SOFT BAYESIAN PURSUIT ALGORITHM FOR SPARSE REPRESENTATIONS

Angélique Drémeau [a,b], *Cédric Herzet* [c] *and Laurent Daudet* [a]

[a] Institut Langevin, ESPCI ParisTech, Univ Paris Diderot, CNRS UMR 7587, F-75005 Paris, France
[b] Fondation Pierre - Gilles De Gennes pour la Recherche, 29 rue d'Ulm, F-75005 Paris, France
[c] INRIA Centre Rennes - Bretagne Atlantique, Campus universitaire de Beaulieu, F-35000 Rennes, France

ABSTRACT

In this paper, we address the problem of sparse representation within a Bayesian framework. As a continuation of previous work [1], we consider a Bernoulli-Gaussian model and the use of a mean-field approximation. The resulting algorithm is shown to have very good performance over a wide range of sparsity levels.

Index Terms— Sparse representations, Bernoulli-Gaussian model, mean-field approximation.

1. INTRODUCTION

Sparse representations (SR) aim at describing a signal as the combination of a small number of atoms chosen from an overcomplete dictionary. More precisely, let $\mathbf{y} \in \mathbb{R}^N$ be an observed signal and $\mathbf{D} \in \mathbb{R}^{N\times M}$ a rank-N matrix whose columns are normalized to 1. One possible formulation of the SR problem writes

$$\mathbf{x}^\star = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_0, \tag{1}$$

where $\|\mathbf{x}\|_0$ denotes the number of nonzero elements in $\mathbf{x}$ and λ is a parameter specifying the trade-off between sparsity and distortion.

Finding the exact solution of (1) is usually an intractable problem. Hence, suboptimal algorithms have to be considered in practice. Among the large number of SR algorithms available in the literature, let us mention: *iterative hard thresholding* (IHT) [2], which iteratively thresholds to zero certain coefficients of the projection of the SR residual on the considered dictionary; *matching pursuit* (MP) [3] or *subspace pursuit* (SP) [4] which build up the sparse vector $\mathbf{x}$ by making a succession of greedy decisions; and *basis pursuit* (BP) [5] which solves a relaxed version of (1) by means of standard convex optimization procedures.

A particular family of SR algorithms relies on a Bayesian formulation of the SR problem, see *e.g.*, [6, 7, 8]. In a nutshell, the idea of these approaches is to model $\mathbf{y}$ as the output of a stochastic process (promoting sparsity on $\mathbf{x}$) and apply statistical tools to infer the value of $\mathbf{x}$. In this context, we recently introduced [1] a new family of Bayesian pursuit algorithms based on a Bernoulli-Gaussian probabilistic model. These algorithms generate a solution of the SR problem by making a sequence of *hard decisions* on the support of the sparse representation.

In this paper, exploiting our previous work [1], we propose a novel SR algorithm dealing with "soft" decisions on the support of the sparse representation. Our algorithm is based on the combination of a Bernoulli-Gaussian (BG) model and a mean-field (MF) approximation. The proposed methodology allows for keeping a measure of the uncertainty on the decisions made on the support throughout the whole estimation process. We show that, as long as our simulation setup is concerned, the proposed algorithm is very competitive with state-of-the-art procedures.

2. MODEL AND BAYESIAN PURSUIT

In this section, we first introduce the probabilistic model which will be used to derive our SR algorithm. Then, for the sake of comparison with the proposed methodology, we briefly recall the main expressions of the Bayesian Matching Pursuit (BMP) algorithm introduced in [1].

2.1. Probabilistic Model

Let $\mathbf{s} \in \{0,1\}^M$ be a vector defining the SR *support*, *i.e.,* the subset of columns of $\mathbf{D}$ used to generate $\mathbf{y}$. Without loss of generality, we will adopt the following convention: if $s_i = 1$ (resp. $s_i = 0$), the ith column of $\mathbf{D}$ is (resp. is not) used to form $\mathbf{y}$. Denoting by $\mathbf{d}_i$ the ith column of $\mathbf{D}$, we then consider the following observation model:

$$\mathbf{y} = \sum_{i=1}^{M} s_i\, x_i\, \mathbf{d}_i + \mathbf{n}, \tag{2}$$

where $\mathbf{n}$ is a zero-mean white Gaussian noise with variance σ_n^2. Therefore,

$$p(\mathbf{y}|\mathbf{x},\mathbf{s}) = \mathcal{N}(\mathbf{D}_\mathbf{s}\mathbf{x}_\mathbf{s}, \sigma_n^2\mathbf{I}_N), \tag{3}$$

where $\mathbf{I}_N$ is the $N \times N$-identity matrix and $\mathbf{D}_\mathbf{s}$ (resp. $\mathbf{x}_\mathbf{s}$) is a matrix (resp. vector) made up of the $\mathbf{d}_i$'s (resp. x_i's) such

that $s_i = 1$. We suppose that $\mathbf{x}$ and $\mathbf{s}$ obey the following probabilistic model:

$$p(\mathbf{x}) = \prod_{i=1}^{M} p(x_i), \qquad p(\mathbf{s}) = \prod_{i=1}^{M} p(s_i), \tag{4}$$

where $p(x_i) = \mathcal{N}(0, \sigma_x^2)$, $p(s_i) = \text{Ber}(p_i)$, and $\text{Ber}(p_i)$ denotes a Bernoulli distribution with parameter p_i.

Note that model (3)-(4) (or variants thereof) has already been used in many Bayesian algorithms available in the literature, see *e.g.*, [1, 6, 9, 10]. The originality of this contribution is in the way we exploit it.

2.2. Bayesian Matching Pursuit

We recently showed in [1] that, under mild conditions, the solution of the maximum a posteriori (MAP) estimation problem,

$$(\hat{\mathbf{x}}, \hat{\mathbf{s}}) = \arg\max_{\mathbf{x},\mathbf{s}} \log p(\mathbf{x}, \mathbf{s}|\mathbf{y}), \tag{5}$$

is equal to the solution of the standard SR problem (1). This result led us to the design of a new family of Bayesian pursuit algorithms. In particular, we recall hereafter the main expressions of the Bayesian Matching Pursuit (BMP) algorithm.

BMP is an iterative procedure looking sequentially for a solution of (5). It proceeds like its standard homologue MP by modifying one unique couple (x_i, s_i) at each iteration, namely the one leading to the highest increase of $\log p(\mathbf{x}, \mathbf{s}|\mathbf{y})$. It can then be shown that the (locally) optimal update of the selected coefficient x_i is given by

$$\hat{x}_i^{(n)} = \hat{s}_i^{(n)} \frac{\sigma_x^2}{\sigma_n^2 + \sigma_x^2} \mathbf{r}_i^{(n)T} \mathbf{d}_i, \tag{6}$$

$$\text{where} \qquad \mathbf{r}_i^{(n)} = \mathbf{y} - \sum_{j \neq i} \hat{s}_j^{(n-1)} \hat{x}_j^{(n-1)} \mathbf{d}_j, \tag{7}$$

and n is the iteration number.

3. A NEW SR ALGORITHM BASED ON A MEAN-FIELD APPROXIMATION

The equivalence between (5) and (1) motivates the use of model (3)-(4) in SR problems and offers interesting perspectives. We study in this paper the possibility of considering some of the variables as hidden. In particular, we consider the problem of making a decision on the SR support as

$$\hat{\mathbf{s}} = \arg\max_{\mathbf{s} \in \{0,1\}^M} \log p(\mathbf{s}|\mathbf{y}), \tag{8}$$

where $p(\mathbf{s}|\mathbf{y}) = \int_{\mathbf{x}} p(\mathbf{x}, \mathbf{s}|\mathbf{y}) d\mathbf{x}$. Note that, as long as (3)-(4) is the true generative model for the observations $\mathbf{y}$, (8) is the decision minimizing the probability of wrong decision on the SR support. It is therefore optimal in that sense.

Unfortunately, problem (8) is intractable since it typically requires to evaluate the cost function, $\log p(\mathbf{s}|\mathbf{y})$, for all possible 2^M sequences in $\{0,1\}^M$. In this paper, we propose to simplify this optimization problem by considering a MF approximation of $p(\mathbf{x}, \mathbf{s}|\mathbf{y})$.

Note that the combination of a BG model and MF approximations to address the SR problem has already been considered in some contributions [8, 11]. However, the latter differ from the proposed approach in several aspects. In [8], the authors considered a tree-structured version of BG model which was dedicated to a specific application (namely, the sparse decomposition of an image in wavelet or DCT bases). Moreover, the authors considered a different MF approximation than the one proposed here (see section 3.1). In [11], we applied MF approximations to a different BG model, which led to different SR algorithms.

3.1. MF approximation $p(\mathbf{x}, \mathbf{s}|\mathbf{y}) \simeq \prod_i q(x_i, s_i)$

A MF approximation of $p(\mathbf{x}, \mathbf{s}|\mathbf{y})$ is a probability distribution constrained to have a "suitable" factorization while minimizing the Kullback-Leibler distance with $p(\mathbf{x}, \mathbf{s}|\mathbf{y})$. This estimation problem can be solved by the so-called "variational Bayes EM (VB-EM) algorithm", which iteratively evaluates the different elements of the factorization. We refer the reader to [12] for a detailed description of the VB-EM algorithm.

In this paper, we consider the particular case where the MF approximation of $p(\mathbf{x}, \mathbf{s}|\mathbf{y})$, say $q(\mathbf{x}, \mathbf{s})$, is constrained to have the following structure:

$$q(\mathbf{x}, \mathbf{s}) = \prod_i q(x_i, s_i). \tag{9}$$

Particularized to (9), the VB-EM algorithm evaluates the $q(x_i, s_i)$'s by computing at each iteration[1]:

$$\forall i \quad q(x_i, s_i) = q(x_i|s_i)\, q(s_i), \tag{10}$$

where

$$q(x_i|s_i) = \mathcal{N}(m(s_i), \Gamma(s_i)), \tag{11}$$

$$q(s_i) \simeq \sqrt{2\pi\Gamma(s_i)} \exp\left(-\frac{1}{2}\frac{m(s_i)^2}{\Gamma(s_i)}\right) p(s_i), \tag{12}$$

$$\text{and} \qquad \Gamma(s_i) = \frac{\sigma_x^2 \sigma_n^2}{\sigma_n^2 + \sigma_x^2 s_i}, \tag{13}$$

$$m(s_i) = s_i \frac{\sigma_x^2}{\sigma_n^2 + \sigma_x^2 s_i} \langle \mathbf{r}_i \rangle^T \mathbf{d}_i, \tag{14}$$

$$\langle \mathbf{r}_i \rangle = \mathbf{y} - \sum_{j \neq i} q(s_i = 1) m(s_i = 1)\, \mathbf{d}_j. \tag{15}$$

Note that the VB-EM algorithm is ensured to converge to a saddle point or a (local or global) maximum of the problem.

[1] When clear from the context, we will drop the iteration indices in the rest of the paper.

At this point of the discussion, it can be interesting to compare both the proposed algorithm and BMP:

i) Although the nature of the update may appear quite different (BMP makes a hard decision on the (x_i, s_i)'s whereas the proposed algorithm rather updates probabilities on the latter), both algorithms share some similarities. In particular, the mean of distribution $q(x_i|s_i)$ computed by the proposed algorithm (14) has the same form as the coefficient update performed by BMP (6). They rely however on different variables, namely the residual $\mathbf{r}_i$, (7), and its mean $\langle \mathbf{r}_i \rangle$, (15). This fundamental difference between both algorithms leads to well distinct approaches. In BMP, a hard decision is made on the SR support at each iteration: the atoms of the dictionary are either used or not (each $\hat{x}_j^{(n-1)}$ is multiplied by $\hat{s}_j^{(n-1)}$ which is equal to 0 or 1). On the contrary, in the proposed algorithm, the contributions of the atoms are simply weighted by $q(s_j = 1)$, *i.e.,* the probability distributions of the s_j's. In a similar way, the coefficients $\hat{x}_j^{(n-1)}$'s used in (7) are replaced by their means $m(s_j = 1)$ in (15), taking into account the uncertainties we have on the values of the x_j's.

ii) The complexity of one update step is similar in both algorithms and equal to MP: the most expensive operation is the update equation (15) which scales as $\mathcal{O}(NM)$. However, in BMP *one* unique couple (x_i, s_i) is involved at each iteration while in the proposed algorithm *all* indices are updated one after the other. To the extend of our experiments (see section 4), we could observe that the proposed algorithm converges in a reasonable number of iterations, keeping it at a competitive place beside state-of-the-art algorithms.

3.2. Simplification of the support decision problem

Coming back to the maximum a posteriori problem (8) and exploiting MF approximation (9), we obtain

$$\hat{s}_i = \underset{s_i \in \{0,1\}}{\arg\max} \log q(s_i) \quad \forall i, \tag{16}$$

where we have used the following approximation:

$$p(\mathbf{s}|\mathbf{y}) \simeq \int_{\mathbf{x}} \prod_i q(x_i, s_i)\, d\mathbf{x}, = \prod_i q(s_i).$$

The solution of (16) can then be found by a simple thresholding operation: $\hat{s}_i = 1$ if $q(s_i = 1) > 1/2$ and $\hat{s}_i = 0$ otherwise.

3.3. Estimation of the noise variance

The estimation of unknown model parameters can easily be embedded within the VB-EM procedure (9)-(15). In particular, we estimate the noise variance via the procedure described in [13]. This leads to the following update on the noise variance:

$$\hat{\sigma}_n^2 = \frac{1}{N} \left\langle \|\mathbf{y} - \sum_i s_i x_i \mathbf{d}_i\|^2 \right\rangle_{\prod_i q(x_i,s_i)} \tag{17}$$

where $\langle f(\theta) \rangle_{q(\theta)} \triangleq \int_\theta f(\theta)\, q(\theta)\, d\theta$.

Note that although being in principle unnecessary when the noise variance is known, we found that including the noise-variance update (17) in the VB-EM iterations improves the convergence. An intuitive explanation of this behavior can be given by observing that, at a given iteration, $\hat{\sigma}_n^2$ is a measure of the (mean) discrepancies between the observation and the sparse model.

4. SIMULATIONS

In this section, we study the performance of the proposed algorithm by extensive computer simulations. In particular, we assess its performance in terms of the reconstruction of the SR support and the estimation of the nonzero coefficients. To that end, we evaluate different figures of merit as a function of the number of atoms used to generate the data, say K: the ratio of the average number of false detections to K, the ratio of the average number of missed detections to K and the mean-square error (MSE) between the nonzero coefficients and their estimates.

Using (16), we reconstruct the coefficients of a sparse representation given its estimated support $\hat{\mathbf{s}}$, say $\hat{\mathbf{x}}_{\hat{\mathbf{s}}}$, by

$$\hat{\mathbf{x}}_{\hat{\mathbf{s}}} = \mathbf{D}_{\hat{\mathbf{s}}}^{+} \mathbf{y}, \tag{18}$$

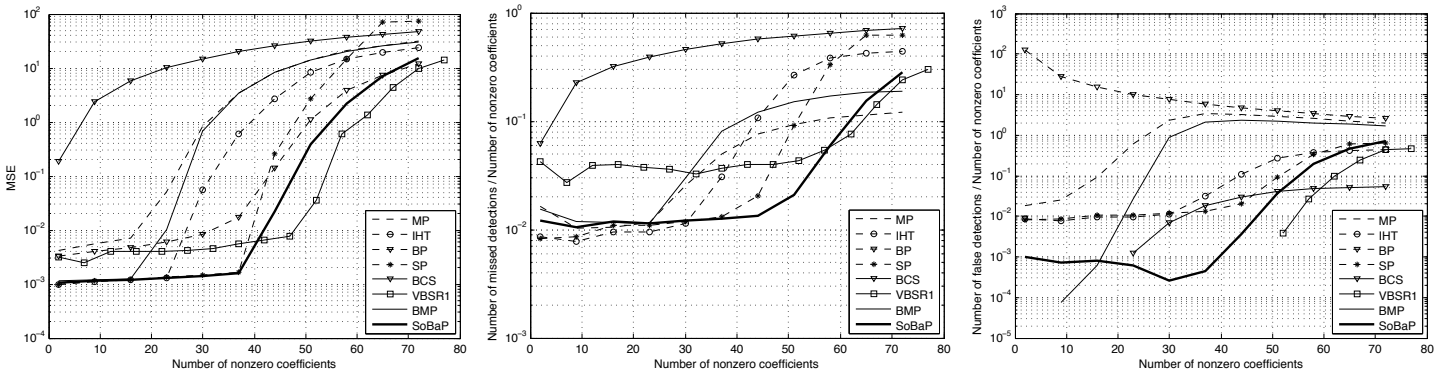
where $\mathbf{D}_{\hat{\mathbf{s}}}^{+}$ is the Moore-Penrose pseudo-inverse of the matrix made up of the $\mathbf{d}_i$'s such that $\hat{s}_i = 1$. In the sequel, we will refer to the procedure defined in (11)-(18) as Soft Bayesian Pursuit (SoBaP) algorithm.

Observations are generated according to model (3)-(4). We use the following parameters: $N = 128$, $M = 256$, $\sigma_n^2 = 10^{-3}$, $\sigma_x^2 = 100$. For the sake of fair comparisons with standard algorithms, we consider the case where all atoms have the same occurrence probability, *i.e.,* $p_i = K/M$, $\forall i$. Finally the elements of the dictionary are *i.i.d.* realizations of a zero-mean Gaussian distribution with variance N^{-1}. For each point of simulation, we run 1500 trials.

We evaluate and compare the performance of 8 different algorithms: MP, IHT, BP, SP, BCS, VBSR1([11]), BMP and SoBaP. We use algorithm implementations available on author's webpages[2]. VBSR1 is run for 50 iterations. MP is run until the ℓ_2-norm of the residual drops below $\sqrt{N\sigma_n^2}$. SoBaP is run until the estimated noise variance drops below 10^{-3}.

Fig.1(a) shows the MSE on the nonzero coefficients according to the number of nonzero coefficients, K, for each considered algorithm. For $K \geq 40$, we can observe that SoBaP is dominated by VBSR1 but outperforms all other algorithms. Below this bound, while VBSR1 presents a quite bad performance with regard to IHT (up to $K = 22$), SP (up to $K = 38$) and BMP (up to $K = 20$), SoBaP keeps a good behavior beside these algorithms.

[2]resp. at *http://www.personal.soton.ac.uk/tb1m08/sparsify/sparsify.html*, *http://sites.google.com/site/igorcarron2/cscodes/*, *http://www.acm.caltech.edu/l1magic/* (ℓ_1-magic)



(a) MSE on nonzero coefficients (b) Average number of missed detections (c) Average number of false detections

Fig. 1. SR reconstruction performance versus number of nonzero coefficients K.

Fig.1(b) and Fig.1(c) represent the algorithm performance for the reconstruction of the SR support. We can observe that SoBaP succeeds in keeping both small missed detection and false detection rates on a large range of sparsity levels. This is not the case for the other algorithms. If some of them (IHT and SP in Fig.1(b), BMP in Fig.1(c)) present better performance for small values of K, the gains are very slight in comparison to the large deficits observed for greater values. Note finally that Fig.1(b) and Fig.1(c) explain to some extent the singular behavior of VBSR1 observed in Fig.1(a). Below $K = 50$, each atom selected by VBSR1 is a "good" one, *i.e.,* has been used to generate the data, but this is performed at the expense of the missed detection rate, which remains quite high for small numbers of nonzero coefficients. This "thrifty" strategy is also chosen by BP to a large extent.

5. CONCLUSION

In this paper, we consider the SR problem within a BG framework. We propose a tractable solution by resorting to a MF approximation and the VB-EM algorithm. The resulting algorithm is shown to have very good performance over a wide range of sparsity levels, in comparison to other state-of-the-art algorithms. This comes with a low complexity per update step, similar to MP. Dealing with soft decisions seems to be a promising way to solve SR problems and is de facto more and more considered in literature (*e.g.*, [14]).

6. REFERENCES

[1] C. Herzet and A. Dremeau, "Bayesian pursuit algorithms," in *Proc. EUSIPCO*, 2010.

[2] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 629–654, December 2008.

[3] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. On Signal Processing*, vol. 41, no. 12, pp. 3397–3415, December 1993.

[4] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. On Information Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.

[5] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.

[6] C. Soussen, J. Idier, D. Brie, and J. Duan, "From bernoulli-gaussian deconvolution to sparse signal restoration," Tech. Rep., CRAN/IRCCyN, 2010.

[7] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[8] L. He, H. Chen, and L. Carin, "Tree-structured compressive sensing with variational bayesian analysis," *IEEE Signal Processing Letters*, vol. 17, pp. 233–236, 2010.

[9] H. Zayyani, M. Babaie-Zadeh, and C. Jutten, "Sparse component analysis in presence of noise using em-map," in *Proc. ICA*, 2007.

[10] H. Zayyani, M. Babaie-Zadeh, and C. Jutten, "An iterative bayesian algorithm for sparse component analysis in presence of noise," *IEEE Trans. On Signal Processing*, vol. 57, pp. 4378–4390, 2009.

[11] C. Herzet and A. Dremeau, "Sparse representation algorithms based on mean-field approximations," in *Proc. ICASSP*, 2010, pp. 2034–2037.

[12] M. J. Beal and Z. Ghahramani, "The variational bayesian em algorithm for incomplete data: with application to scoring graphical model structures," *Bayesian Statistics*, vol. 7, pp. 453–463, 2003.

[13] T. Heskes, O. Zoeter, and W. Wiegerinck, *Approximate Expectation Maximization*, MIT Press. Advances in Neural Information Processing Systems 16, Cambridge, MA, 2004.

[14] A. Divekar and O. Ersoy, "Probabilistic matching pursuit for compressive sensing," Tech. Rep., School of Electrical and Computer Engineering, 2010.