A measure-theoretic variational Bayesian algorithm for large dimensional problems

Aurélia Fraysse, Thomas Rodet

A MEASURE-THEORETIC VARIATIONAL BAYESIAN ALGORITHM FOR LARGE DIMENSIONAL PROBLEMS.

A. FRAYSSE[†] AND T. RODET[‡]

Abstract. In this paper we provide an algorithm adapted to the variational Bayesian approximation. The main contribution is to transpose a classical iterative algorithm of optimization in the metric space of measures involved in the Bayesian methodology. Once given the convergence properties of this algorithm, we consider its application to large dimensional inverse problems, especially for unsupervised reconstruction. The interest of our method is enhanced by its application to large dimensional linear inverse problems involving sparse objects. Finally, we provide simulation results. First we show the good numerical performances of our method compared to classical ones on a small example. On a second time we deal with a large dimensional dictionary learning problem.

keywords: ill-posed inverse problems, variational Bayesian methodology, sparse signal reconstruction, infinite dimensional convex optimization

1. Introduction. The recent development of information technologies has increased the expansion of inverse problems for very large dimensional datasets. There is currently a significant growth in the number of measurements involved in reconstruction problems together with an expansion in the size of considered objects. Thus it is often necessary to handle the reconstruction of more than one million parameters. At the same time, signal processing techniques have helped to overcome the limitations of measurement instruments and to supply the design of systems involving indirect measures. These new instruments introduced different signal processing challenges, such as super resolution, deconvolution, source separation or tomographic reconstruction. All these problems are ill posed, the only information contained in the data and in the model of acquisition are not sufficient to obtain a good estimation of the source objects.

To solve these ill-posed problems, additional *a priori* information is often needed. The Bayesian approach appearing in this paper consists in a modelisation of sources of information as probability density functions [11, 24, 18]. This approach allows the development of unsupervised estimation methods, such that the parameters of probability distributions also called hyperparameters, typically the different variances, are adjusted automatically. These hyperparameters can tune the trade-off between information coming from data (likelihood) and *a priori* information. All the information related to the object and to the hyperparameters is summarized by the posterior distribution. This distribution allows the construction of several estimators such as the maximum *a posteriori* (MAP) or the posterior mean (PM), optimal for some given cost functions. These estimators can in fact be obtained by summing up the information contained in the posterior distribution. However the information given only by these estimators cannot be sufficient in the context of inverse problems, especially when recursive or online methods are required. Moreover the determination of these estimators needs an exact knowledge of the posterior distribution, which is generally intractable. Hence this posterior distribution is in practice approximated thanks, in

[†]L2S, SUPELEC, CNRS, University Paris-Sud, 3 rue Joliot-Curie, 91190 Gif-Sur-Yvette, FRANCE. email:aurelia.fraysse@lss.supelec.fr

[‡]SATIE, ENS Cachan, CNAM, Av. du Prés. Wilson, 94230 Cachan, FRANCE. email: trodet@satie.ens.cachan.fr

general, to numerical methods. The most classical method is given by Markov Chains Monte Carlo(MCMC) methods, see [35, 36], or particular filtering [12] in recursive frameworks. However, these methods are no longer tractable for large dimensional problems. Indeed, in this case, the rate of convergence is too slow to allow reasonable practical use.

Another approach, considered here, is to determine an analytical approximation which leads to Laplace approximation or to Variational Bayesian Approximation (VBA), [43]. These approaches are more suitable when the full information has to be propagated along recursive methods, see [43, 1] or for experimental design [38, 39] where marginal distributions need to be calculated quickly. One of the first works on variational Bayesian inference was introduced by D. MacKay in [25]. This methodology has been used in a computer science context, see [20] for instance, and in signal processing for different applications such as source separation using ICA [27, 10], Mixture Models estimation [30], hyperspectral imaging reduction [5], deconvolution [8, 3] or recursive methods [42, 43]. The main outline of the variational Bayesian methodology is to determine an approximation, in terms of Kullback-Leiber divergence, of the posterior distribution by a simpler, typically separable, law. Thanks to this method, the initial inverse problem appears as a convex optimization problem in a function space. Moreover, in classical Bayesian variational methods, the prior distribution is chosen to be conjugated with the likelihood one. Hence the posterior distribution belongs to a given family and the optimization becomes an optimization of its parameters. Unfortunately, this variational method induces intricate formula that require the implementation of numerical approximations. They are hence approximated by fixed point methods such as alternate minimization. For large dimensional problems this turn out to be an important drawback. Some recent works, such as [4] [39], proposed accelerated variational Bayesian algorithms based on iterative approximations for update equations.

In the present paper we choose a different approach by directly considering a numerical resolution of the infinite dimensional optimization problem involved in the variational Bayesian framework. This approach induces fewer approximations and ensures numerical convergence to the true solution of the variational Bayesian optimization problem. The goal is to construct an iterative algorithm able to provide in a reduced computation time a close approximation of the solution of the functional variational problem. The main principle is to adapt a classical optimization algorithm, the gradient descent method [33], to the space of probability measures. Based on this principle, we derive here an efficient algorithm for the variational Bayesian framework, based on an optimization in the density probability functions. Our algorithm is based on the exponentiated gradient introduced in [23] for discrete measures. The first contribution of this paper is to set out the definition and the convergence properties of this algorithm in the probability density functions set. In Section 3 we also illustrate its application in the context induced by the variational Bayesian methodology.

The second contribution, exposed in Section 4, consists in the application of the mentioned methodology to linear inverse problems given by the Gaussian white noise model. Concerning the prior distribution, we emphasizeinformation of sparsity. A sparse Bayesian methodology using a Laplace prior was developed by Seeger [38] [40] or Elad [49]. But more generally, a sparse prior information is introduced by

heavy-tailed distributions, such as Bernoulli Gaussian law, [16], mixtures of Gaussian [44, 17], Cauchy distribution, or α-stable distributions. The Gaussian Scale Mixture class, a generalization of the previous ones, is introduced in [46] as a model of wavelet coefficients of natural images. The main advantage of Gaussian Scale Mixtures is that they can moreover be easily written as Gaussian distributions, conditioned by a hidden variable and thus allow the use of Gaussian based tools.

In Section 5, we present simulation results, first on a tomographic example where the *a priori* information promotes pulses or extremely sparse solutions and secondly on an identification problem in a very large dictionary learning context. Finally, Section 6 concludes the paper.

2. Optimization algorithm in a measures space. In this part, we assume that we stand in the measurable space $(\mathbb{R}^N, \mathcal{B}(\mathbb{R}^N))$, where $\mathcal{B}(\mathbb{R}^N)$ is the σ-field of Borel sets of $\mathbb{R}^N$. This section is devoted to the construction of a gradient based algorithm adapted to the optimization problem involved in the variational Bayesian methodology. This problem can be seen as a maximization of a concave functional in a probability density functions set.

Concerning probability density functions there are several possible representations of such objects. The first one is to consider that this set is a subset of the positive cone of $L^1(\mathbb{R}^N)$. As $L^1(\mathbb{R}^N)$ is a Banach space, classical optimization algorithms holds in this case. However, one has to pay a particular attention to the fact that the positivity of the functions together with the fixed total mass imposes additional constraints which can be hardly tractable in some case. Another point of view, see [28], is to consider this set as a subset of the space of signed Radon measures $\mathcal{M}(\mathbb{R}^N)$, that is measures that can be written as $\mu = \mu^+ - \mu^-$, endowed with the norm of total variation. Once again this is a Banach space, see [28]. The classical gradient descent can also be adapted to this framework, as shown in [29]. However in [29], measures obtained at each iteration no longer have densities, and this algorithm cannot converge to a solution of our optimization problem. Moreover, this space does not take into account the separability property of measures, which plays a crucial part in our case. Hence in the following we rather consider the Cartesian product of one-dimensional Radon measure spaces, $\widetilde{\mathcal{M}}(\mathbb{R}^N) = \prod_{i=1}^N \mathcal{M}(\mathbb{R})$, endowed with the norm:

$$\forall \mu \in \widetilde{\mathcal{M}}, \quad \|\mu\|_{TV} = \prod_{i=1}^{N} \sup_{A_i \in \mathcal{B}(\mathbb{R})} \int_{A_i} \mathrm{d}\mu_i^+(x_i) + \int_{A_i} \mathrm{d}\mu_i^-(x_i). \tag{2.1}$$

Note that when μ is a density measure, i.e. $\mathrm{d}\mu(\boldsymbol{x}) = q(\boldsymbol{x})\mathrm{d}\mathcal{L}(\boldsymbol{x}) = q(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$, $\mathcal{L}$ standing for the Lebesgue measure, its total variation norm coincides with the L^1 norm of its density function q.

Furthermore, a separable probability density function is considered as an element of the closed convex set Ω, defined by

$$\Omega = \{\mu \in \widetilde{\mathcal{M}};\ \mathrm{d}\mu(\boldsymbol{x}) = \prod_{i=1}^{N} q_i(x_i)\mathrm{d}x_i,\ \text{where } q_i \in L^1(\mathbb{R}),\ q_i \geq 0 \ \text{ a.e. and } \int_{\mathbb{R}^N} \mathrm{d}\mu(\boldsymbol{x}) = 1\}. \tag{2.2}$$

Note that this set can also be written as the Cartesian product of the sets $(\Omega_i)_{i=1,\ldots,N}$

where

$$\Omega_i = \{\mu_i \in \mathcal{M}(\mathbb{R});\ \mathrm{d}\mu_i(\boldsymbol{x}) = q_i(x_i)\mathrm{d}x_i,\ \text{where } q_i \in L^1(\mathbb{R}),\ q_i \geq 0\ \text{ a.e. and } \int_{\mathbb{R}} \mathrm{d}\mu_i(x_i) = 1\}.$$

2.1. Preliminaries. Our purpose in the following is, given a concave functional $F : \widetilde{\mathcal{M}} \to \mathbb{R}$, to define an algorithm which approximates a probability measure μ^{opt} solution of

$$\mu^{opt} = \arg\max_{\mu \in \Omega} F(\mu). \tag{2.3}$$

This problem can be seen as a constrained convex optimization problem in the infinite dimensional Banach space $(\widetilde{\mathcal{M}}, \|.\|_{TV})$. In this framework, most results of optimization are based on duality results, see for instance [22, 9]. In the present paper we consider a gradient-like descent algorithm defined directly on the probability measures set, as in [28].

Let us introduce some notations from [28]. Let $F : \widetilde{\mathcal{M}} \to \mathbb{R}$. As $\widetilde{\mathcal{M}}$ is a Banach space, one can compute the Fréchet derivative of F at $\mu \in \widetilde{\mathcal{M}}$ as the bounded linear functional $\mathrm{d}F_\mu(.) : \widetilde{\mathcal{M}} \to \mathbb{R}$ satisfying

$$F(\mu + \nu) - F(\mu) - \mathrm{d}F_\mu(\nu) = o(\|\nu\|_{TV}), \quad \text{when } \|\nu\| \to 0.$$

We also consider the Gateaux derivative of F:

$$\forall \nu \in \widetilde{\mathcal{M}}, \quad \partial F_\mu(\nu) = \lim_{t \to 0} \frac{F(\mu + t\nu) - F(\mu)}{t}.$$

In some cases, as in the following one can find a function $\mathrm{d}f : \mathcal{M} \times \mathbb{R}^N \to \mathbb{R}$ continuous and upper bounded, such that the Gateaux differential of F can be written as

$$\forall \nu \in \widetilde{\mathcal{M}}, \quad \partial F_\mu(\nu) = \int_{\mathbb{R}^N} \mathrm{d}f(\mu, \boldsymbol{x})\mathrm{d}\nu(\boldsymbol{x}). \tag{2.4}$$

Consider an auxiliary concave functional $G : \widetilde{\mathcal{M}} \to \mathbb{R}$. An important property appearing in the following is that its Fréchet differential is L-Lipschitz on Ω, i.e.

$$\forall (\mu_1, \mu_2) \in \Omega^2,\ \ \forall \nu \in \widetilde{\mathcal{M}} \quad |\mathrm{d}G_{\mu_1}(\nu) - \mathrm{d}G_{\mu_2}(\nu)| \leq L\|\nu\|_{TV}\|\mu_1 - \mu_2\|_{TV}. \tag{2.5}$$

The Lipschitz differential condition of G together with its concavity implies that, see [32] for instance,

$$\forall (\mu_1, \mu_2) \in \Omega^2, \quad 0 \geq G(\mu_1) - G(\mu_2) - \mathrm{d}G_{\mu_2}(\mu_1 - \mu_2) \geq -L\|\mu_1 - \mu_2\|_{TV}^2. \tag{2.6}$$

Furthermore we say that a function F is twice differentiable in the sense of Fréchet at $\mu \in \widetilde{\mathcal{M}}$ if $\mathrm{d}F_\mu$ is differentiable. If it exists, d^2F is a bilinear application from $\widetilde{\mathcal{M}} \times \widetilde{\mathcal{M}}$ to $\mathbb{R}$.

2.2. Statement of the main result. Let us consider as a first step the hypotheses imposed on the functional F in the rest of this part, given by the following definition.

HYPOTHESIS 1. *Let $F : \widetilde{M} \to \mathbb{R}$ be a concave functional. We say that F satisfies Hypothesis 1 if:*

(i) *F can be written as $F = G + H$ where G is a concave L-Lipschitz Fréchet-differentiable functional whereas H corresponds to the entropy of the probability measure, that is the Kullback-Leibler divergence from this measure to the Lebesgue measure.*

(ii) *For $\mu \in \widetilde{\mathcal{M}}$, G is twice differentiable in the sense of Fréchet at μ and the first order derivative of F satisfies Equation* (2.4).

(iii) $\lim_{\|\mu\| \to \infty} F(\mu) = -\infty$.

REMARK 1. *Hypothesis (i) can be replaced by the more restrictive hypothesis that F is L-Lipschitz Fréchet differentiable.*

Our purpose is to construct an iterative algorithm providing a density at each iteration and approximating the solution of (2.3) for a certain class of functionals F. The key principle of our method is given by the Radon-Nikodym Theorem, see [37] for instance. Let $k \geq 0$ be an integer and assume that $\mu^k \in \widetilde{\mathcal{M}}$ is a probability measure absolutely continuous respectively to the Lebesgue measure. We construct $\mu^{k+1} \in \widetilde{\mathcal{M}}$ as a measure which is absolutely continuous with respect to μ^k. In this case, the Radon-Nikodym theorem ensures that this measure should be written as

$$\mathrm{d}\mu^{k+1}(\boldsymbol{x}) = h_k(\boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x}), \tag{2.7}$$

where $h_k \in L^1(\mu^k)$ is a positive function. Our aim is to determine a function $h_k \in L^1(\mu^k)$ which ensures that $F(\mu^{k+1}) \geq F(\mu^k)$, F being a concave functional satisfying Hypothesis 1. Following the classical iterative scheme given by the gradient descent method, we consider h_k as a function of the derivative of F at μ^k and, according to our structure

$$h_k(\boldsymbol{x}) = K_k(\alpha_k) \exp(\alpha_k \mathrm{d}f(\mu^k, \boldsymbol{x})), \tag{2.8}$$

where $\mathrm{d}f$ is defined by (2.4) whereas $\alpha_k > 0$ is the optimal algorithm step-size at iteration k and $K_k(\alpha_k)$ is the normalization constant such that $\int_{\mathbb{R}^N} \mathrm{d}\mu^{k+1}(\boldsymbol{x}) = 1$. We also impose the convention that $h_k(\boldsymbol{x}) = \infty$ when $\exp(\alpha_k \mathrm{d}f(\mu^k, \boldsymbol{x}))$ is not integrable. One can see that as soon as μ^0 is a measure absolutely continuous with respect to the Lebesgue measure, so is each μ^k. This choice of h is motivated by the positive, integrable assumption together with, as mentioned earlier, its coherence with the structure of the gradient descent method. Furthermore the exponential function is the more suitable when dealing with entropy constraint, see [45] for details. This descent algorithm is defined as the "exponentiated gradient" descent in [23]. Since [23] it has been widely studied in the context of machine learning even in the Bayesian framework, see [15] for instance.

The optimization algorithm involved in this paper is the following:

Algorithm 1 Exponentiated Gradient algorithm

1: INITIALIZE($\mu^0 \in \Omega$)
2: **repeat**
3: Compute $\mathrm{d}f(\mu^k, \boldsymbol{x})$
4: Compute $\alpha_k = \arg\max_\alpha K_k(\alpha) \exp(\alpha \mathrm{d}f(\mu^k, .))\mu^k$
5: Compute $\mu^{k+1} = K_k(\alpha_k) \exp(\alpha_k \mathrm{d}f(\mu^k, .))\mu^k$
6: **until** Convergence

Let us now state the main convergence result of this paper.

THEOREM 2.1. *Let F be a concave functional satisfying Hypothesis 1. Let, for every $k \geq 0$, α_k be the optimal stepsize of Algorithm 1. Then the sequence $(\mu^k)_{k\geq 0}$ of elements of $\widetilde{\mathcal{M}}$ given by $\mu^{k+1} = K_k(\alpha_k)\exp(\alpha_k \mathrm{d}f(\mu^k, .))\mu^k$ converges to a maximizer of F on Ω.*

2.3. Proof of Theorem 2.1. Before considering the convergence results, let us introduce some differentiability properties of the entropy function in our context.

Note that as our purpose is to construct measures with a density, thus we consider in the following that for each $k \geq 0$, $\mathrm{d}\mu^k = q^k \mathrm{d}\boldsymbol{x}$. The term H in Hypothesis 1 also corresponds in this case to the entropy $\mathcal{H}$ of the corresponding q^k. Concerning the entropy of density functions, one can notice that it is not differentiable in the sense of Fréchet. However in our case, such as in [13], one can find directions h such that the Gateaux differential of $\mathcal{H}$ at q^k in the direction h can be defined. This is the result stated in the following lemma.

LEMMA 2.2. *Let us assume that $(q^k)_{k\in\mathbb{N}}$ is the sequence of densities corresponding to the measures provided by Algorithm 1. Let $k > 0$ be fixed and define*

$$\mathcal{W}_k := \{h \in L^\infty(q^k \mathrm{d}\mathcal{L}),\ h > -1 \ a.s.\}.$$

Then as soon as $h \in \mathcal{W}_k$, $\mathcal{H}$ is differentiable at q^k in the direction hq^k and

$$\partial\mathcal{H}_{q^k}(hq^k) = -\int (\ln(q^k(\boldsymbol{x})) + 1)h(\boldsymbol{x})q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

Proof.

Following the development made in [13] for simple functions, let us now prove that when $h \in \mathcal{W}_k$, the Gateaux derivative of $\mathcal{H}$ at q^k in the direction hq^k can be defined. Hence we consider

$$\mathcal{H}(q^k+hq^k)-\mathcal{H}(q^k) = -\int_{\mathbb{R}^N} \ln(1+h(\boldsymbol{x}))q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int_{\mathbb{R}^N} \ln(q^k(\boldsymbol{x})+h(\boldsymbol{x})q^k(\boldsymbol{x}))h(\boldsymbol{x})q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x},$$

where the existence of each term is ensured by the fact that $h \in \mathcal{W}_k$.

Let us define

$$\partial\mathcal{H}_{q^k}(hq^k) = -\int (\ln(q^k(\boldsymbol{x})) + 1)h(\boldsymbol{x})q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x},$$

we thus have

$$\begin{aligned}\mathcal{H}(q^k + hq^k) - \mathcal{H}(q^k) - \partial\mathcal{H}_{q^k}(hq^k) &= -\int_{\mathbb{R}^N} \ln(1 + h(\boldsymbol{x}))q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int_{\mathbb{R}^N} \ln(q^k(\boldsymbol{x}))h(\boldsymbol{x})q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \\ &\quad - \int \ln(1 + h(\boldsymbol{x}))h(\boldsymbol{x})q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + \int (\ln(q^k(\boldsymbol{x})) + 1)h(\boldsymbol{x})q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x}) \\ &= -\int_{\mathbb{R}^N} \ln(1 + h(\boldsymbol{x}))(1 + h(\boldsymbol{x}))q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + \int_{\mathbb{R}^N} h(\boldsymbol{x})q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.\end{aligned}$$

In order to conclude, let us now consider that h is such that $\|h\|_{L^\infty} \to 0$. In this case, for almost every $\boldsymbol{x}$ one has $h(\boldsymbol{x}) \to 0$ and $\ln(1 + h(\boldsymbol{x})) \sim h(\boldsymbol{x})$. This induces

$$\mathcal{H}(q^k+hq^k)-\mathcal{H}(q^k)-\partial\mathcal{H}_{q^k}(hq^k) \sim -\int_{\mathbb{R}^N} h(\boldsymbol{x})(1+h(\boldsymbol{x}))q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + \int_{\mathbb{R}^N} h(\boldsymbol{x})q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = -\int_{\mathbb{R}^N} h(\boldsymbol{x})^2 q^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x},$$

and as $\mu^k(\mathrm{d}\boldsymbol{x}) = q^k(\boldsymbol{x})\mathcal{L}(\mathrm{d}\boldsymbol{x})$ is a probability measure, one has

$$\frac{|\mathcal{H}(q^k + hq^k) - \mathcal{H}(q^k) - \partial\mathcal{H}_{q^k}(h)|}{\|h\|_{L^\infty}} \sim \frac{\|h\|^2_{L^2(q^k\mathrm{d}\mathcal{L})}}{\|h\|_{L^\infty(q^k\mathrm{d}\mathcal{L})}} \leq \|\widetilde{h}\|_{L^2(q^k\mathrm{d}\mathcal{L})},$$

which yields the desired result.
□

Concerning the proof of Theorem 2.1, it involves two main steps. As a first step we prove that the sequence $(F(\mu^k))_{k\in\mathbb{N}}$ is an increasing sequence. Secondly, we consider the convergence of the sequence $(\mu^k)_{k\geq 0}$ to a solution of (2.3).

LEMMA 2.3. *Let F be a functional satisfying Hypothesis 1. Let also $(\mu^k)_{k\in\mathbb{N}}$ be the sequence provided by Algorithm 1. Then there exist $\alpha_0 > 0$ such that*

$$\forall \alpha \in (0, \alpha_0) \quad F(\mu^\alpha) - F(\mu^k) \geq 0. \tag{2.9}$$

Proof.

Let $k > 0$ be fixed and μ^k be given. For every $\alpha \geq 0$ we define μ^α as the measure defined for all $\boldsymbol{x} \in \mathbb{R}^N$ by $\mathrm{d}\mu^\alpha(\boldsymbol{x}) = K_k(\alpha)\exp(\alpha \mathrm{d}f(\mu^k, \boldsymbol{x}))\mathrm{d}\mu^k(\boldsymbol{x}) := h_\alpha(\mu^k, \boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x})$.

We define furthermore $g_k(\alpha) := F(\mu^\alpha)$. Thus g_k is a function from $\mathbb{R}^+$ to $\mathbb{R}$ and α_{opt} is an optimal step-size if $g_k(\alpha_{opt}) = \max g_k(\alpha)$, i.e.

$$\alpha_{opt} = \arg\max_\alpha g_k(\alpha). \tag{2.10}$$

According to the point (iii) of Hypothesis 1, $F(\mu) \to -\infty$ when $\|\mu\| \to \infty$ ensures that we can find an α_{opt}, not necessarily unique, such that

$$\forall \alpha > 0, \quad F(\mu^\alpha) \leq F(\mu^{\alpha_{opt}}). \tag{2.11}$$

Let $\alpha > 0$ be given and consider the decomposition given by point (i) of Hypothesis 1. Thanks to Equation (2.6) one has

$$G(\mu^\alpha) - G(\mu^k) \geq \mathrm{d}G_{\mu^k}(\mu^\alpha - \mu^k) - L\|\mu^\alpha - \mu^k\|^2_{TV}. \tag{2.12}$$

Furthermore, as $\mu^\alpha = h_\alpha(\mu^k, .)\mu^k$ and μ^k is a probability measure one can notice that

$$-L\|\mu^\alpha - \mu^k\|^2_{TV} = -L\|h_\alpha(\mu^k, .) - 1\|^2_{L^1(\mu^k)} \geq -L\|h_\alpha(\mu^k, .) - 1\|^2_{L^2(\mu^k)}. \tag{2.13}$$

Furthermore,

$$\begin{aligned} H(\mu^\alpha) &= -\int_{\mathbb{R}^N} \ln\left(\frac{\mathrm{d}\mu^\alpha}{\mathrm{d}\mathcal{L}}\right)\mathrm{d}\mu^\alpha(\boldsymbol{x}) \\ &= H(\mu^k) - \int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k, \boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x}) - \int_{\mathbb{R}^N} \ln\left(\frac{\mathrm{d}\mu^k}{\mathrm{d}\mathcal{L}}\right)(h_\alpha(\mu^k, \boldsymbol{x}) - 1)\mathrm{d}\mu^k(\boldsymbol{x}). \end{aligned}$$

Applying Lemma 2.2 to q^k, the density of μ^k and $h = h_\alpha(\mu^k, .) - 1$ gives

$$\partial H_{\mu^k}(\mu^\alpha - \mu^k) = \int_{\mathbb{R}^N} (-\ln\left(\frac{\mathrm{d}\mu^k}{\mathrm{d}\mathcal{L}}\right) - 1)(h_\alpha(\mu^k, \boldsymbol{x}) - 1)\mathrm{d}\mu^k(\boldsymbol{x}), \tag{2.14}$$

This entails

$$H(\mu^\alpha)-H(\mu^k) \geq \partial H_{\mu^k}(\mu^\alpha-\mu^k)-\int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k,\boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x})-\int_{\mathbb{R}^N}(h_\alpha(\mu^k,\boldsymbol{x})-1)\mathrm{d}\mu^k(\boldsymbol{x}) \tag{2.15}$$

And thus

$$H(\mu^\alpha) - H(\mu^k) - \partial H_{\mu^k}(\mu^\alpha - \mu^k) \geq -\int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k,\boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x}), \tag{2.16}$$

as $\int_{\mathbb{R}^N}(h_\alpha(\mu^k,\boldsymbol{x})-1)\mathrm{d}\mu^k(\boldsymbol{x})=0$.

Finally, from (2.12), (2.13) and (2.16) one has

$$F(\mu^\alpha) - F(\mu^k) \geq \partial F_{\mu^k}(\mu^\alpha - \mu^k) - L\|h_\alpha(\mu^k,.)-1\|^2_{L^2(\mu^k)} - \int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k,\boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x}). \tag{2.17}$$

Finally, $F(\mu^\alpha) - F(\mu^k)$ is positive if the right side of Equation (2.17) is positive. This part being quite technical it is included in Appendix 7.1. □

Lemma 2.3 ensures that for $\alpha > 0$ small enough, $F(\mu^\alpha) \geq F(\mu^k)$. As we choose $\mu^{k+1} = \mu^{\alpha_{opt}}$, where α_{opt} is defined by (2.10), we obviously have $F(\mu^{k+1}) \geq F(\mu^k)$.

Finally the sequence $(F(\mu^k))_{k\in\mathbb{N}}$ is increasing and upper bounded in $\mathbb{R}$, thus convergent. It thus also satisfies that $F(\mu^{k+1}) - F(\mu^k) \to 0$.

In order to conclude we have to show that $(\mu^k)_{k\in\mathbb{N}}$ indeed converges to a maximum of F on Ω. But, for every $k \geq 0$, $\mu^k \in \Omega$, which is a bounded set in $\widetilde{\mathcal{M}}$ and thus in $\mathcal{M}(\mathbb{R}^N)$. Furthermore, the latter is the dual of C_0, the space of continuous functions f that tend to zero at infinity, which is a separable Banach space. As a consequence of the Banach-Alaoglu Theorem, see [37] for instance, there exists $\mu^{lim} \in \widetilde{\mathcal{M}}$ and a subsequence $(\mu^{k_n})_{n\in\mathbb{N}}$ such that for every continuous function that goes to zero at infinity,

$$\int f(\boldsymbol{x})\mu^{k_n}(\mathrm{d}\boldsymbol{x}) \to \int f(\boldsymbol{x})\mu^{lim}(\mathrm{d}\boldsymbol{x}).$$

i.e. when $k \to \infty$, we have $\mu^{k_n} \rightharpoonup^* \mu^{lim} \in \Omega$.
$(\mu^{k_n})_{n\in\mathbb{N}}$ by critical

From Lemma 2.3 we know that

$$F(\mu^{k+1}) = g_k(\alpha_{opt}) \geq g_k(\alpha), \quad \forall \alpha > 0.$$

However the analytic form of α_{opt} is not tractable. We thus approximate it by a calculable α_{subopt}, not necessarily smaller than α_0. In order to determine this α_{subopt} let us determine the differentiability properties of the real valued function $g(\alpha) = F(\mu^\alpha)$. From the definition of μ^α, it has a density with respect to q^k which belongs to $\mathcal{W}_k$. As G is assumed to be differentiable, so is F. Furthermore, G is also assumed to be twice differentiable in the sense of Fréchet and from Equation (2.14), so is H in the direction of h_α. Hence g_k is twice differentiable and the Taylor-Young formula gives, for α small enough,

$$g_k(\alpha) = g_k(0) + \alpha g_k'(0) + \frac{\alpha^2}{2} g_k''(0) + \alpha^2\varepsilon(\alpha) := \varphi_k(\alpha) + \alpha^2\varepsilon(\alpha), \tag{2.18}$$

where $\varepsilon(\alpha) \to 0$ when $\alpha \to 0$.

The critical point of $\varphi_k(\alpha)$ is $\alpha_{subopt} = -\frac{g_k'(0)}{g_k''(0)}$, as soon as $g_k''(0) \neq 0$ exists, which gives in (2.18):

$$g_k(\alpha_{subopt}) = g_k(0) - \frac{g_k'(0)^2}{2g_k''(0)} + \alpha_{subopt}^2 \varepsilon(\alpha_{subopt}), \tag{2.19}$$

and by construction of μ^{k+1},

$$F(\mu^{k+1}) \geq g_k(\alpha_{subopt}) = g_k(0) - \frac{g_k'(0)^2}{2g_k''(0)} + \alpha_{subopt}^2 \varepsilon(\alpha_{subopt}) = F(\mu^k) - \frac{g_k'(0)^2}{2g_k''(0)} + \alpha_{subopt}^2 \varepsilon(\alpha_{subopt}).$$

As $F(\mu^{k+1}) - F(\mu^k) \to 0$, obviously $\lim_{k\to\infty} \frac{g_k'(0)^2}{2g_k''(0)} = 0$. Let us now consider a convergent subsequence $(k_n)_{n\in\mathbb{N}}$ and denote by $(\gamma_{k_n})_{n\in\mathbb{N}}$ the sequence defined $\forall n \in \mathbb{N}$ by $\gamma_{k_n} = -\frac{g_{k_n}'(0)}{g_{k_n}''(0)}$, that is the sequence of suboptimal stepsizes. We have then $-\frac{g_{k_n}'(0)^2}{g_{k_n}''(0)} = g_{k_n}'(0)\gamma_{k_n} \to 0$. As $\mathrm{d}f$ is supposed to be continuous in Equation (2.4), the sequence $(g_{k_n}'(0))_{n\in\mathbb{N}}$ is convergent and either $\gamma_{k_n} \to 0$ or $g_{k_n}'(0) \to 0$. Let us assume that $\gamma_{k_n} \to 0$ and that $g_{k_n}'(0) \to l \neq 0$. As $\gamma_{k_n} \to 0$ we have, for n large enough,

$$\begin{aligned} g_{k_n}(\gamma_{k_n}) - g_{k_n}(0) &= \gamma_{k_n} g_{k_n}'(0) + \frac{\gamma_{k_n}^2}{2} g_{k_n}''(0) + \gamma_{k_n}^2 \varepsilon(\gamma_{k_n}) \\ &= \frac{1}{2}\gamma_{k_n} g_{k_n}'(0) + \gamma_{k_n}^2 \varepsilon(\gamma_{k_n}). \end{aligned}$$

Hence,

$$\frac{g_{k_n}(\gamma_{k_n}) - g_{k_n}(0)}{\gamma_{k_n}} = \frac{1}{2} g_{k_n}'(0) + \gamma_{k_n} \varepsilon(\gamma_{k_n}), \tag{2.20}$$

and when n tends to infinity γ_{k_n} tends to zero and taking limits in (2.20) one obtains

$$l = \frac{l}{2},$$

which is impossible. Hence, $g_{k_n}'(0) \to 0$ when $n \to \infty$.

Let us determine the derivative of the different functions g_k. For this purpose, we have to determine the derivative of the function $\widetilde{g} : \alpha \mapsto h_\alpha(\mu^k, .)$. As $h_\alpha(\mu^k, \boldsymbol{x}) = K_k(\alpha) e^{\alpha \mathrm{d}f(\mu^k, \boldsymbol{x})}$, its derivative is given by

$$\forall \boldsymbol{x} \in \mathbb{R}^N, \quad \frac{\partial \widetilde{g}}{\partial \alpha}(\alpha, \boldsymbol{x}) = \frac{\partial K_k}{\partial \alpha}(\alpha) e^{\alpha \mathrm{d}f(\mu^k, \boldsymbol{x})} + \mathrm{d}f(\mu^k, \boldsymbol{x}) K_k(\alpha) e^{\alpha \mathrm{d}f(\mu^k, \boldsymbol{x})}. \tag{2.21}$$

As α is supposed to be close to zero, one can assume that it is not greater than one and the Lebesgue's Theorem thus allows to invert differentiation and integration in the evaluation of $\frac{\partial K_k}{\partial \alpha}$. As

$$K_k(\alpha) = \frac{1}{\int e^{\alpha \mathrm{d}f(\mu^k, \boldsymbol{x})} \mathrm{d}\mu^k(\boldsymbol{x})}, \tag{2.22}$$

we have

$$\frac{\partial K_k}{\partial \alpha}(\alpha) = -\frac{\int \mathrm{d}f(\mu^k, \boldsymbol{x}) e^{\alpha \mathrm{d}f(\mu^k, \boldsymbol{x})} \mathrm{d}\mu^k(\boldsymbol{x})}{\left(\int e^{\alpha \mathrm{d}f(\mu^k, \boldsymbol{x})} \mathrm{d}\mu^k(\boldsymbol{x})\right)^2}$$
$$= -K_k(\alpha) \int \mathrm{d}f(\mu^k, \boldsymbol{x}) h_\alpha(\mu^k, \boldsymbol{x}) \mathrm{d}\mu^k(\boldsymbol{x}) = -K_k(\alpha) \int \mathrm{d}f(\mu^k, \boldsymbol{x}) \mathrm{d}\mu^\alpha(\boldsymbol{x}). \tag{2.23}$$

Combining (2.21) and (2.23) yields

$$\forall \boldsymbol{x} \in \mathbb{R}^N, \quad \frac{\partial \widetilde{g}}{\partial \alpha}(\alpha, \boldsymbol{x}) \mathrm{d}\mu^k(\boldsymbol{x}) = \mathrm{d}\mu^\alpha(\boldsymbol{x}) \left(\mathrm{d}f(\mu^k, \boldsymbol{x}) - \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{y}) \mathrm{d}\mu^\alpha(\boldsymbol{y}) \right). \tag{2.24}$$

And

$$g_k'(\alpha) = \partial F_{\mu^\alpha}\left(\frac{\partial \widetilde{g}}{\partial \alpha} \mu^k\right) = \int_{\mathbb{R}^N} \mathrm{d}f(\mu^\alpha, \boldsymbol{x}) \left(\mathrm{d}f(\mu^k, \boldsymbol{x}) - \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{y}) \mathrm{d}\mu^\alpha(\boldsymbol{y}) \right) \mathrm{d}\mu^\alpha(\boldsymbol{x}),$$

which leads to

$$g_k'(0) = \partial F_{\mu^k}(\widetilde{g}'(0)\mu^k) = \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{x}) \left(\mathrm{d}f(\mu^k, \boldsymbol{x}) - \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{y}) \mathrm{d}\mu^k(\boldsymbol{y}) \right) \mathrm{d}\mu^k(\boldsymbol{x})$$
$$= \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{x})^2 \mu^k(\mathrm{d}\boldsymbol{x}) - \left(\int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{y}) \mathrm{d}\mu^k(\boldsymbol{y}) \right)^2 \tag{2.25}$$

Hence, for n large enough,

$$\|\mathrm{d}f(\mu^{k_n}, .)\|^2_{L^2(\mu^{k_n})} - \|\mathrm{d}f(\mu^{k_n}, .)\|^2_{L^1(\mu^{k_n})} \to 0,$$

and $\mathrm{d}f(\mu^{k_n}, .)$ converges to a constant λ, independent of $\boldsymbol{x} \in \mathbb{R}^N$.

Let ν be any element of Ω, then $\partial F_{\mu^{lim}}(\nu - \mu^{lim}) = 0$, and from concavity of F we have

$$F(\nu) \leq F(\mu^{lim}) \qquad \forall \nu \in \Omega.$$

Which means that $F(\mu^{lim})$ is a maximum of F over Ω.

In the present part we have presented the convergence properties of our algorithm, which is well adapted to the space of probability measures. Our interest in the following is also in its application in the context of variational Bayesian methodology. For the sake of completeness, let us recall this methodology introduced in [25].

3. Application to variational Bayesian methodology.

3.1. Classical variational Bayesian algorithm. For the sake of simplicity we consider in the following density functions q instead of the corresponding measures μ. We denote by $\boldsymbol{y} \in \mathbb{R}^M$ the M dimensional vector containing the data information whereas $\boldsymbol{w} \in \mathbb{R}^N$ represents the vector to be estimated, which is considered as a realization of a random vector $\boldsymbol{W}$. We also denote by p the prior probability density function (p.d.f.) of $\boldsymbol{W}$. The Bayes rule entails that this prior distribution is

closely related to the posterior one, $p(\boldsymbol{w}|\boldsymbol{y})$, up to a normalization constant. Even in simple cases this posterior may not be separable. Hence, in the variational Bayesian framework, we approximate it by a separable probability density

$$q(\boldsymbol{w}) = \prod_i q_i(w_i). \tag{3.1}$$

Taking separable laws obviously simplifies the problem even if it introduces some approximation errors.

Therefore, the purpose of variational Bayesian methodology is to determine a separable probability density function q close to the true posterior in the sense defined by the Kullback-Leibler divergence see [42] for instance.

The optimal approximating density q is then given by

$$\forall i \in \{1, \ldots, N\}, \quad q_i(w_i) = K_k^i \exp\left(< \ln p(\boldsymbol{y}, \boldsymbol{w}) >_{\prod_{j \neq i} q_j} \right), \tag{3.2}$$

where K_k^i is the normalization constant and $< \ln p(\boldsymbol{y}, \boldsymbol{w}) >_{\prod_{j \neq i} q_j} = \int_{\mathbb{R}^{N-1}} \ln p(\boldsymbol{y}, \boldsymbol{w}) \prod_{j \neq i} q_j(w_j) \mathrm{d}w_j$ is the mean of $\ln p(\boldsymbol{y}, \boldsymbol{w})$ under the probability $\prod_{j \neq i} q_j$.

Although this solution is obtained analytically, Equation (3.2) clearly does not have an explicit form. In order to have implementable methods, several approaches can be considered. The first is to impose conjugate prior distributions which ensure that the posterior distribution belongs to a given family and to reduce the optimization problem to an approximation of its parameters. A different approach, introduced in [39] is to consider approximating law in a given family, namely the Gaussian one, which does not necessarily contains the posterior distribution. These approaches also reduce the functional optimization problem to a parametric one. However even in these cases the intricate form of (3.2) imposes a numerical approximation of the solution. A natural method in this case is to consider alternate descent algorithms. However this method is time consuming, each update step needing N iterations, one for each component q_i. This drawback can be diminished by relaxing the separability assumption and considering instead a descent by groups of coordinates. In this case the main drawback is given by the covariance matrix of each group which needs to be stored and inverted. Several techniques, such as the numerical preprocessing of this matrix in [4] have thus been considered to overcome this drawback. In [39] an alternative way is developed, based on concave-convex programming. But for very large dimensional problems these methods remain inefficient in general, as they need several intricate approximations. In the present work we consider instead the application of the algorithm defined in Section 5.1.1 to the functional optimization problem induced by the Bayesian variational problem.

3.2. Variational Bayesian Exponentiated Gradient Algorithm. In this section we define an iterative method which allows to compute efficiently at each iteration each q_i independently of the other ones.

A first step is to rewrite the minimization problem as a convex optimization problem independent of the posterior distribution to be approximated. Instead of minimizing the Kullback-Leibler divergence, we thus remark, as in [10], that

$$\ln p(\boldsymbol{y}) = \ln \frac{p(\boldsymbol{y}, \boldsymbol{w})}{p(\boldsymbol{w}|\boldsymbol{y})}, \tag{3.3}$$

where $\boldsymbol{w}$ is the vector of hidden variables and parameters.

As the log-likelihood $\ln p(\boldsymbol{y})$ in (3.3) does not depends on $\boldsymbol{w}$ one can write

$$\ln p(\boldsymbol{y}) = \mathcal{F}(q) + \mathcal{KL}[q||p(.|\boldsymbol{y}))].$$

In this case,

$$\mathcal{F}(q) = \int_{\mathbb{R}^N} q(\boldsymbol{w}) \ln \left(\frac{p(\boldsymbol{y}, \boldsymbol{w})}{q(\boldsymbol{w})} \right) \mathrm{d}\boldsymbol{w}, \tag{3.4}$$

is the negative free energy. Thus, minimizing the Kullback-Leibler divergence is obviously equivalent to maximizing this negative free entropy.

Therefore, in the following we consider the problem of maximizing

$$\begin{aligned} \mathcal{F}(q) &= \int_{\mathbb{R}^N} \ln p(\boldsymbol{y}, \boldsymbol{w}) q(\boldsymbol{w}) \mathrm{d}\boldsymbol{w} - \int_{\mathbb{R}^N} \ln(q(\boldsymbol{w})) q(\boldsymbol{w}) \mathrm{d}\boldsymbol{w} \\ &= \langle \ln p(\boldsymbol{y}, .) \rangle_q + \mathcal{H}(q), \end{aligned} \tag{3.5}$$

where

$$\mathcal{H}(q) = - \int_{\mathbb{R}^N} \ln(q(\boldsymbol{w})) q(\boldsymbol{w}) \mathrm{d}\boldsymbol{w},$$

is the entropy of q. The main advantage of this approach is that the objective functional does not depend on the true posterior anymore but only on the joint distribution $p(\boldsymbol{y}, .)$, which is more easily tractable.

One can also notice that the problem of finding

$$q^{opt} = \underset{q \, \text{separable} \; p.d.f.}{\arg\max} \mathcal{F}(q) \tag{3.6}$$

is equivalent to the problem of finding

$$\mu^{opt} = \underset{\mu \in \Omega}{\arg\max} \, F(\mu). \tag{3.7}$$

Where the functional F is defined $F(\mu) = \mathcal{F}(q)$, as soon as $\mu \in \Omega$ is such that q is the density of μ. Let us also define in the following ∂f by $\forall \boldsymbol{x} \in \mathbb{R}^N$, $\partial f(q, \boldsymbol{x}) = \mathrm{d} f(q\mathcal{L}, \boldsymbol{x})$.

A classical method in this context is to consider each density q as a $L^1(\mathbb{R}^N)$ function and to apply classical optimization algorithms. In the present framework, taking the non-negativity and the total mass assumptions into account, the algorithm involved is given by the projected gradient method which gives:

$$\forall \boldsymbol{w} \in \mathbb{R}^N \qquad q^{k+1}(\boldsymbol{w}) = P_\Theta(q^k(\boldsymbol{w}) + \rho^k \partial f(q^k, \boldsymbol{w})), \tag{3.8}$$

where P_Θ is the projector operator on the subspace $\Theta = \{q \in L^1(\mathbb{R}^N);\ q(\boldsymbol{w}) \geq 0 \text{ and } \|q\|_{L^1} = 1\}$ and ∂f denotes the Fréchet differential of F at q^k. However, this algorithm requires that $\partial f(q^k, \boldsymbol{w}) \in L^1(\mathbb{R}^N)$ which is not the case in general.

Therefore in the following, we rather apply Theorem 2.1 and the algorithm introduced in Section 2 to the Variational Bayesian framework of Section 3.1. One can

easily see that the function F can be written as an entropy term added to a differentiable function $G = \langle \ln p(\boldsymbol{y}, .) \rangle_q$. As $\ln p(\boldsymbol{y}, .)$ does not depend on q, this function is clearly Lipschitz differentiable respectively to q and twice differentiable in the sense of Fréchet. Hence F satisfies Hypothesis 1.

We consider

$$\mathcal{F}(q) = \langle \ln p(\boldsymbol{y}, .) \rangle_q + \mathcal{H}(q).$$

In this case, the Gateaux differential of $F(\mu) = \mathcal{F}(q)$ at $\mu \in \Omega$ separable is given by $\mathrm{d}F_\mu(\nu) = \sum_i \int_{\mathbb{R}^N} \mathrm{d}f(\mu_i, x_i)\nu_i(\mathrm{d}x)$ with $\mathrm{d}f(\mu_i, x_i) = \partial f(q_i, x_i)$ such that

$$\forall i \in \{1, \ldots, N\}, \quad \forall \boldsymbol{w} \in \mathbb{R}^N, \quad \partial f(q_i, w_i) = \langle \ln p(\boldsymbol{y}, \boldsymbol{w}) \rangle_{\prod_{j \neq i} q_j} - \ln q_i(w_i) - 1.$$

Let $k \geq 0$ be given and q^k be constructed. Following the scheme defined by Algorithm 1 and Equation (2.8), at the following iteration we consider q^α given, for $\alpha > 0$, by

$$\forall \boldsymbol{w} \in \mathbb{R}^N, \quad q^{\alpha_k}(\boldsymbol{w}) = K_k(\alpha_k) q^k \exp\left[\alpha_k \partial f(q^k, \boldsymbol{w})\right] \tag{3.9}$$

$$= \tilde{K}_k(\alpha_k) q^k(\boldsymbol{w}) \left(\prod_i \frac{\exp\left(< \ln p(\boldsymbol{y}, \boldsymbol{w}) >_{\prod_{j \neq i} q_j^k(w_j)} \right)}{q_i^k(w_i)} \right)^{\alpha_k}$$

$$= \tilde{K}_k(\alpha_k) q^k(\boldsymbol{w}) \left(\prod_i \frac{q_i^r(w_i)}{q_i^k(w_i)} \right)^{\alpha_k} \tag{3.10}$$

where $\tilde{K}_k(\alpha_k)$ is the normalization constant and q^r is an intermediate function defined by

$$\forall i \in \{1, \ldots, N\}, \quad q_i^r(w_i) = \exp\left(< \ln p(\boldsymbol{y}, \boldsymbol{w}) >_{\prod_{j \neq i} q_j^k(w_j)} \right)$$

The main challenge is to determine the value of $\alpha_k > 0$. This optimal value α_{opt} should satisfy $g_k'(\alpha_{opt}) = 0$. However, this quantity is hardly tractable in practice. Therefore, we consider instead the suboptimal value given by

$$\alpha_{subopt} = -\frac{g_k'(0)}{g_k''(0)}, \tag{3.11}$$

when $g_k''(0) \neq 0$. This leads to the main algorithm of this paper.

Algorithm 2 Variational Bayesian Exponentiated Gradient Like Algorithm

1: INITIALIZE($q^0 \in \Theta$)
2: **repeat**
3: **function** ITERATION(Compute $q^{k+1} = K_k q^k \exp\left[\alpha_k \partial f(q^k, \boldsymbol{w})\right]$)
4: Compute $q_i^r(w_i) = \exp\left(< \ln p(\boldsymbol{y}, \boldsymbol{w}) >_{\prod_{j \neq i} q_j^k(w_j)} \right)$ for every $i = 1, \ldots, N$
5: Compute $q^\alpha(\boldsymbol{w}) = \tilde{K}_k(\alpha) q^k(\boldsymbol{w}) \left(\frac{q^r(\boldsymbol{w})}{q^k(\boldsymbol{w})} \right)^\alpha$.
6: Compute $\alpha_{subopt} = -\frac{g_k'(0)}{g_k''(0)}$
7: Take $q^{k+1} = q^{\alpha_{subopt}}$.
8: **end function**
9: **until** Convergence

4. Application to linear inverse problems.

4.1. Statement of the problem. The next part of this paper presents the application of Algorithm 2 to linear inverse ill-posed problems. The model of observations chosen in the following is given by

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{b}, \tag{4.1}$$

where $\boldsymbol{y} \in \mathbb{R}^M$ is the vector of observations given as a linear function of the unknown vector $\boldsymbol{x} \in \mathbb{R}^N$ to be estimated. Here, $\boldsymbol{b} \in \mathbb{R}^M$ is the noise vector whereas $\boldsymbol{H}$ is a matrix in $M_{N\times M}$. We also suppose that $\boldsymbol{x}$ is a realization of a random vector $\mathbf{X}$.

In the following we stand in a white noise model which induces that the noise is supposed to be an iid Gaussian vector $\mathcal{N}(0, \sigma_b^2\mathbf{I})$. The corresponding likelihood is

$$p(\boldsymbol{y}|\boldsymbol{x}) = (2\pi\sigma_b^2)^{-M/2} \exp\left[-\frac{\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2}{2\sigma_b^2}\right]. \tag{4.2}$$

Concerning the prior distribution we choose to take sparsity into account by considering $\mathbf{X}$ distributed following a separable heavy tailed distribution. The most general case is given by Gaussian Vector Scale Mixture (GVSM) defined in [19]. In this case, for $i = 1, \ldots, N$, we suppose that $X_i \sim U_i/\sqrt{Z_i}$ where $\mathbf{U} \sim \mathcal{N}(0, \sigma_s^2\mathbf{I})$, $\mathbf{Z} = \prod Z_i$ is a positive random vector of independent positive coordinates and $\mathbf{U}$ and $\mathbf{Z}$ are independents. As a consequence the density of $\mathbf{X}$ is given in an integral form as

$$\forall i \in \{1, \ldots, N\}, \qquad p(x_i) = \int_{\mathbb{R}} \frac{\sqrt{z_i}}{(2\pi)^{1/2}\sigma_s} e^{-\frac{z_i x_i^2}{2\sigma_s^2}} \phi_{z_i}(z_i) dz_i.$$

Note that in the definition, for the sake of simplicity, we consider $\mathbf{Z}$ as a precision parameter and not a dilatation one. Gaussian Vector Scale Mixture forms a large class of nongaussian random variables recently developed as a model of wavelet coefficients of natural images, see [46]. The main interest of this model is, by solving an extended problem due to the presence of a hidden random vector $\mathbf{Z}$, to allow the use of Bayesian hierarchic approaches. The Gaussian Scale Mixture family offers a large class of random variables including Gaussian mixture, when $\mathbf{Z} \sim Z\mathbf{I}$ a discrete random vector or Student laws if the Z_i correspond to Gamma random variables. With different hypothesis on the distribution of $\mathbf{Z}$ one can also define Generalized Gaussian distributions or α-stable ones, see [46]. Indeed GSM offers a simple representation of a large class of nongaussian probability distributions, which justify the increasing interest in this model.

In our context, we choose to consider $\mathbf{Z}$ as a independent Gamma random vector, i.e. for $i = 1, \ldots, N$, we have $Z_i \sim \mathcal{G}(\tilde{a}_i, \tilde{b}_i)$ and

$$\forall i \in \{1, \ldots, N\}, \qquad p(x_i) = \frac{\tilde{b}_i^{\tilde{a}_i}}{\Gamma(\tilde{a}_i)} \int_{\mathbb{R}} \frac{\sqrt{z_i}}{(2\pi)^{1/2}\sigma_s} e^{-\frac{z_i x_i^2}{2\sigma_s^2}} z_i^{\tilde{a}_i - 1} e^{-z_i \tilde{b}_i} dz_i. \tag{4.3}$$

For $\tilde{a}_i = \tilde{b}_i = \frac{\nu}{2}$ the p.d.f. of $\mathbf{X}$ corresponds to a Student-t distribution, as in the model used in [8]. This model of $\boldsymbol{Z}$ ensures that $\boldsymbol{X}$ satisfies the conjugate priors condition.

One can easily check that when the prior information is given by (4.3), Equation (4.2) gives the following posterior distribution

$$p(\boldsymbol{x},\boldsymbol{z}|\boldsymbol{y}) \propto \sigma_b^{-M} \exp\left[-\frac{\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{x}\|^2}{2\sigma_b^2}\right] \prod_{i=1}^{N} \frac{\sqrt{z_i}}{\sigma_s} \exp\left[-\frac{z_i x_i^2}{2\sigma_s^2}\right] \frac{\tilde{b}_i^{\tilde{a}_i} z_i^{\tilde{a}_i-1} e^{-z_i\tilde{b}_i}}{\Gamma(\tilde{a}_i)}. \quad (4.4)$$

Considering that we do not know the constants involved and that the mixing matrix $\boldsymbol{H}$ is high dimensional, this posterior distribution cannot be evaluated directly.

4.2. Numerical implementation. The aim of variational Bayesian methodology and therefore of our method in the context established in Part 4 is the approximation of the posterior p.d.f given by (4.4) by a separable one $q(\boldsymbol{x},\boldsymbol{z}) = \prod_i q_i(x_i) \prod_j \tilde{q}_j(z_j)$. As we have chosen conjugate priors for $\boldsymbol{X}$ and $\boldsymbol{Z}$, the optimum approximating distribution of $\boldsymbol{X}$ is known to belong to a Gaussian family, whereas the p.d.f. of $\boldsymbol{Z}$ belongs to a Gamma one.

$$q^k \sim \prod_i \mathcal{N}(\boldsymbol{m}_k(i), \boldsymbol{\sigma}_k^2(i))$$
$$\tilde{q}^k \sim \prod_j \mathcal{G}(a_k(j), b_k(j))$$

Hence at the initialization stage, we consider

$$q^0 = \mathcal{N}(\boldsymbol{m}_0, \mathrm{Diag}(\boldsymbol{\sigma}_0^2))$$
$$\tilde{q}^0 = \prod_j \mathcal{G}(a_0(j), b_0(j))$$

where $\mathrm{Diag}(\boldsymbol{v})$ is a diagonal matrix with $\boldsymbol{v}$ on its diagonal, and $\boldsymbol{\sigma}_0^2 \in \mathbb{R}^N$ is the vector of initial variances.

Our minimization problem can be analyzed following the alternate iterative scheme:

$$\tilde{q}^{k+1} = \arg\max_{\tilde{q}} \mathcal{F}\big(q^k \tilde{q}\big)$$
$$q^{k+1} = \arg\max_{q} \mathcal{F}\big(q \tilde{q}^{k+1}\big)$$

4.2.1. Approximation of $\tilde{q}$. One can see in Equation (4.4) that the conditional posterior $p(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})$ is fully separable. In this case the classical Bayesian variational approach is efficient enough to be implemented directly. Hence all the $\tilde{q}_i^{k+1}$ can be computed simultaneously, knowing only q^k. Thanks to the classical variational Bayesian approach, [25], described in Section 3, we deduce $\tilde{q}^{k+1}$ thanks to Equation (3.2) and Equation (4.4), for every $i = 1, \ldots, N$

$$\tilde{q}_i^{k+1}(z_i) \propto \exp\Big(< \ln p(\boldsymbol{y},\boldsymbol{x},\boldsymbol{z}) >_{\prod_{j\neq i} \tilde{q}_j^k(z_j) q^k(\boldsymbol{x})}\Big) \quad (4.5)$$
$$\propto \exp\left((\tilde{a}_i - \frac{1}{2})\ln(z_i) - \int (\frac{x_i^2 z_i}{2\sigma_s^2} + z_i\tilde{b}_i) \prod_l q_l^k(x_l) \prod_{j\neq i} \tilde{q}_j^k(z_j) \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{z}\right)$$
$$\propto \exp\left((\tilde{a}_i - \frac{1}{2})\ln(z_i) - z_i\tilde{b}_i - \int \frac{x_i^2 z_i}{2\sigma_s^2} q_i^k(x_i)\mathrm{d}x_i\right)$$
$$\propto \exp\left((\tilde{a}_i - \frac{1}{2})\ln(z_i) - z_i\tilde{b}_i - \frac{(\boldsymbol{\sigma}_k^2(i) + \boldsymbol{m}_k^2(i))z_i}{2\sigma_s^2}\right)$$
$$\propto z_i^{\tilde{a}_i - \frac{1}{2}} \exp\left(-z_i\Big[\tilde{b}_i + \frac{(\boldsymbol{\sigma}_k^2(i) + \boldsymbol{m}_k^2(i))}{2\sigma_s^2}\Big]\right) \quad (4.6)$$

This entails that $\tilde{q}_i^{k+1}$ corresponds to a Gamma p.d.f. of parameters:

$$\forall i \in \{1, \dots, N\}, \qquad \boldsymbol{a}_{k+1}(i) = \tilde{a}_i + \frac{1}{2}, \tag{4.7}$$

$$\boldsymbol{b}_{k+1}(i) = \frac{\boldsymbol{m}_k^2(i) + \boldsymbol{\sigma}_k^2(i)}{2\sigma_s^2} + \tilde{b}_i. \tag{4.8}$$

4.2.2. Approximation of q by Algorithm 2. Let us assume that at the initialization stage, q^0 is a Gaussian p.d.f. with mean $\boldsymbol{m}_0$ and covariance matrix $\mathrm{Diag}(\boldsymbol{\sigma}_0^2)$. At each iteration $k+1$ we determine the approximation of q^{k+1} thanks to our method. Let us define an auxiliary function q^r by $q_i^r(x_i) = \exp\left(\langle \ln p(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{z})\rangle_{\prod_{j\neq i} q_j^k(x_j) q^{k+1}(\boldsymbol{z})}\right)$, thus $\forall i \in \{1, \dots, N\}$,

$$q_i^r(x_i) = \exp\left(< \ln p(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{z}) >_{\prod_{j\neq i} q_j^k(x_j)\tilde{q}^{k+1}(\boldsymbol{z})}\right) \tag{4.9}$$

$$\begin{aligned}
&\propto \exp\left(-\int \left(\frac{\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2}{2\sigma_b^2} + \frac{x_i^2 z_i}{2\sigma_s^2}\right) \prod_{j\neq i} q_j^k(x_j)\tilde{q}^{k+1}(\boldsymbol{z}) \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{z}\right) \\
&\propto \exp\left[-\int \left(\frac{\boldsymbol{x}^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{H}^T\boldsymbol{y}}{2\sigma_b^2} + \frac{x_i^2 z_i}{2\sigma_s^2}\right) \prod_{j\neq i} q_j^k(x_j)\tilde{q}^{k+1}(z_i) \mathrm{d}\boldsymbol{x}\mathrm{d}z_i\right] \\
&\propto \exp\left[-\frac{1}{2\sigma_b^2}\Big(x_i^2 \mathrm{diag}(\boldsymbol{H}^T\boldsymbol{H})_i - 2x_i(\boldsymbol{H}^T\boldsymbol{y})_i + 2x_i(\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{m}_k)_i \right. \\
&\quad \left. -2x_i \mathrm{diag}(\boldsymbol{H}^T\boldsymbol{H}))_i \boldsymbol{m}_k(i)\Big) + \frac{x_i^2 \boldsymbol{a}_{k+1}(i)}{2\sigma_s^2 \boldsymbol{b}_{k+1}(i)}\right]
\end{aligned} \tag{4.10}$$

where $\mathrm{diag}(A)$ is the vector composed by the diagonal entries of A. Note that q^r corresponds, up to the normalization term, to the density of a Gaussian distribution with mean $\boldsymbol{m}_r$ and variance $\boldsymbol{\sigma}_r^2$, where, for every $i = 1, \dots, N$,

$$\boldsymbol{\sigma}_r^2(i) = \left(\frac{\mathrm{diag}(\boldsymbol{H}^T\boldsymbol{H})_i}{\sigma_b^2} + \frac{\boldsymbol{a}_{k+1}(i)}{\boldsymbol{b}_{k+1}(i)\sigma_s^2}\right)^{-1} \tag{4.11}$$

and

$$\boldsymbol{m}_r(i) = \boldsymbol{\sigma}_r^2(i) \times \left(\frac{\boldsymbol{H}^T\boldsymbol{y} - (\boldsymbol{H}^T\boldsymbol{H} - \mathrm{diag}(\boldsymbol{H}^T\boldsymbol{H}))\boldsymbol{m}_k}{\sigma_b^2}\right)_i \tag{4.12}$$

Therefore, by Equation (3.10), we have for every $i = 1, \dots, N$,

$$
\begin{aligned}
q_i^\alpha(x_i) &= K_k(\alpha) q_i^k(x_i) \left(\frac{q_i^r(x_i)}{q_i^k(x_i)}\right)^\alpha \\
&= \sqrt{\frac{\boldsymbol{\sigma}_k^2(i)}{\boldsymbol{\sigma}_r^2(i)}}^\alpha \; K_k(\alpha) \exp\left[-\frac{(x_i - \boldsymbol{m}_k(i))^2}{2\boldsymbol{\sigma}_k^2(i)}\right] \exp\left[-\alpha \frac{x_i^2(\boldsymbol{\sigma}_k^2(i) - \boldsymbol{\sigma}_r^2(i))}{2\boldsymbol{\sigma}_r^2(i)\boldsymbol{\sigma}_k^2(i)}\right] \\
&\times \exp\left[-\alpha \frac{-2x_i(\boldsymbol{m}_r(i)\boldsymbol{\sigma}_k^2(i) - \boldsymbol{m}_k(i)\boldsymbol{\sigma}_r^2(i)) + \boldsymbol{m}_r(i)^2\boldsymbol{\sigma}_k^2(i) - \boldsymbol{m}_k(i)^2\boldsymbol{\sigma}_r^2(i)}{2\boldsymbol{\sigma}_r^2(i)\boldsymbol{\sigma}_k^2(i)}\right] \\
&= \sqrt{\frac{\boldsymbol{\sigma}_k^2(i)}{\boldsymbol{\sigma}_r^2(i)}}^\alpha \; K_k(\alpha) \exp\left[-\frac{1}{2}\left(x_i^2 \frac{\boldsymbol{\sigma}_r^2(i) + \alpha(\boldsymbol{\sigma}_k^2(i) - \boldsymbol{\sigma}_r^2(i))}{\boldsymbol{\sigma}_r^2(i)\boldsymbol{\sigma}_k^2(i)}\right)\right] \\
&\times \exp\left[-\frac{1}{2}\left(-2x_i \frac{\boldsymbol{m}_k(i)\boldsymbol{\sigma}_r^2(i) + \alpha(\boldsymbol{m}_r(i)\boldsymbol{\sigma}_k^2(i) - \boldsymbol{m}_k(i)\boldsymbol{\sigma}_r^2(i))}{\boldsymbol{\sigma}_r^2(i)\boldsymbol{\sigma}_k^2(i)} + t(\alpha)\right)\right]
\end{aligned}
$$

where q^α is defined in Section 5.1.1, and $t(\alpha) = \alpha \frac{\boldsymbol{m}_r(i)^2\boldsymbol{\sigma}_k^2(i) - \boldsymbol{m}_k(i)^2\boldsymbol{\sigma}_r^2(i)}{2\boldsymbol{\sigma}_r^2(i)\boldsymbol{\sigma}_k^2(i)}$ is a constant. Finally, q_i^α still corresponds to a Gaussian p.d.f. with parameters $\boldsymbol{m}_\alpha$ and $\mathrm{Diag}(\boldsymbol{\sigma}_\alpha^2)$ satisfying:

$$
\boldsymbol{\sigma}_\alpha^2(i) = \frac{\boldsymbol{\sigma}_r^2(i)\boldsymbol{\sigma}_k^2(i)}{\boldsymbol{\sigma}_r^2(i) + \alpha(\boldsymbol{\sigma}_k^2(i) - \boldsymbol{\sigma}_r^2(i))} \tag{4.13}
$$

$$
\boldsymbol{m}_\alpha(i) = \frac{\boldsymbol{m}_k(i)\boldsymbol{\sigma}_r^2(i) + \alpha(\boldsymbol{m}_r(i)\boldsymbol{\sigma}_k^2(i) - \boldsymbol{m}_k(i)\boldsymbol{\sigma}_r^2(i))}{\boldsymbol{\sigma}_r^2(i) + \alpha(\boldsymbol{\sigma}_k^2(i) - \boldsymbol{\sigma}_r^2(i))}. \tag{4.14}
$$

In order to construct q^{k+1} we choose in the previous equation $\alpha = \alpha_{subopt}$ defined in Equation (3.11).

Finally, we obtain the following algorithm.

Algorithm 3 Supervised Sparse Reconstruction algorithm (SSR)

```
INITIALIZE(q^0, q̃^0)
repeat
    function ESTIMATE q̃^{k+1}(q^k)
        update a_{k+1} by Equation (4.7)
        update b_{k+1} by Equation (4.8)
    end function
    function ESTIMATE q^{k+1}(q̃^{k+1})
        compute q^r ← (m_r, σ_r^2) by Equation (4.12) and Equation (4.11)
        compute q^α ← (m_α, σ_α^2) by Equation (4.14) and Equation (4.13)
        compute α_subopt and Equation (??).
        compute q^{k+1} = q^{α_subopt}.
    end function
until Convergence
```

4.3. Unsupervised algorithm. The algorithm described in the previous part is not a fully Bayesian one as it still depends on some hyperparameters, namely the variances induced by the model (4.1) and (4.3). We see in the following how this method can be extended to an unsupervised one by estimating these parameters. The

parameters of the underlying Gamma random variable are not estimated in the following as they define the sharpness of the prior distribution. We thus only estimate the variance parameter of this prior together with the trade off between the prior and the noise.

In order to simplify the different expressions, we introduce in the following the notations $\gamma_b = 1/\sigma_b^2$ and $\gamma_s = 1/\sigma_s^2$. Hence, γ_b and γ_s are the precision parameters of the noise and of the prior distribution. From now on they are also assumed to be random variable with Gamma prior of parameters $(\tilde{a}_b, \tilde{b}_b)$ resp. $(\tilde{a}_s, \tilde{b}_s)$. As we do not have any information on these precision parameters γ_b and γ_s, this prior is a Jeffrey's prior obtained by fixing $\tilde{a}_b = 0, \tilde{b}_b = 0$ resp. $\tilde{a}_s = 0, \tilde{b}_s = 0$.

With these assumptions, the posterior distribution from Equation (4.4) can be written as

$$p(\boldsymbol{x}, \boldsymbol{z}, \gamma_b, \gamma_s | \boldsymbol{y}) \propto \gamma_b^{\frac{M}{2}} \exp\left[-\frac{\gamma_b \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2}{2}\right] \gamma_s^{\frac{N}{2}} \prod_i \sqrt{z_i} \exp\left[-\frac{\gamma_s z_i x_i^2}{2}\right] \frac{\tilde{b}_i^{\tilde{a}_i} z_i^{\tilde{a}_i - 1} e^{-z_i \tilde{b}_i}}{\Gamma(\tilde{a}_i)} \times \frac{\tilde{b}_b^{\tilde{a}_b} \gamma_b^{\tilde{a}_b - 1} e^{-\gamma_b \tilde{b}_b}}{\Gamma(\tilde{a}_b)} \frac{\tilde{b}_s^{\tilde{a}_s} \gamma_s^{\tilde{a}_s - 1} e^{-\gamma_s \tilde{b}_s}}{\Gamma(\tilde{a}_s)}. \tag{4.15}$$

As in the previous section, the conditional posterior $p(\boldsymbol{z}, \gamma_b, \gamma_s | \boldsymbol{x}, \boldsymbol{y})$ is separable and can be approximated thanks to the classical variational Bayesian approach. Once again only the distribution of $\boldsymbol{X}$ needs the use of Algorithm 2. Here the alternate optimization scheme to carry out is:

$$\begin{aligned}
\tilde{q}^{k+1} &= \arg\max_{\tilde{q}} \mathcal{F}\left(q^k \tilde{q} q_b^k q_s^k\right) \\
q^{k+1} &= \arg\max_{q} \mathcal{F}\left(q \tilde{q}^{k+1} q_b^k q_s^k\right) \\
q_b^{k+1} &= \arg\max_{q_b} \mathcal{F}\left(q^{k+1} \tilde{q}^{k+1} q_b q_s^k\right) \\
q_s^{k+1} &= \arg\max_{q_s} \mathcal{F}\left(q^{k+1} \tilde{q}^{k+1} q_b^{k+1} q_s\right)
\end{aligned}$$

4.3.1. Optimization of the approximate p.d.f. q_b. Concerning the random vectors $\boldsymbol{Z}$ and $\boldsymbol{X}$, the updating process follows the same scheme as the supervised case, see Section 4.2, and is not recalled here. The main differences reside in the update of the parameter distributions.

As the distributions of γ_b and γ_s are supposed to be Gamma, which is coherent with the conjugate priors hypothesis, at each iteration we just adapt the parameters. Hence we initialize our algorithm by considering that

$$q_b^0 \sim \mathcal{G}(a_b^0, b_b^0)$$

At iteration $k+1$ we consider the maximum of the free energy from Equation (3.2)

which gives

$$
\begin{aligned}
q_b^{k+1}(\gamma_b) &\propto \exp\Big[< \ln p(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\gamma_b,\gamma_s) >_{\prod_j q_j^{k+1}(x_j)\prod_j \tilde{q}_j^{k+1}(z_j) q_s^k(\gamma_s)}\Big] \\
&\propto \exp\left[< (\frac{M}{2}+\tilde{a}_b-1)\ln(\gamma_b) - \gamma_b(\frac{\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{x}\|^2}{2}+\tilde{b}_b) >_{\prod_j q_j^{k+1}(x_j)}\right] \\
&\propto \exp\Bigg[(\frac{M}{2}+\tilde{a}_b-1)\ln(\gamma_b) \\
&\quad -\gamma_b\Big(\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{m}_{k+1}\|^2 + \frac{1}{2}\sum_{i=1}^N \mathrm{diag}\left(\boldsymbol{H}^{\mathrm{t}}\boldsymbol{H}\right)_i \boldsymbol{\sigma}_{k+1}^2(i) + \tilde{b}_b\Big)\Bigg]
\end{aligned}
$$

So q_b^{k+1} is a Gamma p.d.f. of parameters:

$$
a_b^{k+1} = \frac{M}{2} + \tilde{a}_b \tag{4.16}
$$

$$
b_b^{k+1} = \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{m}_{k+1}\|^2 + \frac{1}{2}\sum_{i=1}^N \mathrm{diag}\left(\boldsymbol{H}^{\mathrm{t}}\boldsymbol{H}\right)_i \boldsymbol{\sigma}_{k+1}^2(i) + \tilde{b}_b \tag{4.17}
$$

4.3.2. Optimization of the approximate p.d.f. q_s. As for γ_b, the approximation of q_s is performed in the family of Gamma distributions. Hence at the initialization step we assume that

$$
q_s^0 \sim \mathcal{G}(a_s^0, b_s^0)
$$

and at iteration $k+1$, thanks again to Equation (3.2), we obtain

$$
\begin{aligned}
q_s^{k+1}(\gamma_s) &\propto \exp\Big[< \ln p(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\gamma_b,\gamma_s) >_{\prod_j q_j^{k+1}(x_j)\prod_j \tilde{q}_j^{k+1}(z_j) q_b^{k+1}(\gamma_b)}\Big] \\
&\propto \exp\left[< (\frac{N}{2}+\tilde{a}_s-1)\ln(\gamma_s) - \gamma_s(\frac{1}{2}\sum_{i=1}^N z_i x_i^2 + \tilde{b}_s) >_{\prod_j q_j^{k+1}(x_j)\tilde{q}_j^{k+1}(z_j)}\right] \\
&\propto \exp\left[< (\frac{N}{2}+\tilde{a}_s-1)\ln(\gamma_s) - \gamma_s(\frac{1}{2}\sum_{i=1}^N \frac{\boldsymbol{a}_{k+1}(i)}{\boldsymbol{b}_{k+1}(i)}(\boldsymbol{m}_{k+1}^2(i)+\boldsymbol{\sigma}_{k+1}^2(i)) + \tilde{b}_s)\right]
\end{aligned}
$$

So q_s^{k+1} is a Gamma p.d.f. and its parameters are deduced by identification.

$$
a_s^{k+1} = \frac{N}{2} + \tilde{a}_s \tag{4.18}
$$

$$
b_s^{k+1} = \frac{1}{2}\sum_{i=1}^N \frac{\boldsymbol{a}_{k+1}(i)}{\boldsymbol{b}_{k+1}(i)}(\boldsymbol{m}_{k+1}^2(i)+\boldsymbol{\sigma}_{k+1}^2(i)) + \tilde{b}_s \tag{4.19}
$$

Finally, the algorithm can be summed up as follows.

Algorithm 4 UnSupervised Sparse Reconstruction algorithm (USSR)

INITIALIZE($q^0, \tilde{q}^0, q_b^0, q_s^0$)
repeat
  function ESTIMATE $\tilde{q}^{k+1}(q^k, q_b^k, q_s^k)$
    update $\boldsymbol{a}_{k+1}$ using Equation (4.7)
    update $\boldsymbol{b}_{k+1}$ using Equation (4.8)
  end function
  function ESTIMATE $q^{k+1}(\tilde{q}^{k+1}, q_b^k, q_s^k)$
    compute $q^r(\boldsymbol{x}) \leftarrow (\boldsymbol{m}_r, \boldsymbol{\sigma}_r^2)$ using Equation (4.12) and Equation (4.11)
    compute $q^\alpha(\boldsymbol{x}) \leftarrow (\boldsymbol{m}_\alpha, \boldsymbol{\sigma}_\alpha^2)$ using Equation (4.14) and Equation (4.13)
    compute α_{subopt}
    compute $q^{k+1} = q^{\alpha_{subopt}}$.
  end function
  function ESTIMATE $q_b^{k+1}(\tilde{q}^{k+1}, q^{k+1})$
    update a_b^{k+1} using Equation (4.16)
    update b_b^{k+1} using Equation (4.17)
  end function
  function ESTIMATE $q_s^{k+1}(\tilde{q}^{k+1}, q^{k+1})$
    update a_s^{k+1} using Equation (4.18)
    update b_s^{k+1} using Equation (4.19)
  end function
until Convergence

5. Simulations. This section is devoted to numerical validations of the method proposed in this paper. For the sake of completeness we will treat two inverses problems. The first one is given by a tomographic example. The goal is to enhance the accuracy and the effectiveness of our approach, in comparison with classical ones, such as classical Variational Bayesian methods or Monte Carlos Markov Chain (MCMC) methods. From the limitations of these concurrent approaches, we choose to consider only a small dimensional inverse problem (4096 unknowns), and thus to invert the Radon transform of a small sparse image (64×64 pixels).

The second experimental result is devoted to a relatively large inverse problem (≈ 300000 unknowns). In this case, the problem is to identify different components in a dictionary learning process. This learning is performed in a very noisy environment, where the signal-to-noise ratio can have negative values. This signal processing problem can appear for instance in an astrophysical context (detection of gravitational waves [34]) or in radar imaging [47, 2]. This second example illustrates the performances of the method for large dimensional problems.

In both cases the sparsity information is introduced by an iid Student-t prior. This prior is a particular case of GVSM. In the following we thus fix $\tilde{a}_i = \frac{\nu}{2}$ and $\tilde{b}_i = \frac{\nu}{2}$ in Equation (4.4).

5.1. Tomographic example. For the sake of completeness, a short description of the concurrent approaches is given, enhancing the main differences between them. In a second part, we describe the phantom together with the simulation parameters.

5.1.1. Algorithms descriptions.

Filtered Back Projection (FBP). Filtered Back Projection is the classical approach to invert the Radon transform [31, 21]. This algorithm is obtained by sampling the continuous inversion formula. Each line of the sinogram (see Fig. 5.1) is filtered with a ramp filter. The filtered data are backprojected. The discrete version of the backprojection operator is given by $\boldsymbol{H}^{\mathrm{t}}$.

Monte Carlos Markov Chain. The MCMC method contains a large class of Bayesian algorithms [36]. In the following we consider the Gibbs algorithm for its efficiency when the size of the problem increases. This algorithm is used for a wide class of applications [26, 14, 6]. The principle is to obtain samples of the posterior law given by Equation (4.4) by an alternate sampling with conditional laws.
The algorithm is as follows:

(i) $\boldsymbol{z}^k$ is sampled with $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x}^{k-1})$
(ii) $\boldsymbol{x}^k$ is sampled with $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{z}^k)$
(iii) go to i) until convergence of the Markov chain.

As the conditional law $p(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x}^{k-1})$ is a separable Gamma distribution, the computation of the sample $\boldsymbol{z}^k$ is easy. Furthermore $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{z}^k)$ is a correlated Gaussian distribution with a covariance matrix $\boldsymbol{R}_k = \boldsymbol{M}_k^{\mathrm{t}}\boldsymbol{M}_k = [\frac{1}{\sigma_b^2}\boldsymbol{H}^{\mathrm{t}}\boldsymbol{H} + \frac{1}{\sigma_s^2}\mathrm{Diag}(\boldsymbol{z}^k)]^{-1}$ and a mean $\boldsymbol{m}_k = \frac{1}{\sigma_b^2}\boldsymbol{R}_k\boldsymbol{H}^{\mathrm{t}}\boldsymbol{y}$. The sampling under this correlated distribution is performed by sampling a vector of centered iid Gaussian random variables with variance 1. This vector is then multiplied by the correlation matrix $\boldsymbol{M}_k$ and added to $\boldsymbol{m}_k$.

REMARK 2. *At each sampling iteration the covariance matrix of size $N \times N$ has to be inverted.*

Classical Bayesian Variational approach. This method was already described in Section 3. In the alternate descent algorithm, one can choose the degree of separability of the approximate distribution. In the following we consider two cases. In the first case, the so called VBBloc, we consider, as in [4] that the separation of the approximating law is only between $\boldsymbol{x}$ and $\boldsymbol{z}$. This leads to a consideration of the approximating distribution as:

$$q(\boldsymbol{x}, \boldsymbol{z}) = q(\boldsymbol{x})\tilde{q}(\boldsymbol{z})$$

and

$$\begin{aligned} q &\sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{R}) \\ \tilde{q} &\sim \mathcal{G}(\boldsymbol{a}, \boldsymbol{b}) \end{aligned}$$

Thus, with Equation (3.2), we obtain $\forall i \in \{1, \ldots, N\}$ the following updating equations, at iteration $k+1$:

$$\begin{aligned} \boldsymbol{a}_{k+1}(i) &= \frac{\nu}{2} + \frac{1}{2}, \\ \boldsymbol{b}_{k+1}(i) &= \frac{\nu}{2} + \frac{\boldsymbol{m}_k^2(i) + \mathrm{diag}(\boldsymbol{R}_k)(i)}{2\sigma_s^2} \\ \boldsymbol{R}_{k+1} &= \left(\frac{1}{\sigma_s^2}\mathrm{Diag}(\boldsymbol{a}/\boldsymbol{b}_{k+1}) + \frac{1}{\sigma_b^2}\boldsymbol{H}^{\mathrm{t}}\boldsymbol{H}\right)^{-1} \\ \boldsymbol{m}_{k+1} &= \frac{1}{\sigma_b^2}\boldsymbol{R}_{k+1}\boldsymbol{H}^{\mathrm{t}}\boldsymbol{y}. \end{aligned}$$

REMARK 3. *At each step, the updating of the covariance matrix requires the inversion of an $N \times N$ matrix, but the convergence rate is better than for the MCMC approach.*

To overcome the limit given by a matrix inversion in the classical variational Bayesian framework, we can construct an approximate distribution separable on $\boldsymbol{X}$. Hence, we estimate a vector of variances instead of the matrix of covariance. This approach is called VBComp in the following.

$$q(\boldsymbol{x}, \boldsymbol{z}) = \Pi_i q_i(x_i)\tilde{q}(\boldsymbol{z})$$

In this case Equation (3.2) gives the following updating equations, $\forall i \in \{1, \ldots, N\}$:

$$\begin{aligned}
\boldsymbol{a}(i)_{k+1} &= \frac{\nu}{2} + \frac{1}{2}, \\
\boldsymbol{b}_{k+1}(i) &= \frac{\nu}{2} + \frac{\boldsymbol{m}_k^2(i) + \boldsymbol{\sigma}_k^2(i)}{2\sigma_s^2}
\end{aligned}$$

And, for every $i \in \{1, \ldots, N\}$:

$$\begin{aligned}
\boldsymbol{\sigma}_{k+1}^2(i) &= \left(\frac{1}{\sigma_s^2}\boldsymbol{a}(i)/\boldsymbol{b}_{k+1}(i) + \frac{1}{\sigma_b^2}(\boldsymbol{H}^{\mathrm{t}}\boldsymbol{H})_{(i,i)} \right)^{-1} \\
\boldsymbol{m}_{k+1}(i) &= \frac{\boldsymbol{\sigma}_{k+1}^2(i)}{\sigma_b^2}(\boldsymbol{H}^{\mathrm{t}}\boldsymbol{y}(i) - (\boldsymbol{d}(i) - (\boldsymbol{H}^{\mathrm{t}}\boldsymbol{H})_{(i,i)}\boldsymbol{m}_k(i))) \\
\boldsymbol{d} &= \boldsymbol{H}^{\mathrm{t}}\boldsymbol{H}\boldsymbol{m}_k
\end{aligned}$$

REMARK 4. *For each pixel x_i, the corresponding value of $\boldsymbol{d} = \boldsymbol{H}^{\mathrm{t}}\boldsymbol{H}\boldsymbol{m}_k$ must be determined.*

5.1.2. Simulation configuration. The test image is given by a sparse phantom, composed of 7 peaks on a grid 64×64 (see Table 5.1 and Fig. 5.2(a)). Data have

Coordinate	(28,28)	(25,28)	(28,25)	(40,28)	(32,38)	(48,48)	(8,52)
Value	1	1	1	0.5	0.7	0.8	0.6

TABLE 5.1
Peaks definition in the phantom

been simulated in a parallel beam geometry. These projections are collected from 32 angles θ, uniformly spaced over $[0, 180[$. Each projection is composed of 95 detector cells. We add a white Gaussian noise (iid) with a standard deviation equal to 0.3 (see Fig. 5.1). Data have thus a relatively bad signal to noise ratio and the number of unknowns is larger than the number of data, which leads to an ill-posed inverse problem.

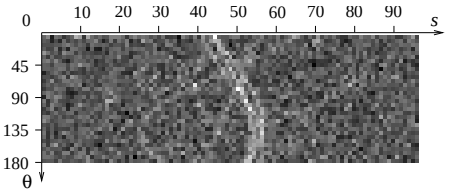


FIG. 5.1. *Data collected : sinogram composed of 32 angles and 95 detector cells.*

5.1.3. Results and discussion. In this section, we expose the inversion of this tomographic problem with the approaches described earlier.

All the iterative approaches are initialized with a zero mean and a variance equal to one, and the hyperparameters σ_b^2, σ_s^2 and ν are respectively fixed at 1, 0.05 and 0.1. The original image and its different reconstructions given by Filtered BackProjection and posterior mean estimators, are summed up on Fig. 5.2. A comparison of Fig. 5.2 (b) with 5.2 (c), 5.2 (d) and 5.2 (e) clearly shows that the analytical inversion of the Radon transform performed by Filtered Back Projection (FBP) is less robust to noise than Bayesian approaches. Asymptotically, in Bayesian cases, theoretical results are favorable to the MCMC approach as they do not need any approximation. In practice, the number of samples is too small to fit with the asymptotic results of MCMC methods, which explains the bad reconstruction observed in Fig. 5.2(c). Finally, the Supervised Sparse Reconstruction (SSR) (see Fig. 5.2(f)) has the same reconstruction quality as the classical variational Bayesian approaches (see VBBloc Fig. 5.2(d) and VBComp Fig. 5.2(e)). However, when we compare the execution time (see Table 5.2), we see that our approach is 10 time faster than the VBBloc approach, 40 time faster than the VBComp approach and 370 faster than the MCMC approach for this small inverse problem. Moreover, this ratio increases with the size of the problem as both MCMC and VBBloc need the inversion of a covariance matrix at each iteration, which is not the case for our algorithm.

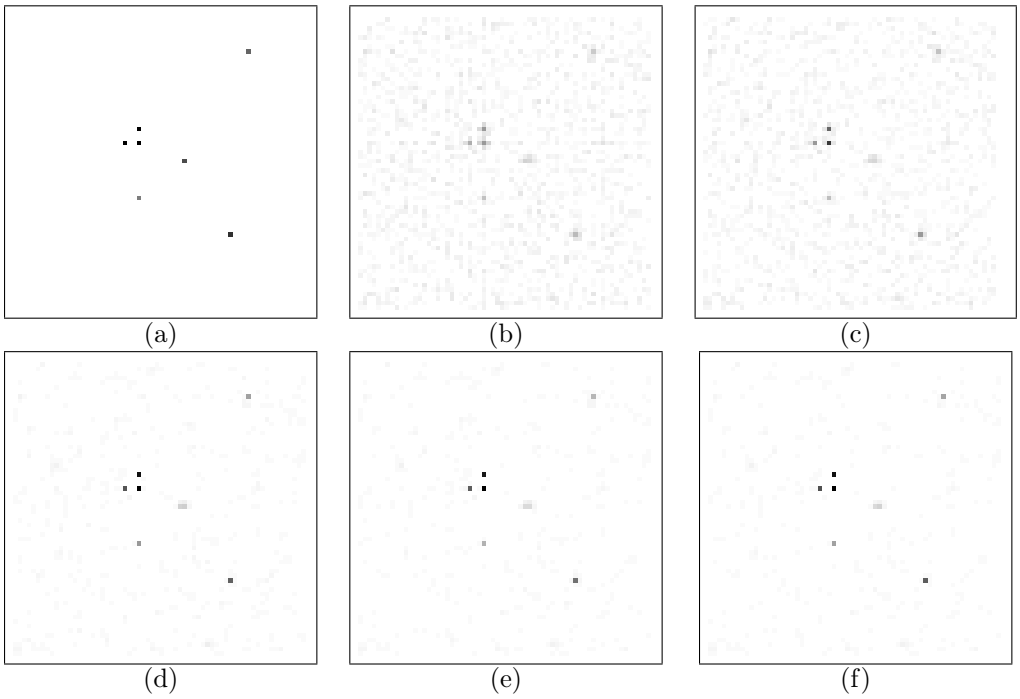


FIG. 5.2. *Images are presented with the same inversed grayscale: (a) true image of 7 peaks, (b) FBP with ramp filter, (c) MCMC Gibbs approach, (d) classical variational Bayesian (VBBloc) with bloc optimization, (e) classical variational Bayesian (VBComp) with component optimization, (f) SSR approach.*

5.1.4. Hyperparameters estimation. As seen in Section 4.2, our approach is defined in a fully Bayesian framework. We thus estimate the values of hyperparameters

TABLE 5.2
Comparison of the different approaches: computing time (second) and quality of estimation (SNR).

Method	FBP	VBBloc	VBComp	VBGrad (SSR)	MCMC Gibbs
CPU time (s)	0.05	586.20	1759.1	44.41	37079.50
Nb of iterate	1	15	8(×4096)	500	1000
SNR	-2.04	5.87	5.76	6.00	-0.60

by introducing a non-informative Jeffrey's prior, as described in Part 4.3. We estimate thus the trade-off between the likelihood and the prior through the estimation of γ_b and γ_s. Hence, we apply the algorithm UnSupervised Sparse Reconstruction (USSR) (see Algorithm 4) in our tomographic dataset. As for the previous simulation, the initial values of the mean are fixed at zero and the variance are fixed at one. For the hyperparameters γ_b and γ_s the initial values are respectively fixed at 1 and 0.05.

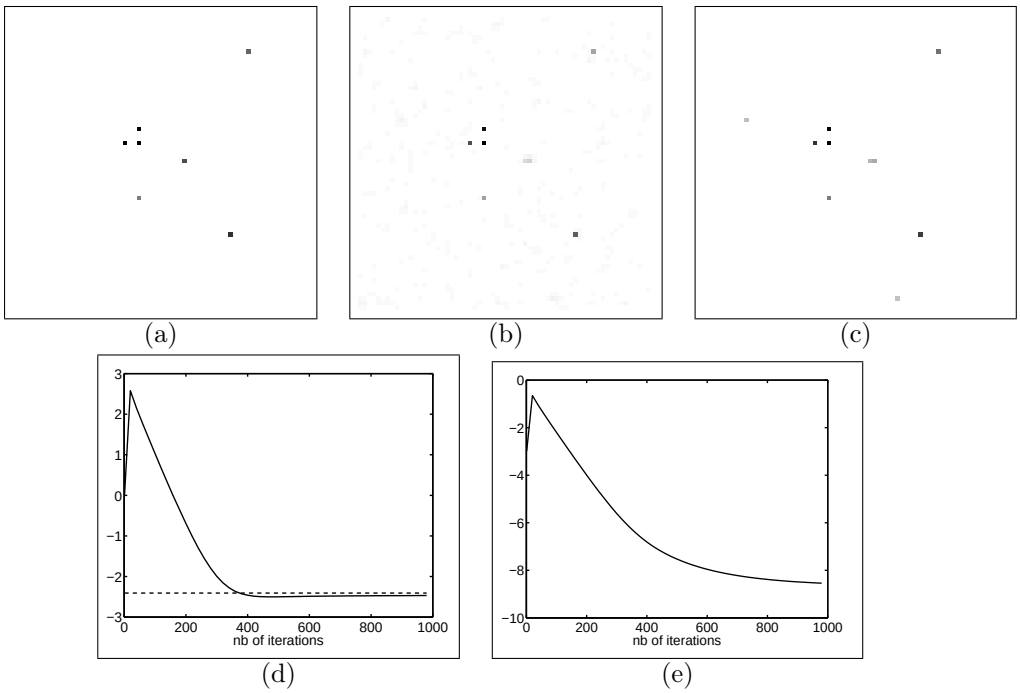


FIG. 5.3. *Results with hyperparameters estimation: (a) True image, (b) reconstructed image with SSR algorithm (hyperparameters are fixed), (c) reconstructed image with USSR (image and hyperparameters are estimated jointly), (d) logarithm of σ_b^2: the dashed line correspond to the true value, (e) logarithm of σ_s^2*

The results are summed up in Fig. 5.3. We observe that the hyperparameters estimation intensifies the sparsity of the reconstructed image together with the SNR, as it goes from 6.00 db in the previous case to 10.06 db. Estimating the true hyperparameters is in this case more relevant than arbitrarily chosen parameters. We observe on Fig. 5.3 (c) that the background is equal to zero even if some additional peaks appear in the reconstructed image.

Finally we see in Fig. 5.3, (d) and (e), the behavior of the estimation of the hyperparameters respective to the number of iterations. This plot is in a logarithm

scale due to the dynamic of the fluctuations. We observe that for σ_b^2 this estimation converges to the true value (dashed line). For σ_s^2 we do not known the true value, but one can notice that this estimation also converges.

5.2. Component identification (denoising). The main advantage of our approach is that it can address larger problems ($\approx$ 300000 unknowns in the present case). In the present part, our purpose is to treat an identification problem in a redundant dictionary decomposition such as in [48, 7, 41]. More precisely, we identify chirps components in a linear mixture. Unfortunately, this mixture is corrupted by noise and spurious signal. To identify each chirp in the mixture and to remove the effect of the spurious signal, we develop a dictionary learning approach. For the sake of simplicity, the construction of the dictionary is included in Appendix 7.2.

The simulated data are sampled at a frequency $F_e = 44$khz, and they are composed of $N = 2^{16}$ points, thus the duration of measurement T_{mes} is equal to 1.5 second. The dictionary components are stored in a matrix $\boldsymbol{H}$ and we denote by $\boldsymbol{x} \in \mathbb{R}^N$ the vector of weights to be estimated. Hence, the measurements can be modeled as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{b}, \tag{5.1}$$

where $\boldsymbol{b} \sim \mathcal{N}(0, \sigma_b^2 I)$ is a Gaussian white noise. Note that, as described in Appendix 7.2, $\boldsymbol{H}$ represents here an overcomplete dictionary.

We also consider different simulation contexts, with different noise levels and a growing number of components. In each case the simulated data are composed of $N = 2^{16}$ points.

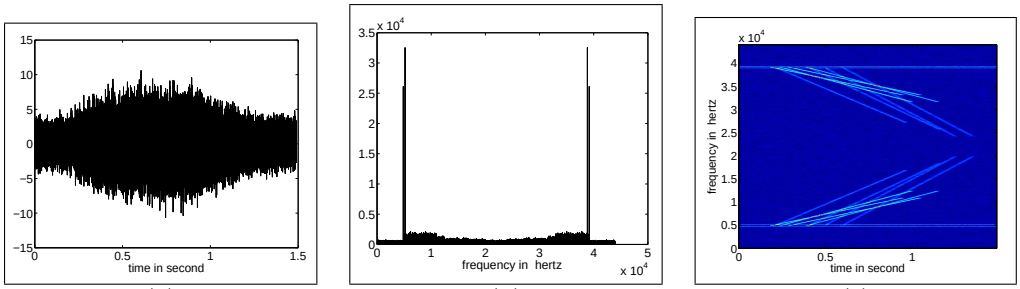
As a first step, the data are composed of two cosine terms together with height chirp functions whose parameters are given in Table 5.3. Simulated data have a relatively bad SNR, at 5.68 db. These data are given in Fig. 5.4.

TABLE 5.3
Parameters of the different components

type of function	Amplitude	frequency (hz)	rate (hz)	first time (s)
cosine	1	5169.7	-	-
cosine	0.8	4834	-	-
Chirp	1.4	-	8000	0.2
Chirp	1.4	-	10000	0.25
Chirp	1.0	-	16000	0.22
Chirp	1.0	-	20000	0.5
Chirp	1.2	-	10000	0.4
Chirp	1.0	-	18000	0.41
Chirp	1.0	-	20000	0.6
Chirp	1.4	-	8000	0.3

This inverse problem is then solved thanks to our unsupervised approach. Our algorithm was launched with the shape parameter of the Student-t equals to $\nu = 0.01$ in order to introduce a very sparse prior. The initialization parameters are:

- The mean of q^0, $\boldsymbol{m}_0 = 0$,
- the variance of q^0, $\boldsymbol{\sigma}_0^2 = 1$,
- the mean of q_b^0, is equal to 10^{-5},



(a) (b) (c)

FIG. 5.4. *Data in different fields: (a) time, (b) frequency, (c) time-frequency*

- the mean of q_s^0, is equal to 10^5.

After 400 iterations (316 s), the algorithm (USSR) converges to a stable solution. We consider here that the estimation $\hat{\boldsymbol{x}}$ is obtained by the mean of the approximate distribution q. Our method find the right position of the coefficients in the signal. Concerning the amplitude, the relative error obtained is between 0.4% and 2%, thus very close to the right value.

Estimation $\hat{\boldsymbol{s}}$ of s is performed thanks to the estimation of the coefficients $\hat{\boldsymbol{x}}$ and Equation (7.6). The SNR of $\hat{\boldsymbol{s}}$ is equal to 22.6 db.

As a second step, we show the performances of the algorithm USSR when the signal s is hidden by the noise, the SNR of data being equal to -5 db (see Fig. 5.5 (a)). We generate a signal composed of four chirps of parameters summed up in Table 5.4.

TABLE 5.4
Parameters of the different components

type of function	Amplitude	Chirp rate (hz)	first time (s)
Chirp	0.9526	16000	0.1530
Chirp	1.1606	16000	0.1945
Chirp	0.7369	18000	0.2000
Chirp	1.1724	18000	0.1865

The estimation of coefficients is performed using our USSR algorithm with the same initialization as in the previous case. After 400 iterations we obtain the coefficients illustrated in Fig. 5.5, (c) and (d). The reconstructed coefficients are represented by a line and the true values of the coefficients are marked by a cross. We observe that all coefficients are in the right place and that the peak amplitudes are systematically underestimated, but the estimated values are relatively close to the true ones. Fig. 5.5 (b) points out the estimator $\hat{\boldsymbol{s}}$. We see that the shape of the signal is close to the true one, and that when the signal is missing, between 1 and 1,45 s, the level of the residual noise is relatively low.

TABLE 5.5
Signal to Noise Ratio (SNR) in db

data	USSR approach	Wiener filter	wavelet shrinkage	STFT shrinkage
-5.0	15.05	1.1941	1.8677	9.2147

In Table 5.5 we compare the reconstructed signal $\hat{\boldsymbol{s}}$ with the reconstruction obtained with three classical denoising approaches, namely the Wiener filter, the soft wavelet shrinkage, with the four vanishing moments symmlet, and the hard shrinkage in the STFT domain. In these three methods, we furthermore have to tune a parameter which is the correlation power for the Wiener filter and the threshold for the soft wavelet and the hard STFT shrinkage.We therefore choose the value of this hyperparameter which minimizes the Signal-to-Noise Ratio (SNR), which depends on the signal. Our approach hugely increases the SNR (20 db) as the noise level is divided by 100 whereas the classical methods reduce the noise only by a factor of 4, 5 and 26 respectively. As expected, Wiener and wavelet approaches are not adapted to the considered problem and give insufficient results. Moreover the USSR approach increases the SNR of 6 db respectively to the redundant decomposition given by STFT shrinkage, even if the STFT decomposition seems well adapted in this context. This result comes from the sparsity of the reconstructed signal, which is larger in the case of the USSR reconstruction.

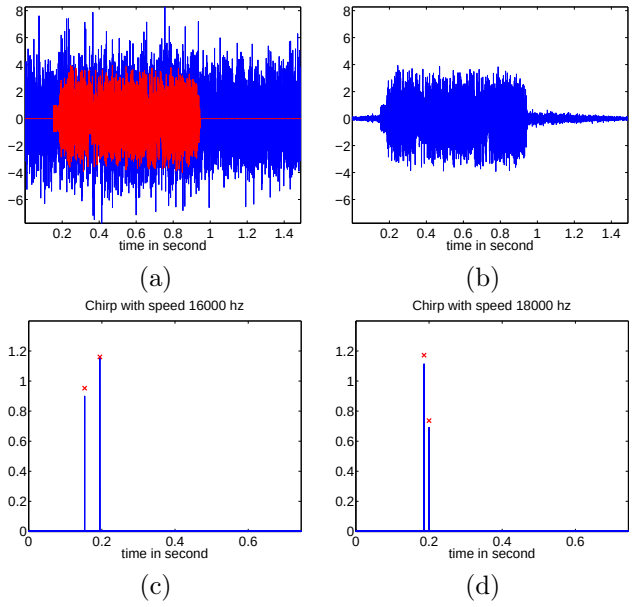


FIG. 5.5. *Limit case: (a) data and true signal, (b) reconstructed signal*

Finally, we study the robustness of the USSR algorithm with respect to the Signal to Noise Ratio. This simulation allows a better understanding of the reconstruction properties of our method. Hence we consider simulated data with 6 different SNR $(-5, -2, 1, 2, 5, 10)$ and with low (4) and high (16) numbers of components. For each SNR and each number of components, we consider 30 datasets, the components of the signal being randomly chosen in the dictionary.

We reconstruct these 360 sets of data by the USSR algorithm taking the same configuration and the same initialization as in previous cases.

The results are summed up in Fig 5.6 (a-d). Each point of the plot is computed by averaging the results of 30 reconstructions. In Fig 5.6 (a), resp. (b), we plot the true positive proportion, resp. false positive proportion, of the significant reconstructed

coefficients*.

At first sight we see that there are no false positives with our approach. Indeed, as the approach is unsupervised with a sparsity prior, the coefficients with low energy are considered as noise. Moreover we can reconstruct 16 components without loss when the SNR is greater or equal to 5 db and resp. 4 components when the SNR is greater to 1 db. Fig. 5.6 shows that the reconstruction is more difficult when the number of components increases. Fig. 5.6 (c) is obtained by calculating the SNR of the reconstructed signal $\hat{\boldsymbol{s}}$. We observe a quite linear behavior. For 4 components the gain is of 17 db whereas for 16 components we gain 11.5 db. Finally, Fig. 5.6 (d) exposes the quadratic error of the peaks amplitude. There are two cases here. When all the components are found this error is linear (see Fig. 5.6 (d) the bottom curve when SNR>1) but it increases more quickly when some components are not found (see Fig. 5.6 (d) the bottom curve when SNR<1).

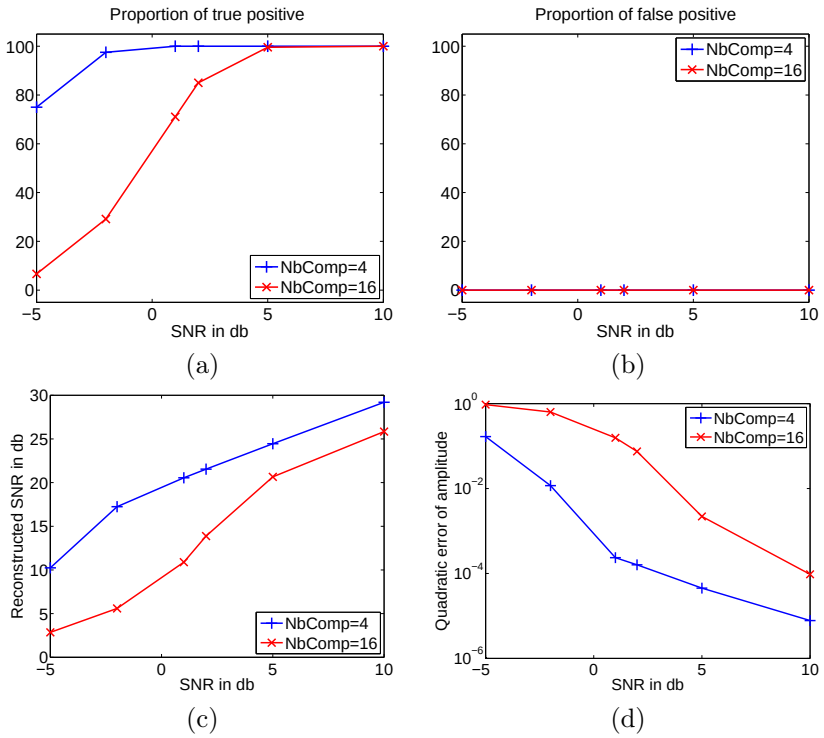


FIG. 5.6. *Study of Robustness versus noise, each point of the curve being calculated by averaging 30 reconstructions with components randomly chosen: (a) True positive proportion of the significant reconstructed coefficients, (b) False positive proportion of the significant reconstructed coefficients, (c)* SNR *of the estimated signal* $\hat{\boldsymbol{s}}$*, (d) Quadratic error of the peaks amplitude*

6. Conclusion. In this paper, we have defined an iterative algorithm based on the descent gradient principle and adapted to the context of variational Bayesian methods. We also prove the convergence of this method in the probability density functions sets, ensuring that it converges to an optimal approximation of the posterior distribution. The main interest of this algorithm is that it converges faster than the

*the significant coefficients are obtained by applying a threshold equal to 0.2 on the coefficients vector. This threshold is equal to the third of the minimum value of the true non zero coefficients.

classical Bayesian methods and allows a use on large dimensional datasets. We have furthermore give its implementation in the case of a white noise model when the prior information enhances some sparse behavior. A small tomographic application allows to compare our method with classical ones. We see that even in small cases, our algorithm can be faster than classical ones. A second simulation part, corresponding to a dictionary identification allows to understand the behavior of our method for large dimensional problems. Once again this method has good reconstruction properties in this case.

7. Appendix.

7.1. Proof of Lemma 2.3. In order to ensure that for small values of α, we have $F(\mu^\alpha) \geq F(\mu^k)$, we show that the right part of Equation (2.17),

$$F(\mu^\alpha) - F(\mu^k) \geq \partial F_{\mu^k}(\mu^\alpha - \mu^k) - L\|h_\alpha(\mu^k, .) - 1\|^2_{L^2(\mu^k)} - \int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k, \boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x}),$$

is positive.

2.$dF_{q^k}(q^\alpha - q^k) - \frac{M}{2}\|(q^\alpha - q^k)^2\| > 0$, for all $\alpha < \alpha_0$. This ensures that $F(q^\alpha) > F(q^k)$ for $\alpha \in (0, \alpha_0)$.

First, one can notice that

$$\begin{aligned} -\int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k, \boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x}) &= -\int_{\mathbb{R}^N} (\alpha \mathrm{d}f(\mu^k, \boldsymbol{x}) + \ln K_k(\alpha))h_\alpha(\mu^k, \boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x}) \\ &= -\alpha\left(\int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{x})h_\alpha(\mu^k, \boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x}) + \frac{1}{\alpha}\ln K_k(\alpha)\right). \end{aligned}$$

But Jensen's inequality ensures that

$$\ln K_k(\alpha) \leq -\alpha \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x}). \tag{7.1}$$

Which leads to

$$-\int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k, \boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x}) \geq -\alpha \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k, \boldsymbol{x})(h_\alpha(\mu^k, \boldsymbol{x}) - 1)\mathrm{d}\mu^k(\boldsymbol{x}). \tag{7.2}$$

Hence,

$$\begin{aligned} \partial F_{\mu^k}(\mu^\alpha - \mu^k) - \int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k, \boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x}) &\geq \int_{\mathbb{R}^N} (1-\alpha)\mathrm{d}f(\mu^k, \boldsymbol{x})(h_\alpha(\mu^k, \boldsymbol{x}) - 1)\mathrm{d}\mu^k(\boldsymbol{x}) \\ &= \frac{1-\alpha}{\alpha}\int_{\mathbb{R}^N} (\alpha \mathrm{d}f(\mu^k, \boldsymbol{x}) + \ln K_k(\alpha))(h_\alpha(\mu^k, \boldsymbol{x}) - 1)\mathrm{d}\mu^k(\boldsymbol{x}) - \int_{\mathbb{R}^N} \frac{(1-\alpha)\ln K_k(\alpha)}{\alpha}(h_\alpha(\mu^k, \boldsymbol{x}) - 1)\mathrm{d}\mu^k(\boldsymbol{x}) \\ &= \frac{1-\alpha}{\alpha}\int_{\mathbb{R}^N} (\alpha \mathrm{d}f(\mu^k, \boldsymbol{x}) + \ln K_k(\alpha))(h_\alpha(\mu^k, \boldsymbol{x}) - 1)\mathrm{d}\mu^k(\boldsymbol{x}), \end{aligned}$$

as $\frac{(1-\alpha)\ln K_k(\alpha)}{\alpha}$ is constant and $\int_{\mathbb{R}^N}(h_\alpha(\mu^k, \boldsymbol{x}) - 1)\mathrm{d}\mu^k(\boldsymbol{x}) = 0$.

Finally,

$$
\begin{aligned}
&\partial F_{\mu^k}(\mu^\alpha-\mu^k) - L\|h_\alpha - 1\|^2_{L^2(\mu^k)} - \int_{\mathbb{R}^N} \ln(h_\alpha(\mu^k,\boldsymbol{x}))\mathrm{d}\mu^\alpha(\boldsymbol{x})\\
&\geq \int_{\mathbb{R}^N} \left((\alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha))(\frac{1}{\alpha}-1) - L(h_\alpha(\mu^k,\boldsymbol{x})-1)\right)(h_\alpha(\mu^k,\boldsymbol{x})-1)\mathrm{d}\mu^k(\boldsymbol{x})\\
&= \int_{\{x:\alpha \mathrm{d}f(\mu^k,\boldsymbol{x})+\ln K_k(\alpha)>0\}} \left((\alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha))(\frac{1}{\alpha}-1) - L(h_\alpha(\mu^k,\boldsymbol{x})-1)\right)(h_\alpha(\mu^k,\boldsymbol{x})-1)\mathrm{d}\mu^k(\boldsymbol{x})\\
&+ \int_{\{x:\alpha \mathrm{d}f(\mu^k,\boldsymbol{x})+\ln K_k(\alpha)\leq 0\}} \left((\alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha))(\frac{1}{\alpha}-1) - L(h_\alpha(\mu^k,\boldsymbol{x})-1)\right)(h_\alpha(\mu^k,\boldsymbol{x})-1)\mathrm{d}\mu^k(\boldsymbol{x}).
\end{aligned}
\tag{7.3}
$$

Let us consider each integrals appearing in Equation (7.3) separately. First, we can notice that if $\alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha) < 0$, so is $h_\alpha(\mu^k,.) - 1$. Furthermore, for every $\alpha > 0$ and $\boldsymbol{x} \in \mathbb{R}^N$, we have $h_\alpha(\mu^k,.) - 1 \geq \alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha)$. Hence if $\boldsymbol{x} \in \mathbb{R}^N$ is such that $\alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha) < 0$ and $\alpha < 1$ then

$$
\left((\alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha))(\frac{1}{\alpha}-1) - L(h_\alpha(\mu^k,\boldsymbol{x})-1)\right)(h_\alpha(\mu^k,\boldsymbol{x})-1) \geq (h_\alpha(\mu^k,\boldsymbol{x})-1)^2\left(\frac{1}{\alpha}-1-L\right),
$$

which is positive as soon as $\alpha \leq \frac{1}{1+L}$.

Consider now that $\boldsymbol{x} \in \mathbb{R}^N$ is such that $\alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha) \geq 0$. The Mean Value Theorem applied to the exponential function ensures that one can find, for every $\boldsymbol{x} \in \mathbb{R}^N$ and $\alpha > 0$, a $\theta(\boldsymbol{x},\alpha) \in (0, \alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha))$ such that

$$
h_\alpha(\mu^k,\boldsymbol{x}) = e^{\alpha \mathrm{d}f(\mu^k,\boldsymbol{x})+\ln K_k(\alpha)} = 1+(\alpha \mathrm{d}f(\mu^k,\boldsymbol{x})+\ln K_k(\alpha))e^{\theta(\boldsymbol{x},\alpha)} \geq 1+(\alpha \mathrm{d}f(\mu^k,\boldsymbol{x})+\ln K_k(\alpha)).
$$

This entails that

$$
\left((\alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha))(\frac{1}{\alpha}-1) - L(h_\alpha(\mu^k,\boldsymbol{x})-1)\right) = (\alpha \mathrm{d}f(\mu^k,\boldsymbol{x})+\ln K_k(\alpha))(\frac{1}{\alpha}-1-Le^{\theta(\boldsymbol{x},\alpha)}).
$$

Furthermore, from Jensen's inequality

$$
0 \leq \alpha \mathrm{d}f(\mu^k,\boldsymbol{x}) + \ln K_k(\alpha) \leq \alpha\left(\mathrm{d}f(\mu^k,\boldsymbol{x}) - \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k,\boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x})\right).
$$

Thus, in this case,

$$
\mathrm{d}f(\mu^k,\boldsymbol{x}) - \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k,\boldsymbol{y})\mathrm{d}\mu^k(\boldsymbol{y}) \geq 0.
$$

And

$$
1 \leq e^{\theta(\boldsymbol{x},\alpha)} \leq h_\alpha(\mu^k,\boldsymbol{x}) \leq e^{\alpha\left(\mathrm{d}f(\mu^k,\boldsymbol{x}) - \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k,\boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x})\right)},
$$

If we consider the function $\alpha \mapsto \frac{1}{\alpha}-1-Le^{\alpha\left(\mathrm{d}f(\mu^k,\boldsymbol{x}) - \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k,\boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x})\right)}$, it is decreasing and, as $\mathrm{d}f$ is continuous, it is also positive for small values of α. Hence, one can find an $\alpha_1 > 0$ such that for every $\alpha \leq \alpha_1$,

$$
\frac{1}{\alpha} - 1 - Le^{\alpha\left(\mathrm{d}f(\mu^k,\boldsymbol{x}) - \int_{\mathbb{R}^N} \mathrm{d}f(\mu^k,\boldsymbol{x})\mathrm{d}\mu^k(\boldsymbol{x})\right)} \geq 0.
$$

Finally one has

$$
\forall \alpha \leq \alpha_0 = \min(\alpha_1, \frac{1}{1+L}),\ \ \partial F_{\mu^k}(\mu^\alpha-\mu^k) - L\|h_\alpha-1\|^2_{L^2(\mu^k)} - \int_{\mathbb{R}^N} (\alpha \mathrm{d}f(\mu^k,\boldsymbol{x})+\ln K_k(\alpha))\mathrm{d}\mu^\alpha(\boldsymbol{x}) \geq 0.
\tag{7.4}
$$

7.2. Context of simulation 5.2. In this part we present the dictionary decomposition of Section 5.2 together with the components of the data used to enhance the reconstruction performances of our unsupervised method.

To build our DFT-chirps dictionary we make the following assumptions : all chirps have the same duration T_{Chirp}, the chirps rate ζ is digitalized on very few values (eight in the present case), and the Discrete Fourier Transform basis is included in the dictionary. In this context, the spurious signals can be represented on very few coefficients of the dictionary. However, we do not make any assumption on the variance of the noise or on the number of chirp functions in the mixture.

$$s(t) = \sum_{i=1}^{N_{freq}} (u_i + jv_i)\phi_i(t) + \sum_{l=1}^{N_{trans}} \sum_{k=1}^{N_{rate}} c_{l,k}\psi_k(t - l\Delta_t), \tag{7.5}$$

where

$$\phi_i(t) = \exp\left[2j\pi f_i t\right],$$

corresponds to a pure frequency f_i with $j^2 = -1$ whereas

$$\psi_k(t) = cos(2\pi(f_0 t + \frac{1}{2}t^2\zeta_k))\Pi_{0,T_{Chirp}}(t)$$

corresponds to the chirps components. Here, f_0 is the starting frequency (at time $t = 0$), ζ_k is the chirp rate, that is the increasing rate of the frequency, $\Pi(t)$ is a gate function, T_{Chirp} is the duration of the chirp, Δ_t is the shift between two chirps functions and $t_l = l\Delta_t$ is the first time where $\psi_k(t - l\Delta_t)$ is not null. We merge all the dictionary's coefficients in a single vector $\boldsymbol{x} = (u_1, \ldots, u_{Nfreq}, v_1, \ldots, v_{Nfreq}, c_{1,1}, \ldots, c_{Ntrans,Nrate})^{\mathrm{t}}$. Where $Nfreq$ is the number of pure frequency functions contained on the dictionary whereas $Ntrans$ is the number of chirp shifts and $Nrate$ is the number of chirp rates. The sampled version of Equation (7.5) is given by

$$\boldsymbol{s} = \boldsymbol{H}\boldsymbol{x}, \tag{7.6}$$

where $\boldsymbol{s} = (s(t_0), \ldots, s(t_N))^{\mathrm{t}}$.

Observations are thus given by Equation (5.1).

The dictionary is composed of chirp functions with only 8 different chirp rates ($N_{rate} = 8$). The frequency f_0 is 5 000 hz, the chirp rates ζ_k are uniformly spaced between 6 000 and 20 000 hz, the shift parameter rate Δ_t is fixed up to a sampling period ($T_e = 1/F_e$). Finally, the duration of the chirp (T_{Chirp}) is equal to the half time of the measurements (T_{mes}). Our dictionary is redundant as the number of coefficients is 4.5 times greater than the number of observations.

REFERENCES

[1] B. Ait el Fquih and T. Rodet, *Variational bayesian Kalman filtering in dynamical tomography*, Proc. IEEE ICASSP (Prague), may 2011, pp. 4004 – 4007.

[2] R. Amiri, M. Alaee, H. Rahmani, and M. Firoozmand, *Chirplet based denoising of reflected radar signals*, Asia International Conference on Modelling & Simulation **0** (2009), 304–308.

[3] S. D. Babacan, R. Molina, and A. K. Katsaggelos, *Variationnal Bayesian Blind Deconvolution Using a Total Variation Prior*, IEEE Trans. Image Processing **18** (2009), no. 1, 12–26.

[4] ———, *Variational bayesian super resolution*, IEEE Trans. Image Processing **20** (2011), no. 4, 984–999.
[5] N. Bali and A. Mohammad-Djafari, *Bayesian approach with hidden Markov modeling and mean field approximation for hyperspectral data analysis*, IEEE Trans. on Image Processing **17** (2008), no. 2, 217–225.
[6] Stephanie Bidon, Olivier Besson, and J-Y Tourneret, *Knowledge-aided stap in heterogeneous clutter using a hierarchical bayesian algorithm*, Aerospace and Electronic Systems, IEEE Transactions on **47** (2011), no. 3, 1863–1879.
[7] T. Buchgraber, D. Shutin, and H. V. Poor, *A sliding-window online fast variational sparse Bayesian learning algorithm*, Proc. IEEE ICASSP, IEEE, 2011, pp. 2128–2131.
[8] G. Chantas, N. Galatsanos, A. Likas, and M. Saunders, *Variational Bayesian image restoration based on a product of t-distributions image prior*, IEEE Trans. Image Processing **17** (2008), no. 10, 1795–1805.
[9] R. Chartrand and W. Brendt, *A Gradient Descent Solution to the Monge-Kantorovich Problem*, Applied Math. Sciences (2009).
[10] R. A. Choudrey, *Variational methods for bayesian independent component analysis*, Ph.D. thesis, University of Oxford, 2002.
[11] G. Demoment, *Image reconstruction and restoration: Overview of common estimation structure and problems*, IEEE Trans. Acoust. Speech, Signal Processing **ASSP-37** (1989), no. 12, 2024–2036.
[12] A. Doucet, S. Godsill, and C. Andrieu, *On sequential Monte Carlo sampling methods for Bayesian filtering*, Statistics and Computing **10** (2000), no. 3, 197–208.
[13] B. Frigyik, S. Srivastava, and M. Gupta, *Functional Bregman Divergence and Bayesian Estimation of Distributions*, IEEE Trans. on Information Theory **54** (2008), no. 11, 5130–5139.
[14] J.-F. Giovannelli, *Unsupervised bayesian convex deconvolution based on a field with an explicit partition function*, IEEE Trans. Image Processing **17** (2008), no. 1, 16–26.
[15] A. Globerson, T. Koo, X. Carreras, and M. Collins, *Exponentiated gradient algorithms for log-linear structured prediction*, In Proc. ICML, 2007, pp. 305–312.
[16] Y. Goussard, G. Demoment, and F. Monfront, *Maximum a posteriori dectection-estimation of Bernoulli-Gaussian processes*, J.G. McWhirter ed., Mathematics in Signal Processing II, Clarendon Press, Oxford, UK, 1990.
[17] M.M. Ichir and A. Mohammad-Djafari, *Hidden markov models for wavelet-based blind source separation*, IEEE Trans. on Image Processing **15** (2006), no. 7, 1887–1899.
[18] J. Idier (ed.), *Bayesian approach to inverse problems*, ISTE Ltd and John Wiley & Sons Inc., London, 2008.
[19] S. Jana and P. Moulin, *Optimality of KLT for high-rate transform coding of gaussian vector-scale mixtures : Application to reconstruction, estimation, and classification*, IEEE Trans. on Info. Theory (2006).
[20] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, *An Introduction to variational Methods for Graphical Models*, Machine Learning **37** (1999), no. 2, 183–233.
[21] A. C. Kak and M. Slaney, *Principles of computerized tomographic imaging*, IEEE Press, New York, NY, 1988.
[22] H. Kellerer, *Measure theoretic versions of linear programming*, Math. Z. **198** (1988), no. 3, 367–400.
[23] J. Kivinen and M. Warmuth, *Exponentiated gradient versus gradient descent for linear predictors*, Information and Computation **132** (1997), no. 1, 1–63.
[24] D. MacKay, *Information theory, inference, and learning algorithms*, Cambridge University Press, 2003.
[25] D. J. C. MacKay, *Ensemble learning and evidence maximization*, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.54.4083, 1995.
[26] S. Makni, P. Ciuciu, I. Idier, and J.-P. Poline, *Joint detection-estimation of brain activity in functional MRI: a multichannel deconvolution solution*, IEEE Transactions on Signal Processing **53** (2005), no. 9, 3488–3502.
[27] J.W Miskin, *Ensemble learning for independent component analysis*, Phd thesis, Cambridge, 2000, `http://www.inference.phy.cam.ac.uk/jwm1003/`.
[28] I. Molchanov, *Tangent sets in the space of measures: with applications to variational analysis*, J. Math. Anal. Appl. **249** (2000), no. 2, 539–552.
[29] I. Molchanov and S. Zuyev, *Steepest descent algorithms in a space of measures*, Statistics and Computing **12** (2002), 115–123.
[30] K. Morton, P. Torrione, and L. Collins, *Variational bayesian learning for mixture autoregressive models with uncertain-order*, IEEE Trans. Signal Processing **59** (2011), no. 6, 2614–2627.
[31] F. Natterer, *Algorithms in tomography*, The State of the Art in Numerical Analysis, Clarendon

Press, duff, i.s. and watson, g.a. ed., 1997.
[32] Y. Nesterov, *Smooth minimization of non-smooth functions*, Mathematic Programming Ser. A **103** (2005), 127–152.
[33] J. Nocedal and S. J. Wright, *Numerical optimization*, Series in Operations Research, Springer Verlag, New York, 2000.
[34] A. Pai, E. Chassande-Mottin, and O. Rabaste, *Best network chirplet chain: Near-optimal coherent detection of unmodeled gravitational wave chirps with a network of detectors*, Phys. Rev. D **77** (2008), no. 062005, 1–22, Also available at gr-qc/0708.3493.
[35] C. Robert, *Simulations par la méthode MCMC*, Economica, Paris, France, 1997.
[36] C. Robert and G. Casella, *Monte-Carlo statistical methods*, Springer Texts in Statistics, Springer, New York, NY, 2000.
[37] W. Rudin, *Real and complex analysis*, McGraw-Hill Book Co., New York, 1987.
[38] M. Seeger, *Bayesian inference and optimal design for sparse linear model*, Jour. of Machine Learning Research **9** (2008), 759–813.
[39] M. Seeger and Nickisch H., *Large scale bayesian inference and experimental design for sparse linear models*, SIAM Journal on Imaging Sciences **4** (2011), no. 1, 166–199.
[40] M. Seeger and D. Wipf, *Variational bayesian inference techniques*, IEEE signal Processing Magazin **27** (2010), no. 6, 81–91.
[41] D. Shutin, T. Buchgraber, S. R. Kulkarni, and H. V. Poor, *Fast adaptive variational sparse Bayesian learning with automatic relevance determination*, Proc. IEEE ICASSP, IEEE, 2011, pp. 2180–2183.
[42] V. Smidl and A. Quinn, *The variational bayes method in signal processing*, Springer, 2006.
[43] ———, *Variational bayesian filtering*, IEEE Trans. Signal Processing **56** (2008), no. 10, 5020–5030.
[44] H. Snoussi and A. Mohammad-Djafari, *Bayesian unsupervised learning for source separation with mixture of gaussians prior*, Int. Journal of VLSI Signal Processing Systems **37** (2004), no. 2-3, 263–279.
[45] Paul Tseng and Dimitri P Bertsekas, *On the convergence of the exponential multiplier method for convex programming*, Mathematical Programming **60** (1993), no. 1-3, 1–19.
[46] M. Wainwright and E. Simoncelli, *Scale Mixtures of Gaussians and the statistics of natural images*, NIPS **12** (2000).
[47] G. Wang and Z. Bao, *Inverse synthetic aperture radar imaging of maneuvering targets based on chirplet decomposition*, Opt. Eng. **38** (1999), no. 1534.
[48] D. P. Wipf and B. D. Rao, *Sparse Bayesian learning for basis selection*, IEEE Trans. on Signal Processing **52** (2004), no. 8, 2153–2164.
[49] M. Zibulevsky and M. Elad, *L1-L2 optimization in signal and image processing*, IEEE Signal Processing Magazine (2010), 76–88.