On Expectation Propagation for Generalized, Linear and Mixed Models

BY ANDY S.I. KIM AND MATT P. WAND

University of Technology Sydney and
Australian Research Council Centre of Excellence
for Mathematical and Statistical Frontiers

21st April, 2017

Summary

Expectation propagation is a general approach to deterministic approximate Bayesian inference for graphical models, although its literature is confined mostly to machine learning applications. We investigate the utility of expectation propagation in generalized, linear, and mixed model settings. We show that, even though the algebra and computations are complicated, the notion of message passing on factor graphs affords streamlining of the required calculations and we list the algorithmic steps explicitly. Numerical studies indicate expectation propagation is marginally more accurate than a competing method for the models considered, but at the expense of bigger algebraic and computational overheads.

Keywords: Bayesian computing; Factor graphs; Infer.NET; Mean field variational Bayes, Message passing.

1 Introduction

Fitting and inference for linear models and their various extensions continues to be a major area of research, with technological advances constantly changing the landscape. Increases in sizes of data sets and complexity of models are necessitating agile methodological development. One manifestation of this state of affairs is the emergence of *Bayesian inference engines*, such as Infer.NET (Minka *et al.*, 2014) and Stan (Stan Development Team, 2016), that utilise fast deterministic algorithms such as those based on variational approximations.

In this article our focus is the fast approximate inference paradigm known as *expectation propagation* (Minka, 2001; Minka, 2005). In Kim & Wand (2016) we derived the explicit form of expectation propagation for the simple statistical problem of Bayesian inference from independent and identically distributed observations from a Normal distribution. Here, we investigate extensions to linear model scenarios. McCulloch, Searle & Neuhaus (2008) provides a particularly attractive coverage of these families of models and we borrow their succinct title *Generalized, Linear, and Mixed Models* in titling this article. Linear models, generalized linear models, linear mixed models and generalized linear mixed models is a fuller description of the types of models covered here. Various semiparametric regression models, such as generalized additive models, can be embedded within these structures courtesy of mixed model-based penalized splines (e.g. Ruppert *et al.*, 2009).

Expectation propagation is an alternative to mean field variational Bayes (e.g. Wainwright & Jordan, 2008) for potentially fast approximate inference in large statistical models. There is some empirical evidence (e.g. Minka, 2001; Kim & Wand, 2016) that expectation propagation is the more accurate of the two, although this needs to be traded off against its onerous algebraic and computational overheads. However, the literature on ex-

pectation propagation in statistical contexts is rather scant. For example, the algorithmic details needed to fit a Gaussian response linear regression model are difficult to discern from any of the existing literature of which we are aware. Removal of this obstacle is one of our major contributions. The upcoming sections provide explicit ready-to-implement details on expectation propagation for a wide range of generalized, linear and mixed models. As we shall see, the derivations for these models are very similar to the derivations given in Kim & Wand (2016). Therefore we will regularly refer to this earlier article of ours. It is essential background reading for the current article.

Our treatment of expectation propagation, geared towards linear models, can be contrasted with the existing relevant literature as follows. The very general formulation of expectation propagation in Appendix B of Minka & Winn (2008), in terms of message passing on factor graphs, has the beauty of distilling the iterative updates to just two equations. However, as demonstrated in Kim & Wand (2016), a great deal of algebra is required to obtain the explicit forms needed for actual implementation. An additional issue is that, for linear models, Minka & Winn's equations require the extension of ordinary factor graphs to a structure that we call *derived variable* factor graphs. However, this subtlety is not spelt out in Appendix B of Minka & Winn (2008). These comments also apply to the factor graph-based description of expectation propagation given in Section 10.7.2 of Bishop (2006). Gelman *et al.* (2014, Section 13.8) explains expectation propagation mainly by way of a Bayesian logistic regression illustrative example. Factor graphs are not used, but rather they give explicit expressions that facilitate implementation for this special case. A drawback is that it is quite difficult to transfer their description of expectation propagation to, say, Gaussian response linear models and logistic mixed models.

The message passing approach to expectation propagation has the advantage that messages of particular forms only have to be worked out once and can be re-used in models with similar components. Indeed, many of the message types arising in the Univariate Normal model treated in Kim & Wand (2016) also arise in the generalized, linear and mixed models treated here and the amount of additional algebra required is relatively small. As explained in Wand (2017), message passing approaches afford compartmentalisation of the algebra and facilitates efficient extension to arbitrarily large models. The Infer.NET package exploits this fact.

The models dealt with in this article are supported, essentially, in Infer.NET. Descriptions of the type of code needed to fit linear models and various extensions are included in Wang & Wand (2011) and Luts *et al.* (2017), albeit with a few 'tricks'. However, in these articles there is an emphasis on Infer.NET's mean field variational Bayes inference engine rather than the expectation propagation alternative. Given the existence of Infer.NET, our main contributions are providing the explicit details for expectation propagation fitting of such models and facilitating extensions beyond capabilities of Infer.NET or any other software products that support expectation propagation to varying degrees of sophistication.

We also contribute to the relatively small literature on numerical evaluation of expectation propagation and comparison with comparable mean field variational Bayes methods. In the settings considered, expectation propagation is very accurate and slightly ahead of mean field variational Bayes in this regard. However, it is much more algebraically and computationally demanding and the accuracy improvements are not practically significant. Therefore the overall superiority of expectation propagation is not borne out by our investigations.

In Section 2 we provide the various background definitions and results needed for our exposition on expectation propagation for generalized, linear and mixed models. Section 3 focusses exclusively on the Gaussian response linear model situation. As we will see, most of the mechanics of expectation propagation for the linear model can be explained in this familiar setting. Then we progress through extensions to non-Gaussian responses and random effects model structures in Sections 4 and 5, which enables handling of generalized linear mixed models and their various special cases. Section 6 reports on our

numerical investigations into practical implementation and performance of expectation propagation for linear models and linear mixed models. We close with some discussion, which includes connections with the career of Peter Hall, in Section 7.

2 Background Material

Our description of expectation propagation for generalized, linear, and mixed models requires various definitions, concepts and results to be laid down. We do so in this section.

2.1 Function Definitions

In Kim & Wand (2016) we defined the following non-analytic functions, which are also needed here:

$$\begin{aligned} \mathcal{A}(p,q,r,s,t,u) &\equiv \int_{-\infty}^{\infty} \frac{x^p \exp(qx - rx^2)\,dx}{(x^2+sx+t)^u}, \\ &\qquad p \geq 0,\ q \in \mathbb{R}, r > 0, s \in \mathbb{R}, t > \tfrac{1}{4}\,s^2, u > 0 \\ \text{and} \quad \mathcal{B}(p,q,r,s,t,u) &\equiv \int_{-\infty}^{\infty} \frac{x^p \exp\{qx - re^x - se^x/(t+e^x)\}\,dx}{(t+e^x)^u}, \\ &\qquad p \geq 0,\ q \in \mathbb{R}, r > 0, s \geq 0, t > 0, u > 0. \end{aligned} \tag{1}$$

An additional family of non-analytic functions that we need is:

$$\mathcal{C}_b(p,q,r) \equiv \int_{-\infty}^{\infty} x^p \exp\{qx - rx^2 - b(x)\}\,dx,$$

where $q \in \mathbb{R}$, $r > 0$ and $b : \mathbb{R} \to \mathbb{R}$ is any function for which $\mathcal{C}_b(p,q,r)$ exists. As discussed in Section 2.1 of Kim & Wand (2016), it is advisable to work with the logarithms of these integrals to avoid overflow and underflow.

The functions G^{N}, G^{IG1} and G^{IG2} defined in Kim & Wand (2016) are also needed here. They involve the $\mathcal{A}$ and $\mathcal{B}$ functions and the inverse of the function $(\log - \text{digamma})$, also discussed in Kim & Wand (2016). Otherwise G^{N}, G^{IG1} and G^{IG2} are simple, albeit long-winded, functions with multiple vector arguments. Their definitions are given in Section A.4 of Kim & Wand (2016) and are repeated here for convenience. First we need:

$$\alpha\left(k, \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}\right) \equiv \mathcal{A}\left(k, a_1, -a_2, \frac{-2c_2}{c_1}, \frac{c_3 - 2b_2}{c_1}, \frac{c_1 - 2b_1 - 2}{2}\right)$$

and

$$\begin{aligned} &\beta\left(k, \ell, v, w, \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}\right) \equiv \\ &\qquad \mathcal{B}\left(k, \frac{\ell + c_1 - 1}{2} - a_1, \frac{c_1c_3 - c_2^2}{2c_1} - a_2, -b_2\left(\frac{c_2}{c_1} + \frac{b_1}{2b_2}\right)^2, v, w\right). \end{aligned}$$

Then let

$$\begin{aligned} g(\ell, v, w, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \equiv (\log - \text{digamma})^{-1}\Bigg(&\log\left\{\frac{\beta(0, \ell+1, v, w, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})}{\beta(0, \ell-1, v, w, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})}\right\} \\ &- \frac{\beta(1, \ell-1, v, w, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})}{\beta(0, \ell-1, v, w, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})}\Bigg). \end{aligned}$$

We are now in a position to give the expressions for G^{N}, G^{IG1} and G^{IG2}:

$$G^{N}(\boldsymbol{a},\boldsymbol{b};\boldsymbol{c}) \equiv \left[\frac{\alpha(2,\boldsymbol{a},\boldsymbol{b},\boldsymbol{c})}{\alpha(0,\boldsymbol{a},\boldsymbol{b},\boldsymbol{c})} - \left\{\frac{\alpha(1,\boldsymbol{a},\boldsymbol{b},\boldsymbol{c})}{\alpha(0,\boldsymbol{a},\boldsymbol{b},\boldsymbol{c})}\right\}^2\right]^{-1} \left[\begin{array}{c} \alpha(1,\boldsymbol{a},\boldsymbol{b},\boldsymbol{c})/\alpha(0,\boldsymbol{a},\boldsymbol{b},\boldsymbol{c}) \\ -1/2 \end{array}\right] - \boldsymbol{a},$$

$$G^{\text{IG1}}\left(\boldsymbol{a}, \left[\begin{array}{c} b_1 \\ b_2 \end{array}\right]; \left[\begin{array}{c} c_1 \\ c_2 \\ c_3 \end{array}\right]\right) \equiv \left[\begin{array}{c} -1 - g(0, -2b_2/c_1, \frac{1}{2}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \\ \dfrac{-g(0, -2b_2/c_1, \frac{1}{2}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})\, \beta(0, -1, -2b_2/c_1, \frac{1}{2}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})}{\beta(0, 1, -2b_2/c_1, \frac{1}{2}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})} \end{array}\right] - \boldsymbol{a}$$

and

$$G^{\text{IG2}}\left(\boldsymbol{a}, \left[\begin{array}{c} b_1 \\ b_2 \end{array}\right]; k\right) \equiv$$

$$\left[\begin{array}{c} -1 - g\left(k-2, -b_2, 1-k/2-b_1, \boldsymbol{a}, \left[\begin{array}{c} 0 \\ b_2 \end{array}\right], \left[\begin{array}{c} 2 \\ 0 \\ 0 \end{array}\right]\right) \\ \dfrac{\left\{\begin{array}{c} -g\left(k-2, -b_2, 1-k/2-b_1, \boldsymbol{a}, \left[\begin{array}{c} 0 \\ b_2 \end{array}\right], \left[\begin{array}{c} 2 \\ 0 \\ 0 \end{array}\right]\right) \\ \times\beta\left(0, k-3, -b_2, 1-k/2-b_1, \boldsymbol{a}, \left[\begin{array}{c} 0 \\ b_2 \end{array}\right], \left[\begin{array}{c} 2 \\ 0 \\ 0 \end{array}\right]\right) \end{array}\right\}}{\beta\left(0, k-1, -b_2, 1-k/2-b_1, \boldsymbol{a}, \left[\begin{array}{c} 0 \\ b_2 \end{array}\right], \left[\begin{array}{c} 2 \\ 0 \\ 0 \end{array}\right]\right)} \end{array}\right] - \boldsymbol{a}.$$

The *Dirac delta* function, denoted by δ, is also needed. A basic property of δ is

$$\int_{-\infty}^{\infty} f(x)\delta(x-c)\,dx = f(c) \tag{2}$$

for any function $f : \mathbb{R} \to \mathbb{R}$ and $c \in \mathbb{R}$. Vector arguments of δ are also allowed and evaluate as the product of Dirac delta functions of the components. Specifically,

$$\delta(x_1, \ldots, x_d) = \delta(x_1) \cdots \delta(x_d).$$

For a $d \times d$ matrix $\boldsymbol{A}$ we let $\text{vec}(\boldsymbol{A})$ denote the $d^2 \times 1$ vector obtained by stacking the columns of $\boldsymbol{A}$ underneath each other in order from left to right. For a $d^2 \times 1$ vector $\boldsymbol{a}$ we let $\text{vec}^{-1}(\boldsymbol{a})$ denote the $d \times d$ matrix formed from listing the entries of $\boldsymbol{a}$ in a column-wise fashion in order from left to right. Note that vec^{-1} is the usual function inverse when the domain of vec is restricted to square matrices. In particular, $\text{vec}^{-1}\{\text{vec}(\boldsymbol{A})\} = \boldsymbol{A}$ for $d \times d$ matrices $\boldsymbol{A}$ and $\text{vec}\{\text{vec}^{-1}(\boldsymbol{a})\} = \boldsymbol{a}$ for $d^2 \times 1$ vectors $\boldsymbol{a}$. If s is a scalar function s and $\boldsymbol{a}$ is a vector then $s(\boldsymbol{a})$ denotes the vector formed by applying s to each entry of $\boldsymbol{a}$. The elementwise product of two equal-sized vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ is denoted by $\boldsymbol{a} \odot \boldsymbol{b}$.

2.2 Fundamental Normal Density Function Theorem

We now present a theorem that is fundamental to expectation propagation in linear model settings. It involves the integration of the Multivariate Normal density function against a particular Dirac delta function form. Let

$$p_{N_d}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \equiv (2\pi)^{-d/2}|\boldsymbol{\Sigma}|^{-1/2} \exp\{-\tfrac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\}$$

denote the density function of a d-variate $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ random vector. Then we have:

Theorem 1.: *For all $d \times 1$ vectors $\boldsymbol{\ell} \neq \mathbf{0}$ and $v \in \mathbb{R}$:*

$$\int_{\mathbb{R}^d} p_{N_d}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})\, \delta\big(v - \boldsymbol{\ell}^T \boldsymbol{x}\big)\, d\,\boldsymbol{x} = p_{N_1}(v; \boldsymbol{\ell}^T \boldsymbol{\mu}, \boldsymbol{\ell}^T \boldsymbol{\Sigma} \boldsymbol{\ell}). \tag{3}$$

A proof of Theorem 1 is given in the Appendix.

2.3 Factor Graphs

A *factor graph* (Frey *et al. 1998*) is a graphical representation of the factor/argument dependencies of a real-valued function. For example, the following function $h : \mathbb{R}^5 \to \mathbb{R}$:

$$h(x_1, x_2, x_3, x_4, x_5) \equiv \sin(x_1 + x_2\, x_5) \sqrt{x_2 + \cos(x_3^2 x_4^7)}\, \tanh\{\cos(x_3^2 x_4^7) x_5\}\, \log(|x_5| + 6) \tag{4}$$

is such that the first factor, $\sin(x_1 + x_2\, x_5)$, depends on the arguments x_1, x_2 and x_5. This dependence is conveyed in the graph shown in Figure 1 by having an edge between the white circular nodes for the three arguments and the black rectangular node for the factor. Similar dependencies are depicted by the edges between other circles and rectangles in Figure 1. If two nodes on a factor graph have an edge joining them then we say that the nodes are *neighbours* of each other.

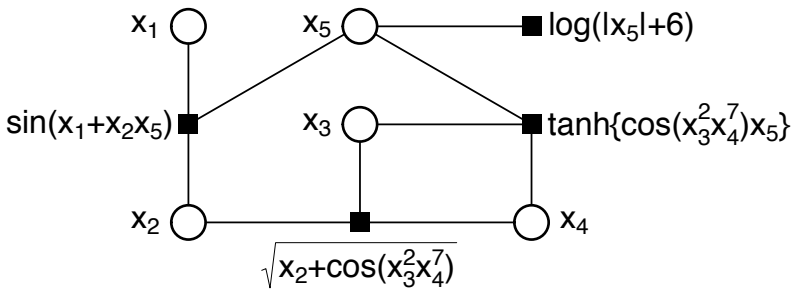

Figure 1: *A factor graph corresponding to the function defined by (4).*

Note that factor graphs of functions are not unique. For example, if the first factor in (4) is divided by x_3 and the second factor is multiplied by x_3 then a different factor graph arises. The factor graph in Figure 1 corresponds to the fully simplified version of h.

2.3.1 Derived Variable Adjustment

Note that, courtesy of (2), the function h in (4) may be rewritten as follows:

$$\begin{aligned} &h(x_1, x_2, x_3, x_4, x_5) = \\ &\qquad \sin(x_1 + x_2\, x_5) \left\{ \int_{-\infty}^{\infty} \sqrt{x_2 + v}\, \tanh(v\, x_5)\, \delta\big(v - \cos(x_3^2 x_4^7)\big)\, d\, v \right\} \log(|x_5| + 6). \end{aligned} \tag{5}$$

In keeping with the nomenclature of Minka & Winn (2008), we call $v \equiv \cos(x_3^2 x_4^7)$ a *derived variable*. The diagram in Figure 2 is a graphical representation of the factor/argument dependencies of the factors appearing on the right-hand side of (5). The solid line edges correspond to factors that are outside of the integral over the derived variable v, whereas those inside the integral have their dependencies depicted as dashed lines. Note that, since h depends on x_3 and x_4 only through v, we are grouping these two variables together in one of the circular nodes.

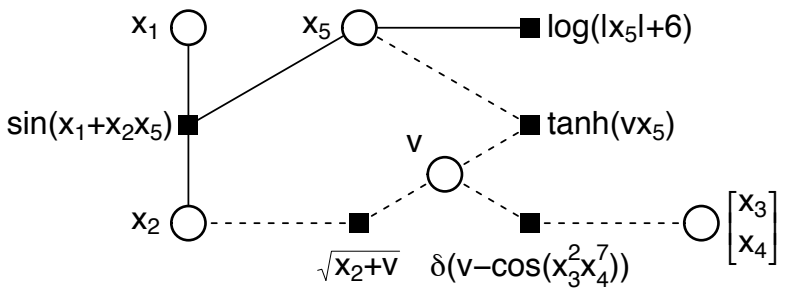

Figure 2: *A derived variable factor graph representation of (5). The dashed line edges depict factor/argument dependencies for the factors inside the integral over the derived variable v.*

The graph shown in Figure 2 is not a factor graph in the usual sense because of the presence of the integral. We will call it a *derived variable factor graph*. As we will explain in Section 2.4, derived variable factor graphs are central to expectation propagation in linear model settings.

2.4 Approximate Bayesian Inference via Message Passing on Factor Graphs

In this section we are mainly concerned with a very general description of expectation propagation, but we will also discuss its 'rival', mean field variational Bayes. The representation of mean field variational Bayes in terms of message passing on factor graphs is known as *variational message passing*. Both types of approaches are driven by the goal of minimisation of the *Kullback-Leibler divergence* between the exact posterior joint density function of the model parameters and its approximation. For arbitrary density functions p_1 and p_2 on $\mathbb{R}^d$, the Kullback-Leibler divergence of p_2 from p_1 is given by

$$\mathrm{KL}(p_1 \,\|\, p_2) \equiv \int_{\mathbb{R}^d} p_1(\boldsymbol{x}) \, \log\left\{p_1(\boldsymbol{x})/p_2(\boldsymbol{x})\right\} d\boldsymbol{x}.$$

For the remainder of this subsection, we mainly follow the descriptions of expectation propagation already given in Section 3 of Kim & Wand (2016) and variational message passing given in Section 2.5 of Wand (2017). Hence, we will be brief. A point of difference, however, is the extension to derived variable factor graphs.

Of concern is approximate Bayesian inference for a parameter vector $\boldsymbol{\theta}$ based on observed data $\boldsymbol{D}$. Given a partition, $\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M\}$, of $\boldsymbol{\theta}$ we consider approximations to the posterior density function of interest that have general form

$$p(\boldsymbol{\theta}|\boldsymbol{D}) \approx \prod_{i=1}^{M} q(\boldsymbol{\theta}_i).$$

Mean field variational Bayes/variational message passing is driven by the goal of minimising

$$\mathrm{KL}\Big(\prod_{i=1}^{M} q(\boldsymbol{\theta}_i) \,\Big\|\, p(\boldsymbol{\theta}|\boldsymbol{D})\Big)$$

whereas expectation propagation is driven by minimisation of the reversed Kullback-Leibler divergence:

$$\mathrm{KL}\Big(p(\boldsymbol{\theta}|\boldsymbol{D}) \,\Big\|\, \prod_{i=1}^{M} q(\boldsymbol{\theta}_i)\Big). \tag{6}$$

As explained in (3.3)–(3.4) of Kim & Wand (2016), the joint density function of the parameter vector and observed data, $p(\boldsymbol{\theta}, \boldsymbol{D})$ can be written as a product of N functions, f_j,

$1 \leq j \leq N$. Since each f_j is a function of a subset of $\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M\}$, we can represent the factor/argument dependencies in $p(\boldsymbol{\theta}, \boldsymbol{D})$ as a factor graph. Example generic factor graphs are given in Section 3 of Kim & Wand (2016) and Section 2.5 of Wand (2017). Factor graphs specific to generalized, linear and mixed models are given in Sections 3–5 of this article. The nodes in the factor graph corresponding to the $\boldsymbol{\theta}_i$ are usually called *stochastic* nodes.

Minka (2005) devised iterative schemes for solving the above minimum Kullback-Leibler divergence problems in terms of *messages* passed between neighbouring nodes on the relevant factor graph. A message is a function of the stochastic node that sends or receives the message. See equations (7)–(9) of Wand (2017) for the messages updates used by mean field variational Bayes/variational message passing.

The expectation propagation alternative is more delicate. As discussed in Minka (2005) and Section 3 of Kim & Wand (2016), the notion of *projection onto exponential family density functions* is required to make minimisation problem (6) viable. The most common such projection is onto the family of Univariate Normal density functions. However, here we will define the concept for the Multivariate Normal case. Suppose that p is a d-variate density function. Then the Kullback-Leibler projection of p onto the d-variate Normal family, which we denote by $\text{proj}_N[p]$, is the $N(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$ density function where

$$\boldsymbol{\mu}^* = \int_{\mathbb{R}^d} \boldsymbol{x}\, p(\boldsymbol{x})\, d\boldsymbol{x} \quad \text{and} \quad \boldsymbol{\Sigma}^* = \int_{\mathbb{R}^d} (\boldsymbol{x} - \boldsymbol{\mu}^*)(\boldsymbol{x} - \boldsymbol{\mu}^*)^T\, p(\boldsymbol{x})\, d\boldsymbol{x}.$$

In other words, $\text{proj}_N[p]$ is chosen to be the d-variate Normal density function with the same mean vector and covariance matrix as p. For the models treated in this article, the other main type of projection is that onto the Inverse Gamma family of density functions. We denote this type of projection by $\text{proj}_{\text{IG}}[p]$, where p is a univariate density function supported on the positive half-line, and refer the reader to Result 2 of Kim & Wand (2016) for details.

We now present the message passing updates for expectation propagation. Based on (45) of Minka & Winn (2008), the stochastic node to factor messages are updated according to

$$m_{\boldsymbol{\theta}_i \to f_j}(\boldsymbol{\theta}_i) \longleftarrow \prod_{j' \neq j:\, i \in \text{neighbours}(j')} m_{f_{j'} \to \boldsymbol{\theta}_i}(\boldsymbol{\theta}_i) \tag{7}$$

and the factor to stochastic node messages updates are

$$m_{f_j \to \boldsymbol{\theta}_i}(\boldsymbol{\theta}_i) \longleftarrow \frac{\text{proj}\left[m_{\boldsymbol{\theta}_i \to f_j}(\boldsymbol{\theta}_i) \int f_j(\boldsymbol{\theta}_{\text{neighbours}(j)}) \prod_{i' \in \text{neighbours}(j) \setminus \{i\}} m_{\boldsymbol{\theta}_{i'} \to f_j}(\boldsymbol{\theta}_{i'})\, d\boldsymbol{\theta}_{\text{neighbours}(j) \setminus \{i\}} / Z \right]}{m_{\boldsymbol{\theta}_i \to f_j}(\boldsymbol{\theta}_i)}, \tag{8}$$

where Z is the normalizing factor that ensures that the function of $\boldsymbol{\theta}_i$ inside the $\text{proj}[\cdot]$ is a density function. The normalizing factor in (8) involves summation if some of the $\boldsymbol{\theta}_{i'}$ have discrete components. The $\text{proj}[\cdot]$ in (8) denotes Kullback-Leibler projection onto an appropriate exponential family of density functions. However, in Kim & Wand (2016) illustration was done only via a simple example in which all of the stochastic nodes were univariate. In the case of linear models, in which vector parameters are present, some adjustments are necessary to avoid intractable multivariate integrals. The first one is an intrinsically important convention and is now spelt out:

Convention 1. *Derived variable factor graphs are treated as ordinary factor graphs when it comes to applying the message passing expressions (7) and (8).*

In expectation propagation for linear models, derived variable factor graphs are commonplace and Convention 1 needs to be adopted for the notion of message passing on a factor

graph to make sense. We are not aware of this subtlety being pointed out previously. Concrete illustrations will be given in the upcoming sections. Also, we are not aware of any rigorous theoretical underpinning for use of Convention 1 in combination with (8) and the ramifications for the Kullback-Leibler divergence minimisation problem that drives expectation propagation. Nevertheless, this is the factor graph version of the approach described in Section 13.8 of Gelman *et al.* (2014).

The other adjustment, recommended in Minka (2005), is referred to there as *damping*. The damping adjustment to (8) of amount $\varepsilon \in [0, 1)$ is

$$m_{f_j \to \boldsymbol{\theta}_i}(\boldsymbol{\theta}_i) \longleftarrow m_{f_j \to \boldsymbol{\theta}_i}(\boldsymbol{\theta}_i)^{\varepsilon} \times \{\text{right-hand side of (8)}\}^{1-\varepsilon}. \tag{9}$$

Without damping, corresponding to $\varepsilon = 0$ in (9), we sometimes found that the expectation propagation iterations did not converge. In our numerical investigations (Section 6), setting ε to a small positive value such as 0.1 overcame this problem.

An optional add-on to expectation propagation and mean field variational Bayes is calculation of an appropriate approximate marginal log-likelihood. Equations (3.9)–(3.11) of Kim & Wand (2016) give the required formulae for expectation propagation via the factor graph-based approach described here.

3 Linear Models

Consider the linear model

$$\boldsymbol{y} \,|\, \boldsymbol{\beta}, \sigma^2 \sim N(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2 \boldsymbol{I})$$

with prior distributions

$$\boldsymbol{\beta} \sim N(\boldsymbol{0}, \sigma^2_{\boldsymbol{\beta}} \boldsymbol{I}) \quad \text{and} \quad \sigma \sim \text{Half-Cauchy}(A),$$

where the second of these statements corresponds to σ having a prior density function $p(\sigma) = 2/[A\pi\{1 + (\sigma/A)^2\}]$ for $\sigma > 0$. An equivalent representation of the model is

$$\begin{gathered} \boldsymbol{y} \,|\, \boldsymbol{\beta}, \sigma^2 \sim N(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2 \boldsymbol{I}), \quad \boldsymbol{\beta} \sim N(\boldsymbol{0}, \sigma^2_{\boldsymbol{\beta}} \boldsymbol{I}), \\ \sigma^2 |\, a \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/a), \quad a \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/A^2). \end{gathered} \tag{10}$$

We use this version from now onwards since it is more amenable to expectation propagation and mean field variational Bayes approximate inference algorithms. Exact Bayesian inference for the coefficient vector $\boldsymbol{\beta}$ involves the posterior density function

$$p(\boldsymbol{\beta}|\boldsymbol{y}) = \frac{p(\boldsymbol{\beta}) \int_0^\infty \left\{ \int_0^\infty p(\boldsymbol{y}|\boldsymbol{\beta}, \sigma^2) p(\sigma^2 |\, a)\, d\sigma^2 \right\} p(a) d\, a}{\int_{\mathbb{R}^d} p(\boldsymbol{\beta}') \int_0^\infty \left\{ \int_0^\infty p(\boldsymbol{y}|\boldsymbol{\beta}', \sigma^2) p(\sigma^2 |\, a)\, d\sigma^2 \right\} p(a) d\, a\, d\boldsymbol{\beta}'}. \tag{11}$$

Further calculations reveal that numerical integration over $\mathbb{R}^d$ is needed to obtain Bayesian point estimates and credible sets for the entries of $\boldsymbol{\beta}$. Similar problems arise when endeavouring to make exact Bayesian inference for σ^2.

Mean field variational Bayes approximate inference for (10) involves approximation of the joint posterior density function of the model parameters by

$$p(\boldsymbol{\beta}, \sigma^2, a|\boldsymbol{y}) \approx q(\boldsymbol{\beta}, \sigma^2, a)$$

where $q(\boldsymbol{\beta}, \sigma^2, a)$ has a product density form such as

$$q(\boldsymbol{\beta}, \sigma^2, a) = q(\boldsymbol{\beta})\, q(\sigma^2)\, q(a). \tag{12}$$

Minimisation of $\text{KL}\Big(q(\boldsymbol{\beta})\, q(\sigma^2)\, q(a) \Big\| p(\boldsymbol{\beta}, \boldsymbol{u}, \sigma^2|\boldsymbol{y})\Big)$ leads to Algorithm 1 of Luts *et al.* (2014).

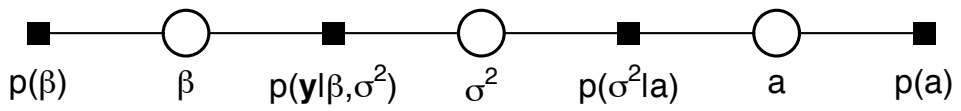

Figure 3: *Factor graph corresponding to the model (10) and q-density product restriction (12).*

An alternative approach, which leads to the same approximation produced by Algorithm 1 of Luts *et al.* (2014), is based on variational message passing (Winn & Bishop, 2005). Section 3.1 of Wand (2017) explains variational message passing steps for (10) according to product restriction (12). The starting point is the factor graph corresponding to the following factorisation of the joint density function of all random variables in the model

$$p(\boldsymbol{y}, \boldsymbol{\beta}, \sigma^2, a) = p(\boldsymbol{\beta})\, p(\boldsymbol{y}|\boldsymbol{\beta}, \sigma^2)\, p(\sigma^2|\, a)\, p(a)$$

and is shown in Figure 3. As demonstrated in Section 3.1 of Wand (2017), each of the messages and their parameter updates admit simple closed forms. For example, the message from $p(\boldsymbol{y}|\boldsymbol{\beta}, \sigma^2)$ to $\boldsymbol{\beta}$ is proportional to a Multivariate Normal density function in $\boldsymbol{\beta}$.

If expectation propagation is applied to the same factor graph (i.e. Figure 3) then (8) leads to

$$m_{p(\boldsymbol{y}|\boldsymbol{\beta},\,\sigma^2)\,\to\,\boldsymbol{\beta}}(\boldsymbol{\beta}) \longleftarrow \text{proj}_N\big[h(\boldsymbol{\beta})/Z\big] \exp\left\{-\begin{bmatrix} \boldsymbol{\beta} \\ \text{vec}(\boldsymbol{\beta}\boldsymbol{\beta}^T) \end{bmatrix}^T \boldsymbol{\eta}_{\boldsymbol{\beta}\,\to\,p(\boldsymbol{y}|\boldsymbol{\beta},\,\sigma^2)}\right\},$$

where

$$h(\boldsymbol{\beta}) \equiv \exp\left\{\begin{bmatrix} \boldsymbol{\beta} \\ \text{vec}(\boldsymbol{\beta}\boldsymbol{\beta}^T) \end{bmatrix}^T \boldsymbol{\eta}_{\boldsymbol{\beta}\,\to\,p(\boldsymbol{y}|\boldsymbol{\beta},\,\sigma^2)}\right\} \left(\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} - 2\boldsymbol{y}^T\boldsymbol{X}\boldsymbol{\beta} + \|\boldsymbol{y}\|^2 - 2\eta_2^{\#}\right)^{-\frac{n}{2}+\eta_1^{\#}+1},$$

$\boldsymbol{\eta}^{\#} \equiv \boldsymbol{\eta}_{\sigma^2\,\to\,p(\boldsymbol{y}|\boldsymbol{\beta},\,\sigma^2)}$ and Z is the normalizing factor. However, $\text{proj}_N[h(\boldsymbol{\beta})/Z]$ requires computation of

$$\int_{\mathbb{R}^d} h(\boldsymbol{\beta})\, d\boldsymbol{\beta}, \quad \int_{\mathbb{R}^d} \boldsymbol{\beta}\, h(\boldsymbol{\beta})\, d\boldsymbol{\beta} \quad \text{and} \quad \int_{\mathbb{R}^d} \boldsymbol{\beta}\boldsymbol{\beta}^T\, h(\boldsymbol{\beta})\, d\boldsymbol{\beta}. \tag{13}$$

which are each analytically intractable integrals over $\mathbb{R}^d$. For moderate values of d quadrature may be feasible. However, the same applies to exact Bayesian inference via (11), and therefore there is no apparent advantage of expectation propagation for the Figure 3 factor graph. Instead, the commonly-used version of expectation propagation works with a different representation of $p(\boldsymbol{y}, \boldsymbol{\beta}, \sigma^2, a)$ and an appropriate derived variable factor graph. Let

$$\boldsymbol{x}_i, 1 \le i \le n, \text{be defined by } \boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \vdots \\ \boldsymbol{x}_n^T \end{bmatrix}$$

and then note that

$$p(\boldsymbol{y}, \boldsymbol{\beta}, \sigma^2, a) = p(\boldsymbol{\beta}) \left[\prod_{i=1}^{n} \int_{-\infty}^{\infty} \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})(2\pi\sigma^2)^{-1/2} \exp\left\{-\frac{1}{2\sigma^2}(y_i - \alpha_i)^2\right\} d\alpha_i\right] \times p(\sigma^2|\, a)\, p(a) \tag{14}$$

which involves the derived variables α_i corresponding to $\boldsymbol{x}_i^T\boldsymbol{\beta}$, $1 \le i \le n$. The derived variable factor graph that matches (14) is shown in Figure 4. Note that the product restriction (12) is still apparent from the Figure 4 factor graph.

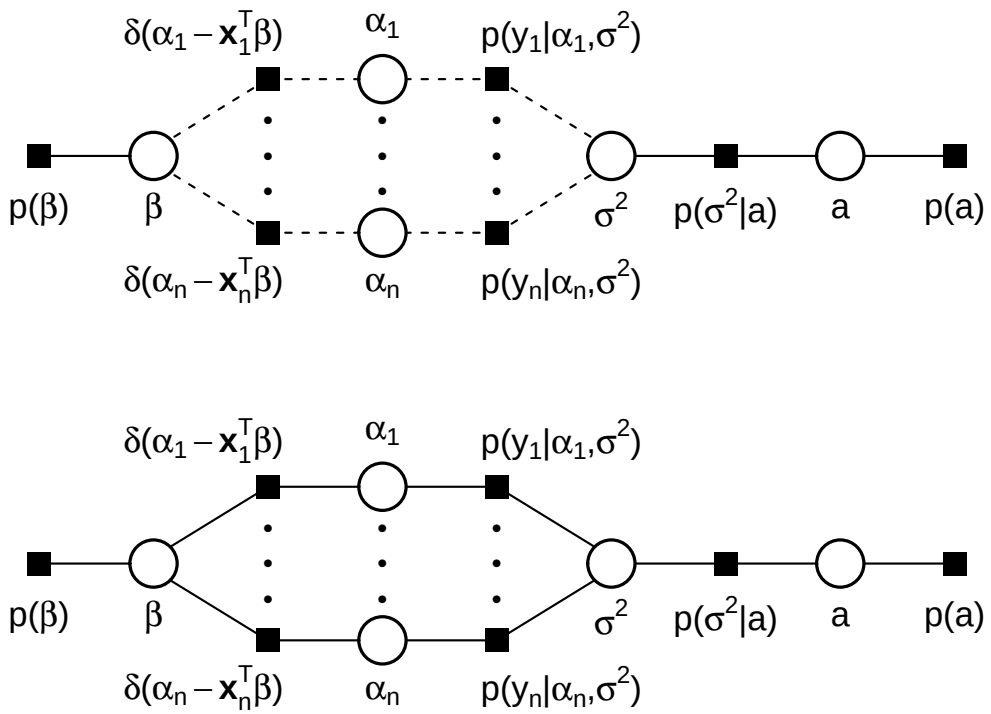

Figure 4: *Upper panel: Derived variable factor graph corresponding to the representation of the joint density function of the model random variables given by (14) in which derived variables* α_i, $1 \leq i \leq n$, *have been introduced. Lower panel: The factor graph for application of the expectation propagation message updates (7) and (8) after adoption of Convention 1. Under this convention, the dashed edges that join factors and stochastic variables inside the integrals over the* α_i *are treated as ordinary edges.*

The requisite algebra for the messages passed on the Figure 4 factor graph mimics that given in Kim & Wand (2016) which deals, essentially, with the $d = 1$ special case. Also, the updates for the messages sent from $p(\sigma^2|a)$ and $p(a)$ to their neighbours are exactly the same as those given in Kim & Wand (2016), so those calculations do not need to be repeated here.

The main new message types are those passed from the $\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})$ factors. From (2) and (8),

$$m_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \boldsymbol{\beta}}(\boldsymbol{\beta}) = m_{\alpha_i \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}(\boldsymbol{x}_i^T\boldsymbol{\beta}).$$

Standard matrix manipulations lead to

$$m_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \boldsymbol{\beta}}(\boldsymbol{\beta}) = \exp\left\{ \begin{bmatrix} \boldsymbol{\beta} \\ \text{vec}(\boldsymbol{\beta}\boldsymbol{\beta}^T) \end{bmatrix}^T \boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \boldsymbol{\beta}} \right\},$$

where the message's natural parameter update is

$$\boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \boldsymbol{\beta}} \longleftarrow \begin{bmatrix} \boldsymbol{x}_i \left(\boldsymbol{\eta}_{\alpha_i \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\right)_1 \\ \text{vec}(\boldsymbol{x}_i\boldsymbol{x}_i^T) \left(\boldsymbol{\eta}_{\alpha_i \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\right)_2 \end{bmatrix}. \tag{15}$$

Here, for $1 \leq i \leq n$, $1 \leq j \leq 2$, $\left(\boldsymbol{\eta}_{\alpha_i \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\right)_j$ are the entries of the natural parameters for the messages passed from the α_i to the $\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})$.

Application of (8) again leads to the messages passed from the $\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})$ to the α_i equalling

$$m_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \alpha_i}(\alpha_i) = \int_{\mathbb{R}^d} \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) m_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}(\boldsymbol{\beta})\, d\boldsymbol{\beta}.$$

Using Theorem 1 and the mappings between Multivariate Normal natural and common parameters given in, for example, equation (S.4) of Wand (2017), we obtain

$$m_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \alpha_i}(\alpha_i) \longleftarrow \exp\left\{ \begin{bmatrix} \alpha_i \\ \alpha_i^2 \end{bmatrix}^T \boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \alpha_i} \right\},$$

where

$$\boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \alpha_i} \longleftarrow \begin{bmatrix} \boldsymbol{x}_i^T \boldsymbol{\mu}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \Big/ \boldsymbol{x}_i^T \boldsymbol{\Sigma}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \boldsymbol{x}_i \\ -\frac{1}{2} \Big/ \boldsymbol{x}_i^T \boldsymbol{\Sigma}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \boldsymbol{x}_i \end{bmatrix}$$

with

$$\boldsymbol{\Sigma}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \longleftarrow -\tfrac{1}{2} \left\{ \text{vec}^{-1}\Big(\big(\boldsymbol{\eta}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\big)_2 \Big) \right\}^{-1}$$

and

$$\boldsymbol{\mu}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \longleftarrow \boldsymbol{\Sigma}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \big(\boldsymbol{\eta}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\big)_1,$$

where for $1 \le i \le n$, $\Big(\boldsymbol{\eta}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\Big)_1$, contains the first d^β entries of the natural parameter vector for the messages passed from $\boldsymbol{\beta}$ to the $\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})$ and $\Big(\boldsymbol{\eta}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\Big)_2$ contains the remaining $(d^\beta)^2$ entries.

After all of the natural parameter updates are distilled then Algorithm 1 emerges. The following notation:

$$a \overset{\varepsilon}{\longleftarrow} b \quad \text{defined to be} \quad a \longleftarrow \varepsilon\, a + (1-\varepsilon) b \quad \text{for any } 0 \le \varepsilon \le 1$$

is used to convey damping adjustments of the type given by (9). Note that the natural parameter updates correspond to message updates on the logarithmic scale.

4 Generalized Linear Model Extensions

The handling of generalized linear extensions via expectation propagation involves working with modifications of the Figure 4 factor graph in the neighbourhood of the likelihood factor. The messages away from the likelihood factor are identical to those in the Gaussian response case but take different forms around that factor.

For simplicity we will work with canonical one-parameter exponential family generalized linear models which have general form

$$p(\boldsymbol{y}|\boldsymbol{\beta}) = \exp\{\boldsymbol{y}^T\boldsymbol{X}\boldsymbol{\beta} - \mathbf{1}^T b(\boldsymbol{X}\boldsymbol{\beta}) + \mathbf{1}^T c(\boldsymbol{y})\}, \quad \boldsymbol{\beta} \sim N(\boldsymbol{\mu}_{\boldsymbol{\beta}}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}}). \tag{16}$$

The two most prominent cases are logistic regression for which $b(x) = \log(1 + e^x)$ and $c(x) = 0$ and Poisson regression where $b(x) = e^x$ and $c(x) = -\log(x!)$. The factor graph appropriate for such generalized linear models is shown in Figure 5 and corresponds to the following representation of the joint distribution of the model random variables:

$$p(\boldsymbol{y}, \boldsymbol{\beta}) = p(\boldsymbol{\beta}) \left[\prod_{i=1}^n \int_{-\infty}^{\infty} \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \exp\left\{ y_i\alpha_i - b(\alpha_i) + c(y_i) \right\} d\alpha_i \right]. \tag{17}$$

Inputs: $\boldsymbol{X}$ $(n \times d)$, $\boldsymbol{y}$ $(n \times 1)$; $\boldsymbol{\mu}_{\boldsymbol{\beta}}$ $(d \times 1)$, $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}$ $(d \times d$, symmetric and positive definite), $A > 0$, $0 \leq \varepsilon < 1$.

Initialise:

For $i = 1, \ldots, n$: assign $\boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \boldsymbol{\beta}}$, $\boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \alpha_i}$,
$\boldsymbol{\eta}_{p(y_i|\alpha_i, \sigma^2) \to \alpha_i}$ and $\boldsymbol{\eta}_{p(y_i|\alpha_i, \sigma^2) \to \sigma^2}$ to values within appropriate parameter space.

$$\boldsymbol{\eta}_{p(\boldsymbol{\beta}) \to \boldsymbol{\beta}} \longleftarrow \begin{bmatrix} \boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{-1}\boldsymbol{\mu}_{\boldsymbol{\beta}} \\ -\frac{1}{2}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{-1} \end{bmatrix} \; ; \; \boldsymbol{\eta}_{p(a) \to a} \longleftarrow \begin{bmatrix} -3/2 \\ -1/A^2 \end{bmatrix}$$

Cycle:

$$\text{SUM}\left\{\boldsymbol{\eta}_{\delta(\boldsymbol{\alpha} - \boldsymbol{x}^T\boldsymbol{\beta}) \to \boldsymbol{\beta}}\right\} \longleftarrow \sum_{i=1}^{n} \boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \boldsymbol{\beta}}$$

$$\text{SUM}\left\{\boldsymbol{\eta}_{p(\boldsymbol{y}|\boldsymbol{\alpha}, \sigma^2) \to \sigma^2}\right\} \longleftarrow \sum_{i=1}^{n} \boldsymbol{\eta}_{p(y_i|\alpha_i, \sigma^2) \to \sigma^2}$$

for $i = 1, \ldots, n$:

$$\boldsymbol{\eta}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \longleftarrow \boldsymbol{\eta}_{p(\boldsymbol{\beta}) \to \boldsymbol{\beta}} + \text{SUM}\left\{\boldsymbol{\eta}_{\delta(\boldsymbol{\alpha} - \boldsymbol{x}^T\boldsymbol{\beta}) \to \boldsymbol{\beta}}\right\} - \boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \boldsymbol{\beta}}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \longleftarrow -\tfrac{1}{2}\left\{\text{vec}^{-1}\left(\left(\boldsymbol{\eta}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\right)_2\right)\right\}^{-1}$$

$$\boldsymbol{\mu}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \longleftarrow \boldsymbol{\Sigma}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\left(\boldsymbol{\eta}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\right)_1$$

$$\boldsymbol{\eta}_{\alpha_i \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \longleftarrow \boldsymbol{\eta}_{p(y_i|\alpha_i, \sigma^2) \to \alpha_i} \; ; \boldsymbol{\eta}_{\alpha_i \to p(y_i|\alpha_i, \sigma^2)} \longleftarrow \boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \alpha_i}$$

$$\boldsymbol{\eta}_{\sigma^2 \to p(y_i|\alpha_i, \sigma^2)} \longleftarrow \boldsymbol{\eta}_{p(\sigma^2|\, a) \to \sigma^2} + \text{SUM}\left\{\boldsymbol{\eta}_{p(\boldsymbol{y}|\boldsymbol{\alpha}, \sigma^2) \to \sigma^2}\right\} - \boldsymbol{\eta}_{p(y_i|\alpha_i, \sigma^2) \to \sigma^2}$$

$$\boldsymbol{\eta}_{\sigma^2 \to p(\sigma^2|\, a)} \longleftarrow \text{SUM}\left\{\boldsymbol{\eta}_{p(\boldsymbol{y}|\boldsymbol{\alpha}, \sigma^2) \to \sigma^2}\right\} \; ; \boldsymbol{\eta}_{a \to p(\sigma^2|\, a)} \longleftarrow \boldsymbol{\eta}_{p(a) \to a}$$

for $i = 1, \ldots, n$:

$$\boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \boldsymbol{\beta}} \overset{\varepsilon}{\longleftarrow} \begin{bmatrix} \boldsymbol{x}_i\left(\boldsymbol{\eta}_{\alpha_i \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\right)_1 \\ \text{vec}(\boldsymbol{x}_i\boldsymbol{x}_i^T)\left(\boldsymbol{\eta}_{\alpha_i \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\right)_2 \end{bmatrix}$$

$$\boldsymbol{\eta}_{\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \to \alpha_i} \overset{\varepsilon}{\longleftarrow} \begin{bmatrix} \boldsymbol{x}_i^T\boldsymbol{\mu}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})} \Big/ \boldsymbol{x}_i^T\boldsymbol{\Sigma}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\boldsymbol{x}_i \\ -\frac{1}{2}\Big/\boldsymbol{x}_i^T\boldsymbol{\Sigma}_{\boldsymbol{\beta} \to \delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\boldsymbol{x}_i \end{bmatrix}$$

$$\boldsymbol{\eta}_{p(y_i|\alpha_i, \sigma^2) \to \alpha_i} \overset{\varepsilon}{\longleftarrow} G^N\left(\boldsymbol{\eta}_{\alpha_i \to p(y_i|\alpha_i, \sigma^2)}, \boldsymbol{\eta}_{\sigma^2 \to p(y_i|\alpha_i, \sigma^2)}; \begin{bmatrix} 1 \\ y_i \\ y_i^2 \end{bmatrix}\right)$$

$$\boldsymbol{\eta}_{p(y_i|\alpha_i, \sigma^2) \to \sigma^2} \overset{\varepsilon}{\longleftarrow} G^{\text{IG1}}\left(\boldsymbol{\eta}_{\sigma^2 \to p(y_i|\alpha_i, \sigma^2)}, \boldsymbol{\eta}_{\alpha_i \to p(y_i|\alpha_i, \sigma^2)}; \begin{bmatrix} 1 \\ y_i \\ y_i^2 \end{bmatrix}\right)$$

$$\boldsymbol{\eta}_{p(\sigma^2|\, a) \to \sigma^2} \longleftarrow G^{\text{IG2}}\left(\boldsymbol{\eta}_{\sigma^2 \to p(\sigma^2|\, a)}, \boldsymbol{\eta}_{a \to p(\sigma^2|\, a)}; 3\right)$$

$$\boldsymbol{\eta}_{p(\sigma^2|\, a) \to a} \longleftarrow G^{\text{IG2}}\left(\boldsymbol{\eta}_{a \to p(\sigma^2|\, a)}, \boldsymbol{\eta}_{\sigma^2 \to p(\sigma^2|\, a)}; 1\right)$$

until the change in all natural parameter vectors is negligible.

$$\boldsymbol{\eta}_{q(\boldsymbol{\beta})} \longleftarrow \boldsymbol{\eta}_{p(\boldsymbol{\beta}) \to \boldsymbol{\beta}} + \text{SUM}\left\{\boldsymbol{\eta}_{\delta(\boldsymbol{\alpha} - \boldsymbol{x}^T\boldsymbol{\beta}) \to \boldsymbol{\beta}}\right\}$$

$$\boldsymbol{\eta}_{q(\sigma^2)} \longleftarrow \text{SUM}\left\{\boldsymbol{\eta}_{p(\boldsymbol{y}|\boldsymbol{\alpha}, \sigma^2) \to \sigma^2}\right\} + \boldsymbol{\eta}_{p(\sigma^2|\, a) \to \sigma^2} \; ; \; \boldsymbol{\eta}_{q(a)} \longleftarrow \boldsymbol{\eta}_{p(\sigma^2|\, a) \to a} + \boldsymbol{\eta}_{p(a) \to a}$$

Algorithm 1: *Expectation propagation algorithm for determining the natural parameter vectors* $\boldsymbol{\eta}_{q(\boldsymbol{\beta})}$, $\boldsymbol{\eta}_{q(\sigma^2)}$ *and* $\boldsymbol{\eta}_{q(a)}$ *of the optimal density functions* $q^*(\boldsymbol{\beta})$, $q^*(\sigma^2)$ *and* $q^*(a)$ *for approximate Bayesian inference in the Gaussian response linear model (10). The natural parameter updates correspond to the messages passed on the Figure 4 factor graph.*

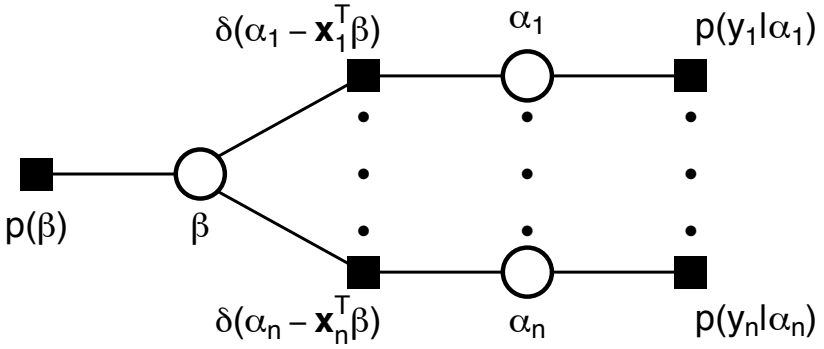

Figure 5: *Derived variable factor graph corresponding to the generalized linear model joint density function (17). All edges have solid lines to indicate adoption of Convention 1.*

The messages passed from the factors $p(\boldsymbol{\beta})$ and $\delta(\alpha_i - \boldsymbol{x}_i^T\boldsymbol{\beta})$ to their neighbours are identical to those for the Gaussian response factor graph. The natural parameter updates given in Algorithm 1 also apply to the generalized linear model extension.

The new factor to stochastic node messages are

$$m_{p(y_i|\alpha_i) \to \alpha_i}(\alpha_i) = \frac{\text{proj}_N[m_{\alpha_i \to p(y_i|\alpha_i)}(\alpha_i)\, p(y_i|\,\alpha_i)/Z]}{m_{\alpha_i \to p(y_i|\alpha_i)}(\alpha_i)}, \quad 1 \le i \le n. \tag{18}$$

Introducing the shorthand $\boldsymbol{\eta}_i^{\oplus} \equiv \boldsymbol{\eta}_{\alpha_i \to p(y_i|\alpha_i)}$ and noting that

$$m_{\alpha_i \to p(y_i|\alpha_i)}(\alpha_i) \longleftarrow \exp(\eta_{i1}^{\oplus}\alpha_i + \eta_{i2}^{\oplus}\alpha_i^2),$$

(18) becomes

$$m_{p(y_i|\alpha_i) \to \alpha_i}(\alpha_i) = \frac{\text{proj}_N[\exp\{(y_i + \eta_{i1}^{\oplus})\alpha_i + \eta_{i2}^{\oplus}\alpha_i^2 - b(\alpha_i)\}/Z]}{\exp(\eta_{i1}^{\oplus}\alpha_i + \eta_{i2}^{\oplus}\alpha_i^2)}, \quad 1 \le i \le n. \tag{19}$$

Using Result 1 of Kim & Wand (2016), routine manipulations show that the numerator of the right-hand side of (19) equals the Normal density function with mean and variance

$$\mu^* = \frac{\mathcal{C}_b(1, y_i + \eta_{i1}^{\oplus}, -\eta_{i2}^{\oplus})}{\mathcal{C}_b(0, y_i + \eta_{i1}^{\oplus}, -\eta_{i2}^{\oplus})} \quad \text{and} \quad (\sigma^2)^* = \frac{\mathcal{C}_b(2, y_i + \eta_{i1}^{\oplus}, -\eta_{i2}^{\oplus})}{\mathcal{C}_b(0, y_i + \eta_{i1}^{\oplus}, -\eta_{i2}^{\oplus})} - (\mu^*)^2.$$

Further straightforward manipulations then lead to

$$m_{p(y_i|\alpha_i) \to \alpha_i}(\alpha_i) = \exp\left\{ \begin{bmatrix} \alpha_i \\ \alpha_i^2 \end{bmatrix}^T \boldsymbol{\eta}_{p(y_i|\alpha_i) \to \alpha_i} \right\},$$

where

$$\boldsymbol{\eta}_{p(y_i|\alpha_i) \to \alpha_i} \longleftarrow H_b\big(\boldsymbol{\eta}_{\alpha_i \to p(y_i|\alpha_i)}; y_i\big) \tag{20}$$

and the function H_b is given by

$$H_b\left(\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}; y\right) \equiv \begin{bmatrix} \dfrac{\mathcal{C}_b(1, a_1 + y, -a_2)/\mathcal{C}_b(0, a_1 + y, -a_2)}{\dfrac{\mathcal{C}_b(2, a_1 + y, -a_2)}{\mathcal{C}_b(0, a_1 + y, -a_2)} - \dfrac{\mathcal{C}_b(1, a_1 + y, -a_2)^2}{\mathcal{C}_b(0, a_1 + y, -a_2)^2}} \\ \\ \dfrac{-1/2}{\dfrac{\mathcal{C}_b(2, a_1 + y, -a_2)}{\mathcal{C}_b(0, a_1 + y, -a_2)} - \dfrac{\mathcal{C}_b(1, a_1 + y, -a_2)^2}{\mathcal{C}_b(0, a_1 + y, -a_2)^2}} \end{bmatrix} - \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$$

for any $a_1 \in \mathbb{R}$, $a_2 < 0$ and $y \in \mathbb{R}$.

Each of the other messages required for expectation propagation on the Figure 5 factor graph are already reflected in the natural parameter updates for the Gaussian response linear model listed in Algorithm 1. Therefore (20) can be combined with the relevant updates from Algorithm 1 to list each of the natural parameter updates to obtain the explicit expectation propagation algorithm for these generalized linear model extensions.

4.1 Probit Regression

The case of probit regression deserves special mention since the messages passed from the $p(y_i|\alpha_i)$s to the α_is admit closed form expressions, which obviates the need for numerical integration.

The probit regression alternative to (16) is

$$p(\boldsymbol{y}|\boldsymbol{\beta}) = \exp\Big[\mathbf{1}^T \log\big\{\Phi\big((2\boldsymbol{y}-\mathbf{1}) \odot (\boldsymbol{X}\boldsymbol{\beta})\big)\big\}\Big], \quad \boldsymbol{\beta} \sim N(\boldsymbol{\mu}_{\boldsymbol{\beta}}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}})$$

where the entries of $\boldsymbol{y}$ are either 0 or 1, and

$$\phi(x) \equiv (2\pi)^{-1/2} \exp(-\tfrac{1}{2}\, x^2) \quad \text{and} \quad \Phi(x) \equiv \int_{-\infty}^{x} \phi(t)\, dt$$

are, respectively, the density function and cumulative distribution function of the $N(0,1)$ distribution. For this model (18) reduces to

$$m_{p(y_i|\alpha_i) \,\to\, \alpha_i}(\alpha_i) = \frac{\text{proj}_N[\exp(\eta_{i1}^{\oplus}\alpha_i + \eta_{i2}^{\oplus}\alpha_i^2)\Phi((2y_i-1)\alpha_i)/Z]}{\exp(\eta_{i1}^{\oplus}\alpha_i + \eta_{i2}^{\oplus}\alpha_i^2)}, \quad 1 \le i \le n. \tag{21}$$

Projection of the function inside the $\text{proj}_N[\cdot]$ is aided by the following integration results for general $a, b \in \mathbb{R}$:

$$\begin{aligned}
\int_{-\infty}^{\infty} \Phi(a+bx)\phi(x)\, dx &= \Phi\left(\frac{a}{\sqrt{b^2+1}}\right), \\
\int_{-\infty}^{\infty} x\, \Phi(a+bx)\phi(x)\, dx &= \frac{b}{\sqrt{b^2+1}}\, \phi\left(\frac{a}{\sqrt{b^2+1}}\right) \\
\text{and} \quad \int_{-\infty}^{\infty} x^2\, \Phi(a+bx)\phi(x)\, dx &= \Phi\left(\frac{a}{\sqrt{b^2+1}}\right) - \frac{ab^2}{\sqrt{(b^2+1)^3}}\, \phi\left(\frac{a}{\sqrt{b^2+1}}\right).
\end{aligned} \tag{22}$$

Results (22) and a good deal of algebra can be used to show that the natural parameter update arising from (21) is

$$\boldsymbol{\eta}_{p(y_i|\alpha_i) \,\to\, \alpha_i} \longleftarrow H_{\text{probit}}\big(\boldsymbol{\eta}_{\alpha_i \,\to\, p(y_i|\alpha_i)}; y_i\big), \quad 1 \le i \le n,$$

where the function H_{probit} has the explicit form

$$H_{\text{probit}}\left(\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}; y\right) \equiv \frac{1}{1-2a_2-\zeta'(r)\big\{r+\zeta'(r)\big\}} \begin{bmatrix} a_1(1-2a_2) \\ +(2y-1)\zeta'(r)\sqrt{2a_2(2a_2-1)} \\ a_2(1-2a_2) \end{bmatrix} - \begin{bmatrix} a_1 \\ \\ a_2 \end{bmatrix}$$

with $r \equiv (2y-1)a_1/\sqrt{2a_2(2a_2-1)}$ and $\zeta(x) \equiv \log\{2\Phi(x)\}$ implying that $\zeta'(x) = (\phi/\Phi)(x)$. Stable computation of $\zeta'(r)$ requires some care and use of tailored software such as the function `zeta()` in the R package sn (Azzalini, 2016) is recommended.

5 Mixed Model Extensions

First consider the purely random effects model

$$\begin{aligned} \boldsymbol{y}|\,\boldsymbol{u},\sigma_\varepsilon^2 &\sim N(\boldsymbol{Z}\boldsymbol{u},\sigma_\varepsilon^2\,\boldsymbol{I}), \quad \boldsymbol{u}|\,\sigma_u^2 \sim N(\boldsymbol{0},\sigma_u^2\boldsymbol{I}), \\ \sigma_\varepsilon^2|\,a_\varepsilon &\sim \text{Inverse-Gamma}(\tfrac{1}{2},1/a_\varepsilon), \quad a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2},1/A_\varepsilon^2), \\ \sigma_u^2|\,a_u &\sim \text{Inverse-Gamma}(\tfrac{1}{2},1/a_u), \quad a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2},1/A_u^2) \end{aligned} \tag{23}$$

where $\boldsymbol{u}$ is a $d^u \times 1$ vector of random coefficients and where $\boldsymbol{Z}$ is an $N \times d^u$ design matrix. The most common instance of such a model is the random intercept model for grouped data where the first line of (23) corresponds to

$$y_{ij}|U_i,\sigma_\varepsilon^2 \overset{\text{ind.}}{\sim} N(U_i,\sigma_\varepsilon^2), \quad U_i|\,\sigma_u^2 \overset{\text{ind.}}{\sim} N(0,\sigma_u^2), \quad 1 \le i \le m,\ 1 \le j \le n_i,$$

with $\boldsymbol{y}$ containing the y_{ij} and $\boldsymbol{u}$ containing the U_i. In this special case, $d^u = m$ (the number of groups), $N = \sum_{i=1}^m n_i$ (the total number of response measurements) and $\boldsymbol{Z} = \text{blockdiag}_{1\le i\le m}\,\boldsymbol{1}_{n_i}$. Here $\boldsymbol{1}_d$ denotes the $d \times 1$ vector of ones. However, $\boldsymbol{Z}$ can assume various other forms as well, and is taken to be a general $N \times d^u$ matrix throughout this section. Let $\boldsymbol{z}_i^T$ denote the ith row of $\boldsymbol{Z}$, $1 \le i \le N$.

Note that the joint density function of all random variables in the model is

$$\begin{aligned} p(\boldsymbol{y},\boldsymbol{u},\sigma_\varepsilon^2,a_\varepsilon,\sigma_u^2,a_u) &= p(a_u)\,p(\sigma_u^2|a_u)\,p(\boldsymbol{u}|\,\sigma_u^2)\,p(\boldsymbol{y}|\,\boldsymbol{u},\sigma_\varepsilon^2)\,p(\sigma_\varepsilon^2|a_\varepsilon)\,p(a_\varepsilon) \\ &= p(a_u)\,p(\sigma_u^2|a_u)\,p(\boldsymbol{u}|\,\sigma_u^2)\,\left\{\prod_{i=1}^N \int_{-\infty}^{\infty} p(y_i|\alpha_i,\sigma_\varepsilon^2)\delta\left(\alpha_i - \boldsymbol{z}_i^T\boldsymbol{u}\right)\,d\alpha_i\right\} \\ &\quad \times p(\sigma_\varepsilon^2|\,a_\varepsilon)\,p(a_\varepsilon), \end{aligned}$$

where the α_i derived variables correspond to the $\boldsymbol{z}_i^T\boldsymbol{u}$. However, expectation propagation applied to the derived variable factor graph for this representation of $p(\boldsymbol{y},\boldsymbol{u},\sigma_\varepsilon^2,a_\varepsilon,\sigma_u^2,a_u)$ has the problem that the message passed from $p(\boldsymbol{u}|\sigma_u^2)$ to $\boldsymbol{u}$ involves d^u-dimensional intractable integrals with forms similar to those of (13). This tractability problem is overcome by working with

$$\begin{aligned} p(\boldsymbol{y},\boldsymbol{u},\sigma_\varepsilon^2,a_\varepsilon,\sigma_u^2,a_u) &= p(a_u)\,p(\sigma_u^2|\,a_u)\,\left\{\prod_{k=1}^{d^u}\int_{-\infty}^{\infty} p(\widetilde{u}_k|\,\sigma_u^2)\,\delta(\widetilde{u}_k - \boldsymbol{e}_k^T\boldsymbol{u})\,d\widetilde{u}_k\right\} \\ &\quad \times \left\{\prod_{i=1}^N \int_{-\infty}^{\infty} p(y_i|\,\alpha_i,\sigma_\varepsilon^2)\delta\left(\alpha_i - \boldsymbol{z}_i^T\boldsymbol{u}\right)\,d\alpha_i\right\}\,p(\sigma_\varepsilon^2|\,a_\varepsilon)\,p(a_\varepsilon), \end{aligned} \tag{24}$$

where $\boldsymbol{e}_k$ is the $d^u \times 1$ vector with 1 in the kth position and zeroes elsewhere. The corresponding derived variable factor graph is shown in Figure 6.

Each of the messages passed between the nodes on (23) take a similar form to those given in Kim & Wand (2016) or earlier sections of the current article. For example, algebraic steps similar to those given in Section A.5.3 of Kim & Wand (2016) lead to the natural parameter updates of Univariate Normal messages being

$$\boldsymbol{\eta}_{p(\widetilde{u}_k|\sigma_u^2)\,\to\,\widetilde{u}_k} \longleftarrow G^N\left(\boldsymbol{\eta}_{\widetilde{u}_k\,\to\,p(\widetilde{u}_k|\sigma_u^2)}, \boldsymbol{\eta}_{\sigma_u^2\,\to\,p(\widetilde{u}_k|\sigma_u^2)}; \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}\right), \quad 1 \le k \le K,$$

and a rejigging of those given Section A.5.5 gives

$$\boldsymbol{\eta}_{p(\widetilde{u}_k|\sigma_u^2)\,\to\,\sigma_u^2} \longleftarrow G^{\text{IG1}}\left(\boldsymbol{\eta}_{\sigma_u^2\,\to\,p(\widetilde{u}_k|\sigma_u^2)}, \boldsymbol{\eta}_{\widetilde{u}_k\,\to\,p(\widetilde{u}_k|\sigma_u^2)}; \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}\right), \quad 1 \le k \le K,$$

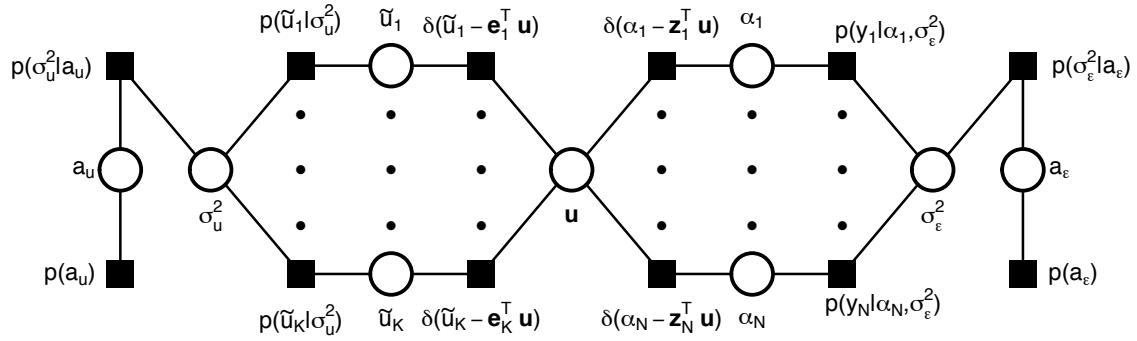

Figure 6: *Derived variable factor graph corresponding to the representation given in (24) of the joint density function of all random variables in the Gaussian response pure random effects model (23).*

for the natural parameters of Inverse Gamma messages.

The final model that we discuss is the Gaussian response linear mixed model

$$\begin{aligned} \boldsymbol{y}|\,\boldsymbol{\beta},\boldsymbol{u},\sigma_\varepsilon^2 &\sim N(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{Z}\boldsymbol{u},\sigma_\varepsilon^2\,\boldsymbol{I}), \quad \boldsymbol{u}|\,\sigma_u^2 \sim N(\boldsymbol{0},\sigma_u^2\boldsymbol{I}), \quad \boldsymbol{\beta}\sim N(\boldsymbol{\mu}_{\boldsymbol{\beta}},\boldsymbol{\Sigma}_{\boldsymbol{\beta}}),\\ \sigma_\varepsilon^2|\,a_\varepsilon &\sim \text{Inverse-Gamma}(\tfrac{1}{2},1/a_\varepsilon), \quad a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2},1/A_\varepsilon^2),\\ \sigma_u^2|\,a_u &\sim \text{Inverse-Gamma}(\tfrac{1}{2},1/a_u), \quad a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2},1/A_u^2). \end{aligned} \tag{25}$$

The joint density function of the random variables in this model may be written

$$\begin{aligned} p(\boldsymbol{y},\boldsymbol{\beta},\boldsymbol{u},\sigma_\varepsilon^2,a_\varepsilon,\sigma_u^2,a_u) = &\left\{\int_{\mathbb{R}^{d^\beta}} p(\widetilde{\boldsymbol{\beta}})\,\boldsymbol{\delta}\left(\widetilde{\boldsymbol{\beta}}-\boldsymbol{E}_{d^\beta}^T\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\right)d\widetilde{\boldsymbol{\beta}}\right\}\,p(a_u)\,p(\sigma_u^2|\,a_u)\\ &\times\left\{\prod_{k=1}^{d^u}\int_{-\infty}^{\infty} p(\widetilde{u}_k|\,\sigma_u^2)\,\delta\left(\widetilde{u}_k-\boldsymbol{e}_{d^\beta+k}^T\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\right)d\widetilde{u}_k\right\}\\ &\times\left\{\prod_{i=1}^{N}\int_{-\infty}^{\infty} p(y_i|\,\alpha_i,\sigma_\varepsilon^2)\delta\left(\alpha_i-\boldsymbol{c}_i^T\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\right)d\alpha_i\right\}\\ &\times p(\sigma_\varepsilon^2|\,a_\varepsilon)\,p(a_\varepsilon) \end{aligned} \tag{26}$$

where the $\boldsymbol{c}_i$ are defined by

$$\boldsymbol{C}\equiv[\boldsymbol{X}\ \boldsymbol{Z}]=\begin{bmatrix}\boldsymbol{c}_1^T\\ \vdots\\ \boldsymbol{c}_N^T\end{bmatrix}$$

and now the α_i are derived variables corresponding to the

$$\boldsymbol{c}_i^T\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix},\quad 1\le i\le N.$$

Also, $\boldsymbol{E}_{d^\beta}$ is the $(d^\beta+d^u)\times d^\beta$ matrix with the $d^\beta\times d^\beta$ identity matrix at the top and zeroes elsewhere. A factor graph for tractable expectation propagation is given in Figure 7.

Most of the messages in Figure 7 are either identical or have similar forms to those given earlier. The messages that take different forms are

$$m_{\delta\left(\widetilde{\boldsymbol{\beta}}-\boldsymbol{E}_{d\boldsymbol{\beta}}^T\left[\begin{smallmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{smallmatrix}\right]\right)\to\widetilde{\boldsymbol{\beta}}}(\widetilde{\boldsymbol{\beta}}) = \exp\left\{\begin{bmatrix}\widetilde{\boldsymbol{\beta}}\\ \text{vec}\left(\widetilde{\boldsymbol{\beta}}\widetilde{\boldsymbol{\beta}}^T\right)\end{bmatrix}^T\boldsymbol{\eta}_{\delta\left(\widetilde{\boldsymbol{\beta}}-\boldsymbol{E}_{d\boldsymbol{\beta}}^T\left[\begin{smallmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{smallmatrix}\right]\right)\to\widetilde{\boldsymbol{\beta}}}\right\}$$

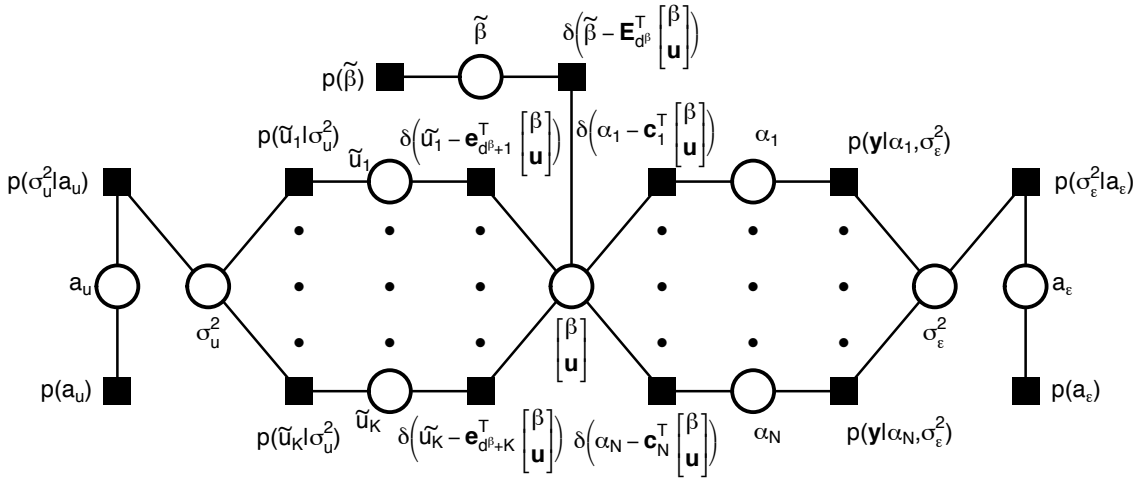

Figure 7: *Derived variable factor graph corresponding to the representation given in (26) of the joint density function of all random variables in the Gaussian response linear mixed model (25).*

where

$$\boldsymbol{\eta}_{\delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^{\boldsymbol{\beta}}}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right) \to \widetilde{\boldsymbol{\beta}}} \longleftarrow \left[\begin{array}{c} \left(\boldsymbol{E}_{d^\beta}^T \boldsymbol{\Sigma}^{\hexagram} \boldsymbol{E}_{d^\beta}\right)^{-1} \boldsymbol{E}_{d^\beta}^T \boldsymbol{\mu}^{\hexagram} \\ \\ -\frac{1}{2}\mathrm{vec}\left(\{\boldsymbol{E}_{d^\beta}^T \boldsymbol{\Sigma}^{\hexagram} \boldsymbol{E}_{d^\beta}\}^{-1}\right) \end{array}\right].$$

Here

$$\boldsymbol{\mu}^{\hexagram} \equiv \boldsymbol{\mu}_{\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right] \to \delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^\beta}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right)} \quad \text{and} \quad \boldsymbol{\Sigma}^{\hexagram} \equiv \boldsymbol{\Sigma}_{\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right] \to \delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^\beta}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right)}$$

with

$$\begin{aligned} &\boldsymbol{\mu}_{\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right] \to \delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^\beta}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right)} \\ &\quad \equiv -\frac{1}{2}\left\{\mathrm{vec}^{-1}\left(\boldsymbol{\eta}_{\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right] \to \delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^\beta}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right)}\right)_2\right\}^{-1} \left(\boldsymbol{\eta}_{\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right] \to \delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^\beta}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right)}\right)_1 \end{aligned}$$

and

$$\boldsymbol{\Sigma}_{\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right] \to \delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^\beta}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right)} \equiv -\frac{1}{2}\left\{\mathrm{vec}^{-1}\left(\boldsymbol{\eta}_{\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right] \to \delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^\beta}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right)}\right)_2\right\}^{-1}.$$

Also, using the identity $\mathrm{vec}(\boldsymbol{ABC}) = (\boldsymbol{C}^T \otimes \boldsymbol{A})\mathrm{vec}(\boldsymbol{B})$, we have

$$\begin{aligned} &m_{\delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^{\boldsymbol{\beta}}}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right) \to \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]}\left(\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right) \\ &\qquad = \exp\left\{\left[\begin{array}{c} \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right] \\ \\ \mathrm{vec}\left(\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]^T\right) \end{array}\right]^T \boldsymbol{\eta}_{\delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^{\boldsymbol{\beta}}}^T \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\right) \to \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]}\right\}. \end{aligned}$$

where

$$\boldsymbol{\eta}_{\delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^{\boldsymbol{\beta}}}^T \left[\begin{array}{c} \boldsymbol{\beta} \\ \boldsymbol{u} \end{array}\right]\right) \to \left[\begin{array}{c} \boldsymbol{\beta} \\ \boldsymbol{u} \end{array}\right]} \longleftarrow \left[\begin{array}{c} \boldsymbol{E}_{d^\beta}\left(\boldsymbol{\eta}_{\delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^{\boldsymbol{\beta}}}^T \left[\begin{array}{c} \boldsymbol{\beta} \\ \boldsymbol{u} \end{array}\right]\right) \to \widetilde{\boldsymbol{\beta}}}\right)_1 \\ (\boldsymbol{E}_{d^\beta} \otimes \boldsymbol{E}_{d^\beta})\left(\boldsymbol{\eta}_{\delta\left(\widetilde{\boldsymbol{\beta}} - \boldsymbol{E}_{d^{\boldsymbol{\beta}}}^T \left[\begin{array}{c} \boldsymbol{\beta} \\ \boldsymbol{u} \end{array}\right]\right) \to \widetilde{\boldsymbol{\beta}}}\right)_2 \end{array}\right].$$

In the special case of $d^\beta = 1$ the matrix $\boldsymbol{E}_{d^\beta}$ reduces to $\boldsymbol{e}_1$ and

$$\boldsymbol{E}_{d^\beta} \otimes \boldsymbol{E}_{d^\beta} = (\boldsymbol{e}_1 \otimes \boldsymbol{e}_1)\text{vec}(1) = \text{vec}(\boldsymbol{e}_1\boldsymbol{e}_1^T)$$

which matches the form exemplified by (15).

Expectation propagation for generalized linear mixed models involves a combination of the generalized response extensions described in Section 4 and the mixed model extensions described in this section. For example, expectation propagation fitting and inference for the Poisson mixed model

$$\begin{gathered} y_i|\,\boldsymbol{\beta}, \boldsymbol{u} \overset{\text{ind.}}{\sim} \text{Poisson}\{\exp(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{u})\}, \quad \boldsymbol{u}|\,\sigma^2 \sim N(\boldsymbol{0}, \sigma^2\boldsymbol{I}), \quad \boldsymbol{\beta} \sim N(\boldsymbol{\mu}_{\boldsymbol{\beta}}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}}), \\ \sigma^2|\,a \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/a), \quad a \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/A^2) \end{gathered} \tag{27}$$

involves message passing on a modification of the Figure 7 factor graph for which likelihood factors resemble those appearing in Figure 5.

6 Numerical Investigations

We implemented expectation propagation schemes for various generalized, linear and mixed models in the R language (R Core Team, 2016). Comparable (semiparametric) mean field variational Bayes algorithms were also implemented in R. We then ran simulation studies, each having 100 replications, for four specific models and recorded accuracy values of the deterministic approximation methods for key model parameters. For a generic univariate parameter θ, the accuracy of an approximation $q(\theta)$ to $p(\theta|\boldsymbol{D})$ is defined to be

$$\text{accuracy} \equiv 100\left\{1 - \tfrac{1}{2}\int_{-\infty}^{\infty} \left|q(\theta) - p(\theta|\boldsymbol{D})\right| d\theta\right\}\%$$

(e.g. Faes *et al.*, 2011). However, since the exact posterior density function $p(\theta|\boldsymbol{D})$ is not easily computable, we used a binned kernel density estimation approximation based on $1,000,000$ Markov chain Monte Carlo draws from $\theta|\boldsymbol{D}$. All binned kernel density estimates were obtained using the R function `bkde()` in the package KernSmooth (Wand & Ripley, 2015) with direct plug-in bandwidth selection via the function `dpik()`.

6.1 Linear Model Setting

The linear model setting involved a $d = 5$ version of (10) with data generating according to the true values

$$\boldsymbol{\beta}_{\text{true}} = [-2.3 \;\; 1.4 \;\; -0.9 \;\; 2.1 \;\; 0.2]^T \quad \text{and} \quad \sigma^2_{\text{true}} = 1. \tag{28}$$

The sample size was fixed at $n = 100$. For the design matrix $\boldsymbol{X}$, the first column was set to a vector of ones and the remaining columns corresponded to independent random samples from the Uniform distribution on $(0, 1)$. The hyperparameters were fixed at $\sigma_{\boldsymbol{\beta}} = A = 10,000$.

Expectation propagation approximate inference involved running Algorithm 1. Convergence was deemed to have occurred when the absolute relative change in the natural

parameters of each of the posterior density functions fell below a very small threshold, nominally set to be 10^{-10}. We also obtained mean field variational Bayes approximate posterior density functions using Algorithm 1 of Luts *et al.* (2014), with convergence based on the relative increase in the approximate marginal log-likelihood falling below 10^{-10}.

Figure 9 shows approximate posterior density functions of each of the main model parameters, $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_4)$ and σ^2, for the first replication from the simulation study. The 'exact' Markov chain Monte Carlo-based posterior density functions are also shown as well as corresponding accuracy values. The expectation propagation and mean field variational Bayes approximations to the posterior density functions of the β_js are very accurate in this example, and cannot be visually distinguished from the 'exact' posterior density functions. The accuracy values are all above 99.5%. There is slight degradation in quality for approximation of $p(\sigma^2|\boldsymbol{y})$ with an 98.5% accuracy for expectation propagation and 98.0% accuracy for mean field variational Bayes.

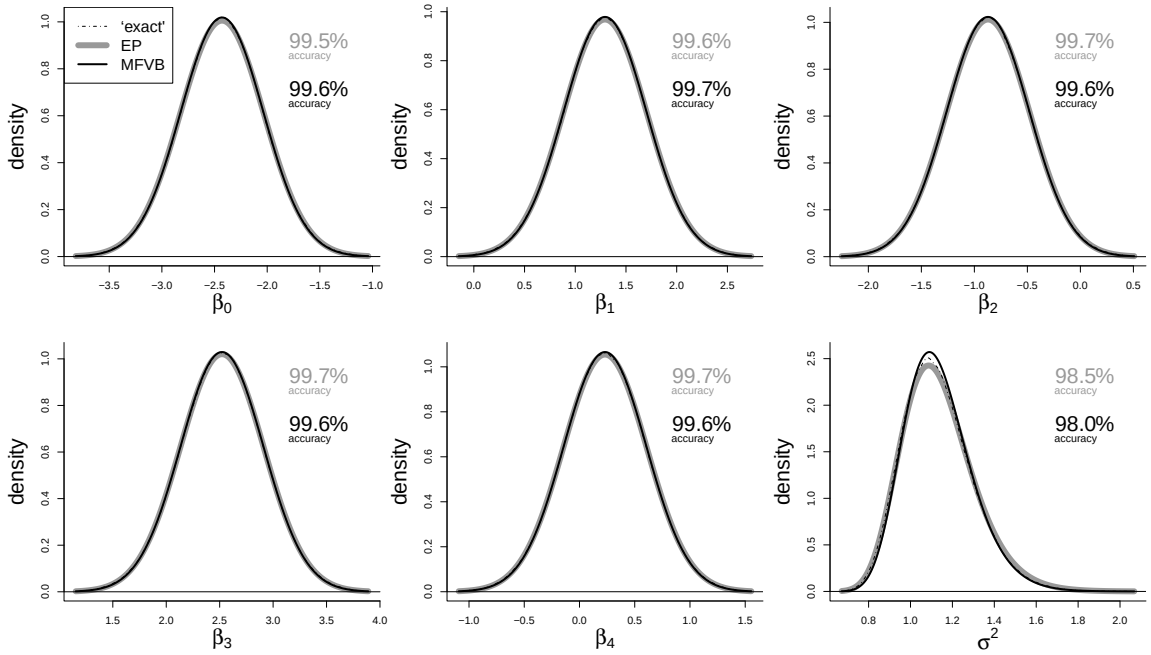

Figure 8: *Comparison of approximate posterior density functions with their 'exact' inference counterparts for the main model parameters in the first replication of the linear model simulation study described in the text. The approximate posterior density functions are based on expectation propagation (EP) as described by Algorithm 1 and mean field variational Bayes (MFVB) as described by Algorithm 1 of Luts et al. (2014). The 'exact' posterior density functions are kernel density estimates based on 1,000,000 Markov chain Monte Carlo draws from the relevant posterior distribution. The accuracy of each approximate posterior density function is also displayed.*

Summaries of the accuracy values across the entire 100 replications are shown in Figure 9. In the left panel we focus on the expectation propagation accuracy values whereas the their difference from the mean field variational Bayes accuracy values are summarised in the right panel.

The left panel of Figure 9 shows that expectation propagation achieves very high accuracy for estimation of the regression coefficients. The accuracy values for estimation of the error variance are lower, but still very good with the median exceeding 98%. The comparison boxplots on the right panel indicates that expectation propagation has superior accuracy compared with mean field variational Bayes in terms of statistical significance. However, the practical significance is quite small with a typical improvement of 0.1% for the regression coefficients and 0.4% for the error variance.

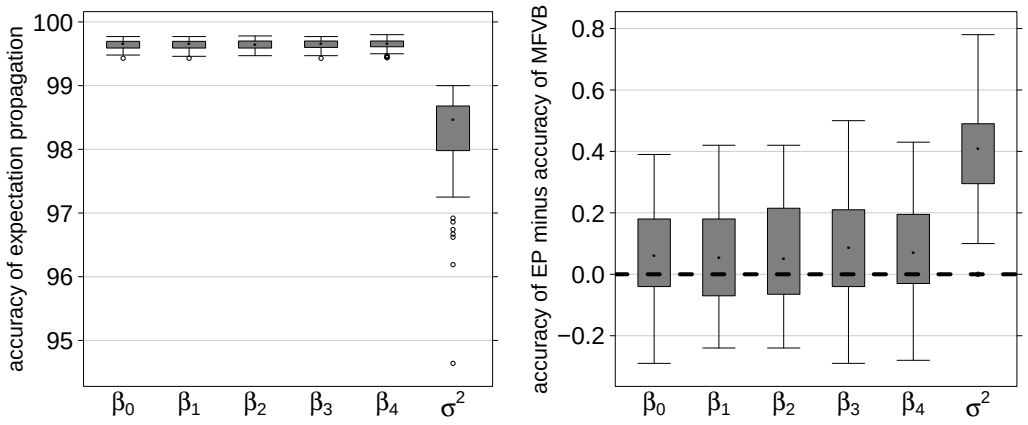

Figure 9: *Boxplot summaries for the linear model simulation study described in the text. Left panel: boxplots of accuracy values of expectation propagation. Right panel: boxplots of the differences between the expectation propagation (EP) and mean field variational Bayes (MFVB) accuracy values.*

6.2 Poisson Regression Setting

The second part of our numerical investigations involved simulation from the Poisson regression model

$$y_i|\boldsymbol{\beta} \overset{\text{ind.}}{\sim} \text{Poisson}\left[\exp\{(\boldsymbol{X}\boldsymbol{\beta})_i\}\right], \quad \boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_{\boldsymbol{\beta}}\boldsymbol{I}), \quad 1 \leq i \leq 100.$$

The true coefficients were generated according to $\boldsymbol{\beta}_{\text{true}}$ as given in (28) and $\sigma_{\boldsymbol{\beta}}$ was set to $10,000$ again.

The expectation propagation approximate density functions were obtained using the message passing approach described in Section 4 for the generic factor graph depicted in Figure 5. In the Poisson case $b(x) = \exp(x)$ and $\mathcal{C}_b(p, q, r) = (-1)^p \mathcal{J}(p, -q, r, 1)$ where $\mathcal{J}$ is defined in Section 2.1 of Wand *et al.* (2011), which allowed us to use code and computational advice given in this earlier article.

Ordinary mean field variational Bayes is thwarted by tractability problems for Poisson response models. As explained in, for example, Tan & Nott (2013) and Luts & Wand (2015) a semiparametric extension in which the q-density function of $\boldsymbol{\beta}$ is pre-specified to have a Multivariate Normal distribution leads to closed form updating algorithm and is a special case of Algorithm 1 of Luts & Wand (2015). Indeed, in this relatively simple model there is no mean field aspect to the approximation and the approximation to $p(\boldsymbol{\beta}|\boldsymbol{y})$ is of the Gaussian minimum Kullback-Leibler type laid out in, for example, Challis & Barber (2013). Analogous semiparametric mean field variational Bayes schemes exist for other Bayesian generalized linear model settings such as logistic regression. The Poisson case has the attraction of admitting closed form updates.

Figure 10 is the analogue of Figure 9 for the Poisson regression setting. As in the Gaussian response case, expectation propagation is seen to achieve excellent accuracy for this setting. Moreover, it is uniformly more accurate than mean field variational Bayes but only by a very small amount — typically less than 0.1%.

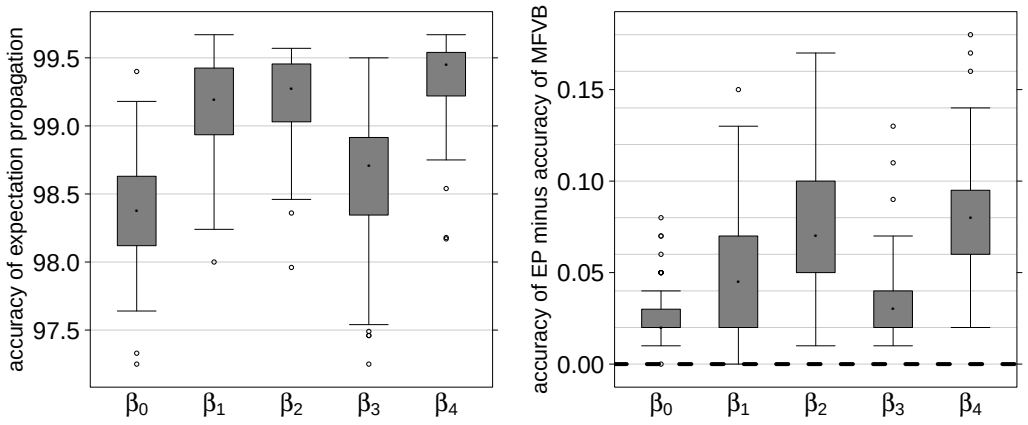

Figure 10: *Boxplot summaries for the Poisson regression simulation study described in the text. Left panel: boxplots of accuracy values of expectation propagation. Right panel: boxplots of the differences between the expectation propagation (EP) and mean field variational Bayes (MFVB) accuracy values.*

6.3 Linear Mixed Model Setting

We next investigated the efficacy of expectation propagation for the following linear mixed model:

$$y_{ij}|\beta_0, U_i, \beta_1, \sigma_\varepsilon^2 \overset{\text{ind.}}{\sim} N(\beta_0 + U_i + \beta_1 x_{ij}, \sigma_\varepsilon^2), \quad U_i|\sigma_u^2 \overset{\text{ind.}}{\sim} N(0, \sigma_u^2),\ 1 \le i \le 50,\ 1 \le j \le 5,$$

$$\beta_0, \beta_1 \overset{\text{ind.}}{\sim} N(0, \sigma_{\boldsymbol{\beta}}^2), \quad \sigma_\varepsilon^2|\, a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/a_\varepsilon), \quad a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/A_\varepsilon^2),$$

$$\sigma_u^2|\, a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/a_u), \quad a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/A_u^2).$$

The true values in the simulation study were

$$\beta_{0,\text{true}} = 2.41, \quad \beta_{1,\text{true}} = 1.89, \quad \sigma_{\varepsilon,\text{true}} = 0.1 \quad \text{and} \quad \sigma_{u,\text{true}} = 1.58.$$

The expectation propagation approximate density functions were obtained using the message passing approach described in Section 5 for the factor graph depicted in Figure 7. The mean field variational Bayes analogue is a slight modification of the $r = 1$ version of Algorithm 3 of Ormerod & Wand (2010) to accommodate different priors on the variance parameters.

The accuracy results are summarised in Figure 11. For the fixed effects parameters the accuracy of expectation propagation is seen to be excellent, but for the variance parameters there is a noticeable degradation in accuracy with accuracy scores between about 94.5% and 96%. Interestingly, mean field variational Bayes is more accurate for estimation of β_0 and β_1, but less accurate for estimation of σ_ε^2 and σ_u^2. Once again, the differences are not practically significant.

6.4 Poisson Mixed Model Setting

The final simulation setting involved generalized linear mixed model fitting and inference via a version of the model:

$$y_i|\boldsymbol{\beta}, \boldsymbol{u} \overset{\text{ind.}}{\sim} \text{Poisson}\left[\exp\{(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{u})_i\}\right], \quad 1 \le i \le 300, \quad \boldsymbol{u}|\sigma^2 \sim N(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$$

$$\boldsymbol{\beta} \sim N(\boldsymbol{0}, \sigma_{\boldsymbol{\beta}}^2 \boldsymbol{I}), \quad \sigma^2|\, a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/a), \quad a \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/A^2)$$

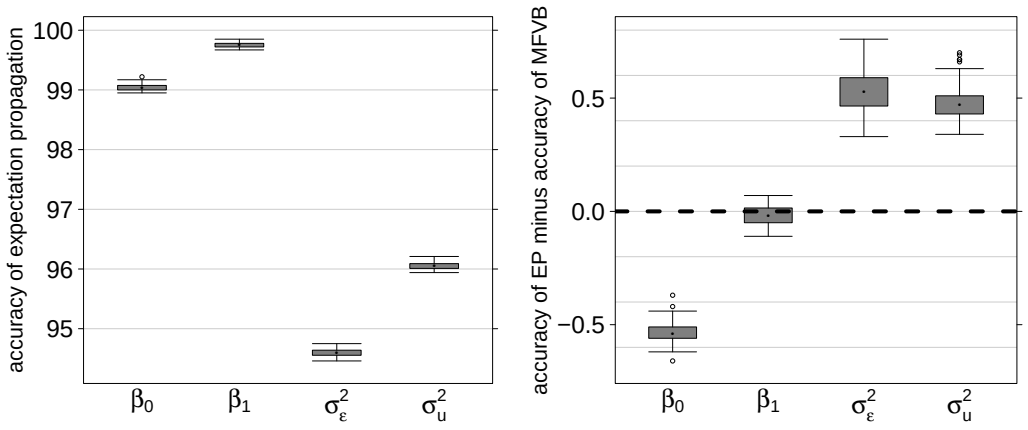

Figure 11: *Boxplot summaries for the linear mixed model simulation study described in the text. Left panel: boxplots of accuracy values of expectation propagation. Right panel: boxplots of the differences between the expectation propagation (EP) and mean field variational Bayes (MFVB) accuracy values.*

In the interests of variety we chose the design matrices to correspond to mixed model-based penalized spline fitting (e.g. Ruppert *et al.*, 2009), with

$$\boldsymbol{X} = [1 \; x_i]_{1 \le i \le n} \quad \text{and} \quad \boldsymbol{Z} = [z_k(x_i)]_{\substack{1 \le k \le K \\ 1 \le i \le n}}$$

where $z_k(\cdot)$, $1 \le k \le K$ are a suitable spline basis. In our simulation study we used canonical O'Sullivan splines as described in Section 2.1 of Ruppert *et al.* (2009). We generated the data according to the Poisson nonparametric regression model

$$x_i \overset{\text{ind.}}{\sim} \text{Uniform}(0,1), \quad y_i \overset{\text{ind.}}{\sim} \text{Poisson}\{f_{\text{true}}(x_i)\} \quad \text{where} \quad f_{\text{true}}(x) = \exp\{2x + \cos(4\pi x)\}.$$

Expectation propagation approximations to the posterior density functions of the model parameters were obtained using message passing on a modification of the Figure 7 factor graph, appropriate for the Poisson likelihood. Semiparametric mean field variational Bayes fitting involved running a special case of Algorithm 1 of Luts & Wand (2015). Figure 12 shows the fitted curves and corresponding pointwise 95% credible sets for the first replication of the simulation study. For this example the is no visually discernible difference between the two approximation methods.

To evaluate the performance of expectation propagation we used accuracy values for estimation of f at the hexiles of the predicator data. Specifically, the parameters of interest were set to be

$$f(H_i), \quad 1 \le i \le 5,$$

where H_i is the ith sample hexile of the x_is. Once again, expectation propagation is very accurate and slightly ahead of mean field variational Bayes in this metric.

7 Concluding Discussion

We have provided full algorithmic prescriptions for implementation of expectation propagation in generalized, linear and mixed model settings. The factor graph/message passing approach is shown to facilitate extensions to larger models with minimal algebraic overhead. Such an approach is used by the Infer.NET software package. For mean field variational Bayes/variational message passing Wand (2017) introduces the notion of factor graph fragments to organise and streamline the required algebra and computing and

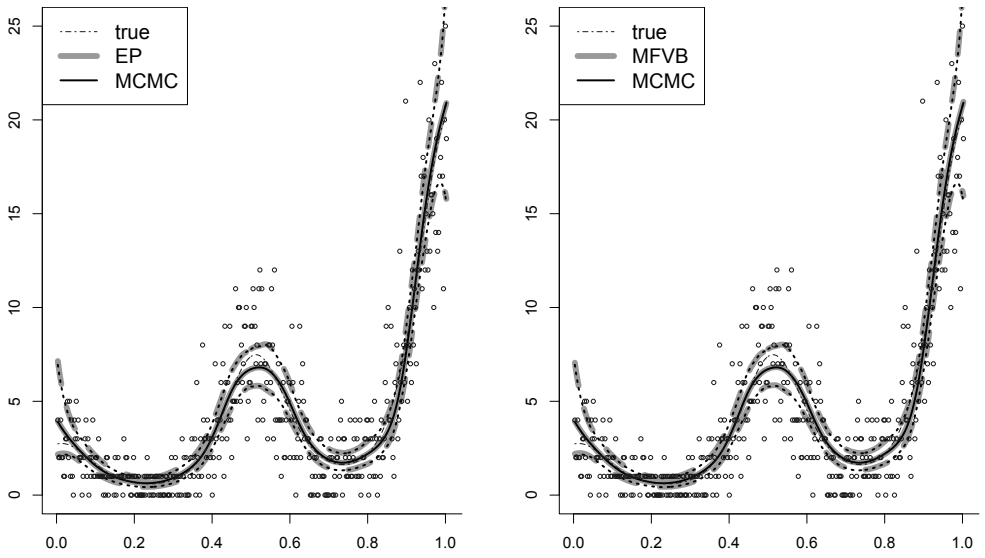

Figure 12: *Comparison of approximate Poisson nonparametric regression fits with their 'exact' inference counterparts in the first replication of the Poisson mixed model simulation study described in the text. The solid curves correspond to the pointwise posterior means and the dashed curves correspond to pointwise 95% credible intervals. The 'exact' inference fits are based on 1,0000,000 Markov chain Monte Carlo draws. Left panel: the comparison for expectation propagation. Right panel: the comparison for mean field variational Bayes.*

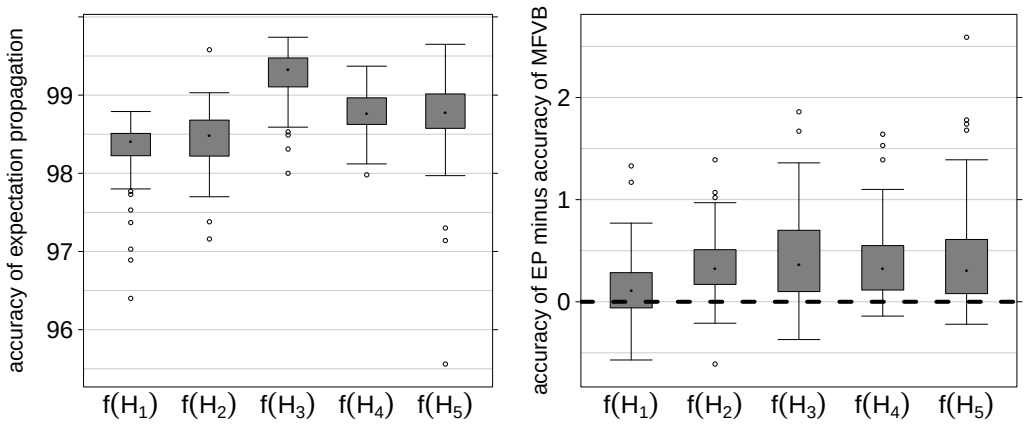

Figure 13: *Boxplot summaries for the Poisson mixed model-based penalized splines simulation study described the text. The accuracy values correspond to approximate Bayesian inference for $f(H_i)$, $1 \leq i \leq 5$, where H_i is the ith hexile of the predictor variable. Left panel: boxplots of accuracy values of expectation propagation. Right panel: boxplots of the differences between the expectation propagation (EP) and mean field variational Bayes (MFVB) accuracy values.*

catalogues important fragments for semiparametric regression. The fragment idea also applies to expectation propagation and similar cataloguing could be done for generalized, linear and mixed models, based on the formulae developed in this article.

Despite the elegance of expectation propagation, we have not been able to find any compelling reasons for its adoption in generalized, linear and mixed models given that its accuracy is very similar to that of (semiparametric) mean field variational Bayes but

at the cost of much more complicated algebra and computing. Our numerical studies are necessarily limited and we would welcome further investigations on this issue.

Lastly, we discuss some connections with the research of Peter Hall, to whom this special issue of the *Australian and New Zealand Journal of Statistics* is dedicated. Second author Wand worked with Hall over a period of almost 30 years starting with Hall supervising Wand's PhD thesis at the Australian National University during 1986–1988. First author Kim is a PhD student of Wand, and therefore a 'grandstudent' of Hall. In 2008, after a long intermission, Hall and Wand recommenced their research collaborations in response to Wand's interest in deterministic approximate inference methodology. In contrast to the present article on approximate Bayesian inference our work focussed on the *frequentist* statistical properties of deterministic approximations, with non-Bayesian mixed models being the main vehicle for our theoretical exploits. Two publications resulted: Hall, Ormerod & Wand (2011) and Hall, Pham, Wand & Wang (2011). A third project, concerned with frequentist theory for expectation propagation, was commenced in 2011 but was not completed before the loss of Hall's unique theoretical prowess. Nevertheless, efforts are being made to bring this work to fruition.

Our 2008–2013 collaborations on deterministic approximate inference are emblematic of Hall's astonishing theoretical talent, his willingness to help fellow researchers and to move into different areas. When Wand pitched his variational inference problem to Hall in September 2008 Hall had barely worked in this area. Within weeks Hall had produced several pages of theory of a quality sufficient for an eventual *Statistica Sinica* publication. A similar story applies to a *The Annals of Statistics* article which was published in the same year. Peter had the rare ability to make such novel and profound theoretical contributions very quickly, despite his many other commitments and projects. Author Wand is extremely fortunate and grateful to have had such a talented and generous mentor, supporter, collaborator and friend.

Appendix: Proof of Theorem 1

Throughout this proof we let

$$\phi(x) \equiv (2\pi)^{-1/2} e^{-x^2/2} = p_{N_1}(x; 0, 1),$$

the Standard Normal density function. Our proof relies on the following easy-to-derive lemma:

Lemma 1: *For real numbers $a \neq 0$, b and c,*

$$|a|^{-1} \int_{-\infty}^{\infty} \phi(x)\phi\left(\frac{b\,x - c}{a}\right) dx = \left(a^2 + b^2\right)^{-1/2} \phi\left(\frac{c}{(a^2+b^2)^{1/2}}\right).$$

Via the change of variable $\boldsymbol{z} = \boldsymbol{\Sigma}^{-1/2}(\boldsymbol{x} - \boldsymbol{\mu})$ the left-hand side of (3) is

$$\int_{\mathbb{R}^d} (2\pi)^{-d/2} \exp\left(-\tfrac{1}{2}\boldsymbol{z}^T\boldsymbol{z}\right) \delta(v - \boldsymbol{\ell}^T\boldsymbol{\mu} - \boldsymbol{\ell}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{z})\, d\boldsymbol{z} = \int_{\mathbb{R}^d} \prod_{j=1}^{d} \phi(z_j)\delta\left(v^\dagger - (\boldsymbol{\ell}^\dagger)^T\boldsymbol{z}\right) d\boldsymbol{z}$$

where z_j is the jth entry of $\boldsymbol{z}$,

$$v^\dagger \equiv v - \boldsymbol{\ell}^T\boldsymbol{\mu} \quad \text{and} \quad \boldsymbol{\ell}^\dagger \equiv \boldsymbol{\Sigma}^{1/2}\boldsymbol{\ell}.$$

Since $\boldsymbol{\Sigma}^{1/2}$ is invertible $\boldsymbol{\ell}^\dagger \neq 0$. Therefore at least one entry of $\boldsymbol{\ell}^\dagger$ is non-zero and, without loss of generality, we can re-order the entries of $\boldsymbol{\ell}^\dagger = (\ell_1^\dagger, \ldots, \ell_d^\dagger)$ and $\boldsymbol{z} = (z_1, \ldots, z_d)$ in

$$\int_{\mathbb{R}^d} \prod_{j=1}^{d} \phi(z_j)\delta\left(v^\dagger - (\boldsymbol{\ell}^\dagger)^T\boldsymbol{z}\right) d\boldsymbol{z} = \int_{\mathbb{R}^{d-1}} \prod_{j=2}^{d} \phi(z_j) \int_{-\infty}^{\infty} \phi(z_1)\delta\left(v^\dagger - \sum_{j=1}^{d} \ell_j^\dagger z_j\right) dz_1 \cdots dz_d$$

so that $\ell_1^\dagger \neq 0$. Using the change of variable $u = -\ell_1^\dagger z_1$, the innermost integral is

$$|\ell_1^\dagger|^{-1} \int_{-\infty}^{\infty} \phi\left(\frac{u}{\ell_1^\dagger}\right) \delta\left(u - \left(\sum_{j=2}^{d} \ell_j^\dagger z_j - v^\dagger\right)\right) du = |\ell_1^\dagger|^{-1} \phi\left(\frac{\sum_{j=2}^{d} \ell_j^\dagger z_j - v^\dagger}{\ell_1^\dagger}\right)$$

which leads to

$$\begin{aligned}
&\int_{\mathbb{R}^d} \prod_{j=1}^{d} \phi(z_j) \delta(v^\dagger - (\boldsymbol{\ell}^\dagger)^T \boldsymbol{z})\, d\boldsymbol{z} \\
&\quad = \int_{\mathbb{R}^{d-2}} \prod_{j=3}^{d} \phi(z_j) |\ell_1^\dagger|^{-1} \int_{-\infty}^{\infty} \phi(z_2) \phi\left(\frac{\ell_2^\dagger z_2 - (v^\dagger - \sum_{j=3}^{d} \ell_j^\dagger z_j)}{\ell_1^\dagger}\right) dz_2 \cdots dz_d \\
&\quad = \int_{\mathbb{R}^{d-2}} \prod_{j=3}^{d} \phi(z_j) \left\{(\ell_1^\dagger)^2 + (\ell_2^\dagger)^2\right\}^{-1/2} \phi\left(\frac{\sum_{j=3}^{d} \ell_j^\dagger z_j - v^\dagger}{\{(\ell_1^\dagger)^2 + (\ell_2^\dagger)^2\}^{1/2}}\right) dz_3 \cdots dz_d,
\end{aligned}$$

where Lemma 1 has been used to evaluate the integral with respect to z_2. Repeated use of Lemma 1 for the remaining integrals leads to

$$\begin{aligned}
\int_{\mathbb{R}^d} \prod_{j=1}^{d} \phi(z_j) \delta(v^\dagger - (\boldsymbol{\ell}^\dagger)^T \boldsymbol{z})\, d\boldsymbol{z} &= \left\{(\ell_1^\dagger)^2 + \cdots + (\ell_d^\dagger)^2\right\}^{-1/2} \phi\left(\frac{v^\dagger}{\left\{(\ell_1^\dagger)^2 + \ldots + (\ell_d^\dagger)^2\right\}^{1/2}}\right) \\
&= \{(\boldsymbol{\ell}^\dagger)^T \boldsymbol{\ell}^\dagger\}^{-1/2} \phi\big(v^\dagger / \{(\boldsymbol{\ell}^\dagger)^T \boldsymbol{\ell}^\dagger\}^{1/2}\big) \\
&= (\boldsymbol{\ell}^T \boldsymbol{\Sigma} \boldsymbol{\ell})^{-1/2} \phi\big((v - \boldsymbol{\ell}^T \boldsymbol{\mu}) / (\boldsymbol{\ell}^T \boldsymbol{\Sigma} \boldsymbol{\ell})^{1/2}\big)
\end{aligned}$$

which gives the required result.

Acknowledgements

This research was partially supported by the Australian Research Council Centre of Excellence for Mathematical and Statistical Frontiers.

References

Azzalini, A. (2016). The R package 'sn': The skew-normal and skew-t distributions (version 1.2). `http://azzalini.stat.unipd.it/SN`

Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*. New York: Springer.

Challis, E. & Barber, D. (2013). Gaussian Kullback-Leibler approximate inference. *Journal of Machine Learning Research*, **14**, 2239–2286.

Faes, C., Ormerod, J.T. & Wand, M.P. (2011). Variational Bayesian inference for parametric and nonparametric regression with missing data. *Journal of the American Statistical Association*, **106**, 959–971.

Frey, B.J., Kschischang, F.R., Loeliger, H.A. & Wiberg, N. (1998). Factor graphs and algorithms. In *Proceedings of the 35th Allerton Conference on Communication, Control and Computing 1997*.

Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A. & Rubin, D.B. (2014). *Bayesian Data Analysis, Third Edition*, Boca Raton, Florida: CRC Press.

Hall, P., Ormerod, J.T. & Wand, M.P. (2011). Theory of Gaussian variational approximation for a Poisson linear mixed model. *Statistica Sinica*, **21**, 369–389.

Hall, P., Pham, T., Wand, M.P. & Wang, S.S.J. (2011). Asymptotic normality and valid inference for Gaussian variational approximation. *The Annals of Statistics*, **39**, 2502–2532.

Kim, A.S.I. & Wand, M.P. (2016). The explicit form of expectation propagation for a simple statistical model. *Electronic Journal of Statistics*, **10**, 550–581.

Luts, J., Broderick, T. & Wand, M.P. (2014). Real-time semiparametric regression. *Journal of Computational and Graphical Statistics*, **23**, 589–615.

Luts, J. & Wand, M.P. (2015). Variational inference for count response semiparametric regression. *Bayesian Analysis*, **10**, 991–1023.

Luts, J., Wang, S.S.J., Ormerod, J.T. & Wand, M.P. (2017). Semiparametric regression analysis via Infer.NET. *Journal of Statistical Software*, in press.

McCulloch, C.E., Searle, S.R. and Neuhaus, J.M. (2008). *Generalized, Linear, and Mixed Models, Second Edition*. New York: John Wiley & Sons.

Minka, T.P. (2001). Expectation propagation for approximate Bayesian inference. In J.S. Breese & D. Koller (eds), *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, pp. 362–369. Burlington, Massachusetts: Morgan Kaufmann.

Minka, T. (2005), Divergence measures and message passing. *Microsoft Research Technical Report Series*, **MSR-TR-2005-173**, 1–17.

Minka, T. & Winn, J. (2008), Gates: A graphical notation for mixture models. *Microsoft Research Technical Report Series*, **MSR-TR-2008-185**, 1–16.

Minka, T., Winn, J.M., Guiver, J.P., Webster, S., Zaykov, Y., Yangel, B., Spengler, A. & Bronskill, J. (2014). Infer.NET 2.6, Microsoft Research Cambridge, 2014. `http://research.microsoft.com/infernet`

Ormerod, J.T. & Wand, M.P. (2010). Explaining variational approximations. *The American Statistician*, **64**, 140–153.

R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. `https://www.R-project.org/`

Ruppert, D., Wand, M.P. & Carroll, R.J. (2009). Semiparametric regression during 2003-2007. *Electronic Journal of Statistics*, **3**, 1193–1256.

Tan, L.S.L. & Nott, D.J. (2013). Variational inference for generalized linear mixed models using partially noncentered parametrizations. *Statistical Science*, **28**, 168–188.

Wainwright, M.J. & Jordan, M.I. (2008). Graphical models, exponential families and variational inference. *Foundations and Trends in Machine Learning*, **1**, 1–305.

Wand, M.P. (2017). Fast approximate inference for arbitrarily large semiparametric regres-

sion models via message passing (with discussion). *Journal of the American Statistical Association*, **112**, in press.

Wand, M.P., Ormerod, J.T., Padoan, S.A. & Frühwirth, R. (2011). Mean field variational Bayes for elaborate distributions. *Bayesian Analysis*, **6**, 847–900.

Wand, M.P. & Ripley, B.D. (2015). KernSmooth 2.23 Functions for kernel smoothing supporting Wand & Jones (1995). R package. `https://cran.R-project.org`.

Wang, S.S.J. & Wand, M.P. (2011). Using Infer.NET for statistical analyses. *The American Statistician*, **65**, 115–126.

Winn, J. & Bishop, C.M. (2005). Variational message passing. *Journal of Machine Learning Research*, **6**, 661–694.