Estimation of Sparse Nonnegative Sources from Noisy Overcomplete Mixtures Using MAP

Cesar F. Caiafa
ccaiafa@brain.riken.jp
LABSP, RIKEN Brain Science Institute, Wako, Saitama 351-0198, Japan, and Engineering Faculty, University of Buenos Aires, Buenos Aires, C1063ACV, Argentina

Andrzej Cichocki
cia@brain.riken.jp
LABSP, RIKEN Brain Science Institute, Wako, Saitama 351-0198, Japan, and Warsaw University of Technology, Warsaw 00-661, Poland, and Systems Research Institute, Polish Academy of Sciences, Warsaw 01-447, Poland

In this letter, we propose a new algorithm for estimating sparse nonnegative sources from a set of noisy linear mixtures. In particular, we consider difficult situations with high noise levels and more sources than sensors (underdetermined case). We show that when sources are very sparse in time and overlapped at some locations, they can be recovered even with very low signal-to-noise ratio, and by using many fewer sensors than sources. A theoretical analysis based on Bayesian estimation tools is included showing strong connections with algorithms in related areas of research such as ICA, NMF, FOCUSS, and sparse representation of data with overcomplete dictionaries. Our algorithm uses a Bayesian approach by modeling sparse signals through mixed-state random variables. This new model for priors imposes \( \ell_0 \) norm-based sparsity. We start our analysis for the case of nonoverlapped sources (1-sparse), which allows us to simplify the search of the posterior maximum avoiding a combinatorial search. General algorithms for overlapped cases, such as 2-sparse and \( k \)-sparse sources, are derived by using the algorithm for 1-sparse signals recursively. Additionally, a combination of our MAP algorithm with the NN-KSVD algorithm is proposed for estimating the mixing matrix and the sources simultaneously in a real blind fashion. A complete set of simulation results is included showing the performance of our algorithm.

1 Introduction

We approach a blind source separation (BSS) problem consisting of the recovery of sparse nonnegative signals (sources) from a set of noisy linear
mixtures. To be more precise, we mathematically formulate the problem by the following equation,

$$y_t = Ax_t + n_t,$$ (1.1)

where $t = 1, 2, \ldots, T$, is the sample index (which will be avoided throughout the letter to simplify notation), $y_t \in \mathbb{R}^I$ is the observation vector (mixtures), $x_t \in \mathbb{R}^J$ is the source vector, $A \in \mathbb{R}^{I \times J}$ is the mixing matrix, and $n_t \in \mathbb{R}^I$ is the noise vector. Note that the variable $t$ usually represents time, but it can also refer to frequency, pixel position, or index number of a coefficient in a transform domain such as Fourier, spline, or wavelet domain. We will refer to sources as functions of index $t$ in either of the following ways $x_i(t) = x_{it}$.

In this letter, we are faced with the problem of estimating sources $x_t$ and the mixing matrix $A$ by using only observations $y_t$ in addition to some statistical assumptions (priors) on $A$ and $x_t$. As an example, our method is suitable for neuroscience applications where sources are associated with neuron outputs and mixtures correspond to multisite electrode measurements at different locations (Kaneko, Suzuki, Okada, & Akamatsu, 1996). In this case, $A$ is a positive matrix ($a_{ij} \geq 0$) representing the attenuation paths from different neurons to sensor locations. Neuronal source signals consist of short electrical pulses—so-called action potentials or spikes—which usually are well separated in time; in other words, we say that a particular source signal $x_i(t)$ is sparse in time since it is composed mainly of zeroes (Gerstner & Kistler, 2001). Additionally, at some index $t$, we have a small probability of having more than one signal in an active state (nonzero value), in which case we say that those signals overlap in time.

There has been increased interest in the development of new techniques for signal processing based on sparsity assumptions since many ill-posed problems become resolvable and meaningful when sparsity is taken into account. There are many applications of sparsity models in signal processing; in particular, our model becomes attractive in neuroscience applications because of recent evidence of sparse representations in the brain (DeWeese, Wehr, & Zador, 2003; Vinje & Gallant, 2000). Sparse codes transmit information with minimal redundancy and relatively few spikes. Consequently, both are informationally and metabolically more efficient than dense codes (Laughlin, de Ruyter van Steveninck, & Anderson, 1998; Levy & Baxter, 1996; Vinje & Gallant, 2000).

Additionally, we are interested in providing a solution even in difficult situations where the level of noise is high (SNR $\leq 10$ dB for example), with more sources than sensors (usually known as the overcomplete or underdetermined case with $I < J$) and for the case where the mixing matrix $A$ is unknown (double-blind scenario). We also assume white gaussian noise.
and independence between sources and noise (i.e., \( p(x, n) = p(x)p(n) \)) and between sources and a mixing matrix (i.e., \( p(x, A) = p(x)p(A) \)).

As we show in this work, the recovery of sources under these difficult conditions is achieved when very sparse signals are considered. More specifically, we model the sparsity of a signal by defining a very low probability of being in an (active) nonzero state at a given index \( t \).

It is interesting to note that our blind source separation problem (see equation 1.1) is closely related to previous research in other areas such as independent component analysis, nonnegative matrix factorization, sparse coding, overcomplete dictionaries for signal representation, and others. Following are some of the most important previous results connected to our problem:

**Independent component analysis (ICA).** In a blind source separation context, when sources are assumed to be independent and if the number of sources is equal to or less than the number of sensors, then they can be blindly separated by exploiting their independence, referred to as ICA (Cichocki & Amari, 2002; Hyvärinen, Karhunen, & Oja, 2001). Existing ICA algorithms are not well suited for the problem considered in this work for three reasons: (1) most ICA algorithms are very sensitive to noise, (2) the underdetermined case (more sources than sensors) is quite difficult to solve by most ICA algorithms, and (3) the dependence of sources significantly affects their performance (Caiafa and Proto, 2006). Some efforts were made to adapt ICA to each of these difficult situations: noisy ICA was approached in Cao, Murata, Amari, Cichocki, and Takeda (2003), Davies (2004), and DeLathauwer, DeMoor, and Vandewalle (1996); algorithms for the underdetermined case were proposed in Bofill and Zibulevsky (2000) and Li, Amari, Cichocki, Ho, and Xie (2006) to take advantage of signal sparsity leading to sparse component analysis (SCA); and the case of dependent sources was approached in Barros (2000), and Caiafa and Proto (2006).

**Nonnegative matrix factorization (NMF).** The idea of NMF is to obtain a decomposition of a given matrix \( Y \in \mathbb{R}^{I \times T} \) as a product of two nonnegative matrices \( A \in \mathbb{R}^{I \times J} \) and \( X \in \mathbb{R}^{J \times T} \). Given that perfect factorization is not always possible, there is an error in the approximation arriving at the following equation,

\[
Y = AX + N, \tag{1.2}
\]

which corresponds to equation 1.1 if vectors \( y_t, x_t, \) and \( n_t \) are arranged as columns of matrices \( Y, X, \) and \( N \) respectively. NMF was introduced by Paatero and Trapper (Tapper, 1994) and investigated by many other researchers (Hoyer, 2004; Lee & Seung, 1999, 2000); for a recent review of NMF algorithms, see (Cichocki, Zdunek, & Amari, 2008). Current NMF algorithms are not well suited for our problem because most of them do
not work for the underdetermined case. Additionally, NMF was not designed for removing noise; instead it is intended to minimize the difference between \( Y \) and \( AX \) by using a specific divergence measure and imposing nonnegativity and sparsity constraints (Cichocki, Phan, & Caiafa, 2008).

**FOCUSS (FOCal Underdetermined System Solver).** This algorithm was originally designed to obtain a sparse solution of the noiseless underdetermined (overcomplete) problem, which is nontrivial:

\[
y = Ax,
\]

where \( y \in \mathbb{R}^I \) is the available data, \( x \in \mathbb{R}^J \) is the unknown, and matrix \( A \in \mathbb{R}^{I \times J} \) is known (Gorodnitsky, George, & Rao, 1995). Note that equation 1.3 corresponds to the noiseless case of equation 1.1. In FOCUSS, the sparsity of the solution is imposed by using regularization based on the \( \ell_p \) norm of vector \( x \) with \( 0 < p \leq 1 \). Some research was also done to understand the behavior of FOCUSS in noisy environments (Rao & Kreutz-Delgado, 1998). Also, an extension to the nonnegative decomposition, FOCUSS+, was proposed by Murray and Kreutz-Delgado (2004) by setting to zero all negative elements after each iteration of the algorithm. A well-known problem of the FOCUSS algorithm is that when \( p < 1 \), which corresponds to a generalized gaussian prior distribution (Wipf & Rao, 2007), the objective function tends to be multimodal, and the algorithm often gets trapped in suboptimal solutions (local extrema). For this reason, many authors adopted \( p = 1 \), for which global maximum is guaranteed but solutions are not very sparse, corresponding to a Laplacian (or double exponential) prior distribution (Seeger, 2008; Wipf & Rao, 2007).

**Sparse representations and overcomplete dictionaries.** The general idea of a sparse representation of data is that by using an overcomplete dictionary matrix \( A \in \mathbb{R}^{I \times J} \) that contains \( J \) prototype signals and atoms in its columns \( (J > I) \), a signal \( y \in \mathbb{R}^I \) can be well represented as a sparse linear combination of these atoms as in equation 1.3, where the vector \( x \in \mathbb{R}^J \) contains the representation coefficients (Donoho, Elad, & Temlyakov, 2006; Kreutz-Delgado et al., 2003; Lewicki & Sejnowski, 2000). It is known that overcomplete representations allow sparse representations and can have greater flexibility in matching structure in the data (Kreutz-Delgado et al., 2003; Lewicki & Sejnowski, 2000). Overcomplete codes have also been proposed as a model of some of the response properties of neurons in primary visual cortex (Vinje & Gallant, 2000). Two main tasks are associated with sparse representations: sparse coding and dictionary design. Sparse coding is the process of computing the sparse representation coefficients \( x \) based on the given signal \( y \) and a known dictionary \( A \). It is usually done by a pursuit algorithm that typically selects the dictionary atoms sequentially,
(Mallat & Zhang, 1993; Tropp, 2004). In this context, sparsity is defined as the number of coefficients used in the representation—the $\ell_0$ norm $\|x\|_0$. An interesting and useful result is that under specific conditions, the minimum $\ell_0$ norm solution is obtained by minimizing the $\ell_1$ norm $\|x\|_1$ whose implementation is much simpler by means of linear programming (LP) techniques (Donoho, 2006; Donoho, Elad, & Temlyakov, 2006). Dictionary design consists of training a dictionary based on a set of examples (He, Xie, Zhang, & Cichocki, 2008; Lewicki & Sejnowski, 2000). This is obtained, for example, by the K-SVD algorithm (Aharon, Elad, & Bruckstein, 2006). The problem of sparse coding over noisy overcomplete channels was also recently approached in Doi, Balcan, and Lewicki (2007) and Donoho et al. (2006) where additive noise is considered as in our equation 1.1.

This letter is organized as follows. In section 2, the classical Bayesian approach is presented, and a discussion of several choices of prior distributions is included, showing that most existing algorithms can be interpreted as Bayesian solutions for different underlying probabilistic models. In section 3, a new model for sparse signals is introduced by using the concept of mixed-state random variables, which allows us to better represent sparse signal; in addition, models for independent, dependent nonoverlapped (1-sparse), and $k$-sparse nonnegative sources are derived. In section 4, a new algorithm for estimating sources for $A$ fixed using maximum a posteriori (MAP) is proposed based on our mixed-state model for the prior distributions. In section 5, a complete approach to our BSS problem is proposed by using a combination of our MAP algorithm with the NN-KSVD algorithm for estimating mixing matrix and sources iteratively. In section 6, simple techniques for estimating model parameters are derived. In section 7, a complete set of experimental results on simulated data is presented showing that our technique successfully solves the BSS problem under extreme conditions of noise for the underdetermined case. Finally, in section 8, the main conclusions are outlined.

## 2 Bayesian Estimation of Sources with Matrix $A$ Fixed

As is well known, the natural way to introduce a priori knowledge about sources in a statistical framework is to use a Bayesian approach (Ichir & Mohammad-Djafari, 2006; Lewicki & Sejnowski, 2000; Mohammad-Djafari, 1999, 2006; Moussaoui, Brie, Mohammad-Djafari, & Carteret, 2006; Seeger, 2008; Wipf & Nagarajan, 2008). We write the posterior distribution of sources $x$ for $A$ known as follows:

$$p(x | y) = p(x) \frac{p(y | x)}{p(y)},$$

where $p(x)$ is the prior distribution, $p(y | x)$ is the likelihood, and $p(y)$ is the evidence (Robert, 1994). Any a priori information that we may have
available about sources, like nonnegativity or sparsity, can be introduced in this model through the right selection of the prior distribution \( p(x) \). Maximum a posteriori (MAP) estimation consists of finding the maximum of equation 2.1 over \( x \), which is equivalent to

\[
\hat{x} = \arg \max_x (\log p(y | x) + \log p(x)). \tag{2.2}
\]

It is interesting to note that some existing algorithms can be obtained as MAP estimators with specific assumptions on the a priori information about sources and noise. For example, it is well known that classical Lee-Seung NMF algorithms (Lee & Seung, 1999, 2000) can be obtained as ML estimators when noise is modeled with gaussian or Poisson random variables, and ML estimation is a particular case of MAP where the prior distribution \( p(x) \) is flat. Recently it was shown that the NMF algorithm based on Itakura-Saito (IS) divergence can be obtained as an ML estimation of the sum of gaussian components or as an ML estimation in the gamma multiplicative noise case (Févotte, Bertin, & Durrieu, 2008).

Since we are considering gaussian white noise, we have

\[
\log p(y | x) \propto -\frac{1}{2\sigma_n^2} \sum_t \| y_t - Ax_t \|_2^2. \tag{2.3}
\]

From equations 2.2 and 2.3, we obtain the general objective function to be maximized as follows:

\[
J(x) = -\frac{1}{2\sigma_n^2} \sum_t \| y_t - Ax_t \|_2^2 + \sum_t \log [p(x_t)], \tag{2.4}
\]

where \( \sigma_n^2 \) is the variance of noise.

By assuming different a priori distributions of sources, we arrive at diverse objective functions:

- **Positive gaussian distribution**: If sources are modeled by independent positive gaussian variables, the resulting objective function is

\[
J(x) = -\frac{1}{2\sigma_n^2} \sum_t \| y_t - Ax_t \|_2^2 - \frac{1}{2\sigma_x^2} \sum_t \| x_t \|_2^2, \text{ with } x_t \geq 0, \tag{2.5}
\]

where \( \sigma_x \) is related to the variance of sources. Equation 2.5 is exactly the same objective function as proposed in the regularized Lee-Seung NMF algorithm for smooth solutions by using \( \ell_2 \) norm as a regularization term (Cichocki, Zdunek, & Amari, 2006). Using gaussian priors (positive or standard distribution) allows a simple analytical
treatment, but such a prior does not encode sparsity, leading to a uniform shrinkage of recovered sources (Seeger, 2008).

- **Gamma distribution:** If sources are modeled by independent gamma variables as in Mohammad-Djafari (2006); Moussaoui, Mohammad-Djafari, Brie, and Caspary (2004); and Moussaoui et al. (2006), that is,
  \[ p(x) = \frac{b^a}{\Gamma(a)} x^{a-1} \exp(-bx) \text{ for } x \geq 0, \]
  then the resulting objective function is
  \[ J(x) = -\frac{1}{2\sigma_n^2} \sum_t \|y_t - Ax_t\|_2^2 + (a - 1) \sum_t g(x_t) - b \sum_t \|x_t\|_1, \]
  with \( x_t \geq 0 \), where \( g(x) = \log[\prod_{i=1}^I x_i] \). Gamma priors have been applied to spectrometry since this distribution takes into account non-negativity and its parameters allow a better fit to the spectra distribution (Mohammad-Djafri, 2006; Moussaoui et al., 2004, 2006).

- **Exponential distribution:** As a particular case of gamma distribution for \( a = 1 \), we obtain the exponential distribution, and the resulting objective function is
  \[ J(x) = -\frac{1}{2\sigma_n^2} \sum_t \|y_t - Ax_t\|_2^2 - b \sum_t \|x_t\|_1, \text{ with } x_t \geq 0, \]
  which is the same function used in the regularized Lee-Seung NMF algorithm for sparse solutions by adopting \( \ell_1 \) norm as a regularization term (Cichocki et al., 2006). In fact, the exponential distribution is the positive version of the Laplace distribution, which is known to be a good option for imposing sparsity on the solutions. The exponential distribution puts much more weight close to zero than the gaussian, while still having higher probabilities for large values, that is, tails are also comparatively heavy, an essential feature of a sparsity prior (Seeger, 2008). Nevertheless, some applications require stronger sparsity priors, which can be imposed by generalized gaussian distribution, as explained below.

- **Positive generalized gaussian:** When sources are modeled by independent positive generalized gaussian variables, we obtain the same model as used in FOCUSS+ (Murray & Kreutz-Delgado, 2004), that is,
  \[ p(x) = c \exp\left(-\frac{1}{2} \left(\frac{x}{\sigma_x}\right)^p\right), \text{ for } x \geq 0, \]
where \( p \) and \( \sigma_x \) are parameters and \( c \) is a normalization constant. This distribution contains as particular cases the gaussian distribution for \( p = 2 \) and the exponential distribution for \( p = 1 \). Generalized Gaussian distribution was also used in blind source separation in Kim and Yoo (2006) and Mohammad-Djafari (2006). Following the Bayesian approach we can easily obtain the objective function:

\[
J(x) = -\frac{1}{2\sigma^2} \sum_t \| y_t - Ax_t \|^2_2 - \frac{1}{2\sigma_x^p} \sum_t \| x_t \|^p_p, \text{ with } x_t \geq 0.
\]  

(2.10)

Note that this model is the same as that used in the regularized Lee-Seung NMF algorithm (Cichocki et al., 2006) where the parameter \( p \) controls the sparsity (\( 0 < p \leq 1 \)) or smoothness (\( p = 2 \)) of recovered sources. This approach arrives also at the same objective function used in FOCUSS, where sparsity is imposed by regularization based on the \( \ell_p \) norm with \( 0 < p \leq 1 \) (Gorodnitsky et al., 1995). Nevertheless, it is well known that only for \( p = 1 \) does the objective function have a unique optimum solution, and for \( 0 < p \leq 1 \), do the objective function has many suboptimal local solutions and it is difficult to find the global solution. This situation is even worst for \( p \to 0 \) (Seeger, 2008).

It is important to note that prior distributions are based on continu- ous distributions that can be sparse in the sense that they concentrate their weights around zero. In this work, we propose a new model for prior distribu-tions (sources) that are a mixture of discrete and continuous distributions. Our model is based on the basic definition of sparsity, assigning high probability to the zero state and low probabilities to nonzero states. A distinctive feature of our models is that values close to zero also have low probability. As our experimental results show, this mixed-state model represents a great advantage for noise-reduction purposes. It is not a surprise that this mixed-state prior distribution, together with the Bayesian approach, leads to an objective function having a regularization term based on the \( \ell_0 \) norm \( \| x \|_0 \) (see section 3) for which an iterative maximization based on derivatives is not applicable. We also note that our method provides a solution to the intractable limiting case \( p \to 0 \) of FOCUSS+ (Murray & Kreutz-Delgado, 2004) (positive generalized gaussian priors), which becomes tractable in the limit of very sparse signals.

Table 1 provides a summary of choices for prior distributions together with their associated norms and related algorithms. The last row of Table 1 includes our probabilistic model (uniformly distributed mixed state), which we develop in the section 3.

3 Probabilistic Model for Nonnegative Sparse Sources

The common idea of sparsity is that many signals found in real-world applications or their transforms in some specific domain have only a few
Table 1: Relationship Among Prior Distributions, Norms, and Algorithms.

<table>
<thead>
<tr>
<th>Prior Distribution</th>
<th>Associated Norm</th>
<th>Main Related Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. Positive generalized gaussian</td>
<td>$\ell_p$</td>
<td>Regularized NMF ALS: (Cichocki et al. 2006)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FOCUSS(+) - Gorodnitsky et al. (1995); Rao &amp; Kreutz-Delgado (1998)</td>
</tr>
<tr>
<td>Ia: Exponential ($p = 1$)</td>
<td>$\ell_1$</td>
<td>Regularized NMF ALS sparse solutions: Cichocki et al. (2006); Basis pursuit (BP), linear</td>
</tr>
<tr>
<td></td>
<td></td>
<td>programming (LP): Doi et al. (2007); Donoho &amp; Elad (2003); Donoho (2006)</td>
</tr>
<tr>
<td>Ib: Positive gaussian ($p = 2$)</td>
<td>$\ell_2$</td>
<td>Regularized NMF ALS smooth solutions: Cichocki et al. (2006)</td>
</tr>
<tr>
<td>II. Gamma</td>
<td>$\ell_1$</td>
<td>Mohammad-Djafari (1999), Moussaoui et al. (2006)</td>
</tr>
<tr>
<td>IIa: Exponential ($\alpha = 1$)</td>
<td>$\ell_1$</td>
<td>Regularized NMF ALS sparse solutions: Cichocki et al. (2006), Basis pursuit (BP), linear</td>
</tr>
<tr>
<td>III. Uniformly distributed</td>
<td>$\ell_0$</td>
<td>Mixed state nonnegative sources recovery: our approach</td>
</tr>
<tr>
<td>mixed state</td>
<td></td>
<td>Laplacian prior: Seeger (2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Very sparse sources recovery: our approach</td>
</tr>
</tbody>
</table>

nonzero entries (Chen, Donoho, & Saunders, 1998; Donoho & Elad, 2003). In previous work, the probabilistic approach to sparsity was usually based on continuous pdfs with their weights concentrated at the origin (super gaussian continuous pdfs). Instead of using continuous distributions, our probabilistic model of generalized gaussian sparsity is based on the definition of mixed-state random variables (Bouthemy, Piriou, & Yao, 2006). A mixed-state random variable has two mutually exclusive states: a zero state (inactive) and a nonzero continuously distributed state (active). Since we are interested in nonnegative random variables $x \geq 0$, we define the following probabilities:

\[
P(x = 0) = \rho, \\
P(x > 0) = 1 - \rho. \tag{3.1}
\]

We define a continuous generic distribution $f(x)$ associated with the nonzero state. To formally define the associated probability density function
of a mixed state random variable, we follow the treatment of Bouthemy et al. (2006):

\[ p(x) = \rho \delta(x) + (1 - \rho)\delta^*(x) f(x), \] (3.2)

where \( \delta(x) \) is the indicator function; that is,

\[ \delta(x) = \begin{cases} 1 & \text{if } x = 0, \\ 0 & \text{if } x \neq 0, \end{cases} \]

and \( \delta^*(x) = 1 - \delta(x) \) is its complementary function. The sample space \( E = \{0\} + (0, \infty) \) is equipped with a mixed reference measure,\(^1\)

\[ m(dx) = \delta_0(dx) + \lambda(dx), \]

where \( \delta_0 \) is the Dirac measure at 0 and \( \lambda \) is the Lebesgue measure on \((0, \infty)\). Equation 3.2 can be also written in an exponential form for \( \rho \neq 0 \) as follows:

\[ p(x) = \exp \left( \log \rho + \delta^*(x) \log \left[ \frac{1 - \rho}{\rho} f(x) \right] \right). \] (3.3)

This model allows one to fit the distribution of data by choosing among different options for the continuous positive distribution \( f(x) \) such as uniform, gaussian, generalized gaussian, mix of gaussian, and exponential, (see Figure 1). However, in this work we are focused on the uniformly distributed mixed-state variable, which corresponds to the case of \( f(x) \) being the uniform distribution in \((0, a)\) for the nonzero state (see Figure 1a). This is the simplest form of a mixed-state distribution and allows us to derive simple algorithms.

3.1 Independent Sparse Sources. Equation 3.3 determines the probability distribution associated with a single source \( x_j \), which is composed by an impulse at zero and a continuous distribution for positive values, as Figure 1 shows. In order to have a probabilistic model for the source vector \( x \), we also need to use some assumption about the statistical interactions among sources. For instance, the simplest assumption could be to consider sources mutually independent as in an ICA context, which means to consider that the joint distribution can be factorized:

\[ p(x) = \prod_{j=1}^J p(x_j). \]

\(^1\)To complete the specification of a measurable space requires defining an appropriate sigma-algebra for which \( m(dx) \) is a meaningful reference measure (see details in Bouthemy et al. (2006)).
For uniformly distributed mixed-state variables, we have \( f(x) = 1/a \), and by putting the prior, equation 3.3, into the Bayesian approach, equation 2.4, we obtain the following objective function:

\[
J(x) = -\frac{1}{2\sigma^2} \sum_{t=1}^{T} \|y_t - Ax_t\|_2^2 + \log \left( \frac{1 - \rho}{\rho a} \right) \times \sum_{t=1}^{T} \|x_t\|_0 \quad \text{with} \ 0 < x_t < a,
\]

(3.4)

which explicitly shows that the \( \ell_0 \) norm of vectors \( x_t \) needs to be minimized for estimating sources by MAP arriving at sparse solutions.\(^2\)

It is interesting to note also that when one considers other mixed-state distributions, that is, by assuming \( f(x) \) not as the uniform distribution, we arrive at different objective functions. For instance, by considering sparse sources modeled by an impulse at zero plus an exponential distribution for positive values, \( f(x_j) = b \exp(-bx_j) \), we obtain the following cost function:

\[
J(x) = -\frac{1}{2\sigma^2} \sum_{t=1}^{T} \|y_t - Ax_t\|_2^2 + \log \left( \frac{b(1 - \rho)}{\rho} \right) \times \sum_{t=1}^{T} \|x_t\|_0 - b\|x_t\|_1,
\]

(3.5)

\(^2\)Note that the term associated with the \( \ell_0 \) norm becomes negative only if \( \rho > \frac{1}{a+1} \), which means sparse sources.
where we see that the \(\ell_0\) and \(\ell_1\) norms are involved in the optimization problem.\(^3\) Note also that by depending on the parameters \(\rho\) and \(b\), the \(\ell_0\) norm term can be more or less dominant in the global optimization problem.

Similarly, if we assume sources with a positive gaussian mixed-state distribution, \(f(x_j) = \frac{2}{\sqrt{2\pi}\sigma} \exp(-\frac{x_j^2}{2\sigma^2})\) we obtain the following function:

\[
J(x) = -\frac{1}{2\sigma^2} \sum_t \|y_t - Ax_t\|^2_2 + \log \left( \frac{2(1-\rho)}{\sqrt{2\pi}\rho\sigma} \right) \sum_t \|x_t\|_0
\]

\[
- \frac{1}{2\sigma^2} \|x_t\|^2_2, \tag{3.6}
\]

which contains terms associated with the \(\ell_0\) and \(\ell_2\) norms.

Comparing equations 3.4 to 2.3, equation 3.5 to 2.8, and equation 3.6 to 2.5, we see that the effect of considering mixed-state distributions as priors is to introduce an \(\ell_0\) norm term in the objective function.

### 3.2 Dependent Nonoverlapped Sources

In this section we derive the model for the case where sources are nonoverlapped, that is, at a given index \(t\), only one source is allowed to be in an active state. As we will show, a fast and simple algorithm can be developed for this case, and, more important, it can be used for the case of overlapped sources by sequentially applying this basic algorithm.

We consider the following simple generative model for nonoverlapped sources. At each index \(t\), a random variable \(\nu_t\) determines the variable \(x_{\nu_t}\) that is allowed to be active. We choose a uniform distribution for random variable \(\nu\) in the range \(1, 2, \ldots, J\),

\[
P(\nu = j) = \frac{1}{J}. \tag{3.7}
\]

Then for that variable \(x_j\), we define a pdf according to our uniformly distributed mixed-state variable model 3.3. This model can be written in mathematical terms as follows:

\[
p(x_j | \nu = j) = \rho \delta(x_j) + (1 - \rho)\delta^*(x_j)/a \quad \text{with } x_j \in (0, a), \tag{3.8}
\]

\[
p(x_j = 0 | \nu \neq j) = 1. \tag{3.9}
\]

In order to construct the MAP estimator, we need to find the joint distribution for \(x\) whose entries \(x_j\) are not statistically independent. Note that the nonoverlapping assumption forces the entries to be dependent. It is

\(^3\)Here the term associated with the \(\ell_0\) norm becomes negative only if \(\rho > \frac{b}{1+b}\), which means sparse sources.
not difficult to obtain an explicit formula for the joint distribution as follows. The probability of having more than one variable in an active state is zero because it is a prohibited configuration. The probability of having all variables in a zero state is equal to the probability of selecting \( v = j \) as the allowed active source (see equation 3.7) multiplied by \( \rho \) (the probability of the zero state). If we take into account that there is a total number of \( J \) sources, we obtain

\[
P(x = 0) = \sum_{j=1}^{J} P(v = j)\rho = \frac{1}{J} \sum_{j=1}^{J} \rho = \rho.
\]  

(3.10)

Finally, the probability of having only the variable \( x_k \) in an active state is the probability of selecting \( v = k \) as the allowed active source, equation 3.7, multiplied by \((1 - \rho)/a\). Then, putting this all together, we write the joint distribution for \( x \) as follows:

\[
p(x) = \begin{cases} 
\rho & \text{if } x_j = 0 \ \forall \ j \ (\text{all signals are inactive}) \\
\frac{1 - \rho}{Ja} & \text{if } 0 < x_k < a \ (\text{only } x_k \text{ as active signal}), \\
0 & \text{otherwise}
\end{cases}
\]  

(3.11)

which means that only \( J + 1 \) cases are possible: all signals in inactive states or only one signal in an active state.\(^4\)

It can be seen that for a fixed coefficient \( \rho \), the nonoverlapping constraint produces different levels of sparsity in signals depending on the number of sources. In order to measure the sparsity of signals independently of the number of sources, we define the sparsity rate \( \rho^* \) as the probability that one of the \( J \) sources under consideration is in a zero state at a given index \( t \). By applying the total probability formula, we obtain that the probability of a particular variable \( x_k \) in the zero state is equal to the probability of selecting \( v = k \) as the allowed active source (see equation 3.7) multiplied by \( \rho \) plus the probability of selecting any other variable \( x_j \), which is \((J - 1)/J \) i.e. \( J - 1 \) cases):

\[
\rho^* = \frac{1}{J} \rho + \frac{(J - 1)}{J}.
\]  

(3.12)

Note that \( \frac{J-1}{J} \leq \rho^* \leq 1 \), which means that any signal \( x_k \), as a time series, is very sparse independent of parameter \( \rho \) as a consequence of the nonoverlapping assumption.

\(^4\)Additionally we require that each source at least one time is active: \( \exists \ t_i \) such that \( x_i(t_i) \neq 0 \).
3.3 Overlapped Sources as a Sum of Nonoverlapped Sources. In this section we consider the case where more than one source is allowed to be active at a given index \( t \). First, we introduce notation about sparse vectors. We say that a vector \( x \in \mathbb{R}^J \) is \( k \)-sparse if it has at most \( k \) nonzero entries. For example, for the case of nonoverlapped sources, the vector \( x_t \) is 1-sparse at every index to \( t \). We observe that any \( k \)-sparse vector can be decomposed as a sum of \( k \) 1-sparse vectors, which can be done by assigning each of the overlapped values to different vectors.

More specifically, let us consider the case for which the source vector \( x \) is 2-sparse. Then we can decompose our model of equation 1.1 as:

\[
y = Ax + n = A(x^{(1)} + x^{(2)}) + n = Ax^{(1)} + v,
\]

where \( v = Ax^{(2)} + n \) can be considered as noise (which we assume near gaussian distributed\(^5\)) in the estimation of the 1-sparse vector \( x^{(1)} \).

This simple observation suggests an algorithm for estimating 2-sparse sources by applying the algorithm for nonoverlapped (1-sparse) sources twice. More specifically, under the assumption that the matrix \( A \) is known, after the source vector \( x^{(1)} \) is obtained, we can remove it from the mixtures by doing:

\[
y \leftarrow y - Ax^{(1)},
\]

and by applying again our non overlapped source estimation algorithm, we obtain an estimate of \( x^{(2)} \). Finally we calculate the sources as

\[
x = x^{(1)} + x^{(2)}.
\]

In the next section we discuss in detail the algorithm for nonoverlapped (1-sparse) sources and 2-sparse sources, with an extension to the general case with \( k \)-sparse sources.

4 MAP Source Recovery Algorithm

4.1 Nonoverlapped Case (1-Sparse). Using previous mathematical definitions, we are now able to derive our MAP algorithm for the recovery of nonnegative, nonoverlapped sources from noisy mixtures under the assumption of \( A \) fixed (known). Note that the combinatorial search of the minimum \( \ell_0 \) norm solution is avoided by using the non overlapped assumption on sources and arriving at a very simple and fast algorithm.

\(^5\)Note that entries of vector \( v \) are calculated as the sum of a uniformly distributed variable plus gaussian noise. It is clear that the resulting distribution for \( v_i \) is more likely gaussian than the uniform distribution.
It is important to note that the objective function of equation 2.4 can be minimized in a parallel way. At every index $t$, a different maximization problem consisting of obtaining the vector $x_t$ maximizes the following function:

$$J_t(x) = -\frac{1}{2\sigma_n^2} \|y_t - Ax_t\|_2^2 + \log[p(x_t)]. \quad (4.1)$$

As at every index $t$, not more than one signal is allowed to be active; the objective function of equation 4.1 can be maximized by the parallel maximization of the $J + 1$ allowed cases according to equation 3.11 and by choosing the maximum among them. Note that this is a nontrivial change from the factorial model given in equation 3.4, as the regularization term is now nonadditive. Furthermore, even though the regularization term is nonadditive, one is able to obtain a parallel algorithm under the assumption of one-at-a-time active sources. More specifically, let us now define the partial objective function for each $J + 1$ case as $C_j$. By $C_0$, we denote the posterior log probability for the case with all sources at zero-state ($x = 0$), and by $C_j$ ($j = 1, 2, \ldots, J$) we denote the posterior log probability for cases where only source $j$ is active. Then we obtain the following set of partial objective functions (see the derivation in the appendix):

$$C_0 = -\frac{1}{2\sigma_n^2} \|y\|_2^2 + \log[\rho], \quad (4.2)$$

$$C_j(x_j) = -\frac{1}{2\sigma_n^2} \|y\|_2^2 - \alpha_j x_j - \beta_j x_j^2 + \log\left[\frac{1 - \rho}{Ja}\right], \text{ with } x_j \in (0, a), \quad (4.3)$$

where

$$\alpha_j = \frac{1}{\sigma_n^2} \sum_{i=1}^{l} y_i a_{ij} \text{ and } \beta_j = -\frac{1}{2\sigma_n^2} \sum_{i=1}^{l} a_{ij}^2. \quad (4.4)$$

Note that the first term $-\frac{1}{2\sigma_n^2} \|y\|_2^2$ in equations 4.2 and 4.3 is constant and can be avoided in the maximization process for which we define the modified partial objective functions:

$$C_0^* = \log[\rho], \quad (4.5)$$

$$C_j^*(x_j) = -\alpha_j x_j - \beta_j x_j^2 + \log\left[\frac{1 - \rho}{Ja}\right]. \quad (4.6)$$
Note that $C_0^*$ can be directly calculated from the model parameter $\rho$. On the other hand, $C_j^*(x_j)$ with $j = 1, 2, \ldots, J$ are quadratic forms over a constrained range of values $x_j \in (0, a)$ and can be easily maximized by assigning

\[
x_{j \max} = \begin{cases} 
-\frac{\alpha_j}{2\beta_j} & \text{if } 0 < -\frac{\alpha_j}{2\beta_j} < a, \\
0 & \text{if } -\frac{\alpha_j}{2\beta_j} < 0, \\
a & \text{if } -\frac{\alpha_j}{2\beta_j} > a, 
\end{cases}
\] (4.7)

as shown in Figure 2.

Note that when $\rho \ll 1$ (or $\rho \to 0$), $C_0^*$ becomes unimportant compared with $C_j^*$, which means that the algorithm will assign the activity to one of the sources ($x_j$). On the other hand, when $\rho$ is higher, for example, $\rho = 0.5$, then the algorithm will assign the activity to one of the sources or to noise depending on the comparison of values $C_0^*$ and $C_j^*$. Finally, when sources are very sparse ($\rho \to 1.0$), most of the time the algorithm will assign the activity on mixtures to pure noise. The MAP estimation algorithm is depicted in algorithm 1:
Algorithm 1: MAP Estimate of 1-Sparse Sources

- For $t = 1$ to $T$
  - Calculate $C_0^*$ (see equation 4.5)
  - for $j = 1$ to $J$
    * Calculate $C_j^*(x_{j\text{ max}})$ (see equations 4.6 and 4.7)
  - endfor
  - Choose $k = \arg \max_j (C_j^*(x_j))$
  - if $k = 0$ then $\hat{x} = 0$ (all sources in zero state)
  - else $\hat{x}_k = x_k\text{ max}$ and $\hat{x}_j = 0 \forall j \neq k$ (source $k$ in a nonzero state)
- endfor

Figure 3: Example of the 2-sparse estimation algorithm. Subsets of samples (1–1000) of only two sources (original and their estimates) are shown for a case with $I = 6$ sensors, $J = 10$ sources, SNR = 10 dB.

Algorithm 1 has complexity $O(T \times I \times J)$, i.e. it is linear in the three parameters $T, I, J$ which can be considered as very fast.

4.2 Overlapped Case (2-Sparse Sources). As we have explained, 2-sparse sources can be estimated by applying the algorithm of the previous section (1-sparse sources) twice. The idea is that our algorithm for nonoverlapped sources is robust in the sense that when it is applied to $k$-sparse sources, the algorithm recovers the sources very well at indexes $t$ for which only one signal is active, and for those indexes $t$ where more than one signal is active, it assigns the activity to only one recovered source, making the other sources inactive and absorbing the rest of the activity by the noise. Nevertheless, the value assigned to the active source is usually overestimated, as shown in Figure 3. For this reason, we have developed an iterative algorithm that successively improves the quality of the estimation:
Algorithm 2: MAP Estimate of 2-Sparse Sources

- Set $\tilde{y} = y$
- for pass = 1 to Npass
  - $x^{(1)} = \text{MAP 1-sparse} (\tilde{y}, A)$
  - Remove estimated sources $\tilde{y} = y - Ax^{(1)}$
  - $x^{(2)} = \text{MAP 1-sparse} (\tilde{y}, A)$
  - Remove estimated sources $\tilde{y} = y - Ax^{(2)}$
- endfor
- Calculate 2-sparse sources $x = x^{(1)} + x^{(2)}$

Figure 3 shows how this iterative algorithm works. Note that after the first iteration (first pass), the overlapped sources were detected, but the estimation of its values at that index $t = 812$ is not accurate. After 10 iterations, the algorithm converges to the correct estimation.

4.3 General Case ($k$-Sparse Sources). Let us consider the case where $k$ is a power of 2 number. Then we can decompose a $k$-sparse vector $x$ as the sum of two $(k/2)$-sparse vectors $x = x^{(1)} + x^{(2)}$ and we can use a recursive algorithm for estimating general $k$-sparse sources as shown in algorithm 3:

Algorithm 3: MAP Estimate of $k$-Sparse Sources

- if $k = 1$ then $x = \text{MAP 1-sparse} (\tilde{y}, A)$
- else
  * Set $\tilde{y} = y$
  * for pass = 1 to Npass
    - $x^{(1)} = \text{MAP } k/2\text{-sparse} (\tilde{y}, A)$
    - Remove estimated sources $\tilde{y} = y - Ax^{(1)}$
    - $x^{(2)} = \text{MAP } k/2\text{-sparse} (\tilde{y}, A)$
    - Remove estimated sources $\tilde{y} = y - Ax^{(2)}$
  * endfor
  * Calculate $k$-sparse sources $x = x^{(1)} + x^{(2)}$

The idea is that when our nonoverlapped algorithm is applied in the leaves of our recursive tree, it assigns the activity to only one source, and the rest of active sources are absorbed by the noise.

It is important to note that the algorithm requires setting the maximum sparsity likely to be encountered: the specification of the sparsity parameter $k$. It also works for the case when we use a parameter $k$ that is greater than the real number of overlapped samples. For example, in the case of having

Note that when $k$ is not a power of 2, we can use the higher closest power of 2 since a $k$-sparse is also $m$-sparse if $m > k$. 
1-sparse sources and using $k = 2$, we obtain at the first iteration $x^{(1)} = x$ and $x^{(2)} = 0$, which corresponds to a correct estimation $x = x^{(1)} + x^{(2)}$.

The complexity of algorithm 3 scales more than linearly with $k$. For example, applying our algorithm for $k = 2$ requires using $2 \times N_{\text{pass}}$ times the 1-sparse algorithm (see algorithm 1). As our algorithm is recursive, it can be shown that the complexity for $k$-sparse signals is $O(k \times (N_{\text{pass}})^{\log_2(k)} \times T \times I \times J)$. It is noted that in the limit $J \to \infty$ or in the limit that $\rho \to 1$, the probability of the overlapping case should become vanishingly small, which also justifies the utility of the proposed algorithm.

5 Estimating the Mixing Matrix and Sources by Combining MAP with NN-KSVD

Now we focus on a more complete approach to our BSS problem that includes not only the estimation of sources $x_t$ but also the estimation of the mixing matrix $A$ from a set of observations $y_1, y_2, \ldots, y_T$. It is important to note that this is equivalent to the problem of estimating a nonnegative dictionary that allows one to achieve a sparse representation of data from noisy examples (Donoho et al., 2006). In equation 1.1, matrix $A$ can be interpreted as a dictionary since every vector $y_t$ is expressed as a linear combination of columns of matrix $A$ (code words) plus noise. Note also that in the nonoverlapped sources case (1-sparse), every vector is approximated by only one code word multiplied by a coefficient that corresponds to the gain-shape VQ method in a context of vector quantization (VQ) (Gersho & Gray, 1991).

Let us consider the case in which we have a current estimation of sources $x$ and want to estimate matrix $A$ in a Bayesian context, that is, we want to maximize the a posteriori distribution of $A$ given $y$ for $x$ known:

$$p(A | y) = p(A) \frac{p(y | A)}{p(y)}.$$ (5.1)

If our a priori knowledge about matrix $A$ is that its values are nonnegative and uniformly distributed with unit-norm columns$^7$ ($\|a_i\| = 1$, $i = 1, 2, \ldots, J$) then the MAP estimation is reduced to the constrained ML (maximum likelihood) criterion:

$$\hat{A} = \arg \max_A \log p(y | A), \text{ with } a_{ij} \geq 0 \text{ and } \|a_i\| = 1.$$ (5.2)

Now, taking into account that noise is gaussian and white (independent and identically distributed), we can write this equation in terms of the Euclidean $\ell_2$ norm:

$$\hat{A} = \arg \max_A \left( - \sum_{t=1}^{T} \|y_t - Ax_t\|^2 \right), \text{ with } a_{ij} \geq 0 \text{ and } \|a_i\| = 1.$$ (5.3)

$^7$The unit norm constraint is usually imposed in BSS/NMF methods given that the solution exists up to a scale factor.
The problem posed in equation 5.3 is exactly the same as that considered in NN-KSVD (nonnegative K-SVD) in the context of dictionary constructions (Aharon, Elad, & Bruckstein, 2005; Aharon et al., 2006). In NN-KSVD the global objective function, equation 5.3, is maximized by sequentially maximizing local objective functions associated with one column (code word) of matrix A at a time and by considering only the signal samples \( x_t \) that actually use the code word under analysis (see Aharon et al., 2005, 2006 for the details). In summary, NN-KSVD consists of an iteration of two main tasks: the sparse coding stage, which selects the best coding (sources) given a known dictionary A and the code book update stage, where the dictionary is updated according to the examples. An important fact of applying NN-KSVD is that in the code book update stage, not only the current estimates of columns of matrix A are updated but also the nonzero entries of signals, that is, the codes that the corresponding code word (atom), are adjusted.

Now we propose an alternate algorithm, algorithm 4, for iteratively estimating the mixing matrix A and the sources \( x_t \), which can be seen as a variation of NN-KSVD—that is, in the sparse coding stage, the basis pursuit algorithm is replaced by one of our MAP estimation algorithms (the map coding stage):

**Algorithm 4: Combining MAP and NN-KSVD**

- Set initial matrix A randomly
- for \( \text{iter} = 1 \) to \( N_{\text{iter}} \)
  - **Sparse coding stage**: estimate sources \( x_t \) by MAP based on observations \( y_t \) and current A by using algorithm 1, 2 or 3.
  - **Code book update stage**: estimate new matrix (column by column) A and update only nonzero values of \( x_t \)
- endfor

6 **Estimation of Model Parameters \( \sigma_n \) and \( \rho^* \)**

The recovery of nonoverlapped signals (see algorithm 1) requires knowing a priori the model parameters—more precisely, the noise standard deviation \( \sigma_n \) and the sparsity parameter \( \rho^* \). In this section, we show that these parameters can be easily estimated from noisy mixtures by computing the first- and second-order moments of signals.\(^8\) Let us consider the one-dimensional signal \( z_t \) constructed as the sum of the entries of vector \( y_t \):

\[
z_t = \sum_{i=1}^{I} y_{it} = \sum_{j=1}^{J} \left( \sum_{i=1}^{I} a_{ij} \right) x_{jt} + \sum_{i=1}^{I} n_{it}. \tag{6.1}
\]

---

\(^8\)Note that a maximum likelihood estimation of \( \rho \) is difficult to implement since \( p(y|\rho) = \int p(y|x, \rho)p(x|\rho)dx \) is not available.
If we calculate the first moment of the signal $z_t$, we obtain

$$E[z_t] = \sum_{j=1}^{J} E[x_j] = J E[x].$$ (6.2)

Similarly, by calculating the second moment of $z_t$ and assuming that sources are nonoverlapped—$E[x_{j_1} x_{j_2}] = 0$ for $j_1 \neq j_2$—we obtain

$$E[z_t^2] = E[x^2] \sum_{i=1}^{I} a_{ij}^2 + \sigma_n^2.$$ (6.3)

Writing $E[x]$ and $E[x^2]$ in terms of the parameter $\rho^*$, we obtain a system of two equations with two unknown parameters ($\rho^*$, $\sigma_n$), which can be directly solved. For instance, for our uniformly distributed mixed-state variable model, the moments are $E[x] = \frac{1-\rho^*}{2}$ and $E[x^2] = \frac{1-\rho^*}{3}$. Then, from equations 6.2 and 6.3, we finally obtain

$$\rho^* = 1 - \frac{2E[z_t]}{a \sum_{i=1}^{I} a_{ij}},$$ (6.4)

$$\sigma_n^2 = \frac{1}{I} E[z_t^2] - \frac{2a}{3I} E[z_t] \sum_{i=1}^{I} a_{ij} \sum_{i=1}^{I} a_{ij}^2.$$ (6.5)

Note that equation 6.4 only is valid only when $\sum_{i=1}^{I} a_{ij} \neq 0$, and when matrix $A$ is positive, it can be written in terms of the $\ell_1$ norm of column $j$: $\|a_j\|_1 = \sum_{i=1}^{I} a_{ij}$. We also note that this estimator is insensitive to the scaling of columns $a_j$, and we can consider $\|a_j\|_1 = 1$, (in the positive matrix $A$ case) arriving at the following simplified equation:

$$\rho^* = 1 - \frac{2E[z_t]}{a J}.$$ (6.6)

7 Experimental Results

In this section we provide several experimental results on simulated data that confirm our theoretical development and show the performance of our source recovery algorithm for different scenarios. While we use simulated data, it is important to bear in mind the range of parameters in potential real applications like the case of spike trains generated by neurons. For

---

9Note that our equation 6.6 can be used also for estimating the number of sources $J$ under the assumption that the sparsity rate of signals $\rho^*$ is known.
example, the mammalian brain contains more than $10^{10}$ densely packed neurons, a huge and unmanageable number; experiments can be performed in the laboratory in order to isolate and measure signals only for a reduced number of neurons (dozens) by using multielectrode systems (Krüger & Aiple, 1988; Kaneko et al., 1996; de Ruyter van Steveninck, Lewen, Strong, Koberle, & Bialek, 1997).

For the evaluation of the results we use the signal-to-interference ratio (SIR), a widely used measure of quality of reconstruction. It is defined as the ratio of the standard deviations of the error and the original signal: $\text{SIR}_i(\text{dB}) = -20 \log\left(\frac{\sigma_{\epsilon_i}}{\sigma_{x_i}}\right)$, where $\sigma_{\epsilon_i}^2 = \text{Var}\left((x_i - \hat{x}_i)\right)$ and $\sigma_{x_i}^2 = \text{Var}\left(x_i\right)$. Depending on the application, a minimum SIR level is required for considering the obtained estimation as a good reconstruction. As can be seen in our simulations on very sparse signals, levels of SIR lower than 10 dB do not necessarily mean poor reconstructions as in other applications without such sparse signals.

Depending on each row of matrix $A$, the standard deviation of mixtures is not the same for all channels (sensors). Because want to add noise with a fixed standard deviation $\sigma_n$ in all channels, we define the SNR in terms of the average standard deviation of mixtures: $\text{SNR}(\text{dB}) = 20 \log\left(\frac{\sigma_{av}}{\sigma_n}\right)$, where $\sigma_{av} = \frac{1}{I} \sum_{i=1}^{I} \sigma_{u_i}$ corresponds to the averaged standard deviation of clean mixtures ($\sigma_{u_i}$ is the standard deviation of the clean mixture $u_i = [Ax]_i$).

In the following sections, we present our simulation results first, in section 7.1, for the case of a known matrix $A$, followed in section 7.2 by some results in a more blind situation where matrix $A$ is assumed to be unknown.

### 7.1 Recovering Sources with a Known Mixing Matrix $A$

#### 7.1.1 Cases with Nonoverlapped Sources (1-Sparse Sources)

The following three examples were analyzed for the case of having only two sensors ($I = 2$). Very sparse signals were considered using $T = 10,000$ samples. In all cases, the mixing matrix $A$ was randomly generated using a uniform distribution in $[0, 1]$ (no unit norm column condition was considered).

In example 1, we applied our MAP estimation algorithm to a case having $J = 6$ sources and a sparsity rate $\rho^* = 0.9995$ (defined in equation 3.12) under an extremely noisy condition with SNR $= 0$ dB. The obtained mean SIR was $21.16$ dB. In Figure 4a the best and worst estimates are shown together with the original sources and noisy mixtures signals. All the obtained SIR levels are also shown. Note that source 6 (the best estimate) was perfectly recovered, which is reflected in its high SIR level (36.52 dB). In the estimation of source 5 (the worst estimate), two spikes of the original source were not detected, and there are two artificially introduced spikes. This error in the estimation is reflected in its low SIR level (9.76 dB). However, it is important to highlight the poor conditions of this case: highly noisy SNR $= 0$ dB and underdetermined (six sources and only two sensors).
Figure 4: Results of example 1 (top), example 2 (middle), and example 3 (bottom). The best and worst estimates are shown in each case. Note that in example 3, only one spike was incorrectly estimated, the other spikes were estimated almost perfectly.

In example 2 we consider the case with $J = 10$ sources having a sparsity rate $\rho^* = 0.9995$ under a very noisy condition with SNR = 10 dB. The obtained mean SIR was 20.76 dB. In Figure 4b the best and worst estimates are shown together with the original sources and noisy mixture signals. All of the obtained SIR levels are also shown. Note that source 9 (the best estimate) was perfectly recovered, having a very high SIR level of 36.72 dB. Source 8 (the worst estimate) has 5 spikes, one of them was not detected, and two spikes were incorrectly detected, which affects dramatically its SIR level (11.59 dB). In this case, it is also important to highlight that it is a very difficult situation since we have 10 sources and only 2 sensors with a high level of noise (SNR = 10 dB).
In example 3 we consider a large-scale system with $J = 1000$ sources having a sparsity rate $\rho^* = 0.995^{10}$ with no noise$^{11}$ and only two observations. The obtained mean SIR was 205.54 dB. In Figure 4c the best and worst estimates are shown together with the original sources and two noisy mixtures signals. All the obtained SIR levels are also shown. Note that all sources were almost perfectly recovered—in particular, source 709 (the best estimate) and source 30 (the worst estimate) were recovered with $\text{SIR} = 220.20$ dB and $\text{SIR} = 172.03$ dB, respectively.

In order to understand the average behavior of our MAP estimation technique, a set of Monte Carlo (MC) simulations was carried out, taking into account different numbers of sources and sensors, a wide range of SNR values, and different levels of sparsity of sources:

- **Mean SIR versus SNR analysis:** We have applied algorithm 1 to randomly selected sources and mixing matrix, and the results were averaged over 50 simulations for each case. We considered different ratios of number of sources and number of sensors for a wide range of SNR (see Figure 5a). The obtained mean SIR is approximately linear with respect to the SNR, and the quality of estimations is better for higher sensor-to-noise ratios $I/J$.

- **Mean SIR versus sparsity analysis:** MC simulations were performed for $I = 2$ sensors, $J = 6$ sources, and three SNR levels (0 dB, 10 dB, and 20 dB) for a range in the parameter $\rho$. It can be shown from Figure 5b that the MAP estimate is very sensitive to the sparsity of signals arriving at very good estimations when sources are very sparse ($\rho > 0.99$) even for very low SNR. It is clear that sparsity of sources is the key factor that allows us to recover signals in such difficult situations with many more sources than sensors and with very high levels of noise.

- **Mean SIR versus number of sources:** MC simulations for different numbers of sources $J$, with only two sensors ($I = 2$) and very sparse signals ($\rho^* = 0.9995$) were performed (see Figure 5c). A high level of noise was added to mixtures, leading to $\text{SNR} = 10$ dB. It is interesting to note that our algorithm was able to recover good-quality sources even in this difficult situation.

7.1.2 Cases with Overlapped Sources (k-Sparse Sources). In order to test our algorithm with sources being overlapped at some indexes $t$, we generated nonoverlapped, nonnegative sources with $\rho^* = 0.9995$ and $T = 5000$ samples and then contaminated them with overlapped samples at random positions. We used $I = 6$ sensors, $J = 10$ sources, and $\text{SNR} = 20$ dB. In Figure 6

---

10 In example 3, fewer sparse sources than examples 1 and 2 are used to avoid having zero signals whose probability is not negligible.

11 In fact, a very small amount of noise was added, corresponding to $\text{SNR} = 200$ dB, which can be considered a noiseless case.
Figure 5: Monte Carlo (MC) results for 50 simulations. Mean SIR values versus SNR (a), sparsity coefficient $\rho$ (b), and number of sources $J$ (c) are shown.

Figure 6: Comparison of our MAP algorithm against nonnegative basis pursuit (NNBP) and FOCUSS+ algorithms for overlapped cases (2-sparse and 4-sparse sources with $I = 6$ sensors, $J = 10$ sources and SNR = 20 dB).

The obtained mean SIR values are shown versus the overlap percentage, which is calculated as the ratio of overlapped spikes to the total number of spikes available. We have generated 2-sparse sources (at most two active sources at a time) and 4-sparse sources (at most, four active sources at a time). In Figure 6 we show some comparisons with other algorithms, such as (nonnegative basis pursuit (Aharon et al., 2005), and FOCUSS+ (Murray & Kreutz-Delgado, 2004). Note that our MAP algorithm always produces better results than the other methods for 2-sparse signals; when 4-sources are considered, our MAP algorithm produces better results than FOCUSS+ only for overlap percentage levels lower than 50%, arriving at similar solutions for higher levels of the overlap percentage.

7.2 Blind Identification of Matrix $A$ and NN-KSVD Algorithm. We have shown that MAP criteria can provide good estimations of sources if we know the mixing matrix $A$. In many real applications, we do not
Figure 7: Mean SIRs for the estimation of matrix $A$ (left) and sources $x$ (right). Comparison of original NN-KSVD algorithm (Aharon et al., 2005) against our method combining MAP with NN-KSVD. $J = 10$ sources, $I = 3$ sensors, coherence coefficient $\mu(A) = 0.9915$ and an effective sparsity coefficient of $\rho^* = 0.999$ was used.

know the matrix $A$ and need to estimate it in a blind fashion together with sources $x$. In this section, we evaluate our method, which combines the MAP estimation technique for estimating sources with the algorithm NN-KSVD for estimating the mixing matrix (dictionary).

It can be shown that the uniqueness of the solution and the quality in the estimation of matrix $A$ are highly dependent on the coherence of column vectors (Bruckstein, Elad, & Zibulevsky, 2008). The coherence is a measure of colinearity of a set of vectors (columns of $A$) defined as follows:

$$\mu(A) = \max \left( \frac{\langle a_i, a_j \rangle}{\|a_i\|^2 \|a_j\|^2} \right), \text{ with } i \neq j. \quad (7.1)$$

Figure 7 shows the efficiency of our algorithm (MAP combined with NN-KSVD) compared with the classic NN-KSVD (using basis pursuit). In this experiment, we have considered a fixed set of $J = 10$ 1-sparse sources, $I = 3$ sensors, and a known matrix $A$ with unit-norm column vectors and an associated coherence coefficient $\mu(A) = 0.9915$. The number of samples was $T = 10,000$, and a sparsity rate of $\rho^* = 0.999$ was used.

In order to illustrate the behavior of both alternatives, we present in Figure 8 an example of our previous experiments for $\text{SNR} = 5$ dB. A particular source and its associated estimates are shown. Our method produces a clean estimation of Source 3 with a SIR level of $30.3$ dB, while the classical NN-KSVD tends to produce noisy reconstructions of sources with low SIR levels ($3.33$ dB for source 3). Moreover, the advantage of our method over the classical NN-KSVD is not only the absence of background noise, which could also be removed by a nonlinear filter (threshold); in addition, our
Figure 8: Comparison of source 3 recovery with original NN-KSVD algorithm (top right) against our combined MAP algorithm (bottom right) for a case with SNR = 5 dB. By using NN-KSVD, we lost one spike, while our method recovered almost all spikes perfectly.

Figure 9: Comparison of SIR levels obtained with the original NN-KSVD algorithm (left) against SIR levels obtained with our combined MAP technique (right).

Our method is able to detect spikes more efficiently than NN-KSVD can (see Figure 8).

In Figure 9, the obtained SIR values for matrix $A$ and sources are shown for both methods in an experiment with the same conditions as before (SNR = 5 dB, $J = 10$ sources, $I = 3$ sensors, $T = 10,000$, and $\rho^* = 0.999$). Our method provides better results than the classical NN-KSVD algorithm.
8 Conclusion

We presented a probabilistic model for sparse signals and developed a fast and simple algorithm for underdetermined noisy BSS that has many potential applications, such as the case of spike trains separation in neuroscience. Furthermore, our theoretical analysis allowed us to analyze the deep connections with other areas of research for signal processing, such as models for sparse representation of data with overcomplete dictionaries. We have shown that the recovery of sources in difficult scenarios, with very high levels of noise and many more sources than sensors (underdetermined case), is achieved when sparse signals are considered.

It is important to highlight that we introduced a new probabilistic model for sparsity based on its basic definition—avoiding the classical probabilistic approach with continuous distributions like exponential or generalized gaussian. Moreover, our model better fits the original definition of sparsity by assigning a high probability to the zero state and low probabilities to any nonzero state. As our experimental results show, this mixed-state model represents a great advantage for noise reduction.

It is not a surprise that this mixed-state prior distribution, together with the Bayesian approach, leads to an objective function having a regularization term based on the \( \ell_0 \) norm for which an iterative maximization method based on derivatives is not applicable. In this sense, our method provides a solution to the intractable limiting case \( p \to 0 \) of FOCUSS+ in the high-sparsity limit. In our model, the combinatorial search of the minimum \( \ell_0 \) norm solution is avoided by using the nonoverlapped assumption, arriving at a very simple and fast algorithm. For the case of overlapped sources, as they can be expressed as a sum of nonoverlapped sources, we propose to apply successively the 1-sparse source extraction for general \( k \)-sparse signals. We also implemented a combination of the NN-KSVD algorithm with our MAP estimation of sources, providing a solution to the estimation of sources, and mixing matrix in a blind scenario. Our algorithm was able to recover sources and a mixing matrix with better quality compared with the original NN-KSVD algorithm.

We have provided a complete set of simulations based on synthetic data inspired in neuroscience applications that showed the effectiveness of our method and suggests that this method can be applied to other source separation problems where sources are characterized by spike trains.

Appendix: Partial Objective Functions Derivation

The derivation of \( C_0 \) in equation (4.2) is easily obtained by replacing \( x = 0 \) and by using equation 3.11 in equation 2.4. For the derivation of the partial objective functions \( C_j \) with \( j = 1, 2, \ldots, J \) we first expand the general
objective function equation 2.4, in the following way:

\[
J(x) = \frac{1}{2\sigma_n^2} \| y - Ax \|_2^2 + \log[p(x)]
\]

\[
= \frac{1}{2\sigma_n^2} \sum_{i=1}^{l} \left( y_i - \sum_{k=1}^{l} a_{ik} x_k \right)^2 + \log[p(x)],
\]

where index \( t \) was omitted for simplifying notation. Now, taking into account configurations with only one variable in active state denoted by \( x_j \), that is, \( x_k = 0 \ \forall \ k \neq j \), and using equation 3.11 we obtain the following equation:

\[
J(x) = \frac{1}{2\sigma_n^2} \sum_{i=1}^{l} (y_i - a_{ij} x_j)^2 + \log \left[ \frac{1 - \rho}{f} \right]
\]

\[
= \frac{1}{2\sigma_n^2} \sum_{i=1}^{l} (y_i^2 - 2y_i a_{ij} x_j + a_{ij}^2 x_j^2) + \log \left[ \frac{1 - \rho}{f} \right]
\]

\[
= \frac{1}{2\sigma_n^2} \sum_{i=1}^{l} y_i^2 - \frac{1}{\sigma_n^2} \left( \sum_{i=1}^{l} y_i a_{ij} \right) x_j + \frac{1}{2\sigma_n^2} \left( \sum_{i=1}^{l} a_{ij}^2 \right) x_j^2 + \log \left[ \frac{1 - \rho}{f} \right].
\]

Then we define the coefficients

\[
\alpha_j = \frac{1}{\sigma_n^2} \sum_{i=1}^{l} y_i a_{ij},
\]

\[
\beta_j = -\frac{1}{2\sigma_n^2} \sum_{i=1}^{l} a_{ij}^2
\]

and the final equation 4.3, is obtained.

References


Received August 21, 2008; accepted April 17, 2009.