Chapter 1

Blind Signal Processing Based on Data Geometric Properties

Konstantinos Diamantaras
Department of Informatics
TEI of Thessaloniki
Sindos, GR 57400, GREECE
kdiamant@it.teithe.gr

1.1 Introduction

Blind signal processing deals with the outputs of unknown systems excited by unknown inputs. At first sight the problem seems intractable, but a closer look reveals that certain signal properties allow us to extract the inputs or to identify the system up to some, usually not important, ambiguities. Linear systems are mathematically most tractable and, naturally, they have attracted most of the attention. Depending on the type of the linear system, blind problems arise in a wide variety of applications, for example, in digital communications (Godard, 1980; Treichler and Agee, 1983; Shalvi and Weinstein, 1990; Talwar et al., 1994; Tong et al., 1994; **?**; **?**; Yellin and Weinstein, 1996; Torlak and Xu, 1997; Paulraj and Papadias, 1997; Papadias and Paulraj, 1997; Tsatsanis and Giannakis, 1997; Diamantaras et al., 2000; Diamantaras and Papadimitriou, 2004a,b), in biomedical signal processing (Choi et al., 2000; Cichocki et al., 1999; Jung et al., 1998; **?**; Makeig et al., 1997; McKeown et al., 1998; Vigário et al., 2000), in acoustics and speech processing (Shamsunder and Giannakis, 1997; Parra and Spence, 2000; Parra and Alvino, 2002; Douglas and Sun, 2003), etc. Many recent books on the subject (Haykin, 2000a,b; Cichocki and Amari, 2002; Hyvärinen et al., 2001) provide extensive discussion on related problems and methods.

The most general finite, linear, time invariant (LTI) system is expressed by a multichannel convolution of length L, operating on a discrete vector signal $\mathbf{s}(k) = [s_1(k), \cdots, s_n(k)]^T$,

$$\mathbf{x}(k) = \sum_{i=0}^{L-1} \mathbf{H}_i \, \mathbf{s}(k-i). \tag{1.1}$$

The FIR *filter taps* $\mathbf{H}_i$ are complex matrices, in general, of size $m \times n$, $m \geq 1$. Thus the output is an m-dimensional complex vector $\mathbf{x}(k)$. For $n, m > 1$, Eq. (1.1) describes a linear, discrete, multi-input multi-output (MIMO) system.

Types of mixing systems. We shall study two special cases of system (1.1) sharing many similarities but also having some special characteristics as described below:.

1. *Instantaneous mixtures.*

In this case we have more than one sources and observations, i.e. $m, n > 1$, but there is no convolution involved, so $L = 1$. The output vector is produced by a linear, instantaneous transformation:

$$\mathbf{x}(k) = \mathbf{H}\mathbf{s}(k). \tag{1.2}$$

This type of system is also called *memoryless*.

2. *Single Input Single Output (SISO) convolution.*

In this case we have exactly one source and one observation, so $m, n = 1$, but the convolution is non-trivial, i.e. $L > 1$.

$$x(k) = \sum_{i=0}^{L-1} h_i \, s(k-i). \tag{1.3}$$

Eq. (1.3) describes a linear, SISO FIR filter.

Types of blind problems. Regardless of the specific system type, there are two kinds of blind problems which are of interest here, depending on whether we desire to extract the input signals or the system parameters.

1. *Blind source extraction.*

In this type of problem our goal is to recover the source(s) given the observation signal $\mathbf{x}(k)$ or $x(k)$. If there are more than one sources the problem is called *Blind Source Separation* (*BSS*). In the case of BSS the linear system may be either instantaneous or convolutive (general MIMO). In the case of *Blind Deconvolution* (*BD*) we want to invert a linear filter which, of course, operates on its input via the convolution operator, hence the name "de-convolution" attributed to this problem. The problem is very important, for example, in wireless communications, where n transmitted signals corrupted by intersymbol interference (ISI), multiuser interference (MUI), and noise are received at m antennas.

The source separation/extraction problem has an inherent ambiguity in the order and the scale of the sources: the original signals can not be retrieved in their original order or scale unless some further information is available. For example, if the source samples (symbols) are drawn from a known finite alphabet then there is no ambiguity in the scale. If however, the alphabet is symmetric with respect to 0, then there exists a sign ambiguity since both signals $s(k)$ and $-s(k)$ are plausible. Furthermore, the ordering ambiguity is always present if the problem involves more than one sources.

2. *Blind system identification.*

In this type of problem our goal is to obtain the system parameters rather than recovering the source signals. If the system is memoryless then our goal is to recover the mixing matrix $\mathbf{H}$. If the system involves non-trivial convolution then the goal is to extract the filter taps h_0, ..., h_{L-1}, or $\mathbf{H}_0$, ..., $\mathbf{H}_{L-1}$.

Approaches to Blind Signal Processing. Typically, blind problems are approached either using statistical properties of the signals involved, or exploiting the geometric structure of the data constellation, as described next.

1. *Higher order methods.*

According to the central limit theorem, the system output –which is the sum of many input samples– will approach the Gaussian distribution, irrespective of the input distribution. A characteristic property of the Gaussian distribution is that all higher-order cumulants (for instance, the kurtosis) are zero. If the inputs are not normally distributed, their higher order cumulants will be non-zero, for example positive, and so (1.1) will work as a "cumulant reducer". Clearly, the blind system inversion –the linear transform that will recover the sources from the output– should function as a "cumulant increaser", i.e. it should maximize the absolute cumulant value for a given signal power. In fact, this is the basic idea behind all higher-order methods.

2. *Second order methods.*

   Alternatively, second order methods can be applied when the sources have colored spectra, regardless of their distribution. If the source colors are not identical then the time-delayed covariance matrices have a certain eigenvalue structure which reveals the mixing operator, in the memoryless case. This information can be used for recovering the sources as well. In the dynamic case, things are more complicated although, again, second order methods have been proposed based on the statistics of either the frequency or the time domain.

3. *A third approach: exploiting the signal geometry.*

   Neither higher-order nor second-order methods exploit the cluster structure or shape of the input data when such a structure or shape exists. Consider for example a source signal $s(k)$ whose samples are drawn from a finite alphabet $\mathcal{A}_M = \{\pm 1, \cdots \pm (M/2)\}$ (M= even). Let the SISO FIR filter described in Eq. (1.3) be excited by $s(k)$. Writing N equations ($N \geq L$) of the form (1.3) for N consecutive values of k, we obtain the following matrix equation:

$$\begin{bmatrix} x(k) \\ x(k+1) \\ \vdots \\ x(k+N-1) \end{bmatrix} = \begin{bmatrix} s(k) & s(k-1) & \cdots & s(k+1-L) \\ s(k+1) & s(k) & & s(k+2-L) \\ \vdots & \vdots & & \vdots \\ s(k+N-1) & s(k+N-2) & \cdots & s(k+N-L) \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{L-1} \end{bmatrix} \tag{1.4}$$

$$\mathbf{x} = \mathbf{S}\mathbf{h} \tag{1.5}$$

   where the $N \times L$ Toeplitz matrix $\mathbf{S}$ involves $(N + L - 1)$ unknown input symbols. It is possible, in principle, to identify $\mathbf{h}$ in a deterministic way by an exhaustive search over all M^{N+L-1} possible $\mathbf{S}$'s such that $\min_{\mathbf{h}} \|\mathbf{x} - \mathbf{S}\mathbf{h}\|^2 = 0$. Although it is highly impractical, this observation tells us that there is more to blind signal processing than statistical processing. If the sources for example, have a certain structure which produces clusters in the data cloud, or the input distribution is bounded (e.g. uniform) then one can exploit the geometric properties of the output constellation and derive fast and efficient deterministic algorithms for blind signal processing. These methods are treated in this chapter. In particular, Section 1.2 discusses blind methods for systems with finite alphabet sources. The discussion covers both instantaneous and the convolutive mixtures and it is based on the geometric properties of the data cloud. Section 1.3 discusses the case of continuous valued sources which are either sparse or have a specific input distribution, for example uniform. Our discussion on continuous sources covers only the case of instantaneous systems. Certainly there is a lot of room for innovation along this line of research since many issues, today, remain open.

1.2 Finite Alphabet sources

Blind problems involving sources with Finite Alphabets (FA) have drawn a lot of attention, because such types of signals are common in digital communications. Popular modulation schemes, for instance, *Quadrature Amplitude Modulation* (*QAM*), *Pulse Amplitude Modulation* (*PAM*) and *Binary Phase Shift Keying* (*BPSK*), produce signals with limited numbers of symbols. A large body of literature exists on the instantaneous mixture problem, not only because it is the simplest one but also because most methods dealing with the more realistic convolutive mixture problem lead to the solution of an instantaneous problem. In (Anand et al., 1995) the blind separation of binary sources from instantaneous mixtures is approached using separate clustering and bit-assignment algorithms. An extension of this method is presented in (Kannan and Reddy, 1997) where a Maximum Likelihood (ML) estimate of the cluster centers is provided. Talwar et al. (1996) presented two iterative least squares methods: ILSP (Iterative Least Squares with Projection) and ILSE (Iterative Least Squares with Enumeration) for the BSS of binary sources. The same problem is treated in (van der Veen, 1997) where the Real Analytical Constant Modulus Algorithm (RACMA) is introduced based on the SVD of the observation matrix. In (Pajunen, 1997) an iterative algorithm is proposed for the blind separation of more binary sources than sensors. Finite alphabet sources and instantaneous mixtures are discussed in (Belouchrani and Cardoso, 1994) where a Maximum Likelihood approach is proposed using the EM algorithm. (Grellier and Comon,

1998) introduce a polynomial criterion and a related minimization algorithm to separate FA sources. In all the above methods the geometric properties of the data cloud are not explicitly used. Geometrical concepts, such as the relative distances between the cluster centers, were introduced in (Diamantaras, 2000; Diamantaras and Chassioti, 2000). It turns out that just one observation signal is sufficient for blindly separating n binary sources, in the noise free case, under mild assumptions. A similar algorithm based on geometric concepts was later proposed in (Li et al., 2003).

In this section we shall study the geometric structure of data constellations generated from linear systems operating on signals with finite alphabets. We'll find that the geometry of the obtained data cloud contains information pertaining to the generating linear operator. This information can be exploited either for the blind extraction of the system parameters or for the blind retrieval of the original sources.

1.2.1 Instantaneous mixtures of binary sources

The simplest alphabet is the two element set, or *binary alphabet* $\mathcal{A}_a = \{-1, 1\}$. We shall assume that the samples of some source signals are drawn from $\mathcal{A}_a$, and the signals will be called *binary antipodal* or, simply, *binary*. In digital communications the carrier modulation scheme using symbols from $\mathcal{A}_a$ is called Binary Phase Shift Keying (BPSK). The reader is encouraged to verify that our results can be easily generalized to any type of binary alphabet, for example, the nonsymmetric set $\mathcal{A}_b = \{0, 1\}$.

In this subsection we shall concentrate on problem type 1, i.e. on linear memoryless mixtures of many sources, $n > 1$. Depending on the number of output signals (observations) m, we treat three distinct cases (a) $m = 1$, (b) $m = 2$, and (c) $m > 2$.

A single mixture

The instantaneous mixture of n sources linearly combined into a single observation is described by the following equation

$$x(k) = \sum_{i=1}^{n} h_i\, s_i(k) = \mathbf{h}^T \mathbf{s}(k), \tag{1.6}$$

$$\mathbf{h} = [h_1 \cdots h_n]^T, \qquad \mathbf{s}(k) = [s_1(k) \cdots s_n(k)]^T.$$

We assume that the mixing coefficients h_i are real and that $s_i(k) \in \mathcal{A}_a$. If the coefficients are complex then the problem corresponds to the case $m = 2$ which is treated in Section 1.2.1. We start by studying the noise-free system since our primary interest is to investigate the structural properties of the signals and not to develop methods to combat the noise. Of course, eventually, the development of a viable algorithm will have to deal with the noise issue.

Eq. (1.6) can be seen as the projection $\tilde{x}(k)$ of $\mathbf{s}(k)$ along the direction of the normal vector $\tilde{\mathbf{h}}$, scaled by $\|\mathbf{h}\|$:

$$x(k) = \|\mathbf{h}\|\, \tilde{x}(k) \tag{1.7}$$
$$\tilde{x}(k) = \tilde{\mathbf{h}}^T \mathbf{s}(k) \tag{1.8}$$
$$\tilde{\mathbf{h}} = \mathbf{h}/\|\mathbf{h}\| \tag{1.9}$$

The set of values of $x(k)$ will be called the *constellation* of $x(k)$ and it will be denoted by $\mathcal{X}$. It is a set of (at most) 2^n points in 1-D space, $\mathbb{R}$.

Two sources. In order to facilitate our understanding of the geometric structure of $\mathcal{X}$ let us start by assuming that there are only $n = 2$ sources. Thus, there exist four possible realizations of the vector $\mathbf{s}(k)$, which form the source constellation $\mathcal{S} = \{\mathbf{s}^{--}, \mathbf{s}^{-+}, \mathbf{s}^{+-}, \mathbf{s}^{++}\}$, where

$$\mathbf{s}^{--} = [-1, -1]^T,\ \mathbf{s}^{-+} = [-1, 1]^T,\ \mathbf{s}^{+-} = [1, -1]^T,\ \text{and } \mathbf{s}^{++} = [1, 1]^T.$$

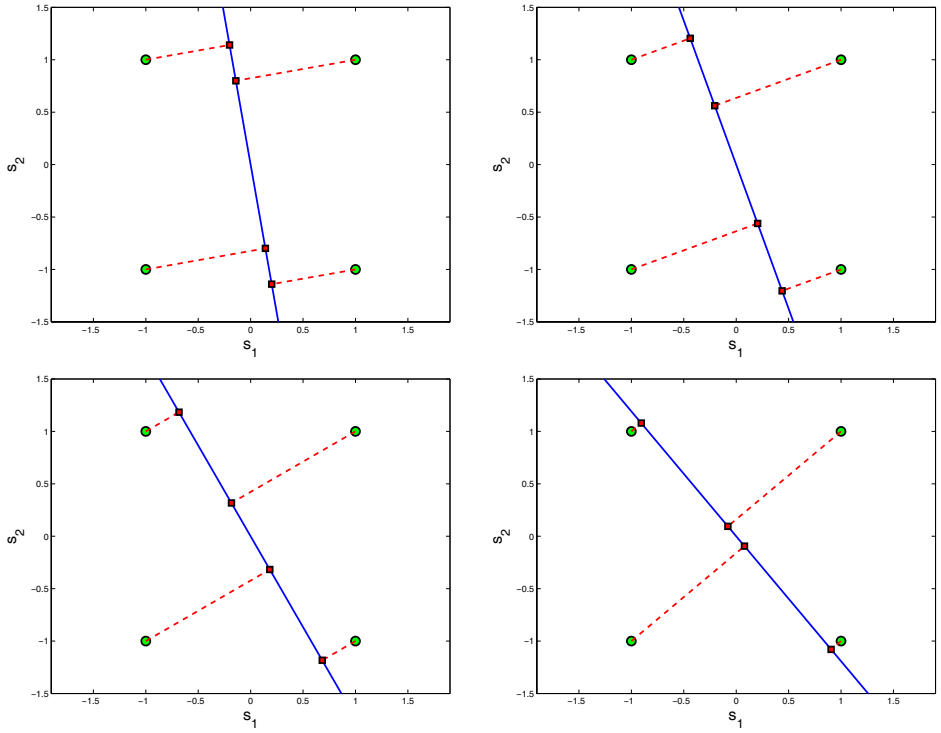


Figure 1.1: The source constellation (circles) of two independent binary sources is projected on four different directions. The relative distances of the projection points (marked by squares) is clearly a function of the slope of the projection line.

Consequently, the output constellation $\mathcal{X}$ also consists of four distinct values:

$$\begin{aligned} x^{--} &= \|\mathbf{h}\|\,\tilde{x}^{--} = \mathbf{h}^T\mathbf{s}^{--}, \\ x^{-+} &= \|\mathbf{h}\|\,\tilde{x}^{-+} = \mathbf{h}^T\mathbf{s}^{-+}, \\ x^{+-} &= \|\mathbf{h}\|\,\tilde{x}^{+-} = \mathbf{h}^T\mathbf{s}^{+-}, \\ x^{++} &= \|\mathbf{h}\|\,\tilde{x}^{++} = \mathbf{h}^T\mathbf{s}^{++}. \end{aligned}$$

Figure 1.1 shows the projections $\tilde{x}^{--}$, $\tilde{x}^{-+}$, $\tilde{x}^{+-}$, $\tilde{x}^{++}$, of the source constellation $\mathcal{S}$ for four different normal mixing vectors $\tilde{\mathbf{h}}$. It is obvious that the relative distance between the points on the projection line is a function of the angle θ between the projection line and the horizontal axis. Apparently, the problem involves a lot of symmetry. In particular, it is straightforward to verify that we obtain the same output constellation $\mathcal{X}$ for the angles $\pm\theta$, $\pm(\pi/2-\theta)$, $\pm(\pi-\theta)$, and $\pm(3\pi/2-\theta)$, (any θ). This multiple symmetry is the result of the interchangeability of the two sources, s_1 and s_2, as well as the invariance of the source constellation to sign changes. These ambiguities are however acceptable, since it is not possible to recover the original source order or the original source signs. Both the source order and the sign are unobservable as it is eminent from the following relations:

$$\begin{aligned} x(k) &= [\pm h_1, \pm h_2]\,[\pm s_1(k), \pm s_2(k)]^T \\ &= [\pm h_2, \pm h_1]\,[\pm s_2(k), \pm s_1(k)]^T \end{aligned}$$

Therefore, let us assume, without loss of generality, that the mixing vector $\mathbf{h}$ satisfies the following constraint

$$h_1 > h_2 > 0. \tag{1.10}$$

Under this assumption, the elements of $\mathcal{X}$ are ordered:

$$x^{--} = -h_1 - h_2 < x^{-+} = -h_1 + h_2 < x^{+-} = +h_1 - h_2 < x^{++} = +h_1 + h_2 \tag{1.11}$$

Indeed, the first and third inequalities in (1.11) are obvious since the mixing coefficients are positive. The second inequality is also true since $x^{+-} - x^{-+} = 2(h_1 - h_2) > 0$.

Thus, by clustering the (observable) output sequence $\{x(1), x(2), x(3), \cdots\}$ we obtain four cluster points c_1, c_2, c_3, c_4, which can be arranged in increasing order and set into one-to-one correspondence with the elements of $\mathcal{X}$.

$$c_1 = x^{--} < c_2 = x^{-+} < c_3 = x^{+-} < c_4 = x^{++} \tag{1.12}$$

$$c_1 = -c_4; \qquad c_2 = -c_3 \ .$$

Then using (1.11) we can recover the mixing parameters:

$$h_1 = (c_3 - c_1)/2, \tag{1.13}$$

$$h_2 = (c_2 - c_1)/2. \tag{1.14}$$

Example 1 *Figure 1.2 shows the position of the cluster points c_1, ..., c_4, for the random mixing vector $\mathbf{h} = [0.9659,\ 0.2588]^T$. According to (1.11), (1.12), these cluster points are:*

$$\begin{array}{rcrcr} c_1 & = & x^{--} & = & -1.2247 \\ c_2 & = & x^{-+} & = & -0.7071 \\ c_3 & = & x^{+-} & = & 0.7071 \\ c_4 & = & x^{++} & = & 1.2247 \end{array}$$

By computing the distances between the pairs (c_3, c_1) and (c_2, c_1), we obtain directly the unknown mixing parameters:

$$\begin{array}{rcl} (c_3 - c_1)/2 & = & 0.9659 = h_1 \\ (c_2 - c_1)/2 & = & 0.2588 = h_2 \end{array}$$

■

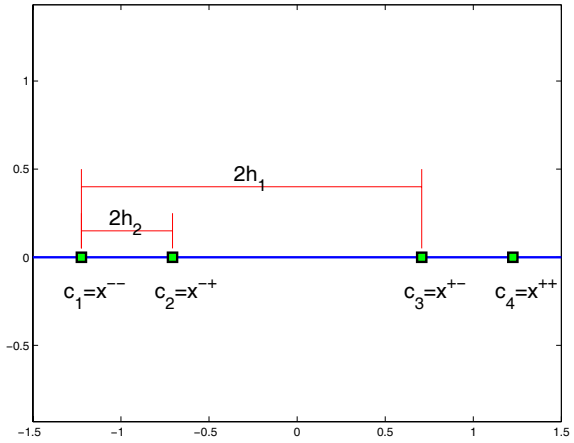


Figure 1.2: The distances c_3-c_1 and c_2-c_1 between the cluster points are equal to twice the size of the unknown mixing parameters.

If our aim is to identify the mixing parameters h_1, h_2, then (1.13) and (1.14) have achieved our goal. If, in addition, we want to extract the hidden sources then we may estimate each input sample $\mathbf{s}(k)$, separately, by finding the binary vector $\mathbf{b} = [b_1, b_2]^T \in \mathcal{A}_a^2$, so that $\mathbf{h}^T\mathbf{b}$ best approximates $x(k)$. This corresponds to the following binary optimization problem,

$$\hat{\mathbf{s}}(k) = \arg \min_{\mathbf{b} \in \mathcal{A}_a^2} |x(k) - \mathbf{h}^T\mathbf{b}|, \qquad \text{for all } k. \tag{1.15}$$

Luckily the above optimization problems are decoupled, for different k, and so the solution is trivial.

More than 2 sources. The whole idea can be extended to more than two sources using recursive "system deflation". This process iteratively identifies and removes the two smallest mixing parameters, thus eventually reducing the problem to either: (a) the 2-input case which is solved as above, or (b) the single-input case which is trivial. Our linear mixture model is again the one described in Eq. (1.6) with $n > 2$ and some real mixing vector $\mathbf{h} = [h_1, \cdots, h_n]^T$.

As before, without loss of generality, we shall assume that the mixing parameters are positive and arranged in decreasing order

$$h_1 > h_2 > \cdots > h_n > 0. \tag{1.16}$$

We have already shown that for $n = 2$ the centers c_i are arranged in increasing order. For $n > 2$ things are a bit more complicated. Let us define $\mathbf{B}^{(n)}$ to be the $2^n \times n$ matrix whose i-th row $\mathbf{b}_i^{(n)^T}$, is the binary representation of the number $(i-1) \in \{0, \cdots, 2^n - 1\}$:

$$\mathbf{B}^{(n)} \triangleq \begin{bmatrix} -1 & -1 & \cdots & -1 & -1 \\ -1 & -1 & \cdots & -1 & 1 \\ -1 & -1 & \cdots & 1 & -1 \\ \vdots & \vdots & & \vdots & \vdots \\ 1 & 1 & \cdots & -1 & 1 \\ 1 & 1 & \cdots & 1 & -1 \\ 1 & 1 & \cdots & 1 & 1 \end{bmatrix} \tag{1.17}$$

Although the sequence $\{c_1, \cdots, c_{2^n}\}$, of the centers

$$c_i = \mathbf{b}_i^{(n)^T} \mathbf{h} = \sum_{j=1}^{n} b_{ij}^{(n)} h_j, \quad i = 1, \cdots, 2^n \tag{1.18}$$

is not exactly arranged in increasing (or decreasing) order, there is a lot of structure in the sequence as summarized by the following facts (Diamantaras and Chassioti, 2000):

- the first three centers $c_1 < c_2 < c_3$ are the three smallest values in the sequence c_i. Similarly, the last three centers $c_{2^n-2} < c_{2^n-1} < c_{2^n}$ are the three largest values in the sequence $\{c_i\}$;
- the sequence c_1, ..., c_{2^n}, defined by Eq. (1.18) consists of consecutive quadruples, each arranged in increasing order:

  $$c_{4i+1} < c_{4i+2} < c_{4i+3} < c_{4i+4}, \quad i = 0, \cdots, 2^{n-2} - 1$$

  The smallest element of the i-th quadruple is

  $$c_{4i+1} = [\sum_{j=1}^{n-2} b_{4i+1,j}^{(n)} h_j] - h_{n-1} - h_n. \tag{1.19}$$

- The differences

  $$\delta_1 = c_{4i+2} - c_{4i+1} = 2h_n \tag{1.20}$$
  $$\delta_2 = c_{4i+3} - c_{4i+1} = 2h_{n-1} \tag{1.21}$$
  $$\delta_3 = c_{4i+4} - c_{4i+1} = 2(h_{n-1} + h_n) \tag{1.22}$$

  between the members of the i-th quadruple are independent of i.

- Since

  $$c_2 = c_1 + 2h_n$$
  $$c_3 = c_1 + 2h_{n-1}$$

  the two smallest mixing parameters h_{n-1}, h_n, can be retrieved using the values of the three smallest centers c_1, c_2, and c_3:

  $$h_n = (c_2 - c_1)/2 \tag{1.23}$$
  $$h_{n-1} = (c_3 - c_1)/2\ . \tag{1.24}$$

Once we have obtained h_{n-1} and h_n we can define a new sequence $\{c'_i\}$ by picking the first elements of each quadruple shifted by the sum $(h_{n-1}+h_n)$ thus obtaining:

$$c'_i = c_{4(i-1)+1} + h_{n-1} + h_n = \sum_{j=1}^{n-2} b^{(n)}_{4(i-1)+1,j}\, h_j, \quad i = 1, \cdots, 2^{n-2} \tag{1.25}$$

Notice however, that the first $n-2$ bits of the $[4(i-1)+1]$-th row of $\mathbf{B}^{(n)}$ are all the bits of the i-th row of $\mathbf{B}^{(n-2)}$. In other words,

$$b^{(n)}_{4(i-1)+1,j} = b^{(n-2)}_{ij}, \quad j = 1, \cdots, n-2$$

therefore,

$$c'_i = \mathbf{b}_i^{(n-2)^T}\mathbf{h} = \sum_{j=1}^{n-2} b^{(n-2)}_{ij} h_j, \quad i = 1, \cdots, 2^{n-2} \tag{1.26}$$

Using these facts the following recursive algorithm is constructed

Algorithm 1 (: n binary sources, 1 observation)
Step 1. *Compute the centers c_i and sort them in increasing order*
Step 2. *Compute h_n, h_{n-1}, from Eqs. (1.23), (1.24).*
Step 3. *Compute the differences δ_i, using Eqs. (1.20), (1.21), (1.22).*
Step 4. *Remove the set $\{c_1, c_2, c_3, c_1+\delta_3\}$ from the sequence $\{c_i\}$. Set $c'_1 = c_1 + h_n + h_{n-1}$ as the first element of a new sequence $\{c'_i\}$.*
Step 5. *Repeat until all elements have been removed:*

> *Find the smallest element c_j of the remaining sequence $\{c_i\}$;*
> *Remove the set $\{c_j, c_j+\delta_1, c_j+\delta_2, c_j+\delta_3\}$ from $\{c_i\}$;*
> *Keep $c_j + h_n + h_{n-1}$ as the next element of the sequence $\{c'_i\}$.*

At the end, the new sequence $\{c'_i\}$ will be 4 times shorter than the original $\{c_i\}$.
Step 6. *Recursively repeat the algorithm for the new sequence $\{c'_i\}$ and for a new $n' = n-2$ to obtain $h_{n'} = h_{n-2}$, $h_{n'-1} = h_{n-3}$. Eventually, $n' = 2$ or $n' = 1$.* ■

Steps 4 and 5 are the basic recursion which reduces the problem size from n to $n-2$ by replacing the sequence c_i by c'_i. At step 6, we will iteratively obtain the pairs (h_n, h_{n-1}), (h_{n-2}, h_{n-3}), ..., until we reach the case where $n' = 2$ or $n' = 1$. The case for $n' = 2$ sources was treated in the previous subsection. The case for $n' = 1$ is trivial since it involves only one source. In this case, the observation is simply a scaled version of the input, $x(k) = h_1 s_1(k)$, thus, the estimation of h_1 and $s(k)$ is easy: we have $h_1 = |x(k)|$ (since $|s(k) = 1|$ and $h_1 > 0$) and so $s(k) = x(k)/h_1$.

Example 2 *Consider the following system with 4 sources and 1 observation:*

$$x(k) = -0.4326 s_1(k) + 1.2656 s_2(k) + 0.1553 s_3(k) - 0.2877 s_4(k)$$

The mixing vector $\mathbf{h} = [-0.4326, 1.2656, 0.1553, -0.2877]$ *does not satisfy (1.16). The algorithm will recover the vector* $\hat{\mathbf{h}} = [1.2656, 0.4326, 0.2877, 0.1553]$ *which does satisfy (1.16) and it is identical to* $\mathbf{h}$ *except for the permutation and sign changes of its elements.*
Step 1: *The sorted sequence of centers is*

$$\begin{aligned} c = \{ \quad & -2.1412, -1.8306, -1.5658, -1.2760, -1.2552, -0.9654, -0.7006, -0.3900, \\ & 0.3900, 0.7006, 0.9654, 1.2552, 1.2760, 1.5658, 1.8306, 2.1412 \ \} \end{aligned}$$

Step 2: *Using Eqs. (1.23), (1.24) we compute* $\hat{h}_3 = 0.2877$, $\hat{h}_4 = 0.1553$.
Step 3: *Using Eqs. (1.20), (1.21), (1.22), we obtain* $\delta_1 = 0.3106$, $\delta_2 = 0.5754$, $\delta_3 = 0.8860$.
Step 4: *Remove* $\{c_1, c_2, c_3, c_1+\delta_3\} = \{-2.1412, -1.8306, -1.5658, -1.2552\}$ *from c. Set* $c'_1 = -1.6982$.
New sorted sequence:

$$\begin{aligned} c = \{ \quad & -1.2760, -0.9654, -0.7006, -0.3900, \\ & 0.3900, 0.7006, 0.9654, 1.2552, 1.2760, 1.5658, 1.8306, 2.1412 \ \}. \end{aligned}$$

Step 5: *Remove* $\{c_1, c_1+\delta_1, c_1+\delta_2, c_1+\delta_3\} = \{-1.2760, -0.9654, -0.7006, -0.3900\}$ *from* c. *Set* $c_2' = -0.8330$. *New sorted sequence:*

$$c = \{0.3900, 0.7006, 0.9654, 1.2552, 1.2760, 1.5658, 1.8306, 2.1412\}.$$

Step 6: *Remove* $\{c_1, c_1+\delta_1, c_1+\delta_2, c_1+\delta_3\} = \{0.3900, 0.7006, 0.9654, 1.2760\}$. *Set* $c_3' = 0.8330$. *New sorted sequence:*

$$c = \{1.2552, 1.5658, 1.8306, 2.1412\}.$$

Step 6: *Remove* $\{c_1, c_1+\delta_1, c_1+\delta_2, c_1+\delta_3\} = \{1.2552, 1.5658, 1.8306, 2.1412\}$. *Set* $c_4' = 1.6982$. *New sorted sequence:*

$$c = \emptyset.$$

The new sequence $c' = \{-1.6982, -0.8330, 0.8330, 1.6982\}$, *yields the estimates of the remaining mixing parameters* $\hat{h}_1 = 1.2656$, $\hat{h}_2 = 0.4326$. ■

Two mixtures

In the case of $m = 2$ mixtures the observed data $\mathbf{x}(k)$ lie in the 2-dimensional space $\mathbb{R}^2$. Although it is possible to see each mixture separately as a single-mixture-multiple-sources problem, as the one treated in Section 1.2.1, this is not the most efficient approach to the problem. It turns out that the 2-D structure of the output constellation reveals the mixing operator $\mathbf{H}$ in a very elegant and straightforward way. To see that, let us start by considering the data constellation of a binary antipodal signal $s_1(k)$ (Figure 1.3a). The constellation actually consists of two points on the real axis: $s^- = -1$ and $s^+ = 1$. Next, consider a linear transformation from $\mathbb{R}^1$ to $\mathbb{R}^2$ which maps $s_1(k)$ to a vector signal $\mathbf{x}^{(1)}(k) = [x_1^{(1)}(k), x_2^{(1)}(k)]^T$:

$$\mathbf{x}^{(1)}(k) = \mathbf{h}_1 s_1(k) \tag{1.27}$$

The linear operator $\mathbf{h}_1 = [h_{11}, h_{12}]^T$ is a 2-dimensional vector shown in Fig. 1.3b. The constellation of $\mathbf{x}(k)$ is shown in Fig. 1.3c and it also consists of two points $\mathbf{x}^- = -\mathbf{h}_1 = s^-\mathbf{h}_1$ and $\mathbf{x}^+ = \mathbf{h}_1 = s^+\mathbf{h}_1$.

Now let us look at shape of the data cloud corresponding to the linear combination of several binary antipodal sources $s_1(k)$, ..., $s_n(k)$. It is instructive to study the shape of this cloud as n increases gradually from $n = 2$ and upwards. The linear mixture of $n = 2$ sources

$$\mathbf{x}^{(2)}(k) = \mathbf{h}_1 s_1(k) + \mathbf{h}_2 s_2(k) \tag{1.28}$$

has the geometric structure shown in Figure 1.4b, for the mixing vectors $\mathbf{h}_1$, $\mathbf{h}_2$, shown in Figure 1.4a. The data cluster contains four points: $\mathbf{x}^{++} = s^+\mathbf{h}_1 + s^+\mathbf{h}_2$, $\mathbf{x}^{+-} = s^+\mathbf{h}_1 + s^-\mathbf{h}_2$, $\mathbf{x}^{-+} = s^-\mathbf{h}_1 + s^+\mathbf{h}_2$, and $\mathbf{x}^{--} = s^-\mathbf{h}_1 + s^-\mathbf{h}_2$.

Adding a third source $s_3(k)$ with the mixing vector $\mathbf{h}_3$ the data mixture

$$\mathbf{x}^{(3)}(k) = \mathbf{h}_1 s_1(k) + \mathbf{h}_2 s_2(k) + \mathbf{h}_3 s_3(k) \tag{1.29}$$

has the constellation shown in Figure 1.5. Now the data cluster contains eight points: $\mathbf{x}^{+++} = s^+\mathbf{h}_1 + s^+\mathbf{h}_2 + s^+\mathbf{h}_3$, $\mathbf{x}^{++-} = s^+\mathbf{h}_1 + s^+\mathbf{h}_2 + s^-\mathbf{h}_3$, $\mathbf{x}^{+-+} = s^+\mathbf{h}_1 + s^-\mathbf{h}_2 + s^+\mathbf{h}_3$, $\mathbf{x}^{+--} = s^+\mathbf{h}_1 + s^-\mathbf{h}_2 + s^-\mathbf{h}_3$, $\mathbf{x}^{-++} = s^-\mathbf{h}_1 + s^+\mathbf{h}_2 + s^+\mathbf{h}_3$, $\mathbf{x}^{-+-} = s^-\mathbf{h}_1 + s^+\mathbf{h}_2 + s^-\mathbf{h}_3$, $\mathbf{x}^{--+} = s^-\mathbf{h}_1 + s^-\mathbf{h}_2 + s^+\mathbf{h}_3$, and $\mathbf{x}^{---} = s^-\mathbf{h}_1 + s^-\mathbf{h}_2 + s^-\mathbf{h}_3$.

By simple inspection of Figs. 1.3, 1.4, and 1.5, one can make the following useful observations:

1. The number of cluster points is 2^n, where n is the number of binary sources.
2. The data constellation is a symmetric, self-repetitive figure. While the symmetry is obvious, the self-repetitive structure can be seen by comparing, for example, Fig. 1.5b against Fig. 1.4b. The first consists of two copies of the latter shifted by the vectors $-\mathbf{h}_3$ and $\mathbf{h}_3$. The same is true for Figs. 1.4b and 1.3c except that the shift is by the vectors $-\mathbf{h}_2$ and $\mathbf{h}_2$.
3. For every cluster point there exist n copies at the directions $\mathbf{h}_1$ or $-\mathbf{h}_1$, and $\mathbf{h}_2$ or $-\mathbf{h}_2$, ..., and $\mathbf{h}_n$ or $-\mathbf{h}_n$.

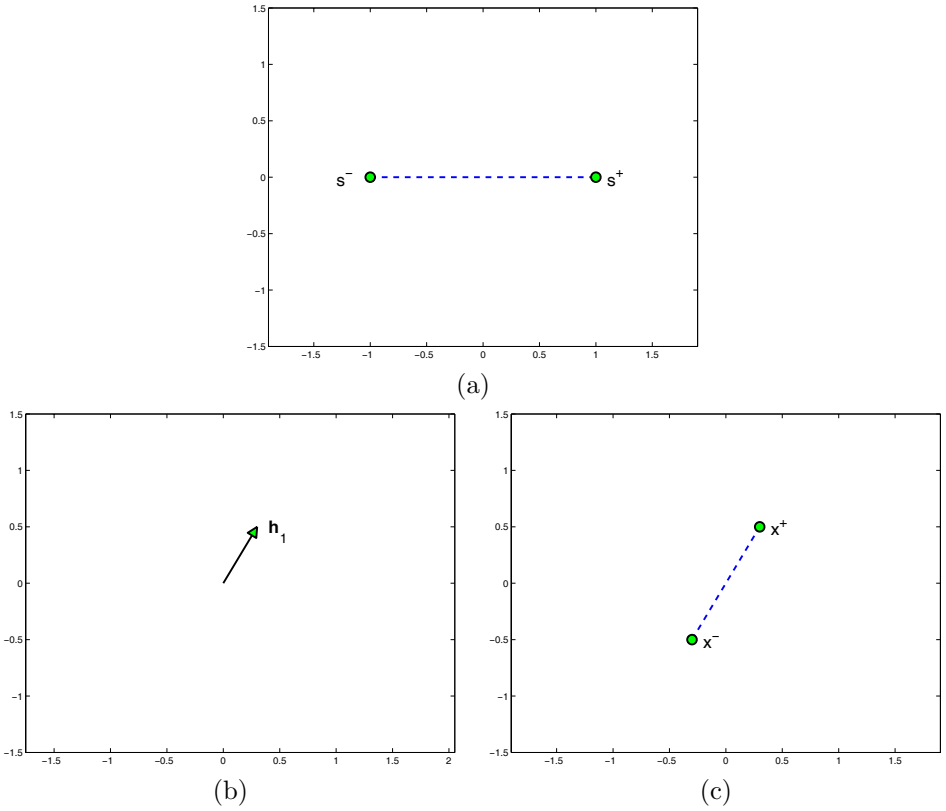


Figure 1.3: (a) Data constellation of a binary antipodal signal $s_1(k)$. (b) Linear transformation vector $\mathbf{h}_1$. (c) Data constellation of the transformed signal $\mathbf{x}^{(1)}(k) = \mathbf{h}_1 s_1(k)$.

It is even more interesting and, in fact, very useful to study the properties of the *convex hull* of the data constellation set. By definition, the convex hull of a set of points in 2-D space is the smallest polygon that contains them or, in other words, the *bounding polygon* for these points. Figures 1.6a-c show the convex hulls H_1, H_2, and H_3, for the data constellations corresponding to the mixtures $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$, and $\mathbf{x}^{(3)}$, respectively. Let d is the distance between the two alphabet symbols. It can be shown that any convex hull H satisfies the following properties (for the proof see (Diamantaras, 2002)):

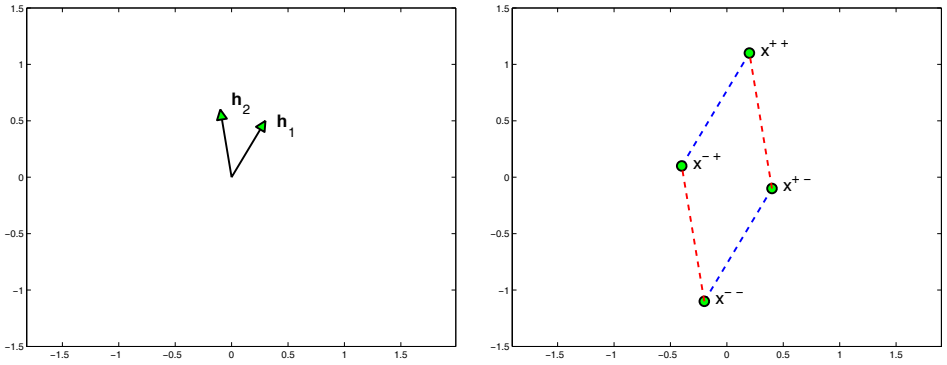
1. Every edge $\mathbf{e}$ of H is parallel to some mixing vector $\mathbf{h}_i$, $i \in \{0, 1, \cdots, n\}$. Also, $\mathbf{e}$ has length $d\|\mathbf{h}_i\|$. For the binary antipodal alphabet $\mathcal{A}_a$, we have $d = 2$.

2. Every vector $\mathbf{h}_i$ corresponds to a pair of edges, i.e. it is parallel to two edges $\mathbf{e}_i$ and $\mathbf{e}'_i$ of equal length $d\|\mathbf{h}_i\|$. It follows that H has $2n$ edges.

3. H is symmetric. If the alphabet is symmetric around 0 (e.g. $\mathcal{A}_a$) then the center of symmetry is the point $\mathbf{x}_O = 0$. Otherwise, the center of symmetry is a non-zero point $\mathbf{x}'_O \in \mathbb{R}^m$.

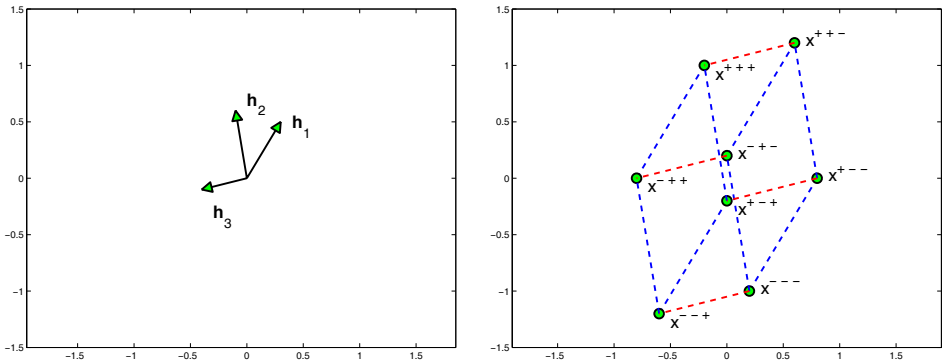
These important results show that there is a two-to-one correspondence between the edges of the convex hull and the unknown mixing vectors: there is a pair of edges parallel to each mixing vector and furthermore, the edges have length equal to d times the length of their corresponding mixing vectors. Thus we easily come to the following procedure for the identifying the $\mathbf{h}_i$'s:

Algorithm 2 (: n binary sources, 2 observations)
Step 1. *Find the constellation set $\mathcal{X}$ of the 2-D mixture $\mathbf{x}(k)$.*
Step 2. *Compute the convex hull H of $\mathcal{X}$.*
Step 3. *H consists of $2n$ edge pairs $\{\mathbf{e}_i, \mathbf{e}'_i\}$, $\mathbf{e}_i \| \mathbf{e}'_i$, $i = 1, \cdots, n$. The number of sources is n.*
Step 4. *The mixing vectors are: $\mathbf{h}_i = \mathbf{e}_i/d$, upto an unknown ordering and sign.*



(a) (b)

Figure 1.4: (a) Mixing vectors $\mathbf{h}_1$, $\mathbf{h}_2$. (b) Data cluster for the mixture $\mathbf{x}^{(2)}(k) = \mathbf{h}_1 s_1(k) + \mathbf{h}_2 s_2(k)$ of two binary antipodal sources.



(a) (b)

Figure 1.5: (a) Mixing vectors $\mathbf{h}_1$, $\mathbf{h}_2$, $\mathbf{h}_3$. (b) Data cluster for the mixture $\mathbf{x}^{(3)}(k) = \mathbf{h}_1 s_1(k) + \mathbf{h}_2 s_2(k) + \mathbf{h}_3 s_3(k)$ of three binary antipodal sources.

Of course, the original order and sign of the vectors are irretrievable. As we have seen, this is a general, problem-inherent limitation and it is not specific to this (or any other) particular method. In fact, the limitation cannot be overcome without additional information regarding the sources or the mixing operators.

More than 2 mixtures.

It is not difficult to see that the whole convex hull idea can be extended to the case where $m \geq 3$. Again, the edges of the convex hull will be parallel to the mixing vectors $\mathbf{h}_i$ except that, now, the convex hull lies in $\mathbb{R}^m$. The algorithms for computing the convex hull in m-dimensional spaces are not as simple as the ones for the 2-D case. For a comprehensive discussion of this topic see (Preparata and Shamos, 1985).

1.2.2 Instantaneous mixtures of M-ary alphabet sources

The results of Section 1.2.1 can be easily extended to M-ary signals, i.e. signals whose alphabet contains M discrete and equally distributed values. For example, the alphabet $\mathcal{A}_5 = \{-1, -1/2, 0, 1/2, 1\}$, contains $M = 5$ symbols symmetrically distributed around 0. Similar results, as in the binary case, hold here as well. Again, the convex hull directly connects the constellation geometry with the unknown mixing

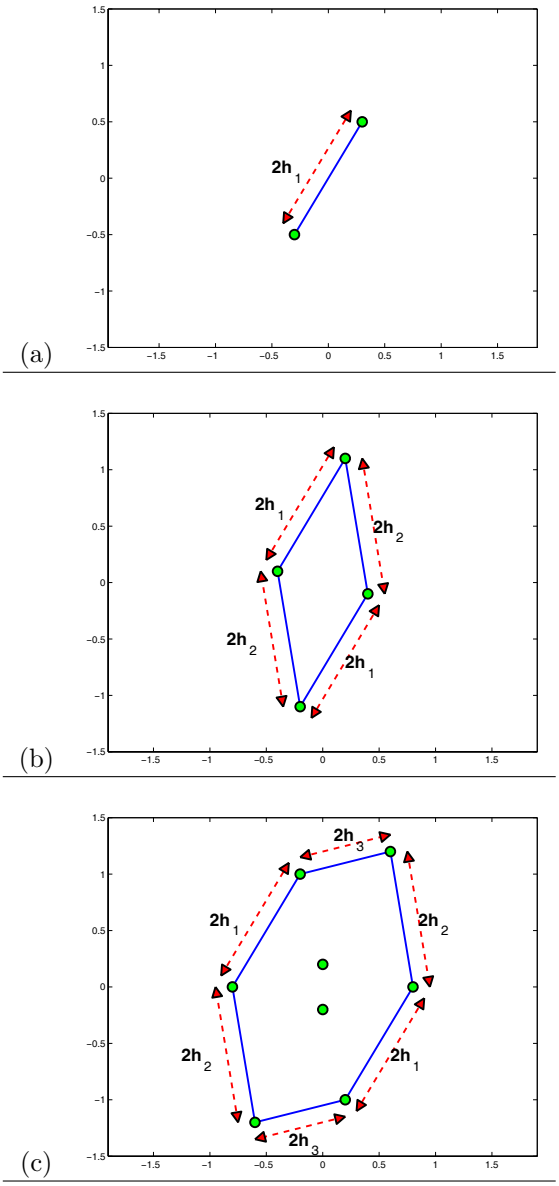


Figure 1.6: Convex hulls for data constellations of mixtures of n binary sources. (a) $n = 1$, (b) $n = 2$, (c) $n = 3$.

Table 1.1: True and estimated mixing vectors.

$\mathbf{h}_1$	$\hat{\mathbf{h}}_1$	$\mathbf{h}_2$	$\hat{\mathbf{h}}_2$	$\mathbf{h}_3$	$\hat{\mathbf{h}}_3$
0.3000	0.3017	− 0.1000	−0.0960	− 0.4000	0.3917
0.5000	0.4902	0.6000	0.6089	−0.1000	0.1009

vectors. Let d be the distance between the maximum and minimum symbols in the M-ary alphabet $\mathcal{A}_M$

$$d = \max\{\mathcal{A}_M\} - \min\{\mathcal{A}_M\}$$

Also let H be the convex hull of the constellation $\mathcal{X}$ of the mixture $\mathbf{x}(k) = \mathbf{h}_1 s_1(k) + \cdots + \mathbf{h}_n s_n(k)$

1. The number of cluster points is M^n, where n is the number of M-ary sources.
2. The data constellation is a symmetric, self-repetitive figure.
3. Every edge $\mathbf{e}$ of H is parallel to some mixing vector $\mathbf{h}_i$, $i \in \{0, 1, \cdots, n\}$, and $\mathbf{e}$ has length $d\|\mathbf{h}_i\|$.
4. Every vector $\mathbf{h}_i$ corresponds to a pair of edges, i.e. it is parallel to two edges $\mathbf{e}_i$ and $\mathbf{e}'_i$ of equal length $d\|\mathbf{h}_i\|$. It follows that H has $2n$ edges.
5. H is symmetric. For alphabets symmetric around zero the center of symmetry is $\mathbf{x}_O = 0$.

We may use Algorithm 2 without modifications for the solution of the M-ary case as well.

1.2.3 Noisy data

The analysis of the previous subsections pertains to systems with noiseless outputs. In most applications however, the observation is burdened with noise, either because the system itself is noisy or the receiving device introduces errors in the measurements. The additive noise model is commonly used for describing the observation error:

$$\mathbf{x}(k) = \mathbf{H}\mathbf{s}(k) + \mathbf{v}(k) \tag{1.30}$$

Without loss of generality, and for the sake of visualization we shall focus on the two-output case. An entirely similar discussion holds for the cases $n = 1$ or $n > 2$. The vector signal $\mathbf{v}(k) = [v_1(k), \cdots, v_n(k)]^T$, contains the noise components $v_i(k)$ for each observed output signal $i = 1, \cdots, n$. The constellation of $\mathbf{x}$ is now less crisp since the true centers are surrounded by a cloud of points (Fig. 1.8a). The methods presented in Sections 1.2.1 and 1.2.2 can be still applied preceded by a clustering process that will estimate the actual centers from the noisy data cloud. Such clustering methods include the ISODATA or K-means algorithm (Lloyd, 1982; MacQueen, 1967; Duda et al., 2001), the EM algorithm (Dempster et al., 1977), the neural gas algorithm (Martinetz et al., 1993), Kohonen's Self-Organizing Feature Maps (SOM) (**?**), RBF neural networks (Moody and Darken, 1989), and many others. For a detailed treatment of clustering methods refer to (Theodoridis and Koutroubas, 1998). Figure 1.8b shows the estimation of the true centers using the k-means algorithm in a system with 3 binary inputs and 2 linear output mixtures with noise power at 15db. Notice that the estimation errors inside the convex hull do not affect the results. It is only the errors at the boundary that are significant. We apply the blind identification method of Section 1.2.1 using the estimated centers provided by k-means, obtaining the results shown in Table 1.1.

1.2.4 Convolutive mixtures of binary sources

The convolutive mixtures of binary sources are described by the output of the MIMO FIR system (1.1). The blind problems related to such systems are considerably more difficult than the corresponding instantaneous mixture problems, but at the same time, they are much more important. Convolutive mixing models, for example, can describe multipath and crosstalk phenomena in wireless communications, being

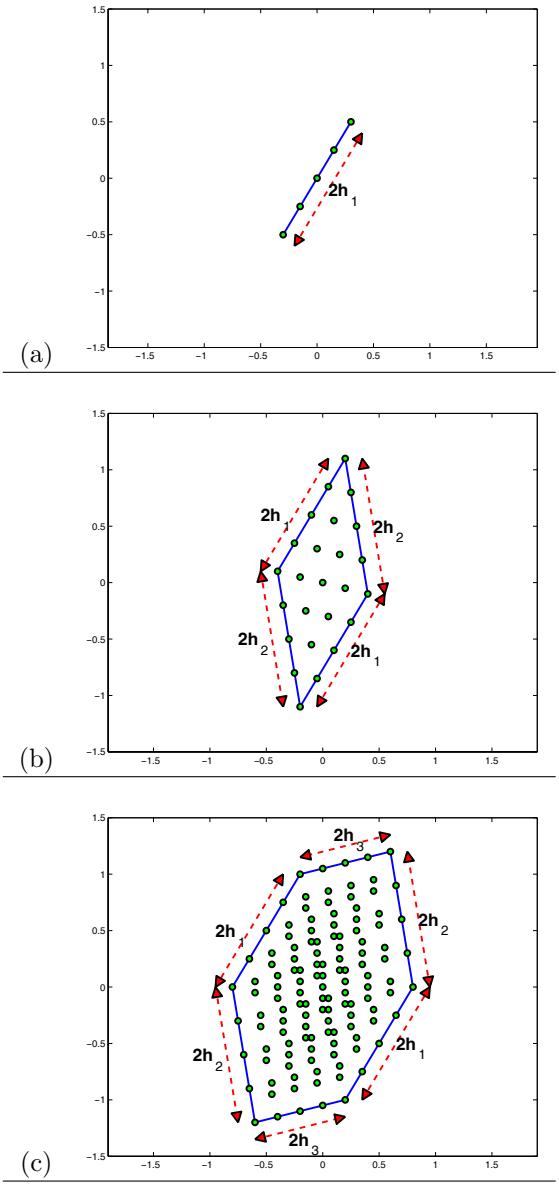
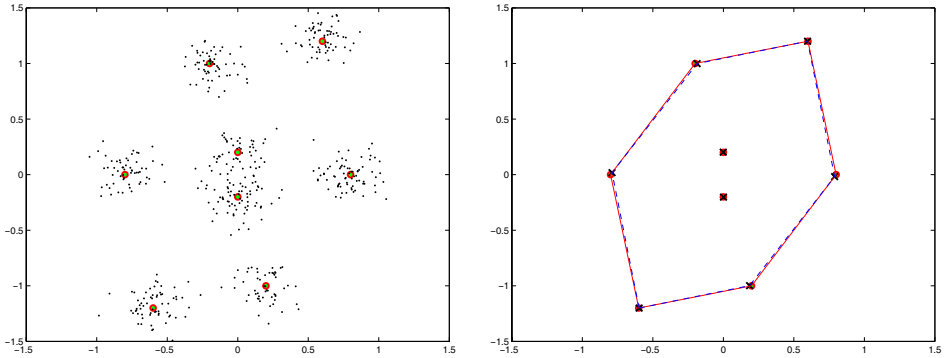


Figure 1.7: Convex hulls of mixture constellations from n M-ary sources ($M = 5$). The source symbols are drawn from the alphabet $\{-1, -0.5, 0, 0.5, 1\}$, with maximum distance $d = 2$. (a) $n = 1$, (b) $n = 2$, (c) $n = 3$.



(a) (b)

Figure 1.8: (a) Data constellation for a noisy memoryless linear system with 3 binary inputs and 2 outputs (mixtures). The noise level is 15db. Superimposed are the true cluster centers marked with 'o'. (b) True cluster centers (o) and estimated cluster centers (x) using the k-means algorithm. Also shown is the true convex hull (solid line) and the estimated convex hull (dashed line).

in that sense, much more realistic than instantaenous models. In this section we shall approach the blind source separation and blind system identification problems of MIMO FIR models using the geometric properties of the data constellation. We shall treat, first, the simpler Single-Input Single-Output (SISO) problem and then continue on to the Multi-Input Single-Output (MISO) case. The proper MIMO problem is not explicitly discussed since it can be seen as a multitude of m decoupled MISO problems.

Blind SISO deconvolution as instantaneous BSS

In this section we shall use the results of the previous sections to solve the blind SISO identification and deconvolution problems. Our approach is to relate any given SISO system with an overdetermined instantaneous mixtures model, hence the same methods can be applied as in Sections 1.2.1 and 1.2.1. Let us consider a linear, FIR, single-input single-output (SISO) system with a binary antipodal input $s(k)$,

$$x(k) = \sum_{i=0}^{L-1} h_i s(k-i) \tag{1.31}$$

We shall assume that the impulse response h_i, $i = 0, \cdots, L-1$, is real. Let us create a vector sequence $\mathbf{x}(k)$ using time-windowing of length m on the output sequence $x(k)$

$$\mathbf{x}(k) = [x(k), \cdots, x(k-m+1)]^T \tag{1.32}$$

Then using the system Equation (1.31) we have

$$\mathbf{x}(k) = \mathbf{H}\mathbf{s}(k) \tag{1.33}$$

where $\mathbf{H}$ is the *Toeplitz system matrix*

$$\mathbf{H} = \begin{bmatrix} h_0 & h_1 & \cdots & h_{L-1} & 0 & \cdots & 0 \\ 0 & h_0 & \cdots & h_{L-2} & h_{L-1} & 0 & 0 \\ & & \ddots & & & \ddots & \\ 0 & \cdots & 0 & h_0 & \cdots & h_{L-2} & h_{L-1} \end{bmatrix} \tag{1.34}$$

and

$$\mathbf{s}(k) = [s(k),\, s(k-1), \cdots,\, s(k-m-L+2)]^T \tag{1.35}$$

Now, Eq. (1.33) describes m linear instantaneous mixtures $x'_i(k) = x(k-i), \quad i = 0, \cdots, m-1$, of n sources $s'_j(k)$ defined as follows:

$$s'_j(k) = s(k-j+1), \qquad j = 1, 2, \cdots, n = m + L - 1$$

Thus, we have successfully transformed the problem into the same form treated in Section 1.2.1:

$$x'_i(k) = \sum_{j=1}^{n} h_{ij} s'_j(k) \tag{1.36}$$

where h_{ij} is the (i,j)-th element of $\mathbf{H}$. Equivalently, we can write

$$\mathbf{x}(k) = \sum_{j=1}^{n} \mathbf{h}_j s'_j(k) \tag{1.37}$$

where the mixing vectors $\mathbf{h}_1$, ..., $\mathbf{h}_n$ are the columns of $\mathbf{H}$. Given the above formulation, the results of Section 1.2.1 apply directly to this problem. There are, however, some special points to be noted:

1. For any non-trivial FIR filter of length $L > 1$, the number of observations x'_1, ..., x'_m, is necessarily less than the number of sources s'_1, ..., s'_n, since $n = m + L - 1 > m$.
2. The mixing vectors have no arbitrary form. For example, $\mathbf{h}_1$ has the form $[\times, 0, \cdots, 0]^T$ and $\mathbf{h}_n$ has the form $[0, \cdots, 0, \times]^T$.
3. The sources are not independent. In fact, any one is a shifted version of any other.

Next, we shall give examples for two cases (a) $m = 1$ and (b) $m = 2$.

Example 3 (Time widow of length $m = 1$.) *Suppose that we observe the output $x(k)$ of a SISO filter $\mathbf{h} = [-0.4937, -1.1330, 0.7632, 0.1604]^T$ excited by the binary input $s(k)$. Using Algorithm 1 we shall identify the filter with the necessary permutation and sign changes so that the estimated taps will be positive and arranged in decreasing order. Thus we shall obtain $\hat{\mathbf{h}} = [1.1330, 0.7632, 0.4937, 0.1604]^T$ and so*

$$\begin{aligned} x(k) &= \hat{h}_1 \hat{s}'_1(k) + \hat{h}_2 \hat{s}'_2(k) + \hat{h}_3 \hat{s}'_3(k) + \hat{h}_4 \hat{s}'_4(k) \\ &= (-h_2)(-s'_2(k)) + h_3 s'_3(k) + (-h_1)(-s'_1(k)) + h_4 s'_4(k) \end{aligned}$$

Obviously, the estimated sources $\hat{s}'_i$ correspond to the true "sources" s'_i as follows:

$$\begin{aligned} \hat{s}'_1(k) &= -s'_2(k) = -s(k-1), \\ \hat{s}'_2(k) &= s'_3(k) = s(k-2), \\ \hat{s}'_3(k) &= -s'_1(k) = -s(k), \\ \hat{s}'_4(k) &= s'_4(k) = s(k-3). \end{aligned}$$

Since the signals $\hat{s}'_i$ are shifted versions of the original source, $s(k)$, it is easy to recover their correct order and relative sign changes by computing for each signal, the time-shift with maximum correlation to an arbitrary reference, for example, $\hat{s}'_1$. Applying the same ordering and sign changes to $\hat{\mathbf{h}}$ we obtain $\pm\mathbf{h}$. ■

Example 4 (Time widow of length $m = 2$.) *Consider the same SISO filter as before and let us use time-widowing of length $m = 2$ to obtain the vector sequence $\mathbf{x}(k)$:*

$$\begin{aligned} \mathbf{x}(k) &= \begin{bmatrix} x(k) \\ x(k-1) \end{bmatrix} \\ &= \begin{bmatrix} -0.4937 & -1.1330 & 0.7632 & 0.1604 & 0 \\ 0 & -0.4937 & -1.1330 & 0.7632 & 0.1604 \end{bmatrix} \begin{bmatrix} s(k) \\ s(k-1) \\ s(k-2) \\ s(k-3) \\ s(k-4) \end{bmatrix} \end{aligned}$$

Using Algorithm 2 we estimate the original mixing vectors $\mathbf{h}_1 = [-0.4937, 0]^T$, $\mathbf{h}_2 = [-1.1330, -0.4937]^T$, $\mathbf{h}_3 = [0.7632, -1.1330]^T$, $\mathbf{h}_4 = [0.1604, 0.7632]^T$, $\mathbf{h}_5 = [0, 0.1604]^T$, *but with an arbitrary order and sign change. The estimated mixing vectors can be put to the correct order by observing that the true system parameters satisfy the following*

$$\begin{aligned} h_{1,1} &= h_{2,2} = -0.4937, \\ h_{2,1} &= h_{3,2} = -1.1330, \\ h_{3,1} &= h_{4,2} = 0.7632, \\ h_{4,1} &= h_{5,2} = 0.1604, \\ h_{5,1} &= h_{1,2} = 0 \ . \end{aligned}$$

Since the sign of each estimated vector is arbitrary, we compare the absolute values $|\hat{h}_{i,1}|$, *against* $|\hat{h}_{j,2}|$ *and we change the signs of either* $\hat{\mathbf{h}}_i$ *or* $\hat{\mathbf{h}}_j$, *as necessary, so that* $\hat{h}_{i,1} = \hat{h}_{j,2}$. *Once the correct order of the mixing vectors is retrieved we automatically obtain the correct filter impulse response (up to a sign). Subsequently, the system input,* $s(k)$, *is retrieved using standard (non-blind) deconvolution methods.* ■

Blind SISO identification

An alternative approach for identifying the impulse response $\mathbf{h} = [h_0, \cdots, h_{L-1}]^T$ of a general SISO system (1.31) has been proposed by Yellin and Porat Yellin and Porat (1993). The method is not based on constellation geometry but rather on the properties of the successor values of "equivalent" observations. The source symbols $s(k)$ may be drawn from an M-ary alphabet $\mathcal{A}_M = \{\pm 1, \cdots, \pm(M/2)\}$ (M=even). Before we proceed we need to introduce the concept of *equivalence* between two observations:

Definition 1 (Observation Equivalence) *Two observations* $x(k)$ *and* $x(l)$ *are said to be equivalent if the input values that produce them according to Eq. (1.31) are identical:* $s(k-i) = s(l-i)$, *for all* $i = 0, \cdots, L-1$.

Note that two equivalent observations are necessarily equal, but the converse may not be true. Indeed, it is possible that two equal observations $x(k) = x(l)$, are produced by two different strings of input symbols $[s(k), \cdots, s(k-L+1)] \neq [s(l), \cdots, s(l-L+1)]$.

Consider four sets of $(N+1)$ consecutive observations from (1.31): $\mathcal{X}_j = \{x(j), x(j+1), \cdots, x(j+N)\}$, $\mathcal{X}_k = \{x(k), x(k+1), \cdots, x(k+N)\}$, $\mathcal{X}_l = \{x(l), x(l+1), \cdots, x(l+N)\}$, $\mathcal{X}_m = \{x(m), x(m+1), \cdots, x(m+N)\}$. Further assume that the pairs $\{x(j), x(k)\}$ and $\{x(l), x(m)\}$ are equivalent. Define

$$\begin{aligned} \sigma_{jki} &= [s(j+i) - s(k+i)]/2, \\ \sigma_{lmi} &= [s(l+i) - s(m+i)]/2; \qquad i = 1, \cdots, N \end{aligned} \tag{1.38}$$

and note that σ_{jki}, $\sigma_{lmi} \in \mathcal{A}_M^0 = \mathcal{A}_M \cup 0$. Let the following conditions be true:

A. σ_{jk1}, σ_{lm1} are nonzero and coprime, i.e. their greatest common divisor is 1.

B. For all $\alpha, \beta \in \mathcal{A}_M$

$$\left|\frac{\sigma_{jk1}}{\sigma_{lm1}}\right| = \left|\frac{\alpha}{\beta}\right| \Rightarrow |\alpha| = |\sigma_{jk1}|, |\beta| = |\sigma_{lm1}|$$

C. For all $\alpha, \beta \in \mathcal{A}_M^0$,

$$\frac{\sigma_{jk1}}{\sigma_{lm1}} \neq \frac{\sigma_{jki} - \alpha}{\sigma_{lmi} - \beta}, \quad \text{for all } i = 2, \cdots, N$$

The method starts by identifying the first filter tap h_0 upto a sign, and continues by recursively identifying the remaining taps given the previous ones. Begin with the remark that $x(j)$ and $x(k)$ are equivalent, so $[s(j), \cdots, s(j-L+1)] = [s(k), \cdots, s(k-L+1)]$. Then, the *successor values* of $x(j)$, $x(k)$, can be written as

$$x(j+1) = h_0 s(j+1) + \sum_{i=1}^{L-1} h_i s(j+1-i),$$

$$x(k+1) = h_0 s(k+1) + \sum_{i=1}^{L-1} h_i s(k+1-i),$$

so,

$$\frac{x(j+1) - x(k+1)}{2} = \sigma_{jk1} h_0. \tag{1.39}$$

Similarly, for $x(l+1)$, $x(m+1)$:

$$\frac{x(l+1) - x(m+1)}{2} = \sigma_{lm1} h_0, \tag{1.40}$$

and so,

$$\frac{x(j+1) - x(k+1)}{x(l+1) - x(m+1)} = \frac{\sigma_{jk1}}{\sigma_{lm1}} \tag{1.41}$$

By assumption B, the ratio $|\sigma_{jk1}/\sigma_{lm1}|$ is produced by a unique enumerator-denominator pair in $\mathcal{A}_M$. Thus both values σ_{jk1} and σ_{lm1} can be uniquely identified, upto a sign, leading to the magnitude estimation of h_0 by:

$$|h_0| = \frac{|x(j+1) - x(k+1)|}{2|\sigma_{jk1}|} = \frac{|x(l+1) - x(m+1)|}{2|\sigma_{lm1}|} \tag{1.42}$$

Without loss of generality, we may assume that $h_0 > 0$, and proceed to the estimation of h_1 as follows: write the second successors of $x(l)$, $x(k)$, as

$$x(j+2) = h_0 s(j+2) + h_1 s(j+1) + \sum_{i=2}^{L-1} h_i s(j+1-i),$$

$$x(k+2) = h_0 s(k+2) + h_1 s(k+1) + \sum_{i=2}^{L-1} h_i s(k+1-i),$$

hence,

$$\frac{x(j+2) - x(k+2)}{2} = \sigma_{jk2} h_0 + \sigma_{jk1} h_1. \tag{1.43}$$

Similarly,

$$\frac{x(l+2) - x(m+2)}{2} = \sigma_{lm2} h_0 + \sigma_{lm1} h_1. \tag{1.44}$$

The pair of Equations (1.43), (1.44) involve three unknowns: σ_{jk2}, σ_{lm2}, h_1. However, it turns out that since the first two unknowns come from the discrete set $\mathcal{A}_M^0$ and condition C is true, the solution is unique. Indeed, assume there existed two different solutions $\{\sigma_{jk2}^{(1)}, \sigma_{lm2}^{(1)}, h_1^{(1)}\}$, $\{\sigma_{jk2}^{(2)}, \sigma_{lm2}^{(2)}, h_1^{(2)}\}$. Then by (1.43), (1.44) we have

$$(\sigma_{jk2}^{(2)} - \sigma_{jk2}^{(1)}) h_0 = (h_1^{(2)} - h_1^{(1)}) \sigma_{jk1} \tag{1.45}$$

$$(\sigma_{lm2}^{(2)} - \sigma_{lm2}^{(1)}) h_0 = (h_1^{(2)} - h_1^{(1)}) \sigma_{lm1} \tag{1.46}$$

Thus,

$$\frac{\sigma_{jk1}}{\sigma_{lm1}} = \frac{\sigma_{jk2}^{(2)} - \sigma_{jk2}^{(1)}}{\sigma_{lm2}^{(2)} - \sigma_{lm2}^{(1)}}$$

which is impossible, according to assumption C. Therefore, there exists a unique solution to Eqs. (1.43), (1.44). From these equations there follows that

$$h_1 = \Big(\frac{x(j+2) - x(k+2)}{2} - \sigma_{jk2} h_0 \Big) / \sigma_{jk1} = \Big(\frac{x(l+2) - x(m+2)}{2} - \sigma_{lm2} h_0 \Big) / \sigma_{lm1}$$

so the unique h_1 can be obtained by finding the intersection between the sets

$$\mathcal{F}_1 = \left\{ \tfrac{x(j+2)-x(k+2)}{2\sigma_{jk1}} + \tfrac{\alpha h_0}{\sigma_{jk1}};\ \ \alpha \in \mathcal{A}_M^0 \right\}$$

$$\mathcal{F}_2 = \left\{ \tfrac{x(l+2)-x(m+2)}{2\sigma_{lm1}} + \tfrac{\beta h_0}{\sigma_{lm1}};\ \ \beta \in \mathcal{A}_M^0 \right\}$$

This is computationally trivial since the two sets are finite with few elements. Inductively, for h_i, $i > 2$, and given the values for h_0, ..., h_{i-1}, we form the "deflated" successors

$$\bar{x}(j+i+1) = x(j+i+1) - \sum_{p=1}^{i-1} \sigma_{jk(p+1)} h_{i+p} \tag{1.47}$$

$$\bar{x}(k+i+1) = x(k+i+1) - \sum_{p=1}^{i-1} \sigma_{jk(p+1)} h_{i+p} \tag{1.48}$$

$$\bar{x}(l+i+1) = x(l+i+1) - \sum_{p=1}^{i-1} \sigma_{lm(p+1)} h_{i+p} \tag{1.49}$$

$$\bar{x}(m+i+1) = x(m+i+1) - \sum_{p=1}^{i-1} \sigma_{lm(p+1)} h_{i+p} \tag{1.50}$$

and we obtain a set of two equations similar to (1.43), (1.44)

$$\frac{\bar{x}(j+i+1) - \bar{x}(k+i+1)}{2} = \sigma_{jk(i+1)} h_0 + \sigma_{jk1} h_i. \tag{1.51}$$

$$\frac{\bar{x}(l+i+1) - \bar{x}(m+i+1)}{2} = \sigma_{lm(i+1)} h_0 + \sigma_{lm1} h_i. \tag{1.52}$$

which are solved in a similar fashion, producing the unknown tap h_i. Thus, the whole approach is summarized in the following algorithm

Algorithm 3 (Yellin & Porat)
Step 1. *Collect T observation measurements.*
Step 2. *Find pairs of equivalent measurements. Estimate h_0 according to (1.42)*
Step 3. *Estimate h_1 using h_0 and the pairs of equivalent observations*
Step 4. *Continue with the estimation of h_2, ..., h_n given the previous estimates*
Step 5. *Use the estimated impulse response to deconvolve the observation sequence and obtain the system input.*

Remarks.

- The choice of pairs of equivalent observations (Step 2 in Algorithm 3) is far from trivial. The indices j, k, l, m, must satisfy various constraints so that the assumptions of the method are met. Firstly, according to condition A, we must have $\sigma_{jk1}, \sigma_{lm1} \neq 0$ implying that $x(j+1) \neq x(k+1)$, $x(l+1) \neq x(m+1)$. Secondly, according to condition C, for all $i = 2, \cdots, N$ the ratios $\sigma_{jki}/\sigma_{lmi}$ should not be equal to $\sigma_{jk1}/\sigma_{lm1}$. A thorough discussion on the implementation details are given in the original paper (Yellin and Porat, 1993).

- The method can be easily extended to handle complex input constellations (such as QAM) and/or complex filter taps.

- For the special case of i.i.d. input signals it is estimated that a sufficient batch size that guarantees $E > 2$ equivalent pairs of measurements is $T = 2.44E^{0.61}NM^{N/2}$.

- It is not possible to satisfy condition C if the source alphabet is binary ($\mathcal{A}_M = \mathcal{A}_a$), because there is a limited choice for the values of σ_{jki}, σ_{lmi}, which belong to the set $\mathcal{A}_a^0 = \{-1, 0, 1\}$.

MISO systems: direct source extraction

The blind source extraction directly from the output of a Multi-Input Single-Output (MISO) system is treated in (Diamantaras and Papadimitriou, 2005). This work is an extension of earlier work on SISO systems (Diamantaras and Papadimitriou, 2004a). The key to the approach in both cases is the

structure of the successor values of equivalent observations induced by the fact that the sources are binary. Subsequently, we shall present the results for the more general MISO case. Let us consider a multi-input single-output (MISO) model described by the following equation:

$$x(k) = \sum_{i=0}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-i) \tag{1.53}$$

where $\mathbf{h}_i$ for $i = 0, ..., L-1$, are a set of unknown real n-dimensional mixing vectors or *filter taps*. The source vector signal $\mathbf{s}(k) = [s_1(k) \cdots, s_n(k)]^T$ is composed of n independent binary antipodal signals: $s_i(k) \in \mathcal{A}_a$. The observations of the mixtures are real-valued scalars. For each k, the vector $\mathbf{s}(k)$ can take one of 2^n values denoted by $\mathbf{b}_i^{(n)}$, $i = 1, \cdots, 2^n$. The vector ${\mathbf{b}_i^{(n)}}^T$ is the i-th row of the matrix $\mathbf{B}^{(n)}$ defined in (1.17).

Let us extend the concept of observation equivalence, defined before for SISO systems, to MISO systems by simply replacing the scalar inputs with vector inputs. Each observation $x(k)$ is generated by the linear combination of L n-dimensional source vectors, therefore, the observation space $\mathcal{X} \ni x(k)$ is a discrete set consisting of, at most, 2^M elements, $M = nL$. The cardinality $|\mathcal{X}|$ will be less than 2^M if and only if there exist two different L-tuples $\{\mathbf{b}_{j_0}^{(n)}, \cdots, \mathbf{b}_{j_{L-1}}^{(n)}\}$ and $\{\mathbf{b}_{l_0}^{(n)}, \cdots, \mathbf{b}_{l_{L-1}}^{(n)}\}$, of binary vectors such that $\sum_{i=0}^{L-1} \mathbf{h}_i^T \mathbf{b}_{j_i}^{(n)} = \sum_{i=0}^{L-1} \mathbf{h}_i^T \mathbf{b}_{l_i}^{(n)}$. The following avoids this situation:

Assumption 1 *Two observations $x(k)$, $x(l)$, are equivalent if and only if they are equal.* ■

Hence, $|\mathcal{X}| = 2^M$. In other words, to every observation value $r \in \mathcal{X}$ corresponds a unique L-tuple $\{\bar{\mathbf{b}}_0(r), \cdots, \bar{\mathbf{b}}_{L-1}(r)\}$ of consecutive source vectors that generates this observation. No other observation value $r' \in \mathcal{X}$ corresponds to the same L-tuple of binary vectors. For any $x(k) = r$, we have

$$x(k) = \sum_{i=0}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r) \tag{1.54}$$

since, by definition,

$$\bar{\mathbf{b}}_i(r) = \mathbf{s}(k-i), \qquad \text{for } i = 0, \cdots, L-1.$$

Now the *successor observation*, $x(k+1)$, can be written as:

$$\begin{aligned} x(k+1) &= \mathbf{h}_0^T \mathbf{s}(k+1) + \sum_{i=1}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-(i-1)) \\ &= \mathbf{h}_0^T \mathbf{s}(k+1) + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r) \end{aligned} \tag{1.55}$$

Since $\mathbf{s}(k+1)$ is an n-dimensional binary antipodal vector, $x(k+1)$ can take one of the following 2^n possible values:

$$y_p(r) = \mathbf{h}_0^T \mathbf{b}_p^{(n)} + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r), \qquad p = 1, \cdots, 2^n \tag{1.56}$$

Note that the successor values $y_p(r)$ do not depend on the specific time index k but only on the observation value r. Therefore, each observation value r creates a class of successors $\mathcal{Y}(r)$ with cardinality $|\mathcal{Y}(r)| = 2^n$. Furthermore, we have $\sum_{p=1}^{2^n} \mathbf{b}_p^{(n)} = 0$, so the mean $\bar{y}(r)$ of the members of $\mathcal{Y}(r)$ is:

$$\begin{aligned} \bar{y}(r) &= \frac{1}{2^n} \sum_{p=1}^{2^n} y_p(r) \\ &= \frac{1}{2^n} \left(\mathbf{h}_0^T \sum_{p=1}^{2^n} \mathbf{b}_p^{(n)} + 2^n \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r) \right) \\ &= \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r). \end{aligned} \tag{1.57}$$

Now, let us replace every $x(k) = r$ with the mean $\bar{y}(r)$ to obtain a new sequence

$$\begin{aligned} x^{(2)}(k) &= \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r) \\ x^{(2)}(k) &= \sum_{i=1}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-i+1) \end{aligned} \tag{1.58}$$

The new MISO system (1.58) has the same taps as the original system (1.53) except that it is shorter since $\mathbf{h}_0$ is missing. We will say that the system has been deflated, and the whole procedure will be . An additional but trivial difference is that the source sequence is time-shifted. Based on the discussion above, the whole *filter-* or *system-deflation* method, is summarized as follows:

Algorithm 4 (System deflation.)
Step 1. *For every $r \in \mathcal{X}$ locate the set of time instances $\mathcal{K}(r) = \{k : x(k) = r\}$.*
Step 2. *Find the successor set $\mathcal{Y}(r) = \{x(k+1) : k \in \mathcal{K}(r)\}$. This set must contain 2^n distinct values $y_1(r)$, ..., $y_{2^n}(r)$.*
Step 3. *Compute the mean $\bar{y}(r) = 1/2^n \sum_{i=1}^{2^n} y_i(r)$.*
Step 4. *Replace $x(k)$ by $\bar{y}(r)$, for all $k \in \mathcal{K}(r)$.* ■

Clearly, for this method it is essential that all observation/successor pairs $[r, y_i(r)]$, $i = 1, \cdots, 2^n$, will appear, at least once, in the output sequence x. Applying the deflation method $L-1$ times, the system will be eventually reduced to an multi-input single-output instantaneous problem:

$$x^{(L)}(k) = \mathbf{h}_{L-1}^T \mathbf{s}(k-L+1) \tag{1.59}$$

The BSS problem of the type (1.59) has been treated in Section 1.2.1.

The main disadvantage of this method stems from the assumption that the dataset must contain every possible obsrervation/successor pair. As the size of the MISO system increases this assumption requires exponentially larger observation datasets. An alternative approach starts by observing that for any $r \in \mathcal{X}$ the *centered successors*:

$$c_i = y_i(r) - \bar{y}(r) = \mathbf{h}_0^T \mathbf{b}_i^{(n)} \qquad i = 1, \cdots, 2^n \tag{1.60}$$

are independent of r. Thus every observation has the same set of centered successors. We shall refer to the set $C = \{c_i;\ i = 1, \cdots, 2^n\}$ as the *centered successor constellation set* of system (1.53). C can be easily computed by first obtaining $\mathcal{Y}(r)$, for any r, and then subtracting the mean $\bar{y}(r)$ from each element $y_i(r) \in \mathcal{Y}(r)$. Note that C is symmetric in the sense that $c \in C \Leftrightarrow -c \in C$.

Now, for every observation value $r = x(k) \in \mathcal{X}$ we have

$$x(k) = \mathbf{h}_0^T \mathbf{s}(k) + \sum_{l=1}^{L-1} \mathbf{h}_l^T \mathbf{s}(k-l) \tag{1.61}$$

$$r = \mathbf{h}_0^T \mathbf{b}_i^{(n)} + \sum_{l=1}^{L-1} \mathbf{h}_l^T \bar{\mathbf{b}}_l(r), \qquad \text{some } i \tag{1.62}$$

$$= c_i + \sum_{l=1}^{L-1} \mathbf{h}_l^T \bar{\mathbf{b}}_l(r), \qquad \text{some } i \tag{1.63}$$

Furthermore, due to the symmetry of the constellation set, there exists a "dual" observation value $r^d \in \mathcal{X}$ such that

$$r^d = -c_i + \sum_{l=1}^{L-1} \mathbf{h}_l^T \bar{\mathbf{b}}_l(r) \tag{1.64}$$

$$r^d = r - 2c_i \tag{1.65}$$

Assume that for every observation $r \in \mathcal{X}$, there exists a unique index $j \in \{1, \cdots, 2^n\}$ such that $r - 2c_j \in \mathcal{X}$. Then the dual value r^d can be identified by testing all $r - 2c_j$, $j = 1, \cdots, 2^n$, for membership in the observation space $\mathcal{X}$. Let us now replace $x(k)$ by the average of r, r^d, to obtain

$$\tilde{x}^{(2)}(k) = (r + r^d)/2 = r - c_j = \sum_{l=1}^{L-1} \mathbf{h}_l^T \bar{\mathbf{b}}_l(r) \tag{1.66}$$

Note that $\mathbf{b}_l(r) = \mathbf{s}(k - l)$, so

$$\tilde{x}^{(2)}(k) = \sum_{l=1}^{L-1} \mathbf{h}_l^T \mathbf{s}(k - l) \tag{1.67}$$

Eq. (1.67) describes a new, shortened MISO system,

Assumption 2 *For* only one $r_0 \in \mathcal{X}$, *there exist at least* 2^n k_i, $i = 1, \cdots, 2^n \in \{1, 2, \cdots, K\}$ *such that* $x(k_i) = r_0$, $x(k_i + 1) = \sigma_i(r_0)$, $i = 1, \cdots, 2^n$. *In addition to that, every possible value of* $\mathcal{X}$ *exists at least once in the dataset.* ■

Summarizing the above results, our second method for obtaining the deflated system (1.67) is described below:

Algorithm 5 (System Deflation 2.)
Step 1. *Locate an observation value* r_0 *for which* 2^n *distinct successors* $\sigma_i(r_0)$, $i = 1, \cdots, 2^n$, *exist in the dataset*
Step 2. *Compute the successor constellation set* C *according to (1.60)*
Step 3. *For every observation* $r = x(k)$ *find the (unique) value* j *for which* $r - 2c_j \in \mathcal{X}$.
Step 4. *Replace* $x(k)$ *by* $r - c_j$ ■

Again, the $L - 1$ times repetition of this algorithm will reduce the system into a memoryless one

$$\tilde{x}^{(L)}(k) = \mathbf{h}_{L-1}^T \mathbf{s}(k) \tag{1.68}$$

which can be treated as described in section 1.2.4.

Example 5 (MISO system identification and source separation.) *We shall demonstrate the application of the second method via a specific example. Assume that we observe the output* $x(k)$ *of a two-input-one-output system (Fig. 1.9a). The system has two binary inputs* s_1, s_2, *convolution length* $L = 3$, *and filter taps* $\mathbf{h}_1 = [-0.9024, 1.5464]^T$, $\mathbf{h}_2 = [-0.6131, 0.7166]^T$, $\mathbf{h}_3 = [-0.4131, -0.1621]^T$. *The output constellation contains* $2^{nL} = 64$ *clusters:* $\mathcal{X} = \{\pm 4.3537, \pm 4.0295, \pm 3.5275, \cdots\}$. *Already, the first value* $r = -4.3537$ *has* $2^n = 4$ *distinct successors in the output sequence* $x(k)$. *From those successor values the centered successor constellation set is easily computed to be*

$$C = \{-2.4488, -0.6440, 0.6440, 2.4488\}.$$

After the deflation steps 2 and 3 we obtain a new sequence $x^{(2)}(k)$ *(Fig. 1.9b). Now the output constellation contains* $2^{n(L-1)} = 16$ *clusters:* $\mathcal{X}^{(2)} = \{\pm 1.9049, \pm 1.5807, \pm 1.0787, \cdots\}$ *and the centered successor constellation set is*

$$C^{(2)} = \{-1.3297, -0.1035, 0.1035, 1.3297\}.$$

We use this set to obtain a second deflated signal $x^{(3)}(k)$ *(Fig. 1.9c). This signal actually corresponds to an instantaneous mixture of the two sources. The output constellation has only 4 clusters:*

$$\mathcal{X}^{(3)} = \{-0.5752, -0.2510, 0.2510, 0.5752\}.$$

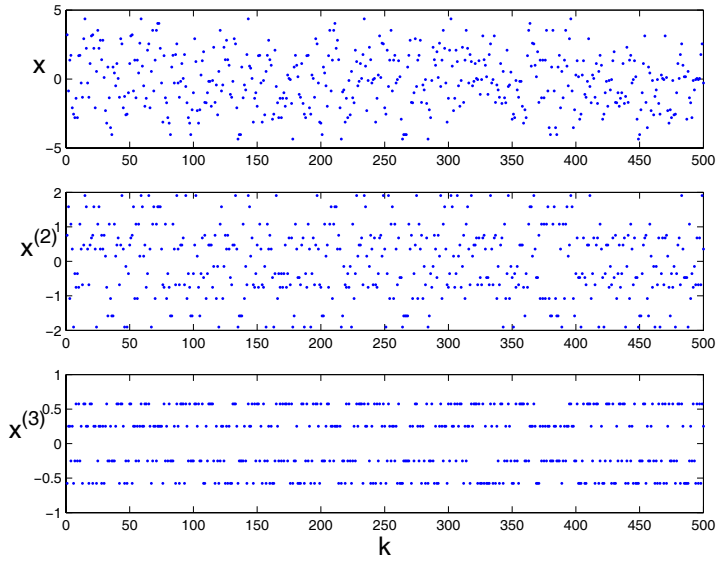


Figure 1.9: (top) Output signal from a 2-input-1-output FIR system of length $L = 3$. The output constellation contains $2^{nL} = 64$ distinct clusters. (middle) First deflated signal with $2^{n(L-1)} = 16$ clusters. (bottom) Second deflated signal with $2^{n(L-2)} = 4$ clusters. The last signal corresponds to an instantaneous mixture of the 2 sources.

We may apply Algorithm 1 from Section 1.2.1 to obtain an estimate of the mixing parameters and of the input signals as well. We obtain $\hat{h}_{3,1} = 0.4131 = -h_{3,1}$, $\hat{h}_{3,2} = 0.1621 = -h_{3,2}$. *Subsequently performing the optimization (1.15) for the estimation of the sources we get perfect reconstruction (except for the sign):*

$$\hat{s}_1(k) = -s_1(k),$$
$$\hat{s}_2(k) = -s_2(k).$$

■

1.3 Continuous sources

In Section 1.2 we exploited the constellation structure of signals generated by linear systems with finite alphabet inputs. In many applications, however, the range of values of the source data is continuous. In this case the geometrical properties of the signals can still be exploited to derive efficient deterministic blind separation methods provided that the sources are sparse, or the input distribution is bounded, or the number of observations is $m = 2$.

1.3.1 Early approaches: 2 mixtures, 2 sources

It is possible to generalize the geometric properties of binary signals described in Section 1.2.1, when the sources symbols are bounded. We start with the simplest case of two instantaneous mixtures x_1, x_2, and two sources s_1, s_2, $(m = n = 2)$:

$$\mathbf{x}(k) = [x_1(k),\, x_2(k)]^T = \mathbf{h}_1 s_1(k) + \mathbf{h}_2 s_2(k) \tag{1.69}$$

We shall describe two of the earliest and most characteristic methods by (Puntonet et al., 1995) and (Mansour et al., 2001).

The method of Puntonet e.a. The geometry of mixtures of binary signals bears similarity to the geometry of mixtures of bounded sources. Consider the mixing model (1.69) and let $s_1(k), s_2(k) \in [-B, B]$. The linear operation (1.69) transforms the original square source constellation (Fig. 1.10a) into a parallelogram-shaped constellation with edges parallel to the vectors $\mathbf{h}_1$ and $\mathbf{h}_2$ (Fig. 1.10b).

The blind identification task is then equivalent to finding the edges of the convex hull of the output constellation. Puntonet e.a. Puntonet et al. (1995) proposed a simple procedure for doing that. This procedure is composed of two steps:

- Locate the outmost corner $\mathbf{x}_O$ of the parallelogram by finding the observation with the maximum norm: $\mathbf{x}_O = \mathbf{x}(k_0)$, $k_0 = \arg\max_k\{\|\mathbf{x}(k)\|^2\}$.
- Translate the observations $\mathbf{x}'(k) = \mathbf{x}(k) - \mathbf{x}(k_0)$ such that $\mathbf{x}_O$ becomes the origin, and compute the slopes of the parallelogram by computing the minimum and maximum ratios: $r_{min} = \min_k(x'_2(k)/x'_1(k))$, $r_{max} = \max_k(x'_2(k)/x'_1(k))$. These are the ratios h_{12}/h_{11}, h_{22}/h_{21}, not necessarily in that order.

Once the slopes of the edges are determined, the mixing matrix is estimated by

$$\hat{\mathbf{H}} = \begin{bmatrix} 1 & 1/r_{min} \\ r_{max} & 1 \end{bmatrix} \tag{1.70}$$

Since $(r_{min}, r_{max}) = (h_{12}/h_{11})$ or (h_{22}/h_{21}) we have,

$$\hat{\mathbf{H}} = \begin{bmatrix} 1 & h_{21}/h_{22} \\ h_{12}/h_{11} & 1 \end{bmatrix} \text{ or } \begin{bmatrix} 1 & h_{11}/h_{12} \\ h_{22}/h_{21} & 1 \end{bmatrix} \tag{1.71}$$

Remember now that

$$\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2] = \begin{bmatrix} h_{11} & h_{21} \\ h_{12} & h_{22} \end{bmatrix}$$

so,

$$\hat{\mathbf{H}} = \mathbf{H}\begin{bmatrix} 1/h_{11} & 0 \\ 0 & 1/h_{22} \end{bmatrix} \text{ or } \hat{\mathbf{H}} = \mathbf{H}\begin{bmatrix} 0 & 1/h_{12} \\ 1/h_{21} & 0 \end{bmatrix}$$

In either case, the source estimate $\hat{\mathbf{s}}(k) = \hat{\mathbf{H}}^{-1}\mathbf{x}(k)$ will be

$$\hat{\mathbf{s}}(k) = [h_{11}s_1(k), h_{22}s_2(k)]^T \text{ or } \hat{\mathbf{s}}(k) = [h_{12}s_2(k), h_{21}s_1(k)]^T \tag{1.72}$$

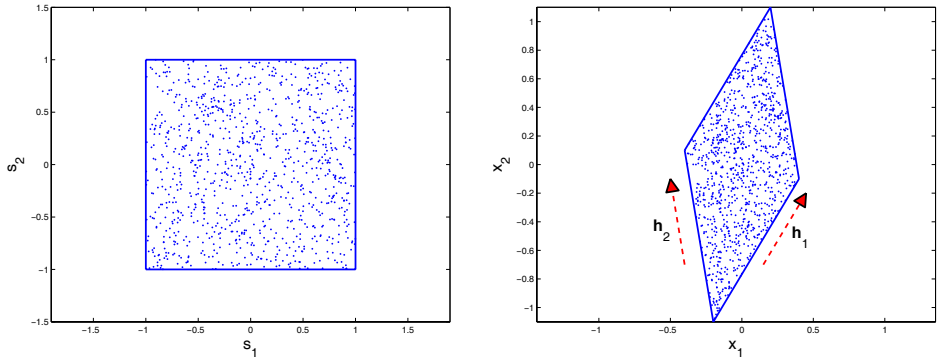
Thus, the estimated sources will be equal to the true ones except for the usual unspecified scale and order.

Note that the method works even if the source pdf is semi-bounded, for example, bounded only from below. In that case the parallelogram is open-ended but the visible corner is sufficient for identifying the two slopes. The main drawbacks of this approach are two: (a) it cannot generalize to more sources or observations and (b) it will not work if the source pdf is not bounded (for example, Gaussian, Laplace, etc).

The method of Mansour e.a. Another simple procedure for the solution of the 2×2 instantaneous BSS problem has been proposed by Mansour e.a. Mansour et al. (2001). The transformation $\mathbf{s} \mapsto \mathbf{x}$ described by (1.69) represents a skew, rotation, and scaling of the original axes in 2 dimensions. The first step of the procedure is to remove the skew by *prewhitening* $\mathbf{x}$ using the covariance matrix $\mathbf{R}_x = E\{\mathbf{x}(k)\mathbf{x}(k)^T\}$. If $\mathbf{R}_x = \mathbf{L}_x\mathbf{L}_x^T$ is the Cholesky factorization of $\mathbf{R}_x$, let

$$\mathbf{z}(k) = \mathbf{L}_x^{-1}\mathbf{x}(k) \tag{1.73}$$

The mapping $\mathbf{x} \mapsto \mathbf{z}$ is called *prewhitening transformation* because the output vector $\mathbf{z}(k)$ is white: $\mathbf{R}_z = \{\mathbf{z}(k)\mathbf{z}(k)^T\} = \mathbf{L}_x^{-1}\mathbf{R}_x\mathbf{L}_x^{-T} = \mathbf{I}$. The prewhitening transformation (1.73) makes the axes become orthogonal again, but the rotation and the scaling remains. The next step is to compensate for the rotation by computing the angle θ of the furthermost point of the constellation of $\mathbf{z}$ from the origin. We consider two cases:



(a) (b)

Figure 1.10: (a) Source constellation for two independent sources uniformly distributed between -1 and 1. (b) Output constellation after a 2×2 linear, memoryless transformation of the sources in (a).

- The sources are uniformly distributed, say, between -1 and 1 (Figure 1.11a). The source constellation is a square and the angle θ corresponds to a corner of the square. Therefore, in order to compensate for θ, the corner should return to its original position at $\pi/4$. This is achieved by the following orthogonal transformation:

$$\mathbf{y}(k) = \begin{bmatrix} \cos(\pi/4 - \theta) & -\sin(\pi/4 - \theta) \\ \sin(\pi/4 - \theta) & \cos(\pi/4 - \theta) \end{bmatrix} \mathbf{z}(k) \tag{1.74}$$

- The sources are super-Gaussian, i.e. $\text{kurt}(s_i) = Es_i^4 - 3(Es_i^2)^2 > 0$, $i = 1, 2$ (Figure 1.11b). The constellation of $\mathbf{s}$ in this case is "pointy" along the directions $[\pm 1, 0]$ and $[0, \pm 1]$. The angle θ corresponds to one of the "hands" of the X-shaped constellation for $\mathbf{x}$. Clearly, θ should be reduced to 0. This is done by the following rotation transformation:

$$\mathbf{y}(k) = \begin{bmatrix} \cos(-\theta) & -\sin(-\theta) \\ \sin(-\theta) & \cos(-\theta) \end{bmatrix} \mathbf{z}(k) \tag{1.75}$$

In both cases there remains an unknown scaling of the sources which cannot be removed since it is unobservable in all BSS problems.

1.3.2 Sparse sources, 2 mixtures

Another special case of continuous sources that can be successfully treated using geometric methods is the case of sparse sources. A signal $s_i(k)$ is *sparse* if it is equal to zero most of the time. The sparseness of the s_i is measured by the "sparseness probability"

$$p_S(s_i) = Pr\{s_i(k) = 0\}$$

Values of p_S closer to 1 correspond to more sparse data, whereas values closer to 0 represent dense data. Consider now the typical instantaneous mixing model:

$$\mathbf{x}(k) = \mathbf{H}\mathbf{s}(k) \tag{1.76}$$

assuming that all the sources are sparse. Then it is highly likely that there exist some time instances such that only one source is active at that instance. If, for example, only s_i is non-zero at time k, then $\mathbf{x}(k)$ is proportional to $\mathbf{h}_i$, the i-th column of $\mathbf{H}$. The number of outputs m is not important, as long as $m \geq 2$. In fact, the number of outputs may even be less than the number of sources ($m < n$). In the subsequent discussion we shall use the convenient value $m = 2$ because it will help us visualize the results. (Bofill and Zibulevsky, 2001) observed that the data are clustered along the directions of the mixing vectors

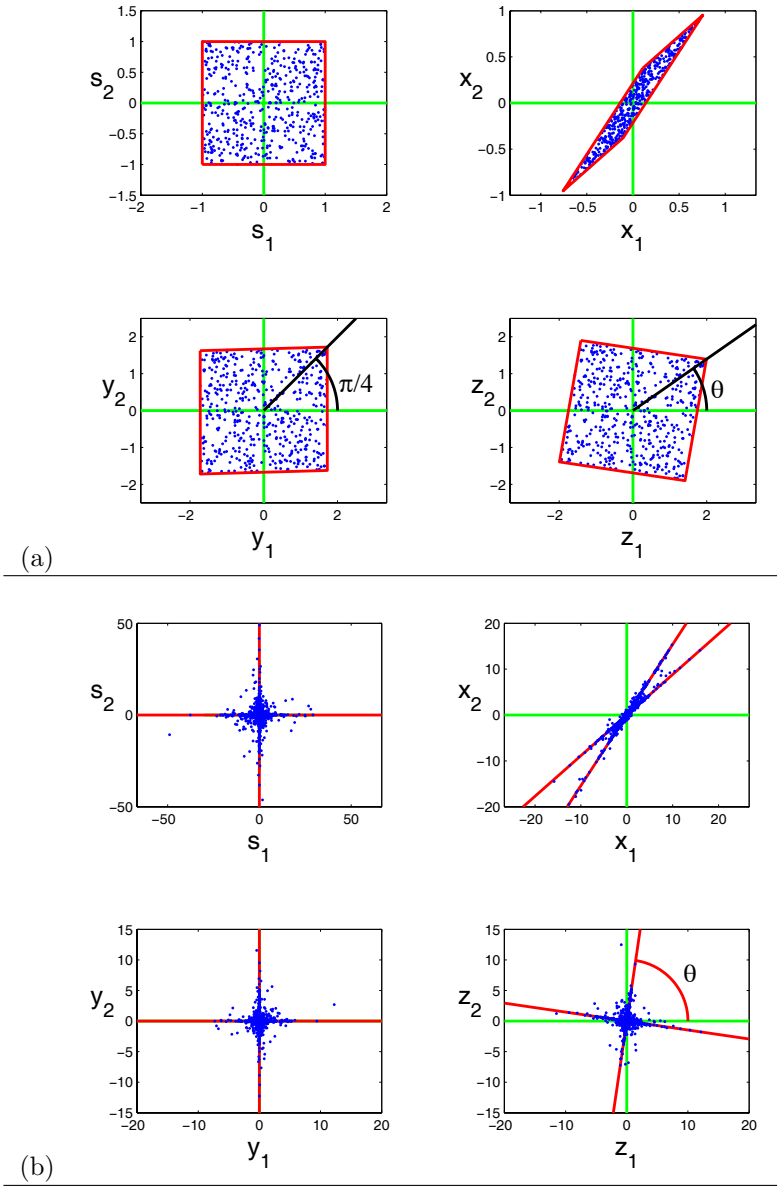


Figure 1.11: The linear, instantaneous transformation $\mathbf{s} \mapsto \mathbf{x}$ introduces skew, rotation, and scaling on the original axes. The whitening transform $\mathbf{x} \mapsto \mathbf{z}$ removes the skew, making the axes orthogonal again. Then the rotation can be removed by an orthogonal transformation $\mathbf{z} \mapsto \mathbf{y}$. (a) If the source distribution is uniform we must rotate so that θ becomes $\pi/4$. (b) If the source distribution is super-Gaussian then we must rotate so that θ becomes 0.

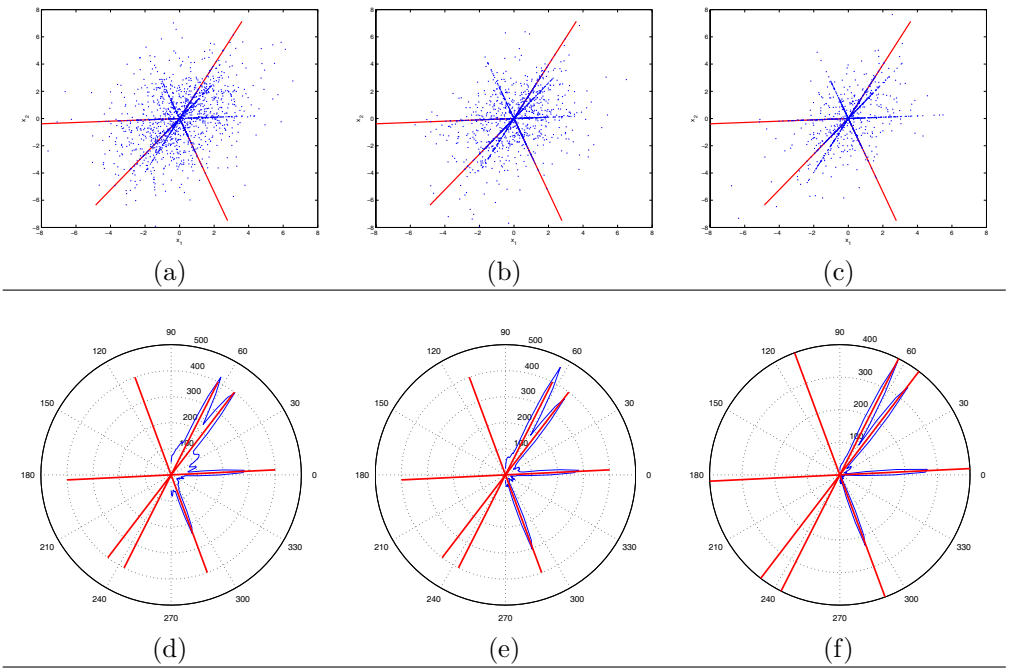


Figure 1.12: Output constellation for $m = 2$ outputs and $n = 4$ sparse sources. The top three plots correspond to different sparseness probabilities (a) $p_S = 0.6$, $p_S = 0.7$, (c) $p_S = 0.8$. The solid lines are the directions of the four vector-columns of H. The three bottom figures (d), (e), and (f) are polar plots of the data density (potential) function with spreading parameter $\sigma = 8$ corresponding to the constellations (a), (b), and (c), respectively.

$\mathbf{h}_i$, i.e. the columns of $\mathbf{H}$. Figure 1.12 shows the output constellation for the memoryless system (1.76) with $m = 2$ outputs, $n = 4$ sparse inputs, and different sparseness levels. As the sparseness of the inputs increases, the four clustering directions become more easily identifiable (see Figs. 1.12a,b,c).

Thus blind system identification is achieved by identifying the directions of maximum data density. Assuming that the sources are zero-mean, so they can take both positive and negative values, the clustering will extend to the negative directions $-\mathbf{h}_i$ as well. Since, for each i, both opposing directions $\mathbf{h}_i$ and $-\mathbf{h}_i$ are equally probable, it is not possible to identify the "true" vector. This is a manifestation of the sign ambiguity which is inherent to the BSS problem. Not surprisingly, the ordering ambiguity is also present in the sense that there is no predefined order on the directions of maximum data density.

For $m = 2$, a practical algorithm has been proposed in (Bofill and Zibulevsky, 2001). For any 2-dimensional vector $\mathbf{x} = [x_1, x_2]^T \neq 0$ let us define the angle of $\mathbf{x}$,

$$\theta(\mathbf{x}) = \arctan(x_2/x_1). \tag{1.77}$$

The directions θ where the random variable $\theta_k = \theta(\mathbf{x}(k))$ has the highest density are the directions of the mixing vectors. The density is estimated by the use of a potential function $U(\theta)$:

$$U(\theta) = \sum_k w(k)\, t(\theta - \theta_k; \sigma) \tag{1.78}$$

$$t(\alpha;\sigma) = \begin{cases} 1 - \frac{\alpha}{\pi/(4\sigma)}, & \text{for } |\alpha| < \pi/(4\sigma); \\ 0, & \text{otherwise.} \end{cases} \tag{1.79}$$

where $w(k) = \|\mathbf{x}(k)\|$ is a weight putting more emphasis on more reliable data, t is a triangular function, and σ adjusts the angular width, i.e. the spread of each local contribution to the potential function (see Figs. 1.12d,e,f). The directions of the mixing vectors are identified as the peaks of the potential function. The number of sources need not be known in advance since it can be identified by the number of peaks.

Following the mixing matrix identification step, the sources can be estimated in a second step, using the N observation samples $\mathbf{x}_1$, ..., $\mathbf{x}(N)$. In the presence of noise the samples $\mathbf{s}(1)$, ..., $\mathbf{s}(N)$, can be estimated by solving N small minimization problems:

$$\min_{\mathbf{s}(k)} \frac{1}{\sigma^2}\|\mathbf{H}\mathbf{s}(k) - \mathbf{x}(k)\|^2 + \lambda \sum_{j=1}^{n} |s_j(k)|, \quad \text{for } k = 1, \cdots, N. \tag{1.80}$$

The first term minimizes the square error (σ is the noise variance), while the second term is a penalty for non-sparsity. In the absence of noise, the optimization problem is formulated in a slightly different fashion:

$$\min_{\mathbf{s}(k)} \sum_{j=1}^{n} |s_j(k)|, \quad \text{subject to } \mathbf{x}(k) = \mathbf{H}\mathbf{s}(k). \tag{1.81}$$

Example 6 (Separation of full ensemble music) *Bofill and Zibulevsky report a number of experiments with real data including mixtures of speech (four voices), music (five songs), single musical tones (six flutes), and simple melodies from a single musical instrument (six flute melodies). The results of these experiments are published on-line (Bofill). Here we shall present the separation experiment of five songs from two mixtures. The source data were 5 seconds long excerpts from five full-ensemble music pieces extracted from standard CDs. The data were downsampled to 11,025 Hz monophonic and were preprocessed as follows:*

- *All sources were normalized to unit energy.*
- *Two mixtures were generated using a 2×5 mixing matrix $\mathbf{H}$. The five mixing vectors (the columns of $\mathbf{H}$) are formed with equally spaced angles.*
- *The mixtures were rescaled between -1 and 1 and processed in frames of length T with a hop distance d, between starting points of successive frames.*
- *Each frame was transformed with FFT of length T and only the coefficients of the positive half spectrum were kept. All FFT segments were concatenated in a single vector which is was the input to the separation algorithm.*

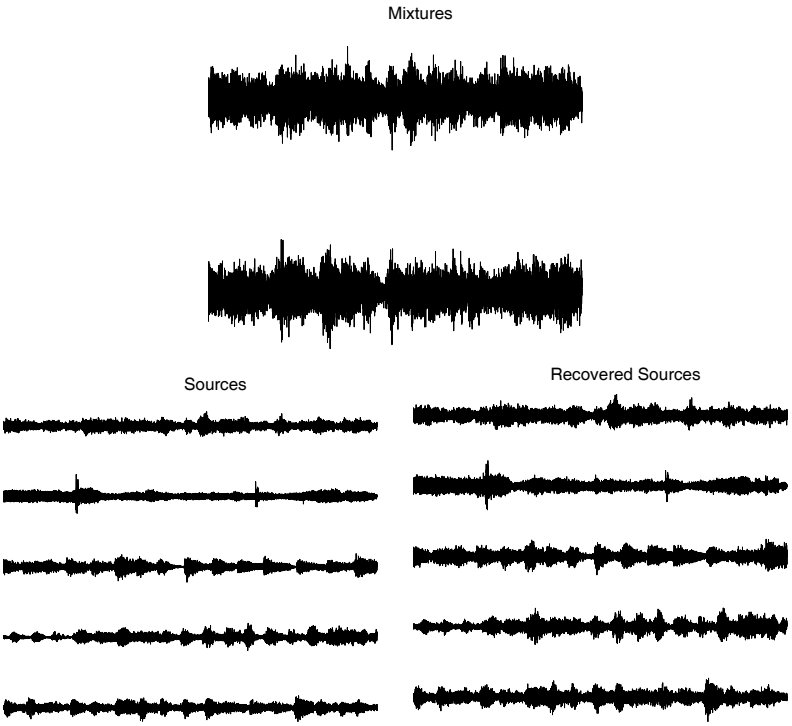


Figure 1.13: Blind separation of 5 full ensemble music pieces from 2 linear mixtures.

In this particular experiment the frame parameters were $T = 4096$ *and* $d = 1228$ *samples. The five sources, the two mixtures, and the five reconstructed signals are shown in Figure 1.13. The signal to reconstruction-error ratio, for a wide range of values of the smoothness parameter* σ*, was around 15db.* ■

1.3.3 Dense sources, 2 mixtures: Geometric ICA

The data-density concepts for sparse or bounded sources cannot be directly extended to non-sparse sources. However, Theis et al. (2003b,a); Jung et al. (2001) have developed a theory relating the data densities in the polar coordinates with the mixing vectors $\mathbf{h}_i$, $i = 1, \cdots, n$, when the sources are non-sparse, provided that their pdf is symmetric, non-Gaussian, and unimodal (i.e. has only one peak). This theory applies to memoryless systems of the type (1.76) with $m = 2$ outputs and $n \geq 2$ inputs. Extensions for $m > 3$ are possible but impractical due to the high computational cost and the extremely large required data sets.

In an analogous way to the sparse case, the method is based on the properties of the density (pdf) $\rho_{\bar{\Theta}}$ of the random variable

$$\bar{\theta} = \theta(\mathbf{x}) \mod \pi,$$

where $\theta(\mathbf{x})$ is the angle of $\mathbf{x}$ defined in (1.77). Here, however, the peaks of the density may not have a one-to-one correspondence with the mixing vectors, especially when the number of sources is greater than the number of observations ($n > m$). The basic result is that the angles

$$\theta_i = \theta(\mathbf{h}_i), \quad i = 1, \cdots, n \tag{1.82}$$

of the mixing vectors $\mathbf{h}_i$, satisfy the *Geometric Convergence Condition (GCC)* defined below:

Definition 2 (Geometric Convergence Condition) *The set of angles* $\{\theta_1, \ldots, \theta_n\}$, $\theta_i \in [0, \pi)$, *satisfies the GCC if, for each* i, θ_i *is the median of* ρ_Y *restricted to the receptive field* $\Phi(\theta_i)$. ■

Definition 3 (Receptive field) *For a set of angles* $\{\theta_1, \cdots, \theta_n\}$, $\theta_i \in [0, \pi)$, *the receptive field* $\Phi(\theta_i)$ *is the set consisting of the angles* θ *closest to* θ_i*:*

$$\Phi(\theta_i) = \{\theta \in [0, \pi) \, : \, |\theta - \theta_i| \leq |\theta - \theta_j| \text{ for all } j \neq i\}$$

■

Since the angles of the true mixing vectors satisfy the GCC we hope that we can find them by devising an algorithm which converges when the GCC is satisfied. This is exactly the aim of the *Geometric ICA Algorithm* (Theis et al., 2003b,a). This iterative algorithm works with a set of n unit-length vectors (and their opposites) and terminates only when the angles of these vectors are the medians of their corresponding receptive fields. It is conjectured that the only stable points of this algorithm are the true mixing vectors.

The algorithm starts by picking n random pairs of opposing vectors: $\{\mathbf{w}_i(0), \, \mathbf{w}'_i(0) = -\mathbf{w}_i(0)\}$, $i = 1, \cdots, n$. For each iteration k, a new observation vector $\mathbf{x}(k)$ is projected onto the unit circle

$$\mathbf{z}(k) = \frac{\mathbf{x}(k)}{\|\mathbf{x}(k)\|}$$

Then we locate the vector $\mathbf{w}_j(k)$ closest to $\mathbf{z}(k)$ and we update the pair $\mathbf{w}_j(k)$, $\mathbf{w}'_j(k)$, as follows:

$$\begin{array}{rcl} \mathbf{w}_j^{temp} & = & \mathbf{w}_j(k) + \eta(k) \frac{\mathbf{z}(k) - \mathbf{w}_j(k)}{\|\mathbf{z}(k) - \mathbf{w}_j(k)\|} \\ \mathbf{w}_j(k+1) & = & \mathbf{w}_j^{temp} / \|\mathbf{w}_j^{temp}\| \end{array} \tag{1.83}$$

$$\mathbf{w}'_j(k+1) = -\mathbf{w}_j(k+1) \tag{1.84}$$

The other $\mathbf{w}$'s are not updated in this iteration. It can be shown that the set $\mathcal{W} = \{\mathbf{w}_1(\infty), \cdots, \mathbf{w}_n(\infty)\}$ is a fixed point of this algorithm if and only if the angles $\theta(\mathbf{w}_1(\infty))$, ..., $\theta(\mathbf{w}_n(\infty))$ satisfy the GCC. We already know that the set $\mathcal{A} = \{\theta(\mathbf{h}_1), \cdots, \theta(\mathbf{h}_n)\}$ satisfies the GCC, therefore, we hope that, at convergence, $\{\theta(\mathbf{w}_1(\infty)), \cdots, \theta(\mathbf{w}_n(\infty))\} = \mathcal{A}$. If this is true then the vectors $\mathbf{w}_1(\infty)$, ..., $\mathbf{w}_n(\infty)$ are parallel to the mixing vectors $\mathbf{h}_1$, ..., $\mathbf{h}_n$, although not necessarily in that order. Since the order and scale are insignificant, this is not a problem. If $m = n$, then the estimated matrix $\hat{\mathbf{H}}^{-1} = [\mathbf{w}_1(\infty), \cdots, \mathbf{w}_n(\infty)]^{-1}$ solves the BSS problem. In the overdetermined case ($m > n$) the general algorithm for the source recovery is the maximization of $P(\mathbf{s})$ under the constraint $\mathbf{x} = \mathbf{H}\mathbf{s}$. This linear optimization problem can be approached using various methods, such as, for example, the one described in Section 1.3.2.

The FastGEO Algorithm. An alternative way to find the mixing vectors is to design a function which is zero exactly when its arguments satisfy the GCC. Then we simply have to compute the zeros of this function, for example, by exhaustive search. This approach describes the so-called *FastGEO* algorithm (Theis et al., 2003b; Jung et al., 2001). Let us separate the interval $[0, \pi)$ into n subintervals with separating boundaries ϕ_1, ..., ϕ_n, and let θ_i be the median of $\bar{\theta}$ in the subinterval $[\phi_i, \phi_{i+1}]$,

$$\theta_i = F_{\bar{\Theta}}^{-1}\left(\frac{F_{\bar{\Theta}}(\phi_i) + F_{\bar{\Theta}}(\phi_{i+1})}{2}\right), \quad i = 1, \cdots, n \tag{1.85}$$

where $F_{\bar{\Theta}}$ is the cumulative distribution function of $\bar{\theta}$, $F_{\bar{\Theta}}^{-1}$ is the inverse function of $F_{\bar{\Theta}}$ (we assume it exists) and $\phi_{n+1} = \phi_1 + \pi$ (see Figure 1.14). Then the function

$$\mu^{(n)}(\phi_1, \cdots, \phi_{n-1}) = \left[\frac{\theta_1 + \theta_2}{2} - \phi_2, \cdots, \frac{\theta_{n-1} + \theta_n}{2} - \phi_n\right]^T \tag{1.86}$$

is zero if and only if

$$\frac{\theta_i + \theta_{i+1}}{2} = \phi_{i+1}, \qquad i = 1, \cdots, n-1$$

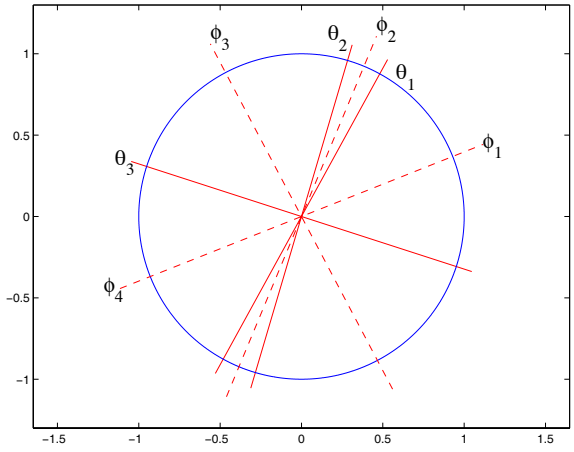


Figure 1.14: The angles θ_i of the mixing vectors $\mathbf{h}_i$ satisfy the Geometric Convergence Condition if they are the median of the random variable $\theta(\mathbf{x})$ within the interval $\Phi(\theta_i) = [\phi_i, \phi_{i+1}]$. $\Phi(\theta_i)$ is called the receptive field of θ_i and it is the set consisting of the angles θ closest to θ_i.

for all i, and so by definition the receptive field $\Phi(\theta_i)$ is exactly the subinterval $[\phi_i, \phi_{i+1}]$ and θ_i is the median of its receptive field, in other words, the set $\{\theta_i, \cdots, \theta_n\}$ satisfies the GCC. For each set of separating boundaries $\{\phi_1, \cdots, \phi_{n-1}\}$ we compute the medians θ_1, ..., θ_n by (1.85) and then the function $\mu^{(n)}(\phi_1, \cdots, \phi_{n-1})$ by (1.86). The FastGeo algorithm is the exhaustive search for the zeros of $\mu^{(n)}$.

Especially for $n = 2$ we let $\phi_1 = \phi$ and we have $\phi_2 = \phi + \pi/2$, so

$$\mu^{(2)}(\phi) = \frac{\theta_1 + \theta_2}{2} - (\phi + \pi/2).$$

Example 7 *Let x_1, x_2 be 2 instantaneous mixtures of 2 uniform sources s_1, s_2. The mixtures were generated by the following mixing operator*

$$\mathbf{H} = \begin{bmatrix} 0.0735 & 0.2913 \\ -0.3391 & 0.3725 \end{bmatrix}$$

The distribution of the angle $y = \phi(\mathbf{x})$ is shown in Fig. 1.15. The same figure shows the receptive field boundaries $\{\phi_1, \phi_2, \phi_3\} = \{77.0998, 167.0998, 257.0998\}$ (in degrees) corresponding to the angles $\{\theta_1, \theta_2\} = \{51.9759, 102.2237\}$ of the mixing vectors $\mathbf{h}_1 = [0.0735, -0.3391]^T$, $\mathbf{h}_2 = [0.2913, 0.3725]^T$. The angles θ_2 and $\theta_1 + 180$ are the medians of the angle distribution in the corresponding receptive fields. ■

1.4 Conclusions

Blind signal processing (BSP), refers to a wide variety of problems where the output of a system is observable but neither the system nor the input are known. The large family of BSP problems includes Blind Signal Separation (BSS), Blind System or Channel Identification (BSI or BCI), and Blind Deconvolution (BD). Traditional approaches exploit statistical properties of second or higher order. Recently a third approach has emerged using the geometric properties of the data cloud. This approach exploits the finite alphabet property of the input data or the shape of the constellation depending on the probability density of the sources. In such an approach our basic tools are methods for data clustering and shape description such as the convex hull. The advantage of the geometric approach is the finite nature of the methodology following the clustering step. Typically, this methodology is fast for small problem sizes, i.e. for few sources or short channels. The main disadvantage is the combinatorial explosion which

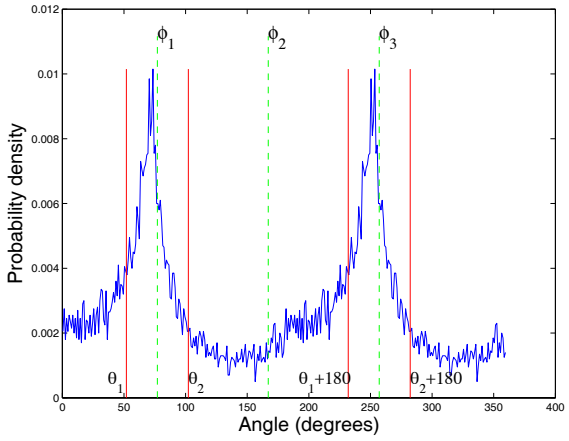


Figure 1.15: The distribution of the angle $\theta(\mathbf{x})$ for 2 instantaneous mixtures of 2 uniformly distributed sources. The receptive field boundaries are defined by the angles ϕ_i. The angles θ_i of the mixing vectors are the medians of each receptive field.

incurs when the problem size grows large. To combat this drawback channel-shortening methods may come to our assistance. The problem however is far from solved and many issues still remain open. In this chapter we presented the main geometric principles used in blind signal processing. We presented a comprehensive literature survey of geometric methods and we outlined the basic methods for blind source separation, blind deconvolution and blind channel identification.

Bibliography

K. Anand, G. Mathew, and V. U. Reddy. Blind Separation of Multiple Co-Channel BPSK Signals Arriving at an Antenna Array. *IEEE Signal Processing Letters*, 2(9):176–178, September 1995.

A. Belouchrani and J.-F. Cardoso. Maximum Likelihood Source Separation for Discrete Sources. In *Proc. EUSIPCO 94*, pages 768–771, Elsevier, Edinburgh, Scotland, September 1994.

P. Bofill. Sound examples, at http://people.ac.upc.es/pau/shpica/instant.html.

P. Bofill and M. Zibulevsky. Underdetermined blind source separation using sparse representations. *Signal Processing*, 81:2353–2362, 2001.

S. Choi, A. Cichocki, and S. Amari. Flexible Independent Component Analysis. *Journal of VLSI Signal Processing*, 26(1-2):25–38, 2000.

A. Cichocki and S. Amari. *Adaptive Blind Signal and Image Processing*. Wiley, Chichester, UK, 2002.

A. Cichocki, J. Cao, S. Amari, N. Murata, T. Takeda, and H. Endo. Enhancement and Blind Identification of Magnetoencephalographic Signals using Independent Component Analysis. In *Proceedings of the Eleventh International Conference on Biomagnetism (BIOMAG-98)*, pages 169–172, Sendai, Japan, 1999.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society Series B*, 39:1–38, 1977.

K. I. Diamantaras. Blind Separation of Multiple Binary Sources using a Single Linear Mixture. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing 2000*, volume 5, pages 2889–2892, Istanbul, Turkey, June 2000.

K. I. Diamantaras. Blind Channel Identification Based on the Geometry of the Received Signal Constellation. *IEEE Transactions on Signal Processing*, 50(5):1133–1143, May 2002.

K. I. Diamantaras and E. Chassioti. Blind Separation of N Binary Sources from one Observation: A Deterministic Approach. In *Proceedings of the Second International Workshop on Independent Component Analysis and Blind Source Separation*, pages 93–98, Helsinki, Finland, June 2000.

K. I. Diamantaras and Th. Papadimitriou. Blind Deconvolution of SISO Systems with Binary Source based on Recursive Channel Shortening. In C. G. Puntonet and A. Prieto, editors, *Proceedings of the Fifth International Symposium on Independent Component Analysis and Blind Signal Separation.*, Volume 3195 of Lecture Notes in Computer Science, pages 548–553, Springer, Granada, Spain, September 2004a.

K. I. Diamantaras and Th. Papadimitriou. MIMO Blind Deconvolution Using Subspace-based Filter Deflation. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 433–436, Montreal, May 2004b.

K. I. Diamantaras and Th. Papadimitriou. Blind Deconvolution of Multi-Input Single-Output Systems with Binary Sources. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 549–552, Piladelphia, PA, March 2005.

K. I. Diamantaras, A. P. Petropulu, and B. Chen. Blind Two-Input-Two-Output FIR Channel Identification Based on Second-Order Statistics. *IEEE Transactions on Signal Processing*, 48(2):534–542, February 2000.

S. C. Douglas and X. Sun. Convolutive Blind Separation of Speech Mixtures using the Natural Gradient. *Speech Communication*, 39:6578, 2003.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* New York, Wiley Interscience, 2nd edition, 2001.

D. N. Godard. Self-recovering Equalization and Carrier Tracking in Two-Dimensional Data Communication Systems. *IEEE Transactions on Communications*, 28(11):1867–1875, November 1980.

O. Grellier and P. Comon. Blind Separation of Discrete Sources. *IEEE Signal Processing Letters*, 5(8): 212–214, August 1998.

S. Haykin. *Unsupervised Adaptive Filtering,* Volume 1: *Blind Source Separation.* Wiley Interscience, New York, 2000a.

S. Haykin. *Unsupervised Adaptive Filtering,* Volume 2*: Blind Deconvolution.* Wiley Interscience, New York, 2000b.

A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis.* Wiley, New York, 2001.

A. Jung, F. Theis, C. G. Puntonet, and E. W. Lang. FASTGEO : A Histogram Based Approach to Linear Geometric ICA. In *Proceedings International Symposium on Independent Component Analysis and Blind Source Separation*, pages 349–354, San Diego, CA, December 2001.

T.-P. Jung, C. Humphries, T.-W. Lee, S. Makeig, M. J. McKeown, V. Iragui, and T. J. Sejnowski. Extended ICA Removes Artifacts from Electroencephalographic Recordings. In *Advances in Neural Information Systems,* 10, MIT Press, Cambridge, MA, 1998.

A. Kannan and V. U. Reddy. Maximum Likelihood Estimation of Constellation Vectors for Blind Separation of Co-Channel BPSK Signals and its Performance Analysis. *IEEE Transactions on Signal Processing*, 45(7):1736–1741, July 1997.

Y. Li, A. Cichocki, and L. Zhang. Blind Separation and Extraction of Binary Sources. *IEICE Transactions on Fundamentals*, E86-A(3):580–589, March 2003.

S. P. Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2): 129–137, March 1982.

J. MacQueen. Some Methods for Classification and Analysis of Multivariate Observations. In L. M. LeCam and J. Neyman, editors, *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297. Univ. California Press, 1967.

S. Makeig, A. J. Bell, T.-P. Jung, D. Ghahremani, and T. J. Sejnowski. Blind Separation of Auditory Evoked Potentials into Independent Components. *Proc. National Academy of Science, USA*, 94:10797–10984, 1997.

A. Mansour, C. G. Puntonet, and N. Ohnishi. A Simple ICA Algorithm Based on Geometrical Approach. In *International Symposium on Signal Processing and its Applications*, pages 9–12, Kuala Lumpur, Malaysia, August 2001.

T. M. Martinetz, S. G. Berkovich, and K. J. Schulten. "Neural-Gas" Network for Vector Quantization and its Application to Time-Series Prediction. *IEEE Transactions on Neural Networks*, 4(4):558–569, July 1993.

M. J. McKeown, S. Makeig, G. G. Brown, T.-P. Jung, S. S. Kinderman, A. J. Bell, and T. J. Sejnowski. Analysis of fMRI Data by Blind Spearation into Independent Spatial Components. *Human Brain Mapping*, 6:160–188, 1998.

J. Moody and C. J. Darken. Fast Learning in Networks of Locally Tuned Processing Units. *Neural Computation*, 1(2):281–284, 1989.

P. Pajunen. Blind Separation of Binary Sources with Less Sensors than Sources. In *Proceedings of the 1997 International Conference on Neural Networks*, pages 1994–1997, Houston, TX, June 1997.

C. B. Papadias and A. Paulraj. A Constant Modulus Algorithm for Multi-User Signal Separation in Presence of Delay Spread using Antenna Arrays. *IEEE Signal Processing Letters*, 4(6):178–181, June 1997.

L. Parra and C. Spence. Convolutive Blind Source Separation of Nonstationary Sources. *IEEE Trans. Speech Audio Processing*, pages 320–327, May 2000.

L. C. Parra and C. V. Alvino. Geometric Source Separation: Merging Convolutive Source Separation With Geometric Beamforming. *IEEE Transactions on Speech and Audio Processing*, 10(6):352–362, September 2002.

A. Paulraj and C. Papadias. Space-Time Processing for Wireless Communications. *IEEE Signal Processing Magazine*, 14(6):49–83, November 1997.

F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer, New York, 1985.

C. Puntonet, A. Mansour, and C. Jutten. A geometrical algorithm for blind separation of sources. In *Actes du Quinzième Colloque GRETSI*, pages 273–276, Juan-Les-Pins, France, September 1995.

O. Shalvi and E. Weinstein. New Criteria for Blind Deconvolution of Nonminimum Phase Systems (Channels). *IEEE Transactions on Information Theory*, 36:312–321, March 1990.

S. Shamsunder and G. Giannakis. Multichannel Blind Signal Separation and Reconstruction. *IEEE Transactions on Speech Audio Processing*, 5:515–528, November 1997.

S. Talwar, M. Viberg, and A. Paulraj. Blind Estimation of Multiple Co-Channel Digital Signals Using an Antenna Array. *IEEE Signal Processing Letters*, 1(2):29–31, February 1994.

S. Talwar, M. Viberg, and A. Paulraj. Blind Separation of Synchronous Co-Channel Digital Signals Using an Antenna Array–Part I: Algorithms. *IEEE Transactions on Signal Processing*, 44(5):1184–1197, May 1996.

F. Theis, C. G. Puntonet, and E. W. Lang. A Histogram-Based Overcomplete ICA Algorithm. In *Proceedings of the Fourth International Symposium on Independent Component Analysis and Blind Source Separation*, pages 1071–1076, Nara, Japan, April 2003a.

F. J. Theis, A. Jung, C. G. Puntonet, and E. W. Lang. Linear Geometric ICA: Fundamentals and Algorithms. *Neural Computation*, 15(2):419–439, 2003b.

S. Theodoridis and K. Koutroubas. *Pattern Recognition*. Academic Press, London, 1998.

L. Tong, G. Xu, and T. Kailath. Blind Identification and Equalization Based on Second-Order Statistics: A Time Domain Approach. *IEEE Transactions on Information Theory*, 40(2):340–349, 1994.

M. Torlak and G. Xu. Blind Multiuser Channel Estimation in Asynchronous CDMA Systems. *IEEE Transactions on Signal Processing*, 45(1):137–147, January 1997.

J. R. Treichler and M. G. Agee. A new approach to multipath correction of constant modulus signals. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 31(2):459–472, April 1983.

M. K. Tsatsanis and G. B. Giannakis. Transmitter Induced Cyclostationarity for Blind Channel Equalization. *IEEE Transactions on Signal Processing*, 45(7):1785–1794, July 1997.

A.-J. van der Veen. Analytical Method for Blind Binary Signal Separation. *IEEE Transactions on Signal Processing*, 45(4):1078–1082, April 1997.

R. Vigário, J. Särelä, V. Jousmäki, M. Hämäläinen, and E. Oja. Independent Component Approach to the Analysis of EEG and MEG Recordings. *IEEE Transactions on Biomedical Engineering*, 47(5): 589–593, 2000.

D. Yellin and B. Porat. Blind Identification of FIR Systems Excited by Discrete Alphabet Inputs. *IEEE Transactions on Signal Processing*, 41(3):1331–1339, March 1993.

D. Yellin and E. Weinstein. Multi-Channel Signal Separation: Methods and Analysis. *IEEE Transactions on Signal Processing*, 44(1):106–118, January 1996.