Efficient Variational Inference in Large-Scale Bayesian Compressed Sensing

George Papandreou[1] and Alan L. Yuille[1,2]
[1]Department of Statistics, University of California, Los Angeles
[2]Department of Brain and Cognitive Engineering, Korea University, Seoul, Korea
[gpapan,yuille]@stat.ucla.edu

Abstract

We study linear models under heavy-tailed priors from a probabilistic viewpoint. Instead of computing a single sparse most probable (MAP) solution as in standard deterministic approaches, the focus in the Bayesian compressed sensing framework shifts towards capturing the full posterior distribution on the latent variables, which allows quantifying the estimation uncertainty and learning model parameters using maximum likelihood. The exact posterior distribution under the sparse linear model is intractable and we concentrate on variational Bayesian techniques to approximate it. Repeatedly computing Gaussian variances turns out to be a key requisite and constitutes the main computational bottleneck in applying variational techniques in large-scale problems. We leverage on the recently proposed Perturb-and-MAP algorithm for drawing exact samples from Gaussian Markov random fields (GMRF). The main technical contribution of our paper is to show that estimating Gaussian variances using a relatively small number of such efficiently drawn random samples is much more effective than alternative general-purpose variance estimation techniques. By reducing the problem of variance estimation to standard optimization primitives, the resulting variational algorithms are fully scalable and parallelizable, allowing Bayesian computations in extremely large-scale problems with the same memory and time complexity requirements as conventional point estimation techniques. We illustrate these ideas with experiments in image deblurring.

1. Introduction

Sparsity: Deterministic and Bayesian viewpoints Sparsity has proven very fruitful in data analysis. Early methods such as total variation (TV) modeling [31], wavelet thresholding [23], sparse coding [26], and independent component analysis [6] have had big impact in signal and image modeling. Recent research in compressed sensing [4, 8] has shown that high-dimensional signals representable with few non-zero coefficients in a linear transform domain are exactly recoverable from a small number of measurements through linear non-adaptive (typically random) operators satisfying certain incoherence properties. Signal recovery in these deterministic models typically reduces to a convex optimization problem and is scalable to problems with millions of variables such as those arising in image analysis.

The deterministic viewpoint on sparsity has certain shortcomings. In real-world applications the theoretical assumptions of compressed sensing are often violated. For example, filter responses of natural images exhibit heavy-tailed marginal histograms but are seldom exactly zero [22]. In practical applications such as image inpainting or deblurring the measurement operators are fixed and do not satisfy the incoherence properties. In these setups it is impossible to exactly reconstruct the underlying latent signal and it is important to quantify the associated estimation uncertainty.

Along these lines, there is a growing number of studies both in the machine learning [1, 10, 19, 36, 39] and the signal processing literature [5, 14] which bring ideas from sparse modeling into a powerful Bayesian statistical approach for describing signals and images. The most distinctive characteristic of Bayesian modeling is that beyond finding the most probable (MAP) solution it also allows us to represent the full posterior distribution on the latent variables, thus capturing the uncertainty in the recovery process. From a practical standpoint, this Bayesian compressed sensing framework allows learning model parameters and devising adaptive measurement designs in a principled way. We employ kurtotic priors for modeling the heavy-tailed nature of filter responses. Beyond sparsity, we can also capture structured statistical dependencies between model variables [38] using tools from probabilistic graphical models [29], yielding methods that more faithfully describe the complex statistical properties of real-world signals.

Variational Bayes for sparse linear models Computing the exact posterior under heavy tailed priors is not tractable. We thus have to contend ourselves with approximate solutions, either of stochastic sampling or deterministic varia-

tional type. In sampling techniques we represent the posterior using random samples drawn by Markov chain Monte-Carlo (MCMC); see [29, 30, 32, 33] for recent related work.

The variational techniques in which we focus in this paper approximate the true posterior distribution with a parameterized Gaussian which allows closed-form computations. Inference amounts to adjusting the variational parameters to make the fit as tight as possible [41]. Mostly related to our work are [1, 10, 19, 36]. There exist multiple alternative criteria to quantify the fit quality, giving rise to approximations such as variational bounding [15], mean field or ensemble learning, and, expectation propagation (EP) [24] (see [2, 28] for discussions about the relations among them), as well as different iterative algorithms for optimizing each specific criterion. These variational criteria involve some sort of integration over the latent variables. We should contrast this with the Laplace approximation [2] which is based on a second-order Taylor expansion around the MAP point estimate and is thus inappropriate for the often non-smooth posterior density under the sparse linear model.

All variational algorithms we study in the paper are of a double-loop nature, requiring Gaussian variance estimation in the outer loop and sparse point estimation in the inner loop [35, 36, 40]. The ubiquity of the Gaussian variance computation routine is not coincidental. Variational approximations try to capture uncertainty in the intractable posterior distribution along the directions of sparsity. These are naturally encoded in the covariance matrix of the proxy Gaussian variational approximation. Marginal Gaussian variance computation is also required in automatic relevance determination algorithms for sparse Bayesian learning [20] and relevance vector machine training [39]; the methods we develop could also be applied in that context.

Variance computation: Lanczos vs. proposed Monte-Carlo algorithm Estimating Gaussian variances is currently the main computational bottleneck and hinders the wider adoption of variational Bayesian techniques in large-scale problems with thousands or millions of variables such as those arising in image analysis, in which explicitly storing or manipulating the full covariance matrix is in general infeasible. Computing variances in Gaussian Markov random fields (GMRFs) with loops is challenging and a host of sophisticated techniques have been developed for this purpose, which often only apply to restricted classes of models [21, 42]. A general-purpose variance computation technique [27, 34] is based on the Lanczos iterative method for solving eigenproblems [11] and has been extensively studied in the variational Bayes context by Seeger and Nickisch [36, 37]. Unless run for a prohibitively large number of iterations, the Lanczos algorithm severely underestimates the required variances, to the extent that Lanczos is inadequate for optimizing criteria like expectation propagation which are sensitive to gross variance estimation errors [35].

The main technical contribution of our work is to demonstrate that the sample-based Monte-Carlo Gaussian variance estimator of [30] performs markedly better than the Lanczos algorithm as the key computational sub-routine in the variational learning context. Our estimator builds on the efficient Perturb-and-MAP sampling algorithm of [30] (c.f. [29, 32]) which draws exact GMRF samples by locally injecting noise to each Gaussian factor independently, followed by computing the mean/mode of the perturbed GMRF by preconditioned conjugate gradients. Being unbiased, the proposed sample estimator does not suffer from the Lanczos systematic underestimation errors. In practice, a few samples suffice for capturing the variances with accuracy sufficient for even the more sensitive expectation propagation algorithm to work reliably. Moreover, correlations (i.e. off-diagonal elements of covariance matrix) needed in certain applications are easy to compute.

The advocated approach to Monte-Carlo variance estimation for variational learning has several other advantages. It is fully scalable, only relying on well-studied computational primitives, thus allowing Bayesian inference with the same memory and time complexity requirements as conventional point estimation. The proposed algorithm is parallelizable, since the required Gaussian samples can be drawn independently on different processors. Further, we show how we can use the samples to estimate the free energy and monitor convergence of the algorithm at no extra cost.

2. Variational Bayes for sparse linear models

2.1. The sparse linear model: Point estimation vs. Bayesian inference

The formulation of the sparse linear model we consider follows the setup of [10, 36]. We consider a hidden vector $\mathbf{x} \in \mathbb{R}^N$ which follows a heavy-tailed prior distribution $P(\mathbf{x})$ and noisy linear measurements $\mathbf{y} \in \mathbb{R}^M$ of it are drawn with Gaussian likelihood $P(\mathbf{y}|\mathbf{x})$. Specifically:

$$P(\mathbf{x};\boldsymbol{\theta}) \propto \prod_{k=1}^{K} t_k(\mathbf{g}_k^T\mathbf{x}), \quad P(\mathbf{y}|\mathbf{x};\boldsymbol{\theta}) = \mathcal{N}(\mathbf{y};\mathbf{H}\mathbf{x},\sigma^2\mathbf{I}), \tag{1}$$

where the K rows of $\mathbf{G} = [\mathbf{g}_1^T;\ldots;\mathbf{g}_K^T]$ and the M rows of $\mathbf{H} = [\mathbf{h}_1^T;\ldots;\mathbf{h}_M^T]$ are two sets of length-N linear filters, the former mapping $\mathbf{x}$ to the domain $\mathbf{s} = \mathbf{G}\mathbf{x}$ in which it exhibits sparse responses and the latter capturing the Gaussian measurement process[1]. The sparsity inducing potentials are denoted by $t_k(s_k)$. The Laplacian $t_k(s_k) = e^{-\tau_k|s_k|}$, $s_k = \mathbf{g}_k^T\mathbf{x}$, is a widely used form for them. In some applications a subset of the model's aspects $(\mathbf{H}, \sigma^2, \mathbf{G})$ can be unknown and dependent on a parameter vector $\boldsymbol{\theta}$; e.g., in

[1] $\mathcal{N}(\mathbf{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = |2\pi\boldsymbol{\Sigma}|^{-1/2}\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$ is the multivariate Gaussian density on $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$.

blind image deconvolution $\boldsymbol{\theta}$ typically is the unknown blurring kernel $\mathbf{k}$ which determines the measurement matrix $\mathbf{H}$.

By Bayes' rule, the posterior distribution of the latent variables $\mathbf{x}$ given $\mathbf{y}$ has the non-Gaussian density

$$P(\mathbf{x}|\mathbf{y}) = Z^{-1}(\boldsymbol{\theta})P(\mathbf{y}|\mathbf{x})\prod_{k=1}^{K} t_k(s_k)\,, \quad \text{where} \tag{2}$$

$$Z(\boldsymbol{\theta}) \triangleq P(\mathbf{y};\boldsymbol{\theta}) = \int P(\mathbf{y}|\mathbf{x})\prod_{k=1}^{K} t_k(s_k)\,d\mathbf{x} \tag{3}$$

is the evidence/ partition function.

Point estimation corresponding to standard compressed sensing amounts to finding the posterior MAP configuration $\hat{\mathbf{x}}_{\text{MAP}} \triangleq \operatorname{argmax}_{\mathbf{x}} \log P(\mathbf{x}|\mathbf{y})$, leading to minimization of

$$\phi_{\text{MAP}}(\mathbf{x}) = \sigma^{-2}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2 - 2\sum_{k=1}^{K}\log t_k(s_k)\,. \tag{4}$$

Point estimation thus reduces to a standard optimization problem and a host of modern techniques have been developed for solving it, scalable to large-scale applications. However, since it ignores the partition function, point estimation neither provides information about the estimation uncertainty nor allows parameter estimation.

In the Bayesian framework we try to overcome these shortcomings by capturing the full posterior distribution. Since it is intractable to manipulate it directly, we consider variational approximations of Gaussian form

$$\begin{aligned} Q(\mathbf{x}|\mathbf{y}) \propto P(\mathbf{y}|\mathbf{x})e^{\boldsymbol{\beta}^T\mathbf{s}-\frac{1}{2}\mathbf{s}^T\boldsymbol{\Gamma}^{-1}\mathbf{s}} = \mathcal{N}(\mathbf{x};\hat{\mathbf{x}}_Q,\mathbf{A}^{-1})\,, \text{ with} \\ \hat{\mathbf{x}}_Q = \mathbf{A}^{-1}\mathbf{b}\,, \quad \mathbf{A} = \sigma^{-2}\mathbf{H}^T\mathbf{H} + \mathbf{G}^T\boldsymbol{\Gamma}^{-1}\mathbf{G}\,, \\ \boldsymbol{\Gamma} = \operatorname{diag}(\boldsymbol{\gamma})\,, \quad \text{and} \quad \mathbf{b} = \sigma^{-2}\mathbf{H}^T\mathbf{y} + \mathbf{G}^T\boldsymbol{\beta}\,. \end{aligned} \tag{5}$$

The implied form for the variational evidence is

$$Z_Q(\boldsymbol{\theta}) \triangleq Q(\mathbf{y};\boldsymbol{\theta}) = \int P(\mathbf{y}|\mathbf{x})e^{\boldsymbol{\beta}^T\mathbf{s}-\frac{1}{2}\mathbf{s}^T\boldsymbol{\Gamma}^{-1}\mathbf{s}}d\mathbf{x}\,. \tag{6}$$

Our task in variational learning is to adjust the set of variational parameters $\boldsymbol{\xi} = (\boldsymbol{\beta},\boldsymbol{\gamma})$ so as to improve the fit of the approximating Gaussian to the true posterior distribution.

We will mostly be focusing on log-concave sparsity inducing potentials $t_k(\cdot)$ – i.e., $\log t_k(\cdot)$ is concave – such as the Laplacian. This guarantees that the posterior $P(\mathbf{x}|\mathbf{y})$ is also log-concave in $\mathbf{x}$, and thus point estimation in Eq. (4) is a convex optimization problem. Log-concavity also implies that $P(\mathbf{x}|\mathbf{y})$ is unimodal and justifies approximating it with a Gaussian $Q(\mathbf{x}|\mathbf{y})$ in Eq. (5).

2.2. Variational bounding

Variational bounding [10, 15, 28, 36] is applicable to sparsity-inducing potentials of super-Gaussian form. The family of even super-Gaussian potentials is quite rich and superset of the family of mixtures of zero-mean Gaussians; it includes the Laplacian and the Student as members [28]. Super-Gaussian potentials have a useful dual representation

$$t_k(s_k) = \sup_{\gamma_k>0} e^{-s_k^2/(2\gamma_k)-h_k(\gamma_k)/2}\,, \quad \text{with} \tag{7}$$

$$h_k(\gamma_k) \triangleq \sup_{s_k} -s_k^2/\gamma_k - 2\log t_k(s_k) \tag{8}$$

Variational bounding amounts to replacing the potentials $t_k(s_k)$ in Eq. (2) with these bounds and tuning the variational parameters $\boldsymbol{\gamma}$ ($\boldsymbol{\beta}$ is fixed to zero in this case) so as the variational evidence lower bounds as tightly as possible the exact evidence $Z \geq Z_Q$. This leads to the variational free energy minimization problem (see [36] for the derivation) $\inf_{\boldsymbol{\gamma}\succ\mathbf{0}}\phi_Q(\boldsymbol{\gamma})$, where

$$\phi_Q(\boldsymbol{\gamma}) = \log|\mathbf{A}| + h(\boldsymbol{\gamma}) + \inf_{\mathbf{x}} R(\mathbf{x},\boldsymbol{\gamma})\,, \tag{9}$$

with $h(\boldsymbol{\gamma}) \triangleq \sum_{k=1}^{K} h_k(\gamma_k)$ and $R(\mathbf{x},\boldsymbol{\gamma}) \triangleq \sigma^{-2}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2 + \mathbf{s}^T\boldsymbol{\Gamma}^{-1}\mathbf{s}$. The $\mathbf{A}$ and $\mathbf{b}$ are given in Eq. (5); note that $\mathbf{A}$ is a function of $\boldsymbol{\gamma}$.

The log-determinant term in Eq. (9) is what makes Bayesian variational inference more interesting and at the same time computationally more demanding than point estimation. Indeed, using Eq. (7), we can re-write the objective function for MAP estimation (4) as $\phi_{\text{MAP}}(\mathbf{x}) = \inf_{\boldsymbol{\gamma}\succ\mathbf{0}} h(\boldsymbol{\gamma}) + R(\mathbf{x},\boldsymbol{\gamma})$, showing that ϕ_{MAP} and ϕ_Q only differ in the $\log|\mathbf{A}|$ term, which endows variational inference with the ability to capture the effect of the partition function. The difficulty lies in the fact that the elements of the vector $\boldsymbol{\gamma}$ are interleaved in $\log|\mathbf{A}|$. Following [28, 36], we can decouple the problem by exploiting the concavity of $\log|\mathbf{A}|$ as a function of $\boldsymbol{\gamma}^{-1} \triangleq (\gamma_1^{-1},\ldots,\gamma_K^{-1})$. Fenchel duality then yields the upper bound $\log|\mathbf{A}| \leq \mathbf{z}^T\boldsymbol{\gamma}^{-1} - g^*(\mathbf{z})$, $\mathbf{z} \succ \mathbf{0}$. For given $\boldsymbol{\gamma}$ the bound becomes tight for $\mathbf{z} = \nabla_{\boldsymbol{\gamma}^{-1}}\log|\mathbf{A}| = \operatorname{diag}(\mathbf{G}\mathbf{A}^{-1}\mathbf{G}^T)$, which can be identified as the vector of marginal variances $z_k = \operatorname{Var}_Q(s_k|\mathbf{y})$ along the directions $s_k = \mathbf{g}_k^T\mathbf{x}$ under the variational posterior $Q(\mathbf{x}|\mathbf{y})$ with the current guess for the parameters $\boldsymbol{\gamma}$.

This approach naturally suggests a **double-loop** algorithm, globally convergent when the potentials t_k are log-concave [28, 36]. In the *outer loop*, we compute the vector of marginal variances $\mathbf{z}$ so as to tighten the upper bound to $\log|\mathbf{A}|$, given the current value of $\boldsymbol{\gamma}$.

In the *inner loop*, instead of ϕ_Q in Eq. (9) we minimize w.r.t. $\mathbf{x}$ and $\boldsymbol{\gamma}$ the upper bound given the newly computed $\mathbf{z}$

$$\begin{aligned} \bar{\phi}_Q(\mathbf{x},\boldsymbol{\gamma};\mathbf{z}) &= \mathbf{z}^T\boldsymbol{\gamma}^{-1} + h(\boldsymbol{\gamma}) + R(\mathbf{x},\boldsymbol{\gamma}) \\ &= \sigma^{-2}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2 + \sum_{k=1}^{K}\left(\frac{s_k^2+z_k}{\gamma_k} + h_k(\gamma_k)\right)\,. \end{aligned} \tag{10}$$

We can minimize this expression explicitly w.r.t. $\boldsymbol{\gamma}$ by noting that it is decoupled in the γ_k and recalling from (7) that

$-2\log t_k(s_k) = \inf_{\gamma_k>0} s_k^2/\gamma_k + h_k(\gamma_k)$. This leaves us with a minimization problem w.r.t. $\mathbf{x}$ alone

$$\bar{\phi}_Q(\mathbf{x};\mathbf{z}) = \inf_{\boldsymbol{\gamma}\succ\mathbf{0}} \bar{\phi}_Q(\mathbf{x},\boldsymbol{\gamma};\mathbf{z}) =$$

$$= \sigma^{-2}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2 - 2\sum_{k=1}^{K}\log t_k\left((s_k^2+z_k)^{1/2}\right). \quad (11)$$

This is just a smoothed version of the MAP point estimation problem (4), also convex when t_k are log-concave, which we minimize in the course of the inner loop with standard quasi-Newton methods [3] to obtain the variational mean $\hat{\mathbf{x}}$. After completion of the inner loop, we recover the minimizing values for the variational parameters $\gamma_k^{-1} = -2\frac{d\log t_k(\sqrt{v})}{dv}\Big|_{v=\hat{s}_k^2+z_k}$, with which we update the vector of marginal variances $\mathbf{z}$ in the subsequent outer loop iteration [36].

2.3. Mean field and expectation propagation

Bounding is not the only way to construct variational approximations to the intractable posterior distribution $P(\mathbf{x}|\mathbf{y})$. The mean field (or ensemble learning) approach amounts to assuming a simplified parametric form Q for the posterior distribution and adjusting the corresponding variational parameters $\boldsymbol{\xi}$ so as to minimize the KL-divergence $D_{KL}(Q||P)$ between Q and P [1]. See [18] for a recent application of the mean field approximation to the problem of image deconvolution, where it is shown that the mean field updates reduce to point estimation and variance computation primitives, exactly as in the variational bounding approximation discussed in detail in Sec. 2.2.

Expectation propagation (EP) is yet another powerful variational approximation criterion, in which the variational parameters of the approximating distribution Q are adjusted so as expectations under Q and the true posterior $P(\mathbf{x}|\mathbf{y})$ are matched [24]. There are various iterative sequential message passing-like algorithms for optimizing the EP criterion. Applying EP to large-scale problems in which our paper focuses is challenging. We will employ the parallel update algorithm of [40], but our methods are also applicable to the recently proposed provably convergent double-loop algorithm of [35]. Once more, variance estimation in the outer loop is the computational bottleneck; see [35, 40].

3. Monte-Carlo posterior variance estimation

As highlighted in Sec. 2, repeatedly computing posterior Gaussian variances turns out to be a key computational routine in all variational approximations of the sparse linear model. With reference to Eq. (5), our goal is to compute certain elements of the covariance matrix $\boldsymbol{\Sigma} \triangleq \mathbf{A}^{-1}$ or marginal variances $z_k = \mathrm{Var}_Q(s_k|\mathbf{y})$ along certain projections $s_k = \mathbf{g}_k^T\mathbf{x}$ under the variational posterior $Q(\mathbf{x}|\mathbf{y})$. Note that $\boldsymbol{\Sigma}$ is a fully dense $N\times N$ matrix. Thus for large-scale models comprising $N \approx 10^6$ variables it is impossible to compute or store the full $\boldsymbol{\Sigma}$ explicitly.

3.1. Lanczos variance estimation

So far, the main candidate for variance estimation in the context of large-scale variational Bayes has been the Lanczos iterative method [36, 37]. As the iteration progresses, the Lanczos algorithm builds a monotonically increasing estimate for the variances [11]. It can reveal in relatively few iterations the rough structure and relative magnitude of variances, but requires a very large number of iterations to accurately approximate their absolute values. Since it scales badly with the number of iterations N_L (its complexity is $\mathcal{O}(N_L^2)$ in time and $\mathcal{O}(N_L)$ in memory due to a required reorthogonalization step), it is only practical to run Lanczos for a relatively small number of iterations, yielding gross underestimates for the variances.

In practice, variational bounding has proven relatively robust to the Lanczos crude variance estimates [36, 37], while expectation propagation completely fails [35]. This starkly contrasting qualitative behavior in the two cases can be explained as follows: In the presence of Lanczos variance underestimation errors, the expression (10) remains an upper bound of (9), albeit not tight any more. Moreover, the variational optimization problem (11) gracefully degrades to the point estimation problem (4) when $0 \le \hat{z}_k \ll z_k$. In other words, despite the variance errors the algorithm does not collapse, although it effectively ends up solving a modified inference problem rather than the one that it was supposed to solve. In contrast, expectation propagation works by moment matching and the gross variance estimation errors make iterative EP algorithms hopelessly break down.

3.2. Efficient Monte-Carlo variance estimation with Perturb-and-MAP sampling

We propose estimating variances using a sampling-based Monte-Carlo technique, leveraging on the efficient Perturb-and-MAP GMRF sampling algorithm of [30]. Although [30] has already suggested this possibility, it has not explored its effectiveness in the variational Bayesian context.

The algorithm of [30] reduces GMRF sampling into a GMRF mean estimation problem. In our notation, an exact Gaussian sample $\tilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{0}, \mathbf{A}^{-1})$, with $\mathbf{A} = \sigma^{-2}\mathbf{H}^T\mathbf{H} + \mathbf{G}^T\boldsymbol{\Gamma}^{-1}\mathbf{G}$, can be drawn by solving the linear system

$$\mathbf{A}\tilde{\mathbf{x}} = \sigma^{-2}\mathbf{H}^T\tilde{\mathbf{y}} + \mathbf{G}^T\tilde{\boldsymbol{\beta}}. \quad (12)$$

The local perturbations $\tilde{\mathbf{y}} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ and $\tilde{\boldsymbol{\beta}} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma}^{-1})$ are trivial to sample since $\boldsymbol{\Gamma}$ is diagonal. We efficiently solve the linear system (12) using preconditioned conjugate gradients (PCG) [11], employing filtering routines for fast evaluation of matrix-vector products $\mathbf{A}\mathbf{x}$, thus

avoiding the costly Cholesky factorization step typically associated with Gaussian simulation. In contrast to Lanczos, the memory footprint of PCG is small as only 4 length-N vectors need to be stored, while multiple samples can be trivially drawn in parallel (using, e.g., `parfor` in Matlab). Also note that, unlike conjugate gradients, employing preconditioning within Lanczos variance estimation is difficult [34] and seldom used in practice.

Having drawn N_s Gaussian samples as described, we employ the standard sample-based covariance estimators

$$\hat{\mathbf{\Sigma}} = \frac{1}{N_s}\sum_{i=1}^{N_s} \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^T\,, \quad \hat{z}_k = \frac{1}{N_s}\sum_{i=1}^{N_s} \tilde{s}_{k,i}^2\,, \qquad (13)$$

with $\tilde{s}_{k,i} \triangleq \mathbf{g}_k^T \tilde{\mathbf{x}}_i$. The variance estimates marginally follow scaled chi-square distributions with N_s degrees of freedom $\hat{z}_k \sim \frac{z_k}{N_s}\chi^2(N_s)$. This implies that $E\{\hat{z}_k\} = z_k$, i.e., this estimator is unbiased, unlike the Lanczos one. Its relative error is $r = \Delta(\hat{z}_k)/z_k = \sqrt{\mathrm{Var}(\hat{z}_k)}/z_k = \sqrt{2/N_s}$, independent from the problem size N. The error drops quite slowly with the number of samples ($N_s = 2/r^2$ samples are required to reach a desired relative error r), but variance estimates sufficiently accurate for even the more sensitive expectation propagation algorithm to work reliably can be obtained after about 20 samples (which translates to $r \approx 32\%$). One can show that $z_k \leq \gamma_k^{-1}$ [36], a consequence of the fact that measurements always reduce the uncertainty in Gaussian models. To enforce this important structural constraint, we use in place of (13) the clipped estimator $\bar{z}_k = \min(\hat{z}_k, \gamma_k^{-1})$ which behaves considerably better in practice while still being (asymptotically) unbiased.

To illustrate the efficiency of the proposed Monte-Carlo variance estimator in the context of variational Bayesian inference, we compare in Fig. 1 the marginal variances obtained by our sample-based estimator with that of Lanczos. The system matrix $\mathbf{A}$ for this particular example is the one of the last iteration of the double-loop variational bounding algorithm of Sec. 2.2 applied to a small-scale 48×73 deblurring problem for which it is feasible to compute the exact marginal variances z_k. We use the clipped version $\bar{z}_k$ of our estimator with $N_s = 20$ samples, each drawn by solving the linear system (12) with 20 PCG iterations as detailed in Sec. 4. Lanczos was run for $N_L = 300$ iterations, so as the runtime for the two algorithms to be the same. We see that the proposed sample-based variance estimator performs markedly better than Lanczos, which grossly underestimates the true marginal variances. Note that for large-scale problems the performance gap will be even more pronounced: as we showed earlier, the relative estimation accuracy r of the sample-based estimator is independent of the latent space dimensionality N, while the relative accuracy of Lanczos further deteriorates for large N [36, Fig. 6].

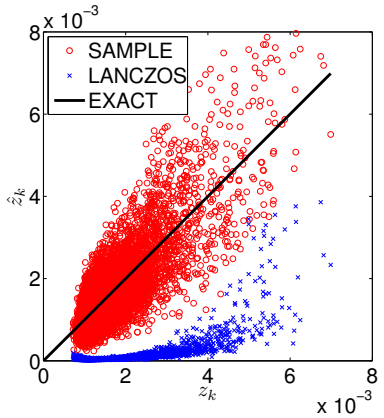


Figure 1. Scatter-plot of exact z_k vs. estimated $\hat{z}_k$ marginal variances for a small-scale deblurring problem. We compare the the proposed sample-based Monte-Carlo estimator with Lanczos.

3.3. Monte-Carlo free energy estimation

To monitor convergence of the free energy (9) and for debugging purposes, it is desirable to estimate $\log|\mathbf{A}|$ during the course of the algorithm. Note that this step is not a requisite for the variational algorithm to yield estimates for $\mathbf{x}$ or estimate the model parameters $\boldsymbol{\theta}$.

By coercing information from the samples $\tilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{0}, \mathbf{A}^{-1})$ drawn for variance estimation, we can reliably estimate $\log|\mathbf{A}|$ at no extra cost, provided that we can analytically compute $\log|\mathbf{P}|$ for some matrix $\mathbf{P}$ that approximates $\mathbf{A}$ well, typically the preconditioner employed by PCG for solving (12). To see this, note that $E\left\{\exp\left(0.5\tilde{\mathbf{x}}^T(\mathbf{A}-\mathbf{P})\tilde{\mathbf{x}}\right)\right\} = |\mathbf{A}|/|\mathbf{P}|$, which suggests the Monte-Carlo estimator

$$\log|\mathbf{A}| \approx \log|\mathbf{P}| - \log N_s + \log\left(\sum_{i=1}^{N_s} 0.5\tilde{\mathbf{x}}_i^T(\mathbf{A}-\mathbf{P})\tilde{\mathbf{x}}_i\right). \qquad (14)$$

A special case of this with $\mathbf{P} = \mathbf{I}$ has been proposed before [7], but for the large-scale problems we consider here using a good reference $\mathbf{P} \approx \mathbf{A}$ is crucial for the estimator (14) to exhibit low variance and thus be useful in practice.

4. Applications to image deconvolution

Our main motivation for this work is solving inverse problems in image analysis and low-level vision such as image deblurring, inpainting, and tomographic reconstruction. These give rise to large-scale inference problems involving millions of variables. We report experimental results on image deconvolution. Our software builds on the `glm-ie` Matlab toolbox [25] designed for variational inference under the variational bounding [36] and expectation propagation [40] criteria, which we have extended to include implementations of the proposed algorithms; our extensions will be integrated in future releases of `glm-ie`.

In image deblurring [12, 13], our goal is to recover the sharp image $\mathbf{x}$ from its blurred version $\mathbf{y}$. We assume a spatially homogeneous degradation, typically due to camera or subject motion, captured by the measurement process $\mathbf{y} = \mathbf{H}\mathbf{x} \triangleq \mathbf{k} * \mathbf{x}$. In the non-blind variant of the problem, the convolution blur kernel $\mathbf{k}$ is considered known (the problem is classically known as image restoration), while in the more challenging blind variant our goal is to recover both the sharp image and the unknown blurring kernel.

Blind image deconvolution In the blind deconvolution case, the blurring kernel is considered as parameter, $\boldsymbol{\theta} = \mathbf{k}$, which we recover by maximum (penalized) likelihood. It is crucial to determine $\mathbf{k}$ by first integrating out the latent variables $\mathbf{x}$ and then maximizing the marginal likelihood $\operatorname{argmax}_{\mathbf{k}} P(\mathbf{y}; \mathbf{k})$, instead of maximizing the joint likelihood $\operatorname{argmax}_{\mathbf{k}} (\max_{\mathbf{x}} P(\mathbf{x}, \mathbf{y}; \mathbf{k}))$ [9,17]. Under the variational approximation, we use $Q(\mathbf{y}; \mathbf{k})$ from (6) in place of $P(\mathbf{y}; \mathbf{k})$. Following [10, 18], we carry out the optimization iteratively using expectation-maximization (EM).

In the *E-step*, given the current estimate $\mathbf{k}^t$ for the blurring kernel, we perform variational Bayesian inference as described in Sec. 2. In the *M-step* of the t-th iteration, we maximize w.r.t. $\mathbf{k}$ the expected complete log-likelihood $E_{\mathbf{k}^t}\{\log Q(\mathbf{x}, \mathbf{y}; \mathbf{k})\}$, with expectations taken w.r.t. $Q(\mathbf{x}|\mathbf{y}; \mathbf{k}^t)$. The updated kernel $\mathbf{k}^{t+1}$ is obtained by minimizing w.r.t. $\mathbf{k}$ (see [18] for the derivation)

$$\begin{aligned} & E_{\mathbf{k}^t}\left\{\frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2\right\} \\ &= \frac{1}{2}\operatorname{tr}\left((\mathbf{H}^T\mathbf{H})(\mathbf{A}^{-1} + \hat{\mathbf{x}}\hat{\mathbf{x}}^T)\right) - \mathbf{y}^T\mathbf{H}\hat{\mathbf{x}} + (\text{const}) \\ &= \frac{1}{2}\mathbf{k}^T\mathbf{R}_{\mathbf{xx}}\mathbf{k} - \mathbf{r}_{\mathbf{xy}}^T\mathbf{k} + (\text{const})\,, \end{aligned} \tag{15}$$

which is a quadratic program in $\mathbf{k}$; see [18] for the formulas for $\mathbf{r}_{\mathbf{xy}}$ and $\mathbf{R}_{\mathbf{xx}}$. The entries in $\mathbf{r}_{\mathbf{xy}}$ accumulate cross-correlations between $\hat{\mathbf{x}}$ and $\mathbf{y}$; we use the variational mean $\hat{\mathbf{x}}$ of (11) for computing them. The entries in $\mathbf{R}_{\mathbf{xx}}$ capture second-order information for $\mathbf{x}$ under $Q(\mathbf{x}|\mathbf{y}; \mathbf{k}^t)$; we estimate them efficiently by drawing a small number of samples (1 or 2 suffice) from $\mathcal{N}(\mathbf{0}, \mathbf{A}^{-1})$, exactly as in Sec. 3.2. Note that [18] estimates $\mathbf{R}_{\mathbf{xx}}$ by making the simplifying assumption that $\mathbf{A}$ is diagonal, which could potentially lead to a poor approximation. We add to (15) an extra L_1 penalty term $\lambda_1\|\mathbf{k}\|_{L_1}$ so as to favor sparse kernels.

It is important to note that while the M-step update for $\mathbf{k}$ in (15) is a convex optimization problem, the overall log-likelihood objective $-\log Q(\mathbf{y}; \mathbf{k})$ is not convex in $\mathbf{k}$. This means that the EM algorithm can get stuck to local minima. Various techniques have been developed to mitigate this fundamental problem, such as coarse-to-fine kernel recovery, gradient domain processing, or regularization of the result after each kernel update with (15) – see [9, 17]. We have not yet incorporated these heuristics into our blind deconvolution implementation, and thus our software may still give unsatisfactory results when the spatial support of the unknown blurring kernel is large.

Efficient circulant preconditioning Our sample-based variance estimator described in Sec. 3.2 requires repeatedly drawing samples $\tilde{\mathbf{x}}$. For each of the samples we solve by PCG a linear system of the form $\mathbf{A}\tilde{\mathbf{x}} = \tilde{\mathbf{c}}$, where $\tilde{\mathbf{c}}$ is the randomly perturbed right hand side in Eq. (12).

The system matrix $\mathbf{A} = \sigma^{-2}\mathbf{H}^T\mathbf{H} + \mathbf{G}^T\boldsymbol{\Gamma}^{-1}\mathbf{G}$ arising in image deblurring is typically poorly conditioned, slowing the convergence of plain conjugate gradients. The key to designing an effective preconditioner for $\mathbf{A}$ is to note that $\mathbf{A}$ would be a stationary operator if $\boldsymbol{\Gamma} = \bar{\gamma}\mathbf{I}$, i.e., the variational parameters γ_k were homogeneous. Following [16], we select as preconditioner the stationary approximation of the system matrix, $\mathbf{P} = \sigma^{-2}\mathbf{H}^T\mathbf{H} + \bar{\gamma}^{-1}\mathbf{G}^T\mathbf{G}$, with $\bar{\gamma}^{-1} \triangleq (1/K)\sum_{k=1}^{K}\gamma_k^{-1}$. One can prove that $\mathbf{P}$ is the stationary matrix nearest to $\mathbf{A}$ in the Frobenius norm, i.e. $\mathbf{P} = \operatorname{argmin}_{\mathbf{X}\in\mathcal{C}}\|\mathbf{X} - \mathbf{A}\|$, where $\mathcal{C}$ is the set of stationary (block-circulant with circulant blocks) matrices [16]. Thanks to its stationarity, $\mathbf{P}$ is diagonalized in the Fourier domain; by employing the 2-D DFT, we can compute very efficiently expressions of the form $\mathbf{P}^{-1}\mathbf{x}$ required by PCG [12]. Moreover, $\log|\mathbf{P}|$ is also readily computable in the Fourier domain, allowing us to use the efficient free energy estimator (14) for monitoring convergence. Note that the applicability of this preconditioner extends beyond our variance estimation setup; e.g. it could be employed in conjunction with the MCMC-based deblurring algorithm of [33].

Circulant preconditioning with $\mathbf{P}$ dramatically accelerates convergence of conjugate gradients. We plot in Fig. 2 the residual in the course of conjugate gradient iteration for a typical system matrix $\mathbf{A}$ arising in deblurring a 190×289 image under the variational bounding approximation. With circulant preconditioning (PCG) we attain within only 10 iterations the same level of accuracy that is reached after 100 iterations of unpreconditioned conjugate gradients (CG). This substantial improvement in the convergence rate more than compensates the roughly 60% time overhead per iteration of PCG relative to CG (respectively, 80 vs. 50 msec per iteration on this problem). We are not aware of any work that similarly exploits the benefits of preconditioning in the context of Lanczos variance estimation.

Image deblurring results We have carried out preliminary image deblurring experiments using the dataset of [17] which contains images degraded by real blur due to camera motion, as well as their sharp versions shot with the camera still. We assume a total-variation prior, which implies simple first-order finite difference filters as rows of $\mathbf{G}$ and Laplacian sparsity inducing potentials $t_k(s_k) = e^{-\tau_k|s_k|}$.

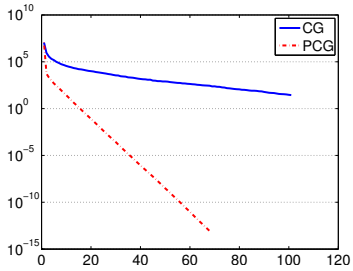


Figure 2. Conjugate gradients residual norm as function of iteration count; No (CG) vs. circulant (PCG) preconditioner.

We fix $\tau_k = 15$ which roughly matches the image derivative scale for typical images with values between 0 and 1. We set the noise variance to $\sigma^2 = 10^{-5}$.

We employ the double-loop algorithms described in Sec. 2 for both the variational bounding (VB) and expectation propagation (EP). We use 20 samples for variance estimation, and allow 20 PCG iterations for solving each of the linear systems (12). We show the deblurred images from both the VB and EP algorithms in Fig. 3 for both the non-blind and blind scenaria. Note that EP completely breaks down if we use the Lanczos variance estimator, while it reliably works under our sample-based variance estimator.

5. Discussion

We have shown that marginal variances required by variational Bayesian algorithms can be effectively estimated using random sampling. This allows applying variational Bayesian inference to large-scale problems, essentially at the same cost as point estimation. The proposed variance estimator can be thought as a stochastic sub-routine in the otherwise deterministic variational framework.

Interestingly, efficient Perturb-and-MAP random sampling turns out to be a key component in both the proposed approach to variational inference and recent MCMC techniques [29, 30, 32, 33]. Systematically comparing these two alternative Bayesian inference alternatives in large-scale applications arises as an interesting topic for future work.

Acknowledgments

This work was supported by the U.S. Office of Naval Research under the MURI grant N000141010933 and by the Korean Ministry of Education, Science, and Technology, under the National Research Foundation WCU program R31-10008. We thank S. Lefkimmiatis for suggesting the circulant preconditioner of Sec. 4, M. Seeger for his feedback on an earlier version of the paper, and H. Nickisch for making the `glm-ie` toolbox publicly available.

References

[1] H. Attias. Independent factor analysis. *Neur. Comp.*, 11:803–851, 1999.

[2] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[3] R. Byrd, P. Lu, and J. Nocedal. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. and Statist. Comp.*, 16(5):1190–1208, 1995.

[4] E. Candes and T. Tao. Decoding by linear programming. *IEEE Trans. Inf. Theory*, 51(12):4203–4215, Dec. 2005.

[5] V. Cevher, P. Indyk, L. Carin, and R. Baraniuk. Sparse signal recovery and acquisition with graphical models. *IEEE Signal Process. Mag.*, 27(6):92–103, Nov. 2010.

[6] P. Comon. Independent component analysis, a new concept? *Signal Processing*, 36(3):287–314, 1994.

[7] N. Cressie, O. Perrin, and C. Thomas-Agnan. Likelihood-based estimation for Gaussian MRFs. *Stat. Meth.*, 2(1):1–16, 2005.

[8] D. L. Donoho. Compressed sensing. *IEEE Trans. Inf. Theory*, 52(4):1289–1306, Apr. 2006.

[9] R. Fergus, B. Singh, A. Hertzmann, S. Roweis, and W. Freeman. Removing camera shake from a single photograph. *Proc. SIGGRAPH*, 25(3):787–794, 2006.

[10] M. Girolami. A variational method for learning sparse and overcomplete representations. *Neur. Comp.*, 13:2517–2532, 2001.

[11] G. Golub and C. Van Loan. *Matrix Computations*. John Hopkins Press, 1996.

[12] P. Hansen, J. Nagy, and D. O'Leary. *Deblurring images: matrices, spectra, and filtering*. SIAM, 2006.

[13] A. Jain. *Fundamentals of digital image processing*. Prentice Hall, 1989.

[14] S. Ji, Y. Xue, and L. Carin. Bayesian compressive sensing. *IEEE Trans. Signal Process.*, 56(6):2346–2356, June 2008.

[15] M. Jordan, J. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.

[16] S. Lefkimmiatis, A. Bourquard, and M. Unser. Hessian-based norm regularization for image restoration with biomedical applications. *IEEE Trans. Image Process.*, 2012. to appear.

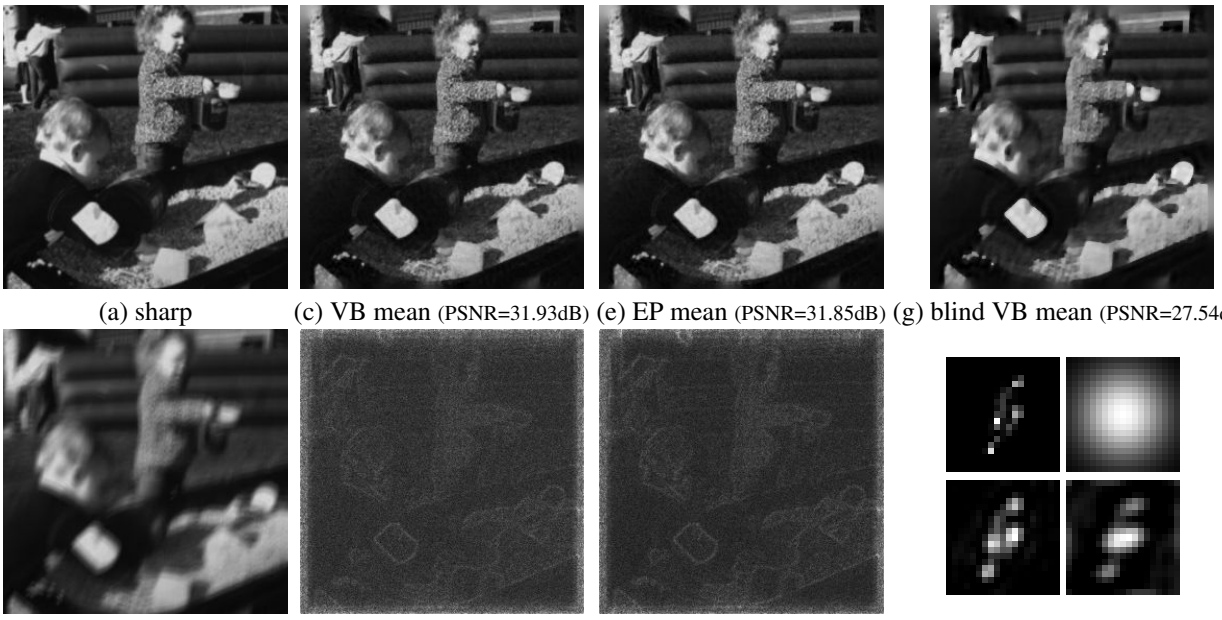
[17] A. Levin, Y. Weiss, F. Durand, and W. Freeman. Understanding and evaluating blind deconvolution algorithms. In *Proc. CVPR*, pages 1964–1971, 2009.

[18] A. Levin, Y. Weiss, F. Durand, and W. Freeman. Efficient marginal likelihood optimization in blind deconvolution. In *Proc. CVPR*, pages 2657–2664, 2011.

[19] M. Lewicki and T. Sejnowski. Learning overcomplete representations. *Neur. Comp.*, 12:337–365, 2000.

[20] D. MacKay. Bayesian interpolation. *Neur. Comp.*, 4(3):415–447, 1992.

[21] D. Malioutov, J. Johnson, M. Choi, and A. Willsky. Low-rank variance approximation in GMRF models: Single and multiscale approaches. *IEEE Trans. Signal Process.*, 56(10):4621–4634, Oct. 2008.



(a) sharp (c) VB mean (PSNR=31.93dB) (e) EP mean (PSNR=31.85dB) (g) blind VB mean (PSNR=27.54dB)

(b) blurred (PSNR=22.57dB) (d) VB stdev (f) EP stdev (h) kernel evolution

Figure 3. Image deblurring experiment with the proposed algorithms. (a) Sharp 255×255 image. (b) Real blurred image. Posterior mean and pointwise estimation uncertainty for non-blind image deblurring under the variational bounding (c, d) and expectation propagation (e, f) criteria. (g) Blind image deblurring with variational bounding. (h) clockwise from upper left, ground-truth 19×19 blurring kernel, initialization, and estimated kernels after the first and the final tenth EM iteration. The model estimates the image values in an extended 273×273 domain, but we only take the central 255×255 area into account when calculating the PSNR.

[22] S. Mallat. A theory for multiresolution signal decomposition: The wavelet transform. *IEEE Trans. PAMI*, 11(7):674–693, 1989.

[23] S. Mallat. *A Wavelet Tour of Signal Processing*. Acad. Press, 2 edition, 1999.

[24] T. Minka. Expectation propagation for approximate bayesian inference. In *Proc. UAI*, 2001.

[25] H. Nickisch. The generalised linear models inference and estimation toolbox (glm-ie v. 1.3). `http://mloss.org/software/view/269`.

[26] B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[27] C. Paige and M. Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Trans. on Math. Soft.*, 8(1):43–71, 1982.

[28] J. Palmer, D. Wipf, K. Kreutz-Delgado, and B. Rao. Variational EM algorithms for non-gaussian latent variable models. In *Proc. NIPS*, 2005.

[29] G. Papandreou, P. Maragos, and A. Kokaram. Image inpainting with a wavelet domain hidden Markov tree model. In *Proc. ICASSP*, pages 773–776, 2008.

[30] G. Papandreou and A. Yuille. Gaussian sampling by local perturbations. In *Proc. NIPS*, 2010.

[31] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[32] U. Schmidt, Q. Gao, and S. Roth. A generative perspective on MRFs in low-level vision. In *Proc. CVPR*, 2010.

[33] U. Schmidt, K. Schelten, and S. Roth. Bayesian deblurring with integrated noise estimation. In *Proc. CVPR*, pages 2625–2632, 2011.

[34] M. Schneider and A. Willsky. Krylov subspace estimation. *SIAM J. Sci. Comp.*, 22(5):1840–1864, 2001.

[35] M. Seeger and H. Nickisch. Fast convergent algorithms for expectation propagation approximate bayesian inference. In *Proc. AISTATS*, 2011.

[36] M. Seeger and H. Nickisch. Large scale bayesian inference and experimental design for sparse linear models. *SIAM J. Imaging Sci.*, 4(1):166–199, 2011.

[37] M. Seeger, H. Nickisch, R. Pohmann, and B. Schölkopf. Bayesian experimental design of magnetic resonance imaging sequences. In *Proc. NIPS*, pages 1441–1448, 2008.

[38] E. P. Simoncelli. Statistical modeling of photographic images. In A. Bovik, editor, *Handbook of Video and Image Processing*, chapter 4.7. Academic Press, 2 edition, 2005.

[39] M. Tipping. Sparse Bayesian learning and the relevance vector machine. *J. of Mach. Learn. Res.*, 1:211–244, 2001.

[40] M. van Gerven, B. Cseke, F. de Lange, and T. Heskes. Efficient Bayesian multivariate fMRI analysis using a sparsifying spatio-temporal prior. *NeuroImage*, 50:150–161, 2010.

[41] M. Wainwright and M. Jordan. Graphical models, exponential families, and variational inference. *Found. and Trends in Machine Learning*, 1(1-2):1–305, 2008.

[42] Y. Weiss and W. Freeman. Correctness of belief propagation in Gaussian graphical models of arbitrary topology. *Neur. Comp.*, 13(10):2173–2200, 2001.