histogram, because it more effectively highlights these peaks. Other forms of weighting are possible [111], but no theoretical or experimental comparative studies were found examining the effect of the chosen smoothing window and weighting function on the quality of the estimation of $\mathbf{A}$.

10.4.2 Scatter plot selection in multiscale representations

In many cases, especially in wavelet-related decompositions, there are distinct groups of coefficients, in which sources have different sparsity properties. The idea is to select those groups of features (coefficients) which are best suited for separation, with respect to the following criteria: (1) sparsity of coefficients; (2) separability of sources' features [117, 114,62]. After the best groups are selected, one uses only these in the separation process, which can be accomplished by any ICA method, including the Relative Newton method (in the case of square mixing matrix), or by clustering.

10.4.2.1 Motivating example: random blocks in the Haar basis

Typical block functions are shown in Fig. 10.8. They are piecewise constant, with random amplitude and duration of each constant piece. Let us take a close look at the Haar wavelet coefficients at different resolutions. Wavelet basis functions at the finest resolution are obtained by translation of the Haar mother wavelet:

$$\varphi_j(t) = \begin{cases} -1 & \text{if } t = 0 \\ 1 & \text{if } t = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{10.36}$$

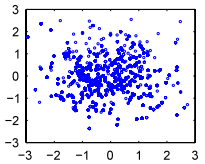
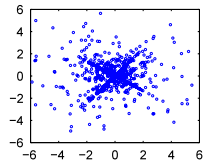
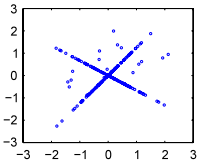
	Raw signals	All wavelet coefficients	Hi resolution wavelet coefficients
			
BS-Infomax	13.9%	4.2%	0.69%
C-means clustering	13.3%	2.4%	0.41%

FIGURE 10.9

Separation of block signals: scatter plots of sensor signals and mean-squared separation errors (%).

Taking a scalar product of a function $s(t)$ with the wavelet $\varphi_j(t-\tau)$, one produces a finite differentiation of the function $s(t)$ at the point $t=\tau$. This means that the number of non-zero coefficients at the finest resolution for a block function will correspond roughly to the number of jumps it has. Proceeding to the next, coarser resolution level

$$\varphi_{j-1}(t)=\begin{cases}-1 & \text{if } t=-1,-2\\ 1 & \text{if } t=0,1\\ 0 & \text{otherwise}\end{cases} \tag{10.37}$$

the number of non-zero coefficients still corresponds to the number of jumps, but the total number of coefficients at this level is halved, and so is the sparsity. If we proceed further in this direction, we will achieve levels of resolution where the typical width of a wavelet $\varphi_j(t)$ is comparable to the typical distance between jumps in the function $s(t)$. In this case, most of the coefficients are expected to be non-zero, and, therefore, sparsity will fade out.

To demonstrate how this influences the accuracy of blind source separation, two block-signal sources were randomly generated (Fig. 10.8, left), and mixed by the matrix

$$\mathbf{A}=\begin{pmatrix}0.8321 & 0.6247\\ -0.5547 & 0.7809\end{pmatrix}. \tag{10.38}$$

The resulting mixtures, $x_1(t)$ and $x_2(t)$ are shown in Fig. 10.8, right. Figure 10.9, first column, shows the scatter plot of $x_1(t)$ versus $x_2(t)$, where there are no visible distinct features. In contrast, the scatter plot of the wavelet coefficients at the highest resolution (Fig. 10.9, third column) shows two distinct orientations, which correspond to the columns of the mixing matrix.

Results of separation of the block sources are presented in Fig. 10.9. The largest error (13%) was obtained on the raw data, and the smallest (below 0.7%) — on the wavelet coefficients at the highest resolution, which have the best sparsity. Use of all wavelet coefficients leads to intermediate sparsity and performance.

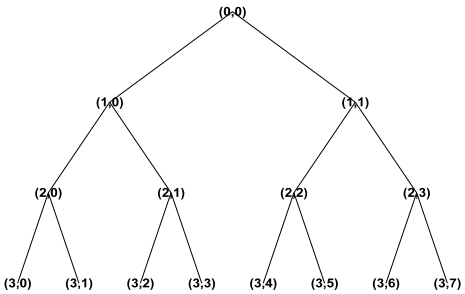


FIGURE 10.10

Wavelet packets tree.

10.4.2.2 Adaptive wavelet packet selection

Multiresolution analysis. Our choice of a particular wavelet basis and of the sparsest subset of coefficients was obvious in the above example: it was based on knowledge of the structure of piecewise constant signals. For sources having oscillatory components (like sounds or images with textures), other systems of basis functions, for example, wavelet packets [26], or multiwavelets [109], might be more appropriate. The wavelet packets library consists of the triple-indexed family of functions:

$$\varphi_{jnk}(t) = 2^{j/2}\varphi_n(2^j t - k), \quad j,k \in \mathbb{Z},\ n \in \mathbb{N}. \tag{10.39}$$

As in the case of the wavelet transform, j,k are the scale and shift parameters, respectively, and n is the frequency parameter, related to the number of oscillations of a particular generating function $\varphi_n(t)$. The set of functions $\varphi_{jn}(t)$ forms a (j,n) wavelet packet. This set of functions can be split into two parts at a coarser scale: $\varphi_{j-1,2n}(t)$ and $\varphi_{j-1,2n+1}(t)$. It follows that these two form an orthonormal basis of the subspace which spans $\{\varphi_{jn}(t)\}$. Thus, one arrives at a family of wavelet packet functions on a binary tree (Fig. 10.10). The nodes of this tree are numbered by two indices: the depth of the level $j = 0, 1, \ldots, J$, and the number of nodes $n = 0, 1, 2, 3, \ldots, 2^j - 1$ at the specified level j. Using wavelet packets allows one to analyze given signals not only with a scale-oriented decomposition but also on frequency sub-bands. Naturally, the library contains the wavelet basis.

The decomposition coefficients $c_{jnk} = \langle s, \varphi_{jnk} \rangle$ also split into (j,n) sets corresponding to the nodes of the tree, and there is a fast way to compute them using banks of *conjugate mirror filters*, as is implemented in the fast wavelet transform.

Choice of the best nodes in the tree. A typical example of scatter plots of the wavelet packet coefficients at different nodes of the wavelet packet tree is shown in Fig. 10.11 (frequency-modulated signals were used as sources.) The upper left scatter plot, labeled "C", corresponds to the set of coefficients at all nodes. The remainder are the scatter

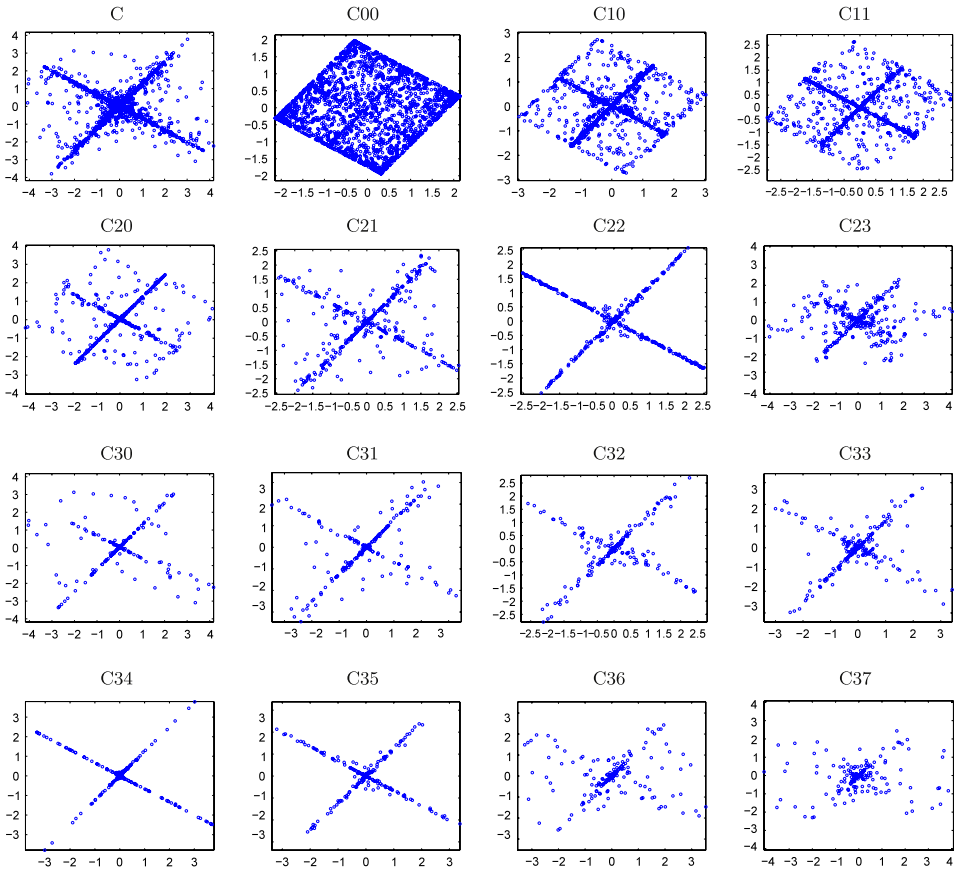


FIGURE 10.11

Scatter plots of the WP coefficients of the FM mixtures.

plots of sets of coefficients indexed in a wavelet packets tree. Generally speaking, the more distinct the directions appearing on these plots, the more precise the estimation of the mixing matrix, and, therefore, the better the separation.

It is difficult to decide in advance which nodes contain the sparsest sets of coefficients. That is why one can use the following simple adaptive approach.

First, apply a clustering algorithm for every node of the tree and compute a measure of clusters' distortion (for example, the mean squared distance of data points to the centers of their own clusters. Here again, the weights of the data points can be incorporated. Second, choose a few best nodes with the minimal distortion, combine their coefficients into one data set, and apply a separation algorithm to these data.

More sophisticated techniques dealing with adaptive choice of best nodes, as well as their number, can be found in [63,62].

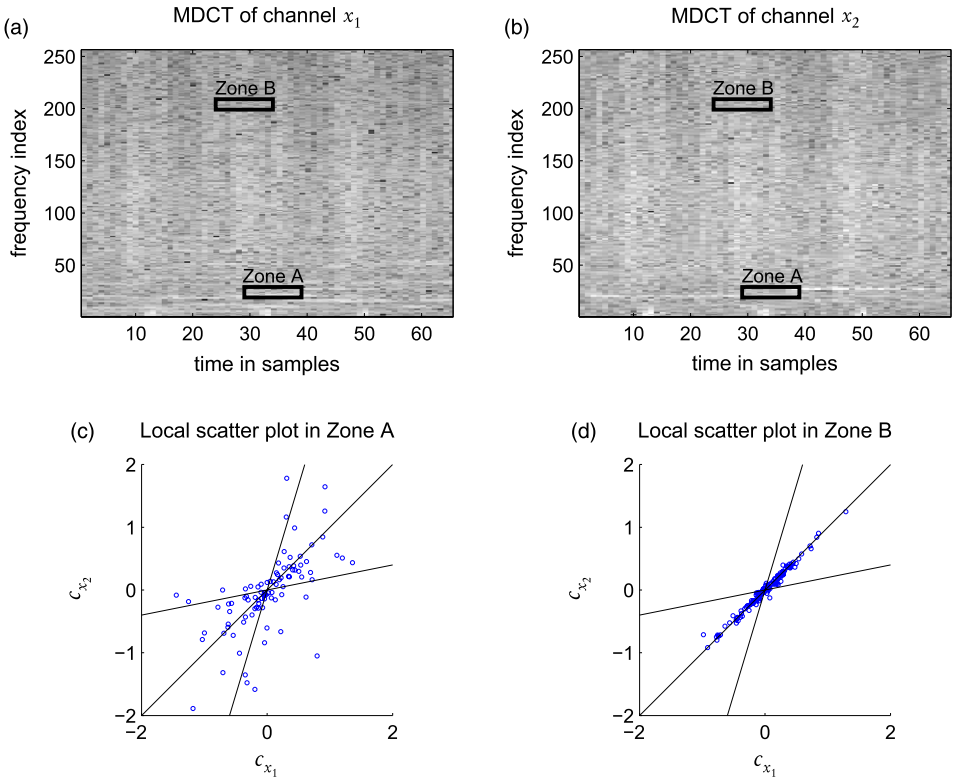


FIGURE 10.12

Top: MDCTs c_{x_1} and c_{x_2} of the two channels x_1 and x_2 of the noiseless stereophonic mixture with three sources represented in Fig. 10.1(c), with grey-level used to show intensity (from black for time-frequency zones with low energy to white for zones with higher energy). Two time-frequency zones (Zone A and Zone B) are indicated. Bottom: corresponding local scatter plots: (a) Zone A, where several sources are simultaneously active; (b) Zone B, where one source is dominant. On average, Zone A points have higher energies than Zone B points, therefore contributing more to the weighted histograms. However, Zone B points provide a more reliable estimation of direction.

10.4.3 Use of local scatter plots in the time-frequency plane

Despite the visible presence of alignments along the mixing matrix columns in the global scatter plot, the most traditional clustering algorithms such as K-means [110] can prove difficult to use for estimating the matrix. While a K-means algorithm is particularly simple to implement, the results depend on proper initialization and correctly determining beforehand the number of sources present. Using the global scatter plot is even more difficult when the mixed sources contribute with very diverse intensities to the mixture, or when their representations are not sufficiently disjoint, as when several musical instruments play in time and in harmony, thereby producing simultaneous activity at certain times and frequencies.

Figure 10.12 illustrates this phenomenon, showing the time-frequency representations (MDCT) of each channel of a stereophonic mixture, as well as the *local scatter plots*

$\{C_{\mathbf{x}}(k)\}_{k\in\Lambda}$ where Λ corresponds to a subset of time-frequency atoms close to a central time and frequency. In the first local scatter plot, one cannot see any distinct alignment along a column of the underlying mixing matrix. In the second plot, on the contrary *all* the points are distinctly aligned around *a single* mixture direction, that of the source whose activity is dominant throughout the corresponding time-frequency zone. The direction of this source can thus be identified using the local scatter plot.

If this type of zone can be detected, $\mathbf{A}$ can be estimated, which is the principle of the methods proposed by Deville *et al*. [3,38,2,88]: the variance of the ratio $c_{x_1}(k)/c_{x_2}(k)$ is calculated for each region considered, then regions with lower variance are selected to estimate the corresponding column directions of $\mathbf{A}$. A similar approach, but one where the different channels play a more symmetrical role, consists in performing Principle Component Analysis (PCA) based on the local scatter plot and selecting the zones whose main direction is the most dominant relative to the others [8]. Techniques based on selecting "simple autoterms" of bilinear time-frequency transforms [39] are also very close to this principle.

10.5 SQUARE MIXING MATRIX: RELATIVE NEWTON METHOD

When the mixing matrix is square non-degenerative, one can perform BSS by any of the methods presented in other chapters of this book. However, use of sparsity prior can greatly improve separation quality in this case as well [115,117]. We consider the model

$$C_{\mathbf{x}} = \mathbf{A}C_{\mathbf{s}} \tag{10.40}$$

where the mixture coefficients $C_{\mathbf{x}}$ can be obtained via analysis transform of the mixture signals (10.14) or, even better, by joint sparse approximation (10.17). The equation above can be treated as a standard BSS problem with $C_{\mathbf{s}}$, $C_{\mathbf{x}}$ and the coefficient index k substituted by S, X and time index t:

$$X = \mathbf{A}S. \tag{10.41}$$

Now we just say that the sources S are sparse themselves. We will use log-likelihood BSS model [81,19] parameterized by the separating matrix $\mathbf{B} = \mathbf{A}^{-1}$. If we assume the sources to be iid, stationary and white, the normalized negative-log-likelihood of the mixtures is

$$L(\mathbf{B};X) = -\log|\det\mathbf{B}| + \frac{1}{T}\sum_{i,k} h(b_i x(t)), \tag{10.42}$$

where b_i is i-th row of $\mathbf{B}$, $h(\cdot) = -\log f(\cdot)$, and $f(\cdot)$ is the probability density function (pdf) of the sources. A consistent estimator can be obtained by the minimization of (10.42), also when $h(\cdot)$ is not exactly equal to $-\log f(\cdot)$ [84]. Such *quasi-ML estimation* is practical when the source pdf is unknown, or is not well-suited for optimization. For example, when the sources are sparse, the absolute value function or its smooth

approximation is a good choice for $h(\cdot)$ [23,80,71,115,117,114]. Among other convex smooth approximations to the absolute value, one can use

$$h_1(c) = |c| - \log(1 + |c|) \tag{10.43}$$

$$h_\lambda(c) = \lambda h_1(c/\lambda) \tag{10.44}$$

with λ a proximity parameter: $h_\lambda(c) \to |c|$ as $\lambda \to 0^+$. The widely accepted natural gradient method does not work well when the approximation of the absolute value becomes too sharp. Below we describe the Relative Newton method [112,113], which overcomes this obstacle. The algorithm is similar in part to the Newton method of Pham and Garat [84], while enriched by positive definite Hessian modification and line search, providing global convergence.

10.5.1 Relative optimization framework

Suppose one has some current estimate of the separating matrix $\mathbf{B}_k$. One can think about the current source estimate

$$S_k = \mathbf{B}_k X \tag{10.45}$$

as a mixture in some new BSS problem

$$S_k = \mathbf{A}_k S$$

where S are the actual sources to be found. This new problem can be approximately solved by one or several steps of some separation method, leading to the next BSS problem, so on. One uses such an idea for the minimization of quasi-ML function (10.42):

- Start with an initial estimate $\mathbf{B}_0$ of the separation matrix;
- **For** $k = 0, 1, 2, \ldots$, Until convergence
  1. Compute current source estimate $S_k = \mathbf{B}_k X$;
  2. For the new BSS problem $S_k = \mathbf{A}_k S$, decrease sufficiently its quasi-ML function $L(V; S_k)$. Namely, get "local" separation matrix V_{k+1} by one or a few steps of a conventional optimization method (starting with $V = I$).
  3. Update the overall separation matrix

$$\mathbf{B}_{k+1} = V_{k+1} \mathbf{B}_k; \tag{10.46}$$

The Relative (or Natural) Gradient method [25,7,22], described in this book in Chapter 4, *Likelihood* and Chapter 6, *Iterative Algorithms*, is a particular instance of this approach, when the standard gradient descent step is used in item 2. The following remarkable *invariance* property of the Relative Gradient method is also preserved in general:

> *Given current source estimate S_k, the trajectory of the method and the final solution do not depend on the original mixing matrix.*

This means that even ill-conditioned mixing matrix does not influence the convergence of the method any more than does a starting point, and does not influence the final solution at all. Therefore one can analyze local convergence of the method assuming the mixing matrix to be close to the identity. In this case, when $h(c)$ is smooth enough, the Hessian of (10.42) is well-conditioned, and high linear convergence rate can be achieved even with the Relative Gradient method. However, if we are interested in $h(c)$ being less smooth (for example, using small λ in modulus approximation (10.43)), use of a Newton step in item 2 of Relative Optimization become rewarding.

A natural question arises: how can one be sure about the global convergence in terms of "global" quasi-log-likelihood $L(\mathbf{B};X)$ given by (10.42) just reducing the "local" function $L(V;S_k)$? The answer is:

One-step reduction of the "local" quasi-log-likelihood leads to the equal reduction of the "global" one:

$$L(\mathbf{B}_k;X)-L(\mathbf{B}_{k+1};X)=L(I;S_k)-L(V_{k+1};S_k).$$

This can be shown by subtracting the following equalities obtained from (10.42), (10.45) and (10.46)

$$L(\mathbf{B}_k;X)=-\log|\det\mathbf{B}_k|+L(I;S_k) \tag{10.47}$$
$$L(\mathbf{B}_{k+1};X)=-\log|\det\mathbf{B}_k|+L(V_k;S_k). \tag{10.48}$$

10.5.2 Newton method

The Newton method is an efficient tool of unconstrained optimization. It converges much faster than the gradient descent when the function has a narrow valley, and provides a quadratic rate of convergence. However, its iteration may be costly, because of the necessity to compute the Hessian matrix of the mixed second derivatives and to solve the corresponding system of linear equations. In the next subsection we will see how this difficulty can be overcome using the Relative Newton method, but first we describe the Modified Newton method in a general setting.

Modified Newton method with a line search

Suppose that we minimize an arbitrary twice-differentiable function $f(x)$ and x_k is the current iterate with the gradient $g_k=\nabla f(x_k)$ and the Hessian matrix $H_k=\nabla^2 f(x_k)$. The Newton step attempts to find a minimum of the second-order Taylor expansion of f around x_k

$$y_k=\arg\min_y\left\{q(x_k+y)=f(x_k)+g_k^T y+\frac{1}{2}y^T H_k y\right\}.$$

The minimum is provided by solution of the Newton system

$$H_k y_k=-g_k.$$

When $f(x)$ is not convex, the Hessian matrix is not necessarily positive definite; therefore the direction y_k is not necessarily a direction of descent. In this case we use the Modified Cholesky factorization[6] [46], which automatically finds a diagonal matrix R such that the matrix $H_k + R$ is positive definite, providing a solution to the modified system

$$(H_k + R)y_k = -g_k. \tag{10.49}$$

After the direction y_k is found, the new iterate x_{k+1} is given by

$$x_{k+1} = x_k + \alpha_k y_k \tag{10.50}$$

where the step size α_k is determined by exact line search

$$\alpha_k = \arg\min_{\alpha} f(x_k + \alpha y_k) \tag{10.51}$$

or by a backtracking line search (see for example [46]):

$$\alpha := 1;\ \text{WHILE}\ f(x_k + \alpha y_k) > f(x_k) + \beta\alpha g_k^T y_k, \quad \alpha := \gamma\alpha$$

where $0 < \beta < 1$ and $0 < \gamma < 1$. The use of line search guarantees monotone decrease of the objective function at every Newton iteration. Our typical choice of the line search constants is $\beta = \gamma = 0.3$. It may also be reasonable to give β a small value, like 0.01.

10.5.3 Gradient and Hessian evaluation

The likelihood $L(\mathbf{B};X)$ given by (10.42) is a function of a matrix argument $\mathbf{B}$. The corresponding gradient with respect to $\mathbf{B}$ is also a matrix

$$G(\mathbf{B}) = \nabla L(\mathbf{B};X) = -\mathbf{B}^{-T} + \frac{1}{T} h'(\mathbf{B}X)X^T, \tag{10.52}$$

where $h'(\cdot)$ is used as an element-wise matrix function. The Hessian of $L(\mathbf{B};X)$ is a linear mapping (4D tensor) $\mathscr{H}$ defined via the differential of the gradient $dG = \mathscr{H}d\mathbf{B}$. We can also express the Hessian in standard matrix form converting $\mathbf{B}$ into a long vector $b = \text{vec}(\mathbf{B})$ using row stacking. We will denote the reverse conversion $\mathbf{B} = \text{mat}(b)$. Denote

$$\hat{L}(b,X) \equiv L(\text{mat}(b),X), \tag{10.53}$$

with the gradient

$$g(b) = \nabla\hat{L}(b;X) = \text{vec}(G(\mathbf{B})). \tag{10.54}$$

[6] MATLAB code of modified Cholesky factorization by Brian Borchers, available at http://www.nmt.edu/ ~ borchers/ldlt.html.

The Hessian of $-\log|\det \mathrm{mat}(b)|$ is determined using i-th column A^i and j-th row A_j of $\mathbf{A} = \mathbf{B}^{-1}$. Namely, the r-th column of H, $r = (i-1)N + j$, contains the matrix $(A^i A_j)^T$ stacked row-wise (see [112,113]):

$$H^r = \mathrm{vec}(A^{j\,T} A_i^T). \tag{10.55}$$

The Hessian of the second term in $\hat{L}(b, X)$ is a block-diagonal matrix with the following $N \times N$ blocks constructed using the rows B_m of $\mathbf{B}$:

$$\frac{1}{T}\sum_t h''(B_m x(t))x(t)x^T(t), \quad m = 1,\ldots,N. \tag{10.56}$$

Hessian simplifications in Relative Newton method

Relative Newton iteration k uses the Hessian of $L(I, S^k)$. Hessian of $-\log|\det \mathbf{B}|$ given by (10.55), becomes very simple and sparse, when $\mathbf{B} = \mathbf{A} = I$. Each column of H contains only one non-zero element, which is equal to 1:

$$H^r = \mathrm{vec}(e_j e_i^T), \tag{10.57}$$

where e_j is an N-element standard basis vector, containing 1 at the j-th position and zeros at others. The second block-diagonal term of the Hessian (10.56) also simplifies greatly when $X = S^k \to S$. When we approach the solution, and the sources are independent and zero mean, the off-diagonal elements in (10.56) converge to zero as sample size grows. As a result, we have only two non-zero elements in every row of the Hessian: one from $\log|\det(\cdot)|$ and another from the diagonal of the second term. After reordering the variables

$$v = [V_{11}, V_{12}, V_{21}, V_{13}, V_{31}, \ldots, V_{22}, V_{23}, V_{32}, V_{24}, V_{42}, \ldots V_{NN}]^T$$

we get the Hessian with diagonal 2×2 and 1×1 blocks, which can be inverted very fast.

In order to guarantee a descent direction and avoid saddle points, we substitute the Hessian with a positive definite matrix: change the sign of the negative eigenvalues (see for example [46]), and force the small eigenvalues to be above some threshold (say, 10^{-8} of the maximal one).

10.5.4 Sequential optimization

When the sources are sparse, the quality of separation greatly improves with reduction of smoothing parameter λ in the modulus approximation (10.44). On the other hand, the optimization of the likelihood function becomes more difficult for small λ. Therefore, we first optimize the likelihood with some moderate λ, then reduce λ by a constant factor (say, 10 or 100), and perform optimization again, and so on. This *Sequential Optimization* approach reduces the overall computations considerably.

Smoothing Method of Multipliers (SMOM)

Even gradual reduction of the smoothing parameter may require a significant number of Newton steps after each update of λ. A more efficient way to achieve an accurate solution of a problem involving a sum of absolute value functions is to use the SMOM method, which is an extension of the Augmented Lagrangian technique [11,86,10] used in constrained optimization. It allows us to obtain an accurate solution without forcing the smoothing parameter λ to go to zero. SMOM can be efficiently combined with the Relative Optimization. We refer the reader to [113] for the exposition of this approach.

10.5.5 Numerical illustrations

Two data sets were used. The first group of sources was random sparse data with Gaussian distribution of non-zero samples, generated by the MATLAB function SPRANDN. The second group of sources consisted of four natural images from [24]. The mixing matrix was generated randomly with uniform iid entries.

Relative Newton method

Figure 10.13 shows the typical progress of different methods applied to the artificial data with 5 mixtures of 10k samples. The Fast Relative Newton method (with 2×2 block-diagonal Hessian approximation) converges in about the same number of iterations as the Relative Newton with exact Hessian, but significantly outperforms it in time. Natural gradient in batch mode requires many more iterations, and has difficulty in converging when the smoothing parameter λ in (10.44) becomes too small.

In the second experiment, we demonstrate the advantage of the batch-mode quasi-ML separation, when dealing with sparse sources. We compared the Fast Relative Newton method with stochastic natural gradient [25,7,22], Fast ICA [60] and JADE [20]. All three codes are available at public web sites [73,59,21]. Stochastic natural gradient and Fast ICA used tanh nonlinearity. Figure 10.14 shows separation of artificial stochastic sparse data: 5 sources of 500 samples, 30 simulation trials. As we see, Fast Relative Newton significantly outperforms other methods, providing practically ideal separation with the smoothing parameter $\lambda = 10^{-6}$ (sequential update of the smoothing parameter was used here). Timing is of about the same order for all the methods, except of JADE, which is known to be much faster with relatively small matrices.

Sequential Optimization vs Smoothing Method of Multipliers (SMOM)

In the third experiment we have used the first stochastic sparse data set: 5 mixtures, 10k samples. Figure 10.15 demonstrates the advantage of the SMOM combined with the frozen Hessian strategy [113]: the Hessian is computed once and then used in several Newton steps, while they are effective enough. As we see, the last six outer iterations do not require new Hessian evaluations. At the same time the Sequential Optimization method without Lagrange multipliers, requires 3 to 8 Hessian evaluations per outer iteration towards the end. As a consequence the method of multipliers converges much faster.

In the fourth experiment, we separated four natural images [24], presented in Fig. 10.16. Sparseness of images can be achieved via various wavelet-type transforms [115,117,114],

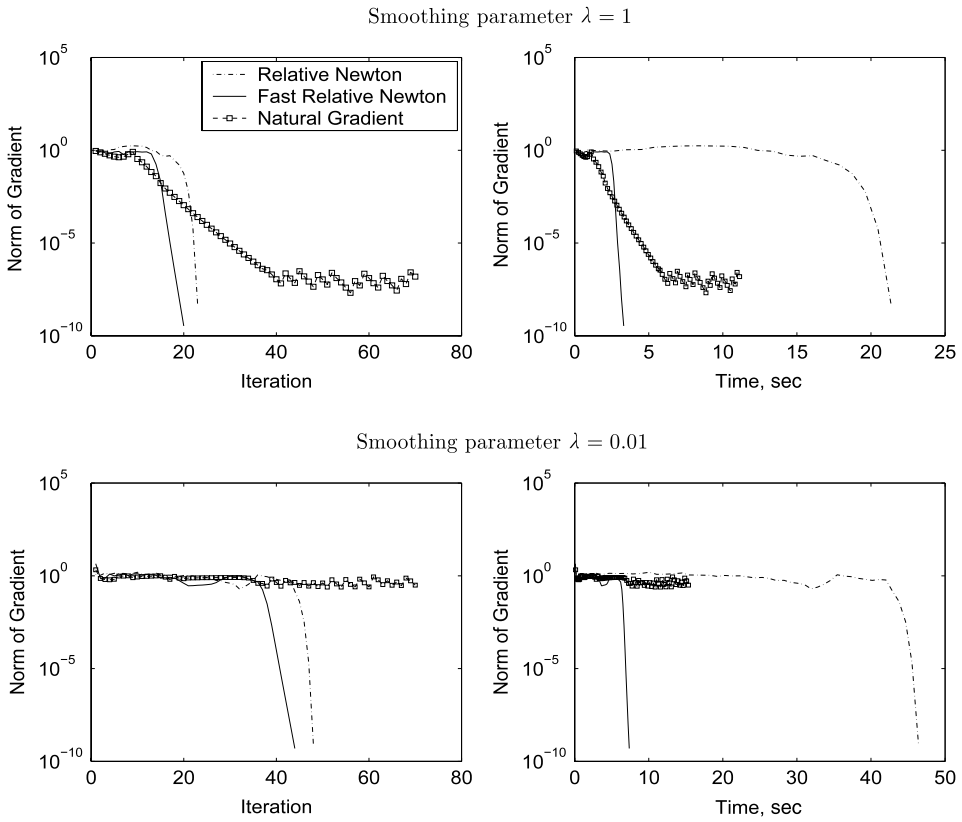


FIGURE 10.13

Separation of artificial sparse data with 5 mixtures by 10k samples: Relative Newton with exact Hessian (dashed line), Fast Relative Newton (continuous line), Natural Gradient in batch mode (squares).

but even simple differentiation can be used for this purpose, since natural images often have sparse edges. Here we used the stack of horizontal and vertical derivatives of the mixture images as an input to the separation algorithms. Figure 10.16 shows the separation quality achieved by stochastic natural gradient, Fast ICA, JADE, the Fast Relative Newton method with $\lambda = 10^{-2}$ and the SMOM. Like in the previous experiments, SMOM provides practically ideal separation with ISR of about 10^{-12}. It outperforms the other methods by several orders of magnitude.

10.5.6 Extension of Relative Newton: blind deconvolution

We only mention two important extensions of the Relative Newton method. First, the block-coordinate version of the method [16] has an improved convergence rate for large problems. Second, the method can be modified for blind deconvolution of signals and images [17,18].

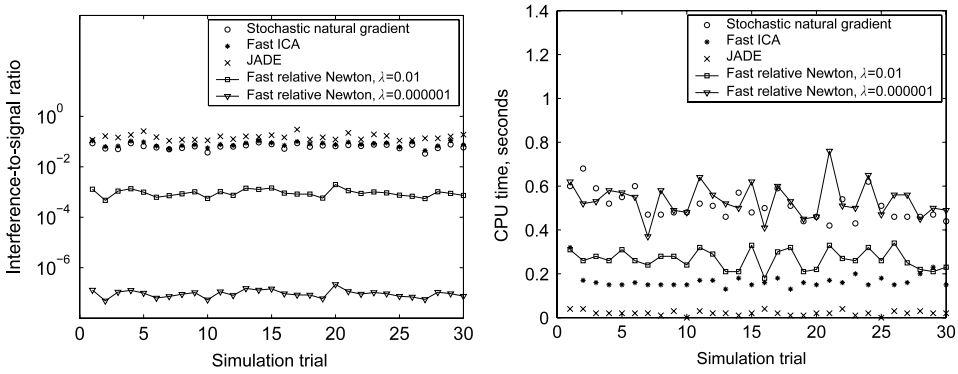


FIGURE 10.14

Separation of stochastic sparse data: 5 sources of 500 samples, 30 simulation trials. Left – interference-to-signal ratio, right – CPU time.

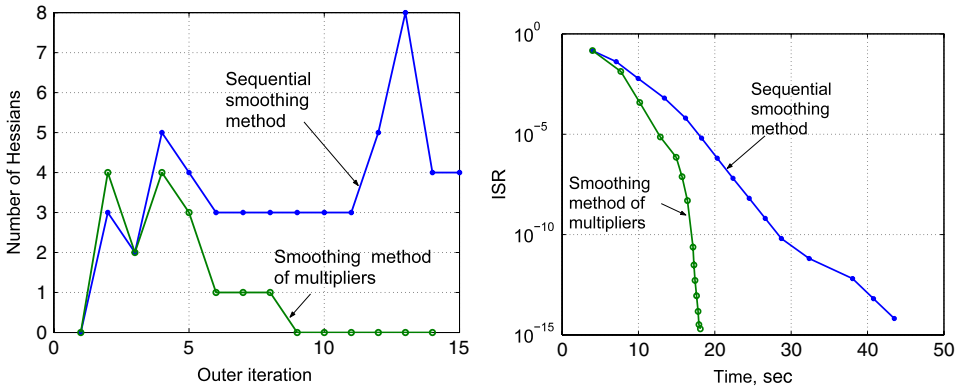


FIGURE 10.15

Relative Newton method with frozen Hessian. Left – number of Hessian evaluations per outer iteration (the first data set); Right – interference-to-signal ratio. Dots correspond to outer iterations.

10.6 SEPARATION WITH A KNOWN MIXING MATRIX

The first two steps of a sparse source separation system (Fig. 10.2) involve calculating:

- a sparse representation/approximation $C_{\mathbf{x}}$ such that $\mathbf{x} \approx C_{\mathbf{x}}\boldsymbol{\Phi}$;
- an estimation of the mixing matrix $\widehat{\mathbf{A}}$.

The sources are finally separated in the last step. According to the type of mixture (determined or not), several approaches are possible, depending more or less on the sparsity and disjointedness of the source representations.

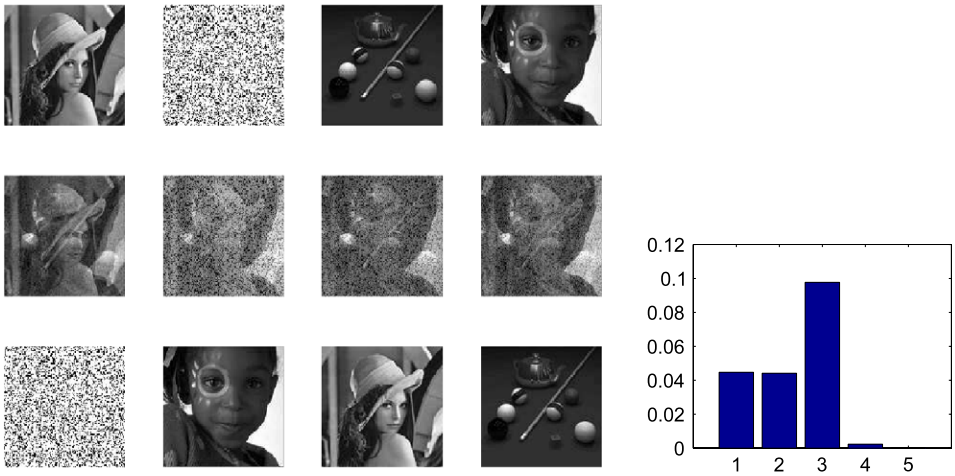


FIGURE 10.16

Separation of images with preprocessing by differentiation. Left: top – sources, middle – mixtures, bottom – separated. Right: Interference-to-signal ratio (ISR) of image separation. 1 – stochastic natural gradient; 2 – Fast ICA; 3 – JADE; 4 – Relative Newton with $\lambda = 10^{-2}$; 5 – SMOM (bar 5 is not visible because of very small ISR, of order 10^{-12}.)

10.6.1 Linear separation of (over-)determined mixtures

When $\widehat{\mathbf{A}}$ is an invertible square matrix, its inverse $\mathbf{B} \coloneqq \widehat{\mathbf{A}}^{-1}$ can be applied to the mixture $\mathbf{x} = \mathbf{A}\mathbf{s}$, assumed noisefree, to obtain $\widehat{s} \coloneqq \widehat{\mathbf{A}}^{-1}\mathbf{x} = \widehat{\mathbf{A}}^{-1}\mathbf{A}\mathbf{s}$. An alternative consists in first estimating a representation of the sources $C_{\widehat{\mathbf{s}}} \coloneqq \widehat{\mathbf{A}}^{-1}C_{\mathbf{x}}$, then estimating the sources by reconstruction, $\widehat{\mathbf{s}} \coloneqq C_{\widehat{\mathbf{s}}}\mathbf{\Phi}$. In the over-determined case where the number of sensors exceeds the number of sources, the pseudo-inverse $\mathbf{B} \coloneqq \widehat{\mathbf{A}}^{\dagger}$ can be used. These two alternatives (separation of the initial mixture or its representation) are strictly identical provided that $C_{\mathbf{x}}$ is an *exact representation* of the mixture $\mathbf{x}$. This is not true when $C_{\mathbf{x}}$ is only a *sparse approximation* of mixture $\mathbf{x} \approx C_{\mathbf{x}}\mathbf{\Phi}$; whereas the first approach guarantees that $\mathbf{x} = \widehat{\mathbf{A}}\widehat{\mathbf{s}}$, the second approach only provides the approximation $\mathbf{x} \approx C_{\mathbf{x}}\mathbf{\Phi} = \widehat{\mathbf{A}}C_{\widehat{\mathbf{s}}}\mathbf{\Phi} = \widehat{\mathbf{A}}\widehat{\mathbf{s}}$. For (over-)determined separation problems, separation in the sparse domain is potentially interesting only when the mixture to separate is noisy, in which case obtaining an approximation $\mathbf{x} \approx \widehat{A}\widehat{\mathbf{s}}$ can effectively denoise the mixture.

Sparse source separation can be particularly useful for separating under-determined mixtures where the number of sources N exceeds the number of sensors P. In this case, it is often much more efficient than any of the linear separation methods, which estimate the sources using a separation matrix $\mathbf{B}$ taking the form $\widehat{\mathbf{s}} = \mathbf{B}\mathbf{x}$. Simple linear algebra arguments show that it is not possible to perfectly separate the sources linearly in the under-determined case, and more specifically [51] that the estimation of at least one of the sources necessarily undergoes a level of residual interference from the other sources

of around $10\log_{10}(N/P-1)$ decibels [51, Lemma 1]. In the following discussion, we focus on sparse methods that can overcome the limits of linear separation. They all start by estimating a representation $C_{\widehat{\mathbf{s}}}$ of the sources before reconstructing the estimated sources as $\widehat{\mathbf{s}} = C_{\widehat{\mathbf{s}}}\mathbf{\Phi}$. Depending on the case, this representation is obtained directly from the initial mixture $\mathbf{x}$ and the estimated matrix $\widehat{\mathbf{A}}$ (sections 10.6.5 to 10.6.8), or from $\widehat{\mathbf{A}}$ and the joint sparse representation $C_{\mathbf{x}}$ of the mixture (sections 10.6.2 to 10.6.4), which assumes that the identity $C_{\mathbf{x}} \approx \widehat{\mathbf{A}} C_{\mathbf{s}}$ is reasonably verified.

10.6.2 Binary masking assuming a single active source

The source model with disjoint support representations described in the introduction corresponds to estimating the representation of each source by means of masking, based on the calculated representation of the mixture $C_{\mathbf{x}}$:

$$c_{\widehat{s}_n}(k) \coloneqq \chi_n(k) \cdot \frac{\widehat{\mathbf{A}}_n^{\mathrm{H}} C_{\mathbf{x}}(k)}{||\widehat{\mathbf{A}}_n||_2^2} \tag{10.58}$$

where the mask $\chi_n(k)$ is defined as:

$$\chi_n(k) \coloneqq \begin{cases} 1 & \text{if } n = n(k); \\ 0 & \text{if not.} \end{cases} \qquad \text{with } n(k) \coloneqq \arg\max_n \frac{|\widehat{\mathbf{A}}_n^{\mathrm{H}} C_{\mathbf{x}}(k)|}{||\widehat{\mathbf{A}}_n||_2^2}; \tag{10.59}$$

that is, only the most active source is attributed a non-zero coefficient, whereas the coefficients of the other sources are set to zero, or "masked". The degree of a source's activity, which determines the mask, is measured as the correlation between the corresponding column $\widehat{\mathbf{A}}_n$ of the estimated mixing matrix and the component $C_{\mathbf{x}}(k)$ of the mixture.

Successful masking depends on the sources mainly having disjoint representations, effectively resulting in the approximation $C_{\mathbf{x}}(k) \approx \widehat{\mathbf{A}}_{n(k)} c_{s_n}(k)$. Although a strong assumption, this can be checked for many audiophonic mixtures [9] in the time-frequency domain. It is the main reason for the good results obtained with the DUET algorithm [111], which performs a variant of masking in the time-frequency domain where the mask $\chi_n(k)$ is applied to one of the channels c_{x_p} of the mixture, rather than to the linear combination of all of the channels $\widehat{\mathbf{A}}_n^{\mathrm{H}} C_{\mathbf{x}}$. Inversely, masking $\widehat{\mathbf{A}}_n^{\mathrm{H}} C_{\mathbf{x}}$ has the advantage of guaranteeing that the source to be estimated is in fact present in the masked "virtual channel", whereas a source not present in a given channel cannot be estimated by masking this channel.

10.6.3 Binary masking assuming $M < P$ active sources

When the number of mixed sources N is high, the mean number of sources simultaneously active can exceed 1, despite the sparse representation of each source. Assuming the

number of active sources does not exceed M, where $M < P$, an approach [50,6] generalizing the binary masking described above involves:

1. determining the set $I_M(k) \subset \{1,\ldots,N\}$ of indices of M active sources in the $C_{\mathbf{x}}(k)$ component;
2. using least squares to calculate the components of the active sources.

Denoting as $\widehat{\mathbf{A}}_I$ the matrix composed of $\widehat{\mathbf{A}}$ columns with index $k \in I$, if the set of active sources is I, the estimated components of the sources are:

$$c_{\widehat{s}_n}(k) \quad := 0, \qquad k \notin I, \tag{10.60}$$

$$\left(c_{\widehat{s}_n}(k)\right)_{k\in I} := \widehat{\mathbf{A}}_I^{\dagger} C_{\mathbf{x}}. \tag{10.61}$$

The set $I_M(k)$ of active sources is thus chosen to minimize the mean square error of reconstructing $C_{\mathbf{x}}$:

$$I_M(k) := \arg \min_{I|\mathrm{card}(I)\leq M} \|C_{\mathbf{x}} - \widehat{\mathbf{A}}_I \widehat{\mathbf{A}}_I^{\dagger} C_{\mathbf{x}}\|_2^2 \tag{10.62}$$

$$= \arg \max_{I|\mathrm{card}(I)\leq M} \|\widehat{\mathbf{A}}_I \widehat{\mathbf{A}}_I^{\dagger} C_{\mathbf{x}}\|_2^2. \tag{10.63}$$

When $M = 1$, the principle of binary masking applies exactly as described above.

10.6.4 Local separation by ℓ^τ minimization

When the number of sources assumed active can reach $M = P$, it is no longer possible to define the set $I(k)$ of active sources by mean square error minimization since generically, with any subset I of P active sources, a perfect reconstruction is obtained, $C_{\mathbf{x}} = \widehat{\mathbf{A}}_I \widehat{\mathbf{A}}_I^{-1} C_{\mathbf{x}}$.

An alternative is to exploit the sparsity of the sources by a maximum likelihood approach that assumes a generalized Gaussian distribution of the source coefficients:

$$C_{\widehat{\mathbf{s}}}(k) := \arg \min_{C_{\mathbf{s}}(k)|\widehat{\mathbf{A}}C_{\mathbf{s}}(k)=C_{\mathbf{x}}(k)} \sum_{n=1}^{N} |c_{\mathbf{s}_n}(k)|^{\tau}. \tag{10.64}$$

which for $0 \leq \tau \leq 1$ results in estimating a source representation where, for each atom φ_k, no more than P sources are "active" [64]. In practice, rather than use the heavy artillery of iterative algorithms for ℓ^τ norm optimization (FOCUSS, etc.) to solve (10.64), a combinatorial approach can be employed in low dimension, which involves selecting the best set of P active sources as:

$$I(k) := \arg \min_{I|\mathrm{card}(I)=P} \|\widehat{\mathbf{A}}_I^{-1} C_{\mathbf{x}}\|_{\tau}. \tag{10.65}$$

For greater numerical efficiency, the $\binom{N}{P}$ possible inverse matrices $\widehat{\mathbf{A}}_I^{-1}$ can be precalculated as soon as the mixing matrix is estimated.

This approach is widely used, particularly for separating audio sources based on their STFTs, and provides good results overall. For stereophonic audio mixtures, Bofill and Zibulevsky [15] proposed this sort of local optimization for $\tau = 1$, and Saab *et al.* [93] experimentally studied how the choice of τ affects the quality of the results, for sound mixtures. For the examples they considered, their results are satisfactory in terms of separation quality, and do not vary substantially with the chosen exponent τ.

For noisy mixtures, the separation criterion (10.68) can be replaced to advantage by the following, where the λ parameter adjusts the compromise between faithful reconstruction of the mixture coefficients and sparsity of the estimated sources:

$$C_{\hat{\mathbf{s}}}(k) \coloneqq \arg\min_{C_{\mathbf{s}}(k)} \left\{ \frac{1}{2\sigma_{\mathbf{b}}^2} \|C_{\mathbf{x}}(k) - \widehat{\mathbf{A}} C_{\mathbf{s}}(k)\|_F^2 + \lambda \sum_{n=1}^{N} |c_{\mathbf{s}_n}(k)|^{\tau} \right\}. \tag{10.66}$$

Depending on the value of λ, the number of active estimated sources varies from zero to P. As the reader may have noticed, for an exponent $\tau = 0$, the system of masking presented in section 10.6.3 applies, but the number of active sources can vary for each component.

10.6.5 Principle of global separation by ℓ^{τ} minimization

When the $\mathbf{\Phi}$ dictionary is an orthonormal basis and a simple orthogonal transform is used to calculate the joint mixture representation, the independent estimation of source components according to the criterion (10.66) actually corresponds to optimizing a more global criterion:

$$C_{\hat{\mathbf{s}}} \coloneqq \arg\min_{C_{\mathbf{s}}} \left\{ \|\mathbf{x} - \widehat{\mathbf{A}} C_{\mathbf{s}} \mathbf{\Phi}\|_F^2 + \lambda \sum_{nk} \left| c_{s_n}(k) \right|^{\tau} \right\}. \tag{10.67}$$

This results in the equation:

$$\begin{aligned} \|\mathbf{x} - \widehat{\mathbf{A}} C_{\mathbf{s}} \mathbf{\Phi}\|_F^2 &= \|\mathbf{x}\mathbf{\Phi}^{\mathrm{H}} - \widehat{\mathbf{A}} C_{\mathbf{s}}\|_F^2 = \|C_{\mathbf{x}}^{\mathbf{\Phi}^{\mathrm{H}}} - \widehat{\mathbf{A}} C_{\mathbf{s}}\|_F^2 \\ &= \sum_{k=1}^{K} \|C_{\mathbf{x}}^{\mathbf{\Phi}^{\mathrm{H}}}(k) - \widehat{\mathbf{A}} C_{\mathbf{s}}(k)\|_2^2. \end{aligned}$$

For an arbitrary dictionary, calculating sources via the criterion (10.67) amounts to a *maximum a posteriori (MAP)* estimation with the assumption of a generalized Gaussian distribution of source coefficients (10.4). In the same vein, Zibulevsky and Pearlmutter [115] popularized the separation of under-determined mixtures by using a sparse source model and MAP estimation:

$$C_{\hat{\mathbf{s}}} \coloneqq \arg\max_{C_{\mathbf{s}}} p(C_{\mathbf{s}} | \mathbf{x}, \widehat{\mathbf{A}}) = \arg\min_{C_{\mathbf{s}}} \left\{ \frac{1}{2\sigma_{\mathbf{b}}^2} \|\mathbf{x} - \widehat{\mathbf{A}} C_{\mathbf{s}} \mathbf{\Phi}\|_F^2 + \sum_{nk} h(c_{s_n}(k)) \right\} \tag{10.68}$$

with $h(c) = -\log p_c(c)$ where $p_c(\cdot)$ is the common distribution of source coefficients, assumed independent. Note that the estimation of $C_{\hat{\mathbf{s}}}$ does not involve the representation

$C_{\mathbf{x}}$ of the mixture, calculated in an initial step (see Fig. 10.2). Here this representation only serves to estimate the mixing matrix. Since optimization of a criterion such as (10.68) or (10.67) is only a principle, we will now provide specific details on algorithms that apply it.

10.6.6 Formal links with single-channel traditional sparse approximation

It may come as a surprise that optimization problems such as (10.67) are formally more conventional than the problems of joint sparse mixture representation (10.16) and (10.17) mentioned in section 10.3. This formal link, while not essential for understanding the definition of iterative thresholding or *Matching Pursuit* algorithms for separation, is more critical for describing the FOCUSS algorithm in this context. Readers more interested in the description of algorithms that are conceptually and numerically simpler can skip to section 10.6.7.

Given that:

$$\widehat{\mathbf{A}}C_{\mathbf{s}}\mathbf{\Phi} = \sum_{nk} c_{s_n}(k)\widehat{\mathbf{A}}_n\varphi_k,$$

the optimization (10.67) consists in approximating the mixture $\mathbf{x}$ (considered as a $P \times T$ matrix) with a sparse linear combination of $P \times T$ matrices taking the form $\{\widehat{\mathbf{A}}_n\varphi_k\}_{nk}$.

In the vector space of the $P \times T$ matrices, the problem is one of sparse approximation of the "vector" $\mathbf{x}$ based on the dictionary Φ[7] comprising $N \times K$ "multi-channel atoms" $\{\widehat{\mathbf{A}}_n\varphi_k\}_{nk}$. Thus, the optimization algorithms for sparse representation of a *one-dimensional* signal do not require substantial modification to their formal expressions: iterative thresholding techniques [28,36] for $1 \leq \tau \leq 2$, linear, quadratic or conic programming [23,12] for $\tau = 1$, and iterative reweighted least squares (such as FOCUSS) for $0 < \tau \leq 2$. The detailed description of these algorithms makes use of the elements described in section 10.3, considering the specific case of $P = 1$, i.e. where the representations are no longer *joint*.

10.6.7 Global separation algorithms using ℓ^τ minimization

While things are simple on paper, the same is not necessarily true numerically: the thresholding algorithms are feasible because they involve no matrix inversion, but minimizing the ℓ^1 norm (*basis pursuit*) or the ℓ^τ norm (FOCUSS) quickly becomes impracticable because of the repeated inversions of $(P \times T) \times (P \times T)$ square matrices. By contrast, we will see that *matching pursuit* is easy to apply, with an algorithm that is efficient in practice.

Algorithm for $0 < \tau \leq 2$: FOCUSS and basis pursuit

The FOCUSS algorithm for minimizing the criterion (10.67) is iterative. Starting from an initialization $C_{\mathbf{s}}^{(0)}$, the value of which can decisively influence the algorithm's point

[7] The italicized notation Φ is used here to distinguish between this dictionary of matrices and the dictionary of atoms $\mathbf{\Phi}$ from which it is built. Whereas $\mathbf{\Phi}$ is a $K \times T$ matrix, Φ should hereafter be considered as a $(N \times K) \times (P \times T)$ matrix.

of convergence, the weighted mean square criterion is iteratively minimized:

$$C_{\mathbf{s}}^{(m)} := \arg\min_{C_{\mathbf{s}}} \left\{ ||\mathbf{x} - \widehat{\mathbf{A}} C_{\mathbf{s}} \mathbf{\Phi}||_F^2 + \frac{\lambda |\tau|}{2} \sum_{nk} \left| (w_{nk}^{(m)})^{-1} \, c_{s_n}(k) \right|^2 \right\} \tag{10.69}$$

where $w_{nk}^{(m)} := |c_{s_n}^{(m-1)}(k)|^{1-\tau/2}$. Denoting as $\mathscr{W}^{(m)}$ the $(N \times K) \times (N \times K)$ square weighting matrix whose diagonal elements are the $N \times K$ coefficients $w_{nk}^{(m)}$, each iteration consists in updating $\mathscr{W}^{(m)}$ as well as the weighted dictionary $\Phi^{(m)}$ in which the $N \times K$ atoms are the multi-channel signals $w_{nk}^{(m)} \widehat{\mathbf{A}}_n \varphi_k$, each of size $P \times T$. Using the description of the M-FOCUSS algorithm in the single-channel case, section 10.3.6, we then calculate

$$\mathscr{C}_{\mathbf{s}}^{(m)} := \mathscr{X} \left((\Phi^{(m)})^{\mathrm{H}} \, \Phi^{(m)} + \frac{\lambda |\tau|}{2} \mathbf{I}_{P \times T} \right)^{-1} (\Phi^{(m)})^{\mathrm{H}} \, \mathscr{W}^{(m)} \tag{10.70}$$

where $\mathscr{X}$ is none other than the $P \times T$ column vector obtained by reorganizing the coefficients of $\mathbf{x}$, and $\mathscr{C}$ is an $N \times K$ column vector whose reorganization as a matrix is precisely $C_{\mathbf{s}}^{(m)}$. This algorithm, which requires prior inversion of a $(P \times T) \times (P \times T)$ square matrix at each iteration can only be used in practice if the dictionary is sufficiently structured to permit blockwise inversion, which is the case when $\mathbf{\Phi}$ is an orthonormal basis. The quadratic programming algorithms used for *basis pursuit* raise the same difficulties.

Algorithms for $1 \leq \tau \leq 2$: iterative thresholding

Given the estimated mixing matrix $\widehat{\mathbf{A}}$ and the dictionary $\mathbf{\Phi}$ considered, the first step is to determine a real $\mu < \infty$ such that, for any $N \times K$ matrix $C_{\mathbf{s}}$, $||\widehat{\mathbf{A}} C_{\mathbf{s}} \mathbf{\Phi}||_F^2 \leq \mu ||C_{\mathbf{s}}||_F^2$. Based on some initialization $C_{\mathbf{s}}^{(0)}$ (often chosen as zero), this is followed by iteratively defining:

$$c_{s_n}(k)^{(m)} = S_{\frac{\lambda}{\mu}, \tau} \left(c_{\mathbf{s}_n}^{(m-1)}(k) + \frac{\langle \widehat{\mathbf{A}}_n^{\mathrm{H}} (\mathbf{x} - \widehat{\mathbf{A}} C_{\mathbf{s}}^{(m-1)} \mathbf{\Phi}), \varphi_k \rangle}{\mu} \right) \tag{10.71}$$

and the resulting sequence necessarily converges to a minimizer of (10.67) as soon as $1 \leq \tau \leq 2$ [28]. If $\tau > 1$ or if $\mathbf{\Phi}$ *and* $\widehat{\mathbf{A}}$ are *both* invertible square matrices, this minimizer is also unique [28].

10.6.8 Iterative global separation: demixing pursuit

Algorithms for solving *global* ℓ^τ optimization problems such as (10.67) for $0 < \tau \leq 1$ are extremely time and memory hungry, especially if the data to be separated are high-dimensional and the computing time or power requirements are considerable. A much faster alternative approach is *demixing pursuit* [56,67], a type of *matching pursuit* applied

directly to the "vector" $\mathbf{x}$ with the atom dictionary $\{\widehat{\mathbf{A}}_n \varphi_k\}_{nk}$. The definition of this algorithm assumes that the atoms are of unit energy, which amounts to assuming the single-channel atoms φ_k are of unit energy and the columns $\widehat{\mathbf{A}}_n$ of the mixing matrix are also of unit norm. This last assumption is not restrictive since it is compatible with the natural indeterminations of source separation [19]. Theoretical research [56] indicates that if (10.67) has a sufficiently sparse solution, then *demixing pursuit* as well as ℓ^1 norm minimization can be used to find it.

More explicitly, to decompose and separate a multi-channel mixture using *demixing pursuit*, the first step is to initialize a residual $\mathbf{r}^{(0)} = \mathbf{x}$, followed by iteration of the following steps starting with $m = 1$:

1. Select the most correlated mixing matrix column and atom:

$$(n_m, k_m) := \arg\max_{nk} |\langle \widehat{\mathbf{A}}_n^{\mathrm{H}} \mathbf{r}_p^{(m-1)}, \varphi_k \rangle|. \tag{10.72}$$

2. Update the residual:

$$\mathbf{r}^{(m)} := \mathbf{r}^{(m-1)} - \langle \widehat{\mathbf{A}}_{n_m} \mathbf{r}_p^{(m-1)}, \varphi_{k_m} \rangle \widehat{\mathbf{A}}_{n_m} \varphi_{k_m}. \tag{10.73}$$

The decomposition below is obtained after M iterations:

$$\mathbf{x} = \sum_{m=1}^{M} c(m) \widehat{\mathbf{A}}_{n_m} \varphi_{k_m} + \mathbf{r}^{(M)} \tag{10.74}$$

with $c(m) := \langle \widehat{\mathbf{A}}_{n_m} \mathbf{r}^{(m-1)}, \varphi_{k_m} \rangle$.

Variants similar to orthogonal *matching pursuit* are possible, and in terms of numerical calculation, *demixing pursuit* can be implemented very efficiently by exploiting the dictionary's structure and the rapid updates for finding the largest scalar product from one iteration to the next. An open, rapid implementation is available [53,65].

10.7 CONCLUSION

In this chapter we have described the main steps of source separation methods based on sparsity. One of the primary advantages these methods offer is the possibility to separate under-determined mixtures, which involves two independent steps in most methods [99]: (a) estimating the mixing matrix; (b) separation using the known mixing matrix. In practice, as shown in Fig. 10.2, these methods usually involve four steps:

1. joint sparse representation $C_{\mathbf{x}}$ of the channels of the mixture $\mathbf{x}$;
2. estimation of $\mathbf{A}$ based on the scatter plot $\{C_{\mathbf{x}}(k)\}$;
3. sparse separation in the transform domain;
4. reconstruction of the sources.

Implementing a method of sparse source separation requires making a certain number of choices.

Choosing the dictionary Φ. This choice is essentially based on expert knowledge of the class of signals considered. A typical choice for audio mixtures or seismic signals is a Gabor time-frequency dictionary, or a wavelet time-scale dictionary; this is also true for biological signals such as cardiac or electro-encephalogram signals. For natural images, dictionaries of wavelets, curvelets or anisotropic Gabor atoms are generally used. Implementing methods to automatically choose a dictionary remains an open problem, even if techniques for learning dictionaries based on reference data sets are now available. The very concept of matching a dictionary to a class of signals is in need of further clarification.

Choosing the algorithm for the joint sparse representation C_x of the mixture. This choice depends *a priori* on several factors. Available computing power can rule out certain algorithms such as M-FOCUSS or *basis pursuit* when the mixture is high-dimensional (many sensors P or samples/pixels/voxels/ etc. T). In addition, depending on the level of noise $\mathbf{b}$ added to the mixture, the distribution of source coefficients in the selected dictionary, etc., the parameters for optimal algorithm performance (exponent τ, penalty factor λ, stop criterion for *matching pursuit*, etc.) can vary. How easy they are to adjust, via a development data set, may then be taken into consideration.

Choosing the separation method. Separation methods do not all have the same robustness in the case of an inaccurately estimated mixing matrix. A comparison of several methods [67] based on time-frequency dictionaries and stereophonic audio mixtures ($P = 2$) showed that, while less effective than when P active sources are assumed (section 10.6.4) and the mixing matrix $\mathbf{A}$ is perfectly known, binary masking separation assuming a single active source (section 10.6.2) is more robust, and thus more effective, than when only an imperfect estimation of $\mathbf{A}$ is available.

Choosing the algorithm for estimating A. This is undoubtedly the most difficult choice; while any "reasonable" choice for the dictionary, for the joint sparse mixture representation algorithm or for the separation method makes it possible to separate the sources *grosso modo* when the mixing matrix, even if under-determined, is known, an excessive error in estimating the mixing matrix has a catastrophic effect on the results, which become highly erratic. In particular, if the number of sources is not known in advance, an algorithm that provides a very robust estimation is an absolute necessity. This tends to eliminate approaches based on detecting peaks in a histogram in favor of those, like the algorithm DEMIX [8], that exploit redundancies (e.g. temporal and frequential persistence in an STFT) in the mixture representation to make estimating the column directions of $\mathbf{A}$ more robust (with measurement of reliability). If the source supports are not disjoint enough, it becomes necessary to use the approaches in section 10.6.4, which are not very robust if $\mathbf{A}$ is not accurately estimated, making a robust estimation of the mixing matrix all the more critical.

10.8 OUTLOOK

To date, most blind source separation techniques based on sparsity have relied on a sparse model of all the sources in a common dictionary, making it possible to exploit their *spatial diversity* in order to separate them. The principle of these techniques consists in decomposing the mixture into the basic components, grouping these components by the similarity of their spatial characteristics, and finally recombining them to reconstruct the sources. This principle may be applied to more general decompositions [97], provided that an appropriate similarity criterion enables regrouping components from the same source. Below we discuss some of the possibilities that are starting to be explored.

Spatial diversity and morphological diversity: looking for diversity in sources. While sources have sparse decompositions in *different* dictionaries, sparsity can also be exploited to separate them. This is a valid approach, including for the separation of sources from a single channel, when no spatial diversity is available. This type of approach, called Morphological Component Analysis (MCA) [97] because it uses differences in the *waveforms* of the sources to separate them, has been successfully applied to separating tonal and transient "layers" in musical sounds [29] for compression, and for the separation of texture from smoother content in images [96]. It is also possible to jointly exploit the spatial diversity and the *morphological diversity* of sources [13], which amounts to looking for a mixture decomposition with the form:

$$\mathbf{x} = \sum_{n=1}^{N} \mathbf{A}_n \cdot c_{s_n} \cdot \mathbf{\Phi}^{(n)} + \mathbf{b} = \sum_{n=1}^{N} \sum_{k=1}^{K} c_{s_n}(k) \mathbf{A}_n \varphi_k^{(n)} + \mathbf{b} \tag{10.75}$$

where $c_{s_n} \cdot \mathbf{\Phi}^{(n)}$ is a sparse approximation of s_n in a specific dictionary $\mathbf{\Phi}^{(n)}$. The estimated sources are then reconstructed as $\widehat{s_n} := \sum_k c_{s_n}(k) \varphi_k^{(n)}$. The validity of such an approach outside a purely academic context is based on the existence of morphological diversity between sources. Beyond its implementation, the real challenge of MCA lies in determining source-specific dictionaries (by expert knowledge or learning).

Separation of under-determined convolutive mixtures. So far, we have mainly discussed the separation of instantaneous mixtures, but sparsity also makes it possible to process simple forms of convolutive mixtures as well as anechoic mixtures (i.e. with a delay) [111,88]. The analysis of real mixtures raises problems of convolutive separation and possibly under-determination, $\mathbf{x} = \mathbf{A} \star \mathbf{s} + \mathbf{b}$. To adapt and extend sparsity-based techniques to this context, the conditions of mixture $\mathbf{A}$ must be identified, and separation must be performed when $\mathbf{A}$ is known or estimated. The first task remains challenging, despite some interesting approaches [77], but the second is easier and should be addressed by an approach similar to the deconvolution methods based on ℓ^1 norm minimization [94,34]. Formally, once $\mathbf{A}$ is estimated, a decomposition of the mixture is sought with the form:

$$\mathbf{x} = \sum_{n=1}^{N} \mathbf{A}_n \star (c_n \cdot \mathbf{\Phi}^{(n)}) + \mathbf{b} = \sum_{n=1}^{N} \sum_{k=1}^{K} c_n(k) \cdot \mathbf{A}_n \star \varphi_k^{(n)} + \mathbf{b} \tag{10.76}$$

and the sources $\widehat{s_n} := \sum_k c_{s_n}(k)\varphi_k^{(n)}$ are reconstructed. The success of such an approach, still to be demonstrated in practice, depends on the global diversity (combined morphological and spatial diversity) of the multi-channel waveforms $\mathbf{A}_n \star \varphi_k^{(n)}$.

Exploring the broad application potential of sparse source separation methods remains substantially limited by the algorithmic complexity of the optimizations involved. It is thus critical to develop and implement new algorithms which ideally combine three crucial properties:

- propensity to guarantee or ensure high probability of good sparse approximations;
- high numerical efficiency, i.e. economic use of computing resources (memory, arithmetic operations, etc.);
- robustness in the case of approximate estimation of the mixing matrix.

Algorithms with "certifiable" performance. A number of sparse decomposition algorithms have been proposed, and we have described the main examples in this chapter. They include the iterative algorithms of the *matching pursuit* category and the techniques based on minimizing ℓ^1 criteria, studied extensively at the initiative of Donoho and Huo [33] (see for example [103,56,32,52] and the references cited therein). These algorithms perform well – in a precise mathematical sense that makes it possible to "certify" their performance – provided the sources are close enough to a sparse model. Non-convex techniques for ℓ^τ optimization, $0 < \tau < 1$, are currently being analyzed [52], but somewhat surprisingly, there is less interest in convex ℓ^τ optimization techniques for $1 < \tau \leq 2$. "Certification" using theorems on the quality of results from joint approximation algorithms (see section 10.3) or separation algorithms (see section 10.6) is thus an area in which numerous questions are still open.

In the single-channel case, Gribonval and Nielsen [54,55] proved that while a highly sparse representation exists (implying a sufficiently small number of atoms), it is the only solution common to all the optimization problems (10.16) for $0 \leq \tau \leq 1$. Similar results [52] lead to the conclusion that all the single-channel sparse approximation problems (10.17) have very similar solutions for $0 \leq \tau \leq 1$, provided that the analyzed signal can be approximated effectively (in terms of mean square error) by combining a sufficiently small number of atoms from the dictionary. Tropp *et al*. [104,102] have shown that for $\tau = 1$, these results extend to the noisy multi-channel case.

Numerically efficient algorithms. From a numerical point of view, we have discussed the difficulties of solving the optimization problems (10.17) or (10.67) for $0 \leq \tau \leq 1$, which can involve inverting large-size matrices. Other approaches, such as the *matching pursuit* family, offer an interesting alternative in terms of algorithmic complexity, both for joint approximation and separation with a known mixing matrix. A fast implementation of complexity $\mathcal{O}(T \log T)$ called MPTK (*the matching pursuit toolkit*) was proposed and is freely available [53,65]. With redundant dictionaries such as multi-scale Gabor dictionaries, it is then possible to calculate good sparse approximations in a timeframe close to the signal's duration, even when very long. Other proposals such as iterative thresholding algorithms [28,42,36,44] are also extremely promising in terms of numerical efficiency.

Estimating the mixing matrix... and the dictionary! When sparsity is assumed, the problem of identifiability (and identification) of the mixing matrix is not a trivial one, given its direct link to the robustness of separation algorithms as a function of the accuracy of mixing matrix estimation. While we are starting to understand how to analyze and "certify" the performance of sparse approximation algorithms, initial results from similar assessments of mixing matrix identification algorithms [5], although encouraging, are only valid under restrictive sparse models that cannot tolerate noise. The choice of dictionary(/ies) $\mathbf{\Phi}/\mathbf{\Phi}^{(n)}$ is to some extent a dual problem, whose impact on separation performance is yet to be understood. While current practice involves choosing the analysis dictionary(/ies) from a library of traditional dictionaries, based on prior available knowledge of the sources, *learning* the dictionary based on the data to separate, or *sparse coding* is formally equivalent to the problem of estimating $\mathbf{A}$ (aside from the transposition given that $\mathbf{x}^{\mathrm{H}} = \mathbf{\Phi}^{\mathrm{H}} C_{\mathrm{s}}^{\mathrm{H}} \mathbf{A}^{\mathrm{H}} + \mathbf{b}^{\mathrm{H}}$). The approaches may differ considerably in practice because of the substantial difference in dimension of the matrices to estimate ($\mathbf{A}$ is $P \times N$, $\mathbf{\Phi}$ is $K \times T$, and generally $K \geq T \gg N \approx P$), which fundamentally changes the problem's geometry and one's intuitive grasp of it.

The idea of representing a mixture in a domain where it is sparse in order to facilitate separation has made it possible to successfully tackle the problem of separating under-determined instantaneous linear mixtures. The methods most frequently used today [91] rely on traditional dictionaries (Gabor, wavelets) identical for all sources, and representations based on linear transforms (STFT, etc.). To go further and determine whether sparsity – promising *a priori* for handling more complex and more realistic source separation problems – can move beyond this stage in practice, it is clear that certain technical obstacles identified above must be overcome. But above all, a serious effort to comparatively evaluate and analyze sparsity-based approaches is now more critical than ever.

Acknowledgements

R. Gribonval would like to warmly thank Simon Arberet for his precious help in preparing many of the figures in this chapter, as well as Gilles Gonon, Sacha Krsulović, Sylvain Lesage and Prasad Sudhakar for carefully reading an initial draft and offering many useful suggestions, comments and criticisms, which enabled substantial improvement of the organization of this chapter.

References

[1] S. Abdallah, M. Plumbley, If edges are the independent components of natural images, what are the independent components of natural sounds? in: Proc. Int. Conf. Indep. Component Anal. and Blind Signal Separation, ICA2001, San Diego, California, December 2001, pp. 534–539.

[2] F. Abrard, Y. Deville, A time-frequency blind signal separation method applicable to underdetermined mixtures of dependent sources, Signal Processing 85 (2005) 1389–1403.

[3] F. Abrard, Y. Deville, P. White, From blind source separation to blind source cancellation in the underdetermined case: a new approach based on time-frequency analysis, in: Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation, ICA 2001, San Diego, California, December 2001.
[4] M. Aharon, M. Elad, A. Bruckstein, The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representation, IEEE Transactions on Signal Processing 54 (2006) 4311–4322.
[5] M. Aharon, M. Elad, A. Bruckstein, On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them, Journal of Linear Algebra and Applications 416 (2006) 48–67.
[6] A. Aïssa-El-Bey, K. Abed-Meraim, Y. Grenier, Underdetermined blind source separation of audio sources in time-frequency domain, in: Proc. First Workshop on Signal Processing with Sparse/Structured Representations, SPARS'05, Rennes, France, November 2005.
[7] S. Amari, A. Cichocki, H.H. Yang, A new learning algorithm for blind signal separation, in: Advances in Neural Information Processing Systems, vol. 8, MIT Press, 1996.
[8] S. Arberet, R. Gribonval, F. Bimbot, A robust method to count and locate audio sources in a stereophonic linear instantaneous mixture, in: J.P. Rosca, D. Erdogmus, S. Haykin (Eds.), Proc. of the Int'l. Workshop on Independent Component Analysis and Blind Signal Separation, ICA 2006, Charleston, South Carolina, USA, in: LNCS Series, vol. 3889, Springer, March 2006, pp. 536–543.
[9] R. Balan, J. Rosca, Statistical properties of STFT ratios for two channel systems and applications to blind source separation, in: Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation, ICA 2000, Helsinki, Finland, June 2000, pp. 429–434.
[10] A. Ben-Tal, M. Zibulevsky, Penalty/barrier multiplier methods for convex programming problems, SIAM Journal on Optimization 7 (1997) 347–366.
[11] D. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press, New York, 1982.
[12] D. Bertsekas, Non-Linear Programming, 2nd ed., Athena Scientific, Belmont, MA, 1995.
[13] J. Bobin, Y. Moudden, J.-L. Starck, M. Elad, Multichannel morphological component analysis, in: Proc. First Workshop on Signal Processing with Sparse/Structured Representations, SPARS'05, Rennes, France, November 2005.
[14] J. Bobin, J.-L. Starc, J. Fadili, Y. Moudden, D.L. Donoho, Morphological component analysis: An adaptative thresholding strategy, IEEE Transactions on Image Processing 16 (2007) 2675–2681.
[15] P. Bofill, M. Zibulevsky, Underdetermined blind source separation using sparse representations, Signal Processing 81 (2001) 2353–2362.
[16] A.M. Bronstein, M.M. Bronstein, M. Zibulevsky, Blind source separation using block-coordinate relative Newton method, Signal Processing 84 (2004) 1447–1459.
[17] A.M. Bronstein, M.M. Bronstein, M. Zibulevsky, Relative optimization for blind deconvolution, IEEE Transactions on Signal Processing 53 (2005) 2018–2026.
[18] A.M. Bronstein, M.M. Bronstein, M. Zibulevsky, Y.Y. Zeevi, Blind deconvolution of images using optimal sparse representations, IEEE Transactions on Image Processing 14 (2005) 726–736.
[19] J.-F. Cardoso, Blind signal separation: Statistical principles, Proceedings of the IEEE 9 (1998) 2009–2025. Special Issue on Blind Identification and Estimation.
[20] J.-F. Cardoso, High-order contrasts for independent component analysis, Neural Computation 11 (1999) 157–192.
[21] J.-F. Cardoso, JADE for real-valued data, tech. report, ENST, 1999. http://sig.enst.fr:80/~cardoso/guidesepsou.html.
[22] J.-F. Cardoso, B. Laheld, Equivariant adaptive source separation, IEEE Transactions on Signal Processing 44 (1996) 3017–3030.
[23] S.S. Chen, D.L. Donoho, M.A. Saunders, Atomic decomposition by basis pursuit, SIAM Journal on Scientific Computing 20 (1998) 33–61.
[24] A. Cichocki, S. Amari, K. Siwek, ICALAB toolbox for image processing – benchmarks, tech. report, The Laboratory for Advanced Brain Signal Processing, RIKEN Brain Science Institute, 2002. http://www.bsp.brain.riken.go.jp/ICALAB/ICALABImageProc/benchmarks/.

[25] A. Cichocki, R. Unbehauen, E. Rummert, Robust learning algorithm for blind separation of signals, Electronics Letters 30 (1994) 1386–1387.
[26] R.R. Coifman, Y. Meyer, M.V. Wickerhauser, Wavelet analysis and signal processing, in: B. Ruskai, et al. (Eds.), Wavelets and their Applications, Jones and Barlett, Boston, 1992.
[27] S. Cotter, B. Rao, K. Engan, K. Kreutz-Delgado, Sparse solutions to linear inverse problems with multiple measurement vectors, IEEE Transactions on Signal Processing 53 (2005) 2477–2488.
[28] I. Daubechies, M. Defrise, C. De Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, Communications on Pure and Applied Mathematics 57 (2004) 1413–1457.
[29] L. Daudet, B. Torrésani, Hybrid representations for audiophonic signal encoding, Signal Processing 82 (2002) 1595–1617. Special Issue on Coding Beyond Standards.
[30] M. Davies, N. Mitianoudis, A sparse mixture model for overcomplete ICA, IEE Proceedings – Vision Image and Signal Processing 151 (2004) 35–43. Special issue on Nonlinear and Non-Gaussian Signal Processing.
[31] G. Davis, S. Mallat, M. Avellaneda, Adaptive greedy approximations, Construction Approximation 13 (1997) 57–98.
[32] D. Donoho, M. Elad, V. Temlyakov, Stable recovery of sparse overcomplete representations in the presence of noise, IEEE Transactions on Information Theory 52 (2006) 6–18.
[33] D. Donoho, X. Huo, Uncertainty principles and ideal atomic decompositions, IEEE Transactions on Information Theory 47 (2001) 2845–2862.
[34] C. Dossal, Estimation de fonctions géométriques et déconvolution, PhD thesis, École Polytechnique, Palaiseau, France, 2005.
[35] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, Least angle regression, Annals of Statistics 32 (2004) 407–499.
[36] M. Elad, Why simple shrinkage is still relevant for redundant representations? IEEE Transactions on Information Theory 52 (2006) 5559–5569.
[37] M. Elad, B. Matalon, M. Zibulevsky, Coordinate and subspace optimization methods for linear least squares with non-quadratic regularization, Applied and Computational Harmonic Analysis 23 (2007) 346–367.
[38] Y.F. Abrard, Blind separation of dependent sources using the "time-frequency ratio of mixtures" approach, in: ISSPA, IEEE, Paris, France, 2003.
[39] C. Févotte, C. Doncarli, Two contributions to blind source separation using time-frequency distributions, IEEE Signal Processing Letters 11 (2004) 386–389.
[40] C. Févotte, S.J. Godsill, A Bayesian approach for blind separation of sparse sources, IEEE Transactions on Audio, Speech and Language Processing 14 (6) (2006) 2174–2188.
[41] D. Field, B. Olshausen, Emergence of simple-cell receptive field properties by learning a sparse code for natural images, Nature 381 (1996) 607–609.
[42] M. Figueiredo, R. Nowak, An EM algorithm for wavelet-based image restoration, IEEE Transactions on Image Processing 12 (2003) 906–916.
[43] R.M. Figueras i Ventura, P. Vandergheynst, P. Frossard, Low rate and flexible image coding with redundant representations, IEEE Transactions on Image Processing 15 (2006) 726–739.
[44] M. Fornasier, H. Rauhut, Recovery algorithms for vector valued data with joint sparsity constraints, Tech. Report 27, Johns Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, 2006.
[45] J.-J. Fuchs, Some further results on the recovery algorithms, in: Proc. First Workshop on Signal Processing with Sparse/Structured Representations, SPARS'05, Rennes, France, November 2005, pp. 67–70.
[46] P.E. Gill, W. Murray, M.H. Wright, Practical Optimization, Academic Press, New York, 1981.
[47] G.H. Golub, C. Van Loan, Matrix Computations, 2nd ed., The Johns Hopkins University Press, Baltimore and London, 1989.

[48] I.F. Gorodnitsky, B.D. Rao, Sparse signal reconstruction from limited data using focuss: A reweighted norm minimization algorithm, IEEE Transactions on Signal Processing 45 (1997) 600–616.
[49] R. Gribonval, Sparse decomposition of stereo signals with matching pursuit and application to blind separation of more than two sources from a stereo mixture, ICASSP'02, in: Proc. Int. Conf. Acoust. Speech Signal Process., vol. 3, IEEE, Orlando, Florida, 2002, pp. III/3057–III/3060.
[50] R. Gribonval, Piecewise linear source separation, in: M. Unser, A. Aldroubi, A. Laine, (Eds.), Proc. SPIE '03, in: Wavelets: Applications in Signal and Image Processing X, vol. 5207, San Diego, CA, August 2003, pp. 297–310.
[51] R. Gribonval, L. Benaroya, E. Vincent, C. Févotte, Proposals for performance measurement in source separation, in: Proc. 4th Int. Symp. on Independent Component Anal. and Blind Signal Separation, ICA2003, Nara, Japan, April 2003, pp. 763–768.
[52] R. Gribonval, R.M. Figueras i Ventura, P. Vandergheynst, A simple test to check the optimality of sparse signal approximations, EURASIP Signal Processing 86 (2006) 496–510. Special issue on Sparse Approximations in Signal and Image Processing.
[53] R. Gribonval, S. Krstulović, MPTK, The Matching Pursuit Toolkit, 2005.
[54] R. Gribonval, M. Nielsen, On the strong uniqueness of highly sparse expansions from redundant dictionaries, in: Proc. Int. Conf. Independent Component Analysis, ICA'04, in: LNCS, Springer-Verlag, Granada, Spain, 2004.
[55] R. Gribonval, M. Nielsen, Highly sparse representations from dictionaries are unique and independent of the sparseness measure, Appl. Comput. Harm. Anal. 22 (2007) 335–355.
[56] R. Gribonval, M. Nielsen, Beyond sparsity: Recovering structured representations by ℓ^1-minimization and greedy algorithms, Advances in Computational Mathematics 28 (2008) 23–41.
[57] R. Gribonval, H. Rauhut, K. Schnass, P. Vandergheynst, Atoms of all channels, unite! Average case analysis of multi-channel sparse recovery using greedy algorithms, Journal of Fourier Analysis and Applications 14 (2008) 655–687.
[58] R. Gribonval, P. Vandergheynst, On the exponential convergence of Matching Pursuits in quasi-incoherent dictionaries, IEEE Transactions on Information Theory 52 (2006) 255–261.
[59] A. Hyvärinen, The Fast-ICA MATLAB package, tech. report, HUT, 1998. http://www.cis.hut.fi/~aapo/.
[60] A. Hyvärinen, Fast and robust fixed-point algorithms for independent component analysis, IEEE Transactions on Neural Networks 10 (1999) 626–634.
[61] A. Jourjine, S. Rickard, O. Yilmaz, Blind separation of disjoint orthogonal signals: Demixing n sources from 2 mixtures, in: Proc. IEEE Conf. Acoustics Speech and Signal Proc., ICASSP'00, vol. 5, Istanbul, Turkey, June 2000, pp. 2985–2988.
[62] P. Kisilev, M. Zibulevsky, Y. Zeevi, A multiscale framework for blind separation of linearly mixed signals, The Journal of Machine Learning Research 4 (2003) 1339–1363.
[63] P. Kisilev, M. Zibulevsky, Y.Y. Zeevi, B.A. Pearlmutter, Multiresolution framework for sparse blind source separation, tech. report, Department of Electrical Engineering, Technion. Haifa, Israel, 2000. http://ie.technion.ac.il/~mcib/.
[64] K. Kreutz-Delgado, B. Rao, K. Engan, T.-W. Lee, T. Sejnowski, Convex/schur-convex (csc) log-priors and sparse coding, in: 6th Joint Symposium on Neural Computation, Institute for Neural Computation, 1999, pp. 65–71.
[65] S. Krstulovic, R. Gribonval, MPTK: Matching Pursuit made tractable, in: Proc. Int. Conf. Acoust. Speech Signal Process., ICASSP'06, vol. 3, Toulouse, France, May 2006, pp. III–496 – III–499.
[66] T.-W. Lee, M.S. Lewicki, M. Girolami, T.J. Sejnowski, Blind source separation of more sources than mixtures using overcomplete representations, IEEE Signal Processing Letters 6 (1999) 87–90.
[67] S. Lesage, S. Krstulovic, R. Gribonval, Under-determined source separation: comparison of two approaches based on sparse decompositions, in: J.P. Rosca, D. Erdogmus, S. Haykin (Eds.), Proc. of the Int'l. Workshop on Independent Component Analysis and Blind Signal Separation, ICA 2006, Charleston, South Carolina, USA, in: LNCS Series, vol. 3889, Springer, 2006, pp. 633–640.

[68] D. Leviatan, V. Temlyakov, Simultaneous approximation by greedy algorithms, Tech. Report 0302, IMI, Dept. of Mathematics, University of South Carolina, Columbia, SC 29208, 2003.
[69] M. Lewicki, Efficient coding of natural sounds, Nature Neuroscience 5 (2002) 356–363.
[70] M. Lewicki, T. Sejnowski, Learning overcomplete representations, Neural Computation 12 (2000) 337–365.
[71] M.S. Lewicki, B.A. Olshausen, A probabilistic framework for the adaptation and comparison of image codes, Journal of the Optical Society of America 16 (1999) 1587–1601.
[72] J. Lin, D. Grier, J. Cowan, Faithful representation of separable distributions, Neural Computation 9 (1997) 1305–1320.
[73] S. Makeig, ICA toolbox for psychophysiological research, Computational Neurobiology Laboratory, the Salk Institute for Biological Studies, 1998. http://www.cnl.salk.edu/~ica.html.
[74] D. Malioutov, M. Cetin, A. Willsky, Homotopy continuation for sparse signal representation, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'05, vol. V, March 2005, pp. 733–736.
[75] S. Mallat, A Wavelet Tour of Signal Processing, Academic Press, San Diego, CA, 1998.
[76] S. Mallat, Z. Zhang, Matching pursuit with time-frequency dictionaries, IEEE Transactions on Signal Processing 41 (1993) 3397–3415.
[77] T. Melia, S. Rickard, Extending the DUET blind source separation technique, in: Proc. First Workshop on Signal Processing with Sparse/Structured Representations, SPARS'05, Rennes, France, November 2005, pp. 67–70.
[78] B. Natarajan, Sparse approximate solutions to linear systems, SIAM Journal on Computing 25 (1995) 227–234.
[79] D. Needell, J.A. Tropp, Cosamp: Iterative signal recovery from incomplete and inaccurate samples, tech. report, Caltech, 2008.
[80] B.A. Olshausen, D.J. Field, Sparse coding with an overcomplete basis set: A strategy employed by v1? Vision Research 37 (1997) 3311–3325.
[81] B.A. Pearlmutter, L.C. Parra, Maximum likelihood blind source separation: A context-sensitive generalization of ICA, in: Advances in Neural Information Processing Systems, vol. 9, MIT Press, 1997.
[82] E. Pearson, The Multiresolution Fourier Transform and its application to Polyphonic Audio Analysis, PhD thesis, University of Warwick, September 1991.
[83] L. Peotta, L. Granai, P. Vandergheynst, Image compression using an edge adapted redundant dictionary and wavelets, Signal Processing 86 (2006) 444–456.
[84] D. Pham, P. Garat, Blind separation of a mixture of independent sources through a quasi-maximum likelihood approach, IEEE Transactions on Signal Processing 45 (1997) 1712–1725.
[85] M. Plumbley, Geometry and homotopy for ℓ^1 sparse signal representations, in Proc. First Workshop on Signal Processing with Sparse/Structured Representations, SPARS'05, Rennes, France, November 2005, pp. 67–70.
[86] R. Polyak, Modified barrier functions: Theory and methods, Mathematical Programming 54 (1992) 177–222.
[87] J. Princen, A. Bradley, Analysis/synthesis filter bank design bases on time domain aliasing cancellation, IEEE Transactions on Acoustics, Speech and Signal Proc. ASSP-34 (1986) 1153–1161.
[88] M. Puigt, Y. Deville, Time-frequency ratio-based blind separation methods for attenuated and time-delayed sources, Mechanical Systems and Signal Processing 19 (2005) 1348–1379.
[89] L. Rabiner, R. Schafer, Digital Processing of Speech Signals, Prentice Hall, 1978.
[90] V.P. Rahmoune A, F. P, Flexible motion-adaptive video coding with redundant expansions, IEEE Transactions on Circuits and Systems for Video Technology 16 (2006) 178–190.
[91] S. Rickard, R. Balan, J. Rosca, Real-time time-frequency based blind source separation, in: 3rd International Conference on Independent Component Analysis and Blind Source Separation, ICA2001, San Diego, CA, December 2001.
[92] R. Rubinstein, M. Zibulevsky, M. Elad, Double sparsity: Learning sparse dictionaries for sparse signal approximation, IEEE Transactions on Signal Processing (2009) (in press).

[93] R. Saab, Ö Yilmaz, M. McKeown, R. Abugharbieh, Underdetermined sparse blind source separation with delays, in Proc. First Workshop on Signal Processing with Sparse/Structured Representations, SPARS'05, Rennes, France, November 2005, pp. 67–70.
[94] F. Santosa, W. Symes, Linear inversion of band-limited reflection sismograms, SIAM J. Sci. Statistic. Comput. 7 (1986) 1307–1330.
[95] J.L. Starck, E.J. Candès, D.L. Donoho, The curvelet transform for image denoising, IEEE Transactions on Image Processing 11 (2000) 670–684.
[96] J.-L. Starck, M. Elad, D. Donoho, Image decomposition : Separation of textures from piecewise smooth content, in: M. Unser, A. Aldroubi, A. Laine (Eds.), Wavelet: Applications in Signal and Image Processing X, in: Proc. SPIE '03, vol. 5207, SPIE (The International Society for Optical Engineering), San Diego, CA, August 2003, pp. 571–582.
[97] J.-L. Starck, Y. Moudden, J. Bobin, M. Elad, D. Donoho, Morphological component analysis, in: Proceedings of the SPIE conference wavelets, vol. 5914, July 2005.
[98] F. Theis, A. Jung, C. Puntonet, E. Lang, Linear geometric ICA: Fundamentals and algorithms, Neural Computation 15 (2003) 419–439.
[99] F.J. Theis, E.W. Lang, Formalization of the two-step approach to overcomplete BSS, in: Proc. SIP 2002, Kauai, Hawaii, USA, 2002, pp. 207–212.
[100] F.J. Theis, C. Puntonet, E.W. Lang, A histogram-based overcomplete ICA algorithm, in: Proc. 4th Int. Symp. on Independent Component Anal. and Blind Signal Separation, ICA2003, Nara, Japan, 2003, pp. 1071–1076.
[101] J. Tropp, Greed is good: Algorithmic results for sparse approximation, IEEE Transactions on Inform. Theory 50 (2004) 2231–2242.
[102] J. Tropp, Algorithms for simultaneous sparse approximation. Part II: Convex relaxation, Signal Processing 86 (2006) 589–602. Special issue on Sparse Approximations in Signal and Image Processing.
[103] J. Tropp, Just relax: Convex programming methods for identifying sparse signals in noise, IEEE Transactions on Information Theory 52 (2006) 1030–1051.
[104] J. Tropp, A. Gilbert, M. Strauss, Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit, Signal Processing 86 (2006) 572–588. Special issue on Sparse Approximations in Signal and Image Processing.
[105] B. Turlach, On algorithms for solving least squares problems under an ℓ^1 pernalty or an ℓ^1 constraint, in: 2004 Proc. of the American Statistical Association, vol. Statistical Computing Section [CDROM] Alexandria, VA, American Statistical Association, 2005, pp. 2572–2577.
[106] M. Van Hulle, Clustering approach to square and non-square blind source separation. in: IEEE Workshop on Neural Networks for Signal Processing, NNSP99, 1999, pp. 315–323.
[107] L. Vielva, D. Erdogmus, J. Principe, Underdetermined blind source separation using a probabilistic source sparsity model, in: Proc. Int. Conf. on ICA and BSS, ICA2001, San Diego, California, 2001, pp. 675–679.
[108] E. Vincent, Complex nonconvex l_p norm minimization for underdetermined source separation, in: Proc. Int. Conf. Indep. Component Anal. and Blind Signal Separation, ICA2001, Springer, 2007, pp. 430–437.
[109] D. Weitzer, D. Stanhill, Y.Y. Zeevi, Nonseparable two-dimensional multiwavelet transform for image coding and compression, Proc. SPIE 3309 (1997) 944–954.
[110] R. Xu, D. Wunsch II, Survey of clustering algorithms, IEEE Transactions on Neural Networks 16 (2005) 645–678.
[111] O. Yilmaz, S. Rickard, Blind separation of speech mixtures via time-frequency masking, IEEE Transactions on Signal Processing 52 (2004) 1830–1847.
[112] M. Zibulevsky, Blind source separation with Relative Newton method, Proceedings ICA-2003, (2003), pp. 897–902.
[113] M. Zibulevsky, Relative Newton and smoothing multiplier optimization methods for blind source separation, in: S. Makino, T. Lee, H. Sawada (Eds.), Blind Speech Separation, in: Springer Series: Signals and Communication Technology XV, Springer, 2007.

[114] M. Zibulevsky, P. Kisilev, Y.Y. Zeevi, B.A. Pearlmutter, Blind source separation via multinode sparse representation, in: Advances in Neural Information Processing Systems, vol. 12, MIT Press, 2002.
[115] M. Zibulevsky, B.A. Pearlmutter, Blind source separation by sparse decomposition in a signal dictionary, Neural Computation 13 (2001) 863–882.
[116] M. Zibulevsky, B.A. Pearlmutter, Blind source separation by sparse decomposition, Tech. Report CS99-1, Univ. of New Mexico, July 1999.
[117] M. Zibulevsky, B.A. Pearlmutter, P. Bofill, P. Kisilev, Blind source separation by sparse decomposition, in: S.J. Roberts, R.M. Everson (Eds.), Independent Components Analysis: Principles and Practice, Cambridge University Press, 2001.