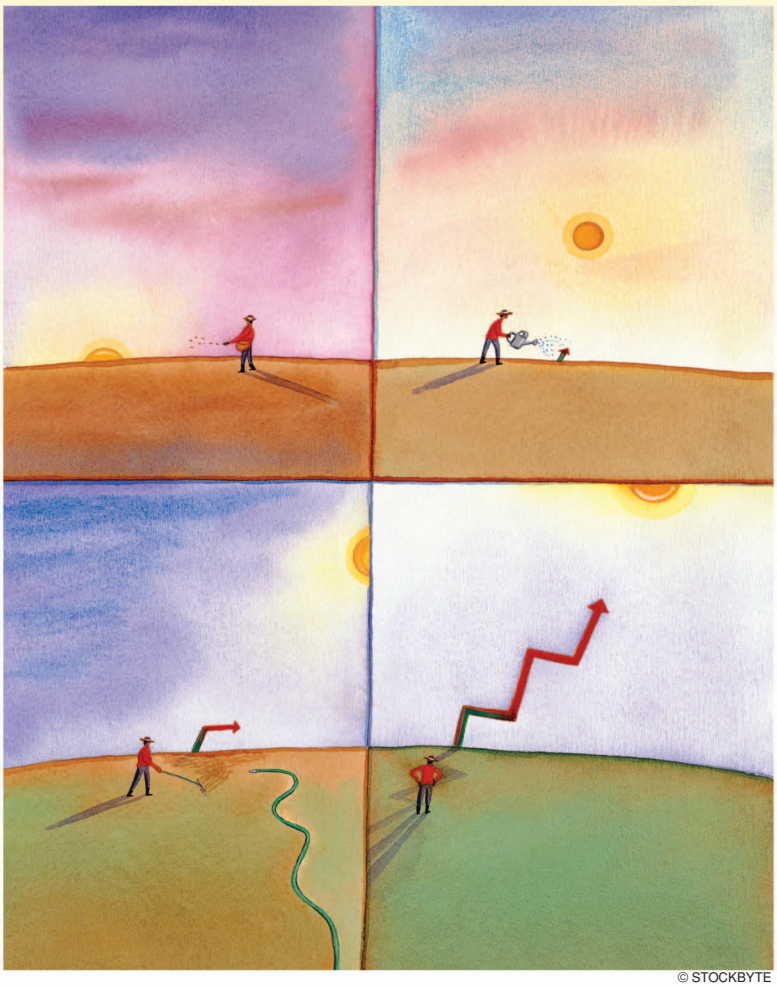
Dimitris G. Tzikas, Aristidis C. Likas, and Nikolaos P. Galatsanos

The Variational Approximation for Bayesian Inference

Life after the EM algorithm

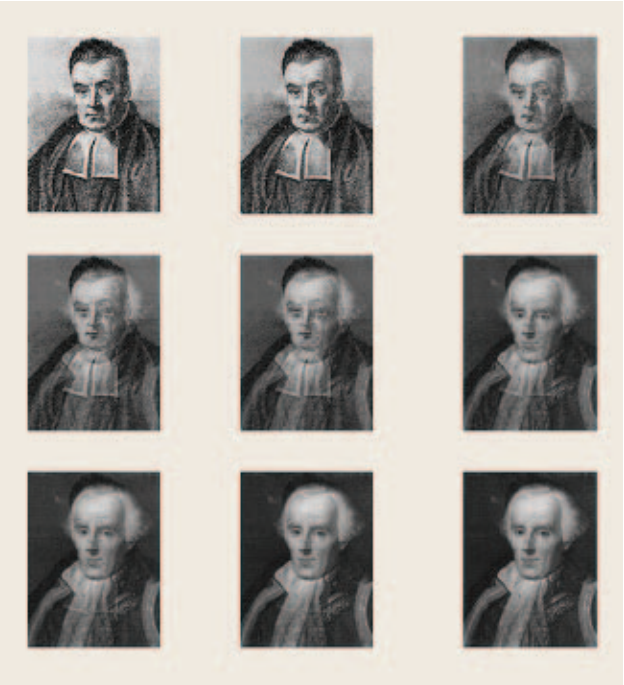
Thomas Bayes (1701–1761), shown in the upper left corner of Figure 1, first discovered Bayes' theorem in a paper that was published in 1764 three years after his death, as the name suggests. However, Bayes, in his theorem, used uniform priors [1]. Pierre-Simon Laplace (1749–1827), shown in the lower right corner of Figure 1, apparently unaware of Bayes' work, discovered the same theorem in more general form in a memoir he wrote at the age of 25 and showed its wide applicability [2]. Regarding these issues S.M. Stiegler writes:

> The influence of this memoir was immense. It was from here that "Bayesian" ideas first spread through the mathematical world, as Bayes's own article was ignored until 1780 and played no important role in scientific debate until the 20th century. It was also this article of Laplace's that introduced the mathematical techniques for the asymptotic analysis of posterior distributions that are still employed today. And it was here that the earliest example of optimum estimation can be found, the derivation and characterization of an estimator that minimized a particular measure of posterior expected loss. After more than two centuries, we mathematicians, statisticians cannot only recognize our roots in this masterpiece of our science, we can still learn from it. [3]



© STOCKBYTE

Digital Object Identifier 10.1109/MSP.2008.929620



[FIG1] Thomas Bayes (upper left) and Pierre-Simon Laplace (lower right) discovered similar theorems in mathematics in the 1700s, spreading new techniques throughout the mathematic world that are still used more than two centuries later.

Maximum likelihood (ML) estimation is one of the most popular methodologies used in modern statistical signal processing. The expectation maximization (EM) algorithm is an iterative algorithm for ML estimation that has a number of advantages and has become a standard methodology for solving statistical signal processing problems. However, the EM algorithm has certain requirements that seriously limit its applicability to complex problems. Recently, a new methodology termed "variational Bayesian inference" has emerged, which relaxes some of the limiting requirements of the EM algorithm and is gaining rapidly popularity. Furthermore, one can show that the EM algorithm can be viewed as a special case of this methodology. In this article, we first present a tutorial introduction of Bayesian variational inference aimed at the signal processing community. We use linear regression and Gaussian mixture modeling as examples to demonstrate the additional capabilities that Bayesian variational inference offers as compared to the EM algorithm.

INTRODUCTION

The ML methodology is one of the basic staples of modern statistical signal processing. The EM algorithm is an iterative algorithm that offers a number of advantages for obtaining ML estimates. Since its formal introduction in 1977 by Dempster et al. [4], the EM algorithm has become a standard methodology for ML estimation. In the IEEE community, the EM is steadily gaining popularity and is being used in an increasing number of applications. The first publications in IEEE journals making reference to the EM algorithm appeared in 1988 and dealt with the problem of tomographic reconstruction of photon limited images [5], [6]. Since then, the EM algorithm has become a popular tool for statistical signal processing used in a wide range of applications, such as recovery and segmentation of images and video, image modelling, carrier frequency synchronization, and channel estimation in communications and speech recognition.

The concept behind the EM algorithm is very intuitive and natural. EM-like algorithms existed in the statistical literature even before [4], however such algorithms were actually EM algorithms in special contexts. The first known such reference dates back to 1886, when Newcomb considers the estimation of the parameters of a mixture of two univariate normals [7]. However, it was in [4] where such ideas were synthesized and the general formulation of the EM algorithm was established. A good survey on the history of the EM algorithm before [4] can be found in [8].

The present article is not a tutorial on the EM algorithm. Such a tutorial appeared in 1996 in *IEEE Signal Processing Magazine* [9]. The present article is aimed at presenting an emerging new methodology for statistical inference that ameliorates certain shortcomings of the EM algorithm. This methodology is termed variational approximation [10] and can be used to solve complex Bayesian models where the EM algorithm cannot be applied. Bayesian inference based on the variational approximation has been used extensively by the machine learning community since the mid-1990s when it was first introduced.

BAYESIAN INFERENCE BASICS

Assume that $\mathbf{x}$ are the observations and $\boldsymbol{\theta}$ the unknown parameters of a model that generated $\mathbf{x}$. In this article, the term estimation will be used strictly to refer to parameters and inference to refer to random variables. The term estimation refers to the calculated approximation of the value of a parameter from incomplete, uncertain and noisy data. In contrast, the term inference will be used to imply Bayesian inference and refers to the process in which prior evidence and observations are used to infer the posterior probability $p(\boldsymbol{\theta}|\mathbf{x})$ of the random variables $\boldsymbol{\theta}$ given the observations $\mathbf{x}$.

One of the most popular approaches for parameter estimation is ML. According to this approach, the ML estimate is obtained as

$$\hat{\boldsymbol{\theta}}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{x};\boldsymbol{\theta}), \tag{1}$$

where $p(\mathbf{x};\boldsymbol{\theta})$ describes the probabilistic relationship between the observations and the parameters based on the assumed model that generated the observations $\mathbf{x}$. At this point, we would like to clarify the difference between the notation $p(\mathbf{x};\boldsymbol{\theta})$ and $p(\mathbf{x}|\boldsymbol{\theta})$. When we write $p(\mathbf{x};\boldsymbol{\theta})$ we imply that $\boldsymbol{\theta}$ are parameters and as a function of $\boldsymbol{\theta}$ is called the likelihood function. In contrast, when we write $p(\mathbf{x};\boldsymbol{\theta})$, we imply that $\boldsymbol{\theta}$ are random variables.

In many cases of interest direct assessment of the likelihood function $p(\mathbf{x};\boldsymbol{\theta})$ is complex and is either difficult or

impossible to compute it directly or optimize it. In such cases the computation of this likelihood is greatly facilitated by the introduction of hidden variables $\mathbf{z}$. These random variables act as links that connect the observations to the unknown parameters via Bayes' law. The choice of hidden variables is problem dependent. However, as their name suggests, these variables are not observed and they provide enough information about the observations so that the conditional probability $p(\mathrm{x}|\mathbf{z})$ is easy to compute. Apart from this role, hidden variables play another role in statistical modeling. They are an important part of the probabilistic mechanism that is assumed to have generated the observations and can be described very succinctly by a graph that is termed "graphical model." More details on graphical models is given in the section "Graphic Models."

Once hidden variables and a prior probability for them $p(\mathbf{z};\boldsymbol{\theta})$ have been introduced, one can obtain the likelihood or the marginal likelihood as it is called at times by integrating out (marginalizing) the hidden variables according to

$$p(\mathrm{x};\boldsymbol{\theta}) = \int p(\mathrm{x},\mathrm{z};\boldsymbol{\theta})\,d\mathrm{z} = \int p(\mathrm{x}|\mathrm{z};\boldsymbol{\theta})p(\mathrm{z};\boldsymbol{\theta})d\mathrm{z}. \quad (2)$$

This seemingly simple integration is the crux of the Bayesian methodology because in this manner we can obtain both the likelihood function, and by using Bayes' theorem, the posterior of the hidden variables according to

$$p(\mathrm{z}|\mathrm{x};\boldsymbol{\theta}) = \frac{p(\mathrm{x}|\mathrm{z};\boldsymbol{\theta})p(\mathrm{z};\boldsymbol{\theta})}{p(\mathrm{x};\boldsymbol{\theta})}. \quad (3)$$

Once the posterior is available, inference as explained above for the hidden variables is also possible. Despite the simplicity of the above formulation, in most cases of interest the integral in (2) is either impossible or very difficult to compute in closed form. Thus, the main effort in Bayesian Inference is concentrated on techniques that allow us to bypass or approximately evaluate this integral.

Such methods can be classified into two broad categories. The first is numerical sampling methods also known as Monte Carlo techniques and the second is deterministic approximations. This article will not address at all Monte Carlo methods. The interested reader for such methods is referred to a number of books and survey articles on this topic, for example [11] and [12]. Furthermore, maximum posteriori (MAP) inference, which is an extension of the ML approach, can be considered as a very crude Bayesian approximation, see "Maximum A Posteriori: Poor Man's Bayesian Inference."

As it will be shown in what follows, the EM algorithm is a Bayesian inference methodology that assumes knowledge of the posterior $p(\mathbf{z}|\mathrm{x};\boldsymbol{\theta})$ and iteratively maximizes the likelihood function without explicitly computing it. A serious shortcoming of this methodology is that in many cases of interest this posterior is not available. However, recent developments in

MAXIMUM A POSTERIORI: POOR MAN'S BAYESIAN INFERENCE

One of the most commonly used methodologies in the statistical signal processing literature is the maximum a posteriori (MAP) method. MAP is often referred to as Bayesian, since the parameter vector $\boldsymbol{\theta}$ is assumed to be a random variable and a prior distribution $p\boldsymbol{\theta}$ is imposed on $\boldsymbol{\theta}$. In this appendix, we would like to illuminate the similarities and differences between MAP estimation and Bayesian inference. For $\mathbf{x}$ the observation and $\boldsymbol{\theta}$ an unknown quantity the MAP estimate is defined as

$$\hat{\boldsymbol{\theta}}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathbf{x}) \quad (A.1)$$

Using Bayes' theorem, the MAP estimate can be obtainedfrom

$$\hat{\boldsymbol{\theta}}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{x}|\boldsymbol{\theta})p\boldsymbol{\theta} \quad (A.2)$$

where $p(\mathbf{x}|\boldsymbol{\theta})$ is the likelihood of the observations. The MAP estimate is easier to obtain from (A.2) than (A.1). The posterior in (A.1) based on Bayes' theorem is given by

$$p(\boldsymbol{\theta}|\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})\,d\boldsymbol{\theta}} \quad (A.3)$$

and requires the computation of the Bayesian integral in the denominator of (A.3) to marginalize $\boldsymbol{\theta}$.

From the above, it is clear that both MAP and Bayesian estimators assume that $\boldsymbol{\theta}$ is a random variable and use Bayes' theorem, however, their similarity stops there. For Bayesian inference, the posterior is used and thus $\boldsymbol{\theta}$ has to be marginalized. In contrast, for MAP the mode of the posterior is used. One can say that Bayesian inference, unlike MAP, averages over all the available information about $\boldsymbol{\theta}$. Thus, it can be stated that MAP is more like "poor man's" Bayesian inference.

The EM can be used to also obtain MAP estimates of $\boldsymbol{\theta}$. Using Bayes' theorem we can write

$$\begin{aligned}\ln p(\boldsymbol{\theta}|\mathbf{x}) &= \ln p(\boldsymbol{\theta}|\mathbf{x}) - \ln p(\mathbf{x}) \\ &= \ln p(\mathbf{x}|\boldsymbol{\theta}) + \ln p(\boldsymbol{\theta}) - \ln p(\mathbf{x}). \quad (A.4)\end{aligned}$$

Using a similar framework as for the ML-EM case in the section "An Alternative View of the EM Algorithm," we can write

$$\begin{aligned}\ln p(\boldsymbol{\theta}|\mathbf{x}) &= F(q,\boldsymbol{\theta}) + KL(q||p) + \ln p(\boldsymbol{\theta}) - \ln p(\mathbf{x}) \\ &\geq F(q,\boldsymbol{\theta}) + \ln p(\boldsymbol{\theta}) - \ln p(\mathbf{x}), \quad (A.5)\end{aligned}$$

where in this context $\ln p(\mathbf{x})$ is a constant. The right-hand side of (A.5) can be maximized in an alternating fashion as in the EM algorithm. Optimization with respect to $q(\mathbf{z})$ gives an identical E-step as for the ML case previously explained. Optimization with respect to $\boldsymbol{\theta}$ gives a different M-step since the objective function now contains also the term $\ln p(\boldsymbol{\theta})$. In general, the M-step for the MAP-EM algorithm is more complex than in its ML counterpart, see for example [30] and [31]. Strictly speaking, in such a model MAP estimation is used only for the $\boldsymbol{\theta}$ random variables, while Bayesian inference is used for hidden variables $\mathbf{z}$.

Bayesian inference allow us to bypass this difficulty by approximating the posterior. They are termed "variational Bayesian" and they will be the focus of this tutorial.

GRAPHICAL MODELS

Graphical models provide a framework for representing dependencies among the random variables of a statistical modelling problem and they constitute a comprehensive and elegant way to graphically represent the interaction among the entities involved in a probabilistic system. A graphical model is a graph whose nodes correspond to the random variables of a problem and the edges represent the dependencies among the variables. A directed edge from a node A to a node B in the graph denotes that the variable B stochastically depends on the value of the variable A. Graphical models can be either directed or undirected. In the second case they are also known as Markov random fields [13], [14], [15]. In the rest of this tutorial, we will focus on directed graphical models also called Bayesian Networks, where all the edges are considered to have a direction from parent to child denoting the conditional dependency among the corresponding random variables. In addition we assume that the directed graph is acyclic (i.e., contains no cycles).

Let $G = (V, E)$ be a directed acyclic graph with V being the set of nodes and E the set of directed edges. Let also x_s denote the random variable associated with node s and $\pi(s)$ the set of parents of node s. Associated with each node s is also a conditional probability density $p(x_s|x_{\pi(s)})$ that defines the distribution of x_s given the values of its parent variables. Therefore, for a graphical model to be completely defined, apart from the graph structure, the conditional probability distribution at each node should also be specified. Once these distributions are known, the joint distribution over the set of all variables can be computed as the product:

$$p(x) = \prod_s p(x_s|x_{\pi(s)}). \tag{4}$$

The above equation constitutes a formal definition of a directed graphical model [13] as a collection of probability distributions that factorize in the way specified in the above equation (which of course depends on the structure of the underlying graph).

In Figure 2 we show an example of a directed graphical model. The random variables depicted at the nodes are a, b, c, and d. Each node represents a conditional probability density that quantifies the dependency of the node from its parents. The densities at the nodes might not be exactly known and can be parameterized by a set of parameters θ_i. Using the chain rule of probability we would write the joint distribution as:

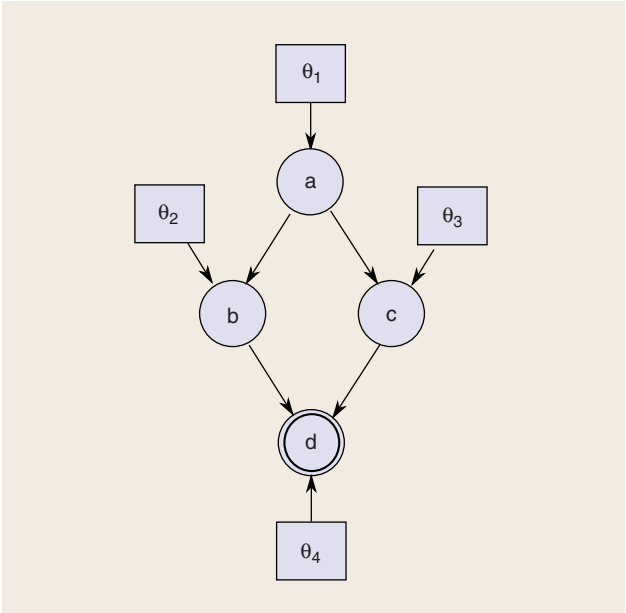
$$p(a, b, c, d; \boldsymbol{\theta}) = p(a; \theta_1)p(b|a; \theta_2)p(c|a, b; \theta_3)p(d|a, b, c; \theta_4). \tag{5}$$

However, we can simplify this expression by taking into account the independencies that the graph structure implies. In general, in a graphical model each node is independent of its ancestors given its parents. This means that node c does not depend on node a given node b, and node d does not depend on a given nodes b and c. Thus, from (4) we can write:

$$p(a, b, c, d; \boldsymbol{\theta}) = p(a; \boldsymbol{\theta}_1)p(b\,|a; \boldsymbol{\theta}_2)p(c|a; \boldsymbol{\theta}_3)p(d\,|b, c; \boldsymbol{\theta}_4). \tag{6}$$

Another useful characterization arising in graphical modeling is that in the presence of some observations, usually called dataset, the random variables can be distinguished as observed (or visible) for which there exist observations and hidden for which direct observations are not available. A useful consideration is to assume that the observed data are produced through a generation mechanism, described by the graphical model structure, which involves the hidden variables as intermediate sampling and computational steps. It must also be noted that a graphical model can be either parametric or nonparametric. If the model is parametric, the parameters appear in the conditional probability distributions at some of the graph nodes, i.e., these distributions are parameterized probabilistic models.

Once a graphical model is completely determined (i.e., all parameters have been specified), then several inference problems could be defined such as computing the marginal distribution of a subset of random variables, computing the conditional distribution of a subset of variables given the values of the rest variables and computing the maximum point in some of the previous densities. In the case where the graphical model is parametric, then we have the problem of



[FIG2] Example of directed graphical model. Nodes denoted with circles correspond to random variables, while nodes denoted with squares correspond to parameters of the model. Doubly circled nodes represent observed random variables, while single circled nodes represent hidden random variables.

learning appropriate values of the parameters given some dataset with observations. Usually, in the process of parameter learning, several inference steps are involved.

AN ALTERNATIVE VIEW OF THE EM ALGORITHM

In this article, we will follow the exposition of the EM in [16] and [13]. It is straightforward to show that the log-likelihood can be written as

$$\ln\, p(\mathbf{x};\boldsymbol{\theta}) = F(q,\boldsymbol{\theta}) + KL(q\,\|p) \tag{7}$$

with

$$F(q,\boldsymbol{\theta}) = \int q(\mathbf{z}) \ln\left(\frac{p(\mathbf{x},\mathbf{z};\boldsymbol{\theta})}{q(\mathbf{z})}\right) d\mathbf{z} \tag{8}$$

and

$$\mathrm{KL}(q\,\|p) = -\int q(\mathbf{z}) \ln\left(\frac{p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta})}{q(\mathbf{z})}\right) d\mathbf{z} \tag{9}$$

where $q(\mathbf{z})$ is any probability density function. $\mathrm{KL}(q\,\|p)$ is the Kullback-Leibler divergence between $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta})$ and $q(\mathbf{z})$, and since $\mathrm{KL}(q\,\|p) \geq 0$, it holds that $\ln p(\mathbf{x};\boldsymbol{\theta}) \geq F(q,\boldsymbol{\theta})$. In other words, $F(q,\boldsymbol{\theta})$ is a lower bound of the log-likelihood. Equality holds only when $\mathrm{KL}(q\,\|p) = 0$, which implies $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}) = q(\mathbf{z})$. The EM algorithm and some recent advances in deterministic approximations for Bayesian inference can be viewed in the light of the decomposition in (7) as the maximization of the lower bound $F(q,\boldsymbol{\theta})$ with respect to the density q and the parameters $\boldsymbol{\theta}$.

In particular, the EM is a two step iterative algorithm that maximizes the lower bound $F(q,\boldsymbol{\theta})$ and hence the log-likelihood. Assume that the current value of the parameters is $\boldsymbol{\theta}^{\mathrm{OLD}}$. In the E-step the lower bound $F(q,\boldsymbol{\theta}^{\mathrm{OLD}})$ is maximized with respect to $q(\mathbf{z})$. It is easy to see that this happens when $KL(q\,\|p) = 0$, in other words, when $q(\mathbf{z}) = p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}})$. In this case the lower bound is equal to the log-likelihood. In the subsequent M-step, $q(\mathbf{z})$ is held fixed and the lower bound $F(q,\boldsymbol{\theta})$ is maximized with respect to $\boldsymbol{\theta}$ to give some new value $\boldsymbol{\theta}^{\mathrm{NEW}}$. This will cause the lower bound to increase and as a result, the corresponding log-likelihood will also increase. Because $q(\mathbf{z})$ was determined using $\boldsymbol{\theta}^{\mathrm{OLD}}$ and is held fixed in the M-step, it will not be equal to the new posterior $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{NEW}})$ and hence the KL distance will not be zero. Thus, the increase in the log-likelihood is greater than the increase in the lower bound. If we substitute $q(\mathbf{z}) = p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}})$ into the lower bound and expand (8) we get

$$\begin{aligned} F(q,\boldsymbol{\theta}) &= \int p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}}) \ln\, p(\mathbf{x},\mathbf{z};\boldsymbol{\theta})\, d\mathbf{z} \\ &\quad - \int p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}}) \ln\, p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}})\, d\mathbf{z} \\ &= Q(\boldsymbol{\theta},\boldsymbol{\theta}^{\mathrm{OLD}}) + \text{constant} \end{aligned} \tag{10}$$

where the constant is simply the entropy of $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}})$ which does not depend on $\boldsymbol{\theta}$. The function

$$\begin{aligned} Q(\boldsymbol{\theta},\boldsymbol{\theta}^{\mathrm{OLD}}) &= \int p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}}) \ln\, p(\mathbf{x},\mathbf{z};\boldsymbol{\theta})\, d\mathbf{z} \\ &= \langle \ln p(\mathbf{x},\mathbf{z};\boldsymbol{\theta})\rangle_{p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}})} \end{aligned} \tag{11}$$

is the expectation of the log-likelihood of the complete data (observations + hidden variables) which is maximized in the M-step. The usual way of presenting the EM algorithm in the signal processing literature has been via use of the $Q(\boldsymbol{\theta},\boldsymbol{\theta}^{\mathrm{OLD}})$ function directly, see for example [9] and [17].

In summary, the EM algorithm is an iterative algorithm involving the following two steps:

$$\text{E-step : Compute} \quad p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}}) \tag{12}$$

$$\text{M-step : Evaluate} \quad \boldsymbol{\theta}^{\mathrm{NEW}} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta},\boldsymbol{\theta}^{\mathrm{OLD}}). \tag{13}$$

Furthermore, we would like to point out that the EM algorithm requires that $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta})$ is explicitly known, or at least we should be able to compute the conditional expectation of its sufficient statistics $\langle \ln p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta})\rangle_{p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}^{\mathrm{OLD}})}$, see (11). In other words, we have to know the conditional pdf of the hidden variables given the observations in order to use the EM algorithm. While $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta})$ is in general much easier to infer than $p(\mathbf{x};\boldsymbol{\theta})$, in many interesting problems this is not possible and thus the EM algorithm is not applicable.

THE VARIATIONAL EM FRAMEWORK

One can bypass the requirement of exactly knowing $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta})$ by assuming an appropriate $q(\mathbf{z})$ in the decomposition of (7). In the E-step $q(\mathbf{z})$ is found such that it maximizes $F(q,\boldsymbol{\theta})$ keeping $\boldsymbol{\theta}$ fixed. To perform this maximization, a particular form of $q(\mathbf{z})$ must be assumed. In certain cases it is possible to assume knowledge of the form of $q(\mathbf{z};\boldsymbol{\omega})$, where $\boldsymbol{\omega}$ is a set of parameters. Thus, the lower bound $F(\boldsymbol{\omega},\boldsymbol{\theta})$ becomes a function of these parameters and is maximized with respect to $\boldsymbol{\omega}$ in the E-step and with respect to $\boldsymbol{\theta}$ in the M-step, see for example [13].

However, in its general form the lower bound $F(q,\boldsymbol{\theta})$ is a functional in terms of q, in other words, a mapping that takes as input a function $q(\mathbf{z})$, and returns as output the value of the functional. This leads naturally to the concept of the functional derivative, which in analogy to the function derivative, gives the functional changes for infinitesimal changes to the input function. This area of mathematics is called calculus of variations [18] and has been applied to many areas of mathematics, physical sciences and engineering, for example fluid mechanics, heat transfer, and control theory.

Although there are no approximations in the variational theory, variational methods can be used to find approximate solutions in Bayesian inference problems. This is done by assuming that the functions over which optimization is performed have specific forms. For example, we can assume only

quadratic functions or functions that are linear combinations of fixed basis functions. For Bayesian inference, a particular form that has been used with great success is the factorized one, see [19] and [20]. The idea for this factorized approximation stems from theoretical physics where it is called mean field theory [21].

According to this approximation, the hidden variables $\mathbf{z}$ are assumed to be partitioned into M partitions z_i with $i = 1, \ldots, M$. Also it is assumed that $q(\mathbf{z})$ factorizes with respect to these partitions as

$$q(\mathbf{z}) = \prod_{i=1}^{M} q_i(z_i). \tag{14}$$

Thus, we wish to find the $q(\mathbf{z})$ of the form of (14) that maximizes the lower bound $F(q, \boldsymbol{\theta})$. Using (14) and denoting for simplicity $q_j(z_j) = q_j$ we have

$$\begin{aligned} F(q, \boldsymbol{\theta}) &= \int \prod_i q_i \left[\ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta} - \sum_i \ln q_i \right] d\mathbf{z} \\ &= \int \prod_i q_i \ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta} \prod_i dz_i \\ &\quad - \sum_i \int \prod_j q_j \ln q_i \, dz_i \\ &= \int q_j \left[\ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta} \prod_{i \neq j} (q_i \, dz_i) \right] dz_j \\ &\quad - \int q_j \ln q_j \, dz_j - \sum_{i \neq j} \int q_i \ln q_i \, dz_i \\ &= \int q_j \ln \tilde{p}(\mathbf{x}, z_j; \boldsymbol{\theta} \, dz_i - \int q_j \ln q_j \, dz_j \\ &\quad - \sum_{i \neq j} \int q_i \ln q_i \, dz_i \\ &= -\mathrm{KL}(q_j \| \tilde{p}) - \sum_{i \neq j} \int q_i \ln q_i \, dz \end{aligned} \tag{15}$$

where

$$\ln \tilde{p}(\mathbf{x}, z_j; \boldsymbol{\theta}) = \langle \ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta}) \rangle_{i \neq j} = \int \ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta}) \prod_{i \neq j} (q_i \, dz_i).$$

Clearly the bound in (15) is maximized when the Kullback-Leibler distance becomes zero, which is the case for $q_j(z_j) = \tilde{p}(\mathbf{x}, z_j; \boldsymbol{\theta})$, in other words the expression for the optimal distribution $q_j^*(z_j)$ is

$$\ln q_j^*(z_j) = \langle \ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta}) \rangle_{i \neq j} + \text{const.} \tag{16}$$

The additive constant in (16) can be obtained through normalization, thus we have

$$q_j^*(z_j) = \frac{\exp\left(\langle \ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta}) \rangle_{i \neq j} \right)}{\int \exp\left(\langle \ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta}) \rangle_{i \neq j} \right) dz_j}. \tag{17}$$

The above equations for $j = 1, \ldots, M$ are a set of consistency conditions for the maximum of the lower bound subject to the factorization of (14). They do not provide an explicit solution since they depend on the other factors $q_i(z_i)$ for $i \neq j$. Therefore, a consistent solution is found by cycling through these factors and replacing each in turn with the revised estimate.

In summary, the variational EM algorithm is given by the following two steps:

Variational E-Step: Evaluate $q^{\mathrm{NEW}}(\mathbf{z})$ to maximize $F(q, \boldsymbol{\theta}^{\mathrm{OLD}})$ solving the system of (17).

Variational M-Step: Find $\boldsymbol{\theta}^{\mathrm{NEW}} = \arg\max_{\boldsymbol{\theta}} F(q^{\mathrm{NEW}}, \boldsymbol{\theta})$.

At this point it is worth noting that in certain cases a Bayesian model can contain only hidden variables and no parameters. In such cases the variational EM algorithm has only an E-step in which $q(\mathbf{z})$ is obtained using (17). This function $q(\mathbf{z})$ constitutes an approximation to $p(\mathbf{z}|\mathbf{x})$ that can be used for inference of the hidden variables.

LINEAR REGRESSION

In this section, we will use the linear regression problem as an example to demonstrate the Bayesian inference methods of the previous sections. Linear regression was selected because it is simple and constitues an excellent introductory example. Furthermore, it occurs in many signal processing applications ranging from deconvolution, channel estimation, speech recognition, frequency estimation, time series prediction, and system identification.

For this problem, we consider an unknown signal $y(\mathbf{x}) \in \Re$, $\mathbf{x} \in \Omega \subseteq \Re^N$ and want to predict its value $t_* = y(\mathbf{x}_*)$ at an arbitrary location $\mathbf{x}_* \in \Omega$, using a vector $\mathbf{t} = (t_1, \ldots, t_N)^T$ of N noisy observations $t_n = y(\mathbf{x}_n) + \varepsilon_n$, at locations $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^T$, $\mathbf{x}_n \in \Omega$, $n = 1, \ldots, N$. The additive noise ε_n is commonly assumed to be independent, zero-mean, Gaussian distributed:

$$p(\boldsymbol{\varepsilon}) = N(\boldsymbol{\varepsilon}|0, \beta^{-1}\mathbf{I}), \tag{18}$$

where β is the inverse variance and $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_N)^T$.

The signal $\mathbf{y}$ is commonly modeled as the linear combination of M basis functions $\phi_m(\mathbf{x})$:

$$y(\mathbf{x}) = \sum_{m=1}^{M} w_m \phi_m(\mathbf{x}), \tag{19}$$

where $\mathbf{w} = (w_1, \ldots, w_M)^T$ are the weights of the linear combination. Defining the design matrix $\boldsymbol{\Phi} = (\phi_1, \ldots, \phi_M)$, with $\phi_m = (\phi_m(\mathbf{x}_1), \ldots, \phi_m(\mathbf{x}_N))^T$, the observations t are modeled as

$$\mathbf{t} = \boldsymbol{\Phi}\mathbf{w} + \boldsymbol{\varepsilon} \tag{20}$$

and the likelihood is

$$p(\mathrm{t};\mathrm{w},\beta) = N(\mathrm{t}|\mathbf{\Phi}\mathrm{w},\beta^{-1}\mathrm{I}). \tag{21}$$

In what follows we will apply the theory from earlier sections to the linear regression problem and demonstrate three methodologies to compute the unknown weights w of this linear model. First, we apply typical ML estimation of the weights which are assumed to be parameters. As it will be demonstrated, since the number of parameters is the same as the number of our observations, the ML estimates are very sensitive to the model noise and over fit the observations. Subsequently, to ameliorate this problem a prior is imposed on the weights which are assumed to be random variables. First, a simple Bayesian model is used which is based on a stationary Gaussian prior for the weights. For this model, Bayesian inference is performed using the EM algorithm and the resulting solution is robust to noise. Nevertheless, this Bayesian model is very simplistic and does not have the ability to capture the local signal properties. For this purpose it is possible to introduce a more sophisticated spatially varying hierarchical model which is based on a nonstationary Gaussian prior for the weights and a hyperprior. This model is too complex to solve using the EM algorithm. For this purpose, the variational Bayesian methodology described in the section "Variational EM Framework" is used to infer values for the unknowns of this model. Finally, the three methods are used to obtain estimates of a simple artificial signal, in order to demonstare that the added complexity in the Bayesian model improves the solution quality. In Figure 3(a), (b), and (c) we show the graphical models for the three approaches to Linear Regression.

The simplest estimate of the weights w of the linear model is obtained by maximizing the likelihood of the model. This ML estimate assumes the weights w to be parameters, as shown in the graphical model of Figure 3(a). The ML estimate is obtained by maximizing the likelihood function

$$p(\mathrm{t};\mathrm{w},\beta) = (2\pi)^{-\frac{N}{2}}\beta^{\frac{N}{2}}\exp\left(-\frac{\beta}{2}\|\mathrm{t}-\mathbf{\Phi}\mathrm{w}\|^2\right).$$

This is equivalent to minimizing $E_{LS}(\mathrm{w}) = \|\mathrm{t}-\mathbf{\Phi}\mathrm{w}\|^2$. Thus, in this case the ML is equivalent with the least squares (LS) estimate

$$\mathrm{w}_{LS} = \arg\max_{w} p(\mathrm{t};\mathrm{w},\beta) = \arg\min_{w} E_{LS}(\mathrm{w}) = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathrm{t}. \tag{22}$$

In many situations (and depending on the basis functions that are used), the matrix $\mathbf{\Phi}^T\mathbf{\Phi}$ may be ill-conditioned and difficult to invert. This means that if noise $\boldsymbol{\varepsilon}$ is included in the signal observations, it will heavily affect the estimation w_{LS} of the weights. Thus, when using ML linear regression (MLLR), the basis functions should be carefully chosen to ensure that matrix $\mathbf{\Phi}^T\mathbf{\Phi}$ can be inverted. This is generally achieved by using a sparse model with few basis functions, which also has the advantage that only few parameters have to be estimated.

EM-BASED BAYESIAN LINEAR REGRESSION

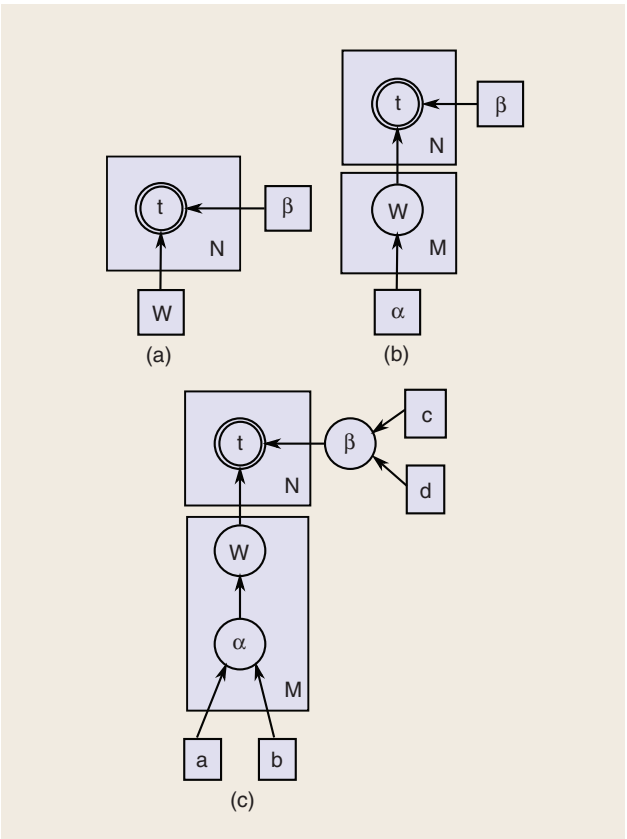
A Bayesian treatment of the linear model begins by assigning a prior distribution to the weights of the model. This introduces bias in the estimation but also greatly reduces its variance, which is a major problem of the ML estimate. Here, we consider the common choice of independent, zero-mean, Gaussian prior distribution for the weights of the linear model:

$$p(\mathrm{w};\alpha) = \prod_{m=1}^{M} N(w_m|0,\alpha^{-1}). \tag{23}$$

This is a stationary prior distribution, meaning that the distribution of all the weights is identical. The graphical model for this problem is shown in Figure 3(b). Notice that here the weights w are hidden random variables and the model parameters are the parameter α of the prior for w and the inverse variance β of the additive noise.

Bayesian inference proceeds by computing the posterior distribution of the hidden variables:

$$p(\mathrm{w}|\mathrm{t};\alpha,\beta) = \frac{p(\mathrm{t}|\mathrm{w};\beta)p(\mathrm{w};\alpha)}{p(\mathrm{t};\alpha,\beta)}. \tag{24}$$



[FIG3] Graphical models for linear regression solved using (a) direct ML estimation (model without prior), (b) EM (model with stationary prior), and (c) variational EM (model with hierarchical prior).

Notice that the marginal likelihood $p(\mathrm{t};\alpha,\beta)$ that appears on the denominator can be computed analytically:

$$p(\mathrm{t};\alpha,\beta)=\int p(\mathrm{t}|\mathrm{w};\beta)p(\mathrm{w};\alpha)\,d\mathrm{w}=N(\mathrm{t}|0,\beta^{-1}\mathbf{I}+\alpha^{-1}\mathbf{\Phi}\mathbf{\Phi}^T). \tag{25}$$

Then, the posterior of the hidden variables is

$$p(\mathrm{w}|\mathrm{t};\alpha,\beta)=N(\mathrm{w}|\boldsymbol{\mu},\mathbf{\Sigma}), \tag{26}$$

with

$$\boldsymbol{\mu}=\beta\mathbf{\Sigma}\mathbf{\Phi}^T\mathrm{t}, \tag{27}$$

$$\mathbf{\Sigma}=(\beta\mathbf{\Phi}^T\mathbf{\Phi}+\alpha\mathbf{I})^{-1}. \tag{28}$$

The parameters of the model can be estimated by maximizing the logarithm of the marginal likelihood $p(\mathrm{t};\alpha,\beta)$:

$$(\alpha_{\mathrm{ML}},\beta_{\mathrm{ML}})=\arg\min_{\alpha,\beta}\Big\{\log|\beta^{-1}\mathbf{I}+\alpha^{-1}\mathbf{\Phi}\mathbf{\Phi}^T|+\mathrm{t}^T(\beta^{-1}\mathbf{I}+\alpha^{-1}\mathbf{\Phi}\mathbf{\Phi}^T)^{-1}\mathrm{t}\Big\}. \tag{29}$$

Direct optimization of (29) presents several computational difficulties, since its derivatives with respect to the parameters (α,β) are difficult to compute. Furthermore, the problem requires a constrained optimization algorithm since the estimates of (α,β) have to be positive since they represent inverse variances. Instead, the EM algorithm described earlier, provides an efficient framework to simultaneously obtain estimates for (α,β) and infer values for w. Notice, that although the EM algorithm does not involve computations with the marginal likelihood (25), the algorithm converges to a local maximum of it. After initializing the parameters to some values $(\alpha^{(0)},\beta^{(0)})$, the algorithm proceeds by iteratively performing the following steps:

■ **E- step**

Compute the expected value of the logarithm of the complete likelihood :

$$\begin{aligned}Q^{(t)}(\mathrm{t},\mathrm{w};\alpha,\beta)&=\langle\ln\,p(\mathrm{t},\mathrm{w};\alpha,\beta)\rangle_{p(\mathrm{w}|\mathrm{t};\alpha^{(t)},\beta^{(t)})}\\&=\langle\ln p(\mathrm{t}|\mathrm{w};\alpha,\beta)p(\mathrm{w};\alpha,\beta)\rangle_{p(\mathrm{w}|\mathrm{t};\alpha^{(t)},\beta^{(t)})}.\end{aligned} \tag{30}$$

This is computed using (21) and (23) as

$$\begin{aligned}Q^{(t)}(\mathrm{t},\mathrm{w};\alpha,\beta)&=\left\langle\frac{N}{2}\ln\beta-\frac{\beta}{2}(\|\mathrm{t}-\mathbf{\Phi}\mathrm{w}\|^2)+\frac{M}{2}\ln\alpha-\frac{\alpha}{2}(\|\mathrm{w}\|^2)\right\rangle+\text{const}\\&=\frac{N}{2}\ln\beta-\frac{\beta}{2}\langle\|\mathrm{t}-\mathbf{\Phi}\mathrm{w}\|^2\rangle+\frac{M}{2}\ln\alpha-\frac{\alpha}{2}(\langle\|\mathrm{w}\|^2\rangle)+\text{const}.\end{aligned} \tag{31}$$

These expected values are with respect to $p(\mathrm{w}|\mathrm{t};\alpha^{(t)},\beta^{(t)})$ and can be computed from (26), giving

$$Q^{(t)}(\mathrm{t},\mathrm{w};\alpha,\beta)=\frac{N}{2}\ln\beta-\frac{\beta}{2}\left(\|\mathrm{t}-\mathbf{\Phi}\boldsymbol{\mu}^{(t)}\|^2+\mathrm{tr}[\mathbf{\Phi}^T\mathbf{\Sigma}^{(t)}\mathbf{\Phi}]\right)+\frac{M}{2}\ln\alpha-\frac{\alpha}{2}\left(\|\boldsymbol{\mu}^{(t)}\|^2+\mathrm{tr}[\mathbf{\Sigma}^{(t)}]\right)+\text{const} \tag{32}$$

where $\boldsymbol{\mu}^{(t)}$ and $\mathbf{\Sigma}^{(t)}$ are computed using the current estimates of the parameters $\alpha^{(t)}$ and $\beta^{(t)}$:

$$\boldsymbol{\mu}^{(t)}=\beta^{(t)}\mathbf{\Sigma}^{(t)}\mathbf{\Phi}^T\mathrm{t}, \tag{32}$$

$$\mathbf{\Sigma}^{(t)}=(\beta^{(t)}\mathbf{\Phi}^T\mathbf{\Phi}+\alpha^{(t)}\mathbf{I})^{-1}. \tag{34}$$

■ **M- step**

Maximize $Q^{(t)}(\mathrm{t},\mathrm{w};\alpha,\beta)$ with respect to the parameters α and β:

$$(\alpha^{(t+1)},\beta^{(t+1)})=\arg\max_{(\alpha,\beta)}Q^{(t)}(\mathrm{t},\mathrm{w};\alpha,\beta). \tag{35}$$

The derivatives of $Q^{(t)}(\mathrm{t},\mathrm{w};\alpha,\beta)$ with respect to the parameters are:

$$\frac{\partial Q^{(t)}(\mathrm{t},\mathrm{w};\alpha,\beta)}{\partial\alpha}=\frac{M}{2\alpha}-\frac{1}{2}\left(\|\boldsymbol{\mu}^{(t)}\|^2+\mathrm{tr}[\mathbf{\Sigma}^{(t)}]\right), \tag{36}$$

$$\frac{\partial Q^{(t)}(\mathrm{t},\mathrm{w};\alpha,\beta)}{\partial\beta}=\frac{N}{2\beta}-\frac{1}{2}\left(\|\mathrm{t}-\mathbf{\Phi}\boldsymbol{\mu}^{(t)}\|^2+\mathrm{tr}[\mathbf{\Phi}^T\mathbf{\Sigma}^{(t)}\mathbf{\Phi}]\right). \tag{37}$$

Setting these to zero, we obtain the following formulas to update the parameters α and β:

$$\alpha^{(t+1)}=\frac{M}{\|\boldsymbol{\mu}^{(t)}\|^2+\mathrm{tr}[\mathbf{\Sigma}^{(t)}]}, \tag{38}$$

$$\beta^{(t+1)}=\frac{N}{\|\mathrm{t}-\mathbf{\Phi}\boldsymbol{\mu}^{(t)}\|^2+\mathrm{tr}[\mathbf{\Phi}^T\mathbf{\Sigma}^{(t)}\mathbf{\Phi}]}. \tag{39}$$

Notice that the maximization step can be analytically performed in contrast to direct maximization of the marginal likelihood in (25), which would require numerical optimization. Furthermore, (38) and (39) guarantee that positive estimations for the parameters α and β are produced, which is a requirement since these represent inverse variance parameters. However, the parameters should be initialized with care, since depending on the initialization a different local maximum may be attained. Inference for w is obtained directly since the sufficient statistics of the posterior $p(\mathrm{w}|\mathrm{t};\alpha,\beta)$ are computed in the E-step. The mean of this posterior (33) can be used as Bayesian linear minimum mean square error (LMMSE) inference for w.

VARIATIONAL EM-BASED BAYESIAN LINEAR REGRESSION

In the Bayesian approach described in the previous section, due to the use of a stationary Gaussian prior distribution for the weights of the linear model, exact computation of the marginal likelihood is possible and Bayesian inference is performed analytically. However, in many situations, it is important to allow the flexibility to model local characteristics of the signal, which the simple stationary Gaussian prior distribution is unable to do. For this reason, a nonstationary Gaussian prior distribution with a distinct inverse variance α_m for each weight is considered:

$$p(\mathrm{w}|\boldsymbol{\alpha}) = \prod_{m=1}^{M} N\left(w_m\,|0, \alpha_m^{-1}\right). \tag{40}$$

However, such a model is over-parameterized since there are almost as many observations as parameters to be estimated. For this purpose, the precision parameters $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_M)^T$ are constrained by treating them as random variables and imposing a Gamma prior distribution to them according to

$$p(\boldsymbol{\alpha};\, a, b) = \prod_{m=1}^{M} \mathrm{Gamma}(\alpha_m\,|a, b). \tag{41}$$

This prior is selected because it is conjugate to the Gaussian [13]. Furthermore, we assume a Gamma distribution as prior for the noise inverse variance β

$$p(\beta;\, c, d) = \mathrm{Gamma}(\beta|c, d\,). \tag{42}$$

The graphical model for this Bayesian approach is shown in Figure 3(c) where the dependence of the hidden variables w on the hidden variables $\boldsymbol{\alpha}$ is apparent. Also, the parameters $a, b, c,$ and d of this model and the hidden variables that depend on them are also apparent.

Bayesian inference requires the computation of the posterior distribution

$$p(\mathrm{w}, \boldsymbol{\alpha}, \beta|\mathrm{t}) = \frac{p(\mathrm{t}|\mathrm{w}, \beta)p(\mathrm{w}|\boldsymbol{\alpha})p(\boldsymbol{\alpha})p(\beta)}{p(\mathrm{t})}. \tag{43}$$

However, the marginal likelihood $p(\mathrm{t}) = \int p(\mathrm{t}|\mathrm{w}, \beta)p(\mathrm{w}|\boldsymbol{\alpha})p(\boldsymbol{\alpha})$ $p(\beta)d\mathrm{w}d\boldsymbol{\alpha}d\beta$ cannot be computed analytically, and thus the normalization constant in (43) cannot be computed. Thus, we resort to approximate Bayesian inference methods and specifically to the variational inference methodology. Assuming posterior independence between the weights w and the variance parameters $\boldsymbol{\alpha}$ and β,

$$p(\mathrm{w}, \boldsymbol{\alpha}, \beta|\mathrm{t};\, a, b, c, d) \approx q(\mathrm{w}, \boldsymbol{\alpha}, \beta) = q(\mathrm{w})q(\boldsymbol{\alpha})q(\beta), \tag{44}$$

the approximate posterior distributions q can be computed from (16) as follows. Keeping only the terms of $\ln q(\mathrm{w})$ that depend on w, we have

$$\begin{aligned}
\ln q(\mathrm{w}) &= \langle \ln p(\mathrm{t}, \mathrm{w}, \boldsymbol{\alpha}, \beta)_{q(\boldsymbol{\alpha})q(\beta)} \rangle + \mathrm{const} \\
&= \langle \ln p(t\,|\mathrm{w}, \beta)p(\mathrm{w}|\boldsymbol{\alpha})_{q(\boldsymbol{\alpha})q(\beta)} \rangle + \mathrm{const} \\
&= \langle \ln p(t\,|\mathrm{w}, \beta) + \ln p(\mathrm{w}|\boldsymbol{\alpha})_{q(\boldsymbol{\alpha})q(\beta)} \rangle + \mathrm{const} \\
&= \left\langle -\frac{\beta}{2}(\mathrm{t} - \boldsymbol{\Phi}\mathrm{w})^T(\mathrm{t} - \boldsymbol{\Phi}\mathrm{w}) - \frac{1}{2}\sum_{m=1}^{M} \alpha_m w_m^2 \right\rangle + \mathrm{const} \\
&= -\frac{\langle\beta\rangle}{2}[\mathrm{t}^T\mathrm{t} - 2\mathrm{t}^T\boldsymbol{\Phi}\mathrm{w} + \mathrm{w}^T\boldsymbol{\Phi}^T\boldsymbol{\Phi}\mathrm{w}] \\
&\quad - \frac{1}{2}\sum_{m=1}^{M} \langle\alpha_m\rangle w_m^2 + \ \mathrm{const} \\
&= -\frac{1}{2}\mathrm{w}^T(\langle\beta\rangle\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \langle\mathrm{A}\rangle)\mathrm{w} - \langle\beta\rangle\mathrm{w}^T\boldsymbol{\Phi}^T\mathrm{t} + \ \mathrm{const} \\
&= -\frac{1}{2}\mathrm{w}^T\boldsymbol{\Sigma}^{-1}\mathrm{w} - \mathrm{w}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \ \mathrm{const}
\end{aligned} \tag{45}$$

where $\mathrm{A} = \mathrm{diag}(\alpha_1, \ldots, \alpha_\mathrm{M})$.

Notice that this is the exponent of a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ given by

$$\boldsymbol{\mu} = \langle\beta\rangle\boldsymbol{\Sigma}\boldsymbol{\Phi}^T\mathrm{t}, \tag{46}$$

$$\boldsymbol{\Sigma} = (\langle\beta\rangle\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \langle\mathrm{A}\rangle)^{-1}. \tag{47}$$

Therefore, $q(\mathrm{w})$ is given by

$$q(\mathrm{w}) = N(\mathrm{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}). \tag{48}$$

The posterior $q(\boldsymbol{\alpha})$ is similarly obtained by computing the terms of $\ln q(\boldsymbol{\alpha})$ that depend on $\boldsymbol{\alpha}$

$$\begin{aligned}
\ln q(\boldsymbol{\alpha}) &= \langle \ln p(\mathrm{t}, \mathrm{w}, \boldsymbol{\alpha}, \beta)\rangle_{q(\mathrm{w})q(\beta)} \\
&= \langle \ln p(\mathrm{w}|\boldsymbol{\alpha})p(\boldsymbol{\alpha})\rangle_{q(\mathrm{w})} \\
&= \frac{1}{2}\sum_{m=1}^{M} \ln\alpha_m - \sum_{m=1}^{M} \alpha_m \left\langle w_m^2 \right\rangle \\
&\quad + (a-1)\sum_{m=1}^{M} \ln\alpha_m - b\sum_{m=1}^{M} \alpha_m \\
&= \left(a - \frac{1}{2}\right)\sum_{m=1}^{M} \ln\alpha_m - \sum_{m=1}^{M} \left(\frac{1}{2}\left\langle w_m^2 \right\rangle + b\right) \\
&= \tilde{a}\sum_{m=1}^{M} \ln\alpha_m - \sum_{m=1}^{M} \tilde{b}_m\alpha_m + \mathrm{const}.
\end{aligned} \tag{49}$$

This is the exponent of the product of M independent Gamma distributions with parameters $\tilde{a}$ and $\tilde{b}_m$, given by

$$\tilde{a} = a + 1/2, \tag{50}$$

$$\tilde{b}_m = b + \frac{1}{2}\left\langle w_m^2 \right\rangle. \tag{51}$$

Thus, $q(\boldsymbol{\alpha})$ is given by

$$q(\boldsymbol{\alpha}) = \prod_{m=1}^{M} \text{Gamma}(\alpha_m \,|\tilde{a}, \tilde{b}_m). \tag{52}$$

The posterior distribution of the noise inverse variance can be similarly computed as

$$q(\beta) = \text{Gamma}(\beta \,|\tilde{c}, \tilde{d}_m), \tag{53}$$

with

$$\tilde{c} = c + N/2, \tag{54}$$

$$\tilde{d} = d + \frac{1}{2}\langle \|\mathrm{t} - \boldsymbol{\Phi}\mathrm{w}\|^2 \rangle. \tag{55}$$

The approximate posterior distributions in (48), (52), and (53) are then iteratively updated until convergence, since they depend on the statistics of each other, see [22] for details.

Notice here, that the true prior distribution of the weights can be computed by marginalizing the hyperparameters $\boldsymbol{\alpha}$

$$\begin{aligned} p(\mathrm{w};\, \mathrm{a}, b) &= \int p(\mathrm{w}|\boldsymbol{\alpha}) p(\boldsymbol{\alpha};\, a, b)\, d\boldsymbol{\alpha} \\ &= \int \prod_{m=1}^{M} N\left(w_m \,|0, \alpha_m^{-1}\right) \text{Gamma}(\alpha_m \,|a, b)\, d\alpha_m \\ &= \prod_{m=1}^{m} \text{St}(w_m \,|\lambda, \nu) \end{aligned} \tag{56}$$

and is a student-t pdf,

$$\begin{aligned} \text{St}(x \,|\mu, \lambda, \nu) = \frac{\Gamma((\nu+1)/2)}{\Gamma(\nu/2)} \left(\frac{\lambda}{\pi\nu}\right)^{1/2} \\ \times \left[1 + \frac{\lambda(x-\mu)^2}{\nu}\right]^{-(\nu+1)/2}, \end{aligned}$$

with mean $\mu = 0$, parameter $\lambda = a/b$ and degrees of freedom $\nu = 2a$. This distribution, for small degrees of freedom ν, exhibits very heavy tails. Thus, it favours sparse solutions, which include only few of the basis functions and prunes the remaining basis functions by setting the corresponding weights to very small values. Those basis functions that are actually used in the final model are called relevance basis functions.

For simplicity, we have assumed fixed the parameters a, b, c, and d of the student-t distributions. In practice, we can often obtain good results by assuming uninformative distributions, which are obtained by setting these parameters to very small values, i.e., $a = b = c = d = 10^{-6}$. Alternatively, we can estimate these parameters using a variational EM algorithm. Such an algorithm would add an M-step to the described method, in which the variational bound would be maximized with respect to these parameters. However, the typical approach in Bayesian modeling is to fix the hyperparameters to define uninformative hyperpriors at the highest level of the model.

LINEAR REGRESSION EXAMPLES

Next, we present numerical examples to demonstrate the properties of the previously described linear regression models. We also demonstrate the advantages that can be reaped by using the variational Bayesian inference. An artificially generated signal $y(\mathrm{x})$ is used, so that the original signal which generated the observations is known and therefore the quality of the estimations can be evaluated. We have obtained $N = 50$ samples of the signal and added Gaussian noise of variance $\sigma^2 = 4 \times 10^{-2}$, which corresponds to signal to noise ratio SNR = 6.6 dB. We used N basis functions and, specifically, one basis function centred at the location of each signal observation. The basis functions were Gaussian kernels of the form

$$\phi_i(\mathrm{x}) = K(\mathrm{x}, \mathrm{x}_i) = \exp\left(-\frac{1}{2\sigma_\phi^2}\|\mathrm{x} - \mathrm{x}_i\|^2\right). \tag{57}$$

In this example we set $a = b = 0$, in order to obtain a very heavy-tailed, uninformative Student-t distribution.

We then used the observations to predict the output of the signal, using i) ML estimation (22), ii) EM-based Bayesian inference (33), and iii) variational EM-based Bayesian inference (46). Results are shown in Figure 4(a). Notice that the ML estimate follows exactly the noisy observations. Thus, it is the worst in terms of mean square error. This should be expected, since in this formulation we use as many basis functions as the observations and there is no constraint on the weights. The Bayesian methodology overcomes this problem since the weights are constrained by the priors. However, since this signal contains regions with large variance and some with very small variance, it is clear that the stationary prior is not capable of accurately modeling its local behavior. In contrast, the hierarchical nonstationary prior is more flexible and seems to achieve better local fit. Actually, the solution corresponding to the latter prior, uses only a small subset of the basis functions, whose locations are shown as circled observations in Figure 4. This happens because we have set $a = b = 0$, which defines an uninformative Student-t distribution. Therefore, most weights are estimated to be exactly zero and only few basis functions are used in the signal estimation. Those basis functions that are actually used in the final model are called relevance basis function and the vectors where they are centered are called relevance vectors (RV) and are shown in Figure 4.

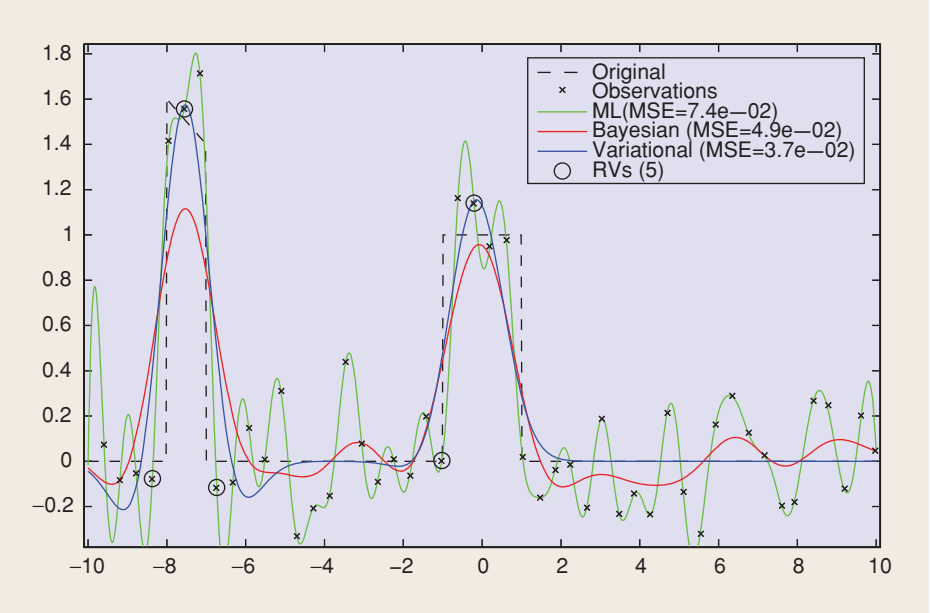
In the same spirit as this example, a hierarchical nonstationary prior has been proposed for the image restoration, image super-resolution, and image blind deconvolution problems in [23], [24], and [25], respectively. In image reconstruction problems, such priors demonstrated the ability to preserve image edges and at the same time suppress noise in flat areas of the image. In addition, priors of this nature have also been used with success to design watermark detectors when the underlying image is unknown [26].

GAUSSIAN MIXTURE MODELS

Gaussian mixture models (GMM) are a valuable statistical tool for modeling densities. They are flexible enough to approximate

any given density with high accuracy and in addition, they can be interpreted as a soft clustering solution. They have been widely used in a variety of signal processing problems ranging from speech understanding, image modeling, tracking, segmentation, recognition, watermarking, and denoising.

A GMM is defined as a convex combination of Gaussian densities and is widely used to describe the density of a dataset in cases where a single distribution does not suffice. To define a mixture model with M components we have to specify the probability density $p_j(\mathrm{x})$ of each component j as well as the probability vector $(\pi_1, \ldots, \pi_M)$ containing the mixing weights π_j of the components ($\pi_j \geq 0$ and $\sum_{j=1}^{M} \pi_j = 1$).



[FIG4] **Linear regression solutions obtained by ML estimation, EM-based Bayesian inference and variational-EM Bayesian inference.**

An important assumption when using such a mixture to model the density of a dataset X is that each datum has been generated using the following procedure:

1) We randomly sample one component k using the probability vector $\pi_1, \ldots, \pi_M$.
2) We generate an observation by sampling from the density $p_k(\mathrm{x})$ of component k.

The graphical model corresponding to above generation process is presented in Figure 5(b), where the discrete hidden random variable Z represents the component that has been selected to generate an observed sample x, i.e., to assign the value $\mathrm{X} = \mathrm{x}$ to the observed random variable X. In this graphical model, the node distributions are $P(Z = j) = \pi_j$ and $P(\mathrm{X} = \mathrm{x}|Z = j) = p_j(\mathrm{x})$. For the joint pdf of X and Z it holds that

$$p(\mathrm{X}, Z) = p(\mathrm{X}|Z)p(Z) \tag{58}$$

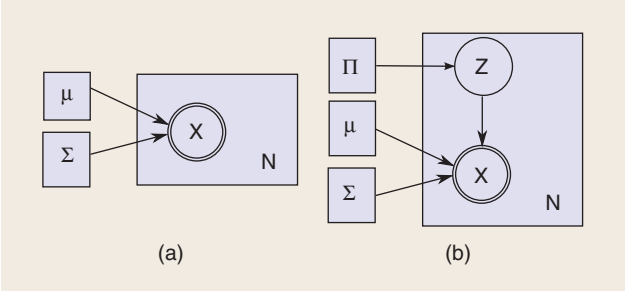
and through marginalization of Z we obtain the well-known formula for mixture models

$$p(\mathrm{X} = \mathrm{x}) = \sum_{j=1}^{M} p(\mathrm{X} = \mathrm{x}|Z = j)p(Z = j) = \sum_{j=1}^{M} \pi_j p_j(\mathrm{x}). \tag{59}$$

In the case of GMMs, the density of each component j is $p_j(\mathrm{x}) = N(\mathrm{x}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ where $\boldsymbol{\mu}_j \in \Re^d$ denotes the mean and $\boldsymbol{\Sigma}_j$ is the $d \times d$ covariance matrix. Therefore

$$p(\mathrm{x}) = \sum_{j=1}^{M} \pi_j N(\mathrm{x}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j). \tag{60}$$

A notable convenience in mixture models is that using Bayes' theorem it is straightforward to compute the posterior



[FIG5] **Graphical models (a) for a single Gaussian component, (b) for a Gaussian mixture model.**

probability $P(j|\mathrm{x}) = p(Z = j|\mathrm{x})$ that an observation x has been generated by sampling from the distribution of mixture component j

$$P(j|\mathrm{x}) = \frac{p(\mathrm{x}|Z = j)p(Z = j)}{p(\mathrm{x})} = \frac{\pi_j N(\mathrm{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_{l=1}^{M} \pi_l N(\mathrm{x}|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}. \tag{61}$$

This probability is sometimes referred to as the responsibility of component j for generating observation x. In addition, by assigning a data point x to the component with maximum posterior, it is easy to obtain a clustering of the dataset X into M clusters, with one cluster corresponding to each mixture component.

EM FOR GMM TRAINING

Let $\mathrm{X} = \{\mathrm{x}_n | \mathrm{x}_n \in \Re^d, n = 1, \ldots, N\}$ denote a set of data points to be modeled using a GMM with M components: $p(\mathrm{x}) = \sum_{j=1}^{M} \pi_j N(\mathrm{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$. We assume that the number of components M is specified in advance. The vector θ of

mixture parameters to be estimated consists of the mixing weights and the parameters of each component, i.e., $\boldsymbol{\theta} = \{\pi_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j \,|\, j = 1, \ldots, M\}$.

Parameter estimation can be achieved through the maximization of the log-likelihood

$$\boldsymbol{\theta}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} \log p(\mathbf{X}; \boldsymbol{\theta}), \tag{62}$$

where assuming independent and identically distributed observations the likelihood can be written as

$$p(\mathbf{X}; \boldsymbol{\theta}) = \prod_{n=1}^{N} p(\mathbf{x}_n; \boldsymbol{\theta}) = \prod_{n=1}^{N} \sum_{j=1}^{M} \pi_j N(\mathbf{x}_n; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j). \tag{63}$$

From the graphical model in Figure 5(b) it is clear that to each observed variable $\mathbf{x}_n \in \mathbf{X}$ corresponds a hidden variable $\mathbf{z}_n$ representing the component that was used to generate $\mathbf{x}_n$. This hidden variable can be represented using a binary vector with M elements z_{jn}, such that $z_{jn} = 1$ if $\mathbf{x}_n$ has been generated from mixture component j and $z_{jn} = 0$ otherwise. Since $z_{jn} = 1$ with probability π_j and $\sum_{j=1}^{M} \pi_j = 1$, then $\mathbf{z}_n$ follows the multinomial distribution. Let $\mathbf{Z} = \{\mathbf{z}_n, n = 1, \ldots, N\}$ denote the set of hidden variables. Then $p(\mathbf{Z}|\boldsymbol{\theta})$ is written

$$p(\mathbf{Z}; \boldsymbol{\theta}) = \prod_{n=1}^{N} \prod_{j=1}^{M} \pi_j^{z_{jn}} \tag{64}$$

and

$$p(\mathbf{X}|\mathbf{Z}; \boldsymbol{\theta}) = \prod_{n=1}^{N} \prod_{j=1}^{M} N(\mathbf{x}_n; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)^{z_{jn}}. \tag{65}$$

As previously noted, the convenient issue with mixture models is that we can easily compute the exact posterior $p(z_{jn} = 1|\mathbf{x}_n; \boldsymbol{\theta})$ of the hidden variables given the observations using (61). Therefore application of the exact EM algorithm is feasible.

More specifically, if $\boldsymbol{\theta}^{(t)}$ denotes the parameter vector at EM iteration t, the expected value of the posterior $p(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta}^{(t)})$ of hidden variables z_{jn} is given as

$$\left\langle z_{jn}^{(t)} \right\rangle = \frac{\pi_j^{(t)} N\left(\mathbf{x}_n; \boldsymbol{\mu}_j^{(t)}, \boldsymbol{\Sigma}_j^{(t)}\right)}{\sum_{j=1}^{M} \pi_j^{(t)} N\left(\mathbf{x}_n; \boldsymbol{\mu}_j^{(t)}, \boldsymbol{\Sigma}_j^{(t)}\right)}. \tag{66}$$

The above equation specifies the computations that should be performed in the E-step for $j = 1, \ldots, M$ and $n = 1, \ldots, N$.

The expected value of the complete log-likelihood $\log P(\mathbf{X}, \mathbf{Z})$ with respect to the posterior $p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)})$ is given by

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}) &= \langle \log p(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta}) \rangle_{p(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta}^{(t)})} \\ &= \langle \log p(\mathbf{X}|\mathbf{Z}; \boldsymbol{\theta}) \rangle + \langle \log p(\mathbf{Z}; \boldsymbol{\theta}) \rangle_{p(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta}^{(t)})} \\ &= \sum_{n=1}^{N} \sum_{j=1}^{M} \left\langle z_{jn}^{(t)} \right\rangle \log \pi_j \\ &\quad + \sum_{n=1}^{N} \sum_{j=1}^{M} \left\langle z_{jn}^{(t)} \right\rangle \log N(\mathbf{x}_n; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j). \end{aligned} \tag{67}$$

In the M-step the expected complete log-likelihood Q is maximized with respect to the parameters $\boldsymbol{\theta}$. Taking the corresponding partial derivatives equal to zero and using a Lagrange multipliers for the constraint $\sum_{j=1}^{M} \pi_j = 1$, we can derive the following equations for the updates of the M-step:

$$\pi_j^{(t+1)} = \frac{1}{N} \sum_{n=1}^{N} \left\langle z_{jn}^{(t)} \right\rangle \tag{68}$$

$$\boldsymbol{\mu}_j^{(t+1)} = \frac{\sum_{n=1}^{N} \left\langle z_{jn}^{(t)} \right\rangle \mathbf{x}_n}{\sum_{n=1}^{N} \left\langle z_{jn}^{(t)} \right\rangle} \tag{69}$$

$$\boldsymbol{\Sigma}_j^{(t+1)} = \frac{\sum_{n=1}^{N} \left\langle z_{jn}^{(t)} \right\rangle \left(\mathbf{x}_n - \boldsymbol{\mu}_j^{(t)}\right) \left(\mathbf{x}_n - \boldsymbol{\mu}_j^{(t)}\right)^T}{\sum_{n=1}^{N} \left\langle z_{jn}^{(t)} \right\rangle}. \tag{70}$$

The above update equations for GMM training are quite simple and easy to implement. They constitute a notable example on how the employment of EM may facilitate the solution of likelihood maximization problems.

A possible problem of the above approach is related to the fact that the covariance matrices may become singular, as shown in the example presented Figure 6. This figure provides the contour plot of a solution obtained when applying EM to train a GMM with 20 components on a two-dimensional (2-D) dataset. It is clear that some of the GMM components are singular, i.e., their density is concentrated around a data point and their variance along some principal axis tends to zero. Another drawback of the typical ML approach for GMM training is that it cannot be used for model selection, i.e., determination of the number of components. A solution to those issues may be obtained by using Bayesian GMMs.

VARIATIONAL GMM TRAINING

FULL BAYESIAN GMM

Let $\mathbf{X} = \{\mathbf{x}_n\}$ be a set of N observations, where each $\mathbf{x}_n \in \Re^d$ is a feature vector. Let also $p(\mathbf{x})$ be a mixture with M Gaussian components

$$p(\mathbf{x}) = \sum_{j=1}^{M} \pi_j N(x; \boldsymbol{\mu}_j, \mathbf{T}_j), \tag{71}$$

where $\boldsymbol{\pi} = \{\pi_j\}$ are the mixing coefficients (weights), $\boldsymbol{\mu} = \{\boldsymbol{\mu}_j\}$ the means (centers) of the components, and $\mathbf{T} = \{\mathbf{T}_j\}$ the precision (inverse covariance) matrices (it must be noted that in Bayesian GMMs it is more convenient to use the precision matrix instead of the covariance matrix).

A Bayesian Gaussian mixture model is obtained by imposing priors on the model parameters $\boldsymbol{\pi}$, $\boldsymbol{\mu}$ and $\mathbf{T}$. Typically conjugate priors are used, that is Dirichlet for $\boldsymbol{\pi}$ and Gauss-Wishart for $(\boldsymbol{\mu}, \mathbf{T})$. The Dirichlet prior for $\boldsymbol{\pi}$ with parameters $\{\alpha_j\}$ is given by

$$\mathrm{Dir}(\boldsymbol{\pi}|\alpha_1,\ldots,\alpha_M)=\frac{\Gamma\left(\sum_{j=1}^{M}\alpha_j\right)}{\prod_{j=1}^{M}\Gamma(\alpha_j)}\prod_{j=1}^{M}\pi_j^{\alpha_j-1},$$

where $\Gamma(x)$ denotes the Gamma function. Usually, we assume that all α_j are equal, i.e., $\alpha_j=\alpha_0$, $j=1,\ldots,M$.

The Gauss-Wishart prior for $(\boldsymbol{\mu},\mathrm{T})$ is $p(\boldsymbol{\mu},\mathrm{T})=\prod_{j=1}^{M}p(\boldsymbol{\mu}_j,\mathrm{T}_j)=\prod_{j=1}^{M}p(\boldsymbol{\mu}_j|\mathrm{T}_j)p(\mathrm{T}_j)$, where $p(\boldsymbol{\mu}_j|\mathrm{T}_j)=N(\boldsymbol{\mu}_j;\boldsymbol{\mu}_0,\beta_0\mathrm{T}_j)$ (with parameters $\boldsymbol{\mu}_0$ and β_0) and $p(\mathrm{T}_j)$ is the Wishart distribution

$$\mathrm{W}(\mathrm{T}_j|\nu,\mathrm{V})=\frac{|\mathrm{T}_j|^{(\nu-d-1)/2}\exp\mathrm{tr}\left\{-\frac{1}{2}\mathrm{V}\mathrm{T}_j\right\}}{2^{\nu d/2}\pi^{d(d-1)/4}|\mathrm{V}|^{-n/2}\prod_{i=1}^{d}\Gamma((\nu+1-i)/2)},$$

with parameters ν and V denoting the degrees of freedom and the scale matrix respectively. Notice that the Wishart distribution is the multidimensional generalization of the Gamma distribution. In linear regression, we used the Gauss-Gamma prior, assuming independent precisions α_i and thus assigning them independent Gamma prior distributions. Here, however, because there may be significant correlations between data, we could use the Wishart prior to capture these correlations.

The graphical model corresponding to this Bayesian GMM is presented in Figure 7(a). This is a full Bayesian GMM and if all the hyperparameters (i.e., the parameters α,μ_0,β_0,ν and V of the priors) are specified in advance, then the model does not contain any parameter to be estimated, but only hidden random variables $h=(\mathrm{Z},\boldsymbol{\pi},\boldsymbol{\mu},\mathrm{T})$ whose posterior $p(\mathrm{h}|\mathrm{x})$ given the data must be computed. It is obvious that in this case the posterior cannot be computed analytically, thus an approximation $q(\mathrm{h})$ is computed by applying the variational mean field (16) to the specific Bayesian model [27].

The mean-field approximation assumes q to be a product of the form

$$q(\mathrm{h})=q_Z(\mathrm{Z})\,q_\pi(\boldsymbol{\pi})\,q_{\mu T}(\boldsymbol{\mu},\mathrm{T}),\tag{72}$$

and the solution is given by (16). After performing the necessary calculations, the result is the following set of densities:

$$q_Z(\mathrm{Z})=\prod_{n=1}^{N}\prod_{j=1}^{M}r_{jn}^{z_{jn}}\tag{73}$$

$$q_\pi(\boldsymbol{\pi})=\mathrm{Dir}(\boldsymbol{\pi}|\{\lambda_j\})\tag{74}$$

$$q_{\mu T}(\boldsymbol{\mu},\mathrm{T})=\prod_{j=1}^{M}q_\mu(\boldsymbol{\mu}_j|\mathrm{T}_j)q_T(\mathrm{T}_j)\tag{75}$$

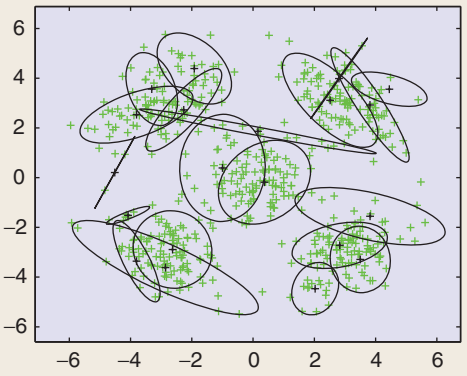
$$q_\mu(\boldsymbol{\mu}_j|\mathrm{T})=\prod_{j=1}^{M}N(\boldsymbol{\mu}_j;\mathrm{m}_j,\beta_j\mathrm{T}_j)\tag{76}$$

$$q_T(\mathrm{T})=\prod_{j=1}^{M}W(\mathrm{T}_j;\eta_j,\mathrm{U}_j)\tag{77}$$

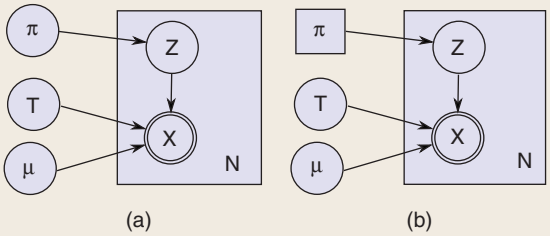
and the detailed formulas for updating the parameters $(r_{jn},\lambda_j,\mathrm{m}_j,\beta_j,\eta_j,\mathrm{U}_j)$ of the densities can be found in [27]. By solving the above system of equations using a simple iterative update procedure, we obtain an optimal approximation $q(\mathrm{h})$ to the true posterior $p(\mathrm{h}|\mathrm{x})$ under the mean-field constraint.

The typical approach in Bayesian modeling is to specify the hyperparameters α, ν, V, μ_0, and β_0 of the model so that uninformative prior distributions are defined. We follow this approach, although it would be possible to incorporate an M-step to the algorithm, in order to adjust these parameters. However, this is usually not followed.

One advantage of the full Bayesian GMM compared to GMM without priors is that it does not allow the singular solutions often arising in the ML approach where a Gaussian component becomes responsible for a single data point. A second advantage is that it is possible to use the Bayesian GMM for directly determining the optimal number of components without resorting to methods such as cross-validation. However, the effectiveness of the full Bayesian mixture is limited for this problem, since the Dirichlet prior does not allow mixing weight of a component to become zero and to be eliminated from the mixture. Also, in this case the final result depends highly on the hyperparameters of the priors (and especially of the parameters of the Dirichlet prior) that must be specified in advance [13]. For a specific set of hyperparameters, it is possible to run the variational algorithm for several values of the number M of mixture components and keep the solution corresponding to the best value of the variational lower bound.



[FIG6] EM-based GMM training using 20 Gaussian components.



[FIG7] (a) Graphical model for the full Bayesian GMM. (b) Graphical model for the Bayesian GMM of [27]. Notice the difference in the role of π in the two models. Also, the parameters of the priors on π, μ, Σ are fixed, thus they are not shown.

REMOVING THE PRIOR FROM THE MIXING WEIGHTS

In [28], another example of a Bayesian GMM model has been proposed that does not assume a prior on the mixing weights $\{\pi_j\}$, which are treated as parameters and not as random variables. The graphical model for this approach is depicted in Figure 7(b).

In this Bayesian GMM, which we will call CB model from the initials of the two authors of this work, Gaussian and Wishart priors are assumed for $\boldsymbol{\mu}$ and $\mathbf{T}$, respectively,

$$p(\boldsymbol{\mu}) = \prod_{j=1}^{M} N(\boldsymbol{\mu}_j | 0, \beta \mathbf{I}) \tag{78}$$

$$p(\mathbf{T}) = \prod_{j=1}^{M} W(\mathbf{T}_j | \nu, \mathbf{V}). \tag{79}$$

This Bayesian model is capable (to some extent) to estimate the optimal number of components. This is achieved through maximization of the marginal likelihood $p(\mathbf{X}; \boldsymbol{\pi})$ obtained by integrating out the hidden variables $\mathbf{h} = \{\mathbf{Z}, \boldsymbol{\mu}, \mathbf{T}\}$

$$p(\mathbf{X}; \boldsymbol{\pi}) = \int p(\mathbf{X}, \mathbf{h}; \boldsymbol{\pi})\, d\mathbf{h} \tag{80}$$

with respect to the mixing weights $\boldsymbol{\pi}$ that are treated as parameters. The variational approximation suggests the maximization of a lower bound of the logarithmic marginal likelihood

$$F[q, \boldsymbol{\pi}] = \int q(\mathbf{h}) \log \frac{p(\mathbf{X}, \mathbf{h}; \boldsymbol{\pi})}{q(\mathbf{h})}\, d\mathbf{h} \leq \log p(\mathbf{X}; \boldsymbol{\pi}), \tag{81}$$

where $q(\mathbf{h})$ is an arbitrary distribution approximating the posterior $p(\mathbf{h}|\mathbf{X})$. A notable property is that during maximization of F, if some of the components fall in the same region in the data space, then there is strong tendency in the model to eliminate the redundant components (i.e., setting their π_j equal to zero), once the data in this region are sufficiently explained by fewer components. Consequently, the competition between mixture components suggests a natural approach for addressing the model selection problem: fit a mixture initialized with a large number of components and let competition eliminate the redundant.

Following the variational methodology, our aim is to maximize the lower bound F of the logarithmic marginal likelihood $\log p(\mathbf{X}; \boldsymbol{\pi})$

$$F[q, \boldsymbol{\pi}] = \sum_{\mathbf{z}} \int q(\mathbf{Z}, \boldsymbol{\mu}, \mathbf{T}) \log \frac{p(\mathbf{X}, \mathbf{Z}, \boldsymbol{\mu}, \mathbf{T}; \boldsymbol{\pi})}{q(\mathbf{Z}, \boldsymbol{\mu}, \mathbf{T})} d\boldsymbol{\mu}\, d\mathbf{T}, \tag{82}$$

where q is an arbitrary distribution that approximates the posterior distribution $p(\mathbf{Z}, \boldsymbol{\mu}, \mathbf{T}|\mathbf{X}; \boldsymbol{\pi})$. The maximization of F is performed in an iterative way using the variational EM algorithm. At each iteration two steps take place: first maximization of the bound with respect to q and, subsequently, maximization of the bound with respect to $\boldsymbol{\pi}$.

To implement the maximization with respect to q, the mean-field approximation has been adopted (14) that assumes q to be a product of the form

$$q(\mathbf{h}) = q_Z(\mathbf{Z})\, q_\mu(\boldsymbol{\mu})\, q_T(\mathbf{T}). \tag{83}$$

After performing the necessary calculations in (16), the result is the following set of densities:

$$q_Z(\mathbf{Z}) = \prod_{n=1}^{N} \prod_{j=1}^{M} r_{jn}^{z_{jn}} \tag{84}$$

$$q_\mu(\boldsymbol{\mu}) = \prod_{j=1}^{M} N(\boldsymbol{\mu}_j | \mathbf{m}_j, \mathbf{S}_j) \tag{85}$$

$$q_T(\mathbf{T}) = \prod_{j=1}^{M} W(\mathbf{T}_j | \eta_j, \mathbf{U}_j), \tag{86}$$

where the parameters of the densities can be computed as

$$r_{jn} = \frac{\tilde{r}_{jn}}{\sum_{k=1}^{M} \tilde{r}_{jn}} \tag{87}$$

$$\tilde{r}_{jn} = \pi_j \exp\left\{ \frac{1}{2} \langle \log |\mathbf{T}_j| \rangle - \frac{1}{2} \mathrm{tr}\left\{ \langle \mathbf{T}_j \rangle \left(\mathbf{x}_n \mathbf{x}_n^T - \mathbf{x}_n \langle \boldsymbol{\mu}_j \rangle^T + \langle \boldsymbol{\mu}_j \rangle \mathbf{x}_n^T + \left\langle \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \right\rangle \right) \right\} \right\} \tag{88}$$

$$\mathbf{m}_j = \mathbf{S}_j^{-1} \langle \mathbf{T}_j \rangle \sum_{n=1}^{N} \langle z_{jn} \rangle\, \mathbf{x}_n \tag{89}$$

$$\mathbf{S}_j = \beta \mathbf{I} + \langle \mathbf{T}_j \rangle \sum_{n=1}^{N} \langle z_{jn} \rangle \tag{90}$$

$$\eta_j = \nu + \sum_{n=1}^{N} \langle z_{jn} \rangle \tag{91}$$

$$\mathbf{U}_j = \mathbf{V} + \sum_{n=1}^{N} \langle z_{jn} \rangle \left(\mathbf{x}_n \mathbf{x}_n^T - \mathbf{x}_n \langle \boldsymbol{\mu}_j \rangle^T + \langle \boldsymbol{\mu}_j \rangle \mathbf{x}_n^T + \left\langle \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \right\rangle \right). \tag{92}$$

The expectations with respect to $q(\mathbf{h})$ used in the above equations satisfy the equations: $\langle \mathbf{T}_j \rangle = \eta_j \mathbf{U}_j^{-1}$, $\langle \log |\mathbf{T}_j| \rangle = \sum_{i=1}^{d} \psi(0.5(\eta_j + 1 - i)) + d \ln 2 - \ln |\mathbf{U}_j|$, $\langle \boldsymbol{\mu}_j \rangle = \mathbf{m}_j$, $\langle \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \rangle = \mathbf{S}_j^{-1} + \mathbf{m}_j \mathbf{m}_j^T$ [Here ψ denotes the digamma function, defined as $d/dx \ln \Gamma(x) = \Gamma'(x)/\Gamma(x)$] and $\langle z_{jn} \rangle = r_{jn}$. It can be observed that the densities are coupled through their expectations, thus an iterative estimation of the parameters is needed. However, in practice a single pass seems to be sufficient for the variational E-step.

After the maximization of F with respect to q, the second step of each iteration of the training method requires maximization of F with respect to $\boldsymbol{\pi}$, leading to the following simple update equation for the variational M-step:

$$\pi_j = \frac{\sum_{n=1}^{N} r_{jn}}{\sum_{k=1}^{M} \sum_{n=1}^{N} r_{kn}}. \tag{93}$$

The above variational EM update equations are applied iteratively and converge to a local maximum of the variational

bound. During the optimization some of the mixing coefficients converge to zero thus the corresponding components are eliminated from the mixture. In this way complexity control is achieved. This happens because the prior distribution on $\boldsymbol{\mu}$ and $\mathbf{T}$ penalizes overlapping components. Qualitatively speaking, the variational bound can be written as a sum of two terms: the first one is a likelihood term (that depends on the quality of data fitting) and the other is a penalty term due to the priors that penalizes complex models.

Figure 8 provides an illustrative example of the performance on this method using the 2-D dataset already presented in Figure 6. The method starts with 20 components and, as the number of iterations increases, the number of components gradually decreases (some π_j become zero) and, finally, a good GMM model for this dataset is attained. It can also be observed, that the existence of the prior on the covariance matrices, does not allow to reach singular solutions in contrast to the GMM solution without priors presented in Figure 6.

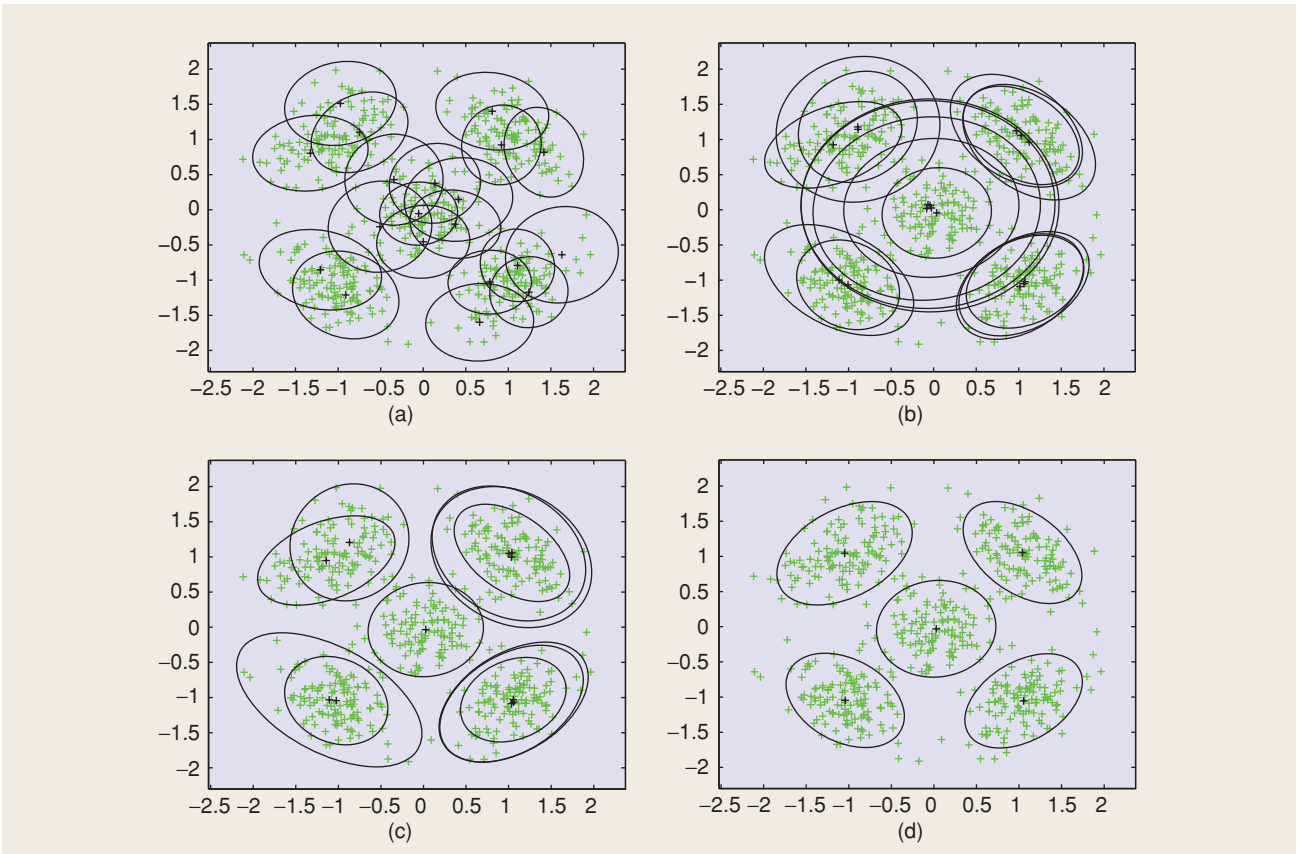
In general, the CB constitutes an effective method exhibiting good performance in the case where the components are well separated. However, its performance exhibits sensitivity on the specification of the scale matrix $\mathbf{V}$ of the Wishart prior imposed on the precision matrix. An incremental method for building the above mixture model has been proposed in [29]. At each step, learning is restricted in the data region occupied by a specific mixture component j, thus a local precision prior can be specified based on the precision matrix $\mathbf{T}_j$. In order to achieve this behavior, a modification to the generative model of Figure 7 was made that restricts the competition in a subset of the components only.

SUMMARY

The EM algorithm is an iterative methodology that offers a number of advantages for ML estimation. It provides simple iterative solutions, with guaranteed local convergence, for problems where direct optimization of the likelihood function is difficult. In many cases it provides solutions that satisfy several constraints for the estimated parameters, for example covariance matrices are positive definite, probability vectors are positive, and sum to one, etc. Furthermore, the application of EM does not require explicit evaluation of the likelihood function.

However, to apply the EM algorithm we must have knowledge of the posterior of the hidden variables given the observations. This is a serious drawback since the EM cannot be applied to complex Bayesian models. However, complex Bayesian models can be very useful since, if properly constructed, they have the ability to model salient properties of the data generation mechanism and provide very good solutions to difficult problems.

The variational methodology is an iterative approach that is gaining popularity within the signal processing community to ameliorate this shortcoming of the EM algorithm.



[FIG8] Variational Bayesian GMM training using the model presented in [28]. (a) Initialization with 20 Gaussian components, (b), (c) model evolution during EM iterations, and (d) final solution. Notice the avoidance of singularities.

According to this methodology an approximation to the posterior of the hidden variables given the observations is used. Based on this approximation, Bayesian inference is possible by maximizing a lower bound of the likelihood function which also guarantees local convergence. This methodology allows inference in the case of complex graphical models, that in certain cases provide significant improvements as compared to simpler ones that can be solved via the EM.

This issue was demonstrated in this article within the context of linear regression and Gaussian mixture modeling, which are two fundamental problems for signal processing applications. More specifically, we demonstrated that complex Bayesian models that were solved by the variational methodology, in the context of linear regression were able to better capture local signal properties and avoid ringing in areas of signal discontinuities. In the context of Gaussian mixture modeling, the models solved by the variational methodology were able to avoid singularities and to estimate the number of the model components. These results demonstrate the power of the variational methodology to provide solution to difficult problems that have plagued signal processing applications for a long time. The main drawback of this methodology is the lack of results that allow (at least for the time being) assessing the tightness of the bound that is used.

AUTHORS

Dimitris G. Tzikas (tzikas@cs.uoi.gr) received a B.Sc. degree in informatics and telecommunications from the University of Athens, Greece, in 2002 and a M.Sc. degree from the University of Ioannina, Greece, in 2004. He is a Ph.D. candidate in the Department of Computer Science, University of Ioannina. His research interests include machine learning, Bayesian methods, and statistical image processing.

Aristidis C. Likas (arly@cs.uoi.gr) received the diploma degree in electrical engineering and the Ph.D. degree in electrical and computer engineering, both from the National Technical University of Athens, Greece. Since 1996, he has been with the Department of Computer Science, University of Ioannina, Greece, where he is currently an associate professor. His research interests include machine learning, neural networks, statistical signal processing, and bioinformatics. He is a Senior Member of the IEEE.

Nickolaos P. Galatsanos (ngalatsanos@upatras.gr) received the diploma of electrical engineering from the National Technical University of Athens, Greece in 1982. He received the M.S.E.E. and Ph.D. degrees from the Electrical and Computer Engineering Department of the University of Wisconsin, Madison, in 1984 and 1989, respectively. He is with the Department of Electrical and Computer Engineering of the University of Patras-Greece. His research interests include Bayesian methods for image processing, medical imaging, bioinformatics, and visual communications applications. He was an associate editor for *IEEE Transactions on Image Processing* and *IEEE Signal Processing Magazine* and currently is an associate editor for the *Journal of Electronic Imaging*. He coedited *Image Recovery Techniques for Image and Video Compression and Transmission*. He is a Senior Member of the IEEE.

REFERENCES

[1] S.M. Stigler, "Thomas Bayes's inference," *J. Roy. Statis. Soc. A*, vol. 145, pp. 250–258, 1982.

[2] P.S. Laplace, "Mémoire sur la probabilité des causes par les événements," *Mémoires de mathématique et de physique presentés á l'Académie royale des sciences par divers savants & lus dans ses assemblées*, vol. 6, pp. 621–656, 1774.

[3] S.M. Stigler, "Laplace's 1774 memoir on inverse probability," *Statist. Sci.*, vol. 1, no. 3, pp. 359–363, 1986.

[4] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm,"*J. Roy. Statis. Soc. A*, vol. 39, no. 1, pp. 1–38, 1977.

[5] W. Jones, L. Byars, and M. Casey, "Positron emission tomographic images and expectation maximization: A VLSI architecture for multiple iterations per second," *IEEE Trans. Nuclear Sci.*, vol. 35, no. 1, pp. 620–624, 1988.

[6] Z. Liang and H. Hart, "Bayesian reconstruction in emission computerized tomography," *IEEE Trans. Nuclear Sci.*, vol. 35, no. 1, pp. 788–792, 1988.

[7] S. Newcomb, "A generalized theory of the combination of observations so as to obtain the best result," *Amer. J. Math.*, vol. 8, pp. 343–366, 1886.

[8] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. New York: Wiley, 1997.

[9] T.K. Moon, "The EM algorithm in signal processing," *IEEE Signal Process. Mag.*, vol. 13, no. 6, pp. 47–60, 1996.

[10] V. Smídl and A. Quinn, *The Variational Bayes Method in Signal Processing*. New York: Springer-Verlag, 2005.

[11] C. Robert and G. Cassela, *Monte Carlo Statistical Methods*. New York: Springer-Verlag, 1999.

[12] C. Andrieu, N. de Freitas, A. Doucet, and M. Jordan, "An introduction to MCMC for machine learning," *Mach. Learn.*, vol. 50, no. 1, pp. 5–43, 2003.

[13] C. Bishop, *Pattern Recognition and Machine Learning*. New Yoek: Springer- Verlag, 2006.

[14] C. Borgelt and R. Kruse, *Graphical Models Methods for Data Analysis and Mining*. Hoboken, NJ: Wiley, 2002.

[15] R. Neapolitan, *Learning Bayesian Networks*. Englewood Cliffs, NJ: Prentice- Hall, 2004.

[16] R.M. Neal and G.E. Hinton, "A view of the EM algorithm that justifies incremental, sparse and other variants," in *Learning in Graphical Models*, M.I. Jordan, Ed. Cambridge, MA: MIT Press, 1998, pp. 355–368.

[17] S.M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[18] R. Weinstock, *Calculus of Variations*. New York: Dover, 1974.

[19] T.S. Jaakkola, "Variational methods for inference and learning in graphical models," Ph.D. dissertation, Dept. Elect. Eng. Comp. Sci., MIT, 1997.

[20] M. Jordan, Z. Gharamani, T. Jaakkola, and L. Saul, "An introduction to variational methods for graphical models," in *Learning in Graphical Models*, M. Jordan, Ed., Cambridge, MA: MIT Press, 1998, pp. 105–162.

[21] G. Parisi, *Statistical Field Theory*. Reading, MA: Addison-Wesley, 1988.

[22] C. Bishop and M. Tipping, "Variational Relevance Vector Machines," in *Proc. 16th Conf. Uncertainty in Artificial Intelligence*, 2000, pp. 46–53.

[23] G. Chantas, N. Galatsanos, and A. Likas, "Bayesian restoration using a new nonstationary edge preserving image prior," *IEEE Trans. Image Processing*, vol. 15, no. 10, pp. 2987–2997, Oct. 2006.

[24] G. Chantas, N. Galatsanos, and N. Woods, "Super resolution based on fast registration and maximum a posteriori reconstruction," *IEEE Trans. Image Processing*, vol. 16, no. 7, pp. 1821–1830, July 2007.

[25] D. Tzikas, A. Likas, and N. Galatsanos, "Variational Bayesian Blind Image Deconvolution with Student-T Priors," in *Proc. IEEE Int. Conf. Image Processing, (ICIP)*, San Antonio, TX, Sept. 2007, vol. 1, pp. 109–112.

[26] A. Mairgiotis, N. Galatsanos, and Y. Yang, "New additive watermark detectors based on a hierarchical spatially adaptive image model," *IEEE Trans. Information Forensics and Security*, vol. 3, no. 1, pp. 29–37, Mar. 2008.

[27] H. Attias, "A Variational Bayesian Framework for Graphical Models," *Proc. NIPS 12*, MIT Press, 2000, pp. 209–216.

[28] A. Corduneanu and C. Bishop, "Variational bayesian model selection for mixture distributions," in *Proc. AI and Statistics Conf.*, Jan. 2001, pp. 27–34.

[29] C. Constantinopoulos and A. Likas, "Unsupervised learning of gaussian mixtures based on variational component splitting," *IEEE Trans. Neural Networks*, vol. 18, no. 3, pp. 745–755, Mar. 2007.

[30] K. Blekas, A. Likas, N. Galatsanos, and I.E. Lagaris, "A spatially-constrained mixture model for image segmentation," *IEEE Trans. Neural Networks*, vol. 16, no. 2, pp. 494–498, 2005.

[31] C. Nikou, N. Galatsanos, and A. Likas, "A class-adaptive spatially variant mixture model for image segmentation," *IEEE Trans. Image Processing*, vol. 16, no. 4, pp. 1121–1130, Apr. 2007. SP