Doctoral theses at NTNU, 2007:214

Sara Martino

Approximate Bayesian Inference for Latent Gaussian Models

Doctoral Thesis

NTNU
Norwegian University of
Science and Technology
Thesis for the
degree philosophiae doctor
Faculty of Information Technology, Mathematics
and Electrical Engineering
Department of Mathematical Sciences

THESIS OUTLINE

The thesis consists of the following papers:

Paper I: **Approximate Inference for Hierarchical Gaussian Markov Random Field Models.**
With Håvard Rue. Published in *Journal of Statistical Planning and Inference*, October 2007, Vol. 137.

Paper II: **Approximate Bayesian Inference for Latent Gaussian Models Using Integrated Nested Laplace Approximations.**
With Håvard Rue and Nicolas Chopin. Submitted for publication.

Paper III: **Approximate Bayesian Inference in Spatial Generalised Linear Mixed Models.**
With Jo Eidsvik and Håvard Rue. In revision.

Paper IV: **Approximate Bayesian Inference for Multivariate Stochastic Volatility Models**
Report.

Paper V: **Implementing Approximate Bayesian Inference using Integrated Nested Laplace Approximation: a manual for the `inla` program.**
With Håvard Rue. Report.

BACKGROUND

Latent (or hidden) Gaussian models are a large and flexible class of statistical models often encountered in applications. The core of such models is an unobserved multivariate Gaussian random variable $\boldsymbol{x}$, whose density $\pi(\boldsymbol{x}|\boldsymbol{\theta})$ is controlled by a vector of parameters $\boldsymbol{\theta}$. Some of the elements in the random vector $\boldsymbol{x}$ are indirectly observed through the data $\boldsymbol{y}$. These are assumed to be conditionally independent given the latent field $\boldsymbol{x}$, i.e. $\pi(\boldsymbol{y}|\boldsymbol{x}) = \prod \pi(y_i|x_i)$.

The elements of a latent Gaussian model are then i) the likelihood of the data $\pi(\boldsymbol{y}|\boldsymbol{x})$, ii) the Gaussian density of the random vector $\boldsymbol{x}$, $\pi(\boldsymbol{x}|\boldsymbol{\theta})$ and iii) the prior distribution of the parameter vector $\pi(\boldsymbol{\theta})$. The posterior distribution then reads:

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) \propto \pi(\boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta}) \prod_i \pi(y_i|x_i)$$

We assume throughout that the main inferential interest is in the posterior marginals for x_i and, possibly, in the posterior marginals for $\boldsymbol{\theta}$ or for some θ_j.

The latent Gaussian field $\boldsymbol{x}$ provides a flexible tool to model time and spatial dependence within data and among data and potential covariates. A wide range of models well known from the literature can be formulated as special cases of latent Gaussian models, for example: generalised additive models (Hastie and Tibshirani, 1990), generalised additive mixed models (Lin and Zhang, 1999), geoadditive models (Kammand and Wand, 2003), univariate and multivariate stochastic volatility models (Durbin and Koopman, 1997; Yu and Mayer, 2006). Model-based geostatistics (Diggle et al., 1998, 2003) and models for log-Gaussian Cox processes (Møller et al., 1998) also belong to this class. See also Rue and Held (2005) for several references to different application of latent Gaussian models. Typical examples of latent Gaussian models present a high dimensional latent field $\boldsymbol{x}$ and a low dimensional vector of parameters $\boldsymbol{\theta}$.

At present, the standard tool for Bayesian inference on such models is Markov Chain Monte Carlo (MCMC). However, the hierarchical structure of the model, the (often) high dimensionality of the latent field $\boldsymbol{x}$, and the strong correlation within $\boldsymbol{x}$ and between $\boldsymbol{x}$ and $\boldsymbol{\theta}$ create problems for the convergence and the mixing properties of the Markov chain. Block update strategies have been developed to try to overcome such problems (see for example Knorr-Held and Rue (2002) and Rue and Held (2005)) but in many cases, MCMC algorithms remain remarkably slow.

The work of this thesis is driven by the idea that, for a large subset of latent Gaussian models, MCMC simulations can be entirely bypassed in favour of a new approach based on deterministic approximations to the posterior marginals of interest. The main advantage of such simulations-free approach is computational: answers can be obtained in seconds and minutes when MCMC algorithms would require hours and days. Moreover the approximations described in this thesis appear to be extremely accurate so that, in order for any bias to be detected, the MCMC algorithm would have to run for much longer time than it is usually done in practice.

The core of the approximation techniques presented in this thesis is a Gaussian approximation to the full conditional of the latent field $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ built by matching the mode and the curvature at the mode. This is indicated as $\widetilde{\pi}_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$. An approximation to the posterior marginal of the parameters $\boldsymbol{\theta}$ is then built as

$$\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y}) \propto \left. \frac{\pi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\theta})}{\widetilde{\pi}_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})} \right|_{\boldsymbol{x}=\boldsymbol{x}^\star(\boldsymbol{\theta})} \tag{1}$$

where $\boldsymbol{x}^\star(\boldsymbol{\theta})$ is the modal configuration of $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$. The approximation in (1) is equivalent to Tierney and Kadane (1986)'s Laplace approximation of a marginal posterior distribution. Finally, posterior marginals for the latent field are approximated as

$$\widetilde{\pi}(x_i|\boldsymbol{y}) = \sum_k \widetilde{\pi}(x_i|\boldsymbol{y}, \boldsymbol{\theta}_k) \times \widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y}) \times \Delta_k \tag{2}$$

where the sum is over values of $\boldsymbol{\theta}$ with area weights Δ_k and $\widetilde{\pi}(x_i|\boldsymbol{y}, \boldsymbol{\theta}_k)$ are appropriate approximations to the densities of $x_i|\boldsymbol{y}, \boldsymbol{\theta}$. Clearly the performance of $\widetilde{\pi}(x_i|\boldsymbol{y})$ will depend on the accuracy of $\widetilde{\pi}(x_i|\boldsymbol{y}, \boldsymbol{\theta}_k)$ and on the integration scheme used to compute (2).

Most of the latent Gaussian models in the literature admit conditional independence properties, hence the latent variable $\boldsymbol{x}$ is a Gaussian Markov random field (GMRF). A typical feature of GMRF is that the precision matrix (inverse of the covariance matrix) is sparse. Therefore, approximations for latent GMRF models are based on sparse matrices computations which are much quicker than dense matrices ones. One exception are geostatistical models where the latent variable $\boldsymbol{x}$ admits no conditional independence properties. Anyway, approximate inference is still possible for such models using a different computational approach. In this thesis, approximation methods for latent GMRF models are described in Papers I, II, IV and V, while approximations for geostatistical models are discussed in Paper III.

SUMMARY

This thesis consists of five papers, presented in chronological order. Their content is summarised in this section.

Paper I introduces the approximation tool for latent GMRF models and discusses, in particular, the approximation for the posterior of the hyperparameters $\boldsymbol{\theta}$ in equation (1). It is shown that this approximation is indeed very accurate, as even long MCMC runs cannot detect any error in it. A Gaussian approximation to the density of $x_i|\boldsymbol{\theta}, \boldsymbol{y}$ is also discussed. This appears to give reasonable results and it is very fast to compute. However, slight errors are detected when comparing the approximation with long MCMC runs. These are mostly due to the fact that a possible - skewed density is approximated via a symmetric one. Paper I presents also some details about sparse matrices algorithms.

The core of the thesis is presented in **Paper II**. Here most of the remaining issues present in Paper I are solved. Three different approximation for $x_i|\boldsymbol{\theta}, \boldsymbol{y}$ with different degrees of accuracy and computational costs are described. Moreover, ways to assess the approximation error and considerations about the asymptotical behaviour of the approximations are also discussed. Through a series of examples covering a wide range of commonly used latent GMRF models, the approximations are shown to give extremely accurate results in a fraction of the computing time used by MCMC algorithms.

Paper III applies the same ideas as Paper II to generalised linear mixed models where $\boldsymbol{x}$ represents a latent variable at n spatial sites on a two dimensional domain. Out of these n sites k, with $n \gg k$, are observed through data. The n sites are assumed to be on a regular grid and wrapped on a torus. For the class of models described in Paper III the computations are based on discrete Fourier transform instead of sparse matrices. Paper III illustrates also how marginal likelihood $\pi(\boldsymbol{y})$ can be approximated, provides approximate strategies for Bayesian outlier detection and perform approximate evaluation of spatial experimental design.

Paper IV presents yet another application of the ideas in Paper II. Here approximate techniques are used to do inference on multivariate stochastic volatility models, a class of

models widely used in financial applications. Paper IV discusses also problems deriving from the increased dimension og the parameter vector $\boldsymbol{\theta}$, a condition which makes all numerical integration more computationally intensive. Different approximations for the posterior marginals of the parameters $\boldsymbol{\theta}$, $\pi(\theta_i|\boldsymbol{y})$, are also introduced. Approximations to the marginal likelihood $\pi(\boldsymbol{y})$ are used in order to perform model comparison.

Finally, **Paper V** is a manual for a program, named `inla` which implements all approximations described in Paper II. A large series of worked out examples, covering many well known models, illustrate the use and the performance of the `inla` program. This program is a valuable instrument since it makes most of the Bayesian inference techniques described in this thesis easily available for everyone.

REFERENCES

Diggle, P. J., Ribeiro Jr., P. J., and Christensen, O. F. (2003). An introduction to model-based Geostatistics. In Møller, J., editor, *Spatial Statistics and Computational Methods*, Lecture Notes in Statistics; 173, pages 43–86. Springer-Verlag, Berlin.

Diggle, P. J., Tawn, J. A., and Moyeed, R. A. (1998). Model-based geostatistics (with discussion). *Journal of the Royal Statistical Society, Series C*, 47(3):299–350.

Durbin, J. and Koopman, S. J. (1997). Monte Carlo maximum likelihood estimation for non-Gaussian state space models. *Biometrika*, 84(3):669–684.

Hastie, T. J. and Tibshirani, R. J. (1990). *Generalized Additive Models*, volume 43 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Kammand, E. E. and Wand, M. P. (2003). Geoadditive models. *Journal of Royal Statistical Society C*, 52:1–18.

Knorr-Held, L. and Rue, H. (2002). On block updating in Markov random field models for disease mapping. *Scandinavian Journal of Statistics*, 29(4):597–614.

Lin, X. and Zhang, D. (1999). Inference for generalized additive mixed models by using smoothing splines. *Journal of Royal Statistical Society B*, 61:381–400.

Møller, J., Syversveen, A. R., and Waagepetersen, R. P. (1998). Log Gaussian Cox processes. *Scandinavian Journal of Statistics*, 25:451–482.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*, volume 104 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Tierney, L. and Kadane, J. B. (1986). Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86.

Yu, J. and Mayer, R. (2006). Multivariate stochastic volatility models: Bayesian estimation and models comparison. *Econometric Reviews*, 25.

Paper I

Approximate Bayesian Inference for Hierarchical Gaussian Markov Random Fields Model.

Approximate Bayesian Inference for Hierarchical Gaussian Markov Random Fields Models

Håvard Rue and Sara Martino
Department of Mathematical Sciences
NTNU, Norway

Abstract

Many commonly used models in statistics can be formulated as (Bayesian) hierarchical Gaussian Markov random field models. These are characterised by assuming a (often large) Gaussian Markov random field (GMRF) as the second stage in the hierarchical structure and a few hyperparameters at the third stage. Markov chain Monte Carlo is the common approach for Bayesian inference in such models. The variance of the Monte Carlo estimates is $\mathcal{O}_p(M^{-1/2})$ where M is the number of samples in the chain so, in order to obtain precise estimates of marginal densities, say, we need M to be very large.

Inspired by the fact that often one-block and independence samplers can be constructed for hierarchical GMRF models, we will in this work investigate whether MCMC is really needed to estimate marginal densities, which often is the goal of the analysis. By making use of GMRF-approximations, we show by typical examples that marginal densities can indeed be very precisely estimated by deterministic schemes. The methodological and practical consequence of these findings are indeed positive. We conjecture that for many hierarchical GMRF-models there is really no need for MCMC based inference to estimate marginal densities. Further, by making use of numerical methods for sparse matrices the computational costs of these deterministic schemes are nearly instant compared to the MCMC alternative. In particular, we discuss in detail the issue of computing marginal variances for GMRFs.

Keywords: Approximate Bayesian inference, Cholesky triangle, Conditional auto-regressions, Gaussian Markov random fields, Hierarchical GMRF-models, Laplace-approximation, Marginal variances for GMRFs, Numerical methods for sparse matrices.

1 Introduction

A Gaussian Markov random field (GMRF) $\boldsymbol{x} = \{x_i : i \in \mathcal{V}\}$ is a $n = |\mathcal{V}|$-dimensional Gaussian random vector with additional conditional independence, or Markov properties. Assume that $\mathcal{V} = \{1, \ldots, n\}$. The conditional independence properties can be represented using an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V}$ and edges $\mathcal{E}$. Two nodes, x_i and x_j, are conditional independent given the remaining elements in $\boldsymbol{x}$, if and only if $\{i, j\} \notin \mathcal{E}$. Then, we say that $\boldsymbol{x}$ is a GMRF with respect to $\mathcal{G}$. The edges in $\mathcal{E}$ are in one-to-one correspondence with the non-zero elements of the precision matrix of $\boldsymbol{x}$, $\boldsymbol{Q}$, in the sense that $\{i, j\} \in \mathcal{E}$ if and only if $Q_{ij} \neq 0$ for $i \neq j$. When $\{i, j\} \in \mathcal{E}$ we say that i and j are neighbours, which we denote by $i \sim j$.

GMRFs are also known as conditional auto-regressions (CARs) following seminal work of Besag (1974, 1975). GMRFs (and their intrinsic versions) have a broad use in statistics, with important applications in structural time-series analysis, analysis of longitudinal and survival data, graphical models, semiparametric regression and splines, image analysis and spatial statistics. For references and examples, see Rue and Held (2005, Ch. 1).

One of the main areas of application for GMRFs is that of (Bayesian) hierarchical models. A hierarchical model is characterised by several stages of observables and parameters. The first stage, typically, consists of distributional assumptions for the observables conditionally on latent parameters. For example if we observe a time series of counts $\boldsymbol{y}$, we may assume, for y_i, $i \in \mathcal{D} \subset \mathcal{V}$ a Poisson distribution with unknown mean λ_i. Given the parameters of the observation model, we often assume the observations to be conditionally independent. The second stage consists of a prior model for the latent parameters λ_i or, more often, for a particular function of them. For example, in the Poisson case we can choose an exponential link and model the random variables $x_i = \log(\lambda_i)$. At this stage GMRFs provide a flexible tool to model the dependence between the latent parameters and thus, implicitly, the dependence between the observed data. This dependence can be of various kind, such as temporal, spatial, or even spatiotemporal. The third stage consists of prior distributions for the unknown hyperparameters $\boldsymbol{\theta}$. These are typically precision parameters in the GMRF. The posterior of interest is then

$$\pi(\boldsymbol{x}, \boldsymbol{\theta} \mid \boldsymbol{y}) \propto \pi(\boldsymbol{x} \mid \boldsymbol{\theta}) \pi(\boldsymbol{\theta}) \prod_{i \in \mathcal{D}} \pi(y_i \mid x_i). \tag{1}$$

Most hierarchical GMRF-models can be written in this form. If there are unknown parameters also in the likelihood, then also the last term in (1) depends on $\boldsymbol{\theta}$. Such an extension makes only a slight notational difference in the following.

The main goal is often to compute posterior marginals, like

$$\pi(x_i \mid \boldsymbol{y}) = \int_{\boldsymbol{\theta}} \int_{\boldsymbol{x}_{-i}} \pi(\boldsymbol{x}, \boldsymbol{\theta} \mid \boldsymbol{y}) \, d\boldsymbol{x}_{-i} \, d\boldsymbol{\theta} \tag{2}$$

for each i and (sometimes also) posterior marginals for the hyperparameters θ_j. Since analytical integration is usually not possible for the posterior $\pi(\boldsymbol{x}, \boldsymbol{\theta} | \boldsymbol{y})$, it is

common to use MCMC-based inference to estimate the posterior marginals. These marginals can then be used to compute marginal expectations of various statistics. Although single-site schemes, updating each element of $(\boldsymbol{x}, \boldsymbol{\theta})$ individually, are always possible, they may converge slowly due to the hierarchical structure of the problem. We refer to Rue and Held (2005, Ch. 4) for further discussion. (In some cases reparametrisation may solve the convergence problem due to the hierarchical structure (Gelfand et al., 1995; Papaspiliopoulos et al., 2003), but see also Wilkinson (2003).) In the case of disease mapping, Knorr-Held and Rue (2002) discuss various blocking strategies for updating all the unknown variables to improve the convergence, and Rue and Held (2005, Ch. 4) develop these ideas further. Even if using blocking strategies improves the convergence, MCMC techniques require a large number of samples to achieve a precise estimate. In this paper we propose a deterministic alternative to MCMC based inference which has the advantage of being computed almost instant and which, in our examples, proves to be quite accurate. The key for fast computing time lies in the sparseness of the precision matrix $\boldsymbol{Q}$ due to the Markov properties in the GMRFs. This characteristic allows the use of efficient algorithms and, as explained in Section 2, makes it possible to compute marginal variances without the need to invert $\boldsymbol{Q}$.

One way to introduce our approximation technique is to start from the blocking strategies proposed in Knorr-Held and Rue (2002) and Rue and Held (2005, Ch. 4). The main idea behind these is to make use of the fact that the full conditional for the zero mean GMRF $\boldsymbol{x}$,

$$\pi(\boldsymbol{x} \mid \boldsymbol{\theta}, \boldsymbol{y}) \propto \exp\left(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \sum_{i\in\mathcal{D}} \log \pi(y_i|x_i)\right) \tag{3}$$

can often be well approximated with a Gaussian distribution, by matching the mode and the curvature at the mode. The resulting approximation will then be

$$\widetilde{\pi}(\boldsymbol{x} \mid \boldsymbol{\theta}, \boldsymbol{y}) \propto \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T(\boldsymbol{Q} + \mathrm{diag}(\boldsymbol{c}))(\boldsymbol{x}-\boldsymbol{\mu})\right) \tag{4}$$

where $\boldsymbol{\mu}$ is the mode of $\pi(\boldsymbol{x} \mid \boldsymbol{\theta}, \boldsymbol{y})$. Note that $\boldsymbol{\mu}$ and $\boldsymbol{Q}$ (and then (4)) depend on $\boldsymbol{\theta}$ but we suppress the dependence on $\boldsymbol{\theta}$ to simplify the notation. The terms of the vector $\boldsymbol{c}$ are due to the second order terms in the Taylor expansion of $\sum \log \pi(y_i|x_i)$ at the modal value $\boldsymbol{\mu}$, and these terms are zero for the nodes not directly observed through the data. We call the approximation in (4) the GMRF-approximation. The GMRF-approximation is also a GMRF with respect to the graph $\mathcal{G}$ since, by assumption, each y_i depends only on x_i, a fact that is important computationally.

Following Knorr-Held and Rue (2002) and Rue and Held (2005, Ch. 4), we can often construct a one-block sampler for $(\boldsymbol{x}, \boldsymbol{\theta})$, which proposes the new candidate $(\boldsymbol{x}', \boldsymbol{\theta}')$ by

$$\boldsymbol{\theta}' \sim q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}), \quad \text{and} \quad \boldsymbol{x}' \sim \widetilde{\pi}(\boldsymbol{x} \mid \boldsymbol{\theta}', \boldsymbol{y}) \tag{5}$$

and then accept or reject $(\boldsymbol{x}', \boldsymbol{\theta}')$ jointly. This one-block algorithm, is made possible, in practise, by the outstanding computational properties of GMRFs through

numerical algorithms for sparse matrices (Rue, 2001; Rue and Held, 2005). GMRFs of size up to 10^5 are indeed tractable.

In those cases where the dimension of $\boldsymbol{\theta}$ is small (less than three, say) it is possible to derive an independence sampler by reusing (4) to build an approximation of the marginal posterior for $\boldsymbol{\theta}$. The starting point is the identity

$$\pi(\boldsymbol{\theta} \mid \boldsymbol{y}) = \frac{\pi(\boldsymbol{x}, \boldsymbol{\theta} \mid \boldsymbol{y})}{\pi(\boldsymbol{x} \mid \boldsymbol{\theta}, \boldsymbol{y})}. \tag{6}$$

By approximating the denominator via expression (4) and evaluating the right-hand side at the modal value for $\boldsymbol{x}$ (for each $\boldsymbol{\theta}$), we obtain an approximation for the marginal posterior, which we denote by $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. This approximation is in fact the Laplace-approximation suggested by Tierney and Kadane (1986), who showed that its relative error is $\mathcal{O}(N^{-3/2})$ after renormalisation. (Here, N is the number of observations.) The approximation $\widetilde{\pi}(\boldsymbol{\theta'}|\boldsymbol{y})$ then replaces $q(\boldsymbol{\theta'}|\boldsymbol{\theta})$ in the one-block algorithm above. The independence sampler uses the approximation

$$\widetilde{\pi}(\boldsymbol{x}, \boldsymbol{\theta} \mid \boldsymbol{y}) = \widetilde{\pi}(\boldsymbol{\theta} \mid \boldsymbol{y})\ \widetilde{\pi}(\boldsymbol{x} \mid \boldsymbol{\theta}, \boldsymbol{y}). \tag{7}$$

A natural question arises here. If we can use $\widetilde{\pi}(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ to construct an independence sampler to explore $\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$, why not just compute approximations to the marginals from $\widetilde{\pi}(\boldsymbol{x}, \boldsymbol{\theta} \mid \boldsymbol{y})$ directly?

Since (4) is Gaussian, it is, theoretically, always possible to (approximately) compute the marginal for the x_i's as

$$\widehat{\widetilde{\pi}}(x_i \mid \boldsymbol{y}) = \sum_j \widetilde{\pi}(x_i \mid \boldsymbol{\theta}_j, \boldsymbol{y})\ \widetilde{\pi}(\boldsymbol{\theta}_j \mid \boldsymbol{y})\ \Delta_j \tag{8}$$

by simply summing out $\boldsymbol{\theta}$ by some numerical integration rule where Δ_j is the weight associated with $\boldsymbol{\theta}_j$. The approximated marginal posterior $\widehat{\widetilde{\pi}}(x_i \mid \boldsymbol{y})$ is a mixture of Gaussians where the weights, mean and variances, are computed from (7). However, the dimension of $\boldsymbol{x}$ is usually large, thus obtaining the marginal variances for $x_i|\boldsymbol{\theta}, \boldsymbol{y}$ is computationally intensive (recall that only the precision matrix $\boldsymbol{Q}$ is explicitly known). Therefore the marginals in (8) are, in practise, possible to compute only for GMRFs since in, these cases, efficient computations are possible. A recursion algorithm to efficiently compute marginal variances for GMRFs is described in Section 2.

Although any MCMC algorithm will guarantee the correct answer in the end, the question is what happens in finite time. The Monte Carlo error is $\mathcal{O}_p(M^{-1/2})$ where M is the (effective) number of samples, hence, the strength of the MCMC approach is to provide rough (near) unbiased estimates rather quickly, on the other side, precise estimates may take unreasonable long time. Any (deterministic) approximated inference can in fact compete with a MCMC approach, as long as its squared "bias", or error, is comparable with the Monte Carlo error. The most interesting aspect of approximation (8), is that it can be computed almost instantly compared to the time any MCMC algorithm will have to run to obtain any decent accuracy.

The aim of this paper is to investigate how accurate (8) is for some typical examples of hierarchical GMRF models. In Section 3 we report some experiments using models for disease mapping on a varying scale of difficulty. We compare the marginals of interest as approximated by (8) and as estimated from very long MCMC runs. The results are very positive. Before presenting the examples, we will, in Section 2, discuss how to efficiently compute marginal variances needed in expression (8) for GMRFs. This Section also explains (implicit) why fast computations of GMRFs are possible using numerical methods for sparse matrices. Section 2 is unavoidably somewhat technical, but it is not necessary to appreciate the results in Section 3. We end with a discussion in Section 4.

2 Computing marginal variances for a GMRF

GMRFs are nearly always specified by their precision matrix $\boldsymbol{Q}$ meaning that the covariance matrix, $\boldsymbol{\Sigma} = \boldsymbol{Q}^{-1}$ is only implicitly known. Although we can formally invert $\boldsymbol{Q}$, the dimension n is typically large ($10^3 - 10^5$) so inverting $\boldsymbol{Q}$ directly is costly and inconvenient. In this section we discuss a simple and fast algorithm to compute marginal variances, applicable for GMRFs with large dimension. The starting point is the not-well-known matrix identity which appeared in a IEEE conference proceedings (Takahashi et al., 1973). In our setting, the identity is as follows. Let $\boldsymbol{L}\boldsymbol{L}^T = \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}^T$ be the Cholesky-decomposition of $\boldsymbol{Q}$ where $\boldsymbol{L} = \boldsymbol{V}\boldsymbol{D}^{1/2}$ is the (lower triangular) Cholesky triangle, $\boldsymbol{D}$ is a diagonal matrix and $\boldsymbol{V}$ is a lower triangular matrix with ones on the diagonal. Then

$$\boldsymbol{\Sigma} = \boldsymbol{D}^{-1}\boldsymbol{V}^{-1} + (\boldsymbol{I} - \boldsymbol{V}^T)\boldsymbol{\Sigma}. \tag{9}$$

(The proof is simple; Since $\boldsymbol{Q}\boldsymbol{\Sigma} = \boldsymbol{I}$ then $\boldsymbol{V}\boldsymbol{D}\boldsymbol{V}^T\boldsymbol{\Sigma} = \boldsymbol{I}$. Multiplying from left with $(\boldsymbol{V}\boldsymbol{D})^{-1}$ and then adding $\boldsymbol{\Sigma}$ on both sides gives (9) after rearrangement.) A close look at (9) will reveal that the upper triangle of (9) defines recursions for Σ_{ij} (Takahashi et al., 1973), and this provide the basis for fast computations of the marginal variances of x_1 to x_n.

However, the identity (9) gives little insight in how Σ_{ij} depends on the elements of $\boldsymbol{Q}$ and on the graph $\mathcal{G}$. We will therefore, in Section 2.1, derive the recursions defined in (9) "statistically", starting from a simulation algorithm for GMRFs and using the relation between $\boldsymbol{Q}$ and its Cholesky triangle given by the global Markov property. We use the same technique to prove Theorem 1, given in Section 2.1. This theorem locates a set of indexes for which the recursions are to be solved to obtain the marginal variances. A similar result was also given in Takahashi et al. (1973), see also Erisman and Tinney (1975). We also generalise the recursions to compute marginal variances for GMRFs defined with additional soft and hard linear constraints, for example under a sum-to-zero constraint. Practical issues appearing when implementing the algorithm using the Cholesky triangle of $\boldsymbol{Q}$ computed using sparse matrix libraries, are also discussed.

The recursions for Σ_{ij} are applicable to a GMRF with respect to any graph $\mathcal{G}$ and generalise the well known (fixed-interval) Kalman recursions for smoothing applicable for dynamic models. The computational effort needed to solve the recursions depends on both the neighbourhood structure in $\mathcal{G}$ and the size n. For typical spatial applications, the cost is $\mathcal{O}(n \log(n)^2)$ when the Cholesky triangle of $\boldsymbol{Q}$ is available.

2.1 The Recursions

The Cholesky triangle $\boldsymbol{L}$ (of $\boldsymbol{Q}$) is the starting point both for producing (unconditional and conditional) samples from a zero mean GMRF and for evaluating the log-density for any configuration. Refer to Rue and Held (2005, Ch. 2) for algorithms and further details. In short, unconditional samples are found as the solution of $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ where $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. The log-density is computed using that $\log|\boldsymbol{Q}| = 2\sum_i \log L_{ii}$.

Since the solution of $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ is a sample from a zero mean GMRF with precision matrix $\boldsymbol{Q}$, we obtain that

$$x_i \mid x_{i+1}, \ldots, x_n \;\sim\; \mathcal{N}(-\frac{1}{L_{ii}} \sum_{k=i+1}^{n} L_{ki} x_k, 1/L_{ii}^2), \quad i = n, \ldots, 1. \tag{10}$$

Eq. (10) provides a sequential representation of the GMRF backward in "time" i, as

$$\pi(\boldsymbol{x}) = \prod_{i=n}^{1} \pi(x_i \mid x_{i+1}, \ldots, x_n).$$

Let $\boldsymbol{L}_{i:n}$ be the lower-right $(n-i-1) \times (n-i-1)$ submatrix of $\boldsymbol{L}$. It follows directly from $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ that $\boldsymbol{L}_{i:n}\boldsymbol{L}_{i:n}^T$ is the precision matrix of $\boldsymbol{x}_{i:n} = (x_i, \ldots, x_n)^T$. The non-zero pattern in $\boldsymbol{L}$ is important for the recursions, see Rue and Held (2005, Ch. 2) for further details about the relation between $\boldsymbol{Q}$ and $\boldsymbol{L}$. Zeros in the i'th column of $\boldsymbol{L}$, $\{L_{ki}, k = 1, \ldots, n\}$, relates directly to the conditional independence properties of $\pi(\boldsymbol{x}_{i:n})$. For $i < k$, we have

$$-\frac{1}{2}\boldsymbol{x}_{i:n}^T \boldsymbol{L}_{i:n}\boldsymbol{L}_{i:n}^T \boldsymbol{x}_{i:n} = -x_i x_k L_{ii} L_{ki} + \text{remaining terms}$$

hence $L_{ki} = 0$ means that x_i and x_k are conditional independent given $x_{i+1}, \ldots, x_{k-1}, x_{k+1}, \ldots, x_n$. This is similar to the fact that $Q_{ij} = 0$ means that x_i and x_j are conditional independent given the remaining elements of $\boldsymbol{x}$. To ease the notation, define the set

$$F(i,k) = \{i+1, \ldots, k-1, k+1, \ldots, n\}, \quad 1 \le i \le k \le n$$

which is the future of i except k. Then for $i < k$

$$x_i \perp x_k \mid \boldsymbol{x}_{F(i,k)} \iff L_{ki} = 0. \tag{11}$$

Unluckily it is not easy to verify that $x_i \perp x_k \mid \boldsymbol{x}_{F(i,k)}$ without computing $\boldsymbol{L}$ and checking if $L_{ki} = 0$ or not. However, the global Markov property provides a sufficient condition for L_{ki} to be zero. If i and $k > i$ are separated by $F(i,k)$ in $\mathcal{G}$, then $x_i \perp x_k \mid \boldsymbol{x}_{F(i,k)}$ and $L_{ki} = 0$. This sufficient criterion depends only on the graph $\mathcal{G}$. If we use this to conclude that $L_{ki} = 0$, then this is true for all $\boldsymbol{Q} > 0$ with fixed graph $\mathcal{G}$. In particular, if $k \sim i$ then L_{ki} is non-zero in general. This imply that the Cholesky triangle is in general more dense than the lower triangle of $\boldsymbol{Q}$.

To obtain the recursions for $\boldsymbol{\Sigma} = \boldsymbol{Q}^{-1}$, we note that (10) implies that

$$\Sigma_{ij} = \delta_{ij}/L_{ii}^2 - \frac{1}{L_{ii}} \sum_{k \in \mathcal{I}(i)} L_{ki}\Sigma_{kj}, \quad j \geq i,\ i = n, \ldots, 1, \tag{12}$$

where $\mathcal{I}(i)$ includes those k larger than i and where L_{ki} is non-zero,

$$\mathcal{I}(i) = \{k > i \ :\ L_{ki} \neq 0\} \tag{13}$$

and δ_{ij} is one if $i = j$ and zero otherwise. Note that (12) equals the upper triangle of (9). We can compute all covariances directly using (12) but the order of the indexes are important. In the outer loop i runs from n to 1 and the inner loop j runs from n to i. The first and last computed covariance is then Σ_{nn} and Σ_{11}, respectively.

It is possible to derive a similar set of equations to (12) which relates covariances to elements of $\boldsymbol{Q}$ instead of elements of $\boldsymbol{L}$, see Besag (1981). However, these equations does not define recursions.

Example 1 *Let $n = 3$, $\mathcal{I}(1) = \{2, 3\}$, $\mathcal{I}(2) = \{3\}$, then (12) gives*

$$\begin{aligned}
\Sigma_{33} &= \frac{1}{L_{33}^2} & \Sigma_{23} &= -\frac{1}{L_{22}}\left(L_{32}\Sigma_{33}\right) \\
\Sigma_{22} &= \frac{1}{L_{22}^2} - \frac{1}{L_{22}}\left(L_{32}\Sigma_{32}\right) & \Sigma_{13} &= -\frac{1}{L_{11}}\left(L_{21}\Sigma_{23} + L_{31}\Sigma_{33}\right) \\
\Sigma_{12} &= -\frac{1}{L_{11}}\left(L_{21}\Sigma_{22} + L_{31}\Sigma_{32}\right) & \Sigma_{11} &= \frac{1}{L_{11}^2} - \frac{1}{L_{11}}\left(L_{21}\Sigma_{21} + L_{31}\Sigma_{31}\right)
\end{aligned}$$

where we also need to use that $\boldsymbol{\Sigma}$ is symmetric.

Our aim is to compute the marginal variances $\Sigma_{11}, \ldots, \Sigma_{nn}$. In order to do so, we need to compute Σ_{ij} (or Σ_{ji}) for all ij in some set $\mathcal{S}$, as evident from (12). Let the elements in $\mathcal{S}$ be unordered, meaning that if $ij \in \mathcal{S}$ then $ji \in \mathcal{S}$. If the recursions can be solved by only computing Σ_{ij} for all $ij \in \mathcal{S}$ we say that the recursions are solvable using $\mathcal{S}$, or simply that $\mathcal{S}$ is solvable. A sufficient condition for a set $\mathcal{S}$ to be solvable is that

$$ij \in \mathcal{S} \text{ and } k \in \mathcal{I}(i) \implies kj \in \mathcal{S} \tag{14}$$

and that $ii \in \mathcal{S}$ for $i = 1, \ldots, n$. Of course $\mathcal{S} = \mathcal{V} \times \mathcal{V}$ is such a set, but we want $|\mathcal{S}|$ to be minimal to avoid unnecessary computations. Such a minimal set depends,

however, on the numerical values in $\boldsymbol{L}$ or $\boldsymbol{Q}$ implicitly. Denote by $\mathcal{S}(\boldsymbol{Q})$ a minimal set for a certain precision matrix $\boldsymbol{Q}$. The following result identifies a solvable set $\mathcal{S}^*$ containing the union of $\mathcal{S}(\boldsymbol{Q})$ for all $\boldsymbol{Q} > 0$ with a fixed graph $\mathcal{G}$.

Theorem 1 *The union of $\mathcal{S}(\boldsymbol{Q})$ for all $\boldsymbol{Q} > 0$ with fixed graph $\mathcal{G}$, is a subset of*

$$\mathcal{S}^* = \{ij \in \mathcal{V} \times \mathcal{V} \ : \ j \geq i, \ i \text{ and } j \text{ are not separated by } F(i,j)\}$$

and the recursions in (12) are solvable using $\mathcal{S}^*$.

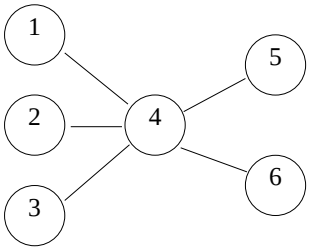
Proof. *To prove the theorem we have to show that $\mathcal{S}^*$ is solvable and that it contains the union of $\mathcal{S}(\boldsymbol{Q})$ for all $\boldsymbol{Q} > 0$ with fixed graph $\mathcal{G}$. To verify that the recursions are solvable using $\mathcal{S}^*$, first note that $ii \in \mathcal{S}^*$, for $i = 1, \ldots, n$ since i and i are not separated by $F(i,i)$. The global Markov property ensures that if $ij \notin \mathcal{S}^*$ then $L_{ji} = 0$ for all $\boldsymbol{Q} > 0$ with fixed graph $\mathcal{G}$. Using this feature we can replace $\mathcal{I}(i)$ with $\mathcal{I}^*(i) = \{k > i : ik \in \mathcal{S}^*\}$ in (14). This is legal since $\mathcal{I}(i) \subseteq \mathcal{I}^*(i)$ and the difference between the two sets only identifies terms L_{ki} which are zero. Then, we have to show that*

$$ij \in \mathcal{S}^* \text{ and } ik \in \mathcal{S}^* \implies kj \in \mathcal{S}^* \tag{15}$$

Eq. (15) is trivially true for $i \leq k = j$. Fix now $i < k < j$. Then $ij \in \mathcal{S}^*$ says that there exists a path $i, i_1, \ldots, i_n, j$, where $i_1, \ldots, i_n$ are all smaller than i, and $ik \in \mathcal{S}^*$ says that there exists a path $i, i'_1, \ldots, i'_{n'}, k$, where $i'_1, \ldots, i'_{n'}$ are all smaller than i. Then there is a path from k to i and from i to j where all nodes are less than or equal to i. Since $i < k$ then all the nodes in the two paths are less than k. Hence, there is a path from k and j where all nodes are less than k. This means that k and j are not separated by $F(k,j)$, so $kj \in \mathcal{S}^*$. Finally, since $\mathcal{S}^*$ only depends on $\mathcal{G}$, it must contain all $\mathcal{S}(\boldsymbol{Q})$ since each $\mathcal{S}(\boldsymbol{Q})$ is minimal, and therefore contains their union too. ■

An alternative interpretation of $\mathcal{S}^*$, is that it identifies only from the graph $\mathcal{G}$, all possible non-zero elements in $\boldsymbol{L}$. Some of these might turn out to be zero depending on the conditional independence properties of the marginal density for $\boldsymbol{x}_{i:n}$ for $i = n, \ldots, 1$, see (11). In particular, if $j \sim i$ and $j > i$ then $ij \in \mathcal{S}^*$. This provides the lower bound for the size of $\mathcal{S}^*$: $|\mathcal{S}^*| \geq n + |\mathcal{E}|$.

Example 2 *Let $\boldsymbol{x} = (x_1, \ldots, x_6)^T$ be a GMRF with respect to the graph*



Then, the set of the possible non-zero terms in $\boldsymbol{L}$ *are*

$$\mathcal{S}^* = \{11, 22, 33, 41, 42, 43, 44, 54, 55, 64, 65, 66\}. \tag{16}$$

The only element in $\mathcal{S}^*$ *where the corresponding element in* $\boldsymbol{Q}$ *is zero, is* 65*, this because* 5 *and* 6 *are not separated by* $F(5,6) = \emptyset$ *in* $\mathcal{G}$ *(due to* 4*), so* $|\mathcal{S}^*| = n + |\mathcal{E}| + 1$*.*

The size of $\mathcal{S}^*$ depends not only on the graph $\mathcal{G}$ but also on the permutation of the vertices in the graph $\mathcal{G}$. It is possible to show that, if the graph $\mathcal{G}$ is decomposable, then there exists a permutation of the vertices, such that $|\mathcal{S}^*| = n + |\mathcal{E}|$ and $\mathcal{S}^*$ *is* the union of $\mathcal{S}(\boldsymbol{Q})$ for all $\boldsymbol{Q} > 0$ with fixed graph $\mathcal{G}$. The typical example is the following.

Example 3 *A homogeneous autoregressive model of order* p *satisfies*

$$x_i \mid x_1, \ldots, x_{i-1} \sim \mathcal{N}(\sum_{j=1}^{p} \phi_j x_{i-j}, 1), \quad i = 1, \ldots, n,$$

for some parameters $\{\phi_j\}$ *where for simplicity we assume that* $x_{-1}, \ldots, x_{-p+1}$ *are fixed. Let* $\{y_i\}$ *be independent Gaussian observations of* x_i *such that* $y_i \sim \mathcal{N}(x_i, 1)$*. Then* $\boldsymbol{x}$ *conditioned on the observations is Gaussian where the precision matrix* $\boldsymbol{Q}$ *is a band-matrix with band-width* p *and* $\boldsymbol{L}$ *is lower triangular with the same bandwidth. When* $\{\phi_j\}$ *are such that* $Q_{ij} \neq 0$ *for all* $|i - j| \leq p$*, then the graph is decomposable. In this case the recursions correspond to the (fixed-interval) smoothing recursions derived from the Kalman filter for (Gaussian) linear state-space models.*

Although the situation is particularly simple for decomposable graphs, most GMRFs are defined with respect to graphs that are not decomposable. This is the case for GMRFs used in spatial or spatio-temporal applications, but also for GMRFs used in temporal models outside the state-space framework. In addition to be able to identify the set $\mathcal{S}^*$ efficiently, we also need to compute the Cholesky triangle $\boldsymbol{L}$. It is important to have efficient algorithms for these tasks as the dimension of GMRFs is typically large. Fortunately, algorithms that compute $\boldsymbol{L}$ efficiently also minimise (approximately) the size of $\mathcal{S}^*$ and then also the cost of solving the recursions. We return to this and other practical issues in Section 2.3, after discussing how to compute marginal variances for GMRFs with additional linear constraints.

2.2 Correcting for hard and soft linear constraints

We will now demonstrate how we can correct the marginal variances computed in (12) to account for additional linear constraints, for example a simple sum-to-zero constraint. Let $\boldsymbol{A}$ be a $k \times n$ matrix of rank k. The goal is now to compute the marginal variances of the GMRF under the linear constraint $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{e}$. If $\boldsymbol{e}$ is fixed we denote the constraint as *hard*, and if $\boldsymbol{e}$ is a realisation of $\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{e}}, \boldsymbol{\Sigma}_{\boldsymbol{e}})$, $\boldsymbol{\Sigma}_{\boldsymbol{e}} > 0$, we denote the constraint as *soft*.

A constrained GMRF is also a GMRF, meaning that the recursions (12) are still valid using the Cholesky triangle for the constrained GMRF. Since linear constraints destroy the sparseness of the precision matrix they will not allow fast computation of the marginal variances. However, the covariance matrix under hard linear constraints, $\widetilde{\boldsymbol{\Sigma}}$, relates to the unconstrained covariance matrix $\boldsymbol{\Sigma}$ as

$$\widetilde{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma} - \boldsymbol{Q}^{-1}\boldsymbol{A}^T \left(\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^T\right)^{-1} \boldsymbol{A}\boldsymbol{Q}^{-1}. \tag{17}$$

There is a similar relation with a soft constraint (Rue and Held, 2005, Ch. 2). In the following we assume a hard constraint. It is evident from (17) that

$$\widetilde{\Sigma}_{ii} = \Sigma_{ii} - \left(\boldsymbol{Q}^{-1}\boldsymbol{A}^T \left(\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^T\right)^{-1} \boldsymbol{A}\boldsymbol{Q}^{-1}\right)_{ii}, \quad i = 1, \ldots, n.$$

Hence, we can compute the diagonal of $\boldsymbol{\Sigma}$ and then correct it to account for the hard constraints. Define the $n \times k$ matrix $\boldsymbol{W}$ as $\boldsymbol{Q}^{-1}\boldsymbol{A}^T$ which is found from solving $\boldsymbol{Q}\boldsymbol{W} = \boldsymbol{A}^T$ for each of the k columns of $\boldsymbol{W}$. As the Cholesky triangle to $\boldsymbol{Q}$ is available, the j'th column of $\boldsymbol{W}$, $\boldsymbol{W}_j$, is found by solving $\boldsymbol{L}\boldsymbol{v} = \boldsymbol{A}_j^T$ and then solving $\boldsymbol{L}^T\boldsymbol{W}_j = \boldsymbol{v}$. We now see that $\widetilde{\Sigma}_{ii} = \Sigma_{ii} - C_{ii}$ where $\boldsymbol{C} = \boldsymbol{W}\left(\boldsymbol{A}\boldsymbol{W}\right)^{-1}\boldsymbol{W}^T$. We only need the diagonal of $\boldsymbol{C}$. Let $\boldsymbol{V} = \boldsymbol{W}(\boldsymbol{A}\boldsymbol{W})^{-1}$, and then $\boldsymbol{C} = \boldsymbol{V}\boldsymbol{W}^T$ and $C_{ii} = \sum_{l=1}^{k} V_{il}W_{il}$. The cost of computing $\boldsymbol{V}$ and $\boldsymbol{W}$ is for large k dominated by factorising the (dense) $k \times k$ matrix $\boldsymbol{A}\boldsymbol{W}$, which is cubic in k. As long as k is not too large it is nearly free to correct for linear soft and hard constraints.

A special case of hard constraint is to condition on a subset, B say, of the nodes in $\mathcal{G}$. This is equivalent to computing the marginal variances for $\boldsymbol{x}_A|\boldsymbol{x}_B$ where $\boldsymbol{x} = (\boldsymbol{x}_A, \boldsymbol{x}_B)$ is a zero mean GMRF. In most cases it is more efficient not to use (17), but utilise that $\boldsymbol{x}_A|\boldsymbol{x}_B$ is a GMRF with precision matrix $\boldsymbol{Q}_{AA}$ and mean $\boldsymbol{\mu}$ given by the solution of $\boldsymbol{Q}_{AA}\boldsymbol{\mu} = -\boldsymbol{Q}_{AB}\boldsymbol{x}_B$. (Note that solving for $\boldsymbol{\mu}$ require only the Cholesky triangle of $\boldsymbol{Q}_{AA}$ which is needed in any case for the recursions.) The marginal variances are then computed using (12), possibly correcting for additional linear constraints using (17).

2.3 Practical issues

Since the precision matrix $\boldsymbol{Q}$ is a sparse matrix we can take advantage of numerical algorithms for sparse symmetric positive definite matrices. Such algorithms are very efficient and make it possible to factorise precision matrices of dimension $10^3 - 10^5$ without too much effort. A major benefit is that these algorithms also minimise (approximately) the size of $\mathcal{S}^*$, and hence the cost of solving the recursions described earlier. Rue (2001) and Rue and Held (2005) discuss numerical algorithms for sparse matrices from a statistical perspective and how to apply them for GMRFs.

An important ingredient in sparse matrix algorithms is to permute the vertices to minimise (approximately) the number of non-zero terms in $\boldsymbol{L}$. The idea is as follows, if L_{ji} is known to be zero, then L_{ji} is not computed. It turns out that the set $\mathcal{S}^*$ is exactly the set of vertices for which L_{ji} is computed, see Rue and Held (2005,

Sec. 2.4.1). A permutation to efficiently compute $\boldsymbol{L}$ minimise (approximately) $|\mathcal{S}^*|$, hence is also an efficient permutation for solving the recursions. However, this implies that we have little control over which Σ_{ij}'s are computed in the recursions, apart from the diagonal and those elements where $i \sim j$.

Permutation schemes based on the idea of nested dissection are particularly useful in statistical applications. The idea is to find a small separating subset that divides the graph into two (roughly) equal parts, label the nodes in the separating set with the highest indexes, and continue recursively. For such a permutation, the computational complexity to compute $\boldsymbol{L}$ for a GMRF on a square $m \times m$ lattice with a local neighbourhood, is $\mathcal{O}(n^{3/2})$ for $n = m^2$. This also gives the optimal complexity in the order sense. The number of possible non-zero terms in $\boldsymbol{L}$ is $\mathcal{O}(n \log(n))$ which corresponds to the size of $\mathcal{S}^*$. The complexity of solving the recursions can be estimated from these numbers. We need to compute $\mathcal{O}(n \log(n))$ covariances, each involving on average $\mathcal{O}(\log(n))$ terms in $\mathcal{I}^*(i)$, which in total gives a cost of $\mathcal{O}(n \log(n)^2)$ operations. For a local GMRF on a $m \times m \times m$ cube with $n = m^3$ the size of $\mathcal{S}^*$ is $\mathcal{O}(n^{4/3})$, and the cost of solving the recursions is then $\mathcal{O}(n^{5/3})$. This cost is dominated by the cost of factorising $\boldsymbol{Q}$, which is $\mathcal{O}(n^2)$.

A practical concern arises when numerical libraries return a list with the non-zero elements in $\boldsymbol{L}$, but the set $\mathcal{S}^*$ or $\mathcal{S}(\boldsymbol{Q})$ is needed by the recursions. In fact, any easily obtainable solvable set $\mathcal{S}(\boldsymbol{Q})^+$, where $\mathcal{S}(\boldsymbol{Q}) \subseteq \mathcal{S}(\boldsymbol{Q})^+ \subseteq \mathcal{S}^*$, is acceptable. A simple approach to obtain *a* $\mathcal{S}(\boldsymbol{Q})^+$ is the following. Let $\mathcal{S}_0 = \{j \geq i \,:\, L_{ji} \neq 0\}$. Traverse the set $\mathcal{S}_0$ with i from n to 1 as the outer loop, and j from n to i such that $ij \in \mathcal{S}_0$. For each ij, check for each $k \in \mathcal{I}(i)$ if $kj \in \mathcal{S}_0$. If this is not true, then add kj to $\mathcal{S}_0$. Repeat this procedure until no changes appear in $\mathcal{S}_0$. By construction, $\mathcal{S}_0 \subseteq \mathcal{S}^*$ and $\mathcal{S}_0$ is solvable, hence we may use $\mathcal{S}(\boldsymbol{Q})^+ = \mathcal{S}_0$. Two iterations are often sufficient to obtain $\mathcal{S}(\boldsymbol{Q})^+$, where the last verify only that $\mathcal{S}_0$ is solvable. Alternatively, $\mathcal{S}^*$ can either computed directly or extracted from an intermediate result in the sparse matrix library, if this is easily accessible.

Needless to say, solving the recursions efficiently requires very careful implementation in an appropriate language, but this is the rule, not the exception when working with sparse matrices. The open-source library `GMRFLib` (Rue and Held, 2005, Appendix B) includes an efficient implementation of the recursions as well as numerous of useful routines for GMRFs. All the examples in Section 3 make extensive use of `GMRFLib`, which can be downloaded from the first author's www-page.

3 Examples

In this section, we will present some results for the approximations for the marginal posteriors computed from (7), and their comparison with estimates obtained from very long MCMC runs. We will restrict ourself to the well-known BYM-model for disease mapping (Section 3.1). The BYM-model is a hierarchical GMRF model with Poisson distributions at the first stage. We will use two different datasets, which

we describe as "easy" (many counts) and "hard" (few counts). The comparison of the marginal posteriors for the hyperparameters (in this case, the precisions) are presented in Section 3.2, while the posterior marginals for the latent GMRF are presented in Section 3.3. In Section 3.4 we present some results for an extended BYM-model, where we include a semi-parametric effect of a covariate and where the latent GMRF has to obey a linear constraint.

Note that the computational speed in the following experiments is not optimal due to rather brute-force approach taken while integrating out the hyperparameters $\boldsymbol{\theta}$. However, this step can be improved considerably, as we discuss in Section 4, while the approximation results themselves remain unaffected.

3.1 The BYM-model for disease mapping

We will now introduce the BYM-model for analysing spatial disease data (Besag et al., 1991). This model is commonly used in epidemiological applications.

The number of incidents y_i, $i = 1, \ldots, N$, of a particular disease is observed over a certain time period in a site of N districts. It is common to assume the observed counts to be conditionally independent and Poisson distributed with mean $e_i \exp(\eta_i)$, where η_i is the log-relative risk and e_i is the expected number of cases computed on some demographic parameters. Further, η_i is decomposed as $\eta_i = u_i + v_i$ where $\boldsymbol{u} = \{u_i\}$ is a spatially structured component and $\boldsymbol{v}$ is an unstructured component. An intrinsic GMRF of the following form is often assumed for the spatially structured component,

$$\pi(\boldsymbol{u} \mid \kappa_{\boldsymbol{u}}) \propto \kappa_{\boldsymbol{u}}^{(n-1)/2} \exp\left(-\frac{\kappa_{\boldsymbol{u}}}{2} \sum_{i \sim j} (u_i - u_j)^2\right) \tag{18}$$

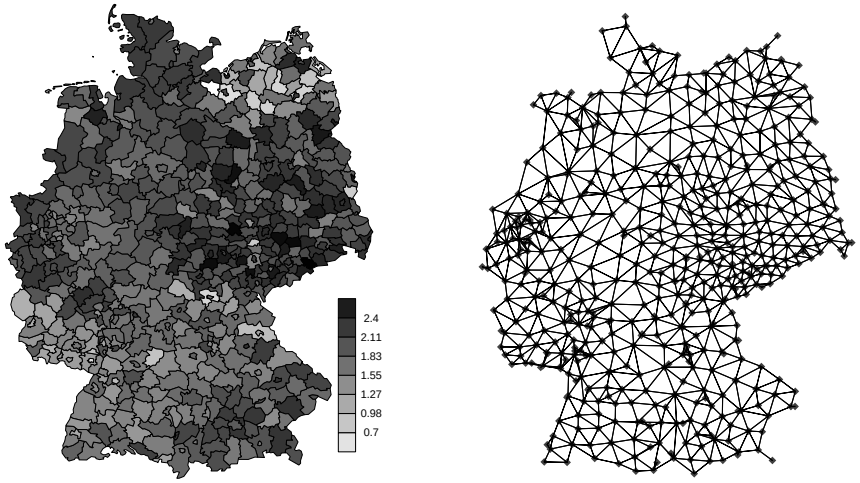
where $\kappa_{\boldsymbol{u}}$ is the unknown precision parameter. Two districts i and j are defined to be neighbours, $i \sim j$, if they are adjacent. Further, $\boldsymbol{v}$ are independent zero mean normals with unknown precision parameter $\kappa_{\boldsymbol{v}}$. The precisions are (most commonly) assigned independent Gamma priors with fixed parameters.

The BYM-model is of course a hierarchical GMRF model, with $y_i \sim \text{Po}(e_i \exp(\eta_i))$ at the first stage. At the second stage the GMRF is $\boldsymbol{x} = (\boldsymbol{\eta}^T, \boldsymbol{u}^T)^T$. The unknown precisions $\boldsymbol{\kappa} = (\kappa_{\boldsymbol{u}}, \kappa_{\boldsymbol{v}})$ constitute the third stage. Note that we have reparametrised the GMRF using $\boldsymbol{x} = (\boldsymbol{\eta}^T, \boldsymbol{u}^T)^T$ instead of $\boldsymbol{x} = (\boldsymbol{v}^T, \boldsymbol{u}^T)^T$, in this way some of the nodes in the graph, namely the $\boldsymbol{\eta}$'s, are observed through the data $\boldsymbol{y}$. The posterior of interest if therefore

$$\pi(\boldsymbol{x}, \boldsymbol{\kappa} \mid \boldsymbol{y}) \propto \kappa_{\boldsymbol{v}}^{N/2} \kappa_{\boldsymbol{u}}^{(N-1)/2} \exp\left(-\frac{1}{2} \boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x}\right) \exp\left(\sum_{i=1}^{N} y_i x_i - e_i \exp(x_i)\right) \pi(\boldsymbol{\kappa}) \tag{19}$$

The $2N \times 2N$ precision matrix for the GMRF, $\boldsymbol{Q}$ is

$$\boldsymbol{Q} = \begin{pmatrix} \kappa_{\boldsymbol{v}} \boldsymbol{I} & -\kappa_{\boldsymbol{v}} \boldsymbol{I} \\ -\kappa_{\boldsymbol{v}} \boldsymbol{I} & \kappa_{\boldsymbol{u}} \boldsymbol{R} + \kappa_{\boldsymbol{v}} \boldsymbol{I} \end{pmatrix} \tag{20}$$



(a) (b)

Figure 1: (a) The standardised mortality ratio y_i/e_i for the oral cavity cancer counts in Germany (1986–1990). (b) The graph associated with (a) where two districts are neighbours if and only if they are adjacent.

where $\boldsymbol{R}$ is the so-called structure matrix for the spatial term, R_{ii} is the number of neighbours to district i, and $R_{ij} = -1$ if $i \sim j$ (district i and j are adjacent) and zero otherwise. We set the priors of the unknown precisions to be independent and Gamma(a, b) distributed with a/b as the expected value. The values of a and b are specified later.

The two datasets we will consider in Section 3.2 and Section 3.3 are classified as the Easy-case and the Hard-case.

Easy-case The observed oral cavity cancer mortality for males in Germany (1986–1990) was previously analysed by Knorr-Held and Raßer (2000). The data have an average observed count of 28.4, median of 19, and the first and third quantile are 9 and 33. For such high counts the Poisson distribution is not too far away from a Gaussian. The observed standardised mortality ratio for the different districts of Germany are shown in Figure 1a. The corresponding graph is displayed in Figure 1b. It has $n = 544$ nodes with average 5.2, minimum 1, and maximum 11 neighbours. The parameters in the prior for the precisions are $a = 1$ and $b = 0.01$ following Rue and Held (2005, Ch. 4).

Hard-case The observed Insulin dependent Diabetes Mellitus in Sardinia. These data were previously analysed by Bernardinelli et al. (1997) and also used by Knorr-Held and Rue (2002) as a challenging case. The graph is similar to the one in Figure 1b, and has $n = 366$ nodes with average 5.4, minimum 1 and maximum 13 neighbours. This is a sparse dataset with a total of 619 cases and median of 1. For such low counts the Poisson distribution is quite different from a Gaussian. The parameters in the prior for the precisions are $a = 1$ and $b = 0.0005$ for $\kappa_{\boldsymbol{u}}$, and $a = 1$ and $b = 0.00025$ for $\kappa_{\boldsymbol{v}}$ following Knorr-Held and Rue (2002).

3.2 Approximating $\pi(\boldsymbol{\theta}|\boldsymbol{y})$

Our first task is to approximate the marginal posteriors for the hyperparameters $\log \kappa_{\boldsymbol{u}}$ and $\log \kappa_{\boldsymbol{v}}$, for the Easy-case and the Hard-case.

The joint marginal posterior for $\boldsymbol{\theta} = (\log \kappa_{\boldsymbol{u}}, \log \kappa_{\boldsymbol{v}})$ was estimated using the approximation to (6). This means using the GMRF-approximation (4) (depending on $\boldsymbol{\theta}$) for the full conditional $\boldsymbol{x}$ in the denominator, and then evaluate the ratio at the modal value for $\boldsymbol{x}$ for each $\boldsymbol{\theta}$. The evaluation is performed for values of $\boldsymbol{\theta}$ on a fine grid centred (approximately) at the modal value. This unnormalised density restricted to the grid is then renormalised so it integrates to one. The results are shown in column (a) in Figure 2, displaying the contour-plot of the estimated posterior marginal for $\boldsymbol{\theta}$.

The marginal posterior for the Easy-case is more symmetric than the one for the Hard-case. This is natural when we take into account the high Poisson counts which makes the likelihood more like a Gaussian. As mentioned in Section 1, this is the Laplace-approximation as derived (differently) by Tierney and Kadane (1986). The relative error in the renormalised density is $\mathcal{O}(N^{-3/2})$ where N is the number of observations, hence it is quite accurate. Note that the quality of this approximation does not change if we consider the posterior marginal for $(\kappa_{\boldsymbol{u}}, \kappa_{\boldsymbol{v}})$ instead of $(\log \kappa_{\boldsymbol{u}}, \log \kappa_{\boldsymbol{v}})$.This is, in fact, only a reparametrisation and the relative error is still $\mathcal{O}(N^{-3/2})$.

By summing out $\log \kappa_{\boldsymbol{v}}$ and $\log \kappa_{\boldsymbol{u}}$, respectively, we obtain the marginal posteriors for $\log \kappa_{\boldsymbol{u}}$ and $\log \kappa_{\boldsymbol{v}}$. These are displayed using solid lines in Figure 2 column (b) and (c). To verify these approximations, we ran MCMC algorithms based on (5) for a long time to obtain at least 10^6 near iid samples. The density estimates based on these samples are shown as dotted lines in column (b) and (c). The estimates based on the MCMC algorithms confirm the accuracy of the Laplace-approximation.

3.3 Approximating $\pi(x_i|\boldsymbol{y})$

Our next task is to approximate the marginal posterior for each x_i making use of (8). Note that $\widetilde{\pi}(x_i|\boldsymbol{\theta}_j, \boldsymbol{y})$ is a GMRF, hence we need to compute the marginal variances for $x_n, \ldots, x_1$. To do this, we make use of the recursions (12) and the practical advises in Section 2.3 which are implemented in `GMRFLib` (Rue and Held, 2005, Appendix B).

The results in Section 3.2 indicate that the quality of (8) depends on how well $\widetilde{\pi}(x_i|\boldsymbol{\theta}_j, \boldsymbol{y})$ approximates $\pi(x_i|\boldsymbol{\theta}_j, \boldsymbol{y})$ for those $\boldsymbol{\theta}_j$ where the probability mass is significant. For this reason, we have compared this approximation for various fixed $\boldsymbol{\theta}_j$ with the estimates for $\pi(x_i|\boldsymbol{\theta}_j, \boldsymbol{y})$ computed from long runs with a MCMC algorithm. The results are displayed in Figure 3 for the Easy-case and Figure 4 for the Hard-case.

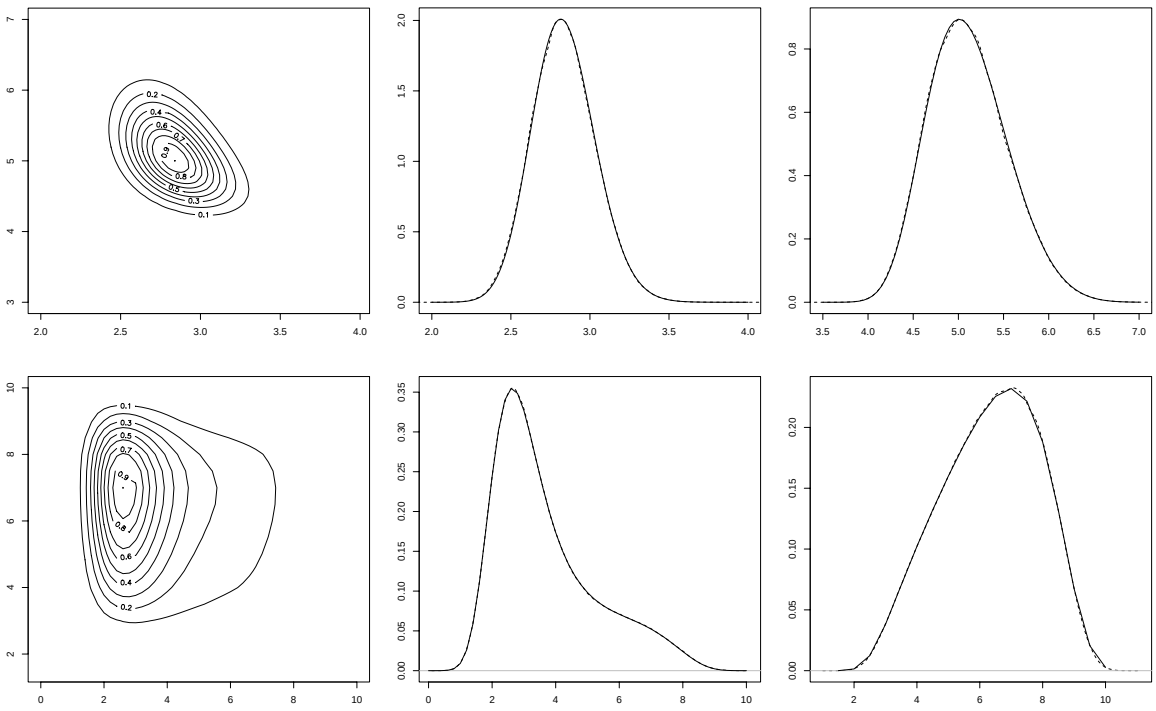


Figure 2: Results for the Easy-case on the top row and the for Hard-case on the bottom row. (a) Approximated marginal posterior density of $(\log \kappa_{\boldsymbol{u}}, \log \kappa_{\boldsymbol{v}})$, (b) approximated marginal posterior density of $\log \kappa_{\boldsymbol{u}}$, and (c) approximated marginal posterior density of $\log \kappa_{\boldsymbol{v}}$. In (b) and (c), the approximated marginals are shown using solid lines, while the estimated marginal posteriors from a long MCMC run are shown with dotted lines.

3.3.1 Marginal posteriors for the spatially structured component for fixed $\boldsymbol{\theta}$

Easy-case Column (d) in Figure 3 shows the value of (the fixed) $\boldsymbol{\theta}_j$ relative to the marginal posterior shown in Figure 2. The first three columns show marginals of the GMRF-approximation for the spatial component $\boldsymbol{u}$ (solid lines) and the estimate obtained from very long MCMC runs (dotted lines). Only three districts are shown. They are selected such that the posterior expected value of u_i for $\boldsymbol{\theta}_j$ located at the modal value, is high (a), intermediate (b) and low (c). The results in Figure 3 indicate that the GMRF-approximation is indeed quite accurate in this case, and only small deviations from the (estimated) truth can be detected.

Hard-case Figure 4 displays the same as Figure 3 but now for the Hard-case. The results for the three first rows are quite good, although the (estimated) true marginal posteriors show some skewness not captured by the Gaussian approximation. The modal value indicated by the Gaussian approximation seems in all cases a little too high, although this is most clear for the last row. In the last row, the precisions for both the spatial structured and unstructured term are (relatively) low and outside the region with significant contribution to the probability mass for $\boldsymbol{\theta}$. With these (relatively) low precisions, we obtain a (relatively) high variance for the non-quadratic term $\exp(x_i)$ in (19), which makes the marginals more skewed. It might appear, at a first glance, that the (estimated) true marginal and the Gaussian approximation are shifted, but this is not the case. There is a skewness factor that is missing in the Gaussian approximation, which has, in this case, nearly the same effect of a shift. The results from this Hard-case are quite encouraging, as the approximations in the central part of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ are all relatively accurate.

3.3.2 Marginal posteriors for the spatially structured component

Figure 5 shows the results using (8) (solid line) to approximate the marginals for the spatial term $\boldsymbol{u}$ in the same three districts that appear in Figure 3 and Figure 4. The (estimated) truth is drawn with dotted lines. The top row shows the Easy-case while the bottom row shows the Hard-case. The columns (a) to (c) relate to the columns of Figure 3 and Figure 4 for the top and bottom row, respectively. Since the accuracy of the Gaussian approximations was verified in Figure 3 and Figure 4 to be quite satisfactory, there is no reason that integrating out $\boldsymbol{\theta}$ will result in inferior results. The approximation (8) is quite accurate for both cases but the marginals are slightly less skewed than the truth. However, the error is quite small. The bottom row demonstrates that (8), which is a mixture of Gaussians, can indeed represent also highly skewed densities.

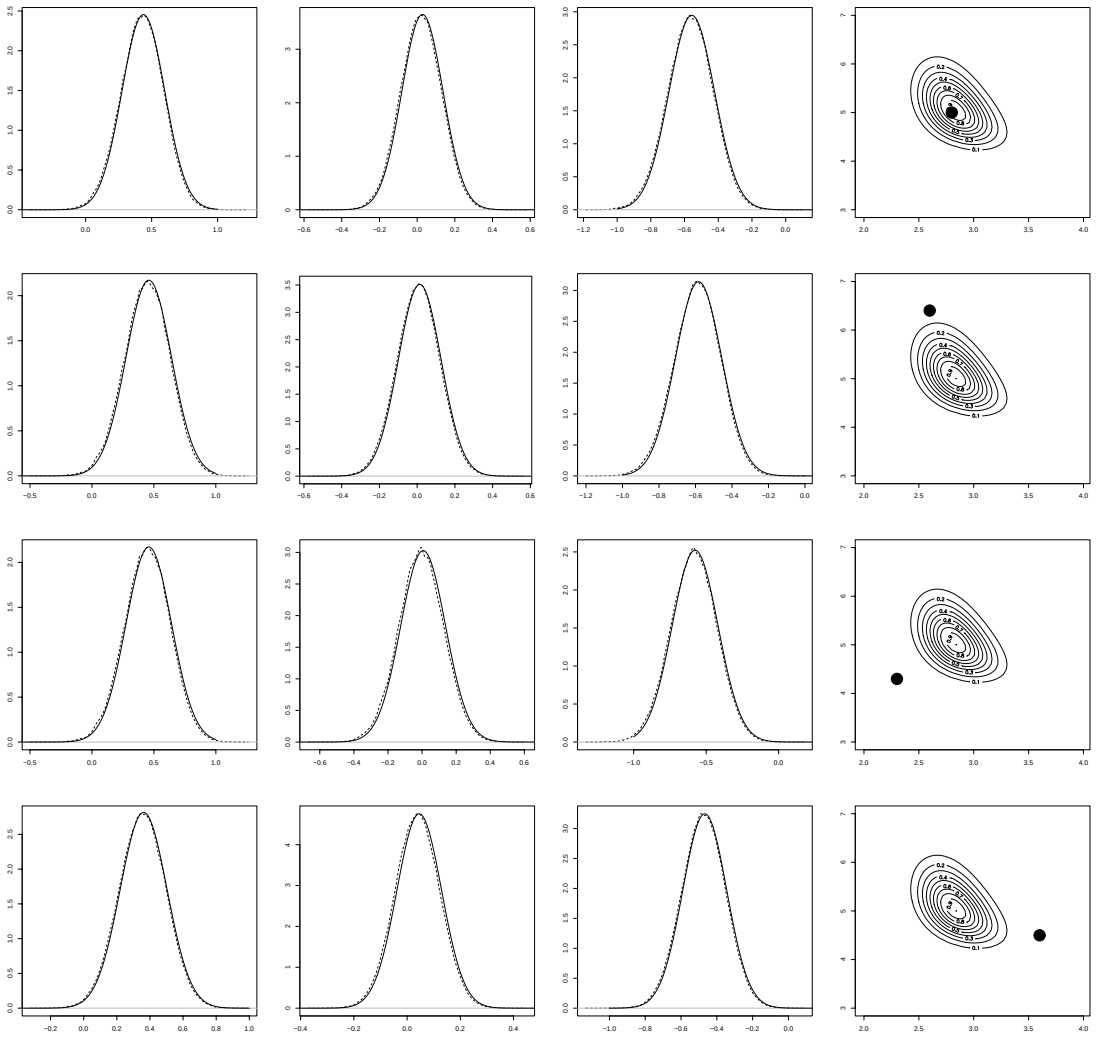


Figure 3: Results for the Easy-case. Each row shows in (d) the location of the fixed $\boldsymbol{\theta}$, and in the first three columns the (estimated) true marginal densities (dotted lines) for the spatial component at three different districts. The solid line displays the Gaussian approximation. The three districts in column (a) to (c) represent districts with (a) high, (b) intermediate, and (c) low value of the posterior expectation of u_i.

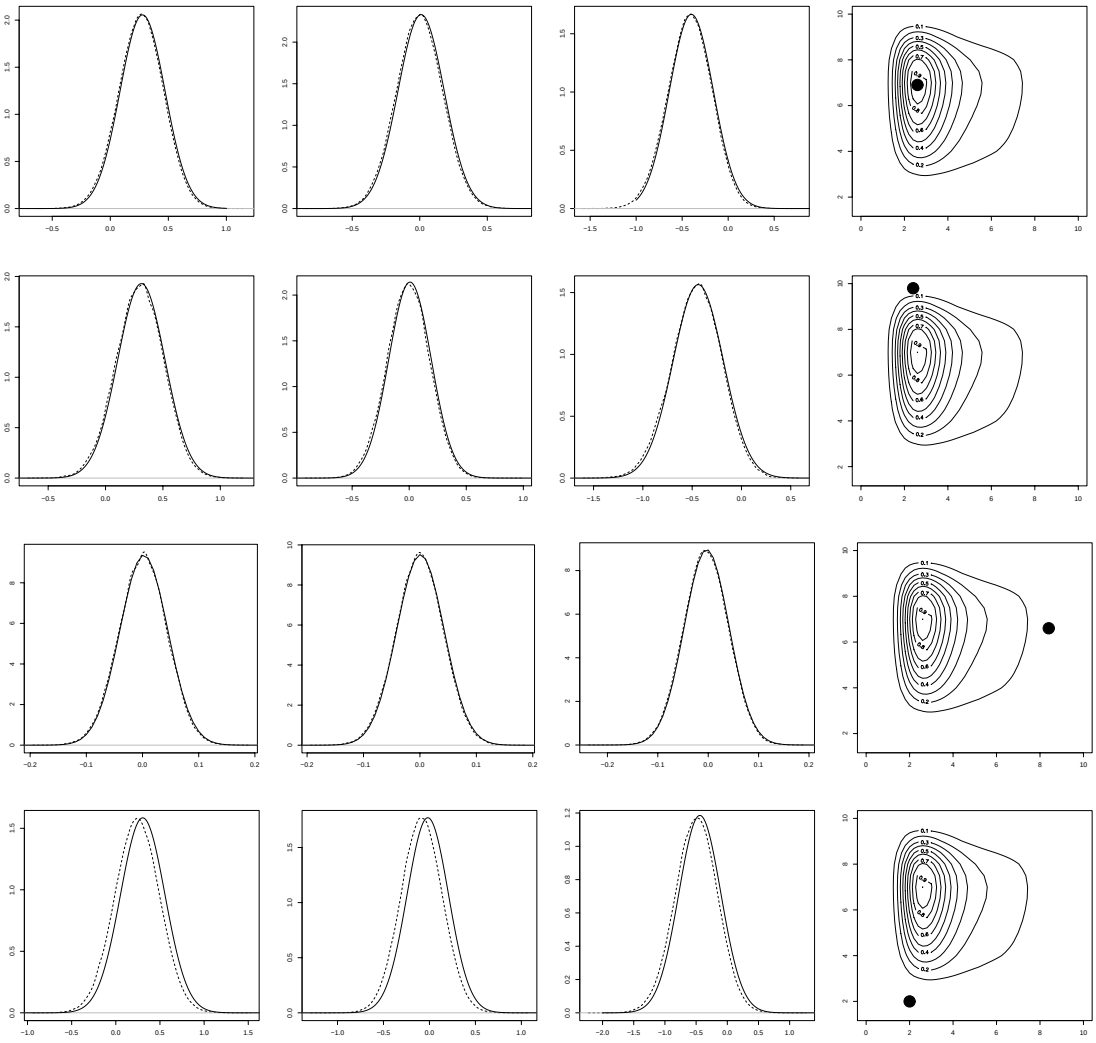


Figure 4: Results for the Hard-case. Each row shows in (d) the location of the fixed $\boldsymbol{\theta}$, and in the first three columns the (estimated) true marginal densities (dotted lines) for the spatial component at three different districts. The solid line displays the Gaussian approximation. The three districts in column (a) to (c) represent districts with (a) high, (b) intermediate, and (c) low value of the posterior expectation of u_i.

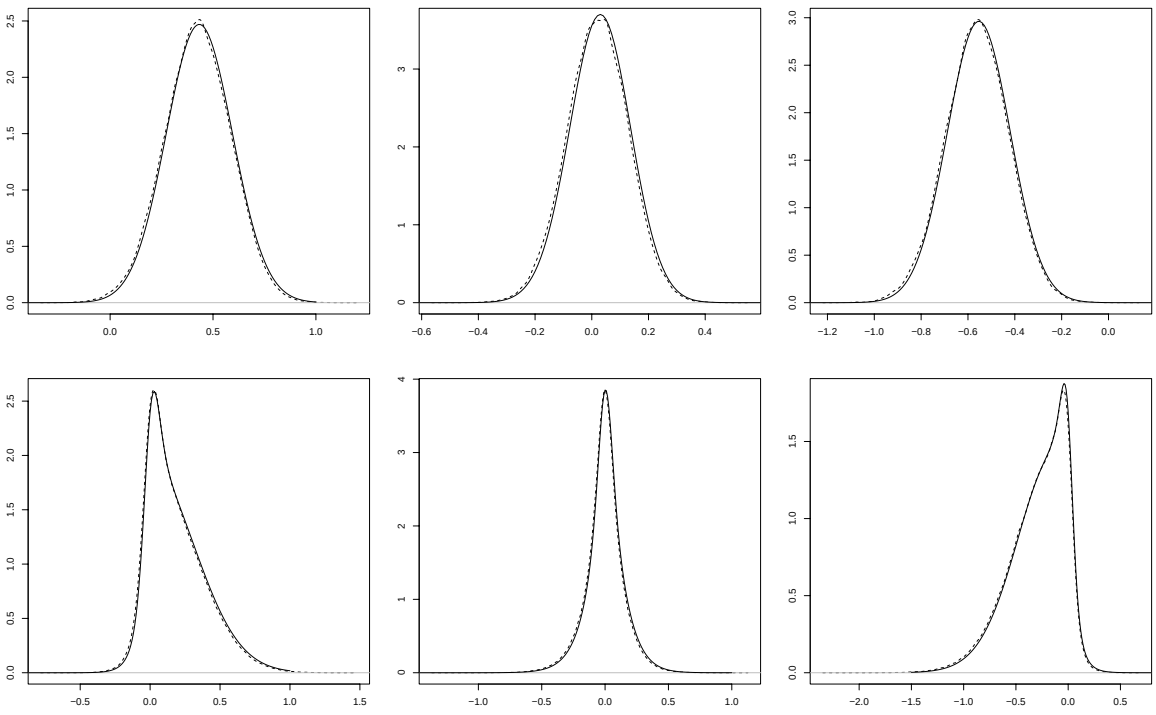
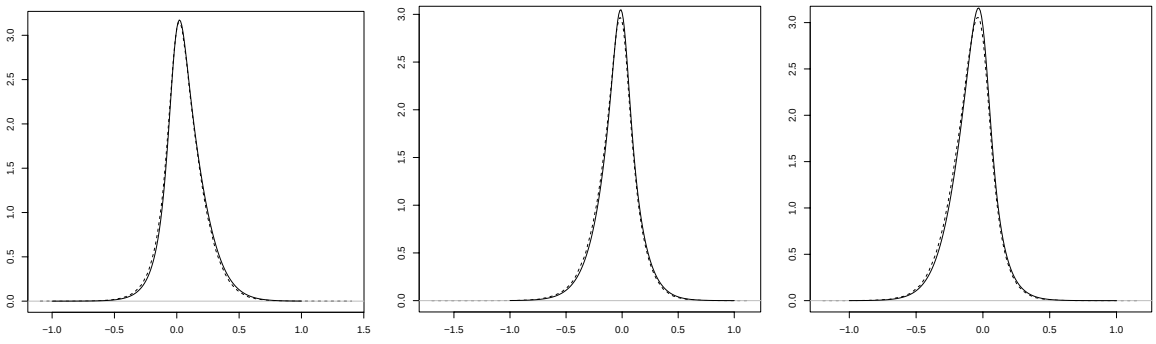


Figure 5: Marginal posteriors for the spatial component in three districts. Easy-case on the top row and Hard-case on the bottom row. Columns (a) to (c) corresponds to the same columns in Figure 3 and Figure 4 for the top and bottom row, respectively. The approximations (8) are drawn with solid line and the (estimated) truth with dotted lines.



(a) (b) (c)

Figure 6: Marginal posteriors for the log-relative risk η_i in three districts for the Hard-case. Columns (a) to (c) corresponds to the same columns in Figure 4 and the bottom row in Figure 5. The approximations (8) are drawn with solid line and the (estimated) truth with dotted lines.

3.3.3 Marginal posteriors for the log-relative risk

We will now present the results for the marginal posteriors for the log-relative risk η_i for the Hard-case. It is not clear how the accuracy for these approximations should relate to those for the spatial component in Figure 5. It is η_i that is indirectly observed through y_i, but on the other hand, the difference between η_i and the spatial component u_i is only an additional unstructured component. The results are shown in Figure 6 for the same three districts shown in Figure 4 and in the last row of Figure 5. Again, the approximation (8) does not capture the right amount of skewness, for the same reason already discussed for Figure 3 and Figure 4. However, when $\boldsymbol{\theta}$ is integrated out, also the marginal posterior for $\boldsymbol{\eta}$ is quite well approximated.

3.4 Semi-parametric ecological regression

We will now consider an extension of the BYM-model (19) given by Natario and Knorr-Held (2003), which allows for adjusting the log-relative risk by a semi-parametric function of a covariate which is believed to influence the risk. The purpose of this example is to illustrate the ability of (8) to account for linear constraints, which we discuss in more details shortly. Similarly to Natario and Knorr-Held (2003), we will use data on mortality from larynx cancer among males in the 544 districts of Germany over the period $1986 - 1990$, with estimates for lung cancer mortality as a proxy for smoking consumption as a covariate. We refer to their report for further details and background for this application.

The extension of the BYM-model is as follows. At the first stage we still assume $y_i \sim \text{Po}(e_i \exp(\eta_i))$ for each i, but now

$$\eta_i = u_i + v_i + f(c_i). \tag{21}$$

The two first terms are the spatially structured and unstructured term as in the

BYM-model, whereas $f(c_i)$ is the effect of a covariate which has value c_i in district i. The covariate function $f(\cdot)$ is a random smooth function with small squared second order differences. The function $f(\cdot)$ is defined to be piecewise linear between the function values $\{f_j\}$ at $m = 100$ equally distant values of c_i, chosen to reflect the range of the covariate. We have scaled the covariates to the interval $[1, 100]$. The vector of $\boldsymbol{f} = (f_1, \ldots, f_m)^T$ is also a GMRF, with density

$$\pi(\boldsymbol{f} \mid \kappa_{\boldsymbol{f}}) \propto \kappa_{\boldsymbol{f}}^{(m-2)/2} \exp\left(-\frac{\kappa_{\boldsymbol{f}}}{2} \sum_{j=2}^{m} (f_j - 2f_{j-1} + f_{j-2})^2\right) \tag{22}$$

This is a so-called second order random walk (RW2) model with (unknown) precision $\kappa_{\boldsymbol{f}}$, see for example Rue and Held (2005, Ch. 3). The density (22) can be interpreted as an approximated Galerkin solution to the stochastic differential equation, $f''(t) = dW(t)/dt$, where $W(t)$ is the Wiener process (Lindgren and Rue, 2005). We further impose the constraint $\sum_i u_i = 0$ to separate out the effect of the covariate. Note that the extended BYM-model is still a hierarchical GMRF-model but now $\boldsymbol{x} = (\boldsymbol{\eta}, \boldsymbol{u}, \boldsymbol{f})^T$. It is easy to derive the corresponding precision matrix and posterior density, but we avoid it here.

Adding a semi-parametric effect of a covariate extends directly the BYM-model presented in Section 3.1. However, the fundamental change is not the addition of the extra hyperparameter $\kappa_{\boldsymbol{f}}$, but the introduction of the linear constraint imposed to separate out the effect of the covariate. We need to make use of the correction in Section 2.2 to adjust marginal variances for the constraint, moreover, we need to do constrained optimisation to locate the mode in order to compute the GMRF-approximations. Both tasks are easily done with GMRFs and a few constraints do not slow down the computations.

We will now present the results focusing on the effect of the covariate. The other marginal posteriors are, in fact, similar to those presented in Section 3.2 and Section 3.3. The unknown precisions were all assigned Gamma-priors with parameters $a = 1$ and $b = 0.00005$ following Natario and Knorr-Held (2003). Figure 7 shows the approximated marginal posterior for $\boldsymbol{f}$, represented by the mean, the 0.025, and 0.975 quantile. The approximations (8) are drawn with solid lines and the (estimated) truth with dotted lines. The middle lines are the posterior mean, the lower curves are the 0.025 quantile and the upper curves are the 0.975 quantiles. The results show that the approximation is quite accurate. However, the approximation (8) does not capture the correct skewness, in a similar way to the last column in Figure 4. This claim is also verified by comparing the marginal posteriors for each f_j (not shown).

4 Discussion

In this report we have investigated how marginal posterior densities can be approximated using the GMRF-approximation in (8). We apply the GMRF-approximation

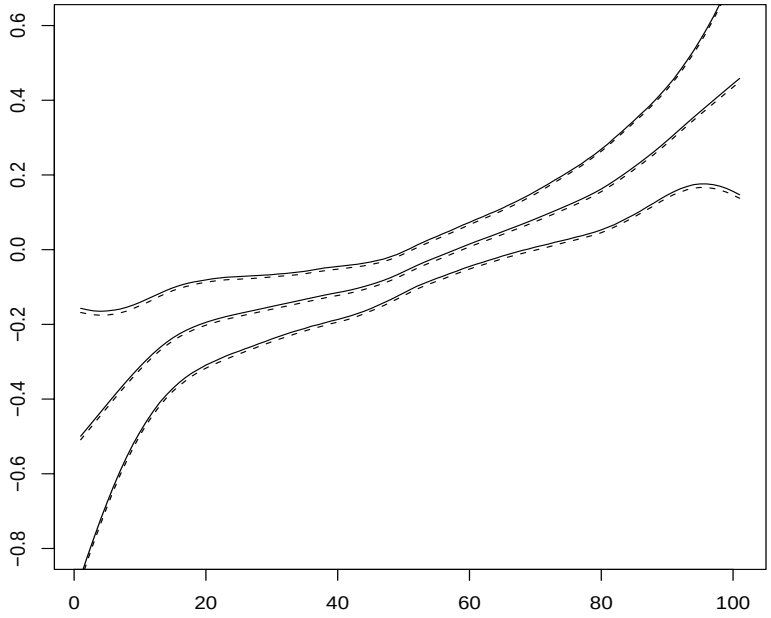


Figure 7: Marginal posteriors for the covariate effect, here represented by the mean, the 0.025 and 0.095 quantile. The approximations (8) are drawn with solid lines and the (estimated) truth with dotted lines. The middle lines are the posterior mean, the lower curves are the 0.025 quantile and the upper curves are the 0.975 quantiles.

to the full conditional for the latent GMRF component in hierarchical GMRF models. We use this to approximate both marginal posteriors for the hyperparameters and marginal posteriors for the components of the latent GMRF itself. We have also discussed how to compute marginal variances for GMRFs with and without linear constraints, and derived the recursion from a statistical point of view. The main motivation for using approximations to estimate marginal posteriors, is *only* computational efficiency. Computations with GMRFs are very efficient using numerical methods for sparse matrices, and make it possible to approximate posterior marginals nearly instant compared to the time required by MCMC algorithms. This makes the class of hierarchical GMRF-models a natural candidate for nearly instant approximated inference. The approximations were verified against very long runs of a one-block MCMC algorithm, with the following conclusions.

- The results were indeed positive in general and we obtained quite accurate approximations for all marginals investigated. Even for a quite hard dataset with low Poisson counts, the approximations were quite accurate.
- All results failed to capture the correct amount of (small) skewness, whereas the mode and the width of the density were more accurately approximated. However, the lack of skewness is a consequence of using symmetric approximations.

The range of application of these findings is, to our point of view, not only restricted to the class of BYM-models considered here but can be extended to many

hierarchical GMRF-models. In particular, we want to mention hierarchical models based on log-Gaussian Cox processes (Møller et al., 1998) and model-based Geostatistics (Diggle et al., 1998). Both these popular model-classes can be considered as hierarchical GMRF-models, where Gaussian fields can be replaced by GMRFs using the results of Rue and Tjelmeland (2002), or sometimes better, using intrinsic GMRFs. The typical feature of these models is that the number of observations N is quite small. The approximation techniques we have presented, will give at least as accurate results than those presented in this paper. Another feature of these models is that, working with Gaussian fields directly, MCMC based inference is indeed challenging to implement and computationally heavy. For these reasons, the ability to use GMRFs and nearly instant approximated inference is indeed a huge step forward. All these results will be reported elsewhere.

Our approach to compute marginal posteriors is based on GMRF-approximations and the accuracy depends on the accuracy of the GMRF-approximation. Although this approximation is sufficiently accurate for many and often typical examples, is not difficult to find cases where such an approximation is not accurate enough, see for example Figure 4 last row. An important task for future work, is to construct methods that can go beyond the GMRF-approximation allowing for non-Gaussian approximations to the full conditional. One such class of approximation was introduced by Rue et al. (2004). This approximation can be applied to compute marginals as well. Preliminary results in this direction are indeed encouraging, and we are confident that improved approximation methods can be constructed without too much extra effort. These improved approximations will also serve as a validation procedure for the class of approximations considered here. They may, in fact, be used to detect if the approximations based on the GMRF-approximation are sufficiently accurate.

It is quite fast to compute our approximations even with our brute-force approach for integrating out the hyperparameters. This step can and need to be improved. This will increase the speed significantly while keeping the results nearly unchanged. There is a natural limit to the number of hyperparameters $\boldsymbol{\theta}$ our approach can deal with. Since we integrate out these numerically, we would like $\dim(\boldsymbol{\theta}) \leq 3$. However, approximated schemes are indeed possible for higher dimensions as well, although we admit that we do not have large experience in this direction. Automatic construction of numerical quadrature rules based on the behaviour near the mode, is also a possibility which we will investigate. The benefit here, is that the numerical integration is adaptive which is also a requirement for constructing black-box algorithms for approximating marginal posteriors for hierarchical GMRF-models.

The results presented in this article imply that for many (Bayesian) hierarchical GMRF-models, namely those with a small number of hyperparameters, at least, MCMC algorithms are not needed to achieve accurate estimations of marginal posteriors. Moreover, approximated inference can be computed nearly instant compared to MCMC algorithms. This does not imply that MCMC algorithms are not needed, only that they are not needed in all cases.

References

Bernardinelli, L., Pascutto, C., Best, N. G., and Gilks, W. R. (1997). Disease mapping with errors in covariates. *Statistics in Medicine*, (16):741–752.

Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems (with discussion). *Journal of the Royal Statistical Society, Series B*, 36(2):192–225.

Besag, J. (1975). Statistical analysis of non-lattice data. *The Statistician*, 24(3):179–195.

Besag, J. (1981). On a system of two-dimentional recurrence equations. *Journal of the Royal Statistical Society, Series B*, 43(3):302–309.

Besag, J., York, J., and Mollié, A. (1991). Bayesian image restoration with two applications in spatial statistics (with discussion). *Annals of the Institute of Statistical Mathematics*, 43(1):1–59.

Diggle, P. J., Tawn, J. A., and Moyeed, R. A. (1998). Model-based geostatistics (with discussion). *Journal of the Royal Statistical Society, Series C*, 47(3):299–350.

Erisman, A. M. and Tinney, W. F. (1975). On computing certain elements of the inverse of a sparse matrix. *Communications of the ACM*, 18(3):177–179.

Gelfand, A. E., Sahu, S. K., and Carlin, B. P. (1995). Efficient parameterisations for normal linear mixed models. *Biometrika*, 82(3):479–488.

Knorr-Held, L. and Raßer, G. (2000). Bayesian detection of clusters and discontinuities in disease maps. *Biometrics*, 56:13–21.

Knorr-Held, L. and Rue, H. (2002). On block updating in Markov random field models for disease mapping. *Scandinavian Journal of Statistics*, 29(4):597–614.

Lindgren, F. and Rue, H. (2005). A note on the second order random walk model for irregular locations. Statistics Report No. 6, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway.

Møller, J., Syversveen, A. R., and Waagepetersen, R. P. (1998). Log Gaussian Cox processes. *Scandinavian Journal of Statistics*, 25:451–482.

Natario, I. and Knorr-Held, L. (2003). Non-parametric ecological regression and spatial variation. *Biometrical Journal*, 45:670–688.

Papaspiliopoulos, O., Roberts, G. O., and Sköld, M. (2003). Non-centered parameterizations for hierarchical models and data augmentation (with discussion). In *Bayesian Statistics, 7*, pages 307–326. Oxford Univ. Press, New York.

Rue, H. (2001). Fast sampling of Gaussian Markov random fields. *Journal of the Royal Statistical Society, Series B*, 63(2):325–338.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*, volume 104 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Rue, H., Steinsland, I., and Erland, S. (2004). Approximating hidden Gaussian Markov random fields. *Journal of the Royal Statistical Society, Series B*, 66(4):877–892.

Rue, H. and Tjelmeland, H. (2002). Fitting Gaussian Markov random fields to Gaussian fields. *Scandinavian Journal of Statistics*, 29(1):31–50.

Takahashi, K., Fagan, J., and Chen, M. S. (1973). Formation of a sparse bus impedance matrix and its application to short circuit study. In *8th PICA Conference proceedings*, pages 63–69. IEEE Power Engineering Society. Papers presented at the 1973 Power Industry Computer Application Conference in Minneapolis, Minnesota.

Tierney, L. and Kadane, J. B. (1986). Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86.

Wilkinson, D. J. (2003). Discussion to "Non-centered parameterizations for hierarchical models and data augmentation" by O. Papaspiliopoulos, G. O. Roberts and M. Sköld. In *Bayesian Statistics, 7*, pages 323–324. Oxford Univ. Press, New York.

Paper II

Approximate Bayesian Inference for Latent Gaussian Models Using Integrated Nested Laplace Approximation.

Approximate Bayesian Inference for Latent Gaussian Models Using Integrated Nested Laplace Approximations

Håvard Rue & Sara Martino
Department of Mathematical Sciences
NTNU, Norway

Nicolas Chopin
CREST-LS and ENSAE, Paris, France

Abstract

We are concerned with Bayesian inference for latent Gaussian models, that is models involving a Gaussian latent field (in a broad sense), controlled by few parameters. This is perhaps the class of models most commonly encountered in applications: the latent Gaussian field can represent, for instance, a mix of smoothing splines or smooth curves, temporal and spatial processes. Hence, popular smoothing-spline models, state-space models, semiparametric regression, spatial and spatio-temporal models, log-Gaussian Cox-processes, and geostatistical models, all fall in this category.

We consider the case where the observational model is non-Gaussian, so that the posterior marginals are not available in closed form. Prominent examples are Poisson and Binomial count data. For such models, Markov chain Monte Carlo methods can be implemented, but they are not with problems, both in terms of convergence and computational time. In some practical applications, the extent of these problems is such that Markov chain Monte Carlo is simply non feasible.

We show that, by using an integrated nested Laplace approximation and its simplified version, we can directly compute very accurate approximations to the posterior marginals. The main benefit of these approximations is computational: where MCMC algorithms need hours and days to run, our approximations provide more precise estimates in seconds and minutes. Another advantage is their ease of use, which should facilitate and automate the analysis of data generated from latent Gaussian models.

KEYWORDS: Approximate Bayesian inference, Gaussian Markov random fields, Hierarchical GMRF-models, Laplace approximation, Numerical methods for sparse matrices, Parallel computing

1 Introduction

1.1 Latent Gaussian models

Latent Gaussian models are widely used in Bayesian analysis. Such models assume a latent Gaussian field $\boldsymbol{x} = (x_1, \ldots, x_n)^T$, which is observed pointwise through n_d conditional independent data $\boldsymbol{y}$. In its simplest form, the covariance matrix of the latent Gaussian field and the likelihood are governed by a few parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_m)^T$, say $m \leq 6$. Linear constraints of the form $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{e}$, where the matrix $\boldsymbol{A}$ has rank k, may also be imposed. The posterior then reads

$$\pi(\boldsymbol{x}, \boldsymbol{\theta} \mid \boldsymbol{y}) \;\propto\; \pi(\boldsymbol{\theta})\, \pi(\boldsymbol{x} \mid \boldsymbol{\theta}) \prod_{i \in \mathcal{I}} \pi(y_i \mid x_i, \boldsymbol{\theta}).$$

In this paper, we assume that the main goal of the inference is to compute all, or some of, the n posterior marginals for x_i plus possibly the posterior marginals for $\boldsymbol{\theta}$ or some θ_j. If needed, the marginal densities can be post-processed to compute posterior expectations, variances, quantiles etc. We are concerned with the case where $\pi(y_i|x_i, \boldsymbol{\theta})$ is well-behaved, albeit non-Gaussian, so that the posterior marginals $\pi(x_i|\boldsymbol{y})$ and $\pi(\theta_j|\boldsymbol{y})$ are not available in closed form.

A few examples will demonstrate the wide use of latent Gaussian models. We loosely classify them with respect to their 'physical dimension', like 1D, 2D and 3D. In 1D, the latent process is often a mix of unstructured Gaussian effects and smooth processes in continuous or discrete 'time', such as integrated Wiener processes or random walk models. These can be used in a temporal context in various applications (Wecker and Ansley, 1983; Carter and Kohn, 1994; Fahrmeir and Tutz, 2001; Kitagawa and Gersch, 1996; Durbin and Koopman, 2000), or to model semiparametrically the effect of covariates in a regression setup (Lang and Brezger, 2004; Biller and Fahrmeir, 1997). In 2D, typical examples are model-based geostatistics (Diggle et al., 1998; Diggle and Ribeiro, 2006), and more generic smoothing models similar to the well-known BYM model for disease mapping (Besag et al., 1991; Weir and Pettitt, 2000), see also Banerjee et al. (2004) for many more examples. Models for spatial log-Gaussian Cox processes (Møller et al., 1998) are also in this class. Spatial models can also include 1D structures, like splines which model various covariate effects, see for example Natario and Knorr-Held (2003) and Fahrmeir and Lang (2001). 3D examples are usually an extension of a spatial model to a temporal or depth dimension, e.g. Allcroft and Glasbey (2003); Carlin and Banerjee (2003); Knorr-Held (2000); Knorr-Held and Besag (1998) and Wikle et al. (1998).

1.2 Inference: MCMC approaches

The common approach to inference for latent Gaussian models is Markov chain Monte Carlo (MCMC). It is well known however that MCMC tends to exhibit poor performance when applied to such models. Various factors explain this. First, the components of the latent field $\boldsymbol{x}$ are strongly dependent on each other. Second, $\boldsymbol{\theta}$ and $\boldsymbol{x}$ are also strongly dependent, especially when n is large. A common approach to (try to) overcome this first problem, is to construct a joint proposal based on a Gaussian approximation to the

full conditional of $\boldsymbol{x}$ (Gamerman, 1997, 1998; Carter and Kohn, 1994; Knorr-Held, 1999; Knorr-Held and Rue, 2002; Rue et al., 2004). The second problem requires, at least partially, a joint update of both $\boldsymbol{\theta}$ and $\boldsymbol{x}$. One suggestion is to (try to) use the one-block approach of Knorr-Held and Rue (2002): make a proposal for $\boldsymbol{\theta}$ to $\boldsymbol{\theta}'$, update $\boldsymbol{x}$ from the Gaussian approximation conditional on $\boldsymbol{\theta}'$, then accept/reject jointly; see Rue and Held (2005, Ch. 4) for variations on this approach. Some models can alternatively be reparameterised to overcome the second problem (Papaspiliopoulos et al., 2007). Independence samplers can also sometimes be constructed (Rue et al., 2004). For some (observational) models, auxiliary variables can be introduced to simplify the construction of Gaussian approximations (Shephard, 1994; Albert and Chib, 1993; Holmes and Held, 2006; Frühwirth-Schnatter et al., 2006; Frühwirth-Schnatter and Frühwirth, 2007; Rue and Held, 2005). Despite all these developments, MCMC remains painfully slow from the end user's point of view.

1.3 Inference: Deterministic approximations

Gaussian approximations play a central role in the development of more efficient MCMC algorithms. This remark leads to the following questions:

- Can we bypass MCMC entirely, and base our inference solely on such closed-form approximations?
- To which extent can we advocate an approach that leads to a (presumably) small approximation error over another approach giving rise to a (presumably) large MCMC error?

Obviously, MCMC errors seem preferable, as they can be made arbitrarily small, for arbitrarily large computational time. We argue however that, for a given computational cost, the deterministic approach developed in this paper outperforms MCMC algorithms to such an extent that, for latent Gaussian models, resorting to MCMC rarely makes sense in practice.

It is useful to provide some orders of magnitude. In typical spatial examples where the dimension n is a few thousands, our approximations for all the posterior marginals can be computed in (less than) a minute or a few minutes. The corresponding MCMC samplers need hours or even days to compute accurate posterior marginals. The approximation bias is in typical examples much less than the MCMC error and negligible in practice. More formally, on one hand it is well-known that MCMC is a last resort solution: Monte Carlo averages are characterised by additive $\mathcal{O}_p(N^{-1/2})$ errors, where N is the simulated sample size. Thus, it is easy to get rough estimates, but nearly impossible to get accurate ones; an additional correct digit requires 100 times more computational power. More importantly, the implicit constant in $\mathcal{O}_p(N^{-1/2})$ often hides a curse of dimensionality with respect to the dimension n of the problem, which explains the practical difficulties with MCMC mentioned above. On the other hand, Gaussian approximations are intuitively appealing for latent Gaussian models. For most real problems and datasets, the conditional posterior of $\boldsymbol{x}$ is typically well-behaved, and looks 'almost' Gaussian. This is clearly due to the latent Gaussian prior assigned to $\boldsymbol{x}$, which has a non-negligible impact on the posterior, especially in terms of dependence between the components of $\boldsymbol{x}$.

1.4 Inference: The new approach

Our approach is based on the following approximation $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ for the marginal posterior of $\boldsymbol{\theta}$:

$$\widetilde{\pi}(\boldsymbol{\theta} \mid \boldsymbol{y}) \propto \left. \frac{\pi(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})}{\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x}=\boldsymbol{x}^\star(\boldsymbol{\theta})} \tag{1}$$

where $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ is the Gaussian approximation to the full conditional of $\boldsymbol{x}$, and $\boldsymbol{x}^\star(\boldsymbol{\theta})$ is the mode of the full conditional for $\boldsymbol{x}$, for a given $\boldsymbol{\theta}$. The proportionality sign (1) comes from the fact that the normalising constant for $\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ is unknown. This expression is equivalent to Tierney and Kadane (1986)'s Laplace approximation of a marginal posterior distribution and this suggests that the approximation error is relative and of order $\mathcal{O}(n_d^{-3/2})$ after renormalisation. However, since n is not fixed but depends on n_d, standard asymptotic assumptions usually invoked for Laplace expansions, see for example Schervish (1995, p. 453), are not verified here. We will discuss the error rate for this case in more detail in Section 4.

Note that $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ itself tends to depart significantly from Gaussianity. This suggests that a cruder approximation based on a Gaussian approximation to $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is not accurate enough for our purposes; this also applies to similar approximations based on 'equivalent Gaussian observations' around $\boldsymbol{x}^\star$, and evaluated at the mode of (1) (Breslow and Clayton, 1993; Ainsworth and Dean, 2006). A critical aspect of our approach is to explore and manipulate $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ and $\widetilde{\pi}(x_i|\boldsymbol{y})$ in a 'nonparametric' way.

Rue and Martino (2007) used (1) to approximate posterior marginals for $\boldsymbol{\theta}$ for various latent Gaussian models. Their conclusion was that $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is particularly accurate: even long MCMC runs could not detect any error in it. For the posterior marginals of the latent field, they proposed to start from $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ and approximate the density of $x_i|\boldsymbol{\theta}, \boldsymbol{y}$ with the Gaussian marginal derived from $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$, i.e.

$$\widetilde{\pi}(x_i \mid \boldsymbol{\theta}, \boldsymbol{y}) = \mathcal{N}\left\{x_i;\ \mu_i(\boldsymbol{\theta}), \sigma_i^2(\boldsymbol{\theta})\right\}. \tag{2}$$

Here, $\boldsymbol{\mu}(\boldsymbol{\theta})$ is the mean (vector) of the Gaussian approximation, whereas $\boldsymbol{\sigma}^2(\boldsymbol{\theta})$ is a vector of corresponding marginal variances. This approximation can be integrated numerically with respect to $\boldsymbol{\theta}$ using (1), to obtain approximations of the marginals of interest for the latent field,

$$\widetilde{\pi}(x_i \mid \boldsymbol{y}) = \sum_k \widetilde{\pi}(x_i \mid \boldsymbol{\theta}_k, \boldsymbol{y}) \ \times\ \widetilde{\pi}(\boldsymbol{\theta}_k \mid \boldsymbol{y}) \ \times\ \Delta_k. \tag{3}$$

The sum is over values of $\boldsymbol{\theta}$ with area-weights Δ_k. Rue and Martino (2007) showed that the approximate posterior marginals for $\boldsymbol{\theta}$ were accurate, while the error in the Gaussian approximation (2) was higher. In particular, (2) can present a slight error in location and/or a lack of skewness. Another issue in Rue and Martino (2007) was the difficulty to detect the x_i's whose approximation is less accurate. Friel and Rue (2007) made use of similar ideas to perform approximate Bayesian inference for factorisable models (in particular, binary Markov random fields) that allow for recursive computing (Bartolucci and Besag, 2002; Reeves and Pettitt, 2004).

In this paper, we solve all the remaining issues in Rue and Martino (2007), and present a fully automatic approach for approximate inference in latent Gaussian models which we

name *Integrated Nested Laplace Approximations* (INLA). The main tool is to apply the Laplace approximation once more, this time to $\pi(x_i|\boldsymbol{y}, \boldsymbol{\theta})$. We also present a faster alternative which corrects the Gaussian approximation (2) for error in the location and lack of skewness at moderate extra cost. The corrections are obtained by a series expansions of the Laplace approximation. This faster alternative is a natural first choice, because of its low computational cost and high accuracy. It is our experience that INLA outperforms without comparison any MCMC alternative, both in terms of accuracy and computational speed. We also derive tools for assessing the approximation error.

Most of the latent fields in the literature admit conditional independence properties, hence the latent field $\boldsymbol{x}$ is a Gaussian Markov random field (GMRF). Thus, we base INLA on sparse matrix calculations, which are much quicker than dense matrix calculations, see Section 2. An exception are geostatistical models, but fast approximate inference is still possible in this case, using a different approach (Eidsvik et al., 2006), or combining the INLA approach with GMRF-proxies to Gaussian fields (Rue and Tjelmeland, 2002).

1.5 Plan of paper

Section 2 contains preliminaries on GMRFs, sparse matrix computations and Gaussian approximations. Section 3 explains how to approximate $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and $\pi(x_i|\boldsymbol{\theta}, \boldsymbol{y})$, using the Integrated nested Laplace approximation (INLA) approach. For the latter distributions, three approximations are discussed: Gaussian, Laplace and simplified Laplace. Section 4 discusses the error rates of the Laplace approximations used in INLA. Section 5 illustrates the performance of INLA through simulated and real examples, which include multiscale analysis of non-Gaussian time-series data, stochastic volatility models, spatial semi-parametric ecological regression and spatial log-Gaussian Cox processes. Section 6 discuss two extensions: approximations of the marginal likelihood and an alternative integration scheme for cases where the number of hyperparameters is not small but moderate. We end with a general discussion in Section 7.

2 Preliminaries

We present here basic properties of GMRFs, and explain how to perform related computations using sparse matrix algorithms. We then discuss how to compute Gaussian approximations for a latent GMRF. See Rue and Held (2005) for more details on both issues. Denote by $\boldsymbol{x}_{-i}$ the vector $\boldsymbol{x}$ minus its ith element, by $\mathcal{N}(\boldsymbol{\mu},\ \boldsymbol{\Sigma})$ the Gaussian distribution, and by $\mathcal{N}(\boldsymbol{x};\ \boldsymbol{\mu},\ \boldsymbol{\Sigma})$, the Gaussian density at configuration $\boldsymbol{x}$.

2.1 Gaussian Markov Random Fields

A GMRF is a Gaussian random variable $\boldsymbol{x} = (x_1, \ldots, x_n)$ with Markov properties: for some $i \neq j$'s, x_i and x_j are independent conditional upon $\boldsymbol{x}_{-ij}$. These Markov properties are conveniently encoded in the precision (inverse covariance) matrix $\boldsymbol{Q}$: $Q_{ij} = 0$ if and only if x_i and x_j are independent conditional upon $\boldsymbol{x}_{-ij}$. Let the undirected graph $\mathcal{G}$ denote the conditional independence properties of $\boldsymbol{x}$, then $\boldsymbol{x}$ is said to be a GMRF with

respect to $\mathcal{G}$. If the mean of $\boldsymbol{x}$ is $\boldsymbol{\mu}$, the density of $\boldsymbol{x}$ is

$$\pi(\boldsymbol{x}) = (2\pi)^{-n/2}|\boldsymbol{Q}|^{1/2}\exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{Q}(\boldsymbol{x}-\boldsymbol{\mu})\right\}. \tag{4}$$

In most cases only $\mathcal{O}(n)$ of the n^2 entries of $\boldsymbol{Q}$ are non-zero, so $\boldsymbol{Q}$ is sparse. This allows for fast factorisation of $\boldsymbol{Q}$ as $\boldsymbol{L}\boldsymbol{L}^T$, where $\boldsymbol{L}$ is the (lower) Cholesky triangle. The sparseness of $\boldsymbol{Q}$ is inherited into $\boldsymbol{L}$, thanks to the global Markov property: for $i < j$, such that i and j are separated by $F(i,j) = \{i+1,\ldots,j-1,\ldots,j+1,\ldots,n\}$ in $\mathcal{G}$, $L_{ji} = 0$. Thus, only non-null terms in $\boldsymbol{L}$ are computed. In addition, nodes can be re-ordered to decrease the number of non-zero terms in $\boldsymbol{L}$. The typical cost of factorising $\boldsymbol{Q}$ into $\boldsymbol{L}\boldsymbol{L}^T$ is $\mathcal{O}(n)$ for 1D, $\mathcal{O}(n^{3/2})$ for 2D and $\mathcal{O}(n^2)$ for 3D GMRFs.

Solving equations which involve $\boldsymbol{Q}$ also makes use of the Cholesky triangle. For example, $\boldsymbol{Q}\boldsymbol{x} = \boldsymbol{b}$ is solved in two steps. First solve $\boldsymbol{L}\boldsymbol{v} = \boldsymbol{b}$, then solve $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{v}$. If $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ then the solution of $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ has precision matrix $\boldsymbol{Q}$. This is the general method for producing random samples from a GMRF. The log density at any $\boldsymbol{x}$, $\log\pi(\boldsymbol{x})$, can easily be computed using (4) since $\log|\boldsymbol{Q}| = 2\sum_i \log L_{ii}$.

Marginal variances can also be computed efficiently. To see this, we can start with the equation $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ where $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Recall that the solution $\boldsymbol{x}$ has precision matrix $\boldsymbol{Q}$. Writing this equation out in detail, we obtain $L_{ii}x_i = z_i - \sum_{k=i+1}^n L_{ki}x_k$ for $i = n,\ldots,1$. Multiplying each side with x_j $j \geq i$, and taking expectation, we obtain

$$\Sigma_{ij} = \delta_{ij}/L_{ii}^2 - \frac{1}{L_{ii}}\sum_{k=i+1}^{n} L_{ki}\Sigma_{kj}, \quad j \geq i,\ i = n,\ldots,1, \tag{5}$$

where $\boldsymbol{\Sigma}$ $(= \boldsymbol{Q}^{-1})$ is the covariance matrix. Thus Σ_{ij} can be computed from (5), letting the outer loop i run from n to 1 and the inner loop j from n to i. If we are only interested in the marginal variances, we only need to compute Σ_{ij}'s for which L_{ji} (or L_{ij}) is not known to be zero, see above. This reduce the computational costs to typically $\mathcal{O}(n(\log n)^2)$ in the spatial case; see Rue and Martino (2007, Sec. 2) for more details.

When the GMRF is defined with additional linear constraints, like $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{e}$ for a $k \times n$ matrix $\boldsymbol{A}$ of rank k, the following strategy is used: if $\boldsymbol{x}$ is a sample from the unconstrained GMRF, then

$$\boldsymbol{x}^c = \boldsymbol{x} - \boldsymbol{Q}^{-1}\boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^T)^{-1}(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{e}) \tag{6}$$

is a sample from the constrained GMRF. The expected value of $\boldsymbol{x}^c$ can also be computed using (6). This approach is commonly called 'conditioning by Kriging', see Cressie (1993) or Rue (2001). Note that $\boldsymbol{Q}^{-1}\boldsymbol{A}^T$ is computed by solving k linear systems, one for each column of $\boldsymbol{A}^T$. The additional cost of the k linear constraints is $\mathcal{O}(nk^2)$. Marginal variances under linear constraints can be computed in a similar way, see Rue and Martino (2007, Sec. 2).

2.2 Gaussian Approximations

Our approach is based on Gaussian approximations to densities of the form:

$$\pi(\boldsymbol{x}) \propto \exp\left\{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \sum_{i\in\mathcal{I}} g_i(x_i)\right\}. \tag{7}$$

where $g_i(x_i)$ is $\log \pi(y_i|x_i, \boldsymbol{\theta})$ in our settings. The Gaussian approximation $\widetilde{\pi}_G(\boldsymbol{x})$ is obtained by matching the modal configuration and the curvature at the mode. The mode is computed iteratively. Let $\boldsymbol{\mu}^{(0)}$ be the initial guess, and expand $g_i(x_i)$ around $\mu_i^{(0)}$ to the second order,

$$g_i(x_i) \approx g_i(\mu_i^{(0)}) + b_i x_i - \frac{1}{2} c_i x_i^2 \tag{8}$$

where $\{b_i\}$ and $\{c_i\}$ depend on $\boldsymbol{\mu}^{(0)}$. A Gaussian approximation is obtained, with precision matrix $\boldsymbol{Q} + \text{diag}(\boldsymbol{c})$ and mode given by the solution of $(\boldsymbol{Q} + \text{diag}(\boldsymbol{c}))\boldsymbol{\mu}^{(1)} = \boldsymbol{b}$. This process is repeated until it converges to a Gaussian distribution with, say, mean $\boldsymbol{x}^\star$ and precision matrix $\boldsymbol{Q}^\star = \boldsymbol{Q} + \text{diag}(\boldsymbol{c})$. If there are linear constraints, the mean is corrected at each iteration using the expected value of (6).

Since the non-quadratic term in (7) is only a function of x_i and not a function of x_i and x_j, say, the precision matrix of the Gaussian approximation is of the form $\boldsymbol{Q} + \text{diag}(\boldsymbol{c})$. This is computationally convenient, as the Markov properties of the GMRF are preserved.

Density (7) may seem restrictive: a more complex density is obtained if, say, y_1 depends on the sum $x_1 + x_2$. This happens for example when the observations are a blurred version of the latent field. In such a case, we find it most convenient to alter the latent field: $\boldsymbol{x}$ is augmented with x_{n+1}, where x_{n+1} is $x_1 + x_2$ plus a tiny Gaussian noise; then y_1 depends on x_{n+1} only, and (7) applies.

3 The Integrated Nested Laplace approximation (INLA)

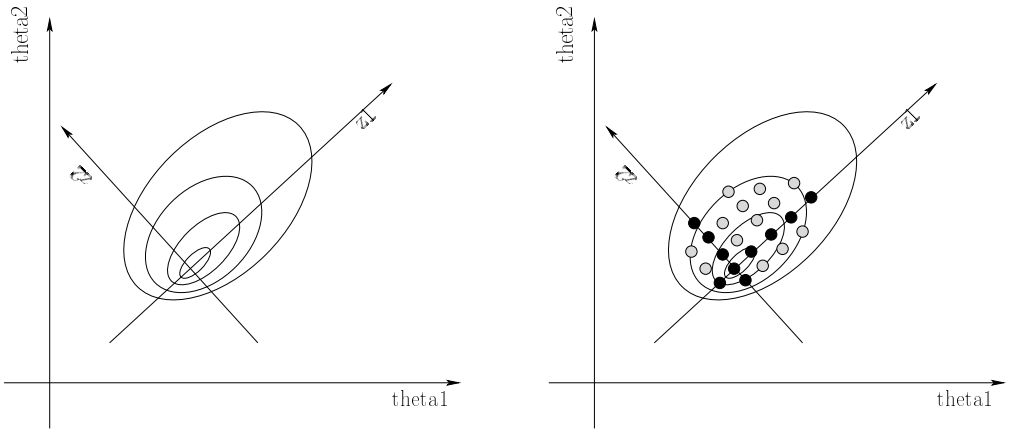
In this section we present the INLA approach for approximating the posterior marginals of the latent Gaussian field, $\pi(x_i|\boldsymbol{y})$, $i = 1, \ldots, n$. The approximation is computed in three steps. The first step (Section 3.1) approximates the posterior marginal of $\boldsymbol{\theta}$ using the Laplace approximation (1). The second step (Section 3.2) computes the Laplace approximation, or the simplified Laplace approximation, of $\pi(x_i|\boldsymbol{y}, \boldsymbol{\theta})$, for selected values of $\boldsymbol{\theta}$, in order to improve on the Gaussian approximation (2). The third step combines the previous two using numerical integration (3).

3.1 Exploring $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$

The first step of the INLA approach is to compute our approximation to the posterior marginal of $\boldsymbol{\theta}$, see (1). The denominator in (1) is the Gaussian approximation to the full conditional for $\boldsymbol{x}$, and is computed as described in Section 2.2. The main use of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is to integrate out the uncertainty with respect to $\boldsymbol{\theta}$ when approximating the posterior marginal of x_i, see (3). For this task, we do not need to represent $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ parametrically, but rather to explore it sufficiently well to be able to select good evaluation points for the numerical integration (3). At the end of this section, we discuss how the posterior marginals $\pi(\theta_j|\boldsymbol{y})$ can be approximated.

Assume for simplicity that $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_m) \in \mathbb{R}^m$, which can always be obtained by reparametrisation;

Step 1 Locate the mode of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, by optimising $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ with respect to $\boldsymbol{\theta}$. This can be done using some quasi-Newton method which builds up an approximation to the



(a) (b)

Figure 1: Illustration of the exploration of the posterior marginal for $\boldsymbol{\theta}$. In (a) the mode is located, the Hessian and the coordinate system for $\boldsymbol{z}$ are computed. In (b) each coordinate direction is explored (black dots) until the log-density drops below a certain limit. Finally the grey dots are explored.

second derivatives of $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ using the difference between successive gradient vectors. The gradient is approximated using finite differences. Let $\boldsymbol{\theta}^\star$ be the modal configuration.

Step 2 At the modal configuration $\boldsymbol{\theta}^\star$ compute the negative Hessian matrix $\boldsymbol{H} > 0$, using finite differences. Let $\boldsymbol{\Sigma} = \boldsymbol{H}^{-1}$, which is the covariance matrix for $\boldsymbol{\theta}$ if the density were Gaussian. To aid the exploration, use standardised variables $\boldsymbol{z}$ instead of $\boldsymbol{\theta}$: let $\boldsymbol{\Sigma} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^T$ be the eigen-decomposition of $\boldsymbol{\Sigma}$, and define $\boldsymbol{\theta}$ via $\boldsymbol{z}$, as follows

$$\boldsymbol{\theta}(\boldsymbol{z}) = \boldsymbol{\theta}^\star + \boldsymbol{V}\boldsymbol{\Lambda}^{1/2}\boldsymbol{z}. \tag{9}$$

If $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is a Gaussian density, then $\boldsymbol{z}$ is $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. This reparametrisation corrects for scale and rotation.

Step 3 Explore $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ using the $\boldsymbol{z}$-parametrisation. Figure 1 illustrates the procedure when $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is unimodal. Panel (a) shows a contour plot of $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ for $m = 2$. Panel (a) also displays the location of the mode and the new coordinate axis for $\boldsymbol{z}$. We want to explore $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ in order to locate the bulk of the probability mass. The result of this procedure is displayed in panel (b). Each dot is a point where $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is considered as significant, and which is used in the numerical integration (3). Details are as follows. We start from the mode ($\boldsymbol{z} = \boldsymbol{0}$), and go in the positive direction of z_1 with step-length δ_z say $\delta_z = 1$, as long as

$$\log \widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{0})|\boldsymbol{y}) - \log \widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{z})|\boldsymbol{y}) < \delta_\pi \tag{10}$$

where, for example $\delta_\pi = 2.5$. Then we switch direction and do similarly. The other coordinates are treated in the same way. This produces the black dots. We can now fill in all the intermediate values by taking all different combinations of the black dots. These new points (shown as grey dots) are included if (10) holds.

Since we layout the points $\boldsymbol{\theta}_k$ in a regular grid, we take all the area-weights Δ_k in (3) to be equal.

Consider now the case where we want to compute the approximation for the posterior marginals for some or all the θ_j's, $\widetilde{\pi}(\theta_j|\boldsymbol{y})$. The rotation of the axis due to $\boldsymbol{V}$ in (9) is inconvenient when summing out the remaining variables $\boldsymbol{\theta}_{-j}$. We can then replace the negative Hessian $\boldsymbol{H}$ by its diagonal, in order to suppress the rotation while retaining the scaling.

3.2 Approximating $\pi(x_i|\boldsymbol{\theta}, \boldsymbol{y})$

We have now a set of weighted points $\{\boldsymbol{\theta}_k\}$ to be used in the integration (3). The next step is to provide accurate approximations for the posterior marginal for the x_i's, conditioned on selected values of $\boldsymbol{\theta}$. We discuss three approximations $\widetilde{\pi}(x_i|\boldsymbol{y}, \boldsymbol{\theta}_k)$, that is the Gaussian, the Laplace, and a simplified Laplace approximation. Although the Laplace approximation is preferred in general, the much smaller cost of the simplified Laplace generally compensates for the slight loss in accuracy.

3.2.1 Using Gaussian Approximations

The simplest (and cheapest) approximation to $\pi(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ is the Gaussian approximation $\widetilde{\pi}_{\mathrm{G}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$, where the mean $\mu_i(\boldsymbol{\theta})$ and the marginal variance $\sigma_i^2(\boldsymbol{\theta})$ are derived using the recursions (5), and possibly correcting for linear constraints. During the exploration of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, see Section 3.1, we already compute $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$, so only marginal variances need to be additionally computed. The Gaussian approximation gives often reasonable results, but there can be errors in the location and/or errors due to the lack of skewness (Rue and Martino, 2007).

3.2.2 Using Laplace Approximations

The natural way to improve the Gaussian approximation is to compute the Laplace approximation

$$\widetilde{\pi}_{\mathrm{LA}}(x_i \mid \boldsymbol{\theta}, \boldsymbol{y}) \propto \left. \frac{\pi(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})}{\widetilde{\pi}_{\mathrm{GG}}(\boldsymbol{x}_{-i}|x_i, \boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x}_{-i}=\boldsymbol{x}^\star_{-i}(x_i,\boldsymbol{\theta})}. \tag{11}$$

Here, $\widetilde{\pi}_{\mathrm{GG}}$ is the Gaussian approximation to $\boldsymbol{x}_{-i}|x_i, \boldsymbol{\theta}, \boldsymbol{y}$, and $\boldsymbol{x}^\star_{-i}(x_i, \boldsymbol{\theta})$ is the modal configuration. Note that $\widetilde{\pi}_{\mathrm{GG}}$ is different from the conditional density corresponding to $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$.

Unfortunately, (11) implies that $\widetilde{\pi}_{\mathrm{GG}}$ must be recomputed for each value of x_i and $\boldsymbol{\theta}$, since its precision matrix depends on i and $\boldsymbol{\theta}$. This is far too expensive, as it requires n factorisations of the full precision matrix. We propose two modifications to (11) which makes it computationally feasible.

Our first modification consists in avoiding the optimisation step when computing $\widetilde{\pi}_{\mathrm{GG}}(\boldsymbol{x}_{-i}|x_i, \boldsymbol{\theta}, \boldsymbol{y})$ by approximating the modal configuration,

$$\boldsymbol{x}^\star_{-i}(x_i, \boldsymbol{\theta}) \approx \mathrm{E}_{\widetilde{\pi}_{\mathrm{G}}}(\boldsymbol{x}_{-i} \mid x_i). \tag{12}$$

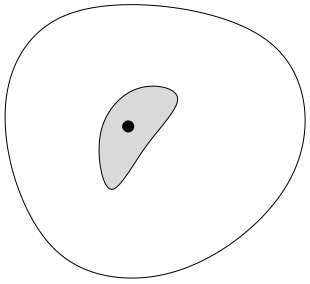


Figure 2: Illustration of the region of interest $R_i(\boldsymbol{\theta})$. The outer circle illustrates the graph of the GMRF, whereas the black dot indicates the node of interest. The conditional expectation (13) locates the nodes that are affected by a change in x_i, that is all the nodes in the grey region.

The right-hand side is evaluated under the conditional density derived from the Gaussian approximation $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$. The computational benefit is immediate. First, the conditional mean can be computed by a rank one update from the unconditional mean, using (6). In the spatial case the cost is $\mathcal{O}(n \log n)$, for each i, which comes from solving $\boldsymbol{Q}\boldsymbol{v} = \mathbf{1}_i$, where $\mathbf{1}_i$ equals one at position i, and zero otherwise. This rank one update is computed only once for each i, as it is linear in x_i. Although their settings are slightly different, Hsiao et al. (2004) show that deviating from the conditional mode does not necessarily degrade the approximation error. Another positive feature of using (12) is that the conditional mode is continuous with respect to x_i, a feature which does not hold in practice when numerical optimisation is used to compute $\boldsymbol{x}^{\star}_{-i}(x_i, \boldsymbol{\theta})$.

Our next modification materialises the following intuition: only those x_j that are 'close' to x_i should have an impact on the marginal of x_i. Figure 2 illustrates this idea. The graph of $\boldsymbol{x}$ is represented by the larger circle. The node i is marked with a black dot. If the dependency between x_j and x_i decays as the distance between nodes i and j increases, only those x_j's in the grey region are of interest regarding the marginal of x_i. Denote by $R_i(\boldsymbol{\theta})$ the 'region of interest' regarding the marginal of x_i. The conditional expectation in (12) implies that

$$\frac{\mathrm{E}_{\widetilde{\pi}_{\mathrm{G}}}(x_j|x_i) - \mu_j(\boldsymbol{\theta})}{\sigma_j(\boldsymbol{\theta})} = a_{ij}(\boldsymbol{\theta}) \, \frac{x_i - \mu_i(\boldsymbol{\theta})}{\sigma_i(\boldsymbol{\theta})} \tag{13}$$

for some $a_{ij}(\boldsymbol{\theta})$ when $j \neq i$. Hence, a simple rule for constructing the set $R_i(\boldsymbol{\theta})$ is

$$R_i(\boldsymbol{\theta}) = \{j \ : \ |a_{ij}(\boldsymbol{\theta})| > 0.001\} \, . \tag{14}$$

The most important computational saving using $R_i(\boldsymbol{\theta})$ comes from the calculation of the denominator of (11), where we now only need to factorise a $|R_i(\boldsymbol{\theta})| \times |R_i(\boldsymbol{\theta})|$ sparse matrix.

Expression (11), simplified as explained above, must be computed for different values of x_i in order to find the density. To select these points, we use the mean and variance of the Gaussian approximation (2), and choose, say, different values for the standardised variable

$$x_i^{(s)} = \frac{x_i - \mu_i(\boldsymbol{\theta})}{\sigma_i(\boldsymbol{\theta})} \tag{15}$$

according to the corresponding choice of abscissas given by the Gauss-Hermite quadrature rule. To represent the density $\widetilde{\pi}_{\text{LA}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$, we use

$$\widetilde{\pi}_{\text{LA}}\left(x_i \mid \boldsymbol{\theta}, \boldsymbol{y}\right) \propto \mathcal{N}\left\{x_i;\ \mu_i(\boldsymbol{\theta}),\ \sigma_i^2(\boldsymbol{\theta})\right\} \times \exp\left\{\text{cubic spline}(x_i)\right\}. \tag{16}$$

The cubic spline is fitted to the difference of the log-density of $\widetilde{\pi}_{\text{LA}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ and $\widetilde{\pi}_{\text{G}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ at the selected abscissa points, and then the density is normalised using quadrature integration.

3.2.3 Using a Simplified Laplace Approximation

In this section we derive a simplified Laplace approximation $\widetilde{\pi}_{\text{SLA}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ by doing a series expansion of $\widetilde{\pi}_{\text{LA}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ around $x_i = \mu_i(\boldsymbol{\theta})$. This allows us to correct the Gaussian approximation $\widetilde{\pi}_{\text{G}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ for location and skewness. For many observational models including the Poisson and the Binomial, these corrections are sufficient to obtain essentially correct posterior marginals. The benefit is purely computational: as most of the terms are common for all i, we can compute all the n marginals in only $\mathcal{O}(n^2 \log n)$ time.

Define

$$d_j^{(3)}(x_i, \boldsymbol{\theta}) = \frac{\partial^3}{\partial x_j^3} \log \pi(y_j \mid x_j, \boldsymbol{\theta})\bigg|_{x_j = \text{E}_{\widetilde{\pi}_{\text{G}}}(x_j|x_i)}$$

which we assume exists. The evaluation point is found from (13). The following trivial Lemma will be useful.

Lemma 1 *Let* $\boldsymbol{x} = (x_1, \ldots, x_n)^T \ \sim\ \mathcal{N}(\mathbf{0},\ \boldsymbol{\Sigma})$*, then for all* x_1

$$-\frac{1}{2}(x_1,\ E(\boldsymbol{x}_{-1}|x_1)^T)\ \boldsymbol{\Sigma}^{-1} \begin{pmatrix} x_1 \\ E(\boldsymbol{x}_{-1}|x_1) \end{pmatrix} = -\frac{1}{2} x_1^2 / \Sigma_{11}.$$

We expand the numerator and denominator of (11) around $x_i = \mu_i(\boldsymbol{\theta})$, using (12) and Lemma 1. Up to third order, we obtain

$$\begin{aligned} \log \pi(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})\bigg|_{\boldsymbol{x}_{-i} = \text{E}_{\widetilde{\pi}_{\text{G}}}(\boldsymbol{x}_{-i}|x_i)} &= -\frac{1}{2}(x_i^{(s)})^2 \\ &\quad + \frac{1}{6}(x_i^{(s)})^3 \sum_{j \in \mathcal{I} \setminus i} d_j^{(3)}\left(\mu_i(\boldsymbol{\theta}), \boldsymbol{\theta}\right)\ \left\{\sigma_j(\boldsymbol{\theta}) a_{ij}(\boldsymbol{\theta})\right\}^3 + \cdots \end{aligned} \tag{17}$$

The first and second order terms give the Gaussian approximation, whereas the third order term provides a correction for skewness. Further, the denominator of (11) reduces to

$$\log \widetilde{\pi}_{\text{GG}}(\boldsymbol{x}_{-i} \mid x_i, \boldsymbol{\theta}, \boldsymbol{y})\bigg|_{\boldsymbol{x}_{-i} = \text{E}_{\widetilde{\pi}_{\text{G}}}(\boldsymbol{x}_{-i}|x_i)} = \text{constant} + \frac{1}{2} \log \left|\boldsymbol{H} + \text{diag}\left\{\boldsymbol{c}(x_i, \boldsymbol{\theta})\right\}\right| \tag{18}$$

where $\boldsymbol{H}$ is the prior precision matrix of the GMRF with ith column and row deleted, and $\boldsymbol{c}(x_i, \boldsymbol{\theta})$ is the vector of minus the second derivative of the log likelihood evaluated at $x_j = \text{E}_{\widetilde{\pi}_{\text{G}}}(x_j|x_i)$, see Section 2.2. Using that

$$d \log \left|\boldsymbol{H} + \text{diag}(\boldsymbol{c})\right| = \sum_j \left[\left\{\boldsymbol{H} + \text{diag}\left(\boldsymbol{c}\right)\right\}^{-1}\right]_{jj}\ dc_j$$

we obtain

$$\log \widetilde{\pi}_{\text{GG}}(\boldsymbol{x}_{-i} \mid x_i, \boldsymbol{\theta}, \boldsymbol{y})\Big|_{\boldsymbol{x}_{-i}=\text{E}_{\widetilde{\pi}_{\text{G}}}(\boldsymbol{x}_{-i}|x_i)} = \text{constant} - \frac{1}{2}x_i^{(s)} \sum_{j\in\mathcal{I}\setminus i} \text{Var}_{\widetilde{\pi}_{\text{G}}}(x_j|x_i)\ \ d_j^{(3)}(\mu_i(\boldsymbol{\theta}), \boldsymbol{\theta})\ \sigma_j(\boldsymbol{\theta})\ a_{ij}(\boldsymbol{\theta}) + \cdots . \tag{19}$$

For Gaussian data (18) is just a constant, so the first order term in (19) is the first correction for non-Gaussian observations. Note that

$$\text{Var}_{\widetilde{\pi}_{\text{G}}}(x_j|x_i) = \sigma_j^2(\boldsymbol{\theta}) \left\{1 - \text{Corr}_{\widetilde{\pi}_{\text{G}}}(x_i, x_j)^2\right\}$$

but the correlation between x_i and x_j is only available for some of the i's and j's. This is because the marginal variances are computed using (5). We approach this problem by simply replacing all correlations not computed by a default value, say 0.05.

We now collect the expansions (17) and (19). Define

$$\begin{aligned}
\gamma_i^{(1)}(\boldsymbol{\theta}) &= \frac{1}{2} \sum_{j\in\mathcal{I}\setminus i} \sigma_j^2(\boldsymbol{\theta}) \left\{1 - \text{Corr}_{\widetilde{\pi}_{\text{G}}}(x_i, x_j)^2\right\}\ \ d_j^{(3)}(\mu_i(\boldsymbol{\theta}), \boldsymbol{\theta})\ \sigma_j(\boldsymbol{\theta})\ a_{ij}(\boldsymbol{\theta}) \\
\gamma_i^{(3)}(\boldsymbol{\theta}) &= \sum_{j\in\mathcal{I}\setminus i} d_j^{(3)}(\mu_i(\boldsymbol{\theta}), \boldsymbol{\theta})\ \ \{\sigma_j(\boldsymbol{\theta}) a_{ij}(\boldsymbol{\theta})\}^3
\end{aligned} \tag{20}$$

then

$$\log \widetilde{\pi}_{\text{SLA}}(x_i^s|\boldsymbol{\theta}, \boldsymbol{y}) = \text{constant} - \frac{1}{2}(x_i^{(s)})^2 + \gamma_i^{(1)}(\boldsymbol{\theta}) x_i^{(s)} + \frac{1}{6}(x_i^{(s)})^3 \gamma_i^{(3)}(\boldsymbol{\theta}) + \cdots . \tag{21}$$

Eq. (21) does not define a density as the third order term is unbounded. A common way to introduce skewness into the Gaussian distribution is to use the Skew-Normal distribution (Azzalini and Capitanio, 1999)

$$\pi_{\text{SN}}(z) = \frac{2}{\omega}\phi\left(\frac{z-\xi}{\omega}\right)\ \ \Phi\left(a\frac{z-\xi}{\omega}\right) \tag{22}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the density and distribution function of the standard normal distribution, and ξ, $\omega > 0$, and a are respectively the location, scale, and skewness parameters. We fit a Skew-Normal density to (21) so that the third derivative at the mode is $\gamma_i^{(3)}$, the mean is $\gamma_i^{(1)}$ and the variance is 1. In this way, $\gamma_i^{(3)}$ only contributes to the skewness whereas the adjustment in the mean comes from $\gamma_i^{(1)}$; see Appendix for details.

We have implicitly assumed that the expansion (17) is dominated by the third order term. This is adequate when the log-likelihood is skewed, but not for symmetric distributions with thick tails like a Student-t_ν with a low degree of freedom. For such cases, we expand only the denominator (19) and fit the spline-corrected Gaussian (16), instead of a skewed Normal. This is slightly more expensive, but is needed.

The simplified Laplace approximation appears to be highly accurate for many observational models. The computational cost is dominated by the calculation of vector $a_{i\cdot}(\boldsymbol{\theta})$,

for each i; thus the 'region of interest' strategy (14) is unhelpful here. Most of the other terms in (20) do not depend on i, and thus are computed only once. The cost for computing (21), for a given i, is of the same order as the number of non-zero elements of the Cholesky triangle, e.g. $\mathcal{O}(n \log n)$ in the spatial case. Repeating the procedure n times gives a total cost of $\mathcal{O}(n^2 \log n)$ for each value of $\boldsymbol{\theta}$. We believe this is close to the lower limit for any general algorithm that approximates all of the n marginals. Since the graph of $\boldsymbol{x}$ is general, we need to visit all other sites, for each i, for a potential contribution. This operation alone costs $\mathcal{O}(n^2)$.

4 Approximation error: Asymptotics and practical issues

4.1 Approximation error of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$

To simplify the discussion, we assume that the dimension of the observations $\boldsymbol{y}$, n_d, equals the dimension of the latent field $\boldsymbol{x}$, n, so that each node x_i is observed as y_i. Equation (1) can be rewritten as

$$\left\{\frac{\widetilde{\pi}_u(\boldsymbol{\theta} \mid \boldsymbol{y})}{\pi(\boldsymbol{\theta} \mid \boldsymbol{y})}\right\}^{-1} \propto |\boldsymbol{Q}^\star(\boldsymbol{\theta})|^{1/2} \int \exp\left[-\tfrac{1}{2}\left\{\boldsymbol{x}-\boldsymbol{x}^\star(\boldsymbol{\theta})\right\}^T \boldsymbol{Q}^\star \left\{\boldsymbol{x}-\boldsymbol{x}^\star(\boldsymbol{\theta})\right\} + r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})\right] d\boldsymbol{x} \quad (23)$$
$$\propto \mathrm{E}_{\widetilde{\pi}_\mathrm{G}}\left[\exp\left\{r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})\right\}\right]$$

where $\widetilde{\pi}_u(\boldsymbol{\theta}|\boldsymbol{y})$ is the unnormalised version of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, $\boldsymbol{x}^\star(\boldsymbol{\theta})$ and $\boldsymbol{Q}^\star(\boldsymbol{\theta})$ are the mean and variance of Gaussian distribution $\widetilde{\pi}_\mathrm{G}$, $r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y}) = \sum_i h_i(x_i)$, and $h_i(x_i)$ is $g_i(x_i)$ minus its Taylor expansion up to order two around $x_i^\star(\boldsymbol{\theta})$, see (7) and (8). The approximation $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is based on a Taylor expansion of order 2, but higher orders can also be computed. Denote by $\boldsymbol{S}(\boldsymbol{\theta}) = (s_{ij}(\boldsymbol{\theta}))$ the inverse of $\boldsymbol{Q}^\star(\boldsymbol{\theta})$. Straightforward calculations show that

$$\begin{aligned}\widetilde{\widetilde{\pi}}_u(\boldsymbol{\theta} \mid \boldsymbol{y}) \;=\; \widetilde{\pi}_u(\boldsymbol{\theta} \mid \boldsymbol{y}) \Bigg[1 &+ \frac{1}{8}\sum_{i=1}^n s_{ii}(\boldsymbol{\theta})^2 \frac{\partial^4 g_i(x_i^\star(\boldsymbol{\theta}))}{\partial x_i^4} + \frac{5}{24}\sum_{i=1}^n \left\{\frac{\partial^3 g_i(x_i^\star(\boldsymbol{\theta}))}{\partial x_i^3}\right\}^2 s_{ii}(\boldsymbol{\theta})^3 \\ &+ \frac{1}{24}\sum_{i\neq j}^n \frac{\partial^3 g_i(x_i^\star(\boldsymbol{\theta}))}{\partial x_i^3}\frac{\partial^3 g_j(x_j^\star(\boldsymbol{\theta}))}{\partial x_j^3} s_{ij}(\boldsymbol{\theta})\left\{2s_{ij}(\boldsymbol{\theta})^2 + 3s_i(\boldsymbol{\theta})s_j(\boldsymbol{\theta})\right\} \Bigg]^{-1}\end{aligned}$$

corresponds to an expansion up to order 7: odd orders produce null coefficients, and order 4 and 6 give the second term, and the two last terms, respectively. The density above is not necessarily positive, but if both approximations are close, this seems an indication that both are accurate. We discuss only $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ from now on.

For sake of exposition, denote p the dimension of integral (23), although $p = n$ in our case. Under standard assumptions, in particular when p is fixed, this integral is $1 + \mathcal{O}(n^{-1})$; see e.g. Tierney and Kadane (1986). Shun and McCullagh (1995) consider the case where p grows with n, but do not establish rigorously the error rate. It does not seem possible in our settings to prove that the multiplicative error is always $o(1)$ with respect to n. For instance, if the x_i's are independent, it is possible to exhibit cases where the error, for any expectation evaluated with respect to the marginal posterior, is $\mathcal{O}(1)$ but not $o(1)$. (More details are available on request.) This discussion is complicated by the difficulty

of defining asymptotics in spatial models: observations may be generated in a larger and larger domain (increasing domain asymptotics), in a fixed volume (infill asymptotics), and other asymptotic schemes could be devised. Instead, we propose heuristic arguments for explaining the good accuracy observed in practical applications.

Remark 1 The 'actual' dimensionality of (23) is typically much smaller than n. Because of the dependency within $\boldsymbol{x}$, $\boldsymbol{x}$ is well approximated by its q principal components, with $q \ll n$. A convenient measure of the dimensionality of (23) is Spiegelhalter et al. (2002)'s measure for the *effective number of parameters*, p_{D}. In the case of approximately Gaussian models, then

$$p_{\mathrm{D}}(\boldsymbol{\theta}) \approx \text{Trace}\left\{\boldsymbol{Q}(\boldsymbol{\theta})\,\boldsymbol{Q}^{\star}(\boldsymbol{\theta})^{-1}\right\}, \tag{24}$$

the trace of the prior precision matrix times the by posterior covariance matrix of the Gaussian approximation. The quantity $p_{\mathrm{D}}(\boldsymbol{\theta})$ indicates how informative the data is, and to which extent the Gaussianity and the dependence structure of the prior are preserved in the posterior of $\boldsymbol{x}$, given $\boldsymbol{\theta}$. The calculation of $p_{\mathrm{D}}(\boldsymbol{\theta})$ is cheap, since the covariances of neighbours are obtained as a by-product of the computation of the marginal variances in the Gaussian approximation based on (5).

Remark 2 The approximation error is reduced through normalisation, provided (23) is roughly constant with respect to $\boldsymbol{\theta}$ within the support of the true marginal. For the Laplace approximation with standard assumptions, renormalisation improves the relative error from $\mathcal{O}(n^{-1})$ to $\mathcal{O}(n^{-3/2})$ (Tierney and Kadane, 1986).

Remark 3 The high accuracy of our approximation which we obtain in the experiments in Section 5, seems to be due both to the Gaussian latent field and the well-behaved observational models usually considered in applications, e.g. an exponential family distribution for $\pi(y_i|x_i,\boldsymbol{\theta})$.

Following these remarks, a more direct way to assess the approximation error is simply to evaluate the order of magnitude of $r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})$: simulate independent samples $\{\boldsymbol{x}^j\}$ from $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})$, and compute the 0.025 lower and upper quantiles of the empirical distribution of the $r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})$'s. This is a rather quick procedure. The first term in the exponential defining the integrand in (23) is distributed according to $\chi^2_n/2$, so we consider the remainder $r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})$ to be small if quantiles are in absolute value much less than n. In the same way, empirical averages of the $r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})$, for different values of $\boldsymbol{\theta}$, can be used to determine the variability of (23) with respect to $\boldsymbol{\theta}$.

4.2 Approximation error of $\widetilde{\pi}_{\mathbf{LA}}(x_i|\boldsymbol{\theta},\boldsymbol{y})$

The approximation error of $\widetilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{\theta},\boldsymbol{y})$ admits a similar expression to that of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. To see this, consider an alternative structure for the model, where the node x_i becomes an additional component of the parameter $\boldsymbol{\theta}$, and the latent field is therefore $\boldsymbol{x}_{-i}$; then, the same manipulations as for (23) leads eventually to

$$\left\{\frac{\widetilde{\pi}_{LA,u}(x_i|\boldsymbol{\theta},\boldsymbol{y})}{\pi(x_i|\boldsymbol{\theta},\boldsymbol{y})}\right\}^{-1} \propto \mathrm{E}_{\widetilde{\pi}_{\mathrm{GG}}}\left[\exp\left\{r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})\right\}\right]$$

where $\widetilde{\pi}_{LA,u}(x_i|\boldsymbol{\theta},\boldsymbol{y})$ is the unnormalised version of $\widetilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{\theta},\boldsymbol{y})$. Thus, we essentially obtain the same result as in Section 4.1; before normalisation, the approximation error of $\widetilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{\theta},\boldsymbol{y})$ is comparable to that of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, and the assessment criteria proposed in the previous section are also good indicators of the accuracy of $\widetilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{\theta},\boldsymbol{y})$. Note however that normalisation has a different effect on $\widetilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{\theta},\boldsymbol{y})$. Under increasing domain asymptotics, new components $x_{i'}$ are generated further and further from x_i, so at some point the additional terms $h_{i'}(x_{i'})$ in the expression of the remainder $r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})$ should be a constant with respect to x_i, and therefore should be cancelled by normalisation. Thus, we conjecture that the error is at worst $\mathcal{O}(1)$ under increasing domain asymptotics.

4.3 Assessing the approximation error

Obviously, there is only one way to assess with certainty the approximation error of our approach, which is to run an MCMC sampler for an infinite time. However, we propose to use the following two strategies to assess the approximation error, which should be reasonable in most situations.

Our first strategy is to verify the overall approximation $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})$, for each $\boldsymbol{\theta}_k$ used in the integration. We do this by computing $p_{\mathrm{D}}(\boldsymbol{\theta})$ (24), and the lower and upper $\alpha/2$ quantiles in the empirical distribution for the remainder $r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})$. If p_{D} is small compared to n, and the quantiles of $r(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{y})$ are in absolute value much less than n, this provides strong confidence that the Gaussian approximation is an adequate approximation.

Our second strategy is based on the simple idea of comparing elements of a sequence of more and more accurate approximations. In our case, this sequence consists of the Gaussian approximation (2), followed by the simplified Laplace approximation (21), then by the Laplace approximation (11). Specifically we compute the integrated marginal (3) based on both the Gaussian approximation and the simplified Laplace approximation, and compute their (symmetric) Kullback-Leibler divergence (KLD). If the divergence is small then both approximations are considered as acceptable. Otherwise, compute (3) using the Laplace approximation (11) and compute the divergence with the one based on the simplified Laplace approximation. Again, if the divergence is small, simplified Laplace and Laplace approximations appear to be acceptable; otherwise, the Laplace approximation is our best estimate but the label 'problematic' should be attached to the approximation to warn the user. (This last option has not yet happened to us.)

To assess the error due to the numerical integration (3), we can compare the KLD between the posterior marginals obtained with a standard and those obtained with a higher resolution. As a such approach is standard in numerical integration, we do not pursue this issue here.

5 Examples

This section provides examples of applications of the INLA approach, with comparisons to results obtained from intensive MCMC runs. Comparisons are expressed in terms of computational time; computations were performed on a 2.1GHz laptop, and programmed in C. We start with simple examples with fixed $\boldsymbol{\theta}$ in Section 5.1 and Section 5.2, to verify

the (simplified) Laplace approximation for $x_i|\boldsymbol{\theta}, \boldsymbol{y}$. We continue with a stochastic volatility model applied to exchange rate data in Section 5.3 and a spatial semi-parametric ecological regression problem in Section 5.4. The dimensions gets really large in Section 5.5, in which we analyse some data using a spatial log-Gaussian Cox process.

5.1 Simple simulated examples

We start by illustrating the various approximations of $\pi(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ in two quite challenging examples. The first model is based on a first order auto-regressive latent field with unknown mean,

$$\eta_t - \mu \mid \eta_1, \ldots, \eta_{t-1}, \mu \;\sim\; \mathcal{N}\left\{\phi(\eta_{t-1} - \mu),\; \sigma^2\right\}, \qquad t = 1, \ldots, 50 \tag{25}$$

where $\mu \sim \mathcal{N}(0, 10)$, $\phi = 0.85$ and $\text{Var}(\eta_t) = 1$. As our observations we take

$$y_t - \eta_t \mid (\boldsymbol{\eta}, \mu) \;\sim\; \text{Student-}t_3 \quad \text{and} \quad y_t \mid (\boldsymbol{\eta}, \mu) \;\sim\; \text{Bernoulli}\left\{\text{logit}^{-1}(\eta_t)\right\}$$

for $t = 1, \ldots, 50$, in both experiments. Note that the Student-t_3 is symmetric so we use the full numerator in the simplified Laplace approximations as described in Section 3.2.3.

To create the observations, we sampled first $\boldsymbol{x} = (\boldsymbol{\eta}^T, \mu^T)^T$ from the prior, then simulated the observations. We computed $\widetilde{\pi}(\eta_t|\boldsymbol{\theta}, \boldsymbol{y})$ for $t = 1, \ldots, 50$ and $\widetilde{\pi}(\mu|\boldsymbol{\theta}, \boldsymbol{y})$ using the simplified Laplace approximation and located the node with maximum Kullback-Leibler divergence (KLD) between the Gaussian and the simplified Laplace approximations. This process was repeated 100 times, and the realisation with the largest maximum KLD was selected. Figure 3 displays the results for the Student-t_3 data (first column) and the Bernoulli data (second column). Panel (a) and (b) display $\boldsymbol{\eta}$ (solid line) and the observed data (circles). In (a) the node with the maximum KLD is marked with a vertical line and solid dot. In (b) the node with the maximum KLD is μ hence not shown. Panel (c) and (d) display the approximated marginals for the node with maximum KLD in the standardised scale (15). The dotted line is the Gaussian approximation, the dashed line is the simplified Laplace and the solid line is the Laplace approximation. In both cases, the simplified Laplace and the Laplace approximation are very close to each other. The KLD between the Gaussian approximation and the simplified Laplace one is 0.20 and 0.05, respectively. The KLD between the simplified Laplace approximation and the Laplace one is 0.001 and 0.0004. Panel (e) and (f) show the simplified Laplace approximation with a histogram based on $10,000$ (near) independent samples from $\pi(\boldsymbol{\eta}, \mu|\boldsymbol{\theta}, \boldsymbol{y})$. The fit is excellent.

The great advantage of the Laplace approximations is the high accuracy and low computational cost. In both examples, we computed all the approximations (for each experiment) in less than 0.08 seconds, whereas the MCMC samples required about 25 seconds.

The results shown in this example are rather typical and are not limited to simple time-series models like (25). The Laplace approximation only 'sees' the log-likelihood model and then uses some of the other nodes (see Figure 2) to compute the correction to the Gaussian approximation. Hence, the form of the log-likelihood is more important than the form of the covariance for the latent field. We expect similar results for spatial or spatio-temporal latent Gaussian models, with Student-t_ν or Binomial observations.

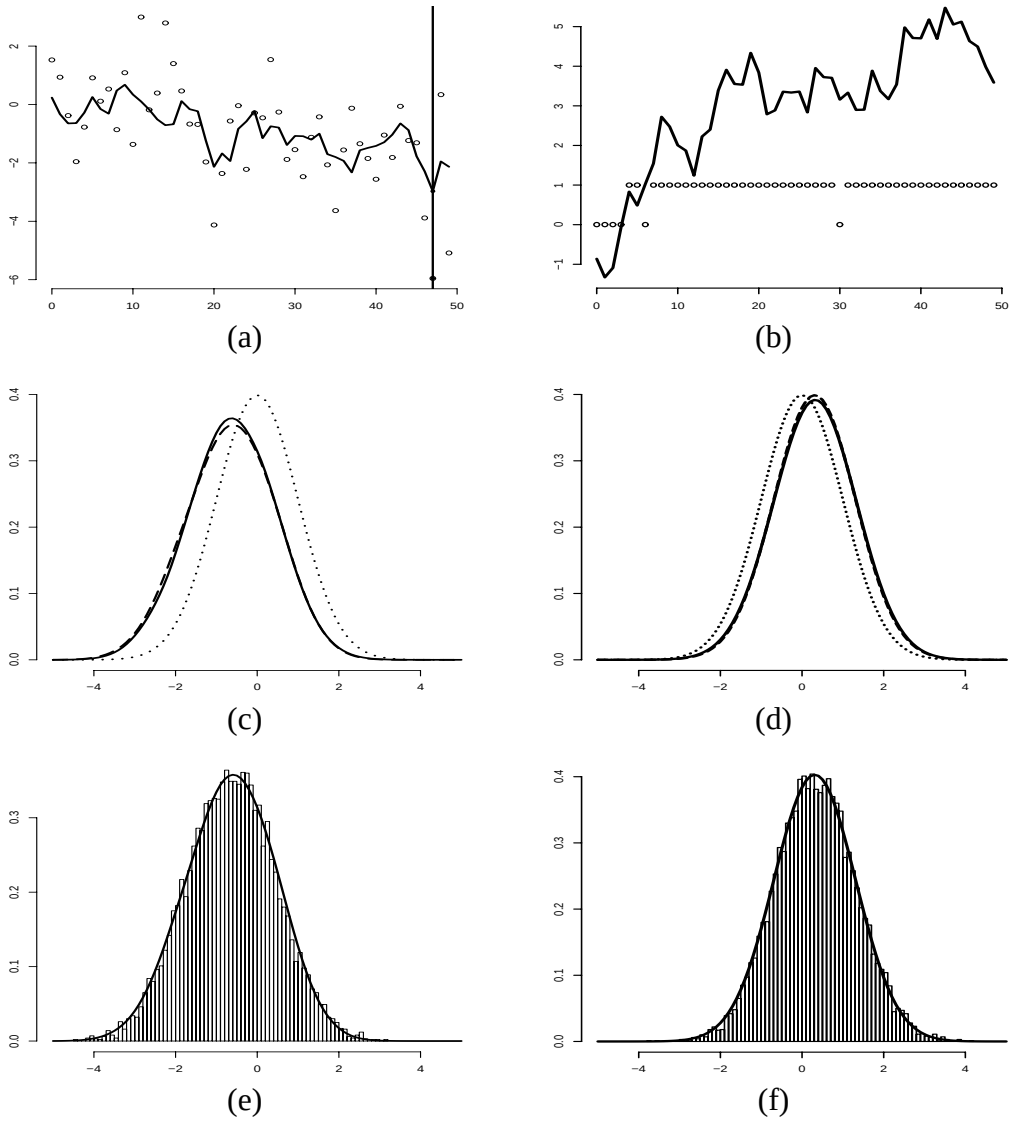


Figure 3: First row shows the true latent Gaussian field (solid line), the observed Student-t_3 data and Bernoulli data (dots). Second row shows the approximate marginal for a selected node using various approximations; Gaussian (dotted), simplified Laplace (dashed) and Laplace (solid). Last row compares samples from a long MCMC chain with the marginal computed with the simplified Laplace approximation.

5.2 Bayesian multiscale analysis for time series data

This example shows a situation where it is useful to have estimates of the marginals with a relative error, so that even the tails can be evaluated accurately. We extend the Bayesian multiscale tool for exploratory analysis of time series data by Øigård et al. (2006) to allow for non-Gaussian observations. The fundamental problem is to detect significant features and important structures of a signal observed with noise. Although a noisy signal can be smoothed, often some of the features are visible only on certain scales, and may disappear if the smoothing is too severe. The multiscale idea consists in considering several levels of smoothing simultaneously. Chaudhuri and Marron (1999) introduced such ideas in nonparametric function estimation in the form of the SIZer methodology (SIgnificant ZERo crossings of derivatives), see also Erästö (2005).

Let $\eta(t)$ be the unknown continuous underlying signal with derivatives $\eta'(t)$, and level of smoothing κ. Let $\boldsymbol{y} = (y_1, \ldots, y_n)^T$ be observations of $\eta(\cdot)$ at time-points $\boldsymbol{t} = (t_1, \ldots, t_n)^T$. The derivative is said to be 'significant' positive at time point t, if

$$\text{Prob}(\eta'(t) > 0 \mid \boldsymbol{y}, \kappa) > 1 - \alpha/2$$

and similarly significant negative, where α is the level of significance. The SIZer map displays regions of significant positive and negative gradients for various levels of smoothing κ.

We now illustrate how to use the simplified Laplace approximation to compute the SIZer map. We use gamma ray burst intensity data previously analysed by Besbeas et al. (2004); the observations are Poisson:

$$y(t_i) \;\sim\; \text{Poisson}\left\{\exp(\eta(t_i)\right\}, \quad t_i = i \quad \text{for } i = 1, \ldots, n = 512$$

where $\eta(t)$ is the latent Gaussian process. The data are displayed in Figure 4(a). We follow Øigård et al. (2006), and model the continuous process $\eta(t)$ as an integrated Wiener process with precision κ. Wecker and Ansley (1983) show that the integrated Wiener process is Markov if augmented with the derivatives $\eta'(t)$,

$$\begin{pmatrix}\eta(t_{i+1})\\ \eta'(t_{i+1})\end{pmatrix} \Bigg| \left\{ \begin{pmatrix}\eta(s)\\ \eta'(s)\end{pmatrix}, s \le t_i \right\}, \kappa \;\sim\; \mathcal{N}\left\{ \begin{pmatrix}1 & \delta_i\\ 0 & 1\end{pmatrix} \begin{pmatrix}\eta(t_i)\\ \eta'(t_i)\end{pmatrix}, \frac{1}{\kappa} \begin{pmatrix}\delta_i^3/3 & \delta_i^2/2\\ \delta_i^2/2 & \delta_i\end{pmatrix} \right\}$$

where $\delta_i = t_{i+1} - t_i$. Hence, the discretely observed integrated Wiener process $\boldsymbol{x} = (\{\eta(t_i)\}, \{\eta'(t_i)\})^T$ is a GMRF of dimension $2n$, see Rue and Held (2005, Sec. 3.5). Note that the derivatives at t_i are a part of the GMRF, hence we can approximate their marginal densities (for a fixed κ) and check whether they are significant negative or positive.

We use the simplified Laplace approximation and compute all the $2n$ marginals for $\log \kappa = 1, \ldots, 15$. This takes about 0.35 seconds for each value of $\log \kappa$. The posterior means of $\{\eta(t_i)\}$ and the SIZer map for $\alpha = 0.05$ are displayed in Figure 4(b) and (c), respectively. In the SIZer map, white indicates significant positive derivative, black indicates significant negative derivative whereas grey indicates none. The vertical scale in the SIZer map goes from $\log \kappa = 1$ to $\log \kappa = 15$.

To verify the results we ran a MCMC sampler for nine hours and estimated the probability for the chain to be below the $\alpha/2$ (where $\alpha = 0.05$) quantiles as computed from our

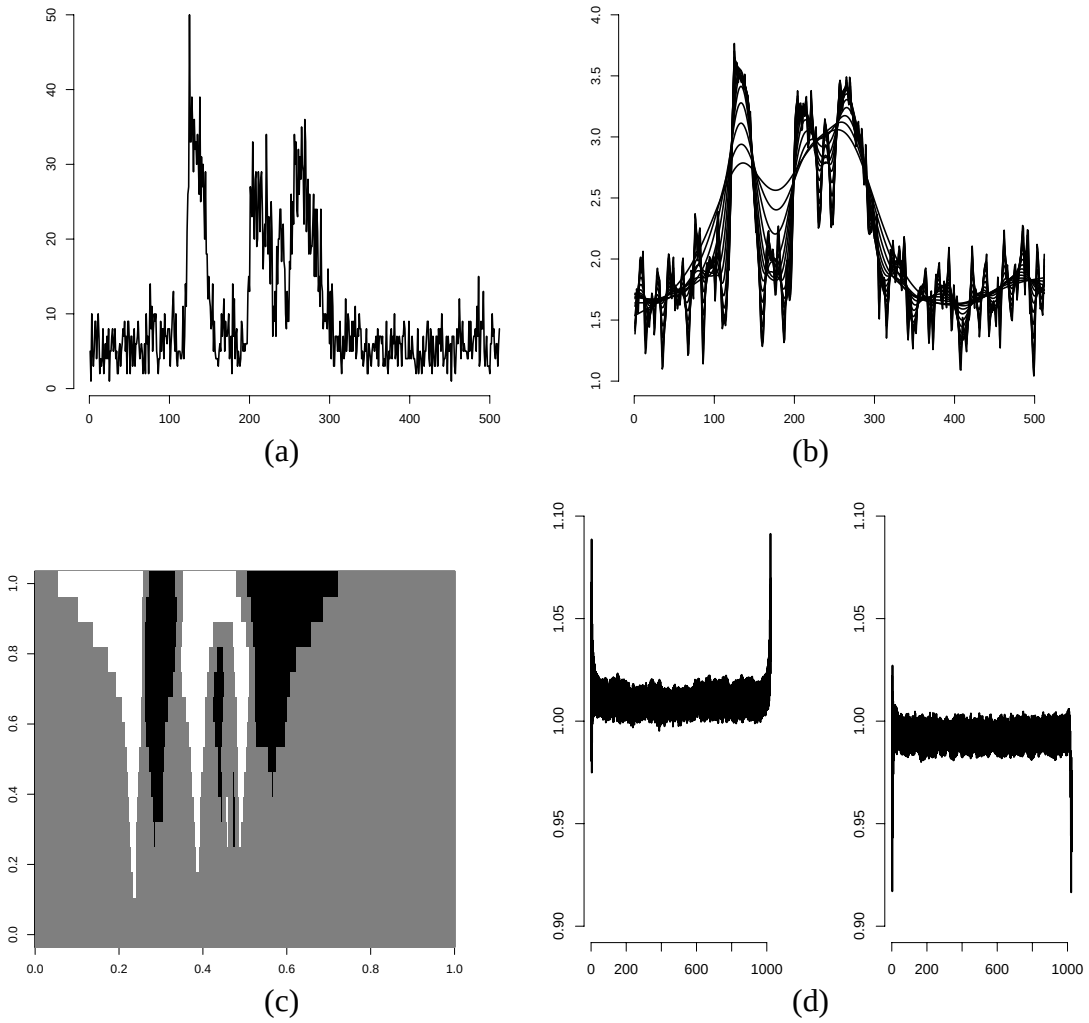


Figure 4: Results for the multiscale analysis example. Panel (a) displays the raw burst data. Panel (b) displays the posterior means for varying degree of smoothing. Panel (c) shows the SIZer map for $\alpha = 0.05$. Panel (d) displays the ratios between the estimated and the real probability of being below the approximate quantiles, lower quantiles on the left and upper quantiles on the right.

approximation $\widetilde{\pi}(\eta'(t_i)|\kappa, \boldsymbol{y})$. Panel (d) displays the ratios of these estimated probabilities and the true value $\alpha/2$ for all the 1024 nodes is the Markov field. Lower (resp. upper) quantiles are displayed on the left (resp. right). In both cases the error is largest at the two ends. The average ratio is 1.01 for the lower quantiles and 0.99 for the upper quantiles, so the average absolute approximation bias is 0.00025. This is indeed impressive; recall that the simplified Laplace approximation fits the skew-Normal parametric density.

Hannig and Marron (2006) develop some theory for computing more accurately (asymptotically) the SIZer map for nonparametric regression. The Bayesian approach taken here does not resort to asymptotic theory, can deal with non-Gaussian observations, can take into account covariates and unstructured effects and so on. The calculations can be done exactly for Gaussian observation models, and, as illustrated here, practically exactly for common non-Gaussian observation models.

5.3 Stochastic volatility models

Stochastic volatility models are frequently used to analyse financial time series. Figure 5(a) displays the log of the daily difference of the pound-dollar exchange rate from October 1st, 1981, to June 28th, 1985. This dataset has been analysed by Durbin and Koopman (2000), among others. There has been much interest in developing efficient MCMC methods for such models, e.g. Shephard and Pitt (1997) and Chib et al. (2002).

Following Durbin and Koopman (2000), we consider a first order auto-regressive latent Gaussian process

$$x_t \mid x_1, \ldots, x_{t-1}, \tau, \phi \;\sim\; \mathcal{N}\left(\phi x_{t-1}, 1/\tau\right),$$

where $|\phi| < 1$ to ensure stationarity. The observations are taken to be

$$y_t \mid x_1, \ldots, x_t, \kappa \;\sim\; \mathcal{N}\left\{0,\, \exp(x_t)/\kappa\right\} \tag{26}$$

where κ is an unknown precision. The log-likelihood (with respect to κ) is quite far from being Gaussian and is non-symmetric. There is some evidence that financial data have heavier tails than the Gaussian, so a Student-t_ν distribution with unknown degrees of freedom can be substituted to the Gaussian in (26); see Chib et al. (2002). We consider this modified model at the end of this example.

We display the results for the simplified Laplace approximation of the posterior marginals of the three unknown hyperparameters (properly transformed so that $\boldsymbol{\theta} \in \mathbb{R}^3$):

$$\theta_1 = \text{logit}\left(\frac{\phi+1}{2}\right), \quad \theta_2 = \log\tau, \quad \text{and} \quad \theta_3 = \log\kappa.$$

We use vague priors for $\boldsymbol{\theta}$, as strong priors make the approximation problem easier. For the same reason, we display the results based on only the first $n = 50$ observations in Figure 5(a). The results for the full dataset are similar, but the posterior marginals for the θ_j's are closer to Gaussians.

Figure 5(b)-(d) displays the approximate posterior marginals for θ_1, θ_2 and θ_3. The histograms are constructed from the output of a MCMC algorithm running for one day. The approximations computed are quite precise and no serious deviance can be detected.

Figure 5(e) displays the approximate posterior marginal for $x_t|\boldsymbol{y}$ based on the simplified Laplace approximation, for the component of $\boldsymbol{x}$ which maximises the KLD between the posterior marginal based on the Gaussian approximation and based on the simplified Laplace approximation. The KLD for all the x_t's are quite small and roughly equal to 3×10^{-4}, so there is no particular gain in using the simplified Laplace approximation compared to the Gaussian one. The fit is quite good, although we slightly underestimate the right hand side tail. The approximation error diminishes as the number of observations increases, but is still visible for the full dataset. A closer inspection reveals that the underestimation is due to the (default) quite rough numerical integration (3). Improving the accuracy of the numerical integration removes the underestimation.

We validated the approximations using all the $n = 945$ observations at the modal value $\boldsymbol{\theta}^\star$. The effective number of parameters (24) was about 53, which is small compared to n. A 95% interval for the remainder $r(\boldsymbol{x}; \boldsymbol{\theta}^\star, \boldsymbol{y})/n$ is $[-0.002, 0.004]$ using $1,000$ independent samples. The computational cost for obtaining all the posterior marginals was about 0.32 seconds for each value of $\boldsymbol{\theta}$, and 32 seconds in total.

We also applied the stochastic volatility model to the full dataset, see Figure 5(a), using a Student-t_ν instead of a Gaussian for the observational model in (26), and a uniform prior for $\log \nu$. The number of hyperparameters is then 4. Figure 5(f) shows predictions for future x_t's using the full dataset.

5.4 Semi-parametric ecological regression

In this example we consider an ecological regression problem and analyse the spatial variation of disease risk in relation to a proxy exposure variable available on the same units. This is taken from Natario and Knorr-Held (2003), which is refereed to for a more throughout background.

The data are male larynx cancer mortality counts in the $n = 544$ districts of Germany from 1986 to 1990:

$$y_i \mid \eta_i \;\sim\; \text{Poisson}\left\{E_i \exp(\eta_i)\right\}, \qquad i = 1, \ldots, n. \tag{27}$$

The (fixed) 'district effect' E_i accounts for the number of people in district i, its age distribution, etc., and $\boldsymbol{\eta}$ is the log-relative risk. The maximum likelihood estimator for η_i using (27) is y_i/E_i and is displayed in Figure 6(a). The model for η_i takes the following form,

$$\eta_i = u_i + v_i + f(c_i) \tag{28}$$

where $\boldsymbol{u}$ is a spatially structured term, $\boldsymbol{v}$ is a unstructured term ('random' effects) and $f(c_i)$ is an unknown effect of the exposure covariate with value c_i at district i. For the exposure covariate we use the lung cancer rate as a proxy for smoking consumption, see Figure 6(b). The spatially structured term is modelled as an intrinsic GMRF (Rue and Held, 2005, Ch. 3)

$$u_i \mid \boldsymbol{u}_{-i}, \kappa_{\boldsymbol{u}} \;\sim\; \mathcal{N}\left(\frac{1}{n_i}\sum_{j\sim i} u_j, \frac{1}{n_i \kappa_{\boldsymbol{u}}}\right)$$

where n_i are the number of neighbour districts of i and $\kappa_{\boldsymbol{u}}$ is the unknown precision. The unstructured term $\boldsymbol{v}$ is taken as a vector of independent $\mathcal{N}(0, \kappa_{\boldsymbol{v}})$. The effect of

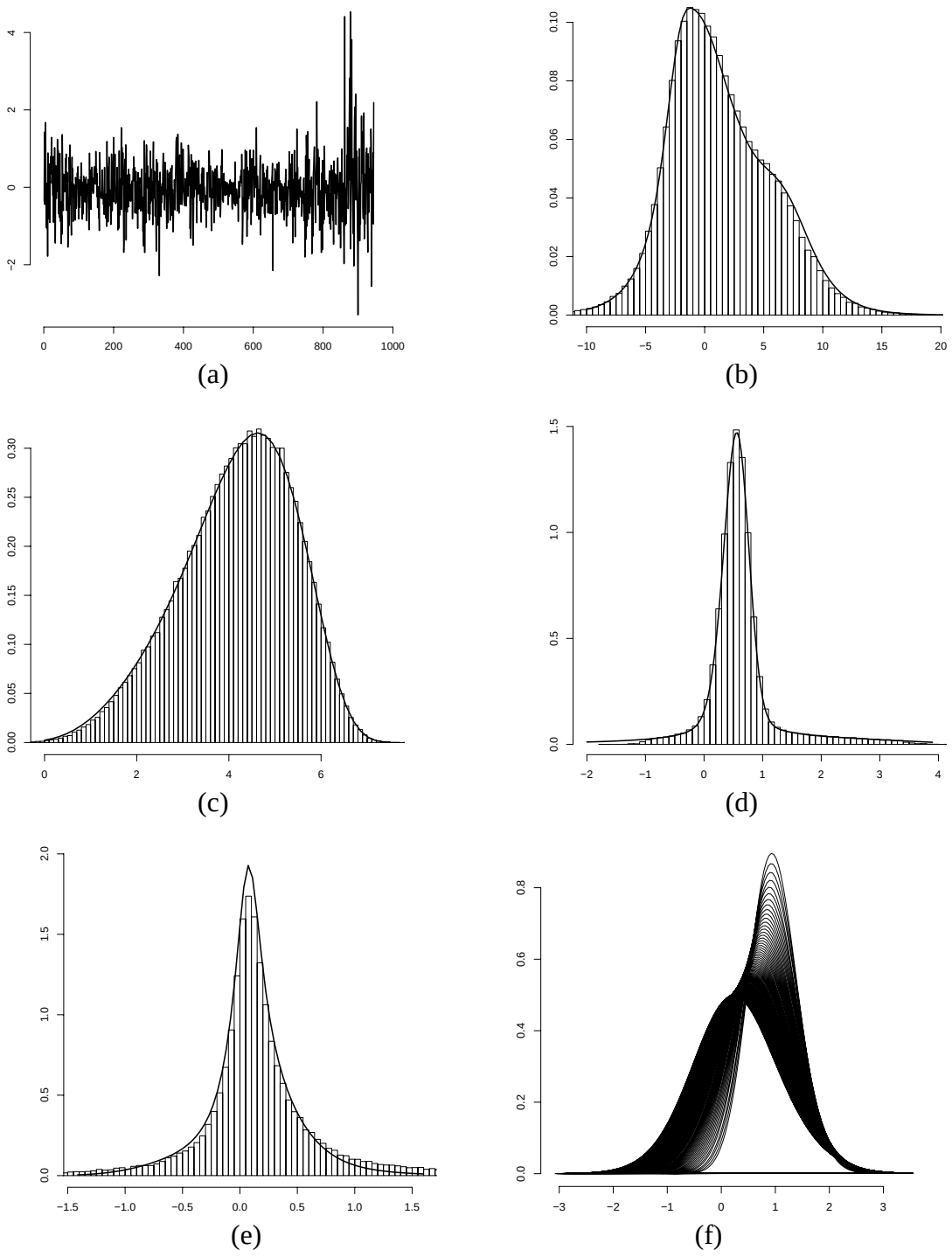


Figure 5: Panel (a) displays the log of the daily difference of the pound-dollar exchange rate from October 1st, 1981, to June 28th, 1985. Panels (b)-(d) display the approximated posterior marginals for θ_1 to θ_3 using only the first $n = 50$ observations in (a). Overlaid are the histograms obtained from a very long MCMC run. The fit is perfect. Panel (e) displays the approximated posterior marginal for the location of the latent field with maximum KLD, compared with the histograms from a very long MCMC run. Our approximation underestimate slightly the behaviour on the right hand side, but this turn out to be an effect of the default (quite rough) integration method. Panel (f) displays the predicted posterior marginals for future x_t's conditioned on the full dataset, assuming in this case that observations are Student-t_ν distributed.

the covariate $\boldsymbol{c}$ is modelled as a smooth function $f(\cdot)$, parametrised as unknown values $\boldsymbol{f} = (f_1, \ldots, f_{100})^T$ for arguments $1, \ldots, n_c = 100$. The values of the covariate are scaled to the interval $[1, n_c]$. The vector $\boldsymbol{f}$ is assumed to follow a second-order random walk,

$$\pi(\boldsymbol{f} \mid \kappa_{\boldsymbol{f}}) \propto \kappa_{\boldsymbol{f}}^{(n_c-1)/2} \exp\left\{-\frac{1}{2}\kappa_{\boldsymbol{f}} \sum_{i=3}^{n_c} (f_i - 2f_{i-1} + f_{i-2})^2\right\} \tag{29}$$

with unknown precision $\kappa_{\boldsymbol{f}}$. To separate the spatial effect and the effect of covariate, we impose $\sum_i u_i = 0$.

Following Natario and Knorr-Held (2003), we assign independent vague Gamma priors to $\boldsymbol{\theta} = (\kappa_{\boldsymbol{u}}, \kappa_{\boldsymbol{v}}, \kappa_{\boldsymbol{f}})$. We use the simplified Laplace approximation for the marginals of $\boldsymbol{x} = (\boldsymbol{\eta}^T, \boldsymbol{u}^T, \boldsymbol{f}^T)^T$ with length $1,188$. The computation took about 52 seconds using 53 evaluation points for the numerical integration. The posterior mean of the spatial term $\boldsymbol{u}$ is displayed in Figure 6(c) whereas the posterior mean of the unstructured effect $\boldsymbol{v}$ is displayed in (d). The posterior mean of the covariate effect $\boldsymbol{f}$ is displayed in (e) with lower and upper 0.025 percentile, computed with the simplified Laplace approximation (solid) and the Gaussian approximation (2) (dotted). The two approximations nearly agree, which is also confirmed by relatively small values of the KLD between the two approximations shown in (f). The maximum KLD for all variables appears for f_{50} and equals 0.032. This indicates that the Gaussian approximation had been sufficient for this example, so the computational cost could had been reduced to 18 seconds (approximating all the marginals). Long MCMC runs conform that the marginals computed using the simplified Laplace approximation are essentially correct.

We validated the approximations by computing $p_{\mathrm{D}}(\boldsymbol{\theta}^\star) \approx 91$ and estimated a 95% interval for the remainder $r(\boldsymbol{x}; \boldsymbol{\theta}^\star, \boldsymbol{y})/n$ as $[-0.001, 0.001]$ using $1,000$ independent samples.

5.5 Log-Gaussian Cox process

Log-Gaussian Cox processes (LGCP) are a flexible class of models that have been successfully used for modelling spatial or spatio-temporal point processes, see for example Møller et al. (1998), Brix and Møller (2001), Brix and Diggle (2001) and Møller and Waagepetersen (2003). In this section we will illustrate how LGCP models can be analysed using our approach for approximate inference.

A LGCP is a hierarchical Poisson process: $\boldsymbol{Y}$ in $W \subset \mathbb{R}^d$ is a Poisson point process with a random intensity function $\lambda(\boldsymbol{\xi}) = \exp(Z(\boldsymbol{\xi}))$, where $Z(\boldsymbol{\xi})$ is a Gaussian field at $\boldsymbol{\xi} \in \mathbb{R}^d$. In this way, the dependency in the point-pattern is modelled through a common latent Gaussian variable $Z(\cdot)$. In the analysis of LGCP, it is common to discretise the observation window W. Divide W into N disjoint cells $\{w_i\}$ located at $\boldsymbol{\xi}_i$ each with area $|w_i|$. Let y_i be the number of occurrences of the realised point pattern within w_i and let $\boldsymbol{y} = (y_1, \ldots, y_N)^T$. Let η_i be the random variable $Z(\boldsymbol{\xi}_i)$. Clearly $\pi(\boldsymbol{y}|\boldsymbol{\eta}) = \prod_i \pi(y_i|\eta_i)$ and $y_i|\eta_i$ is Poisson distributed with mean $|w_i| \exp(\eta_i)$; the same likelihood as for the semi-parametric ecological regression example (27). A straightforward generalisation is to allow for covariates: η_i can be decomposed in the same way as (28), say

$$\eta_i = \beta_0 + \beta_1 c_{1i} + \beta_2 c_{2i} + u_i + v_i, \tag{30}$$

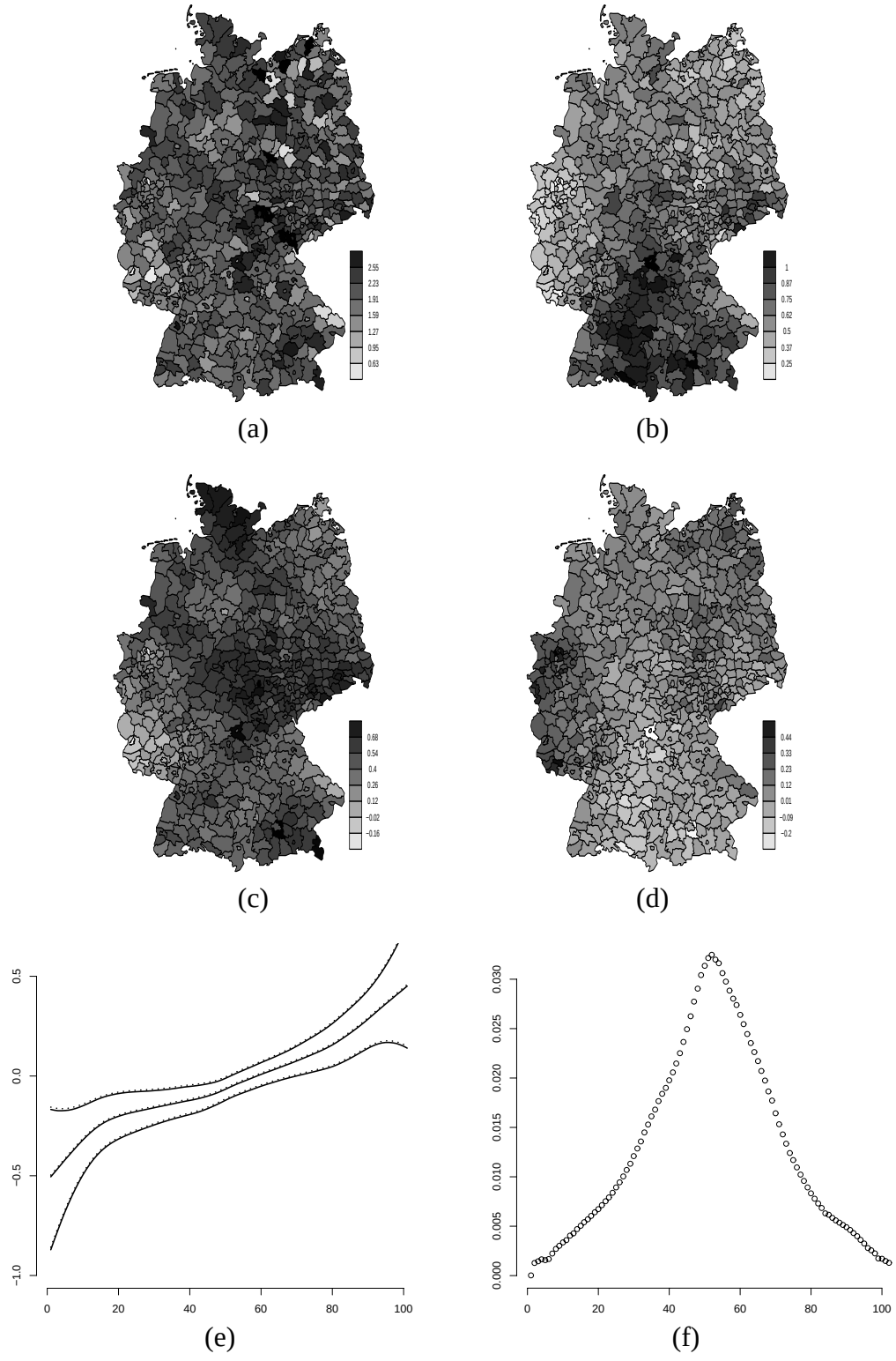


Figure 6: Semi-parametric ecological regression example: panel (a) displays the maximum likelihood estimator for the log relative risk. Panel (b) shows covariate values. Panels (c) and (d) give the posterior mean of the structured ($\boldsymbol{u}$) and unstructured ($\boldsymbol{v}$) effects, respectively. Panel (e) displays the posterior mean of the covariate effect with lower and upper 0.0025 percentiles. The solid line is the simplified Laplace approximation and the dotted line is the Gaussian one. Panel (f) shows the KLD for the covariate effect.

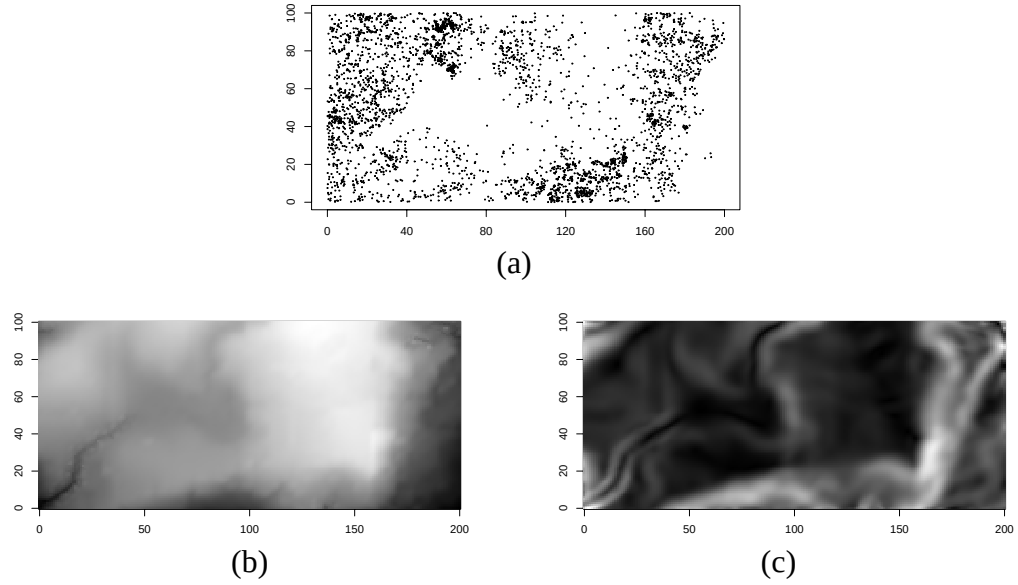


Figure 7: Data and covariates for the log-Gaussian Cox process example: (a) locations of the $3,605$ trees, (b) altitude, and (c) norm of the gradient.

where $\boldsymbol{u}$ represent the spatial component, and $\boldsymbol{v}$ is an unstructured term.An alternative would be to use a semi-parametric model for the effect of the covariates similar to (29).

We apply model (30) to the tropical rain forest data studied by Waagepetersen (2007). These data come from a 50-hectare permanent tree plot which was established in 1980 in the tropical moist forest of Barro Colorado Island in central Panama. Censuses have been carried out every 5th year from 1980 to 2005, where all free-standing woody stems at least 10 mm diameter at breast height were identified, tagged, and mapped. In total, over $350,000$ individual trees species have been censured over 25 years. We will be looking at the tree species *Beilschmiedia pendula Lauraceae* using data collected from the first four census periods. The positions of the 3605 trees are displayed in Figure 7(a). Sources of variation explaining the locations include the elevation and the norm of the gradient. There may be clustering or aggregation due to unobserved covariates or seed dispersal. The unobserved covariates can be either spatially structured or unstructured.

We start by dividing the area of interest into a 200×100 regular lattice, where each square pixel of the lattice represent 25 square metres. Denote elevation and norm of the gradient by $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$, respectively. The scaled versions of these covariates are shown in panel (b) and (c), for $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$, respectively. For the spatial structured term, we use a second order polynomial intrinsic GMRF (see Rue and Held (2005, Sec. 3.4.2)), with following full conditionals in the interior (with obvious notation)

$$\begin{aligned} \mathrm{E}(x_i \mid \boldsymbol{x}_{-i}, \kappa_{\boldsymbol{u}}) &= \tfrac{1}{20}\left(8 \; \begin{array}{ccccc} \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \bullet & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \bullet & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \end{array} \; - 2 \; \begin{array}{ccccc} \circ & \circ & \circ & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \circ & \circ & \circ \end{array} \; - 1 \; \begin{array}{ccccc} \circ & \circ & \bullet & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \bullet & \circ & \circ & \circ & \bullet \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \bullet & \circ & \circ \end{array} \right), \\ \mathrm{Prec}(x_i \mid \boldsymbol{x}_{-i}, \kappa_{\boldsymbol{u}}) &= 20\kappa_{\boldsymbol{u}}. \end{aligned} \tag{31}$$

The precision $\kappa_{\boldsymbol{u}}$ is unknown. The full conditionals are constructed to mimic the thin-plate spline. There are some corrections to (31) near the boundary, which can be found

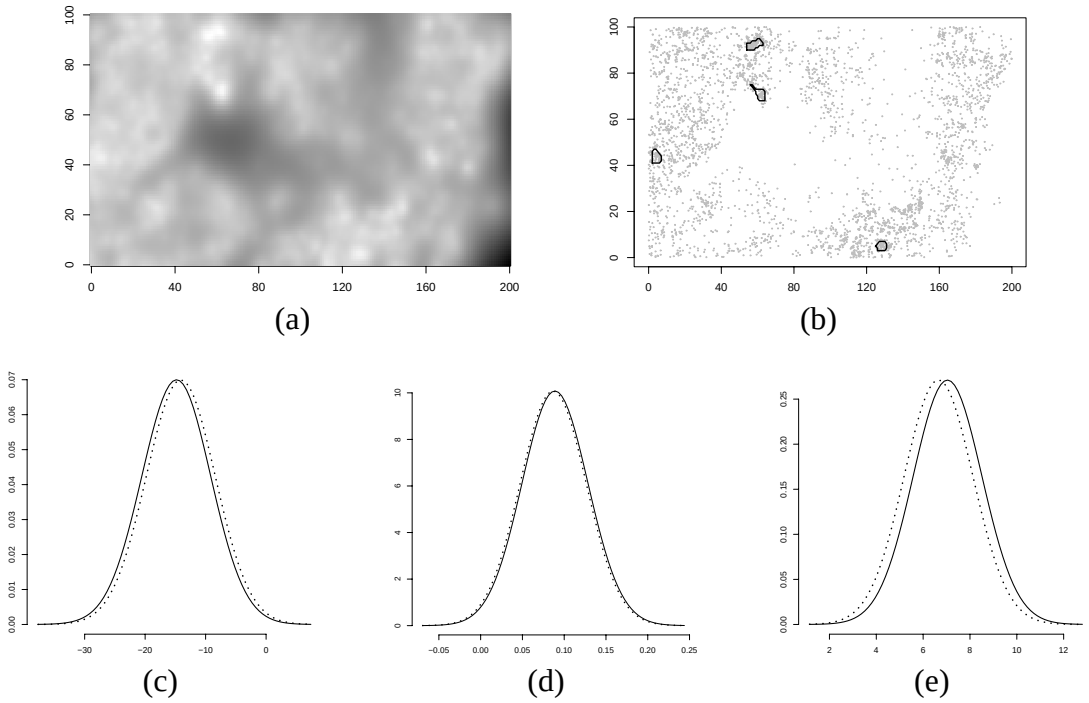


Figure 8: LGCP example: (a) posterior mean of the spatial component $\boldsymbol{u}$, (b) Nodes where the KLD between simplified Laplace and Gaussian approximations exceeds 0.2, (c)-(e) posterior marginals of β_0, β_1 and β_2 using simplified Laplace (solid) and Gaussian approximations (dotted).

using the stencils in Terzopoulos (1988). We impose a sum-to-zero constraint on the spatial term due to β_0. The unstructured terms $\boldsymbol{v}$ are independent $\mathcal{N}(0, \kappa_{\boldsymbol{v}})$, vague Gamma (resp. Gaussian) priors are assigned to $\kappa_{\boldsymbol{u}}$ and $\kappa_{\boldsymbol{v}}$ (resp. to β_0, β_1 and β_2). The GMRF is $\boldsymbol{x} = (\boldsymbol{\eta}^T, \boldsymbol{u}^T, \beta_0, \beta_1, \beta_2)^T$ with dimension $40,003$, and $\boldsymbol{\theta} = (\log \kappa_{\boldsymbol{u}}, \log \kappa_{\boldsymbol{v}})$.

We computed the approximation for $20,003$ posterior marginals using the simplified Laplace approximation, thus ignoring the unstructured components. This task required about 4 hours of computing or about 24 minutes for each value of $\boldsymbol{\theta}$. The high computational cost is due to the large number of computed posterior marginals. The total cost can be reduced to only 10 minutes if using the Gaussian approximation (2). The results are displayed in Figure 8. Panel (a) displays the estimated posterior mean of the spatial component. In (b) we have marked areas where the KLD between the marginal computed with the Gaussian approximation and the one computed with the simplified Laplace approximation exceeds 0.2. These nodes are candidates for further investigation, so we computed their posteriors using also the Laplace approximation; the results agreed well with those obtained from the simplified Laplace approximation. Panel (c) to (e) display the posterior marginals computed with the Gaussian approximation (dotted) and the one computed with the simplified Laplace approximation (solid) for β_0, β_1 and β_2. The difference is mostly due to a horizontal shift, a characteristic valid for all the other nodes as well.

To validate the approximations, we computed $p_{\mathrm{D}}(\boldsymbol{\theta}^*) \approx 1714$ and estimated a 95% interval for the remainder $r(\boldsymbol{x}; \boldsymbol{\theta}^\star, \boldsymbol{y})/n$ as $[0.002, 0.005]$ using $1,000$ independent samples. Varying $\boldsymbol{\theta}$ gave similar results. There are no indications that the approximations does

not works well in this case. Due to the size of the GMRF, the comparison with results from long MCMC runs were performed on a cruder grid, with excellent results. We also compared the conditional marginals in the spatial field for fixed values of $\boldsymbol{\theta}$, and again obtained excellent results.

6 Extensions

6.1 Approximating the marginal likelihood

The marginal likelihood $\pi(\boldsymbol{y})$ is a useful quantity for comparing models, as the Bayes factor is its ratio for two competing models. It is evident from (1) that the natural approximation to the marginal likelihood is the normalising constant for $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$,

$$\widetilde{\pi}(\boldsymbol{y}) = \int \left. \frac{\pi(\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{y})}{\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x}=\boldsymbol{x}^\star(\boldsymbol{\theta})} d\boldsymbol{\theta}. \tag{32}$$

where $\pi(\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{y}) = \pi(\boldsymbol{\theta})\pi(\boldsymbol{x}|\theta)\pi(\boldsymbol{y}|\boldsymbol{x}, \theta)$. An alternative, cruder estimate of the marginal likelihood is obtained by assuming that $\boldsymbol{\theta}|\boldsymbol{y}$ is Gaussian; then (32) turns into some known constant times $|\boldsymbol{H}|^{-1/2}$, where $\boldsymbol{H}$ is the Hessian matrix in Section 3.1, see Kass and Vaidyanathan (1992). Our approximation (32) does not require this assumption, since we treat $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ in a 'nonparametric' way. This allows for taking into account the departure from Gaussianity which, for instance, appears clearly in Figure 5. We have limited experience with using (32) for computing Bayes factors, and for this reason, we have not stressed this issue in the examples. Friel and Rue (2007) use a similar expression as (32) to approximate the marginal likelihood in a different context.

6.2 Moderate number of hyperparameters

Integrating out the hyperparameters as described in Section 3.1 can be quite expensive if the number of hyperparameters, m, is not small but moderate, say, in the range of 6 to 12. Using, for example, $\delta_z = 1$ and $\delta_\pi = 2.5$, the integration scheme proposed in Section 3.1 will require, if $\boldsymbol{\theta}|\boldsymbol{y}$ is Gaussian, $\mathcal{O}(5^m)$ evaluation points. Even if we restrict ourselves to three evaluation points in each dimension, the cost $\mathcal{O}(3^m)$ is still exponential in m. In this section we will discuss an alternative approach which will reduce the computational cost dramatically for high m, but, at the same time, it will also reduce the accuracy of the numerical integration over $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. The aim is to be able to provide useful results even when the number of hyperparameters is so large that the more direct approach in Section 3.1 is unfeasible.

Although many hyperparameters make the integration harder, it is often the case that increasing the number of hyperparameters increases also variability and the regularity, so that the integrand simplifies. Meaningful results can sometimes be obtained even using an extreme choice, that is using only the modal configuration to integrate over $\pi(\boldsymbol{\theta}|\boldsymbol{y})$. This 'plug-in' approach will obviously underestimate variability.

We can consider the integration problem as a design problem where we layout some 'points' in a m-dimensional space. Based on the measured response, we estimate the

response surface at each point. As a first approximation, we can consider only response surfaces of second order, and use a classical quadratic design like the central-composite design (CCD) (Box and Wilson, 1951). A CCD contains an embedded factorial or fractional factorial design with centre points augmented with a group of $2m + 1$ 'star points' which allow for estimating the curvature. For $m = 5$, the design points are chosen (up to an arbitrary scaling) as

$$
\begin{array}{llll}
(1,1,1,1,1,1), & (-1,1,1,1,-1), & (1,-1,1,1,-1), & (-1,-1,1,1,1), \\
(1,1,-1,1,-1), & (-1,1,-1,1,1), & (1,-1,-1,1,1), & (-1,-1,-1,1,-1), \\
(1,1,1,-1,-1), & (-1,1,1,-1,1), & (1,-1,1,-1,1), & (-1,-1,1,-1,-1), \\
(1,1,-1,-1,1), & (-1,1,-1,-1,-1), & (1,-1,-1,-1,-1) & \text{and } (-1,-1,-1,-1,1).
\end{array}
$$

They are all on the surface of the m dimensional sphere with radius $\sqrt{m}$. The star points consist of $2m$ points located along each axis at distance $\pm\sqrt{m}$ and the central point in the origin. For $m = 5$ this makes $n_p = 27$ points in total, which is small compared to $5^5 = 3,125$ or $3^5 = 243$. The number of design-points is 8 for $m = 3$, 16 for $m = 4$ and 5, 32 for $m = 6$, 64 for $m = 7$ and 8, 128 for $m = 9$, 10 and 11, and 256 from $m = 12$ to 17; see Sanchez and Sanchez (2005) for how to compute such designs. For all designs, there are additional $2m + 1$ star-points.

To determine the integration weights Δ_k in (3) and the scaling of the points, assume for simplicity that $\boldsymbol{\theta}|\boldsymbol{y}$ is standard Gaussian. We require that the integral of 1 equals 1, and that the integral of $\boldsymbol{\theta}^T\boldsymbol{\theta}$ equals m. This gives the integration weight for the points on the sphere with radius $f_0\sqrt{m}$

$$\Delta = \left[(n_p - 1)\left(f_0^2 - 1\right)\left\{1.0 + \exp\left(-\frac{mf_0^2}{2}\right)\right\}\right]^{-1}$$

where $f_0 > 1$ is any constant. The integration weight for the central point is $1-(n_p-1)\Delta$.

To validate the CCD integration, we recomputed the posterior marginal for the stochastic volatility model in Section 5.3 using Student-t_ν distributed observations, and for the semi-parametric ecological regression example in Section 5.4 using this integration method instead of the grid search in Section 3.1. The results were indeed positive. The predictions in Figure 5(f) for future x_t's using the full dataset were nearly indistinguishable from those obtained using the CCD integration using only $1/15$ of the computational cost. Same remarks apply for the results shown in Figure 6. The number of hyperparameters in these two cases is 4 and 3 respectively. Although this is not a large number, we should still be able to detect if the CCD integration is too rough, and this does not seem to be the case. Although the CDD integration is not as thoroughly verified as the INLA itself, the results obtained can be viewed as a 'proof of concept' at this stage. We hope to provide further empirical evidence that the CCD integration is adequate when the number of hyperparameters is moderate.

7 Discussion

We have presented a new approach to approximate posterior marginals in latent Gaussian models, based on integrated nested Laplace approximations (INLA). The results obtained

are very encouraging: we obtain practically exact results over a wide range of commonly used latent Gaussian models. We also provide tools for assessing the approximation error, which are able to detect cases where the approximation bias is non-negligible; we note however that this seems to happen only in pathological cases.

We are aware that our work goes against a general trend of favouring 'exact' Monte Carlo methods over non-random approximations, as advocated for instance by Papaspiliopoulos et al. (2006) in the context of diffusions. Our point however is that, in the specific case of latent Gaussian models, the orders of magnitude involved in the computational cost of both approaches are such that this idealistic point of view is simply untenable for these models. As we said already, our approach provides precise estimates in seconds and minutes, even for models involving thousands of variables, in situations where any MCMC computation typically takes hours or even days.

The advantages of our approach are not only computational. It also allows for greater automation and parallel implementation. The core of the computational machinery is based on sparse matrix algorithms, which automatically adapt to any kind of latent field, e.g. 1D, 2D, 3D and so on. All the examples considered in this paper were computed using the same general code, with essentially no tuning. In practice, INLA can be used almost as a black box. The code is now part of `GMRFLib`-library (Rue and Held, 2005, Appendix) and available from the first author's web page. With respect to parallel implementation, the INLA approach computes the approximation of $x_i|\boldsymbol{\theta}, \boldsymbol{y}$ independently for all i for fixed $\boldsymbol{\theta}$. Hence, parallel computing is trivial to implement. This is particularly important for spatial or spatio-temporal latent Gaussian models.

The main disadvantage of the INLA approach is that the computational cost is exponential in the number of hyperparameters m. In most applications m is small, but applications where m goes up to 10 do exist. This problem may be less severe that it appears at first glance: the central composite design approach seems promising, and provides reasonable results when m is not small, but this track needs more research. In fact, we doubt that any MCMC algorithm which would explore the m-dimensional space of $\boldsymbol{\theta}$ in a random fashion would provide more accurate results for the same cost.

It is our view that the prospects of this work are more important than this work itself. Near instant inference will make latent Gaussian models more applicable, useful and appealing for the end user, which has no time or patience to wait for the results of an MCMC algorithm, or has to analyse many different dataset with the same model, or both. Further, near instant inference makes it much easier to challenge the model itself: Bayes factors can be computed through the normalising constant for $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, the model can be assessed through cross-validation, residual analysis, etc., in a reasonable time.

Acknowledgement

The authors acknowledge Jo Eidsvik, Nial Friel, Arnoldo Frigessi, John Hasslet, Leonhard Held, Hanne W. Rognebakke, Judith Rousseau, Håkon Tjelmeland, John Tyssedal and Rasmus Waagepetersen for stimulating discussions related to this work, and the Center for Tropical Forest Science of the Smithsonian Tropical Research Institute for providing the data in Section 5.5.

A Fitting the skew-Normal distribution

We explain here how to fit the skew-Normal distribution (22) to an expansion of the form

$$\log \pi(x) = \text{constant} - \frac{1}{2}x^2 + \gamma^{(1)}x + \frac{1}{6}\gamma^{(3)}x^3 + \cdots . \tag{33}$$

To second order, (33) is Gaussian with mean $\gamma^{(1)}$ and variance 1. The mean and the variance of the skew-Normal distribution are $\xi + \omega\delta\sqrt{2/\pi}$ and $\omega^2(1-2\delta^2/\pi)$, respectively, where $\delta = a/\sqrt{1+a^2}$. We keep these fixed to $\gamma^{(1)}$ and 1, respectively, but adjust a so the third derivative at the mode in (22) equals $\gamma^{(3)}$. This gives three equations to determine (ξ, ω, a). The modal configuration is not available analytically, but a series expansion of the log skew-Normal density around $x = \xi$ gives:

$$x^\star = \left(\frac{a}{\omega}\right) \frac{\sqrt{2\pi} + 2\xi(\frac{a}{\omega})}{\pi + 2(\frac{a}{\omega})^2} + \text{higher order terms}.$$

We now compute the third derivative of the log-density of the skew-Normal at $x^\star$. In order to obtain an analytical (and computationally fast) fit, we expand this third order derivative with respect to a/ω:

$$\frac{\sqrt{2}(4-\pi)}{\pi^{3/2}} \left(\frac{a}{\omega}\right)^3 + \text{higher order terms}. \tag{34}$$

and imposes that (34) equals $\gamma^{(3)}$. This gives explicit formulae for the three parameters of the skewed-normal.

References

Ainsworth, L. M. and Dean, C. B. (2006). Approximate inference for disease mapping. *Computational Statistics & Data Analysis*, 50(10):2552–2570.

Albert, J. H. and Chib, S. (1993). Bayesian analysis of binary and polychotomous responce data. *Journal of the American Statistical Association*, 88(422):669–679.

Allcroft, D. J. and Glasbey, C. A. (2003). A latent Gaussian Markov random field model for spatio-temporal rainfall disaggregation. *Journal of the Royal Statistical Society, Series C*, 52:487–498.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *Journal of the Royal Statistical Society, Series B*, 61(4):579–602.

Banerjee, S., Carlin, B. P., and Gelfand, A. E. (2004). *Hierarchical Modeling and Analysis for Spatial Data*, volume 101 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Bartolucci, F. and Besag, J. (2002). A recursive algorithm for Markov random fields. *Biometrika*, 89:724–730.

Besag, J., York, J., and Mollié, A. (1991). Bayesian image restoration with two applications in spatial statistics (with discussion). *Annals of the Institute of Statistical Mathematics*, 43(1):1–59.

Besbeas, P., Feis, I. D., and Sapatinas, T. (2004). A comparative simulation study of wavelet shrinkage estimators for poisson counts. *International Statistical Review*, 72(2):209–237.

Biller, C. and Fahrmeir, L. (1997). Bayesian spline-type smoothing in generalized regression models. *Computational Statistics*, 12:135–151.

Box, G. E. P. and Wilson, K. B. (1951). On the experimental attainment of optimum conditions (with discussion). *Journal of the Royal Statistical Society, Series B*, 13(1):1–45.

Breslow, N. E. and Clayton, D. G. (1993). Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association*, 88(1):9–25.

Brix, A. and Diggle, P. J. (2001). Spatiotemporal prediction for log-Gaussian Cox processes. *Journal of the Royal Statistical Society, Series B*, 63(4):823–841.

Brix, A. and Møller, J. (2001). Space-time multi type log Gaussian Cox processes with a view to modelling weeds. *Scandinavian Journal of Statistics*, 28:471–488.

Carlin, B. P. and Banerjee, S. (2003). Hierarchical multivariate CAR models for spatio-temporally correlated survival data (with discussion). In *Bayesian Statistics, 7*, pages 45–63. Oxford Univ. Press, New York.

Carter, C. K. and Kohn, R. (1994). On Gibbs sampling for state space models. *Biometrika*, 81(3):541–543.

Chaudhuri, P. and Marron, J. S. (1999). SiZer for exploration of structures in curves. *Journal of the American Statistical Association*, 94:807–823.

Chib, S., Nardari, F., and Shepard, N. (2002). Markov chain Monte Carlo methods for stochastic volatility models. *Journal of Econometrics*, 108:281–316.

Cressie, N. A. C. (1993). *Statistics for spatial data*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. John Wiley & Sons Inc., New York. Revised reprint of the 1991 edition, A Wiley-Interscience Publication.

Diggle, P. J. and Ribeiro, P. J. (2006). *Model-based Geostatistics*. Springer Series in Statistics. Springer.

Diggle, P. J., Tawn, J. A., and Moyeed, R. A. (1998). Model-based geostatistics (with discussion). *Journal of the Royal Statistical Society, Series C*, 47(3):299–350.

Durbin, J. and Koopman, S. J. (2000). Time series analysis of non-Gaussian observations based on state space models from both classical and Bayesian perspectives (with discussion). *Journal of the Royal Statistical Society, Series B*, 62(1):3–56.

Eidsvik, J., Martino, S., and Rue, H. (2006). Approximate bayesian inference in spatial generalized linear mixed models. Technical Report no 2, Department of mathematical sciences, Norwegian University of Science and Technology.

Erästö, P. (2005). *Studies in trend detection of scatter plots with visualization*. PhD thesis, Department of Mathematics and Statistics, University of Helsinki, Finland.

Fahrmeir, L. and Lang, S. (2001). Bayesian inference for generalized additive mixed models based on Markov random field priors. *Journal of the Royal Statistical Society, Series C*, 50(2):201–220.

Fahrmeir, L. and Tutz, G. (2001). *Multivariate Statistical Modelling Based on Generalized Linear Models*. Springer-Verlag, Berlin, 2nd edition.

Friel, N. and Rue, H. (2007). Recursive computing and simulation-free inference for general factorizable models. *Biometrika*, 94(3):661–672.

Frühwirth-Schnatter, S., , and Wagner, H. (2006). Auxiliary mixture sampling for parameter-driven models of time series of small counts with applications to state space modelling. *Biometrika*, 93(4):827–841.

Frühwirth-Schnatter, S. and Frühwirth, R. (2007). Auxiliary mixture sampling with applications to logistic models. *Computational Statistics & Data Analysis*, xx(xx):xx–xx. (to appear).

Gamerman, D. (1997). Sampling from the posterior distribution in generalized linear mixed models. *Statistics and Computing*, 7(1):57–68.

Gamerman, D. (1998). Markov chain Monte Carlo for dynamic generalised liner models. *Biometrika*, 85(1):215–227.

Hannig, J. and Marron, J. S. (2006). Advanced distribution theory for SiZer. *Journal of the American Statistical Association*, 101:254–269.

Holmes, C. C. and Held, L. (2006). Bayesian auxiliary variable models for binary and multinomial regression. *Bayesian Analysis*, 1(1):145–168.

Hsiao, C. K., Huang, S. Y., and Chang, C. W. (2004). Bayesian marginal inference via cadidate's formula. *Statistics and Computing*, 14(1):59–66.

Kass, R. E. and Vaidyanathan, S. K. (1992). Approximate Bayes factors and orthogonal parameters, with application to testing equality of two binomial proportions. *Journal of the Royal Statistical Society, Series B*, 54(1):129–144.

Kitagawa, G. and Gersch, W. (1996). *Smoothness Priors Analysis of Time Series*. Lecture Notes in Statistics no. 116. Springer-Verlag, New York.

Knorr-Held, L. (1999). Conditional prior proposals in dynamic models. *Scandinavian Journal of Statistics*, 26(1):129–144.

Knorr-Held, L. (2000). Bayesian modelling of inseparable space-time variation in disease risk. *Statistics in Medicine*, 19(17-18):2555–2567.

Knorr-Held, L. and Besag, J. (1998). Modelling risk from a disease in time and space. *Statistics in Medicine*, 17(18):2045–2060.

Knorr-Held, L. and Rue, H. (2002). On block updating in Markov random field models for disease mapping. *Scandinavian Journal of Statistics*, 29(4):597–614.

Lang, S. and Brezger, A. (2004). Bayesian P-splines. *Journal of Computational and Graphical Statistics*, 13(1).

Møller, J., Syversveen, A. R., and Waagepetersen, R. P. (1998). Log Gaussian Cox processes. *Scandinavian Journal of Statistics*, 25:451–482.

Møller, J. and Waagepetersen, R. (2003). *Statistical inference and simulation for spatial point processes*, volume 100 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Natario, I. and Knorr-Held, L. (2003). Non-parametric ecological regression and spatial variation. *Biometrical Journal*, 45:670–688.

Øigård, T. A., Rue, H., and Godtliebsen, F. (2006). Bayesian multiscale analysis for time series data. *Computational Statistics & Data Analysis*, 51(3):1719–1730.

Papaspiliopoulos, A. B. O., Roberts, G. O., and Fearnhead, P. (2006). Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes (with discussion). *Journal of the Royal Statistical Society, Series B*, 68(3):333–382.

Papaspiliopoulos, O., Roberts, G. O., and Sköld, M. (2007). A general framework for the parameterisation of hierarchical models. *Statistical Science*, xx(xx):xx–xx. (to appear).

Reeves, R. and Pettitt, A. N. (2004). Efficient recursions for general factorisable models. *Biometrika*, 91(3):751–757.

Rue, H. (2001). Fast sampling of Gaussian Markov random fields. *Journal of the Royal Statistical Society, Series B*, 63(2):325–338.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*, volume 104 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Rue, H. and Martino, S. (2007). Approximate Bayesian inference for hierarchical Gaussian Markov random fields models. *Journal of Statistical Planning and Inference*, 137(10):3177–3192. Special Issue: Bayesian Inference for Stochastic Processes.

Rue, H., Steinsland, I., and Erland, S. (2004). Approximating hidden Gaussian Markov random fields. *Journal of the Royal Statistical Society, Series B*, 66(4):877–892.

Rue, H. and Tjelmeland, H. (2002). Fitting Gaussian Markov random fields to Gaussian fields. *Scandinavian Journal of Statistics*, 29(1):31–50.

Sanchez, S. M. and Sanchez, P. J. (2005). Very large fractional factorials and central composite designs. *ACM Transactions on Modeling and Computer Simulation*, 15(4):362–377.

Schervish, M. J. (1995). *Theory of statistics*. Springer series in statistics. Springer-Verlag, New York, 2nd edition.

Shephard, N. (1994). Partial non-Gaussian state space. *Biometrika*, 81(1):115–131.

Shephard, N. and Pitt, M. K. (1997). Likelihood analysis of non-Gaussian measurement time series. *Biometrika*, 84(3):653–667.

Shun, Z. and McCullagh, P. (1995). Laplace approximation of high dimensional integrals. *Journal of the Royal Statistical Society, Series B*, 57(4):749–760.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and van der Linde, A. (2002). Bayesian measures of model complexity and fit (with discussion). *Journal of the Royal Statistical Society, Series B*, 64(2):583–639.

Terzopoulos, D. (1988). The computation of visible-surface representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):417–438.

Tierney, L. and Kadane, J. B. (1986). Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86.

Waagepetersen, R. P. (2007). An estimating function approach to inference for inhomogeneous Neyman-Scott processes. *Biometrics*, xx(xx):xx–xx. (to appear).

Wecker, W. E. and Ansley, C. F. (1983). The signal extraction approach to nonlinear regression and spline smoothing. *Journal of the American Statistical Association*, 78(381):81–89.

Weir, I. S. and Pettitt, A. N. (2000). Binary probability maps using a hidden conditional autoregressive Gaussian process with an application to Finnish common toad data. *Journal of the Royal Statistical Society, Series C*, 49(4):473–484.

Wikle, C. K., Berliner, L. M., and Cressie, N. A. C. (1998). Hierarchical Bayesian space-time models. *Environmental and Ecological Statistics*, 5(2):117–154.

Paper III

Approximate Bayesian Inference in Spatial Generalized Linear Mixed Models.

Approximate Bayesian Inference in Spatial Generalised Linear Mixed Models

Jo Eidsvik, Sara Martino and Håvard Rue
Department of Mathematical Sciences
NTNU, Norway.

Abstract

In this paper we propose fast approximate methods for computing posterior marginals in spatial generalised linear mixed models. We consider the common geostatistical case with a high dimensional latent spatial variable and observations at known registration sites. The methods of inference are deterministic, using no random sampling.

The first proposed approximation is fast to compute and is 'practically sufficient', meaning that results do not show any bias or dispersion effects that might affect decision making. Our second approximation, an improvement of the first version, is 'practically exact', meaning that one would have to run MCMC simulations for very much longer than is typically done to detect any indication of error in the approximate results. For small count data the approximations are slightly worse, but still very accurate. Our methods are limited to likelihood functions that give unimodal full conditional for the latent variable.

The methods help to expand the future scope of non-Gaussian geostatistical models as illustrated by applications of model choice, outlier detection and sampling design. The approximations take seconds or minutes of CPU time, in sharp contrast to overnight MCMC runs for solving such problems.

KEYWORDS: approximate inference, spatial GLM, spatial design, Bayesian outlier detection, circulant covariance matrix, geostatistics, Newton–Raphson

1 Introduction

Several data are acquired at different geographical locations and require models for the spatial variation. In geostatistics one typically treats these data as indirect measurements of a smooth latent spatial variable. Among the popular applications are geophysics, mining, meteorology and disease mapping, see e.g. Cressie (1993), Diggle et al. (1998) and Banerjee et al. (2004). For analysis of spatial data there are mainly two objectives; i) inference of model parameters, which are regression parameters for explanatory variables, and the standard deviation and range of the latent spatially correlated variable, and ii) prediction of the latent variable at any spatial location.

One topic that has received much attention lately is inference and prediction for the spatial generalised linear mixed model (GLMM), see for instance Diggle et al. (1998) and Christensen et al. (2006) for a Bayesian view, Breslow and Clayton (1993) and Zhang (2002) for a frequentist analogy, and Paciorek and Ryan (2005) and Ainsworth and Dean (2006) who compare penalised likelihood methods with Bayesian solutions using Markov chain Monte Carlo (MCMC) simulations. The common model can briefly be described as follows: Let $\boldsymbol{x}$ represent a latent variable at n spatial sites on a two dimensional domain. Suppose $\boldsymbol{x}$ has a Gaussian prior distribution specified by a mean with regression parameters for explanatory variables, and a covariance matrix. We collectively denote these model parameters by $\boldsymbol{\theta}$. Observations $\boldsymbol{y}$ are made at k of the n sites. These observations are modeled by an exponential family distribution with parameters given by the latent variable $\boldsymbol{x}$ at the sites where the data is acquired. Typical examples of this model include Poisson counts or binomial proportions registrered at some known locations in space, with the objective of predicting the underlying intensity or (log odds) risk surface across the spatial domain of interest, and inferring model parameters. The most common case is probably the situation where one wants to predict across a large spatial domain, but a moderate number of locations register data, i.e. $n \gg k$. For example, this situation occurs in spatial data acquisition for weather forecasting (Gel et al., 2004) and in reserve site selection for predicting the presence of a certain type of species (Polasky and Solow, 2001). One approach for inference and prediciton in such applications is to construct a large regular grid of size n for the latent variable, and then index the grid locations of the k registration sites.

Bayesian methods in spatial GLMMs have been considered difficult because of high dimension and the lack of closed form solutions. The current state of the art is to generate realisations of parameter $\boldsymbol{\theta}$ and latent variable $\boldsymbol{x}$ using Markov chain Monte Carlo (MCMC) algorithms. Since MCMC algorithms have grown mature over the last few decades, see e.g. Robert and Casella (1999), there are a number of fit-for-purpose algorithmic techniques for doing iterative Markov chain updates. Some of these algorithms are more relevant for spatial GLMMs (Diggle and Ribeiro, 2006), but problems remain with convergence and mixing properties of the Markov chain, which in some cases are remarkably slow. Because of these challenges fast inference methods suitable for special cases are needed, possibly avoiding the problems with sampling methods.

The main contribution of this paper is a new method for approximate inference in spa-

tial GLMMs. In particular, this paper provides a recipe for fast approximate Bayesian inference using posterior marginals $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and $\pi(x_j|\boldsymbol{y})$, $j = 1, \ldots, n$. The core of our method is to build a Gaussian approximation for $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$. This is used in i) the Laplace approximation to fit $\pi(\boldsymbol{\theta}|\boldsymbol{y})$, and ii) as an element in a mixture density approximation for $\pi(x_j|\boldsymbol{y})$. Fast approximate inference helps to expand the scope of geostatistical modelling with the possibility of performing spatial design (Diggle and Lophaven, 2006), outlier detection (O'Hagan, 2003), and model choice (Clyde and George, 2004) or model assessment (Johnson, 2004) in a geostatistical setting. In this paper we illustrate how the marginal likelihood $\pi(\boldsymbol{y})$ can be approximated within our framework, provide approximate strategies for Bayesian outlier detection, and perform approximate evaluation of spatial experimental designs.

Another contribution of this paper is an improved approximation for prediction in spatial GLMMs, going beyond our direct solution. While the direct approximation uses the joint full posterior mode of $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ as the core in spatial prediction, the improved approximation uses $\pi(\boldsymbol{x}_{-j}|\boldsymbol{y}, x_j, \boldsymbol{\theta})$, for fixed x_j, when constructing the marginal $\pi(x_j|\boldsymbol{y})$, see Rue et al. (2007). Improvements become important at spatial locations j where the direct approximation is slightly biased, typically near registration sites for the non-Gaussian data. We study differences between direct and improved approximations, and MCMC results in a sensitivity study.

The outline is as follows: In Section 2 we define the special case of spatial GLMMs considered in this paper. The proposed method of approximate inference and prediction is described in Section 3, along with methods for spatial model choice, outlier detection and spatial design. In Section 4 the direct approximation is applied to the Rongelap radionuclide dataset and a rainfall dataset from Norway. We describe the improved approximation for spatial prediction in Section 5, and demonstrate the improved approximations on the Lancashire infection dataset and the rainfall data in Section 6. The computational aspects of our methods are postponed to the Appendix.

2 Spatial GLMM

Let $\boldsymbol{x} = (x_1, \ldots, x_n)'$ represent the latent field at n spatial sites. In an application with binomial proportions data, this spatial variable would denote the latent risk or log odds surface, while it would denote the latent log intensity surface for Poisson count data. We denote the subset of k sites where data is acquired by $\boldsymbol{x}_s = (x_{s_1}, \ldots, x_{s_k})' = \boldsymbol{A}\boldsymbol{x}$ and the $k \times n$ matrix $\boldsymbol{A}$ has entries

$$A_{ij} = I(s_i = j) = \begin{cases} 1 & \text{if} \quad s_i = j \\ 0 & \text{else} \end{cases}, \quad i = 1, \ldots, k, \quad j = 1, \ldots, n, \tag{1}$$

i.e. s_i is the latent field location index of measurement i. With $n \gg k$ the matrix $\boldsymbol{A}$ consists mostly of 0 elements, but it has one 1 value for each row / observation.

The spatial GLMM is specified by the following steps:

1. The latent variable is Gaussian with $\pi(\boldsymbol{x}|\beta_0, \boldsymbol{\beta}, \boldsymbol{\alpha}) = N[\boldsymbol{x}; \mathbf{1}_n\beta_0 + \boldsymbol{H\beta}, \boldsymbol{\Sigma}(\boldsymbol{\alpha})]$, where β_0 is a constant term, $\mathbf{1}_n$ a size $n \times 1$ vector of ones, $\boldsymbol{H}$ an $n \times p$ matrix of covariates, and $\boldsymbol{\beta}$ a $p \times 1$ vector of regression parameters. Further, $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\alpha})$ is a positive definite covariance matrix with $\boldsymbol{\alpha}$ indicating covariance model parameters.

2. The data $\boldsymbol{y} = (y_1, \ldots, y_k)'$ have conditionally independent likelihood $\pi(\boldsymbol{y}|\boldsymbol{x}) = \prod_{i=1}^{k} \pi(y_i|x_{s_i})$. The mean $E(y_i|x_{s_i}) = f^{-1}(x_{s_i})$, where $f(\cdot)$ denotes the chosen link function.

We use a Bayesian model with priors for model parameters $\pi(\boldsymbol{\alpha})\pi(\beta_0)\pi(\boldsymbol{\beta})$.

The regression parameters $\boldsymbol{\beta}$ are important as they capture the variability in data caused by explanatory variables. Many applications of GLMMs focus on estimating $\boldsymbol{\beta}$, also called the fixed effects. In some spatial applications the explanatory variables such as east, west or altitude are not significant, and the trend surface is described using only the constant term β_0. The residual (random effect) with zero mean and covariance $\boldsymbol{\Sigma}$ is then of focus for spatial modelling.

As a simple example of spatial covariance function we give the exponential defined by

$$\Sigma_h(\boldsymbol{\alpha}) = \sigma^2 \exp(-h/\nu), \quad \boldsymbol{\alpha} = (\sigma, \nu), \quad h = \sqrt{h_1^2 + h_2^2}, \tag{2}$$

where (h_1, h_2) are the (North, East) distances between two spatial sites. Many other covariance functions are possible, such as the Matern types which are often recommended, see e.g. Cressie (1993) and Stein (1999). We discuss this more general class of covariance functions in an example in Section 4.1.

If $\pi(\beta_0) = N(\mu, \tau^2)$ the constant term can be integrated out to obtain $\pi(\boldsymbol{x}|\boldsymbol{\theta}) = N(\boldsymbol{x}; \mathbf{1}_n\mu + \boldsymbol{H\beta}, \boldsymbol{C})$, $\boldsymbol{C} = \mathbf{1}_n\tau^2\mathbf{1}_n' + \boldsymbol{\Sigma}$, where $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \boldsymbol{\beta})$. Computational advantages are possible if the n sites are on a regular grid and wrapped on a torus. The covariances $\boldsymbol{\Sigma}$ and $\boldsymbol{C}$ are then block circulant matrices, see Appendix, and the fast discrete Fourier transform (DFT) can be applied as an efficient computational tool. If the prior $\pi(\boldsymbol{\beta})$ is multivariate Gaussian, the regression parameters $\boldsymbol{\beta}$ can similarly be integrated out, but we then lose the computational advantages with DFT since $\boldsymbol{C}$ is no longer block circulant.

The likelihood function can in general be written as

$$\pi(y_i|x_{s_i}) = \exp[g(y_i, x_{s_i})], \quad i = 1, \ldots, k. \tag{3}$$

Certain restrictions apply for maintaining within the exponential family (McCullagh and Nelder, 1989). For an exponential family and a canonical link function $f(\cdot)$ we have

$$\pi(y_i|x_{s_i}) = \exp\{y_i x_{s_i} - b(x_{s_i}) + c(y_i)\}, \quad i = 1, \ldots, k, \tag{4}$$

where $b(x_{s_i}) = \frac{df^{-1}(x_{s_i})}{dx_{s_i}}$ is the cumulant function. For Poisson likelihod $b(x_{s_i}) = m_i \exp(x_{s_i})$, for binomial $b(x_{s_i}) = m_i \log[1+\exp(x_{s_i})]$. Here, m_i is a fixed parameter representing the number of trials in the binomial and the time duration or aggregation interval in the Poisson distribution. This entails log link function for Poisson data, and logit link function for

binomial data. We demonstrate our approximative methods for canonical link functions, but consider a non-canonical link in Section 4.

The likelihood model could be extended to depend on other model parameters, not just the latent variable. For instance, Poisson overdispersion, i.e. $\text{Var}(y_i|x_{s_i}) = \xi \text{E}(y_i|x_{s_i})$, $\xi \geq 1$, can be obtained by a negative binomial likelihood

$$\begin{aligned} \pi(y_i|x_{s_i}, \xi) &\propto \exp(\log\Gamma(y_i + \lambda m_i \exp(x_{s_i})) - \log\Gamma(\lambda m_i \exp(x_{s_i})) \\ &+ \lambda m_i \exp(x_{s_i}) \log\lambda + (y_i + \lambda m_i \exp(x_{s_i}))\log(1+\lambda)), \quad \xi = \frac{1+\lambda}{\lambda} \end{aligned} \tag{5}$$

We then assign a prior to ξ and add this parameter to $\boldsymbol{\theta}$. Other types of overdispersion are possible, for instance the generalised Poisson distribution (Scollnik, 1995).

Note that our model looks at all $\boldsymbol{x}$ jointly and picks out the subset $\boldsymbol{x}_s$ in the likelihood via the $\boldsymbol{A}$ matrix. Another view is to model only $\boldsymbol{x}_s$ and then predict latent variables at other relevant sites in a separate prediction step afterwards, see Zhang (2002). These relevant sites for prediction may for example be on a grid covering the spatial region. Both methods achieve the same goal when the latent field is Gaussian. We prefer to treat them jointly here since this unifies prediction and inference in a one-step procedure, and apply the computational advantages of DFT.

3 Approximate Bayesian inference

The core of our method is to use a Gaussian approximation at the mode of the conditional density $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$. The full conditional density of the latent spatial variable is

$$\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) \propto \exp\{-\frac{1}{2}\boldsymbol{x}'\boldsymbol{C}^{-1}\boldsymbol{x} + \boldsymbol{x}'\boldsymbol{C}^{-1}(\boldsymbol{1}_n\mu + \boldsymbol{H}\boldsymbol{\beta}) + \sum_{i=1}^{k}[y_i x_{s_i} - b(x_{s_i})]\}. \tag{6}$$

The Gaussian approximation $\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ is constructed by locating the mode of equation (6) using Newton–Raphson optimisation, and from fitting the covariance matrix at this mode. At each iteration step we linearise the likelihood part of equation (6) at a fixed value of the latent variable $\boldsymbol{x}_s^0 = \boldsymbol{A}\boldsymbol{x}^0$. This involves linearising the cumulant function $b(x_{s_i}^0)$ for each $i = 1, \ldots, k$. The Gaussian approximation becomes $N[\boldsymbol{x}; \hat{\boldsymbol{\mu}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}(\boldsymbol{y}, \boldsymbol{x}^0), \hat{\boldsymbol{V}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}(\boldsymbol{x}^0)]$. The conditional covariance

$$\hat{\boldsymbol{V}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}(\boldsymbol{x}^0) = \boldsymbol{C} - \boldsymbol{C}\boldsymbol{A}'\boldsymbol{R}^{-1}\boldsymbol{A}\boldsymbol{C}, \quad \boldsymbol{R} = \boldsymbol{A}\boldsymbol{C}\boldsymbol{A}' + \boldsymbol{P}, \tag{7}$$

where $\boldsymbol{P} = \boldsymbol{P}(\boldsymbol{x}_s^0)$ is diagonal with entries $P_{i,i} = 1/b''(x_{s_i}^0)$, $i = 1, \ldots, k$, and $b''(x) > 0$ for all x in our case. The conditional mean

$$\begin{aligned} \hat{\boldsymbol{\mu}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}(\boldsymbol{y}, \boldsymbol{x}^0) &= [\boldsymbol{C}^{-1} + \boldsymbol{A}'\boldsymbol{P}^{-1}\boldsymbol{A}]^{-1}[\boldsymbol{C}^{-1}(\boldsymbol{1}_n\mu + \boldsymbol{H}\boldsymbol{\beta}) + \boldsymbol{A}'\boldsymbol{P}^{-1}\boldsymbol{z}(\boldsymbol{y}, \boldsymbol{x}_s^0)], \\ &= (\boldsymbol{1}_n\mu + \boldsymbol{H}\boldsymbol{\beta}) + \boldsymbol{C}\boldsymbol{A}'\boldsymbol{R}^{-1}[\boldsymbol{z}(\boldsymbol{y}, \boldsymbol{x}_s^0) - \boldsymbol{A}(\boldsymbol{1}_n\mu + \boldsymbol{H}\boldsymbol{\beta})], \end{aligned} \tag{8}$$

$$z_i(y_i, x_{s_i}^0) = [y_i - b'(x_{s_i}^0) + x_{s_i}^0 b''(x_{s_i}^0)]/b''(x_{s_i}^0), \quad i = 1, \ldots, k. \tag{9}$$

The bottomost version in equation (8) is easiest in our case with $n \gg k$. For another link function or the more general likelihood formulation in equation (3) these equations are modified to get $P_{i,i} = -1/g''(y_i, x^0_{s_i})$ and $z_i(y_i, x^0_{s_i}) = -\frac{g'(y_i,x^0_{s_i}) - x^0_{s_i} g''(y_i,x^0_{s_i})}{g''(y_i,x^0_{s_i})}$, $g''(y_i, x^0_{s_i}) < 0$. Differentiation is here with respect to x_{s_i}. After a few iterations we have fitted the Gaussian approximation $\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ at the argument of the posterior mode denoted by $\hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}$, see Appendix. These Newton-Raphson calculations are similar to the traditional ones used for generalised linear models (McCullagh and Nelder, 1989), except the dimension n is usually huge in our case. Nevertheless, effective computation based on DFT applies to our case with block circulant prior covariance matrix, see Appendix. This indicates that large problems of this common type can be handled with modest cost.

The quality of the Gaussian approximation depends on the particular situation. Intuitively one would expect it to be quite good since it is fitted at the posterior mode. Further, we have that $n \gg k$ and the smooth Gaussian prior has much influence. Problems would occur if the likelihood under the general case is such that $g''(y_i, x^0_{s_i}) \geq 0$, which indicates a saddle point or non-positive definite covariance. For instance, a bimodal posterior appears if $g(y_i, x_{s_i}) = -(y_i - x^2_{s_i})^2$, since we cannot detect the sign of x_{s_i} from data. In the class of spatial GLMMs such cases are excluded. If counts are very small, the likelihood is skewed and the Gaussian approximation might have some bias. For large (repeated) counts the central limit theorem applies to the likelihood and the approximation is better. We check this aspect in a sensitivity study in Section 6.

3.1 Parametric inference using $\pi(\boldsymbol{\theta}|\boldsymbol{y})$

The posterior marginal for model parameters $\boldsymbol{\theta}$ is

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{y})\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})} \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})}, \tag{10}$$

which is valid for any value of the spatial variable $\boldsymbol{x}$, for example $\boldsymbol{x} = \hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}$, the argument at the posterior mode for fixed $\boldsymbol{\theta}$. The challenging part of equation (10) is the denominator $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ which we next approximate using the fitted Gaussian density. The approximate density for the model parameters is then

$$\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y}) \propto \left.\frac{\pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\hat{\pi}(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})}\right|_{\boldsymbol{x}=\hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}}. \tag{11}$$

Equation (11) is the Laplace approximation, see e.g. Tierney and Kadane (1986) who show that for fixed dimension n of variables $\boldsymbol{x}$, and dimension k of observations increasing, the relative error of this marginal density approximation is $O(k^{-3/2})$, after renormalisation. Relative error can be advantageous, especially in the tails of the distribution. Monte Carlo error, on the other hand, is additive $O_p(B^{-\frac{1}{2}})$, where B is the number of Monte Carlo samples.

For spatial models of our type asymptotic results of order $O(k^{-3/2})$ are not that easily established since $n \gg k$, and even though we can use only $\boldsymbol{x}_s = \boldsymbol{A}\boldsymbol{x}$ in constructing

$\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, the dimension of the latent variable is of the order of the data. It is unclear which type of asymptotics to use in this case, see Rue et al. (2007). A heuristic argument for the approximation in our case goes as follows: The data are counts at each registration site and can be considered as sums of several Bernoulli trials in the binomial case, or as a sequence of small independent time duration events in the Poisson case. I.e. the count data are used as repeated measurements. Hence, as these intervals or trials increase, the number of data increases while the number of spatial sites remains constant, and the argument for the Laplace approximation in equation (11) holds. Moreover, the non-Gaussian data have influence for all sites in $\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ via the smooth prior. Evidence for good quality of the approximation is further provided in our examples below. See also Rue and Martino (2006) and Rue et al. (2007) who used the Laplace approximation that we use here, but with a Gaussian Markov random field and with $k = n$. Their examples show that the Laplace approximation works well, even for very small counts.

The Laplace approximation has been used extensively in GLMMs, but mostly for approximate integration over the latent variables (random effects) in marginal likelihood estimation, and not so much for approximating posterior marginals, see Tierney and Kadane (1986).

For frequentist models Vidoni (2006) provides an extensive overview with an example of Gaussian random effects and GLMMs, particularly focusing on predicting future outcomes. Ng et al. (2006) present methods for marginal and penalised quasi likelihood inference along with bias correction techniques. They compare results with simulated maximum likelihood where the latent variables are sampled from the density with the current parameter estimate. This is an Expectation-Maximisation algorithm, which is also applied in French and Wand (2004) who use the Laplace approximation in the expectation step of their algorithm, see also Booth and Hobart (1999). French and Wand (2004) use a small spatial example. Skaug and Fournier (2006) use the Laplace approximation with automatic differentiation to find approximate maximum likelihood estimates with a small spatial simulation example.

For Bayesian models the Laplace approximation can be used with Gaussian priors for regression parameters in a GLMM setting, see Raftery (1996). Lewis and Raftery (1997) use the Laplace approximation with a MCMC sampler to compute Bayes factors. In Bayesian model choice the marginal likelihood is sometimes evaluated using an identity similar to equation (11), see e.g. Chib (1995). Since the Gaussian approximation in the denominator can then be quite poor, the common practice is to draw MCMC realisations to approximate the denominator, see e.g. Chib (1995) and Hsiao et al. (2004). In our case we use a Gaussian approximation for the full conditional of $\boldsymbol{x}$ only, to calculate the posterior marginal for $\boldsymbol{\theta}$, which is very different from fitting a Gaussian approximation to $\boldsymbol{x}$ and $\boldsymbol{\theta}$ jointly.

When computing the approximation in equation (11), we first locate the mode of $\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ and fit the dispersion from the Hessian. In the relevant domain of the sample space for $\boldsymbol{\theta}$ we then evaluate equation (11) for a regular set of parameter values $\boldsymbol{\theta}_l$, $l = 1, \ldots, L$,

normalised so that

$$\sum_l \Delta_{\boldsymbol{\theta}} \hat{\pi}(\boldsymbol{\theta}_l|\boldsymbol{y}) = 1. \tag{12}$$

In this equation $\Delta_{\boldsymbol{\theta}}$ is the cell volume in the defined grid for $\boldsymbol{\theta}$ values. The approximate marginal densities for elements of $\boldsymbol{\theta}$ can be obtained by summing out dimensions in the grid of $l = 1, \ldots, L$ parameter values. Note that accurate such numerical approximation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ requires quite small dimension of $\boldsymbol{\theta}$. In our case we always use two parameters of the covariance function, and we add a regression parameter for one explanatory variable or a parameter for overdispersion. At most we get dimension three for the model parameters and this is numerically tractable. The density function $\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ could also be approximated differently, for example by a parametric fit to the density or by numerical quadrature (Press et al., 1996).

3.2 Spatial prediction using $\pi(x_j|\boldsymbol{y})$

For approximate Bayesian spatial prediction we use marginals $\hat{\pi}(x_j|\boldsymbol{y}) = \sum_l \hat{\pi}(x_j|\boldsymbol{y}, \boldsymbol{\theta}_l)\hat{\pi}(\boldsymbol{\theta}_l|\boldsymbol{y})$, $j = 1, \ldots, n$. This is a mixture of Gaussian distributions where weights denote the approximate posterior for model parameters as presented in Section 3.1. The approximate marginal means become

$$\hat{\mu}_{x_j|\boldsymbol{y}} \approx \sum_l \hat{m}_{x_j|\boldsymbol{y},\boldsymbol{\theta}_l} \hat{\pi}(\boldsymbol{\theta}_l|\boldsymbol{y}), \quad j = 1, \ldots, n, \tag{13}$$

where we pick element j of the joint conditional mode. The approximate marginal variances are

$$\hat{V}_{x_j|\boldsymbol{y}} \approx \sum_l \hat{V}_{x_j|\boldsymbol{y},\boldsymbol{\theta}_l} \hat{\pi}(\boldsymbol{\theta}_l|\boldsymbol{y}) + \sum_l (\hat{m}_{x_j|\boldsymbol{y},\boldsymbol{\theta}_l} - \hat{\mu}_{x_j|\boldsymbol{y}})^2 \hat{\pi}(\boldsymbol{\theta}_l|\boldsymbol{y}), \quad j = 1, \ldots, n, \tag{14}$$

where $\hat{V}_{x_j|\boldsymbol{y},\boldsymbol{\theta}} = \hat{V}_{x_j|\boldsymbol{y},\boldsymbol{\theta}}(\hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}})$ denotes diagonal element j of the conditional covariance in equation (7), evaluated at the argument of the posterior mode. For these variance terms we need to calculate the diagonal entries of $\boldsymbol{CA}'\boldsymbol{R}^{-1}\boldsymbol{AC}$ given by

$$(\boldsymbol{CA}'\boldsymbol{R}^{-1}\boldsymbol{AC})_{jj} = \sum_{i=1}^{k}\sum_{i'=1}^{k} C_{j,s_i} R^{-1}_{ii'} C_{s_{i'},j}, \quad j = 1, \ldots, n. \tag{15}$$

Block prediction (Cressie, 1993) is also possible using that any linear combination of elements in $\boldsymbol{x}$ is Gaussian when the joint $\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ is Gaussian. For a linear combination $b = \sum_{i=1}^{B} d_i x_{b,i} = \boldsymbol{Dx}$ for size $1 \times n$ vector $\boldsymbol{D}$ containing mostly zeros for small or moderate blocks, we have that $\hat{\pi}(b|\boldsymbol{y}, \boldsymbol{\theta}) = N(\boldsymbol{D}\hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}, \boldsymbol{D}\hat{\boldsymbol{V}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}(\hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}})\boldsymbol{D}')$. Instead of computing just diagonal elements of the posterior covariance $\hat{\boldsymbol{V}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}(\hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}})$, we must compute all the $(B+1)B/2$ elements.

3.3 Estimation of marginal likelihood $\pi(\boldsymbol{y})$

For assessing the model one often uses the marginal likelihood $\pi(\boldsymbol{y})$, see e.g. Clyde and George (2004). The marginal likelihood is given by

$$\pi(\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{\theta}|\boldsymbol{y})\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})}. \tag{16}$$

This approximate marginal likelihood can be evaluated as the normalising constant required for computing the approximate density $\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ in equation (11). We denote this estimate of the marginal likelihood by $\hat{\pi}(\boldsymbol{y})$. Recall that no Gaussian assumptions are made for $\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, and hence equation (16) is not the Laplace approximation.

For spatial models one natural model assessment is related to the choice of covariance function, another to the significance of explanatory variables.

3.4 Cross-validation for outlier detection

As a quality check of the data or the model one can use cross-validation. Zhang (2003) points out that cross-validation is not useful for checking the covariance function. In this Section we instead consider the prediction of data at one registration site conditional on all other data, and use this for detecting outliers, see e.g. O'Hagan (2003).

Let $\boldsymbol{y}_{-i}$ be the vector of all observations, but y_i excluded. The approximate predictive probability function or density

$$\hat{\pi}(y_i|\boldsymbol{y}_{-i}) = \sum_l \int_{x_{s_i}} \pi(y_i|x_{s_i})\hat{\pi}(x_{s_i}|\boldsymbol{y}_{-i},\boldsymbol{\theta}_l)\hat{\pi}(\boldsymbol{\theta}_l|\boldsymbol{y}_{-i})dx_{s_i}, \quad i = 1,\ldots,k, \tag{17}$$

is relevant for crossvalidation in this case. Note that the rightmost densities in equation (17) are the Gaussian approximation and the posterior marginal for model parameters in Section 3.1, but now calculated using the reduced dataset $\boldsymbol{y}_{-i}$. When quality checking all measurements, we recalculate these different approximations k times in total.

Based on equation (17) we first compute the ϕ and $1-\phi$ percentiles of the predictive distribution, denoted c_ϕ and $c_{1-\phi}$, respectively. We then tag y_i as an outlier or gross error if $y_i < c_\phi$ or $y_i > c_{1-\phi}$. Typically $\phi = 0.01$ or another small number, depending on the application.

We solve equation (17) for any y_i by direct Monte Carlo integration with sampling from $\hat{\pi}(x_{s_i}|\boldsymbol{y}_{-i},\boldsymbol{\theta}_l)$ for every $\boldsymbol{\theta}_l$ parameter.

3.5 Spatial design

The spatial design of registration sites is important for reliable spatial prediction and for parameter estimation. For prediction spatial regularity is wanted, while some non-regularity is useful for estimation of model parameters, see Diggle and Lophaven (2006).

The objective in planning the spatial design is typically to minimise some criterion, most notably the spatially integrated predictive standard deviation, $\sum_{j=1}^{n} \sqrt{\mathrm{V}(x_j|\boldsymbol{y})}$. Quite often there is some data $\boldsymbol{y}$ available, but it is debated whether one should purchase more data $\boldsymbol{y}_a$, i.e. augment the dataset to improve prediction and make better decisions. This entails selecting k_a extra sites in addition to the k registration sites that we already have. Before this augmented dataset is acquired one can estimate, in a prospective setting, the integrated predictive standard deviation as

$$I = \sum_{\boldsymbol{y}_a} \sum_{j=1}^{n} \sqrt{\mathrm{V}(x_j|\boldsymbol{y}_a, \boldsymbol{y})} \pi(\boldsymbol{y}_a|\boldsymbol{y}). \tag{18}$$

Of course, for continuous data $\boldsymbol{y}_a$ the outmost sum in equation (18) is replaced by an integral. This prospective design will always decrease the integrated predictive standard deviation, but two different designs, both of size k_a, will decrease it with different amounts. The reduction could provide valuable insight when planning experiments.

We solve for the integrated predictive standard deviation in equation (18) by direct Monte Carlo integration. Evaluation of design criterions is very hard to do with the standard tool of MCMC since one would have to rerun the Markov chain for each new dataset. Clearly, this shows an area of application for approximate Bayesian inference.

4 Examples of approximate inference

In the first example we demonstrate the accuracy of our fast approximate method by comparing it with time consuming MCMC sampling. We also discuss model choice for this example. The second example illustrates approximate Bayesian inference applied to outlier detection and spatial design.

4.1 Rongelap dataset of radionuclide counts

We redo one of the examples used in Diggle et al. (1998). The data are made at a moderate number of registration sites and the latent spatial variable is modeled by a stationary prior distribution. The dataset consists of $k = 157$ measurements of $y_i =$ radionuclide counts for various time durations m_i, $i = 1, \ldots, k$. All 157 registration sites are displayed in Figure 1. The data are modeled by a spatial GLMM with a Poisson distribution in equation (4). No explanatory variables have been used previously for this dataset and we simply assign hyperparameters for the constant term β_0. These are $\mu = 1.5$, directly calculated as the logarithm of all data scaled with the individual time intervals, and $\tau = 1$. For the latent spatial variable we construct a regular grid with interval spacing 40m covering the island. The gridsize is then $n_1 = 103$ (North) and $n_2 = 187$ (East). Following Christensen et al. (2006) we use an exponential covariance function, see equation (2). A $\log$ link function is used. We use a flat prior for $\boldsymbol{\theta} = (\sigma, \nu)$. We tested with other priors, but it did not have much effect. We used about five-ten Newton-Raphson iterations to locate the mode of the

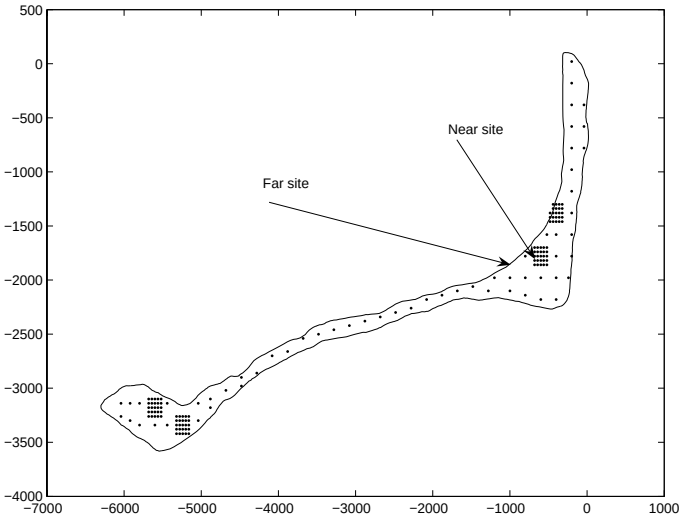


Figure 1: Rongelap island with the 157 registration sites for observations of radionuclide concentrations. The selected prediction sites near registrered data and far away are indicated by arrows.

Gaussian approximation $\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$, after which the machine precision is reached. This optimisation and all evaluations associated with it takes about 0.5 seconds of CPU time using MATLAB code.

The set of parameter values defined in equation (12) covers $\sigma \in \{0.3, 1\}$ and $\nu \in \{50, 300\}$ with $L = 2500$. The approximate density $\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is shown in Figure 2 (upper) along with marginals $\hat{\pi}(\sigma|\boldsymbol{y})$ and $\hat{\pi}(\nu|\boldsymbol{y})$ in Figure 2 (lower, solid curve). Figure 2 (lower, dashed curves) displays estimates of the marginals using MCMC sampling. Since the solid and dashed curves in Figure 2 (right) are hard to distinguish, the Laplace approximation appears to be very good. The MCMC sampler was an independent proposal Metropolis–Hastings (MH) scheme defined by i) Proposing $\boldsymbol{\theta}^\star$ from $\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, ii) Proposing $\boldsymbol{x}^\star$ from $\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}^\star)$, and iii) Accepting or rejecting them jointly.

In Figure 3 we show the marginal predictions in equation (13) and standard deviations given by equation (14). We recognise the registration sites in the standard deviation map and see that climb to about 0.6 as one goes about 500m away from these. Similarly, the spatial predictions in Figure 3 (upper) are near the prior mean as we go away from the island. The main trends are similar to the ones obtained by MCMC sampling in Diggle et al. (1998). The direct approximation takes a few minutes to compute using MATLAB code.

We go on to check the Gaussian approximation that we use for the latent variable via $\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$. For this purpose we use importance sampling and MH methods. In both these Monte Carlo methods we draw independent proposals from the Gaussian approximation $\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$, keeping the parameter fixed at $\boldsymbol{\theta} = (0.6, 152)$, regarded to be a likely parameter value (Figure 2). We choose to evaluate the three approximations (direct Gaussian approximation, MH and importance sampling) at (North,East) coordinates $(-1800, -600)$

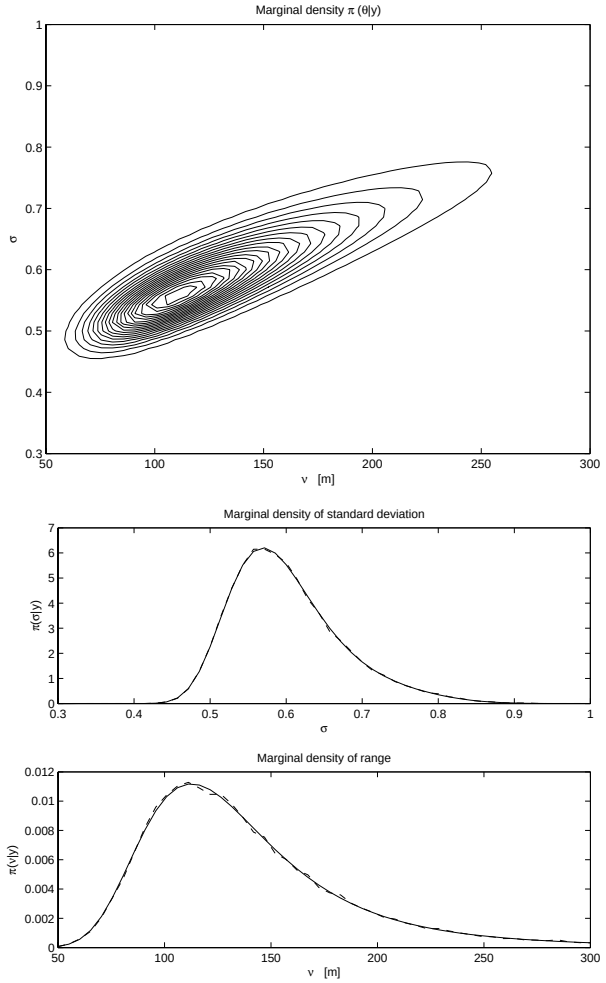


Figure 2: Rongelap dataset. Upper) Direct approximation of posterior for parameters $\boldsymbol{\theta} = (\sigma, \nu)$. Lower) Direct approximation of posterior marginals for standard deviation σ and spatial correlation range ν (solid). MCMC approximation of posterior marginals (dashed).

and $(-1850, -1000)$, see Figure 1. These two are chosen since they represent locations near and far from registration sites. In Figure 4 (solid) we show the approximate densities $\hat{\pi}(x_j|\boldsymbol{y}, \boldsymbol{\theta})$, where j correponds to these two spatial grid locations. Also displayed in Figure 4 are the approximations based on MH (dashed) and importance sampling (dotted). We do not see clear discrepancies between direct approximation and Monte Carlo results, for neither near nor far prediction site. Specifically we note that the results of direct Gaussian approximate inference are hardly distinguishable from the Monte Carlo approximations. The small fluctuations in Figure 4 (solid and dashed) are caused by Monte Carlo error. This error itself is larger than the differences between the direct approximation and the Monte Carlo estimates. Approximate inference is sufficiently accurate for all practical purposes in this example. Possible bias effects are so small that it would have no impact on the decisions made concerning this application. We hence use the

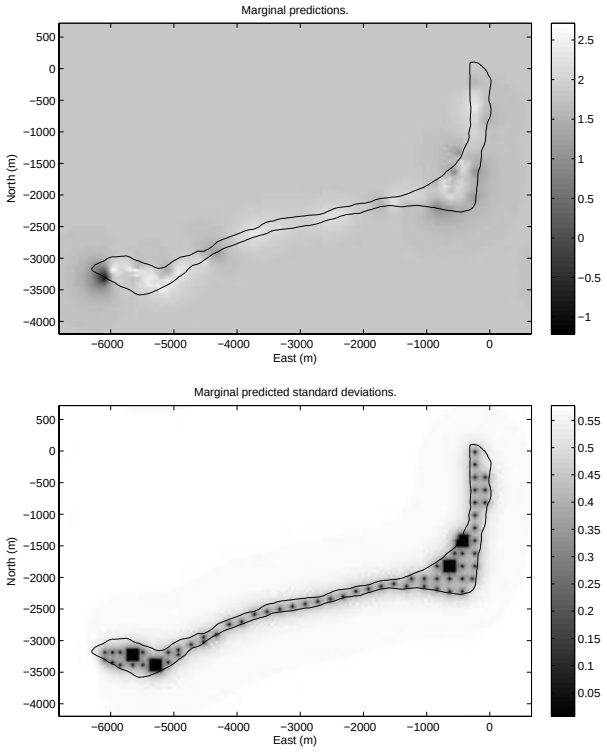


Figure 3: Rongelap dataset. Upper) Predicted spatial variable. Lower) Marginal standard deviation of spatial variable.

term 'practically sufficient' for our approximation. The counts are large in this example, ranging from 75 to 21.000 with varying m_i, and this might help us in constructing the approximations. Nevertheless, a histogram of the data is very skewed and this example has been used before to demonstrate non-Gaussian behaviour in spatial GLMMs (Diggle et al., 1998).

The Monte Carlo algorithms used 100.000 proposals. The acceptance probability of the MH algorithm with joint updating of $\boldsymbol{\theta}$ and $\boldsymbol{x}$ was about 0.8, indicating that the approximation is very good. In fact, this shows that the approximate posteriors have potential as good proposal distributions in MH algorithms. But, then again, this is not needed, since the direct approximation is very accurate. The Monte Carlo run requires several hours. Further, this takes place after the good independent proposal distribution $\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\hat{\pi}(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})$ has been constructed with fast approximate methods. Importance sampling resulted in an effective sample size of 90.000, indicative of small variability in the importance weights for different proposals. For the plotting of Monte Carlo approximations in Figure 4 we split the sample space of x_j into 100 disjoint regions in the interval $\hat{\mu}_{x_j|\boldsymbol{y},\boldsymbol{\theta}} \pm 4\sqrt{\hat{V}_{x_j|\boldsymbol{y},\boldsymbol{\theta}}}$, and thus created the estimated density curves (dashed and dotted).

We next study marginal likelihood values $\pi(\boldsymbol{y})$ for various spatial covariance functions. For this purpose we implement the more general Matern class of covariance functions

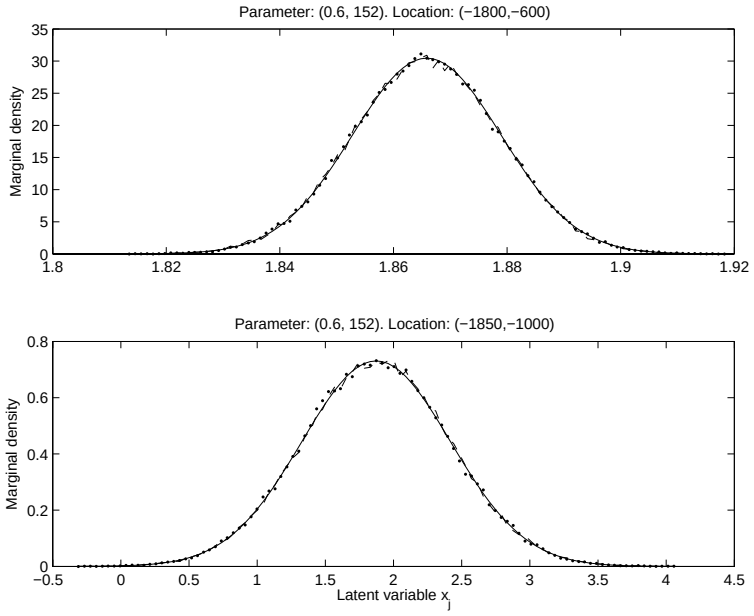


Figure 4: Rongelap dataset. Conditional density $\hat{\pi}(x_j|\boldsymbol{y}, \boldsymbol{\theta})$ obtained by approximate inference at two spatial locations and for parameter values fixed at ($\sigma = 0.6$, $\nu = 152$m). Solid is Gaussian approximation, dashed is approximation obtained by MCMC sampling, and dotted is approximation from importance sampling.

which is defined by

$$\Sigma_h(\sigma, \nu, \kappa) = \sigma^2 \frac{\tau^\kappa K(\tau, \kappa)}{2^{\kappa-1}\Gamma(\kappa)}, \quad \tau = \alpha_\kappa h/\nu, \quad h = \sqrt{h_1^2 + h_2^2}, \tag{19}$$

where $K(\cdot, \kappa)$ denotes a modified Bessel function of order κ and $\Gamma(\cdot)$ is the Gamma function. In equation (19) the α_κ parameter is set so that the correlation is approximately 0.05 at spatial distance $h = \nu$. The Matern family contains the exponential covariance in equation (2) as a special case when $\kappa = 0.5$, while it reduces to the Gaussian (squared exponential) covariance function when $\kappa = \infty$. We calculate the marginal likelihood estimate $\hat{\pi}(\boldsymbol{y})$ in Section 3.3 for four different Matern covariance models. These models are i) exponential covariance ($\kappa = 0.5$), ii) $\kappa = 1$, iii) $\kappa = 2$, and iv) Gaussian covariance ($\kappa = \infty$). The difference in log marginal likelihood is 5.6 when comparing the exponential with case ii), 9.9 when comparing the exponential with case iii), while it is 19.6 in favour of the exponential over the Gaussian covariance. Hence, there is evidence of a steep exponential decline in covariance at zero distance for this dataset.

We finally consider a different likelihood model of a negative binomial type, see equation (5). If the additional overdisperion $\xi = 1$, the variance equals the mean. We use an exponential prior for ξ with mean 2, truncated at 1. The maximum a posteriori estimate is $\xi = 3$, i.e. $\text{Var}(y_i|x_{s_i}) = 3\text{E}(y_i|x_{s_i})$. Posterior 5 percentile is 1.5 and 95 percentile is 7.5. The difference in log marginal likelihood is 4 in favour of this model with overdispersion. In this situation the results seem somewhat sensitive to the exponential prior for ξ.

4.2 Precipitation data in middle Norway

We study data of rainfall in September-October 2006 in the middle part of Norway (Møre and Romsdal, North Trøndelag and South Trøndelag). The data are number of rainy days and the number of days in operation ($m_i = 61$) for $i = 1, \ldots, 92$ registration sites, see Figure 5. The filled and open circles in this Figure indicate the proportion of rainy days at

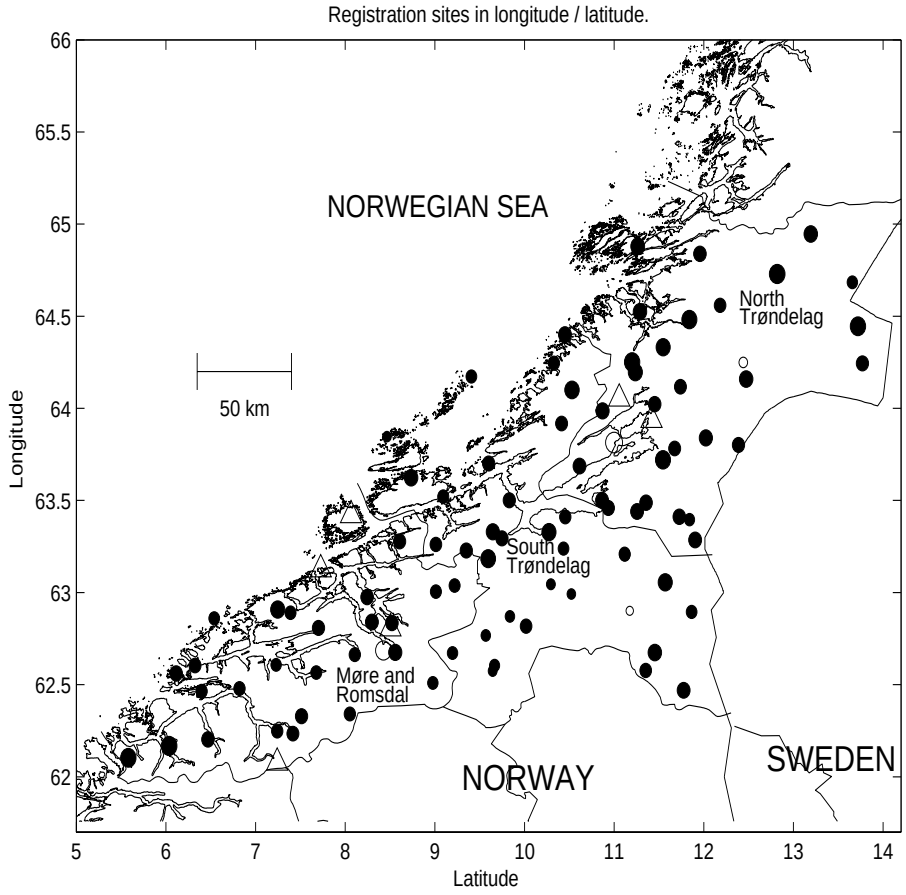


Figure 5: Precipitation data shown in a map of middle Norway. Filled circles are 88 registration sites of rainfall, open circles are 4 sites tagged as outliers. Sizes of circles indicate the proportion of days with rain. Triangles are other registration sites used to evaluate a spatial design of experiment.

each site. Each longitude is about 47.5km and each latitude about 114km in this area. Data are available from the Norwegian Meteorological Insitute (http://www.met.no). Analysis of such precipitation data are important for local weather forecasting and for operating hydroenergy plants. Here we use these data as binomial counts.

We first try to explain the variability in the data using only a constant term and spatial correlation in the latent variable. We use an exponential covariance function in this example, with flat priors for range ν and standard deviation σ. We tested with other priors without much effect. The posterior mode of $\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is at $\sigma = 0.3$ and $\nu = 35$km. We define a

grid of $L = 2500$ parameter values around this mode. The hyperparameters for $\pi(\beta_0)$ are $\mu = 0.8$ and $\tau = 1$. Results are obtained with logit link. The difference in log marginal likelihood is 0.4 in favour of the logit link compared with the probit link.

We next consider adding the distance to the ocean as an explanatory variable. The 95 credibility interval for the corresponding regression parameter β_1 is $(-0.005, 0.004)$, which means that the covariate 'distance to ocean' is not significant. The log marginal likelihood of 0.3 in favour of model with trend is not significant. Of course, other covariates could provide better interpretation, especially for short forecasts, but these are not available to us.

In the following we consider the model with logit link and no regression parameters. Four of the sites in Figure 5 are displayed in open circles because they were tagged as outliers using the method described in Section 3.4. For the south-easternmost of these sites (N$=$ 62.9, E$=$ 11.2) the registered number of rainy days is 25 out of 61 days of operation. Using Bayesian cross-validation based on all other data as in equation (17) we estimate the lower and upper 1 percentiles of $\pi(y_i|\boldsymbol{y}_{-i})$ to be 30 and 50, respectively. Hence, it rains too few days at this site and the observation is tagged as an outlier. Based on circle sizes in Figure 5 this is reasonable as the surrounding sites have higher proportions of rainy days. This result might indicate that the measurement at site (N$=$ 62.9, E$=$ 11.2) was a gross error or that there exists local topology that is hard to capture by our model. Moreover, the registration sites do not all use the same equipment for measuring precipitation, and according to meteorologists there is a lower threshold on the amount some types can measure on days with very little rain. For the other outliers: the site at (N$=$ 64.2, E$=$ 12.4) also registered too few rainy days, while the sites at (N$=$ 62.7, E$=$ 8.4) and (N$=$ 63.8, E$=$ 11.0) registered too many days with rain.

In Figure 5 six sites are displayed as triangles. These are sites that had much downtime or did not registrer data in September-October 2006. To illustrate spatial design for this dataset we imagine what would have happened if these sites had been registrering data. We then integrate out the data at these sites using equation (18). We also include the outlier locations in this calculation, making the total of extra design points $k_a = 10$. Using only the original design points without outliers ($k = 88$ sites), we obtained a spatially integrated standard deviation of 18.68. When we include the outlier sites and sites marked with triangles, the prospective integrated standard deviation decreased to 17.94. The integrated standard deviation is calculated based on a large regular grid of prediction sites covering the region.

As an alternative we test another design also based on $k_a = 10$ extra registration sites. For this other design we place five registration sites randomly within 50km radius of both site (N$=$ 63.0, E$=$ 10.3) and site (N$=$ 63.8, E$=$ 11.7). This type of infill design seems advantageous in Diggle and Lophaven (2006). The prospective integrated predictive standard deviation with this design is 17.85, which is slightly smaller than the one based on the currently installed registration sites. Of course, this seemingly better spatial design is perhaps not possible in practice for economical or political reasons. For decision-making of this kind it might be more reasonable to study other criteria than the integrated predictive standard deviation. For instance, the value of information (Polasky and Solow, 2001),

referring to the cost of experiments and the revenue one could expect.

5 Improved approximation for $\pi(x_j|\boldsymbol{y},\boldsymbol{\theta})$

We illustrate a method for constructing a more accurate approximation to $\pi(x_j|\boldsymbol{y},\boldsymbol{\theta})$. The improved version, denoted $\tilde{\pi}(x_j|\boldsymbol{y},\boldsymbol{\theta})$, is valuable for two reasons: i) It is more accurate than the direct approach $\hat{\pi}(x_j|\boldsymbol{y},\boldsymbol{\theta})$, and ii) If it is indistinguishable from the direct approximation, this verifies the direct one without Monte Carlo sampling, see Rue et al. (2007)

The improved version is based on

$$\pi(x_j|\boldsymbol{y},\boldsymbol{\theta}) \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta})}{\pi(\boldsymbol{x}_{-j}|x_j,\boldsymbol{y},\boldsymbol{\theta})}, \quad j = 1,\ldots,n. \tag{20}$$

For the improved approximation we use a Gaussian approximation in the denominator of equation (20). The improved approximate marginal, denoted $\tilde{\pi}(x_j|\boldsymbol{y},\boldsymbol{\theta})$, can be evaluated at a set of x_j values and normalised. Note that this marginal is based on conditioning on x_j in equation (20), and using a Laplace approximation to cancel out the remaining variables $\boldsymbol{x}_{-j}$. It is hence more accurate than the direct approach which fits a joint Gaussian for all variables $\boldsymbol{x}$. In the choice of evaluation points for x_j in equation (20) we are guided by $\hat{m}_{x_j|\boldsymbol{y},\boldsymbol{\theta}}$ and $\sqrt{\hat{V}_{x_j|\boldsymbol{y},\boldsymbol{\theta}}}$ available from the direct Gaussian approximation.

We fit $\tilde{\pi}(\boldsymbol{x}_{-j}|x_j,\boldsymbol{y},\boldsymbol{\theta})$ by introducing a fictitious measurement $\tilde{x}_j$ defined by $\pi(\tilde{x}_j|x_j) = N(\tilde{x}_j; x_j, \zeta^2)$, where $\zeta = 10^{-6}$, a very small number. In practice this means that x_j is fixed at the value of the fictitious measurement $\tilde{x}_j$. The approximation becomes

$$\tilde{\pi}(\boldsymbol{x}_{-j}|x_j,\boldsymbol{y},\boldsymbol{\theta}) = \tilde{\pi}(\boldsymbol{x}|\tilde{\boldsymbol{z}},\boldsymbol{\theta}), \quad \tilde{\boldsymbol{z}} = [\boldsymbol{z}(\boldsymbol{y},\hat{\boldsymbol{m}}_{\boldsymbol{x}_s|\boldsymbol{y},\tilde{x}_j,\boldsymbol{\theta}}),\tilde{x}_j], \tag{21}$$

where the first part of $\tilde{\boldsymbol{z}}$ consists of $z_i(\boldsymbol{y},\hat{m}_{x_{s_i}|y_i,\tilde{x}_j,\boldsymbol{\theta}})$, $i = 1,\ldots,k$, defined in equation (9), but now with the latent variable at site j fixed when setting the linearisation point $\hat{\boldsymbol{m}}_{\boldsymbol{x}_s|\boldsymbol{y},\tilde{x}_j,\boldsymbol{\theta}}$. We choose to evaluate equation (20) at the mean of the density in equation (21) given by

$$\tilde{\boldsymbol{\mu}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}} = (\mathbf{1}_n\mu + \boldsymbol{H}\boldsymbol{\beta}) + \boldsymbol{C}\tilde{\boldsymbol{A}}'\tilde{\boldsymbol{R}}^{-1}(\tilde{\boldsymbol{z}} - \tilde{\boldsymbol{A}}(\mathbf{1}_n\mu + \boldsymbol{H}\boldsymbol{\beta})), \quad \tilde{\boldsymbol{R}} = \tilde{\boldsymbol{A}}\boldsymbol{C}\tilde{\boldsymbol{A}}' + \tilde{\boldsymbol{P}}, \tag{22}$$

where the matrices $\tilde{\boldsymbol{A}}$ and $\tilde{\boldsymbol{P}}$ are

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{a}_j \end{bmatrix} \qquad \tilde{\boldsymbol{P}} = \begin{bmatrix} \boldsymbol{P} & 0 \\ 0 & \zeta^2 \end{bmatrix}, \tag{23}$$

and $\boldsymbol{a}_j$ is a $1 \times n$ vector of zeros except for entry j which equals one. The mean in equation (22) is computed efficiently using DFT when $\boldsymbol{C}$ is a block circulant matrix, see Appendix. The computationally demanding part of the improved approximation is factorising the term involving $(k+1)\times(k+1)$ covariance matrix $\tilde{\boldsymbol{R}}$.

Recall that the improved approximation $\tilde{\pi}(x_j|\boldsymbol{y},\boldsymbol{\theta})$ is an additional calculation after the direct Gaussian approximation has been fitted at the joint mode. The additional calculations are i) finding the conditional mean for fixed x_j, see equation (22), and ii) evaluating the approximate marginal in equation (20) at this conditional mean. Alternatively, we could evaluate the improved approximation at the new conditional mode, but using the mean is faster and could be as accurate (Hsiao et al., 2004). An improved approximation for the marginal $\pi(x_j|\boldsymbol{y})$ can be obtained by integrating out the model parameter $\boldsymbol{\theta}$, like we did in Section 3.2.

6 Example of improved approximation

For illustrating the improved approximation we consider another example in Diggle et al. (1998) and the rainfall data with different number of days. A goodness-of-fit (GOF) criterion is used to compare the direct and improved approximations to $\pi(x_j|\boldsymbol{y},\boldsymbol{\theta})$. As the 'truth' we use an approximation obtained by a very long MCMC run. Our GOF criterion is based on splitting the sample space of x_j into 10 bins and comparing the percentage of the various approximate densities that fall in each of these bins. We use

$$GOF_a = \sum_{q=1}^{10} \frac{(\hat{F}_{a,q} - F_q)^2}{F_q}, \tag{24}$$

where F_q is the 'true' percentage in bin q estimated by MCMC sampling. Further, $\hat{F}_{a,q}$ denotes the percentage in bin q for approximation a, where a indicates either direct or improved approximation. Such a GOF criterion is an quantitative index for checking the approximation, and can be used along with qualitative plotting. In traditional GOF tests a similar statistic has a chi-square distribution with degrees of freedom equal to the number of bins minus one. For reference we hence give $\chi^2_{0.95} = 16.9$ and $\chi^2_{0.99} = 21.6$ as upper percentiles.

6.1 Infections in Lancashire

The campylobacter infection data in Lancashire district consists of $k = 238$ registration sites, see Figure 6. The observations y_i = number of campylobacter infection out of enteric infections m_i, $i = 1, \ldots, k$. Each observation is tied to a postal code at a spatial registration site. The counts m_i are small, ranging from 1 to 13 in this example. The infection data are modeled by a spatial GLMM with a binomial distribution in equation (4). For the spatial latent variable we use a regular grid of size $n_1 = 135$, $n_2 = 224$ covering the region. Following Diggle et al. (1998) and Steinsland (2007) an exponential covariance function is used, see equation (2). We fix the hyperparameter $\boldsymbol{\theta} = (\sigma, \nu) = (1, 50)$ in this example. This corresponds to quite likely parameter values (Steinsland, 2007).

In Figure 7 we show plots of the marginal density for x_j at two different spatial locations

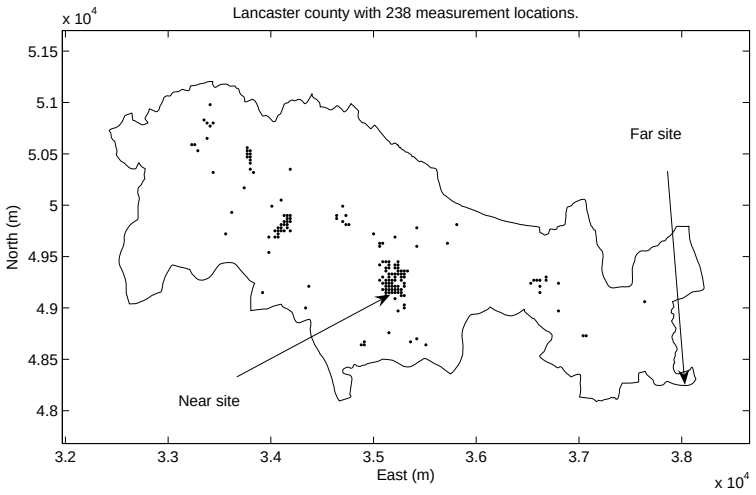


Figure 6: Lancashire county with 238 registration sites for campylobacter infection. The selected prediction sites near registrered data and far away are indicated by arrows.

j. The two locations are $(49250, 35225)$ and $(48250, 38000)$, see Figure 6, representing near and far away from registration sites. Figure 7 shows three approximations to the marginal density $\pi(x_j|\boldsymbol{y}, \boldsymbol{\theta})$ for each of the two locations. In Figure 7 (top), which displays results of a location only one grid cell (30m) from registration sites, we see that the direct Gaussian approximation (solid) is slightly biased to the left, while the improved approximation (crossed) and the MCMC approximation (dashed) are very similar. In this case the improved approximation does have some effect on the approximate marginal. Site j is near registration sites and there is much non-Gaussian influence. Hence, the joint Gaussian at the mode does not capture all features of the marginal density. In Figure 7 (below), which displays results of a location about 800m from the nearest registration site, the three plots are almost identical. This indicates that the direct approximation is accurate when there is less non-Gaussian influence, which is quite natural.

We use the GOF criterion in equation (24) to assess the quality of the approximations. For the direct Gaussian approximation the GOF is 13 (near site) and 2 (far site). For the improved approximation the GOF is 0.33 (near site) and 0.2 (far site). This tells us that the improved approximation gets only slightly worse as we move close to registration sites, while the quality of the direct approximation is poorer with a 6 times increase in GOF.

In this example the direct Gaussian approximation is again 'practically sufficient', meaning that for most purposes the moderate GOF values and the slight differences between the solid and dashed curves in Figure 7 have no effect. The improved approximation has very small GOF values and lies almost on top of the Monte Carlo solutions. We say that the improved approximation is 'practically exact', meaning that MCMC methods cannot detect any possible differences between the improved approximation and the exact, unknown solution.

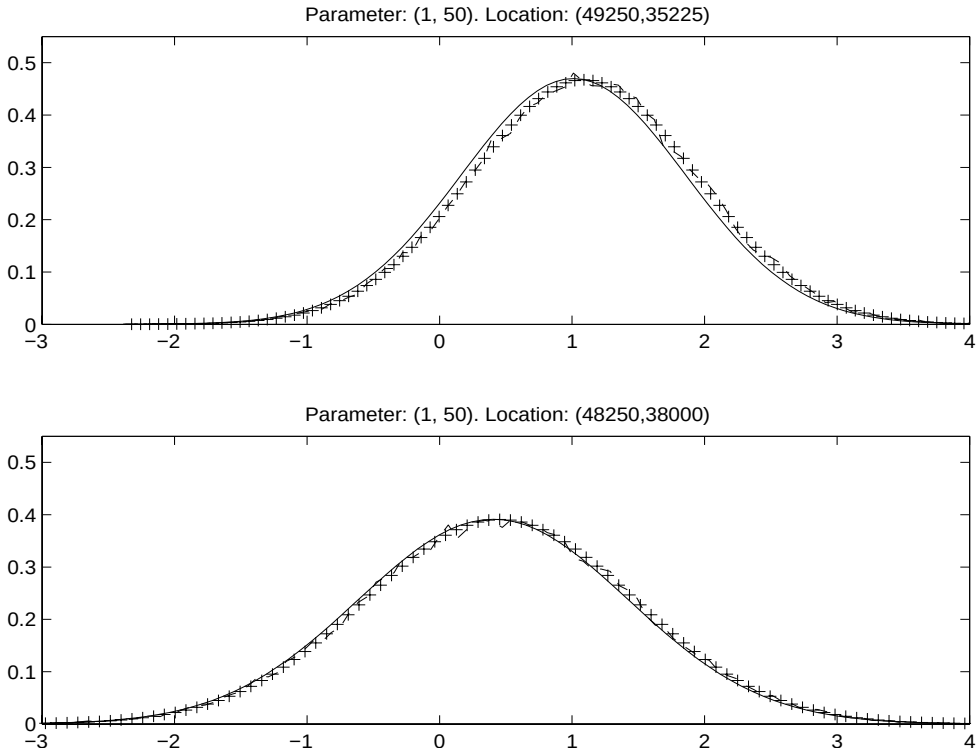


Figure 7: Lancashire dataset. Conditional density $\pi(x_j|\boldsymbol{y}, \boldsymbol{\theta})$ obtained by approximate inference at two spatial locations and for parameter $\boldsymbol{\theta}$ fixed at ($\sigma = 1$, $\nu = 50$m). Solid is direct Gaussian approximation, crossed is improved Gaussian approximation, and dashed is approximation obtained by MCMC sampling.

6.2 Precipitation data in middle Norway

We re-analyse the rainfall data from Section 4.2. The number of trials, m_i, $i = 1, \ldots, k$, in the binomial distribution is now varied from 61 (two months), to 14 (two weeks), and to 2 (two days). We study the quality of the approximations to $\pi(x_j|\boldsymbol{y}, \boldsymbol{\theta})$ for one site j that is near registration sites and one far from registrered data. The far site is at longitude 14.1 and latitude 65.1, i.e. about 66 km from registration sites, while the near site is at longitude 10.6 and latitude 63.4, i.e. about 8 km from the nearest registration site. Finally, we choose two different sets of the parameters $\boldsymbol{\theta}$. These are i) range $\nu = 22$km and standard deviation $\sigma = 0.37$ which is a parameter proximal to the mode from Section 4.2. ii) $\nu = 260$km and $\sigma = 0.54$ which is a more distal parameter value.

In Table 1 we summarise the GOF criterion in equation (24) for the direct (left) and improved (right) approximations. We see that the quality of the approximations decreases as the number of days gets smaller. Nevertheless, the improved approximation stays very good and is 'practically exact'. The direct approximation also has small GOF values, but seems not so good for the two-day case with a GOF of 44 for the near site. Yet, a visual inspection looks very similar to the one in Figure 7 and we still claim that the direct approximation is 'practically sufficient'. For the dataset with only two days there are 3

Table 1: Rainfall dataset. The direct approximation (left) and the improved approximation (right) are compared with a long-run MCMC solution using the goodness-of-fit criterion.

		Direct Gauss		Improved Gauss	
		Near site	Far site	Near site	Far site
m=61	Proximal $\boldsymbol{\theta}$	2.8	1.5	0.12	0.07
m=61	Distal $\boldsymbol{\theta}$	4.1	1.7	0.71	0.11
m=14	Proximal $\boldsymbol{\theta}$	20	13	0.28	0.12
m=14	Distal $\boldsymbol{\theta}$	19	7.5	1.2	0.22
m=2	Proximal $\boldsymbol{\theta}$	44	35	0.67	0.45
m=2	Distal $\boldsymbol{\theta}$	44	16	2.2	0.69

registration sites with no rain, 28 with one day of rain, and 61 with two days of rain.

For all cases in Table 1 the far-site approximations are better than the near-site ones. This is caused by the skewed non-Gaussian data at registration sites. As we move away from registration sites the smooth Gaussian prior has more influence.

For the direct approximation there seems to be no effect of proximal or distal parameter values $\boldsymbol{\theta}$. For the improved approximation we notice that the approximations with proximal parameters are always better. This might be due to a skewed true $\pi(x_j|\boldsymbol{y}, \boldsymbol{\theta})$, while we use a Gaussian approximation $\tilde{\pi}(x_j|\boldsymbol{y}, \boldsymbol{\theta})$. For distal parameters this Gaussian approximation might not be accurate enough and we could consider using more x_j-values to fit a non-Gaussian approximation. This last issue requires more research.

7 Conclusions

In this paper we present fast Bayesian approximations of posterior marginals for a very common geostatistical model with a latent Gaussian field. The approximations are accurate for non-Gaussian data, deterministic and fast to compute since they are based on DFT calculations. We formulate two methods for approximate predictive inference. The first is 'practically sufficient' in all examples we studied, meaning that for purposes regarding decisions or model assessment the approximation is accurate enough. The improved version, which provides a correction to the first approximation, is 'practically exact' in all examples we studied, meaning that we only confirmed the approximation when using very long MCMC simulations. In our opinion one would have to run Markov chains for very much longer than is typically done to verify any possible bias of the improved version. Moreover, our methods of approximate inference allow us to perform high-level inference tasks such as outlier detection and spatial design.

The core of the approximations is a Gaussian approximation to the full conditional for latent spatial variable. If the likelihood has a shape that induces a bimodal full conditional, the approximations will not work. Similarly for likelihood models that do not allow us to

fit a positive definite covariance. For data with small counts the approximations become slightly worse, but remain accurate because of the smooth Gaussian prior which in principle means borrowing data from all registration sites. Approximate predictions are more accurate far from registration sites. Near registration sites we recommend trying the improved version. If this is very similar to the direct one, we trust the direct approximation also when we go further from registration sites.

We briefly discuss the computational costs and limitations. If k becomes very large, say $k > 5000$, the computational cost of matrix inversion seems too high. The dimension of the parameter $\boldsymbol{\theta}$ is a limitation of the method. If there exists several explanatory variables, numerical integration is not tractable and more research is needed to provide good Bayesian solutions in this case. See Rue et al. (2007) for some tentative results in this direction.

References

Ainsworth, L. M. and Dean, C. B. (2006). Approximate inference for disease mapping. *Computational Statistics & Data Analysis*, 50:2552–2570.

Banerjee, S., Carlin, B. P., and Gelfand, A. E. (2004). *Hierarchical Modeling and Analysis for Spatial Data*, volume 101 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Booth, J. G. and Hobart, J. P. (1999). Maximising generalised linear mixed model likelihoods with an automated monte carlo em algorithm. *Journal of the Royal Statistical Society, Series B*, 61:265–285.

Breslow, N. E. and Clayton, D. G. (1993). Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association*, 88:9–25.

Chan, G. and Wood, A. T. A. (1997). An algorithm for simulating stationary gaussian random fields. *Applied Statistics*, 46:171–181.

Chib, S. (1995). Marginal likelihood from the gibbs output. *Journal of American Statistical Association*, 90:1313–1321.

Christensen, O. F., Roberts, G. O., and Sköld, M. (2006). Robust MCMC for spatial GLMM's. *Journal of Computational and Graphical Statistics*, 15:1–17.

Clyde, M. and George, E. I. (2004). Model uncertainty. *Statistical Science*, 19:81–94.

Cressie, N. A. C. (1993). *Statistics for spatial data*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. John Wiley & Sons Inc., New York. Revised reprint of the 1991 edition, A Wiley-Interscience Publication.

Diggle, P. and Lophaven, S. (2006). Bayesian Geostatistical designs. *Scandinavian Journal of Statistics*, 33(1):53–64.

Diggle, P. J. and Ribeiro, P. J. J. (2006). *Model-based geostatistics*. Springer series in Statistics. Springer.

Diggle, P. J., Tawn, J. A., and Moyeed, R. A. (1998). Model-based geostatistics (with discussion). *Journal of the Royal Statistical Society, Ser. C*, 47(3):299–350.

French, J. L. and Wand, M. P. (2004). Generalized additive models for cancer mapping with incomplete covariates. *Biostatistics*, 5:177–191.

Gel, Y., Raftery, A. E., and Gneiting, T. (2004). Calibrated probabilistic mesoscale weather field forecasting: The geostatistical output perturbation method. *Journal of the American Statistical Association*, 99:575–583.

Gray, R. M. (2002). Toeplitz and circulant matrices: A review. Free book available from http://ee.stanford.edu/∼gray, Department of Electrical Engineering, Stanford University.

Hsiao, C. K., Huang, S. Y., and Chang, C. W. (2004). Bayesian marginal inference via cadidate's formula. *Statistics and Computing*, 14(1):59–66.

Johnson, V. E. (2004). A bayesian χ^2 test for goodness-of-fit. *The Annals of Statistics*, 32:2361–2384.

Lewis, S. M. and Raftery, A. E. (1997). Estimating bayesian factors via posterior simulation with the laplace–metropolis estimator. *Journal of the American Statistical Association*, 92:648–655.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models*. Chapman and Hall, London.

Ng, E. S. W., Carpenter, J. R., Goldstein, H., and Rasbash, J. (2006). Estimation in generalised linear mixed models with binary outcomes by simulated maximum likelihood. *Statistical Modelling*, 6:23–41.

O'Hagan, A. (2003). Hsss model criticism. In Green, P. J., Hjort, N. L., and Richardson, S., editors, *Highly Structured Stochastic Systems*, pages 423–444.

Paciorek, C. J. and Ryan, L. (2005). Computational techniques for spatial logisitc regression with large datasets. Technical Report 32, Harvard University Biostatistics Working Paper Series.

Polasky, A. and Solow, A. R. (2001). The value of information in reserve site selection. *Biodiversity and Conservation*, 10:1051–1058.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1996). *Numerical Recipes in C: The art of Scientific Computing*. University Press, Cambridge.

Raftery, A. E. (1996). Approximate bayes factors and accounting for model uncertainty in generalised linear models. *Biometrika*, 83:251–266.

Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods*. Springer-Verlag, New York.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*, volume 104 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Rue, H. and Martino, S. (2006). Approximate Bayesian inference for hierarchical Gaussian Markov random fields models. *Journal of Statistical Planning and Inference*, 137:3177–3192.

Rue, H., Martino, S., and Chopin, N. (2007). Approximate bayesian inference for latent gaussian models using integrated nested laplace approximations. Statistics Report No. 1, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway.

Scollnik, D. P. M. (1995). Bayesian analysis of two overdispersed poisson models. *Biometrics*, 51:1117–1126.

Skaug, H. J. and Fournier, D. A. (2006). Automatic approximation of the marginal likelihood in non-gaussian hierarchical models. *Computational Statistics & Data Analysis*, 51:699–709.

Stein, M. L. (1999). *Interpolation of Spatial Data: Some Theory for Kriging*. Springer-Verlag, New York.

Steinsland, I. (2007). Parallel exact sampling and evaluation of gaussian markov random fields. *Computational Statistics & Data Analysis*, 52:2969–2981.

Tierney, L. and Kadane, J. B. (1986). Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86.

Vidoni, P. (2006). Response prediction in mixed effects models. *Journal of Statistical Planning and Inference*, 136:3948–3966.

Zhang, H. (2002). On estimation and prediction for spatial generalized linear mixed models. *Biometrics*, 58:129–136.

Zhang, H. (2003). Optimal interpolation and the appropriateness of cross-validating variogram in spatial generalized linear mixed models. *Journal of Computational and Graphical Statistics*, 12:698–713.

A Appendix

A.1 Stationary Gaussian prior distribution on a regular grid

Let $\boldsymbol{x} = (x_1, \ldots, x_n)'$ be a Gaussian variable represented on a regular $n_1 \times n_2$ grid, $n = n_1 n_2$. We refer to the $n_1 \times n_2$ matrix $\boldsymbol{x}^m = (x^m_{0,0}, x^m_{0,1}, \ldots, x^m_{n_1-1,n_2-1})$ as the matrix associate of length n vector $\boldsymbol{x}$. The covariance matrix $\boldsymbol{C}$ is defined by the covariance between $x^m_{0,0}$ and all other variables as they are positioned on a torus. In this way $\boldsymbol{C}$ is a block circulant covariance matrix, and we arrange the n covariance entries in an $n_1 \times n_2$ matrix which we denote $\boldsymbol{c}^m$. We further collect the n eigenvalues of $\boldsymbol{C}$ in an $n_1 \times n_2$ matrix $\boldsymbol{\lambda}^m = \text{dft2}(\boldsymbol{c}^m)$ (Gray, 2002). Here dft2 denotes the two dimensional discrete Fourier transform, i.e.

$$\text{dft2}(c^m)_{j_1', j_2'} = \sum_{j_1=0}^{n_1-1} \sum_{j_2=0}^{n_2-1} c^m_{j_1', j_2'} \exp[-2\pi\iota(\frac{j_1 j_1'}{n_1} + \frac{j_2 j_2'}{n_2})], \quad j_1' = 1, \ldots, n_1, j_2' = 1, \ldots, n_2, \tag{25}$$

with $\iota = \sqrt{-1}$. The determinant of $\boldsymbol{C}$ is the product of all entries in $\boldsymbol{\lambda}^m$. We denote by $\text{idft2}(\boldsymbol{d}^m)$ the two dimensional inverse discrete Fourier transform of $n_1 \times n_2$ matrix $\boldsymbol{d}^m$.

Consider first matrix $\boldsymbol{C}$ multiplied with length n vector $\boldsymbol{v}$. The $n_1 \times n_2$ matrix associate of vector $\boldsymbol{w} = \boldsymbol{C}\boldsymbol{v}$ can be evaluated by

$$\boldsymbol{w}^m = \text{Re}\{\text{dft2}[\text{dft2}(\boldsymbol{c}^m) \odot \text{idft2}(\boldsymbol{v}^m)]\}, \tag{26}$$

where $\odot$ represents elementwise multiplication. Further, $\boldsymbol{w} = \boldsymbol{C}^a\boldsymbol{v}$ is given by

$$\boldsymbol{w}^m = \text{Re}\left(\text{dft2}\{[\text{dft2}(\boldsymbol{c}^m)]^{\circledcirc a} \odot \text{idft2}(\boldsymbol{v}^m)\}\right), \tag{27}$$

where $[\cdot]^{\circledcirc a}$ means taking every element to the power of a. This last relation is useful for evaluation and sampling (Rue and Held, 2005). For *evaluation* of the quadratic form we use that $\boldsymbol{v}'\boldsymbol{C}^{-1}\boldsymbol{v} = \boldsymbol{v}'\boldsymbol{w}$, where $\boldsymbol{w}^m$ is evaluated in equation (27) with $a = -1$. For *sampling* we let $\boldsymbol{v}^m$ denote a $n_1 \times n_2$ matrix of independent variables with mean 0 and standard deviation 1. A variable $\boldsymbol{w} \sim N(\boldsymbol{w}; 0, \boldsymbol{C})$ can be obtained via its matrix associate using equation (27) with $a = 1/2$ (Chan and Wood, 1997).

A.2 Conjugate Gaussian posterior distribution:

Suppose we have prior distribution $\pi(\boldsymbol{x}) = N(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{C})$, $\boldsymbol{\mu} = \boldsymbol{1}_n\mu_0$, and likelihood $\pi(\boldsymbol{z}|\boldsymbol{x}) = N(\boldsymbol{z}; \boldsymbol{A}\boldsymbol{x}, \boldsymbol{P})$, $\boldsymbol{z} = (z_1, \ldots, z_k)'$, where $\boldsymbol{A}$ denotes the sparse $k \times n$ matrix of 0s and 1s in equation (1) and assume that $n \gg k$. The posterior is $\pi(\boldsymbol{x}|\boldsymbol{z}) = N(\boldsymbol{x}; \boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{z}}, \boldsymbol{V}_{\boldsymbol{x}|\boldsymbol{z}})$, where

$$\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{z}} = \boldsymbol{\mu} + \boldsymbol{C}\boldsymbol{A}'\boldsymbol{R}^{-1}(\boldsymbol{z} - \boldsymbol{A}\boldsymbol{\mu}), \quad \boldsymbol{V}_{\boldsymbol{x}|\boldsymbol{z}} = \boldsymbol{C} - \boldsymbol{C}\boldsymbol{A}'\boldsymbol{R}^{-1}\boldsymbol{A}\boldsymbol{C}, \quad \boldsymbol{R} = \boldsymbol{A}\boldsymbol{C}\boldsymbol{A}' + \boldsymbol{P}. \tag{28}$$

The mean is efficiently computed using equation (26) which now reads

$$\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{z}} = \boldsymbol{\mu} + \boldsymbol{u}, \quad \boldsymbol{u}^m = \text{Re}\{\text{dft2}[\text{dft2}(\boldsymbol{c}^m) \odot \text{idft2}(\boldsymbol{t}^m)]\}, \tag{29}$$

and $\boldsymbol{t}^m$ is the matrix associate of $\boldsymbol{t} = \boldsymbol{A}'\boldsymbol{R}^{-1}(\boldsymbol{z} - \boldsymbol{A}\boldsymbol{\mu})$ calculated by

$$t_j = \begin{cases} \sum_{i'=1}^{k} R^{-1}_{i,i'}(z_{i'} - \mu_0) & \text{if} \quad s_i \in j \\ 0 & \text{else} \end{cases}, \quad i = 1, \ldots, k, \;\; j = 1, \ldots, n, \tag{30}$$

using the properties of $k \times n$ matrix $\boldsymbol{A}$. The matrix inversion in equation (30) is for $k \times k$ matrix $\boldsymbol{R}$ and k is of moderate size.

For *evaluation* of this posterior we can use that

$$\pi(\boldsymbol{x}|\boldsymbol{z}) = \frac{\pi(\boldsymbol{z}|\boldsymbol{x})\pi(\boldsymbol{x})}{\pi(\boldsymbol{z})}, \quad \pi(\boldsymbol{z}) = N(\boldsymbol{z}; \boldsymbol{A}\boldsymbol{\mu}, \boldsymbol{R}). \tag{31}$$

The prior is evaluated using the relationship following equation (27), while the other expressions only involve $k \times k$ matrices and with moderate k these are fast to evaluate. We can *sample* from the posterior as follows: i) Draw a sample from the unconditional density, $\boldsymbol{v} \sim N(\boldsymbol{v}; \boldsymbol{\mu}, \boldsymbol{C})$ using the relationship following equation (27). ii) Draw a sample $\boldsymbol{w} \sim N(\boldsymbol{w}; \boldsymbol{z}, \boldsymbol{P})$. iii) Set

$$\boldsymbol{x} = \boldsymbol{v} + \boldsymbol{C}\boldsymbol{A}'\boldsymbol{R}^{-1}(\boldsymbol{w} - \boldsymbol{A}\boldsymbol{v}) = \boldsymbol{v} + \boldsymbol{u}, \tag{32}$$

where $\boldsymbol{u}$ is computed from its matrix associate using equation (26). Taking the mean and covariance of this expression, we get the correct values in equation (28).

Note that DFT is $O(n \log n)$, while the matrix inversion is $O(k^3)$. Finding all conditional variance components is $O(nk^2)$, and the most time consuming part for us with $n \sim 10000$, $k \sim 100$.

A.3 Newton-Raphson optimisation for non-Gaussian likelihood:

For the case with non-Gaussian likelihood we find the posterior mode of $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ for fixed $\boldsymbol{\theta}$ by iterative linearisation using the Newton-Raphson algorithm. This is identical to repeated use of equation (29) along with re-setting the transformed measurement $\boldsymbol{z}$ in equation (9) and diagonal $k \times k$ matrix $\boldsymbol{P}$ to account for the non-Gaussian likelihood. The iteration is initiated at a fixed linearisation point $\boldsymbol{x}_s^0$.

Let $\boldsymbol{x}_s^1$ denote the approximate posterior mean in equation (29) obtained by one Newton-Raphson iteration. Define a new transformed measurement by $\boldsymbol{z} = \boldsymbol{z}(\boldsymbol{y}, \boldsymbol{x}_s^1)$ as in equation (9) and $\boldsymbol{P} = \boldsymbol{P}(\boldsymbol{x}_s^1)$. Use this $\boldsymbol{z}$ and $\boldsymbol{P}$ to compute a new posterior mean $\boldsymbol{x}^2$ in equation (29). This iterative process continues until reaching the argument at the posterior mode denoted by $\hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}$.

A.4 Evaluation of the Laplace approximation:

We evaluate the approximate Gaussian posterior $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ using Bayes formula in a similar manner as in equation (31). This gives

$$\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) = N(\boldsymbol{x}; \hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}, \hat{\boldsymbol{V}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}) = \frac{\hat{\pi}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})}{\hat{\pi}(\boldsymbol{z}|\boldsymbol{\theta})}, \quad \boldsymbol{z} = \boldsymbol{z}(\boldsymbol{y}, \hat{\boldsymbol{m}}_{\boldsymbol{x}_s|\boldsymbol{y},\boldsymbol{\theta}}), \tag{33}$$

where $\hat{\pi}(\boldsymbol{z}|\boldsymbol{\theta}) = N(\boldsymbol{z}; \boldsymbol{A}\boldsymbol{\mu}, \boldsymbol{R})$, $\hat{\pi}(\boldsymbol{z}|\boldsymbol{x}) = N(\boldsymbol{z}; \boldsymbol{A}\boldsymbol{x}, \boldsymbol{P})$. The $k \times k$ matrices $\boldsymbol{R}$ and $\boldsymbol{P}$ are now evaluated at the argument at the posterior mode $\hat{\boldsymbol{m}}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}$ from the last Newton-Raphson step. For the Laplace approximation in equation (11) this means that

$$\hat{\pi}(\boldsymbol{\theta}|\boldsymbol{y}) \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\hat{\pi}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})} = \frac{\pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{\theta})\hat{\pi}(\boldsymbol{z}|\boldsymbol{\theta})}{\hat{\pi}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{\theta})}, \quad \boldsymbol{z} = \boldsymbol{z}(\boldsymbol{y}, \hat{\boldsymbol{m}}_{\boldsymbol{x}_s|\boldsymbol{y},\boldsymbol{\theta}}). \tag{34}$$

The expression only involves $k \times k$ matrices and with moderate k these are fast to evaluate.

Paper IV

Approximate Bayesian Inference for Multivariate Stochastic Volatility Models.

Approximate Bayesian Inference for Multivariate Stochastic Volatility Models

Sara Martino
Department of Mathematical Sciences
NTNU, Norway

Abstract

In this report we apply Integrated Nested Laplace approximation (INLA) to a series of multivariate stochastic volatility models. These are a useful construct in financial time series analysis and can be formulated as latent Gaussian Markov Random Field (GMRF) models. This popular class of models is characterised by a GMRF as the second stage of the hierarchical structure and a vector of hyperparameters as the third stage.

INLA is a new tool for fast, deterministic inference on latent GMRF models which provides very accurate approximations to the posterior marginals of the model. We compare the performance of INLA with that of some Markov Chain Monte Carlo (MCMC) algorithms run for a long time showing that the approximations, despite being computed in only a fraction of time with respect to MCMC estimations, are practically exact.

The INLA approach uses numerical schemes to integrate out the uncertainty with respect to the hyperparameters. In this report we cope with problems deriving from an increasing dimension of the hyperparameter vector. Moreover, we propose different approximations for the posterior marginals of the hyperparameters of the model. We show also how Bayes factors can be efficiently approximated using the INLA tools thus providing a base for model comparison.

1 Introduction

1.1 Stochastic volatility models

Financial time series, such as stock returns and exchange rates, present often a non stationary volatility. Volatility is not directly observable in the financial markets, but presents some characteristics which are commonly seen in asset returns. For example, it shows clusters over time, that is there are period of high volatility followed by periods of low volatility. Moreover, it is often stationary and evolves in time in a continuous manner, that is volatility jumps are rare. A typical time series of financial data is represented in

Figure 1. The data are a time series of log-returns of pound-dollar daily exchange rates from October 1st, 1981 to June 28th,1985. In Figure 1 are clearly visible the time varying

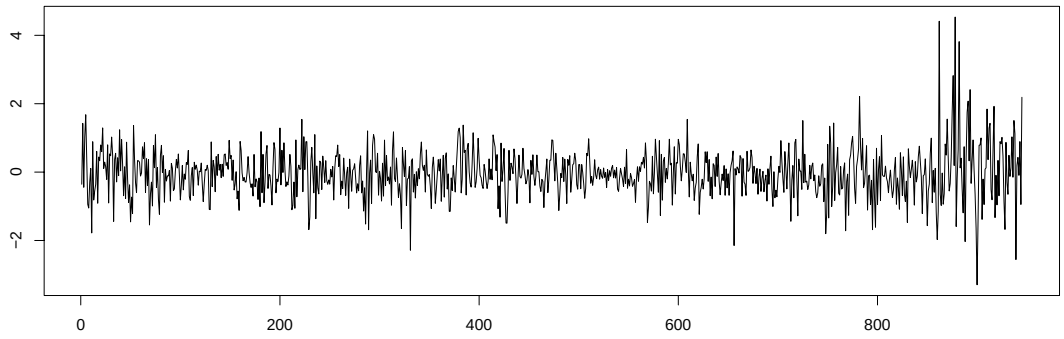


Figure 1: Log-returns of Pound-dollar daily exchange rate from October 1st, 1981 to June 28th,1985.

nature of the volatility and the presence of clusters, for example in the right side of the plot.

The issue of modelling returns accounting for time varying volatility has been widely analysed in the literature. A common model used for returns is defined as:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \sim \text{IID}(0, 1) \tag{1}$$

In (1), ϵ_t, $t = 1, \dots$ is a series of uncorrelated standardised random variable often (but not necessarily) assumed to be Gaussian, and σ_t is the time varying volatility. Model (1) could easily be generalised to allow for a non zero mean. Anyway, for asset returns the behaviour of the conditional mean is, usually, relatively simple, in most cases it is just a constant. Hence, we consider only mean-centred series.

A popular way to look at volatility, is to consider it as a non observed random variable and model its squared logarithm, $h_t = \log \sigma_t^2$, as a linear stochastic process, for example an autoregressive model of order 1 (AR1),

$$h_t = \mu + \phi(h_{t-1} - \mu) + \eta_t, \quad \eta_t \sim \mathcal{N}(0, 1/\tau_\eta) \tag{2}$$

These kind of models, named stochastic volatility (SV) models, were introduced among others by Taylor (1986) and since then have received much attention. Compared to the other class of models for time varying volatility in finance time series, the generalised auto regressive conditional heteroscedasticity (or GARCH) models, SV models are more sophisticated and present some theoretical advantages. GARCH models treat the volatility as a deterministic function of previous observation and past variances, so that the one step ahead forecast is fully determined. The additional error term makes the SV models more flexible than the GARCH ones, see for example Kim et al. (1998). Moreover SV models represent the natural discrete time versions of the continuous time models upon which

much of modern finance theory has been developed. SV models allow for the excess positive kurtosis which is often observed in asset returns and for volatility clustering. Conditions for stationarity of the volatility time series are also easily determined.

The main drawback of SV models is that they are difficult to estimate. Unlike GARCH models where the covariance structure at time t is known given the information up to time $t-1$, the conditional variance is unobserved in SV models. Hence, SV models do not have a closed form for the likelihood function. Maximum likelihood estimation is not possible and, therefore, they require a more statistically and computationally demanding implementation. Another way to understand the difficulty in estimating SV models is to notice that for each data y_t the model uses two innovations, ϵ_t and η_t, instead of just one as in the GARCH model.

Several estimation methods have been proposed for the SV models. They range from the less efficient generalised methods of moments (Andersen and Sorensen, 1996), and quasi likelihood method (Harvey et al., 1994) to more efficient methods such as simulated maximum likelihood (Danielsson, 1994) and Markov Chains Monte Carlo (MCMC). Much attention has been devoted to the development of efficient MCMC algorithms for SV models, e.g. Chib et al. (2002), and Shephard and Pitt (1997), since MCMC is considered one of the most efficient estimation tools, see Andersen et al. (1999).

1.2 Multivariate Stochastic Volatility Models

There are several reasons, both economical and econometric, why multivariate volatility models are important. Financial assets are clearly correlated and the knowledge of such correlation structures is vital in many financial application such as asset pricing, optimal portfolio risk management, and asset allocation. Compared with their univariate counterpart, multivariate models for financial assets have to be able to capture some more features than those mentioned in Section 1.1. Both returns assets and volatility can be cross-dependent. Moreover, volatility can spill over from one market to another so that the knowledge about one asset can help predicting another one. This form of dependency is known as Granger causality.

Multivariate versions exist both for GARCH and SV models. Multivariate GARCH models enjoy a voluminous literature, see, for example Bauwens et al. (2006) for a review. Even though multivariate stochastic volatility (MSV) models have a number of advantages over multivariate GARCH models, the literature on MSV is more limited. This is due to the fact that MSV models pose a series of serious challenges in formulation, estimation and testing. Not only, in fact, they suffer from the inherent problems of multivariate models, such as the high dimensionality of parameter space and the required positive definiteness of covariance matrices but, as for their univariate version, the likelihood has no closed form. There is, however, an increasing interest in MSV models as showed, for example, by Vol. 25 of Econometric Review completely devoted to these models.

1.3 Latent Gaussian Models and Approximate Inference

SV models, as in (1) and (2), and their multivariate counterpart, belong to the larger family of latent Gaussian models. These are a very common construct in statistical analysis and assume a latent Gaussian field $\boldsymbol{x} = \{x_1, \ldots, x_n\}$ to be indirectly observed through n_d conditional independent data $\boldsymbol{y}$. The covariance matrix of the latent Gaussian field and, possibly, the likelihood are governed by a set of hyperparameters, $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_M\}$. We use a Bayesian approach by considering the hyperparameters as random variables with prior density $\pi(\boldsymbol{\theta})$. The goal of the inference is, in general, the posterior distribution

$$\pi(\boldsymbol{x}, \boldsymbol{\theta} \mid \boldsymbol{y}) \;\propto\; \pi(\boldsymbol{\theta})\, \pi(\boldsymbol{x} \mid \boldsymbol{\theta}) \prod_t \pi(y_t \mid x_t, \boldsymbol{\theta}).$$

This is used both for parameter estimation and for filtering or prediction of the latent field.

We are concerned with models where the latent Gaussian field admits conditional independence properties, hence it is a Gaussian Markov random field (GMRF). MCMC is the standard tool for inference in such models. It is, however, not without serious drawbacks. The often large dimension of the latent field, the strong correlation within $\boldsymbol{x}$ and between $\boldsymbol{x}$ and $\boldsymbol{\theta}$, are all possible causes for slow convergence and poor mixing. Block update strategies have been developed aiming to overcome such problems, see for example Knorr-Held and Rue (2002) and Rue et al. (2004). Nevertheless in most cases MCMC algorithms remain very slow.

Rue and Martino (2006) and Rue et al. (2007) propose a deterministic alternative, named Integrated Nested Laplace Approximation (INLA), to MCMC for inference on latent GMRF models. INLA allows fast and accurate approximations to the posterior marginals for x_t and posterior distribution for $\boldsymbol{\theta}$. In the INLA approach, the posterior distribution of the hyperparameters is approximated as:

$$\widetilde{\pi}(\boldsymbol{\theta} \mid \boldsymbol{y}) \propto \left. \frac{\pi(\boldsymbol{x}, \boldsymbol{\theta} | \boldsymbol{y})}{\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x} | \boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x} = \boldsymbol{x}^\star(\boldsymbol{\theta})} \tag{3}$$

In (3), $\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ is a Gaussian approximation to the full conditional for the latent field $\boldsymbol{x}$, and $\boldsymbol{x}^\star(\boldsymbol{\theta})$ is the modal value of $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$. Posterior marginals for the hyperparameters $\widetilde{\pi}(\theta_m|\boldsymbol{y})$ can, in principle, be easily fund via numerical integration of (3). This becomes more involving if the dimension of $\boldsymbol{\theta}$ is large, say above 4.

For the posterior marginals of the latent field Rue et al. (2007) propose to use

$$\widetilde{\pi}(x_t \mid \boldsymbol{y}) = \sum_k \widetilde{\pi}(x_t \mid \boldsymbol{\theta}_k, \boldsymbol{y}) \;\times\; \widetilde{\pi}(\boldsymbol{\theta}_k \mid \boldsymbol{y}) \;\times\; \Delta_k. \tag{4}$$

where the sum is over $\boldsymbol{\theta}$ with area-weights Δ_k, $\widetilde{\pi}(x_t|\boldsymbol{y}, \boldsymbol{\theta})$ is an approximation to the density of $x_t|\boldsymbol{y}, \boldsymbol{\theta}$ and, $\widetilde{\pi}(\boldsymbol{\theta}_k \mid \boldsymbol{y})$ is the approximation in (3). The dimensionality of the sum in (4) depends on the length of vector $\boldsymbol{\theta}$. The approximation $\widetilde{\pi}(x_t|\boldsymbol{y}, \boldsymbol{\theta})$ can either be the Gaussian marginal derived from $\pi_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ or an improved version.

Using INLA it is also possible approximate the marginal likelihood $\pi(\boldsymbol{y})$ as the normalising constant of (3):

$$\widetilde{\pi}(\boldsymbol{y}) = \int \left. \frac{\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})}{\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x}=\boldsymbol{x}^\star(\boldsymbol{\theta})} d\boldsymbol{\theta} \tag{5}$$

The marginal likelihood is a useful quantity for assessing statistical models, see e.g Clyde and George (2004) and Kadane and Lazar (2004). Bayes factor is computed as the ratio of $\pi(\boldsymbol{y})$ for two competing models, therefore efficient computation of marginal likelihood becomes important in model choice.

The computations used in INLA are based on sparse matrix calculations which are much faster that dense matrix ones. The main advantage of INLA over MCMC is computational: results can be obtained in seconds and minutes instead of hours and days. Also, INLA can easily be parallelised and automated.

Rue and Martino (2006) and Rue et al. (2007) provide several examples of applications of INLA for various GMRF models comparing it with long MCMC runs. Their conclusion is that INLA totally outperforms MCMC for both accuracy and speed. Eidsvik et al. (2006) apply the same ideas to geostatistical models, using a different computational approach based on fast discrete Fourier transform for block circulant matrices.

One of the examples used by Rue et al. (2007) to illustrate the performance of INLA is a univariate SV model similar to the one in (1) and (2). In this report we apply INLA to estimate marginal posterior densities for some multivariate SV models. We compare the INLA performance with that of some MCMC algorithms. The main challenge with multidimensional models is the increasing dimension of the hyperparameter vector $\boldsymbol{\theta}$. This, in fact, makes the numerical integration procedures more costly. In this report we verify the CCD integration scheme proposed in Rue et al. (2007) which reduces the cost of numerical integration and propose different way to approximate $\pi(\theta_m|\boldsymbol{\theta})$. We also propose two different approximations for the marginal likelihood, $\pi(\boldsymbol{y})$, and use them as basis for model comparison.

1.4 Plan of the report

Section 2 presents the univariate and multivariate SV models we are interested in, and discusses the choice of prior distributions for $\boldsymbol{\theta}$. Section 3 contains preliminaries about GMRF, the Gaussian approximation $\pi_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ to the full conditional of $\boldsymbol{x}$, and the approximation for $\pi(\boldsymbol{\theta}|\boldsymbol{y})$. Section 4 presents the INLA approach to compute $\widetilde{\pi}(x_t|\boldsymbol{y})$. Two approximations for $\pi(x_t|\boldsymbol{y}, \boldsymbol{\theta})$ are described. In Section 4 we describe how to approximate the marginal likelihood $\pi(\boldsymbol{y})$, and how it can be used to compare models. Examples of applications are presented in Section 6. The problem of approximating marginal posteriors for each hyperparameter $\widetilde{\pi}(\theta_m|\boldsymbol{y})$, is discussed in Section 7. Section 8 explains how INLA can be applied to asymmetric stochastic volatility models. We end with discussion in Section 9.

2 Model description and choice of the prior distribution

Most financial studies involve returns of assets instead of their prices. Campbell et al. (1997) give two main reasons for using returns. First, for average investors, the return is a complete and scale free summary of the investment. Secondly, returns series are easier to handle than price series because the former have more attractive statistical properties. In the literature, there are several definitions of assets returns. Let P_t indicate the price of the asset, or the exchange rate, at time t. The simplest return is called "simple gross return", and defined as

$$1 + R_t = \frac{P_t}{P_{t-1}}$$

In this report we use the continuously compounded return, or *log-return* defined as:

$$y_t = \log(1 + R_t) = \log \frac{P_t}{P_{t-1}}$$

Continuously compounded returns enjoys more tractable statistical properties than simple gross returns, see for example Ruppert (2004).

In this section we describe some SV models (both univariate and multivariate) for log-returns and report some considerations about parametrisation. Finally, we discuss the choice of the prior distribution for $\boldsymbol{\theta}$.

2.1 Univariate Models

Let the series of interest, $\boldsymbol{y} = \{y_1, \ldots, y_n\}$, be made up of a white noise process, with unit variance, multiplied by a time dependent factor σ_t, the standard deviation. In a SV model the logarithm of the standard deviation, $h_t = \log(\sigma_t)$ is unobserved and modelled as a linear stochastic process. A simple, and often used, model for $\boldsymbol{h} = \{h_1, \ldots, h_n\}$ is an auto regressive process of order 1 (AR1). The model is then defined as:

$$y_t = \exp(h_t/2)\epsilon_t, \qquad t = 1, \ldots, n, \quad \epsilon_t \sim \mathcal{N}(0, 1) \tag{6a}$$

$$h_t = \mu + \phi(h_{t-1} - \mu) + \eta_t, \quad t = 1, \ldots, n, \quad \eta_t \sim \mathcal{N}(0, 1/\tau). \tag{6b}$$

with $|\phi| < 1$ to ensure stationarity of the process. The parameter ϕ is sometimes called the persistence parameter. We impose a Gaussian prior to the mean parameter of the latent process, $\mu \sim \mathcal{N}(0, 1/\tau_\mu)$. Hence, by computing the joint density $\pi(h_1, \ldots, h_n, \mu)$, the mean parameter can be included in the latent field. We prefer to include the mean μ in the latent field instead of in the vector of hyperparameters $\boldsymbol{\theta}$ because this is computationally more convenient.

An alternative parametrisation for the SV model in (6) is

$$y_t = \exp(h_t/2)\epsilon_t, \quad t = 1, \ldots, n, \quad \epsilon_t \sim \mathcal{N}(0, 1/\kappa^*). \tag{7a}$$

$$h_t = \phi^* h_{t-1} + \eta_t, \quad t = 1, \ldots, n, \quad \eta_t \sim \mathcal{N}(0, 1/\tau^*). \tag{7b}$$

with $|\phi^*| < 1$ to ensure stationarity. This second parametrisation is used, for example in Durbin and Koopman (2000) and Rue et al. (2007).

The two parametrisation are equivalent since we can write $\log(\kappa^*) = -\mu$, so that the precision term in the likelihood of model (7) corresponds to the mean term of the latent Gaussian files in model (6). The main difference between the two lies in the number of hyperparameters. While model (6), has two hyperparameters, (ϕ, τ), model (7) has three, $(\phi^*, \tau^*, \kappa^*)$. If we use MCMC for inference no big advantage can derive from choosing one or the other. On the other side, in the INLA approach model (6) is preferable since the parameter space is of lower dimensionality. The difference in the hyperparameter space dimensionality between the two parametrisation becomes bigger in the multivariate case. Hence, we parametrise multivariate models in a way similar to (6).

The distribution of ϵ_t in equations (6a) and (7a) does not necessarily have to be Gaussian. If extra kurtosis is needed, we can choose, for example a Student-t distribution with unknown degree of freedom ν. In such case, the dimension of the hyperparameter space becomes 3 and 4 in model (6) and (7) respectively. Considerations regarding the parametrisation hold in exactly the same way.

2.2 Multivariate Models

We describe five different models for multivariate SV as introduced in Yu and Mayer (2006). We focus on the bivariate case but all models presented are amenable to a multidimensional generalisation.

Let $\boldsymbol{I}$ denote the bidimensional unit matrix. Let the observed log-returns at time t, our data, be denoted by $\boldsymbol{y}_t = (y_{t1}, y_{t2})^T$, for $t = 1, \ldots, n$. Let $\boldsymbol{\epsilon}_t = (\epsilon_{t1}, \epsilon_{t2})^T$, $\boldsymbol{\eta}_t = (\eta_{t1}, \eta_{t2})^T$, $\boldsymbol{\mu}_t = (\mu_{t1}, \mu_{t2})^T$ and $\boldsymbol{h}_t = (h_{t1}, h_{t2})^T$. Moreover let

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{pmatrix}, \qquad \boldsymbol{\Sigma}_\epsilon = \begin{pmatrix} 1 & \rho_\epsilon \\ \rho_\epsilon & 1 \end{pmatrix},$$

$$\boldsymbol{\Sigma}_\eta = \begin{pmatrix} 1/\tau_{\eta_1} & \rho_\eta/\sqrt{\tau_{\eta_1}\tau_{\eta_2}} \\ \rho_\eta/\sqrt{\tau_{\eta_1}\tau_{\eta_2}} & 1/\tau_{\eta_2} \end{pmatrix}, \quad \boldsymbol{\Omega}_t = \begin{pmatrix} \exp(h_{1t}/2) & 0 \\ 0 & \exp(h_{2t}/2) \end{pmatrix},$$

In all model considered here we do not use a stationary distribution for $\boldsymbol{h}_t$, rather we assume $\boldsymbol{h}_0 = \boldsymbol{\mu}$.

Model 1 (Basic MSV)

This is the simplest generalisation of the univariate model in (6). It is equivalent to stacking two independent univariate SV models together. The two series are then analysed independently from each other:

$$\begin{aligned} &\boldsymbol{y}_t = \boldsymbol{\Omega}_t \boldsymbol{\epsilon}_t, && \boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \\ &\boldsymbol{h}_t = \boldsymbol{\mu} + \text{diag}(\phi_{11}, \phi_{22})(\boldsymbol{h}_{t-1} - \boldsymbol{\mu}) + \boldsymbol{\eta}_t, && \boldsymbol{\eta}_t \sim \mathcal{N}(\boldsymbol{0}, \text{diag}(1/\tau_{\eta_1}, 1/\tau_{\eta_2})) \end{aligned}$$

This model allows for leptokurtic returns distribution and volatility clustering. However, it does not allow for correlations across returns or across volatility.

Model 2 (Constant correlation MSV)

$$\begin{aligned} \boldsymbol{y}_t &= \boldsymbol{\Omega}_t \boldsymbol{\epsilon}_t, & \boldsymbol{\epsilon}_t &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon) \\ \boldsymbol{h}_t &= \boldsymbol{\mu} + \text{diag}(\phi_{11}, \phi_{22})(\boldsymbol{h}_{t-1} - \boldsymbol{\mu}) + \boldsymbol{\eta}_t, & \boldsymbol{\eta}_t &\sim \mathcal{N}(\mathbf{0}, \text{diag}(1/\tau_{\eta_1}, 1/\tau_{\eta_2})) \end{aligned}$$

This is similar to the multivariate ARCH model proposed by Bollerslev (1990). The returns are correlated but no cross-correlation of the volatility is allowed.

Model 3 (MSV with Granger causality)

$$\begin{aligned} \boldsymbol{y}_t &= \boldsymbol{\Omega}_t \boldsymbol{\epsilon}_t, & \boldsymbol{\epsilon}_t &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon) \\ \boldsymbol{h}_t &= \boldsymbol{\mu} + \boldsymbol{\Phi}(\boldsymbol{h}_{t-1} - \boldsymbol{\mu}) + \boldsymbol{\eta}_t, & \boldsymbol{\eta}_t &\sim \mathcal{N}(\mathbf{0}, \text{diag}(1/\tau_{\eta_1}, 1/\tau_{\eta_2})) \end{aligned}$$

With $\phi_{12} = 0$. This model was first proposed by Yu and Mayer (2006). It allows the second asset to be Granger caused by the the volatility of the first asset. Volatilities are therefore cross-correlated. The correlation between returns is due to both Granger causality and volatility clustering. The model allows also $\phi_{12} \neq 0$. In such case a bilateral Granger causality between the two assets is allowed, we do not take this case into consideration.

Model 4 (Generalised constant correlation MSV)

$$\begin{aligned} \boldsymbol{y}_t &= \boldsymbol{\Omega}_t \boldsymbol{\epsilon}_t, & \boldsymbol{\epsilon}_t &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon) \\ \boldsymbol{h}_t &= \boldsymbol{\mu} + \text{diag}(\phi_{11}, \phi_{22})(\boldsymbol{h}_{t-1} - \boldsymbol{\mu}) + \boldsymbol{\eta}_t, & \boldsymbol{\eta}_t &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\eta) \end{aligned}$$

This model was studied by Harvey et al. (1994) and Danielsson (1998) who used respectively the quasi likelihood and the simulated maximum likelihood methods for estimation. Both returns and volatility are correlated. Clearly, both model 3 and 4 can generate cross-dependence in the volatility, using two different generating mechanisms. Which specification is more appropriate is an interesting question which goes beyond the scope of this report.

Model 5 (Heavy-tailed MSV)

There is some evidence that financial data have heavier tails than those resulting from inserting conditional heteroscedasticity in a Gaussian process. This extra kurtosis can be introduced by using a Student-t distribution instead of a Gaussian in the returns model.

In a univariate context a Student-t distribution is used, for example, in Chib et al. (2002) while in the multivariate SV context it was first used by Harvey et al. (1994) .

$$\begin{aligned} \boldsymbol{y}_t &= \boldsymbol{\Omega}_t \boldsymbol{\epsilon}_t, & \boldsymbol{\epsilon}_t &\sim t(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon, \nu) \\ \boldsymbol{h}_t &= \boldsymbol{\mu} + \text{diag}(\phi_{11}, \phi_{22})(\boldsymbol{h}_{t-1} - \boldsymbol{\mu}) + \boldsymbol{\eta}_t, & \boldsymbol{\eta}_t &\sim \mathcal{N}(\mathbf{0}, \text{diag}(1/\tau_{\eta_1}, 1/\tau_{\eta_2})) \end{aligned}$$

In this model the volatilities are uncorrelated but cross-dependencies in the returns are allowed. It would have been possible to use a different generalisation of the univariate Student-t distribution in a multivariate context, that is assume each variable to be a Student-t with its own degree of freedom. However, according to Yu and Mayer (2006) this model performs empirically worse that the one presented above.

2.3 Choice of prior distributions

In a Bayesian framework, the hyperparameters of the model are considered random variables and assigned a prior distribution $\pi(\boldsymbol{\theta})$. In this section we discuss prior choice for the hyperparameters of the bivariate models presented in Section 2.2. The same considerations hold also for univariate models.

In all models considered we assume a Gaussian prior for the mean parameter $\boldsymbol{\mu}$ so that, by computing the joint density of $\boldsymbol{x} = (\boldsymbol{h}_1, \ldots, \boldsymbol{h}_n, \boldsymbol{\mu})$, it can be included in the latent field. The remaining hyperparameters can be divided into two groups: parameters in the mean equation (ρ_ϵ, ν) and in the variance equation $(\phi_{11}, \phi_{12}, \phi_{22}, \rho_\eta, \tau_{\eta_1}, \tau_{\eta_2})$.

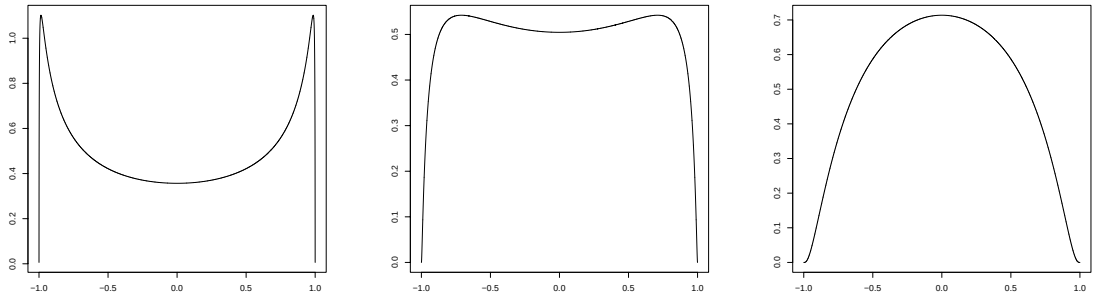
For computational reasons, it is convenient, when applying INLA, that all hyperparameters are defined over the whole real line. Hence, when the original parameters in the model are constrained, we consider a function of them.

We start by defining priors for the hyperparameters in the variance equation. We want the volatility time series to be stationary. This holds if the roots of $\text{diag}(\boldsymbol{I} - \boldsymbol{\Phi} z)$ lie outside the unit circle. For the $\boldsymbol{\Phi}$ matrix in Model 4 this corresponds to $|\phi_{11}| < 1$, $|\phi_{22}| < 1$ and $\phi_{21} \in \mathcal{R}$. We choose a Gaussian prior for ϕ_{21}. As for the two persistence parameters ϕ_{11} and ϕ_{22}, we note that in a univariate AR1 model with persistence parameter $\phi > 0$, the autocorrelation decays like ϕ^κ, where $\kappa > 0$. Define the range of the time series as the distance where the autocorrelation drops below $\alpha = 0.05$. That is $\kappa = \log \alpha / \log \phi$. The range has a "physical" meaning, therefore it is usually easier to interpreter than other parameters. We define, hence, the range of our two time series as $\kappa_1 = \log \alpha / \log \phi_{11}$ and $\kappa_2 = \log \alpha / \log \phi_{22}$ and assign each an exponential prior distribution.
A popular choice for the prior of the precision parameters τ_{η_1} and τ_{η_2}, is Gamma(a, b), with mean a/b and variance a/b^2. We choose a quite vague prior with $a = 0.25$ and $b = 0.025$.
The correlation parameter ρ_η is constrained in the interval $[-1, 1]$. Consider the function

$$f(x) = \text{logit}\left(\frac{x+1}{2}\right); \quad x \in [-1, 1]$$

which assumes values over the whole real line. We choose a Gaussian prior for parameter $\rho_\eta^* = f(\rho_\eta)$ with precision 0.4. This choice of the precision corresponds, roughly, to a

uniform prior in $[-1, 1]$ for the correlation parameter ρ_η. A smaller value for the precision corresponds to a less vague prior for ρ_η. In fact, the distribution of ρ_η derived from a vague Gaussian prior on ρ_η^* assigns most of the probability mass to values close to -1 or 1. A larger precision, on the other side, results in a prior for ρ_η which assign most of the probability mass to values closer to 0, see figure Figure 2.



(a) Precision = 0.2 (b) Precision $= 0.4$ (c) Precision $= 0.8$

Figure 2: Distribution of ρ_η derived from a Gaussian distribution on ρ_η^* with different values of the precision.

We treat the correlation in the mean equation ρ_ϵ in a similar way. Finally, for the degree of freedom for the student-t distribution ν, we consider $\nu^* = \log(\nu - 2)$ and assign a Gaussian prior to ν^*.

All hyperparameters are assumed independent apriori. The prior distributions are listed below:

- $\rho_\epsilon^* \sim \mathcal{N}(0, 0.4)$ where $\rho_\epsilon = f(\rho_\epsilon^*)$
- $\nu^* \sim \mathcal{N}(0, 0.1)$ where $\nu^* = \log(\nu - 2)$
- $\kappa_i^* \sim$ exponential(0.5), where $\kappa_i = \log \alpha / \log \phi_{ii}$ and $i = 1, 2$ and $\alpha = 0.05$
- $\phi_{21}^* \sim \mathcal{N}(0, 0.01)$
- $\rho_\eta^* \sim \mathcal{N}(0, 0.4)$ where $\rho_\eta = f(\rho_\eta^*)$
- $\tau_{\eta_i} \sim$ Gamma$(0.25, 0.025)$ for $i = 1, 2$

3 Gaussian Markov Random Fields

All models in Sections 2.2 and 2.1 can be thought of as different specifications of a general latent GMRF model in three stages. The first stage is a likelihood model for the observables, a two dimensional Gaussian or Student-t distribution. The data are independent conditional on some latent parameters, which in our case consist in the volatility, and, possibly, some additional hyperparameters $\boldsymbol{\theta}_1$. Let $\boldsymbol{y} = (\boldsymbol{y}_1^T, \ldots, \boldsymbol{y}_{n_d}^T)^T$ and

$\boldsymbol{h} = (\boldsymbol{h}_1^T, \ldots, \boldsymbol{h}_n^T)^T$ be two column vectors. Each element of $\boldsymbol{h}$ and $\boldsymbol{y}$ is indexed by two numbers ti where $t = 1, 2, \ldots$ and $i = \{1, 2\}$; that is, t indicates time while i indicates the different assets. For the univariate case the index i is omitted. We assume that each $\boldsymbol{y}_t$ depends only on the corresponding bidimensional vector $\boldsymbol{h}_t$ in the latent field, so that we have:

$$\pi(\boldsymbol{y}|\boldsymbol{h}, \boldsymbol{\theta}_1) = \prod \pi(\boldsymbol{y}_t|\boldsymbol{h}_t, \boldsymbol{\theta}_1) \tag{8}$$

Note that we consider the whole vector $\boldsymbol{y}_t$ as one data point. We say, then, that we have a multivariate model if $\boldsymbol{y}_t$ has dimension greater than one and a univariate model in the other case.

The second stage is a model for the latent field. In the cases analysed here, this is a bivariate autoregressive model of order 1 with an unknown mean and a covariance matrix depending on some hyperparameters $\boldsymbol{\theta}_2$:

$$\boldsymbol{h}_t|\boldsymbol{h}_{t-1}, \boldsymbol{\mu}, \boldsymbol{\theta}_2 \sim \mathcal{N}(\boldsymbol{\mu} + \boldsymbol{\Phi}(\boldsymbol{h}_{t-1} - \boldsymbol{\mu}), \boldsymbol{\Sigma}_\eta) \quad t = 1, \ldots, n$$

With $\boldsymbol{x}_0 = \boldsymbol{\mu}$. Note that it is possible to have $n > n_d$. This is the case, for example, if we are interested in predicting future value of the volatility. We assume a Gaussian prior for the mean term $\boldsymbol{\mu} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_{\boldsymbol{\mu}})$. The mean term $\boldsymbol{\mu}$ can then be included in the latent field by computing the density:

$$\pi(\boldsymbol{h}, \boldsymbol{\mu}|\boldsymbol{\theta}_1) = \pi(\boldsymbol{\mu}) \prod_{t=1}^{n} \pi(\boldsymbol{h}_t|\boldsymbol{h}_{t-1}, \boldsymbol{\theta}_1) \propto |\boldsymbol{Q}|^{1/2} \exp\{-\frac{1}{2}(\boldsymbol{h}^T, \boldsymbol{\mu}^T)\boldsymbol{Q}(\boldsymbol{h}^T, \boldsymbol{\mu}^T)^T\} \tag{9}$$

Where $\boldsymbol{Q}$ is the $N \times N$ precision (inverse of the covariance) matrix. Here $N = 2n + 2$ is the length of the latent vector $\boldsymbol{x} = (\boldsymbol{h}^T, \boldsymbol{\mu}^T)$

The third and last step of our latent Gaussian model is a prior distribution for the hyper-parameter vector $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$, $\pi(\boldsymbol{\theta})$.

The precision matrix in (9) is sparse, meaning that only few of its elements are non-zero. This is a typical characteristic of GMRFs. There is in fact a one to one correspondence between the Markov properties of the field $\boldsymbol{x}$ and the non-zero structure of the precision matrix $\boldsymbol{Q}$, meaning that a off diagonal element $Q_{ij} \neq 0$ if and only if the two random variables x_i and x_j are conditional independent given the rest of the variables in $\boldsymbol{x}$. Great computational efficiency can be achieved by exploiting the sparseness of $\boldsymbol{Q}$. In particular, factorising $\boldsymbol{Q}$ into its Cholesky triangle $\boldsymbol{L}\boldsymbol{L}^T$ can be done in a fast way. The Cholesky triangle $\boldsymbol{L}$ inherits the sparseness of $\boldsymbol{Q}$ thanks to the global Markov property, thus only the non-null terms in $\boldsymbol{L}$ are computed. The nodes in the GMRF can be reordered in such a way to minimise, or reduce, the number of non-null terms in $\boldsymbol{L}$. The Cholesky triangle is then the basis for solving linear equations involving $\boldsymbol{Q}$. For example $\boldsymbol{Q}\boldsymbol{x} = \boldsymbol{b}$ is solved by first solving $\boldsymbol{L}\boldsymbol{v} = \boldsymbol{b}$ and the $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{v}$. This is a typical way to produce random samples from a GMRF. If $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ then the solution of $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ has precision matrix $\boldsymbol{Q}$. Also the log of the density in (9) can easily be computed, for any configuration $\boldsymbol{x}$, using $\boldsymbol{L}$ since $\log|\boldsymbol{Q}| = \sum_i \log L_{ii}$.

If the GMRF is defined with additional linear constraints of the type $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{e}$, where $\boldsymbol{A}$ is a $k \times N$ matrix of rank k and $\boldsymbol{e}$ is a vector of length k, it is possible to correct a sample

$\boldsymbol{x}$ drawn from the unconstrained GMRF in the following way:

$$\boldsymbol{x}^c = \boldsymbol{x} - \boldsymbol{Q}^{-1}\boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^T)^{-1}(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{e}). \tag{10}$$

$\boldsymbol{x}^c$ is then a sample from the constrained density. This method is convenient when the rank of $\boldsymbol{A}$ is small. In fact $\boldsymbol{Q}^{-1}\boldsymbol{A}^T$ is computed by solving k linear systems, one for each column of $\boldsymbol{A}^T$. The additional cost for k linear constraints is $\mathcal{O}(Nk^2)$. This approach is commonly referred to as "conditioning by Kriging", see Cressie (1993) and Rue and Held (2005). For more details about sparse matrix computation see, for example, Rue and Held (2005).

In the GMRF defined in (9) the covariance matrix is only implicitly known. Inverting the precision matrix can be extremely costly due to its dimension. The sparseness of $\boldsymbol{Q}$ comes to help again. To see this, we start with $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ where $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Recall that the solution $\boldsymbol{x}$ has precision matrix $\boldsymbol{Q}$. Writing this out in detail, we obtain $L_{ii}x_i = z_i - \sum_{k=i+1}^{N} L_{ki}x_k$ for $i = N, \ldots, 1$. Multiplying each side with x_j $j \geq i$, and taking expectation, we obtain

$$\Sigma_{ij} = \delta_{ij}/L_{ii}^2 - \frac{1}{L_{ii}} \sum_{k=i+1}^{N} L_{ki}\Sigma_{kj}, \quad j \geq i,\ i = N, \ldots, 1, \tag{11}$$

where $\boldsymbol{\Sigma}$ $(= \boldsymbol{Q}^{-1})$ is the covariance matrix. Thus Σ_{ij} can be computed from (11), letting the outer loop i run from N to 1 and the inner loop j from N to i. If we are only interested in the marginal variances, we only need to compute Σ_{ij}'s for which L_{ji} (or L_{ij}) is not known to be zero. Marginal variances under linear constraints can be computed in a similar way, see Rue and Martino (2006, Sec. 2) for more details.

All computations used by INLA for latent GMRF models are based on algorithms for sparse matrices. The non-zero structure of the precision matrix in (9) is represented in Figure 3. The size of the bandwidth depends on both the order of the AR model and on the size of vector $\boldsymbol{h}_t$. Considering highly multidimensional models or high order AR models makes the precision matrix more dense and therefore the computations less efficient.

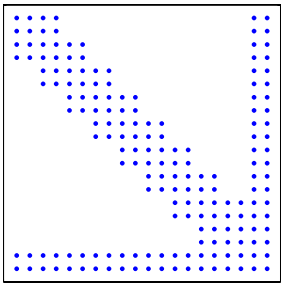


Figure 3: Non zero structure of the precision matrix for a bidimensional AR1 model with unknown mean

3.1 Gaussian Approximation

The core of the INLA approach is a Gaussian approximation to the full conditional of the latent field:

$$\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}) \propto \exp\left\{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \sum_{t=1}^{n_d} g_t(\boldsymbol{x}_t)\right\} \tag{12}$$

where $\boldsymbol{x} = (\boldsymbol{h}^T, \boldsymbol{\mu}^T)$ and $g_t(\boldsymbol{x}_t) = \log \pi(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\theta}_1)$. The approximation, which we denote $\pi_G(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$, is computed by matching the mode of $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ and its curvature at the mode. The mode of $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ is computed using an iterative procedure. Starting from an initial guess $\boldsymbol{m}^{(0)}$ we expand $g_t(\boldsymbol{x}_t)$ around $\boldsymbol{m}_t^{(0)}$ for $t = 1\ldots, n_d$

$$g_t(\boldsymbol{x}_t) \approx g_t(\boldsymbol{m}^{(0)}) + \boldsymbol{b}_t^T\boldsymbol{x}_t - \frac{1}{2}\boldsymbol{x}_t^T\boldsymbol{C}_t\boldsymbol{x}_t \tag{13}$$

where

$$\boldsymbol{C}_t = -\left[\begin{array}{cc} \frac{\partial^2 g_t(\boldsymbol{x}_t)}{\partial x_{t1}^2} & \frac{\partial^2 g_t(\boldsymbol{x}_t)}{\partial x_{t1}\partial x_{t2}} \\ \frac{\partial^2 g_t(\boldsymbol{x}_t)}{\partial x_{t1}\partial x_{t2}} & \frac{\partial^2 g_t(\boldsymbol{x}_t)}{\partial x_{t2}^2} \end{array}\right]_{\boldsymbol{x}_t=\boldsymbol{m}_t^0}$$

and the 2×1 vector $\boldsymbol{b}_t$ is a function of the gradient of $g_t(\boldsymbol{x}_t)$ evaluated at $\boldsymbol{x}_t = \boldsymbol{m}_t^0$. Let diag$(\boldsymbol{C})$ indicate the $N \times N$ matrix

$$\left[\begin{array}{ccccc} \boldsymbol{C}_1 & \boldsymbol{0} & & \ldots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C}_2 & \boldsymbol{0} & \ldots & \boldsymbol{0} \\ \vdots & & & & \\ \boldsymbol{0} & & \ldots & \boldsymbol{C}_n & \boldsymbol{0} \\ \boldsymbol{0} & & & \ldots & \boldsymbol{0} \end{array}\right], \tag{14}$$

that is, diag$(\boldsymbol{C})$ is a band matrix with bandwidth 2. For univariate models diag$(\boldsymbol{C})$ reduces to a diagonal matrix. Moreover, let $\boldsymbol{b}^T = (\boldsymbol{b}_1^T, \boldsymbol{b}_2^T, \ldots, \boldsymbol{0})$. We obtain a Gaussian approximation with precision $\boldsymbol{Q} + \text{diag}(\boldsymbol{C})$ and mean given by the solution of $(\boldsymbol{Q} + \text{diag}(\boldsymbol{C}))\boldsymbol{m}^{(1)} = \boldsymbol{b}$. The process is repeated until it converges to a Gaussian distribution with precision $\boldsymbol{Q}_G = \boldsymbol{Q} + \text{diag}(\boldsymbol{C})$ and mean $\boldsymbol{\mu}_G$. Both the precision matrix and the mean value of the Gaussian approximation depend of the value of the hyperparameters $\boldsymbol{\theta}$. Algorithm 1 displays a naive version of the procedure. In practice some more care has to be put into building the stopping criteria in order to avoid the optimiser to fail. The costly part of Algorithm 1 is solving the linear system in line 7. This operation can be efficiently performed using sparse matrix computations. Note that, since each $\boldsymbol{y}_t$ depends only on $\boldsymbol{x}_t$, the Gaussian approximation $\pi_G(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ preserves the Markov properties of the prior distribution for $\boldsymbol{x}$. This is convenient from a computational point of view.

3.2 Approximating the joint posterior of the hyperparameters $\pi(\boldsymbol{\theta}|\boldsymbol{y})$

The joint posterior for the hyperparameters in the model, $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$, is

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{y})\pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})} \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})} \tag{15}$$

Algorithm 1 Computing the Gaussian approximation $\pi_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$

Given a value for $\boldsymbol{\theta}$ and an initial guess $\boldsymbol{m}^{(0)}$
$\texttt{iter} = 0$, $\texttt{diff} = 10$
while $\texttt{diff} > \alpha$ **do**
  for $t = 1$ to n **do**
    Compute $\boldsymbol{b}_t$ and $\boldsymbol{C}_t$ using (13)
  end for
  Solve $(\boldsymbol{Q} + \text{diag}(\boldsymbol{C}))\boldsymbol{m}^{(1)} = \boldsymbol{b}$
  Compute $\texttt{diff}$ = a distance measure between $\boldsymbol{m}^{(0)}$ and $\boldsymbol{m}^{(1)}$
  Set $\boldsymbol{m}^{(0)} = \boldsymbol{m}^{(1)}$
end while
Return $\boldsymbol{x}_G = \boldsymbol{m}^{(0)}$ and $\boldsymbol{Q}_G = (\boldsymbol{Q} + \text{diag}(\boldsymbol{C}))$

which is valid for any configuration $\boldsymbol{x}$. INLA builds an approximation to the density in (15), for each value of $\boldsymbol{\theta}$, by substituting the denominator $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ with the Gaussian approximation $\pi_G(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ described in Section 3.1, and computing the right hand side of (15) at the modal value $\boldsymbol{\mu}_G(\boldsymbol{\theta})$. That is:

$$\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y}) \propto \left. \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi_G(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x}=\boldsymbol{\mu}_G(\boldsymbol{\theta})} \tag{16}$$

This expression is equivalent to Tierney and Kadane (1986)'s Laplace approximation of a marginal posterior distribution. This suggests that the approximation error is relative and of order $\mathcal{O}(n_d^{-3/2})$ after renormalisation. However standard asymptotic assumption usually invoked for Laplace approximations are not verified here, some considerations about the error rate for the approximation in (16) can be found in Rue et al. (2007).

$\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ can be used to solve three different tasks in the inference process. The main use of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is to integrate out the uncertainty with respect to $\boldsymbol{\theta}$ when computing approximations for the marginal posteriors of the latent field $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ as in (4). Secondly, $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is used to compute an approximation to the marginal likelihood as in (5). Finally, sometimes we are also interested in marginal posteriors for the hyperparameters $\widetilde{\pi}(\theta_m|\boldsymbol{y})$. In this case we have to compute the integrals

$$\widetilde{\pi}(\theta_m|\boldsymbol{y}) = \int \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y}) d\boldsymbol{\theta}_{-m} \quad m = 1, \ldots, M \tag{17}$$

where $\boldsymbol{\theta}_{-m}$ indicates the vector $\boldsymbol{\theta}$ with element m removed.

All these procedures involve numerical integration over a multidimensional domain and, with increasing dimension of $\boldsymbol{\theta}$, computations become soon unfeasible. Even if we are able to locate the area with highest density for $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ and compute the integral on a grid consisting in d points in each direction, the cost of computing the integral is $\mathcal{O}(d^M)$, where M is the dimension of $\boldsymbol{\theta}$, that is, the cost grows exponentially in M.

It turns out that solving the first two tasks is an easier problem. In fact, we only need to explore $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ sufficiently to be able to select good evaluation points for the numerical

integration in (4) and (5): only few points, accurately selected, are enough to achieve satisfying accuracy in (4). With this we mean that the resulting density approximation is indistinguishable from a density estimate obtained from a long MCMC run. We describe this in Section 4.

On the other side, solving integral (17) is more involving. The shape of $\widetilde{\pi}(\theta_m|\boldsymbol{y})$ can be quite irregular and therefore we need more evaluation points to achieve satisfying precision. Moreover the integration needs to be repeated possibly M times. We return to this task in Section 7.

4 Approximating posterior marginals for the latent field

In this section we present INLA for computing approximations for marginal posteriors of the latent field $\pi(x_{ti}|\boldsymbol{y})$ with $t = 1, 2, \ldots$ and $i = 1, 2$. The general strategy is in Algo-

Algorithm 2 INLA strategy for computing $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$

1: Select a set $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$
2: **for** $k = 1$ to K **do**
3: Compute $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$
4: Compute $\widetilde{\pi}(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})$ as a function of x_{ti}
5: **end for**
6: Compute $\widetilde{\pi}(x_{ti}|\boldsymbol{y}) \sum_k \widetilde{\pi}(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})\Delta_k$ as function of x_{ti}, for all indexes ti

rithm 2: first, select a set of configurations $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$ from the hyperparameters space. For each $\boldsymbol{\theta}_k \in \boldsymbol{\Theta}$ compute $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$ as in (16) and an approximation $\widetilde{\pi}(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})$ to the density of $x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y}$. Finally compute the $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ via numerical integration. Note that in Algorithm 2 $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$ is computed for fixed value of $\boldsymbol{\theta}_k$ and, therefore is a scalar, while $\widetilde{\pi}(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})$ is the density distribution of $x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y}$.

For Algorithm 2 to be operative we should first solve two tasks:

1. how to select a (possibly small) set of points $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$
2. how to build a good approximation to $\pi(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})$

We discuss task 1 in Section 4.1 and task 2 in Section 4.2.

4.1 Exploring $\pi(\boldsymbol{\theta}|\boldsymbol{y})$

To compute approximations to the density of $x_{ti}|\boldsymbol{y}$ we need to integrate out the uncertainty with respect to the hyperparameters $\boldsymbol{\theta} \in \mathcal{R}^M$ using numerical integration as in (4). Rue et al. (2007) propose two different ways to explore the domain of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. The first consists in locating a grid over the area with higher density and evaluate $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ at each

point of this grid. This method is quite accurate. It is also efficient when the dimension of $\boldsymbol{\theta}$ is not too high, say less than 4. In cases, like those analysed in this report, where the number of hyperparameters is higher, say between 4 and 11, they propose a different strategy which comes from considering the integration problem as a design problem. This second approach reduces dramatically the computational costs and, in our experience, still gives results which are sufficiently accurate for inference purposes.

We describe the two strategies in Sections 4.1.1 and 4.1.2 respectively. Both strategies assume $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ to be uni-modal. This is the case for most of the real case scenarios. In both cases it is necessary to find the mode of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, denoted as $\boldsymbol{\theta}^*$, and the negative Hessian at the modal configuration $\boldsymbol{H} > 0$. The mode can be found using a multidimensional optimisation algorithm. If the dimension of $\boldsymbol{\theta}$ is high, this operation can be costly, but it has to be done only once. We compute the Hessian using finite differences. The inverse of the negative Hessian $\boldsymbol{\Sigma} = \boldsymbol{H}^{-1}$ would be the covariance matrix if $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ were a Gaussian density.

4.1.1 Exploring $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ using a grid strategy

The idea is to construct a M dimensional grid of points which covers the region of the domain where the majority of the probability mass of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is located. To do this we start by computing the eigen-decomposition $\boldsymbol{\Sigma} = \boldsymbol{V}\boldsymbol{\Lambda}^{1/2}\boldsymbol{V}^T$. Define the variable $\boldsymbol{z}$, such that:

$$\boldsymbol{\theta}(\boldsymbol{z}) = \boldsymbol{\theta}^* + \boldsymbol{V}\boldsymbol{\Lambda}^{1/2}\boldsymbol{z} \tag{18}$$

The variable $\boldsymbol{z} = (z_1, \ldots, z_M)$ is standardised and its components are mutually orthogonal. We explore $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ using the $\boldsymbol{z}$-parametrisation. We start at the mode, $\boldsymbol{z} = \boldsymbol{0}$ and proceed along the z_1 axes, in the positive direction, using a step length of δ_z. We compute $\widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{z})|\boldsymbol{y})$ at this new point and continue as long as

$$\log \widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{0})|\boldsymbol{y}) - \log \widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{z})|\boldsymbol{y}) < \delta_\pi \tag{19}$$

where δ_π is a threshold value. Then, invert the direction and repeat. The same is done for each of the M directions. Once we have located the region of highest probability density, we fill in the grid by exploring all different combinations of the points on the axes. We include these new points only if (19) holds. The procedure is described in Algorithm 3 where $\boldsymbol{1}_i$ indicates a vector on length M whose ith element is 1 an all others are 0.

Since the points are layed out on a regular grid, when computing (4) we can take all the area-weights Δ_k to be equal.

Algorithm 3 has two tuning parameters, the step length δ_z and the threshold δ_π. In general, to obtain satisfying results it is enough to set $\delta_z = 1$ and $\delta_\pi = 2.5$. This means that, if $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ were Gaussian, we would select 5 points on each direction. The number of points to be computed using the grid strategy grows exponentially with the dimension M of the hyperparameters space. This feature makes the grid approach fast only for small hyperparameter spaces.

Algorithm 3 Exploring $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ using a grid strategy

Compute $\boldsymbol{\theta}^*$ and $\boldsymbol{\Sigma} = \boldsymbol{H}^{-1}$
Compute $\boldsymbol{\Sigma} = \boldsymbol{V}\boldsymbol{\Lambda}^{1/2}\boldsymbol{V}^T$
for i in $1 : M$ **do**
  Start at the mode, $\boldsymbol{z} = \mathbf{0}$
  for `dir` in $\{-1, 1\}$ **do**
    while $\log \widetilde{\pi}(\boldsymbol{\theta}(\mathbf{0})|\boldsymbol{y}) - \log \widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{z})|\boldsymbol{y}) < \delta_\pi$ **do**
      $\boldsymbol{z} = \boldsymbol{z} + \texttt{dir} * \mathbf{1}_i$
      Compute $\boldsymbol{\theta}(\boldsymbol{z}) = \boldsymbol{\theta}^* + \boldsymbol{V}\boldsymbol{\Lambda}^{1/2}\boldsymbol{z}$
      Compute $\widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{z})|\boldsymbol{y})$
    end while
  end for
end for
Compute fill in points

4.1.2 Exploring $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ using a central composit design strategy

The idea explained in this section comes from considering the integration problem as a kind of response surface problem: we want to lie out points in a M dimensional space in such a way to learn about the shape of a response surface. We consider second order response surface and use the Box and Wilson (1951) central composit design (CCD). A CCD contains an embedded factorial or fractional design with centre points (design-points) plus an additional group of $2M + 1$ "circle" points which allow to estimate the curvature. All the points in a CCD design lie on the surface of a M dimensional sphere with radius $\sqrt{M}$ times an arbitrary scaling σ_{ccd}. There are always $2M + 1$ "circle" points. Out of them, $2M$ are located along each axis at distance $\pm\sqrt{M}\,\sigma_{ccd}$ and one is located at the origin. Figure 4 illustrates the location of the points in a CCD design for $M = 2$. The number of design-points corresponding to the possible different dimensions M is

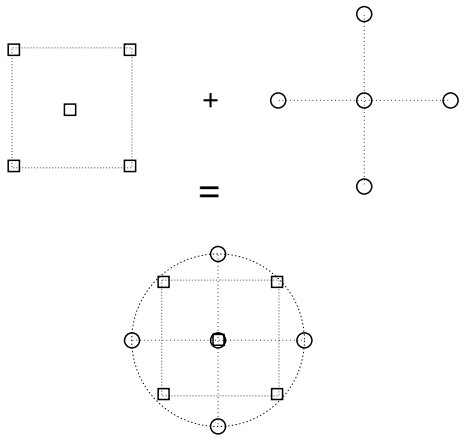


Figure 4: Location of points in a CCD design for $M = 2$. The squares are factorial points (design-points) and the circles are the additional "circle" points.

displayed in Table 1. In addition to those points, each design contains $2M + 1$ "circle"

points. Sanchez and Sanchez (2005) explain how to compute the locations of these points in the M dimensional space.

Dimension of $\boldsymbol{\theta}$	2	3	4-5	6	7-8	9-11	12-17
Number of points	4	8	16	32	64	128	256

Table 1: Number of design-points in a CCD.

The points are located using the $\boldsymbol{z}$ parametrisation defined in (18). Moreover, in order to capture some of the asymmetry possibly present in the domain of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ we allow the scaling parameter σ_{ccd} to vary, not only according to the M different axis but also according to the direction, positive or negative, of each axes. This means that for each design we have $2M$ scaling parameters, $(\sigma_{ccd}^{m+}, \sigma_{ccd}^{m-})$, $m = 1, \ldots, M$. To compute these, we first note that in a Gaussian density, the drop in log density when we move from the mode to ± 2 the standard deviation is -2. We compute our scaling parameters in such a way that this is approximately true for all direction in our design.

To compute the integral (4) we still have to determine the value of the area weights Δ_k. In fact here they cannot be considered all equal like in Section 4.1.1. To determine the weights we assume for simplicity that $\boldsymbol{\theta}|\boldsymbol{y}$ is standard Gaussian. We require the integral of 1 to be 1 and the integral of $\boldsymbol{\theta}^T\boldsymbol{\theta}$ to be M. This two conditions give the integration weights for the points on the sphere with radius $f_0\sqrt{M}$:

$$\Delta = \left[(n_p - 1)\left(f_0^2 - 1\right)\left\{1.0 + \exp\left(-\frac{M f_0^2}{2}\right)\right\}\right]^{-1}$$

where $f_0 > 1$ is any constant. The integration weight for the central point is $1-(n_p-1)\Delta$ where n_p is the total number of points in the design.

The CCD strategy reduces the accuracy of the numerical integral and, for small dimensions of the hyperparameter space the grid strategy is clearly preferable. Anyway, it often happens that when there are many hyperparameters, the shape of the integrand is more regular and therefore simpler. This means that with increasing dimension of $\boldsymbol{\theta}$, the number of evaluations points does not, necessarily, have to increase exponentially to obtain a sufficient accuracy of the integral. Strategies like the 'plug-in' approach brings this idea to extreme by using only the modal value to integrate over $\pi(\boldsymbol{\theta}|\boldsymbol{y})$. The 'plug-in' solution will probably underestimate the variance, but in many cases, still gives useful results. The CCD integration strategy lies somewhere in between the accurate, but expensive, grid strategy and the fast, but possibly imprecise, 'plug-in' strategy. It allows to capture some of the variability in the hyperparameter space also when this is too wide to be explored via the grid strategy.

4.2 Approximating $\pi(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})$

The next task is to build an approximation to the density of $x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y}$. It is clear that the quality of this approximation reflects into the quality of $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ whatever the integra-

tion strategy. We propose here two different approximations: a Gaussian approximation and an improved approximation. Computing the Gaussian approximation, $\widetilde{\pi}_G(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})$, implies almost no extra costs after we have computed $\widetilde{\pi}(\boldsymbol{\theta}_k)$. It is, hence, an extremely fast alternative. It can, however, present some errors due to the lack of skewness. The Gaussian approximation is described in Section 4.2.1. A more accurate alternative is presented in Section 4.2.2. This is a non-parametric approximation and, therefore, it can better capture the shape of the density of $x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y}$. This improved approximation is more computationally demanding. The improved approximation is valuable because it is more accurate, but also because it can serve as a validation for the Gaussian approximation. In fact, if it is indistinguishable or very close the the Gaussian approximation, the latter is checked and confirmed without Monte Carlo sampling. A different strategy for assessing the approximation error based on the effective number of parameters in the model is presented in Rue et al. (2007).

4.2.1 Gaussian approximation

The easiest way to approximate $\pi(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})$ is to use the marginal derived from $\pi_G(\boldsymbol{x}|\boldsymbol{\theta}_k, \boldsymbol{y})$ (Section 3.1). When selecting the points $\boldsymbol{\theta}_k$ and computing $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$ we have already compute $\pi_G(\boldsymbol{x}|\boldsymbol{\theta}_k, \boldsymbol{y})$, therefore we know the mean vector, and the only element which remains to be computed is the vector of marginal variances. This, as mentioned in Section 3 can be done efficiently thanks to the recursions described in Rue and Martino (2006). Also, it makes practically no difference in terms of time, to compute one or all N marginal densities in the GMRF. The approximation is then

$$\widetilde{\pi}_G(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y}) = \mathcal{N}(x_{ti}; \mu_{G_{ti}}(\boldsymbol{\theta}_k), \sigma^2_{G_{ti}}(\boldsymbol{\theta}_k)) \tag{20}$$

where $\sigma_G(\boldsymbol{\theta}_k)$ is the N-dimensional vector of marginal variances.

Rue and Martino (2006) show that the approximation in (20) gives often accurate results, but, especially for values of $\boldsymbol{\theta}_k$ located in extreme regions, there might be slight errors in the location and skewness. These errors are detected by comparing the approximations with density estimates derived from very long MCMC runs. Since these errors appear mainly in regions with low density for $\boldsymbol{\theta}|\boldsymbol{y}$, they become much smaller after integrating out $\boldsymbol{\theta}$. In fact, even if $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ is, in this case, a mixture of Gaussian it can represent precisely also highly skewed densities. Errors using the Gaussian approximation might, anyway, still be detectable in $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$, see Rue and Martino (2006).

4.2.2 Improved approximation

The errors in the Gaussian approximation in Section 4.2.1 are due to the fact that we approximate a (possibly) skewed distribution with a symmetric one. It is natural then, to think of an improved approximation which allows for skewness to be present. The improved approximation described in this section follows the lines of the Simplified Laplace approximation proposed in Rue et al. (2007), with some modifications necessary to adapt

it to the problems described in this report. The improved approximation assumes no parametric form of the density $x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y}$, therefore it is able to capture skewness if present.

The starting point is the identity

$$\pi(x_{ti}|\boldsymbol{\theta}, \boldsymbol{y}) = \frac{\pi(\boldsymbol{x}_{-ti}, x_{ti}|\boldsymbol{\theta}, \boldsymbol{y})}{\pi(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}, \boldsymbol{y})} \propto \frac{\pi(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})}{\pi(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}, \boldsymbol{y})} \tag{21}$$

Where the suffix $-ti$ indicates that the element ti in the vector has been removed. The idea, similar to the one used in Section 3.2 to build $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$, is to substitute the density in the denominator of the rightmost element in equation (21) with a Gaussian approximation. The approximation then reads:

$$\widetilde{\pi}_I(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y}) \propto \left.\frac{\pi(\boldsymbol{x}, \boldsymbol{\theta}_k, \boldsymbol{y})}{\widetilde{\pi}_{GG}(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})}\right|_{\boldsymbol{x}_{-ti}=\boldsymbol{x}^\star_{-ti}(x_{ti}, \boldsymbol{\theta}_k)} \tag{22}$$

where $\boldsymbol{x}^\star_{-ti}(x_{ti}, \boldsymbol{\theta}_k)$ is the mode of $\pi(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k)$. This again is equivalent to the Laplace approximation in Tierney and Kadane (1986).

It has to be noted that the density $\widetilde{\pi}_{GG}(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$, in the denominator of (22), is different from the conditional distribution, $\widetilde{\pi}_G(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$, which can be derived from the Gaussian approximation in (3.1). In fact, $\widetilde{\pi}_G(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$ is computed through a rank 1 update from $\widetilde{\pi}_G(\boldsymbol{x}|\boldsymbol{\theta}_k, \boldsymbol{y})$. Its precision matrix is constant with respect to x_{ti} and its mean is a linear function of x_{ti}. On the other side, $\widetilde{\pi}_{GG}(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$ is computed by first locating the mode $\boldsymbol{x}^\star_{-ti}(x_{ti}, \boldsymbol{\theta}_k)$ of $\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y}$ and then expanding the log-likelihood term around it, in much the same way as in Algorithm 1. The precision matrix in $\widetilde{\pi}_{GG}(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$ varies with x_{ti}. The density in (22) is based on conditioning on x_{ti} and using Laplace approximation to cancel out the remaining variables $\boldsymbol{x}_{-ti}$. Hence, it is more accurate than the approximation in (20) which is based on fitting a Gaussian as the joint distribution of all variables $\boldsymbol{x}$.

Unfortunately, having to locate the mode of $\pi(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$ means that, for each value of x_{ti}, we have to factorise a $(N-1) \times (N-1)$ matrix more than once (see Algorithm 1). Moreover, there are, potentially, N posterior densities for the latent field to be computed. It is clear, then, that the approximation in (22) is far too computationally expensive to be convenient. Hence, we need to slightly modify (22) to make it computationally feasible.

The conditional mean $\mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$ from $\widetilde{\pi}_G(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$, and the conditional mode of $\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y}$ would be coincident if $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}_k)$ was Gaussian. This is of course not the case here, since the log likelihood presents non quadratic terms. Anyway $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ is not too far from a Gaussian, having $\boldsymbol{x}|\boldsymbol{\theta}$ a Gaussian prior. Moreover (22) is valid for any value of $\boldsymbol{x}_{-ti}$ and, though in a different context, Hsiao et al. (2004) show that consideration for efficiency suggest that the value of $\boldsymbol{x}_{-ti}$ should be chosen in an area of high density of $\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y}$ but not necessarily at the modal value. We propose therefore to compute the quantity in (22) at the conditional mean instead of the conditional mode. This entails large computational benefits. First of all we avoid the optimisation step: the conditional mean can easily be computed for each ti, using (10) where $\boldsymbol{x} = \boldsymbol{\mu}_G$ and $\boldsymbol{A} = \mathbf{1}_{ti}$, a vector of zeros with 1 in position ti, and $\boldsymbol{e}$ is the value of

x_{ti}. Moreover, this computation needs to be done only once for each ti, at $x_{ti} = \mu_{G_{ti}} + 1$, say. Exploiting the linearity of the conditional mean with respect to x_{ti}, we can, in fact, evaluate its numerical derivative as:

$$\boldsymbol{\delta}_{\mathrm{E}}^{ti} = \mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-ti}|x_{ti} = \mu_{G_{ti}} + 1, \boldsymbol{\theta}_k, \boldsymbol{y}) - \boldsymbol{\mu}_{G_{-ti}}$$

and, obtain its value at any x_{ti} as:

$$\mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-ti}|x_{ti} = x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y}) = \boldsymbol{\mu}_{G_{-ti}} + \boldsymbol{\delta}^{ti}(x_{ti} - \mu_{G_{ti}})$$

There is also another advantage in considering the conditional mean instead of the conditional mode: the conditional mode $\boldsymbol{x}^{\star}_{-ti}(x_{ti}, \boldsymbol{\theta}_k)$ is a continuous function of x_{ti}, but, since we compute it via numerical optimisation, this continuity might not hold in practice. The conditional mean, on the other side, is always a continuous function of x_{ti}.

Even if using the conditional mean avoids the optimisation step, the approximation in (22) is still too heavy to be computed efficiently. The log denominator of (22) is in fact:

$$\begin{gathered} \log \widetilde{\pi}_{GG}(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{y}, \boldsymbol{\theta}_k)\Big|_{\boldsymbol{x}_{-ti}=\mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-ti}|x_{ti},\boldsymbol{\theta}_k,\boldsymbol{y})} \propto \\ \tfrac{1}{2}\log|\boldsymbol{Q}_{[-ti,-ti]} + \mathrm{diag}(\boldsymbol{C}(x_{ti}, \boldsymbol{\theta}_k))| = f(x_{ti}) \end{gathered} \tag{23}$$

where $\boldsymbol{Q}$ is the prior precision matrix for $\boldsymbol{x}$ and the subscript $[-ti, -ti]$ indicates that row and column corresponding to index ti have been deleted. The matrix $\mathrm{diag}(\boldsymbol{C}(x_{ti}, \boldsymbol{\theta}_k))$ is the band matrix derived from the Taylor expansion of the log-likelihood at the conditional mean $\mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{\theta}_k, \boldsymbol{y})$ in much the same way as in Section 3.1. Computing the determinant in (23) means factorising a $(N-1) \times (N-1)$ matrix, and this has to be done for each value of x_{ti}.

In Rue et al. (2007), the authors propose to approximate (23) by a first order series expansion around $x_{ti} = \mu_{G_{ti}}(\boldsymbol{\theta}_k)$. For the cases analysed in Rue et al. (2007) the matrix $\mathrm{diag}(\boldsymbol{C})$ defined in (14) is a diagonal matrix, it is then possible to derive the exact expression for the first derivative of $f(x_{ti})$, see Appendix for details. The same is not possible for MSV models like those we are interested in this report. We can, anyway, compute the numerical derivative of the quantity in (23)

$$\delta_f^{ti} = \frac{f(x_{ti} + h) - f(x_{ti})}{h}$$

Moreover,at $x_{ti} = \mu_{G_{ti}}$ the log determinant of $(\boldsymbol{Q}_{[-ti,-ti]} + \mathrm{diag}[\boldsymbol{C}(\mu_{ti}, \boldsymbol{\theta}_k)])$ can be computed at almost no extra costs as

$$f(\mu_{G_{ti}}) = \frac{1}{2}\log|\boldsymbol{Q}_{[-ti,-ti]} + \mathrm{diag}[\boldsymbol{C}(\mu_{G_{ti}}, \boldsymbol{\theta}_k)]| = \frac{1}{2}\log|\boldsymbol{Q}_G| + \log\sigma_{G_{ti}} \tag{24}$$

See Appendix for detail about how to derive (24). All elements at the right hand side of equation (24) have already been computed while computing $\widetilde{\pi}_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}_k)$ and $\widetilde{\pi}_G(x_t|\boldsymbol{y}, \boldsymbol{\theta}_k)$. Using a linear approximation for the log denominator of equation (22) makes it necessary

to factorise a $(N-1)\times(N-1)$ matrix only once for each of the N nodes in the latent field.

The quantity in (22), modified as described above, has to be computed for different values of x_{ti} and then normalised in order to obtain a density. We select these points with the help of the mean and variance of the Gaussian approximation (20), by choosing different values for the standardised variable

$$x_{ti}^s = \frac{x_{ti} - \mu_{G_{ti}}(\boldsymbol{\theta}_k)}{\sigma_{G_{ti}}(\boldsymbol{\theta}_k)}$$

according to the corresponding choice of abscissas given by the Gauss-Hermite quadrature rule. To represent the density $\widetilde{\pi}_I(x_{ti}|\boldsymbol{\theta}_k,\boldsymbol{y})$ we use

$$\widetilde{\pi}_I(x_{ti}|\boldsymbol{\theta}_k,\boldsymbol{y}) \propto \mathcal{N}\{x_{ti};\mu_{G_{ti}}(\boldsymbol{\theta}_k),\sigma_{G_{ti}}(\boldsymbol{\theta}_k)\}\times\exp\{\text{cubic spline}(x_{ti})\}$$

The cubic spline is fitted to the difference $\log\widetilde{\pi}_I(x_{ti}|\boldsymbol{\theta}_k,\boldsymbol{y}) - \log\widetilde{\pi}_G(x_{ti}|\boldsymbol{\theta}_k,\boldsymbol{y})$ at the selected abscissa points. The density is then normalised using quadrature integration.

5 Approximating marginal likelihood $\pi(\boldsymbol{y})$

Model comparison is an important part of any statistical analysis and a central pursuit of science in general. In a Bayesian framework, one way to compare models is to use Bayes factors. Given a series of competing models $\mathcal{M}_1,\ldots,\mathcal{M}_K$ with assigned a prior probability $\pi(\mathcal{M}_k)$ the Bayes factor for two of the K models is defined as

$$\mathcal{B}(i,j) = \frac{\pi(\mathcal{M}_i|\boldsymbol{y})\pi(\mathcal{M}_i)}{\pi(\mathcal{M}_j|\boldsymbol{y})\pi(\mathcal{M}_j)}$$

If we choose the models to be apriori equiprobable,$\pi(\mathcal{M}_1)=\cdots=\pi(\mathcal{M}_K)$, then the Bayes factor reduces to

$$\mathcal{B}(i,j) = \frac{\pi(\boldsymbol{y}|\mathcal{M}_i)}{\pi(\boldsymbol{y}|\mathcal{M}_j)}$$

Hence, we can compare models by comparing their marginal likelihood $\pi(\boldsymbol{y}|\mathcal{M}_k)$. Jeffreys (1961) provide a scale for the interpretation of $\mathcal{B}(i,j)$ which we report in Table 2. In the following, to simplify the notation, we suppress the conditioning on $\mathcal{M}_k$ if it is not strictly necessary. In the INLA framework an approximation to the marginal likelihood $\pi(\boldsymbol{y})$ can be computed as the normalising constant for $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$

$$\widetilde{\pi}(\boldsymbol{y}) = \int \left.\frac{\pi(\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{y})}{\widetilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})}\right|_{\boldsymbol{x}=\boldsymbol{x}^\star(\boldsymbol{\theta})} d\boldsymbol{\theta}$$

where $\pi(\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{y}) = \pi(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$. We propose two approximations to $\pi(\boldsymbol{y})$. The first one is based on a Gaussian approximation of the density of $\boldsymbol{\theta}|\boldsymbol{y}$ built by matching the mode and the curvature at the mode, that is

$$\widetilde{\pi}_G(\boldsymbol{\theta}|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{\theta}^*,\boldsymbol{\Sigma}) \tag{25}$$

$\log \mathcal{B}(i,j)$	Strength of the evidence in favour if $\mathcal{M}_i$
< 0	Negative (support for $\mathcal{M}_j$)
$0 : 1.09$	Barely worth mentioning
$1.09 : 2.30$	Substantial
$2.30 : 3.40$	Strong
$3.40 : 4.60$	Very strong
> 4.60	Decisive

Table 2: Jeffreys (1961)'s scale for the interpretation of the Bayes factor

where $\boldsymbol{\theta}^*$ is the mode and $\boldsymbol{\Sigma} = \boldsymbol{H}^{-1}$ is the inverse of the negative Hessian matrix computed at the modal configuration. The normalising constant, and so our approximation for the marginal likelihood, is then given by

$$\widetilde{\pi}_1(\boldsymbol{y}) = (2\pi)^{M/2}|\boldsymbol{H}|^{-1/2} \tag{26}$$

where M is the dimension of $\boldsymbol{\theta}$. This approximation was proposed also by Kass and Vaidyatnatan (1992).

The second approximation is more precise but also more expensive to compute. It assumes no parametric form of the density of $\boldsymbol{\theta}|\boldsymbol{y}$ and uses the same integration scheme as in Section 4.1.1 to compute the normalising constant. The approximation then reads

$$\widetilde{\pi}_2(\boldsymbol{y}) = \sum_k \widetilde{\pi}_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}_k)\Delta_k \tag{27}$$

This second approximation, allows to take into account departures from Gaussianity which are often encountered in $\pi(\boldsymbol{\theta}|\boldsymbol{y})$, and therefore gives more accurate results. Unluckily, as already explained in Section 4.1.1, this integration scheme becomes unfeasible when the dimension of $\boldsymbol{\theta}$ grows. Anyway, as shown in the examples, there seems not to be a big difference in the model ranking obtained from the two approximations.

Note that, when computing an approximation to the marginal likelihood $\pi(\boldsymbol{y})$, aiming to use it for model comparison, it is important to include carefully all normalising constants which appear in the prior for both the hyperparameters $\pi(\boldsymbol{\theta})$ and the latent field $\pi(\boldsymbol{x}|\boldsymbol{\theta})$, and in the likelihood term $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$.

6 Examples of approximate inference for the latent field

In this section we apply INLA to estimate the univariate models in Section 2.1 and the five bivariate models in Section 2.2. To assess the quality of the approximations, we compare them with density estimates obtained from intensive MCMC runs.

Yu and Mayer (2006) propose to use the software package WinBUGS to implement a MCMC algorithm for univariate and multivariate SV models. WinBUGS is an interac-

tive Windows version of the BUGS program for Bayesian analysis of complex statistical problems using MCMC techniques, see Spiegelhalter et al. (2003). The BUGS (and WinBUGS) program provides an implementation of the Gibbs sampling algorithm, a specific MCMC techniques that builds a Markov chain by sampling from all univariate full conditional distributions in a cyclic way. WinBUGS uses a single site update scheme, therefore long runs are necessary since the mixing might be poor due to the correlations within the latent field $\boldsymbol{x}$ and between $\boldsymbol{x}$ and $\boldsymbol{\theta}$. Anyway, since we want to compare our approximation with the "true" posterior densities, we have run the MCMC algorithm for much longer time than it is usually done for inference purposes. The reader is referred to Mayer and Yu (2000) for a comprehensive introduction on using BUGS for fitting SV models.

6.1 Implementation Issues

Running the INLA procedures described in Section 4 so that they are optimised in term of computational time requires a very carefully implementation in an appropriate language. Much speed can be gained from writing the code in a carefully and smart way, for example by appropriately storing computations and using efficient routines for sparse matrix computation. Many of the algorithms described are efficiently implemented in the open-source library `GMRFLib`. This library is written in C, and in addition to the INLA routines, contains also several other routines for GMRF models. It is freely available at the web page `http://www.math.ntnu.no/~hrue/GMRFLib/` and a brief introduction to it can be found in Rue and Held (2005). Rue and Martino (2006) and Rue et al. (2007) make an intensive use of the `GMRFLib`-library in the examples they present.

Unfortunately the `GMRFLib`-library does not support multivariate models like those described in Section 2.2. It was therefore necessary, for the multivariate examples in this report, to rewrite almost every algorithm necessary for the implementation of INLA. For this purpose, we used the statistical package R (Ihaka and Gentleman, 1996). The R language is less fit than C for the purpose, moreover, the code used for the examples in this report, is far from being optimal with respect to computational efficiency and time. Hence, the examples reported here have to be considered as a proof of concept showing another application of approximate inference using INLA. The reader is referred to Rue et al. (2007) for examples showing the gain, in terms of computing time, which can be achieved using the INLA over MCMC.

6.2 Univariate Models

In this section we fit two univariate SV models, first to a simulated data set, and then to the pound-dollar exchange rate data displayed in Figure 1.

Both models are define as in equations (6). In the first model ($\mathcal{M}_1$) we define $\epsilon_t \sim \mathcal{N}(0,1)$, while in the second model ($\mathcal{M}_2$) $\epsilon_t \sim t_\nu$. For each of the two data set, we fit $\mathcal{M}_1$ and $\mathcal{M}_2$ and check the quality of the INLA approach. Then, we compare the two models using the approximated marginal likelihood $\widetilde{\pi}(\boldsymbol{y})$.

6.2.1 Simulated data set

We simulate 500 data from the following model

$$\begin{aligned} y_t &= \exp(h_t/2)\epsilon_t, \quad t = 1, \ldots, n, \quad \epsilon_t \sim t_7. \\ h_t &= 0.1 + 0.53(h_{t-1} - 0.1) + \eta_t, \quad t = 1, \ldots, n, \quad \eta_t \sim \mathcal{N}(0, 1/2.3). \end{aligned} \tag{28}$$

The simulated time series is displayed in Figure 5. Note that the Student-t distribution allows for quite extreme values of the returns.

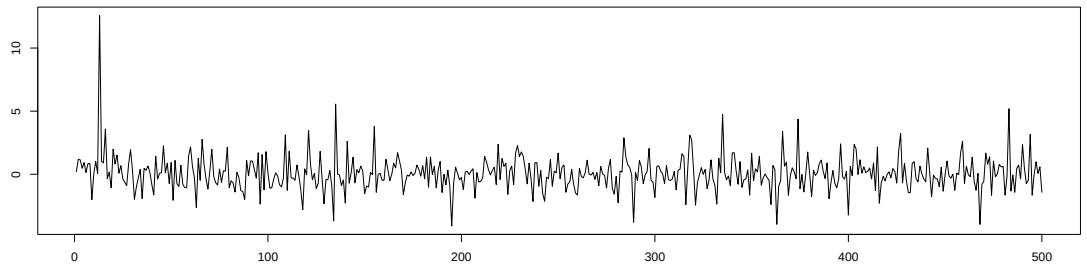
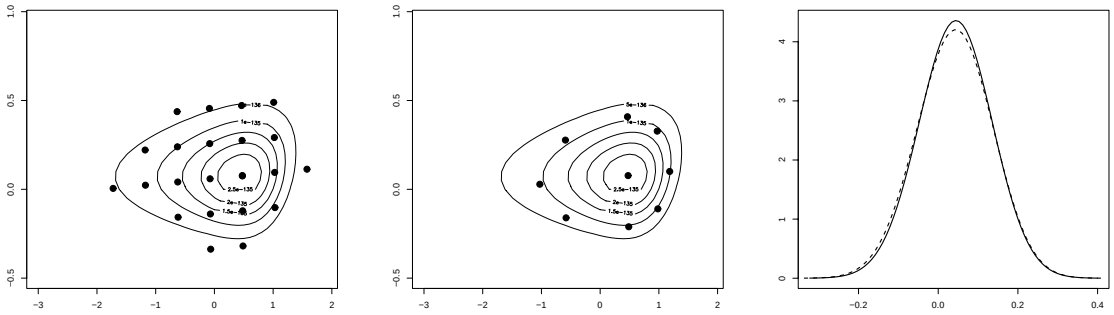


Figure 5: Time series of returns simulated from model (28)

We first fit $\mathcal{M}_1$ to the simulated data. Following Algorithm 2, our first task is to locate a set of points in the hyperparameters space, $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$, where to compute $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$ and $\widetilde{\pi}(x_t|\boldsymbol{\theta}_k, \boldsymbol{y})$. We do this using both the grid and the CCD strategies. In the first case, the number K of points to be computed is 22, while in the second case it reduces to 9. For really low dimension of the hyperparameters space (as in this example) there is no big computational difference in using one integration scheme or the other. Figure 6, panels (a) and (b), show a contour plot of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. Superimposed are the locations of the integration points when using a grid strategy, panel (a), and a CCD strategy, panel (b). Figure 6(c) displays the results of the two integrations strategies when computing the posterior marginal $\widetilde{\pi}(x_t|\boldsymbol{y})$. The density displayed is chosen to be the one for which the two integration schemes gave the most different results. The difference between densities is computed via a (symmetric) Kullback-Leibler measure. Even though the grid strategy uses more points than the CCD strategy, and even thought the density of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is quite far from a Gaussian, the difference in the results of the two integrations is almost unnoticeable.

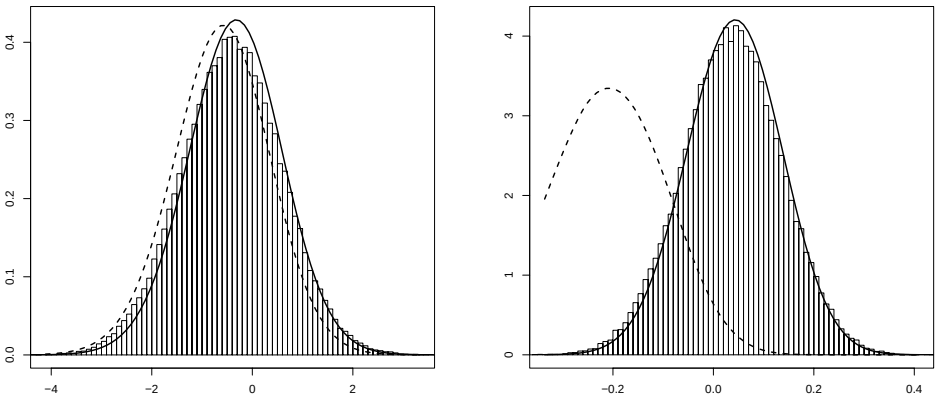
We compare, the approximations for $\pi(x_t|\boldsymbol{y})$ obtained using the Gaussian approximation and the improved one, in Sections 4.2.1 and 4.2.2 respectively, to represent $\pi(x_t|\boldsymbol{\theta}_k, \boldsymbol{y})$. Figure 7, panels (a) and (b), show the two approximations for one of the nodes h_t in the time series, and for the common mean μ respectively. The node h_t showed was chosen to be the one for which the Gaussian and the improved approximation gave the most different result. In the same figures is also displayed an histogram obtained from an intensive MCMC run of model $\mathcal{M}_1$ using WinBUGS. After a burn-in period, we have collected a MCMC samples of 10^6 by keeping every 20th simulated value in the chain. The Gaussian



(a) Grid strategy (b) CCD strategy (c) integration results

Figure 6: $\mathcal{M}_1$, simulated data example. Configurations $\boldsymbol{\theta}_k$ used in the grid strategy (a) and in the CCD strategy (b). In panel (c) is the result of the integration procedure using the grid (solid line) and the CCD strategy (broken line)

approximation appears to be shifted, especially when considering the density of $\pi(\mu|\boldsymbol{y})$. The improved approximation, on the other hand, gives quite an accurate result.



$\pi(h_t|\boldsymbol{y})$ $\pi(\mu|\boldsymbol{y})$

Figure 7: $\mathcal{M}_1$, simulated data example: Gaussian approximation (broken line), improved approximation (solid line) and MCMC density estimate (histogram).

We then fit model $\mathcal{M}_2$ to the same simulated data. In this case the hyperparameters space has dimension 3. The grid integration scheme requires 70 points while the CCD integration scheme only 15. Figure 8 shows the results of the two integration procedures for one of the nodes in the latent field $(\boldsymbol{h}, \mu)$. Also in this case, the CCD integration scheme allows for a quite big computational gain without loosing in accuracy.

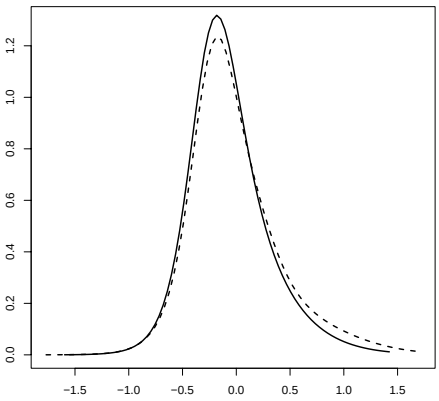
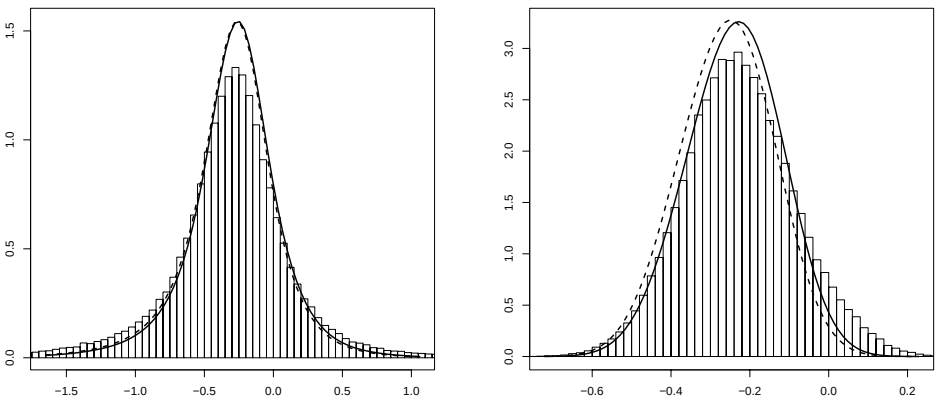


Figure 8: $\mathcal{M}_2$, simulated data example: approximation of $\pi(x_t|\boldsymbol{y})$ computed via the grid integration strategy (solid line) and the CCD integration strategy (broken line).

In Figure 9 the Gaussian and improved approximations for two nodes h_t and μ, are displayed and compared with an histogram derived from a long MCMC sample obtained as before. Notice that there are differences between the MCMC based estimate and the



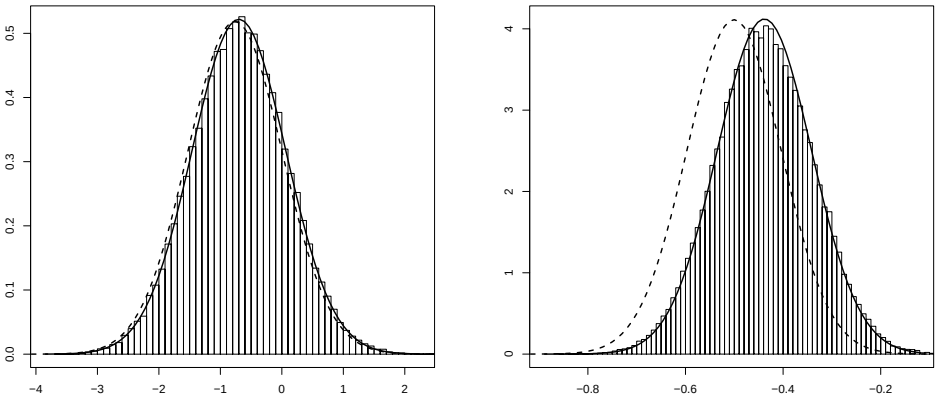
(a) $\pi(h_t|\boldsymbol{y})$ (b) $\pi(\mu|\boldsymbol{y})$

Figure 9: $\mathcal{M}_2$, simulated data example: Gaussian approximation (broken line), improved approximation (solid line) and MCMC density estimate (histogram).

improved approximation especially in the right tail of the density for the common mean

$\pi(\mu|\boldsymbol{y})$ (Figure 9b). We believe that these differences are mostly due to MCMC error, which despite the long run, is still present in the sample. WinBUGS uses a single site algorithm which can be extremely slow and "sticky" especially with heavy tailed data and strongly correlated variables in the latent field.

To reinforce our believes we made two experiments. First, we have fixed the value of the hyperparameter vector $\boldsymbol{\theta}$ to an arbitrary value. This makes the MCMC run faster. Moreover, quality of the INLA approximation for $\pi(x_t|\boldsymbol{y})$ depends directly on the quality of the approximation for $\pi(x_t|\boldsymbol{y}, \boldsymbol{\theta})$. Figure 10 shows results for the same two nodes displays in Figure 9. The hyperparameters value is $\log \kappa = 2$, $\log \tau = 0$ and $\delta = 1$. These values are chosen in a quite extreme region of the posterior density $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ because in our experience (Rue and Martino, 2006), it is in such areas that the approximation problem is more difficult. The Gaussian approximation appears to be slightly shifted with respect to the MCMC estimate while the improved approximation gives an accurate result. The experiment was repeated for different values of the hyperparameters always with the same result.

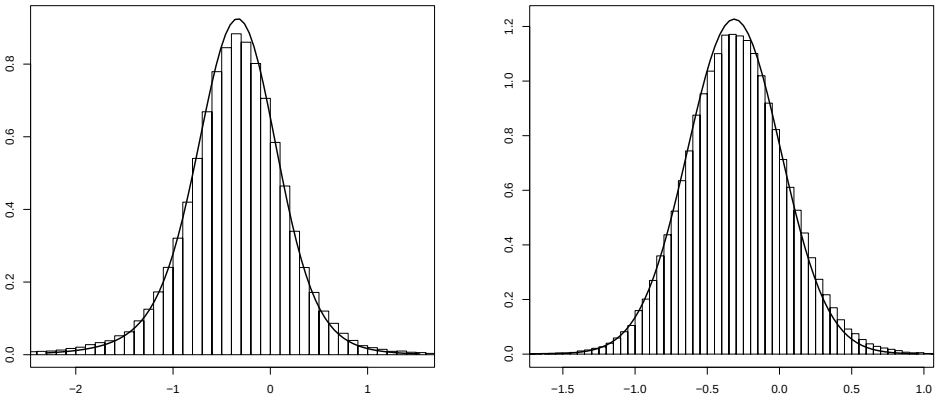


(a) $\pi(h_t|\boldsymbol{y})$ (b) $\pi(\mu|\boldsymbol{y})$

Figure 10: Simulated data, $\mathcal{M}_2$ model with fixed hyperparameters: Gaussian approximation (broken line), improved approximation (solid line) and MCMC density estimate (histogram).

In our second experiment the hyperparameter vector $\boldsymbol{\theta}$ is random but only the first 50 data of the simulated time series are considered. Decreasing the number of data makes the MCMC algorithm run much faster and mix better. On the other side, the approximation problems are easier when the number of data increases, see Rue et al. (2007) for considerations about the asymptotic behaviour of INLA. Figure 11 shows the improved approximation and the MCMC density estimate for the same nodes in Figure 9 when only 50 data are considered. Here the approximations and the MCMC estimates agree almost perfectly.

Based on these results, we believe that, if we run the MCMC algorithm for the full data set for much longer time, the histograms in Figure 9 would finally overlap with the improved



(a) $\pi(h_t|\boldsymbol{y})$ (b) $\pi(\mu|\boldsymbol{y})$

Figure 11: Simulated data, model $\mathcal{M}_2$ considering only 50 data: Gaussian approximation (broken line), improved approximation (solid line) and MCMC density estimate (histogram).

approximations.

To conclude, we compare $\mathcal{M}_1$ and $\mathcal{M}_2$, using the approximated marginal likelihood $\widetilde{\pi}(\boldsymbol{y}|\mathcal{M}_k)$. We compute two approximation for $\widetilde{\pi}(\boldsymbol{y}|\mathcal{M}_k)$ using both the Gaussian approximation for $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ in (26) and numerical integration in (27). Table 3 presents the logarithm of $\widetilde{\pi}(\boldsymbol{y}|\mathcal{M}_k)$. The marginal likelihood is largest for model $\mathcal{M}_2$, which corre-

	$\mathcal{M}_1$: Gaus. returns model	$\mathcal{M}_2$: Stud. return model
$\log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k)$	-209.1083	-206.1067
$\log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k)$	-208.8983	-206.3458

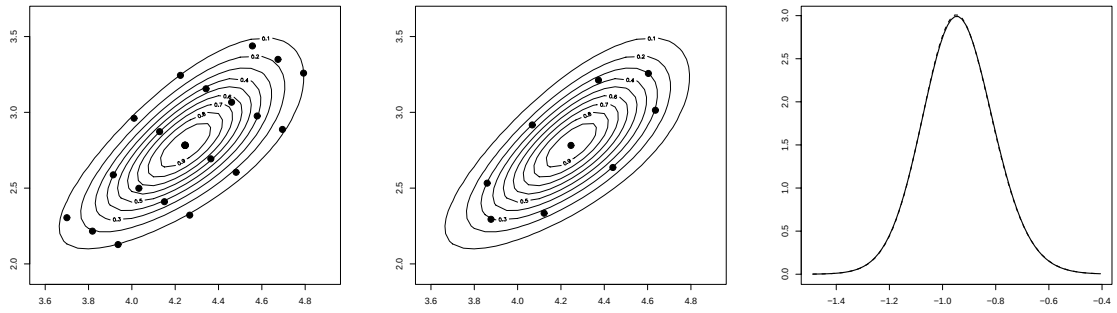
Table 3: Simulated data example: estimated value of the marginal likelihood $\log \pi(\boldsymbol{y}|\mathcal{M}_k)$ for $i = 1, 2$ computed via a Gaussian approximation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and via numerical integration.

sponds to the true model in (28). The difference in the logarithm of the marginal likelihood between the two models is 3 if we consider the Gaussian approximation to $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ in (27) and 2.4 if we compute $\widetilde{\pi}(\boldsymbol{y}|\mathcal{M}_k)$ numerically. This shows evidence that tails heavier than those of a Gaussian distribution are needed to describe the returns process in this example.

6.2.2 Pound-dollar exchange rate data set

Our second example for the univariate SV model consists in the Pound-dollar exchange rates plotted in Figure 1 .The same data set was analysed, among others, by Durbin and Koopman (1997) and Rue et al. (2007).

Consider model $\mathcal{M}_1$ first. For the grid integration scheme 29 points are evaluated, while the CCD strategy evaluates 9. Figure 12, shows contour plots of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. and locations of the integration points when using a grid strategy, panel (a), and a CCD strategy, panel (b). Figure 12(c) displays the results of the two integrations when computing the posterior marginal $\widetilde{\pi}(x_t|\boldsymbol{y})$. This time the difference between the two densities is almost undetectable. This is due to the fact that,compared to that in the previous example, the density of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is more regular. Here by "regular" we mean no too far from a Gaussian.



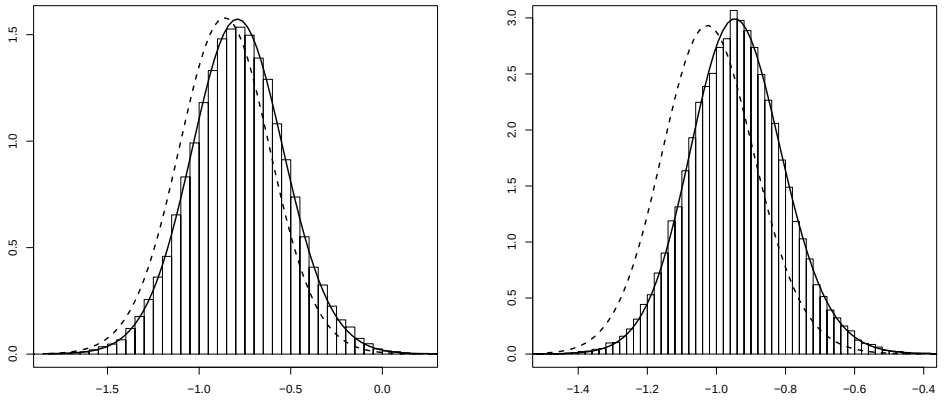
(a) Grid strategy (b) CCD strategy (c) integration results

Figure 12: $\mathcal{M}_1$, real data example: integration points needed to compute $\widetilde{\pi}(x_t|\boldsymbol{y})$. Panel (a) illustrates the grid strategy and panel (b) the CCD strategy. In panel (c) is the result of integration procedure using the grid (solid line) and the CCD strategy (broken line)

We proceed then to check the accuracy of the approximations for $\pi(x_t|\boldsymbol{y})$. Figure 13, panels (a) and (b), show the two approximations for one of the nodes h_t in the time series, and for the common mean μ. The node h_t showed was chosen to be the one for which the Gaussian and the improved approximation gave the most different result. In the same Figure is also an histogram obtained from a long (around 10^6 iterations) MCMC run which represents the "true" density. Again, the Gaussian approximation appears to be shifted, especially when considering the approximation for $\pi(\mu|\boldsymbol{y})$ while the improved approximation is practically perfect.

When fitting $\mathcal{M}_2$, the grid integration scheme requires 73 points while the CCD integration scheme only 15. Figure 14 shows the results of the two integration procedures for one of the nodes in the latent field $(\boldsymbol{h}, \mu)$. The node is chosen to be the one for which two procedures gave the most different results.

In Figure 15 the Gaussian and improved approximation for one node in the time series and for the common mean μ are displayed together with density estimations from a very



(a) $\pi(h_t|\boldsymbol{y})$ (b) $\pi(\mu|\boldsymbol{y})$

Figure 13: $\mathcal{M}_1$, real data example: Gaussian approximation (broken line) improved approximation (solid line) and MCMC density estimate (histogram).

long MCMC run. Again we see that while the Gaussian approximation can present errors in location and skewness, the improved approximation gives very accurate results.

In order to compare $\mathcal{M}_1$ and $\mathcal{M}_2$, we compute the approximation for the marginal likelihoods using both a Gaussian approximation for $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and the numerical integration in (27). Table 4 presents the computed approximations for $\log \pi(\boldsymbol{y}|\mathcal{M}_k)$. The two approxi-

	$\mathcal{M}_1$: Gaus. returns models	$\mathcal{M}_2$:Stud. return model
$\log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k)$	-67.416	-69.150
$\log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k)$	-67.372	-68.949

Table 4: Real data example: estimated value of $\log \pi(\boldsymbol{y}|\mathcal{M}_k)$ for the two univariate models fitted to the pound-dollar exchange rate data. The estimated marginal likelihood is computed via a Gaussian approximation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and via numerical integration.

mations are very close to each other. The difference in log marginal likelihood, close to 1.7, offers a substantial evidence in favour of the Gaussian returns model. The idea that extra kurtosis in not needed for this data set is reinforced if we look at the mode of the posterior distribution $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ for the two models. The modal value of the parameter ν^* in the Student-t model is 3.760, this corresponds to a modal value for the degree of freedom of the Student-t distribution around 46. With such high degree of freedom, a Student-t distribution is practically indistinguishable from a Gaussian. Moreover the modes of the remaining two parameters, the range κ and the precision τ practically coincide in the two models, suggesting that a Gaussian distribution in the returns process well describes these data.

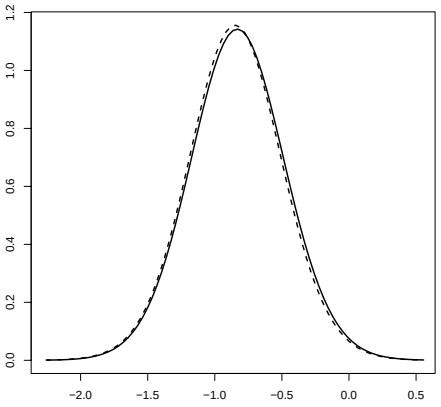
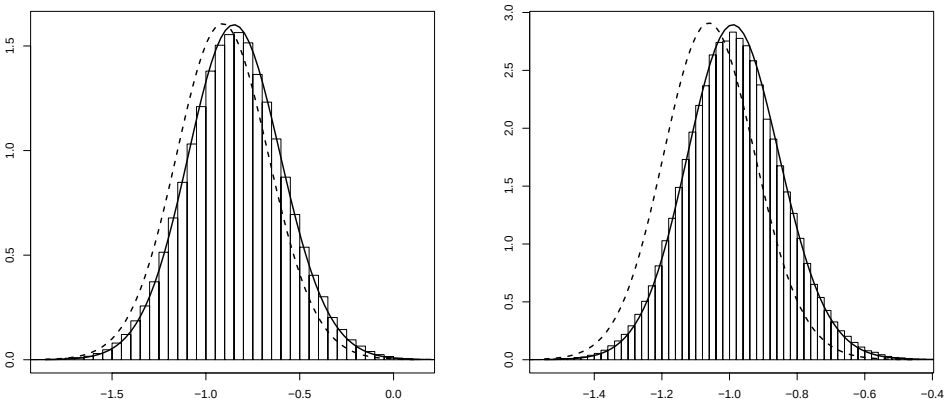


Figure 14: $\mathcal{M}_2$, real data example: approximation of $\pi(x_t|\boldsymbol{y})$ computed via the grid integration strategy (solid line) and the CCD integration strategy (broken line).



(a) $\pi(h_t|\boldsymbol{y})$ (b) $\pi(\mu|\boldsymbol{y})$

Figure 15: $\mathcal{M}_2$, real data example: Gaussian approximation (broken line) improved approximation (solid line) and MCMC density estimate (histogram).

6.3 Multivariate Models

In this section we fit the five bivariate models described in Section 2.2 to two financial time series.
The first data set consists in 300 data points simulated from Model 2 at page 8, with mean vector for the latent field $\boldsymbol{\mu} = (0.1, -0.2)$ and hyperparameters values: $\log \kappa_1 = 3$, $\log \kappa_2 = 5$, $\log \tau_1 = 2$, $\log \tau_2 = 4$, $\rho^*_\epsilon = 1$. The simulated data are plotted in Figure 16.
The second data set consists in 519 weekly mean corrected log-returns of the Australian

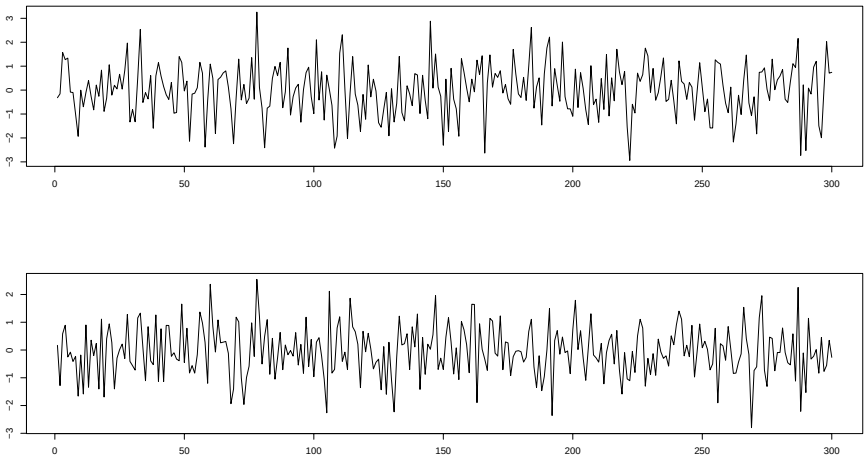


Figure 16: Simulated bivariate time series.

dollar and New Zeland dollar, both against the US dollar, from January 1994 to December 2003. The Australian and the New Zeland economies are closely related to each other, hence we expect the dependence between the two exchange rates to be strong. The two series are plotted in Figure 17 and indeed there appear to be strong cross-dependence both in returns and volatility. The same data set is analysed also in Yu and Mayer (2006). We analyse each of the five models separately and then compare them using the marginal likelihood $\widetilde{\pi}(\boldsymbol{y})$.

Computationally, the main difference between univariate and multivariate models is the increasing number of hyperparameters which makes all numerical integrations more intensive. Here the CCD integration strategy can really help reducing the computational burden. In Table 5 we have reported the number of evaluation points, for all five bivariate models fitted to both data set, necessary to compute the integral in (4) using the CCD and the grid strategy. The tuning parameters for the grid strategy are set to $\delta_z = 1$ and $\delta_\pi = 2.5$ in all cases. These default values have proved to be usually accurate enough (Rue et al., 2007). Notice that, when the dimension of the hyperparameters space increases, the CCD strategy can reduce the number of evaluation points by a factor of 20. To check the accuracy of the CCD integration strategy we compare, for each model, its result with the result obtained via the more computational intensive grid strategy.

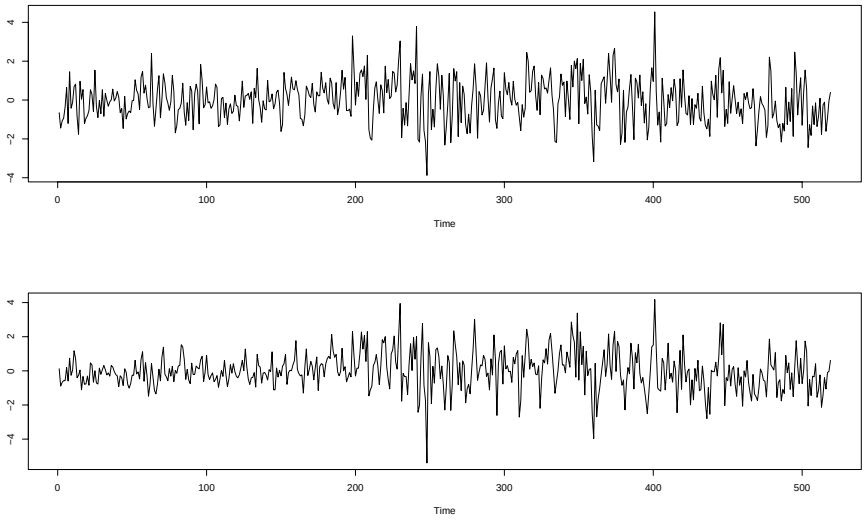


Figure 17: Time series for Australian/US Dollar (upper) and New Zeland/US Dollar (lower) exchange rate returns

	N. of hyperparam.	Simulated Data		Real Data	
		GRID	CCD	GRID	CCD
Model 1	4	124	25	101	25
Model 2	5	277	27	383	27
Model 3	6	774	45	882	45
Model 4	6	810	45	720	45
Model 5	6	619	45	688	45

Table 5: Number of integration points used to compute $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ using the two integration strategies.

MODEL	Variance Equation						Mean Equation	
	κ_1	κ_2	ϕ_{12}	ρ^*_η	$\log \tau_{\eta 1}$	$\log \tau_{\eta 2}$	ρ^*_ϵ	ν^*
Model 1	1.926	2.164	-	-	3.014	2.654	-	-
	(1.017)	(0.760)	-	-	(1.111)	(0.945)	-	-
Model 2	1.821	2.061	-	-	2.906	2.701	0.882	-
	(1.125)	(0.755)	-	-	(1.203)	(0.972)	(0.118)	-
Model 3	1.96	1.730	0.679	-	2.600	3.038	0.889	-
	(0.907)	(0.744)	(0.529)	-	(1.056)	(1.052)	(0.119)	-
Model 4	2.085	2.148	-	1.168	2.860	2.457	0.869	-
	(0.976)	(0.652)	-	(1.377)	(1.115)	(0.824)	(0.120)	-
Model 5	1.837	2.0258	-	-	3.220	2.923	0.886	3.092
	(1.073)	(0.810)	-	-	(1.065)	(1.003)	(0.121)	(0.882)

Table 6: Modal values of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ in the five bivariate models fitted to the simulated bivariate time series. In parentheses is the standard deviation as estimated from the inverse of the negative Hessian matrix computed at the mode.

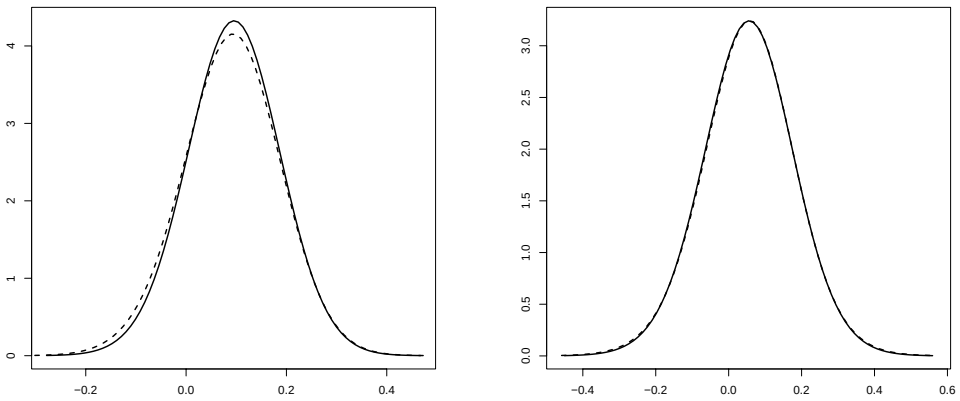
	Variance Equation						Mean Equation	
	$\log \kappa_1$	$\log \kappa_2$	ϕ_{12}	ρ_η^*	$\log \tau_{\eta 1}$	$\log \tau_{\eta 2}$	ρ_ϵ^*	ν^*
Model 1	3.998	4.174	-	-	3.188	2.700	-	-
	(0.333)	(0.351)	-	-	(0.449)	(0.505)	-	-
Model 2	3.391	3.588	-	-	3.792	2.803	1.993	-
	(0.566)	(0.631)	-	-	(0.538)	(0.731)	(0.097)	-
Model 3	3.902	1.750	0.828	-	3.916	2.260	1.940	-
	(0.374)	(0.576)	(0.393)	-	(0.485)	(0.648)	(0.098)	-
Model 4	3.360	2.960	-	2.610	3.264	1.805	1.945	-
	(0.377)	(0.4568)	-	(0.777)	(0.513)	(0.509)	(0.097)	-
Model 5	3.206	3.517	-	-	3.840	2.844	1.991	3.535
	(0.846)	(0.707)	-	-	(0.574)	(0.795)	(0.100)	(0.942)

Table 7: Modal values of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ in the five bivariate models fitted to the Australian/US and New Zeland/US exchange rates. In parentheses is the standard deviation as estimated from the inverse of the negative Hessian matrix computed at the mode.

6.3.1 Model 1 (Basic MSV)

Model 1 is equivalent to stacking together two independent univariate models with Gaussian noise in the returns equation. There is no correlation between volatilities nor between returns and no Granger causality is allowed. The hyperparameters are four and consist in the two log precisions and the two log ranges for the latent field. Table 6 refers to the simulated data set and reports the modal values of the hyperparameters and, in parentheses, the standard deviations as estimated from assuming a Gaussian approximation for $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ as in equation (25). Table 7 reports the same quantities for the Australian/New Zeland data set.

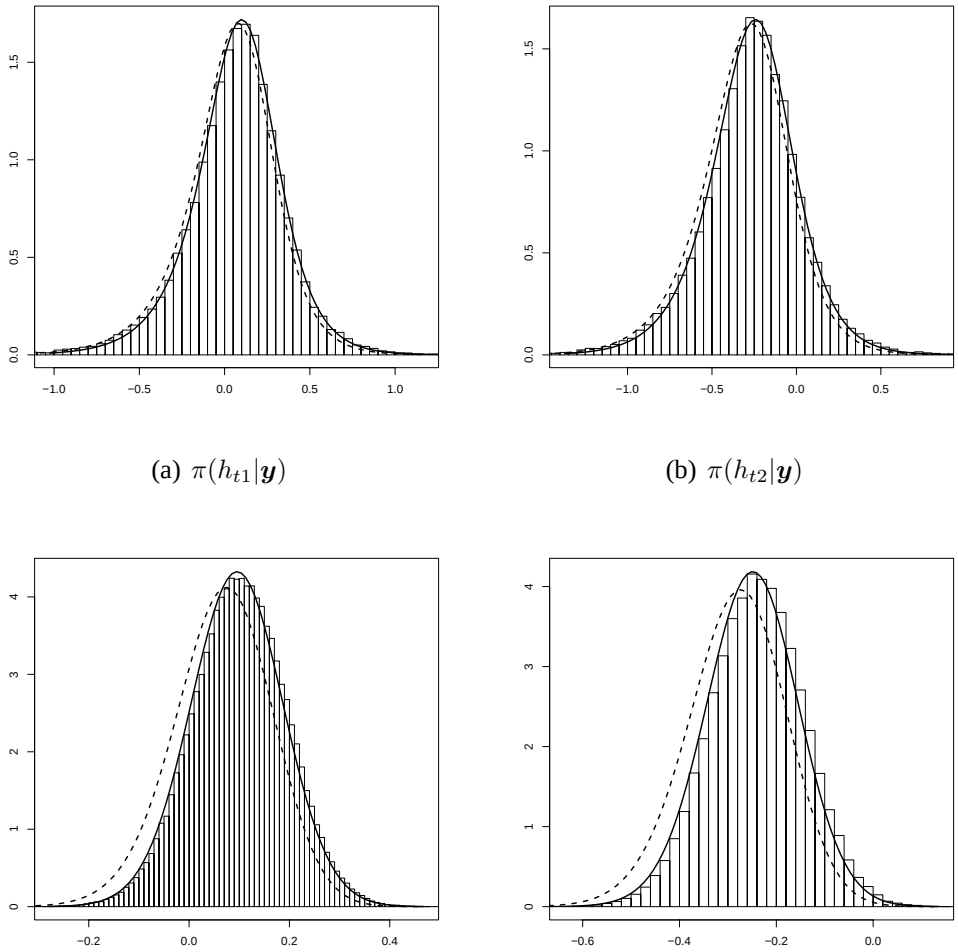
We compare approximations for $\pi(x_{ti}|\boldsymbol{y})$ obtained using the grid and the CCD integration strategy. The results are displayed in Figure 18. For each example we display the node for which the two integrations gave the most different results. Even if the CCD strategy uses four times less evaluations points compared to the grid strategy, the results are practically identical.



(a) Simulated data set (b) Real data set

Figure 18: Model 1. Results of the CCD (broken line) and grid (solid line) integration when computing $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$.

Figures 19 and 20 compare the Gaussian, the improved approximation and a density estimates obtained by an intensive MCMC run of the posterior marginals for four nodes in the latent field. Figure 19 refers to the simulated data set and Figure 20 to the real one. The nodes showed are two log-volatilities h_{t1} and h_{t2}, and the two common means μ_1 and μ_2. In both cases while the Gaussian approximation presents a slight error in locations, the improved approximation gives practically exact results.



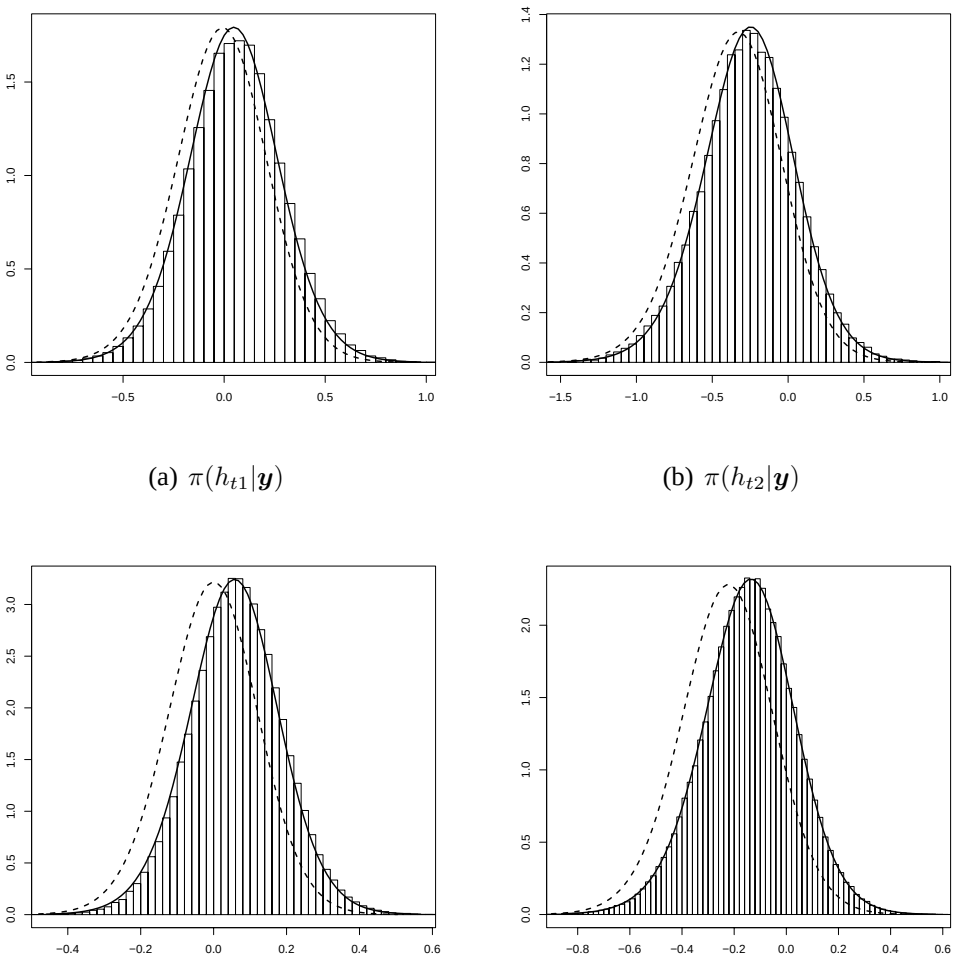
(a) $\pi(h_{t1}|\boldsymbol{y})$

(b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$

(d) $\pi(\mu_2|\boldsymbol{y})$

Figure 19: Simulated bivariate data set, Model 1. Gaussian approximation (broken line), improved approximation (solid line) and MCMC based density estimate (histogram).



(a) $\pi(h_{t1}|\boldsymbol{y})$ (b) $\pi(h_{t2}|\boldsymbol{y})$

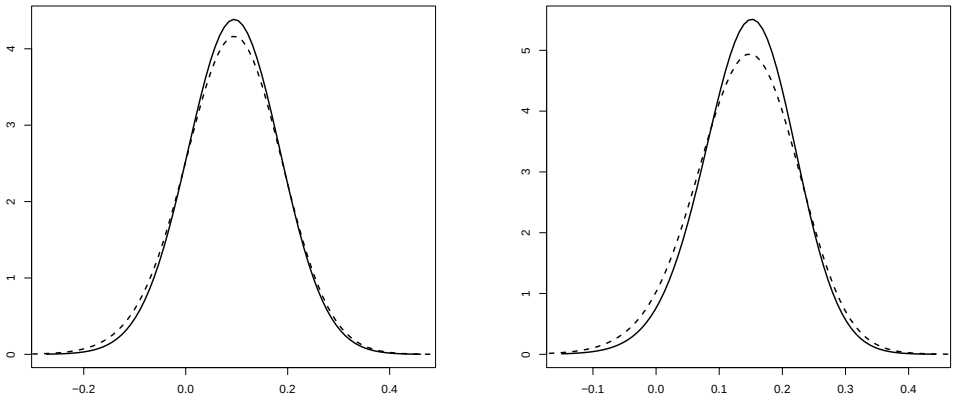
(c) $\pi(\mu_1|\boldsymbol{y})$ (d) $\pi(\mu_2|\boldsymbol{y})$

Figure 20: Australia/New Zeland data set, Model 1. Gaussian approximation (broken line), improved approximation (solid line) and MCMC based density estimate (histogram).

6.3.2 Model 2 (Constant correlation MSV)

In Model 2 the returns are correlated. Hence, in addition to the four hyperparameters of Model 1 we also have the correlation between returns. Tables 6 and 7 show the modal values of the hyperparameters and their standard deviation as approximated from the inverse of the negative Hessian matrix of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. The hyperparameter ρ_ϵ^*, which is a function of the correlations parameters ρ_ϵ (see Section 2.3), has, for the simulated data, a modal value of 0.88, which is quite close to the real value of 1. The standard deviation, if we assume a Gaussian approximation for $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ as in (25), is 0.11. Although this is a very rough estimate of the posterior marginal of ρ_ϵ^*, it suggests that the value of ρ_ϵ^* is significantly different from 0 and that the two returns time series are indeed correlated. The same can be said about the Australia/New Zeland data set where the modal value of ρ_ϵ^* is 1.99 with a Gaussian standard deviation equal to 0.11.

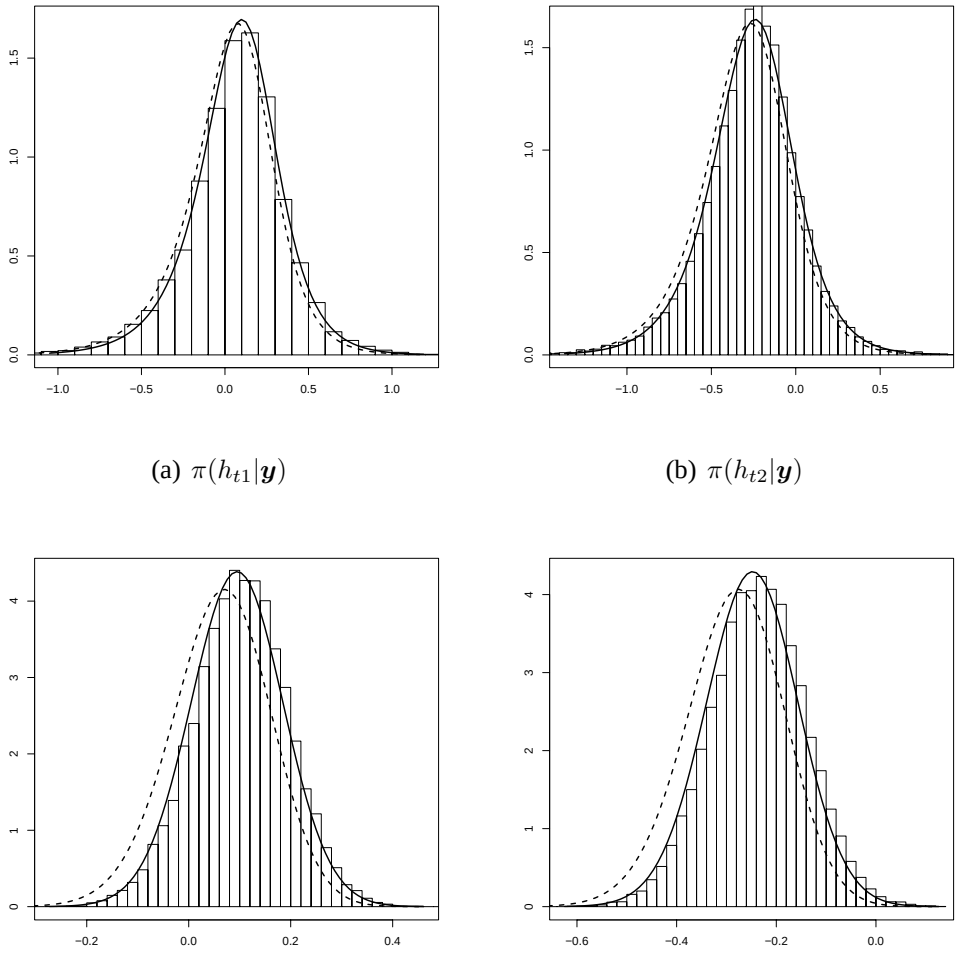
Figure 21 compares the results of the two integration strategies. Again the nodes displayed are those where the CCD integration performs worst. There is indeed a slight difference between the approximations in both examples. Anyway, when compared to the natural scale of the densities, these differences appear to be quite small. On the other side, the savings in computational time due to the use of the CCD strategy is relevant, see Table 5.



(a) Simulated data set (b) Real data set

Figure 21: Model 2. Grid (solid line) and CCD (broken line) integration results.

Figures 22 and 23 show the Gaussian and the improved approximation for some nodes in the latent field for the simulated and real data set respectively. In the same plots is also an histogram derived from an intensive MCMC run. For the real data set, there is a slight disagreement between the improved approximation and the MCMC estimate in the left tail of one of the distribution (Figure 23b). In the simulated case the approximations are practically perfect.



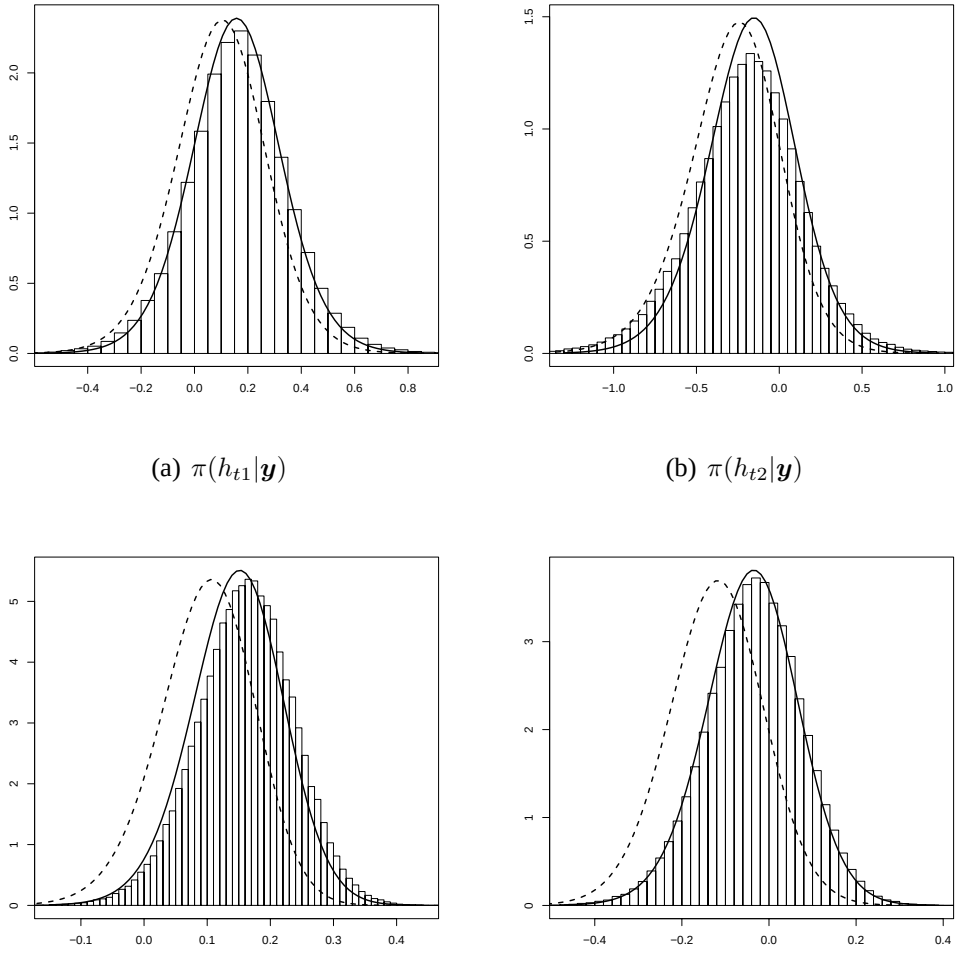
(a) $\pi(h_{t1}|\boldsymbol{y})$

(b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$

(d) $\pi(\mu_2|\boldsymbol{y})$

Figure 22: Simulated bivariate data set, Model 2. Gaussian approximation (broken line), improved approximation (solid line) and MCMC based density estimate (histogram).



(a) $\pi(h_{t1}|\boldsymbol{y})$ (b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$ (d) $\pi(\mu_2|\boldsymbol{y})$

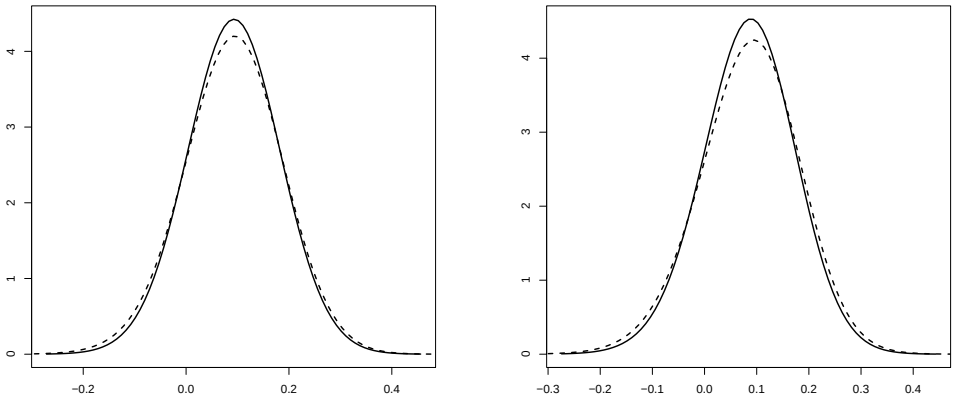
Figure 23: Australia/New Zeland data set, Model 2. Gaussian approximation (broken line) the improved approximation (solid line) and a MCMC based density estimate (histogram) for 4 nodes in the latent field.

6.3.3 Model 3 (MSV with Granger causality)

Model 3 adds one more hyperparameter by allowing the two latent time series to be inter-dependent. The cross-correlation between the time series of log-volatilities is caused by the Granger causality expressed by the non-zero value of the parameter ϕ_{21}.

Consider first the simulate data set. Here, the posterior mode the ϕ_{21} is 0.679 and its standard deviation, as derived from a Gaussian approximation to $\pi(\boldsymbol{\theta}|\boldsymbol{y})$, is 0.523, see Table 6 .This suggests that no Granger causality is present between the latent fields. This corresponds to the true model we simulated the data from.

As for the Australia/New Zeland data set, the modal value of ϕ_{21} is 0.828 and that its standard deviation, as estimated from a Gaussian approximation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$, is 0.39. This suggest ϕ_{21} being significantly different from 0 and, in turns, that the volatility in Australian dollar Granger causes the volatility in the New Zeland dollar. This is consistent with our expectations of the two economies to be strongly dependent. As a result following the Granger causality, the posterior mode of the log-range in the volatility for the New Zeland dollar is reduced from 3.58 to 1.75.

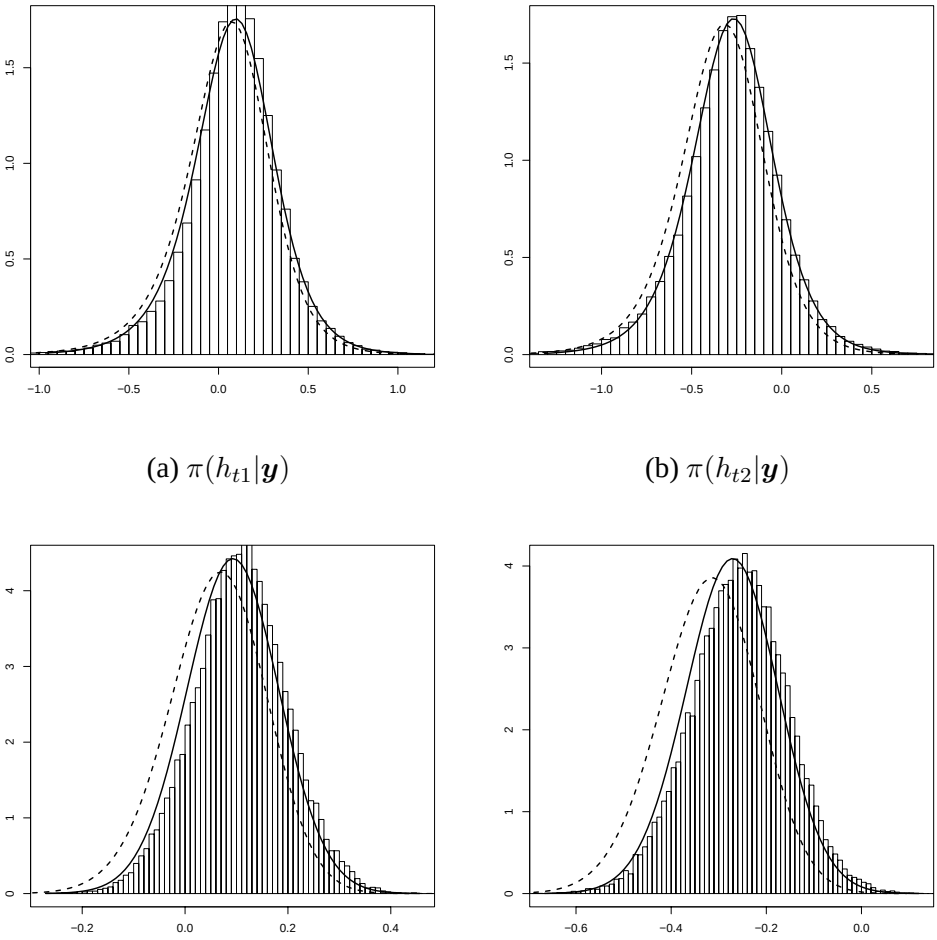


(a) Simulated data set (b) Real data set

Figure 24: Model 3. Grid (solid line) and CCD (broken line) integration results.

Figures 24 displays results obtained using the CCD and the grid strategies when approximating $\pi(x_{ti}|\boldsymbol{y})$. Again we notice that the CCD integration allows for a quite big reduction in computational costs (see Table 5) with only a slight loss in terms of accuracy.

When comparing the Gaussian and the improved approximation with a MCMC based density estimate, Figures 25 and 26 for the simulated and the data respectively, there seems to be, in both cases a slight disagreement between the improved approximation and the MCMC based estimate concerning the posterior density of $\pi(\mu_1|\boldsymbol{y})$ (Figures 25c and 26c). On one side this difference might depend on some MCMC error still present in the sample. We have seen, in fact, that the single site algorithm implemented in the WinBUGS software mixes very slowly. On the other side, when compared with the natural scale of

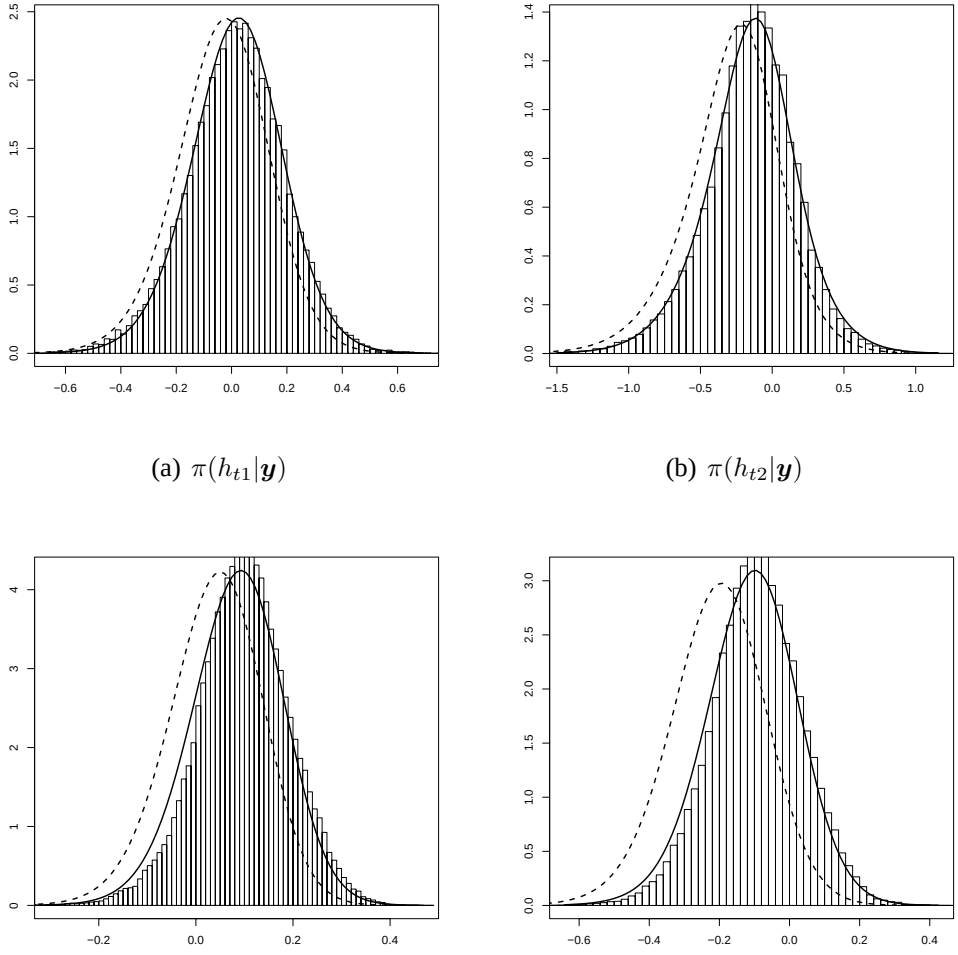


(a) $\pi(h_{t1}|\boldsymbol{y})$ (b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$ (d) $\pi(\mu_2|\boldsymbol{y})$

Figure 25: Simulated data set, Model 3. Gaussian approximation (broken line), improved approximation (solid line) and MCMC based density estimate (histogram).

the density, the small disagreement in skewness would make no difference in practice.



(a) $\pi(h_{t1}|\boldsymbol{y})$ (b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$ (d) $\pi(\mu_2|\boldsymbol{y})$

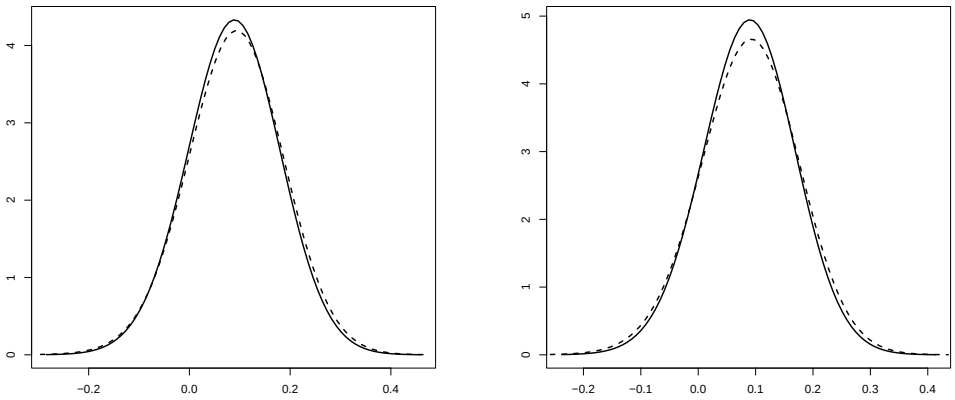
Figure 26: Australia/New Zeland data set, Model 3. Gaussian approximation (broken line) the improved approximation (solid line) and a MCMC based density estimate (histogram) for 4 nodes in the latent field.

6.3.4 Model 4 (Generalised constant correlation MSV)

Model 4 allows for cross-correlation between the volatilities but, unlike Model 3 this dependency is caused by correlations between the two processes and not by Granger causality. Hence, the hyperparameter space keeps the same dimension but ϕ_{21} is substituted by ρ^*_η.

From Table 6 we can see that the estimated modal value of ρ^*_η and the curvature of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ at the mode, suggest that the latent fields are uncorrelated for the simulation data example.

In the Australia/New Zeland case instead, the modal value of $\pi(\rho^*_\eta|\boldsymbol{y})$ is estimated to be 4.826 with a standard deviations computed by approximating $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ with a six dimensional Gaussian distribution is 0.632, see Table 7. This again suggests that the correlation between the two volatilities time series is non-zero.



(a) Simulated data set (b) Real data set

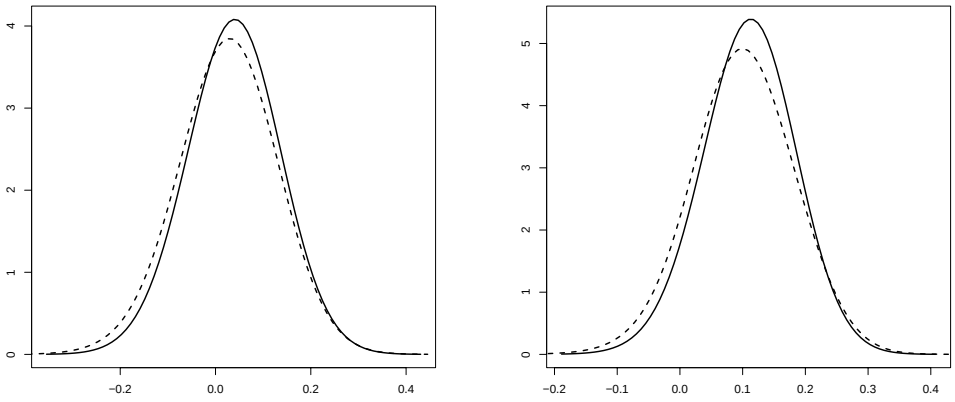
Figure 27: Model 4. Grid (solid line) and CCD (broken line) integration results.

Figure 27 show the approximations obtained by using the grid and the CCD integration scheme for both our bivariate examples. Again we see that, despite the large computational saving, the results obtained via the CCD integration are only slightly different from those obtained via the grid scheme.

When we tried to fit Model 4 to the two data set via WinBUGS we found out that the algorithm runs extremely slowly for this model. When using only the first 30 data points WinBUGS takes around 36 seconds to perform 100 iteration. The time consumed grows to circa 78 seconds for 40 data points and to 140 seconds for 50 data points. To obtain a long enough sample for the complete data set would take an extremely long time. Therefore no comparison with MCMC estimates is presented for this model.

6.3.5 Model 5 (Heavy-tailed MSV)

The last model considered is equivalent to Model 2 concerning the equation for the latent volatility models but uses a Student-t error instead of a Gaussian one in the equation for the returns. No cross-correlation in the volatility process is allowed. The number of hyperparameters is then again six.



(a) Simulated data set (b) Real data set

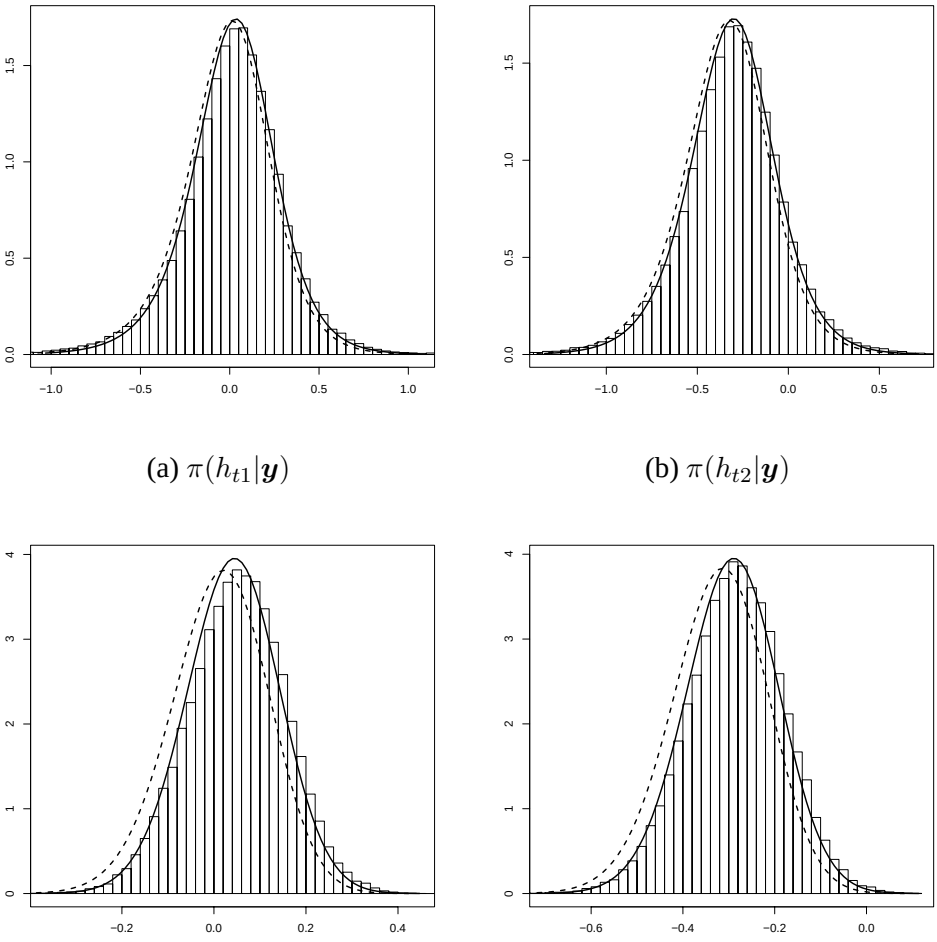
Figure 28: Model 5. Grid (solid line) and CCD (broken line) integration results.

In both our examples the modal value of δ^* is over 3, with a standard deviation computed from the Gaussian approximation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ close to 1. A value of δ^* close to 3 corresponds to a value for the degrees of freedom close to 22. This suggests that the extra kurtosis is not necessary to describe any of the two data sets.

Figure 28 compares the approximations of $\pi(x_{ti}|\boldsymbol{y})$ obtained by using the grid and the CCD strategy. As usual the nodes showing the largest differences are reported. No significant differences can be detected despite the fact that the CCD integration uses almost 20 times less evaluation points.

Figures 29 and 30 compare the Gaussian and the improved approximation with an histogram derived from a long MCMC run. While the improved approximation agrees almost perfectly with the MCMC density estimate in the simulated data example (Figure 29), in the Australia/New Zeland example there is a slight disagreement between the two. This can be seen especially in the left tail of Figure 30b and in the location and skewness of the density in Figure 30c.

As an experiment we have run the same model this time only taking into account the first 50 points in the Australia/New Zeland data set, so that the MCMC algorithm would run faster. Again we have compared the histogram resulting from such MCMC run with the Gaussian and improved approximation. The results are displayed Figure 31. This time the improved approximation and the MCMC density estimates overlap almost perfectly. Following the same argument as for the simulated data in Section 6.2, we believe that

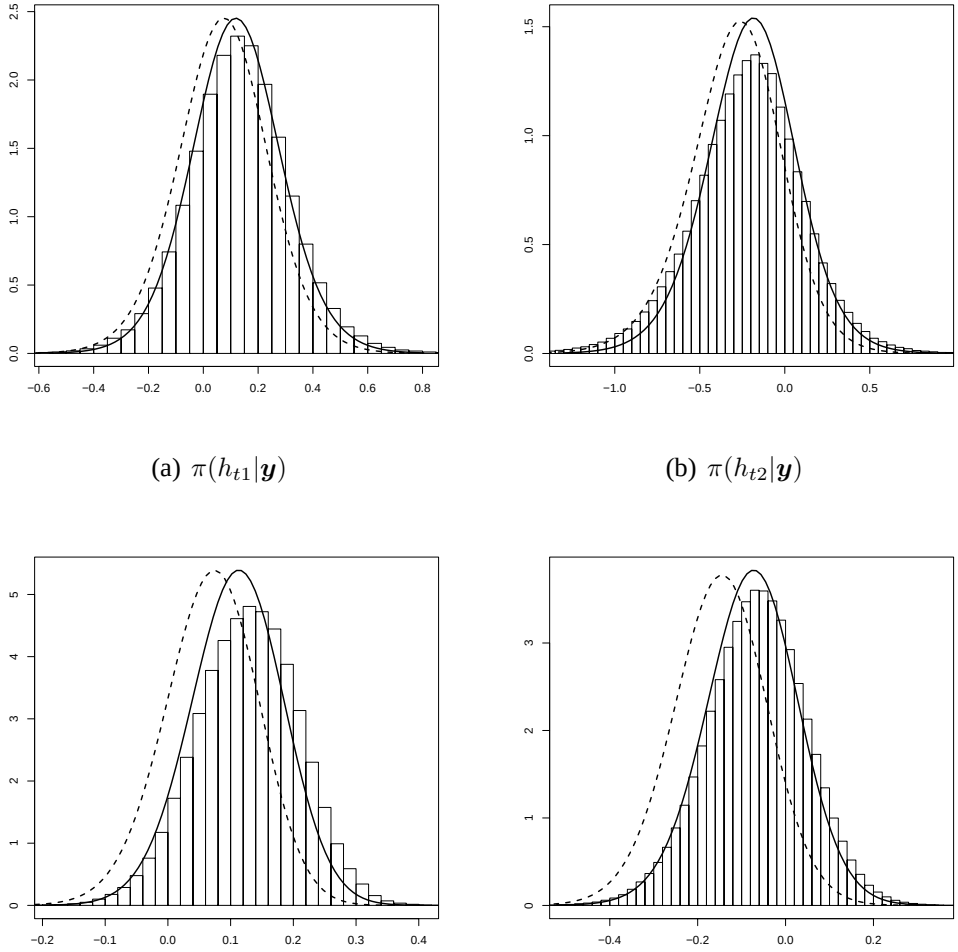


(a) $\pi(h_{t1}|\boldsymbol{y})$ (b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$ (d) $\pi(\mu_2|\boldsymbol{y})$

Figure 29: Simulated data set, Model 5. Gaussian approximation (broken line), improved approximation (solid line) and MCMC based density estimate (histogram).

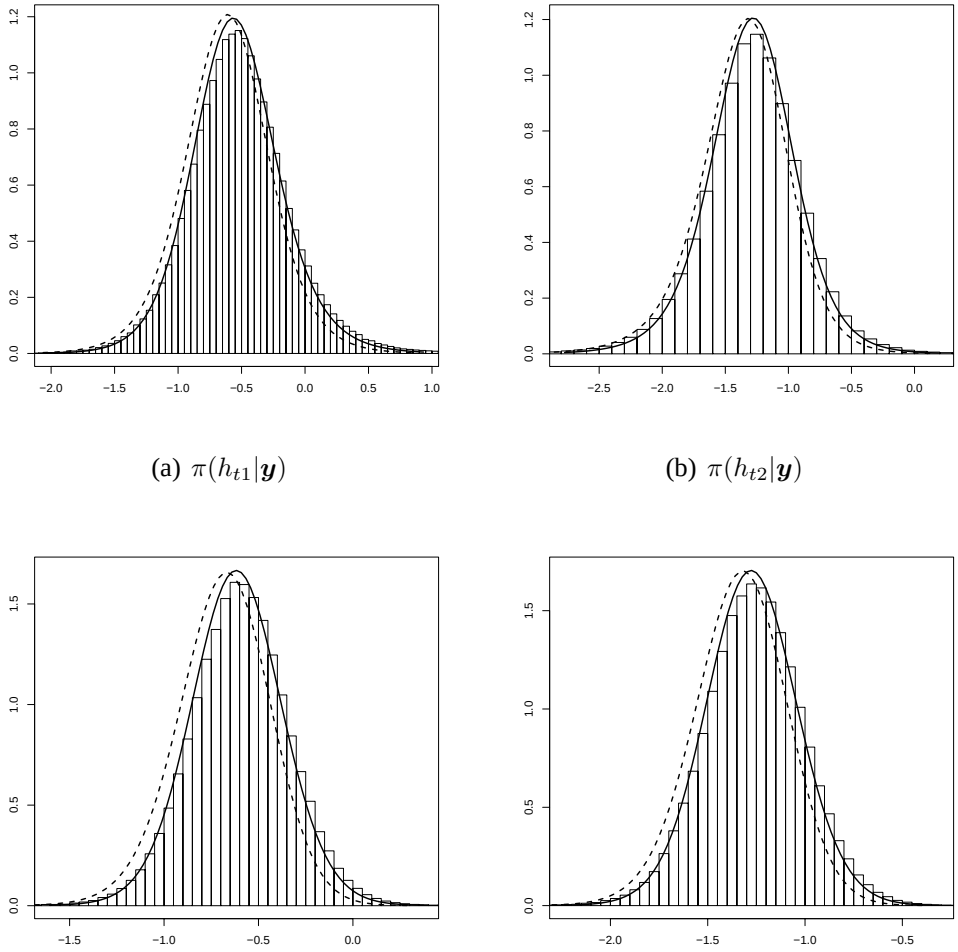
running the MCMC algorithm long enough the approximation and the MCMC estimate would coincide also for the full data set.



(a) $\pi(h_{t1}|\boldsymbol{y})$ (b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$ (d) $\pi(\mu_2|\boldsymbol{y})$

Figure 30: Australia/New Zeland data set, Model 5. Gaussian approximation (broken line), improved approximation (solid line) and MCMC based density estimate (histogram).



(a) $\pi(h_{t1}|\boldsymbol{y})$ (b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$ (d) $\pi(\mu_2|\boldsymbol{y})$

Figure 31: Australia/New Zeland data set, Model 5. Gaussian approximation (broken line), improved approximation (solid line) and MCMC based density estimate (histogram) when only the first 50 data are considered.

6.4 Model comparison

In this section we compare the five bivariate models using the two approximations to the marginal likelihood $\pi(\boldsymbol{y}|\mathcal{M}_k)$ described in Section 5.

Table 8 reports the values of $\log \widetilde{\pi}(\boldsymbol{y}|\mathcal{M}_k)$, for all five models fitted to the simulated data set. In the same table is also the ranking associated with each of the models.

	$\log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k)$	$\log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k)$	Rank	$\log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k) -$ $\max_k \log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k)$	$\log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k) -$ $\max_k \log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k)$
Model 1	-295.782	-296.4741	5	-24.802	-22.5181
Model 2	-270.980	-273.9560	1	0.000	0.0000
Model 3	-273.605	-277.8360	4	-2.625	-3.8800
Model 4	-271.247	-274.4130	2	-0.267	-0.4570
Model 5	-272.435	-275.5620	3	-1.455	-1.6060

Table 8: Simulated data set: approximated value for $\log \pi(\boldsymbol{y}|\mathcal{M}_k)$ for the bivariate models computed via Gaussian approximation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and via numerical integration. In the third column is the ranking of the models according to the value of the marginal likelihood. The last two columns are the relative values of the marginal likelihood.

Although the Gaussian approximation of the marginal likelihood $\pi(\boldsymbol{y}|\mathcal{M}_k)$ is a quite rough approximation since it does not take into account any departure from a multivariate normal distribution, it gives the same ranking as the more accurate approximation computed via numerical integration. When comparing models what counts is not the absolute value of $\pi(\boldsymbol{y}|\mathcal{M}_k)$, but rather the differences between the values of $\pi(\boldsymbol{y}|\mathcal{M}_k)$ relative to different models. We have computed $(\log \widetilde{\pi}(\boldsymbol{y}|\mathcal{M}_k) - \max_k \log \widetilde{\pi}(\boldsymbol{y}|\mathcal{M}_k))$ for both approximations and reported it in Table 8 to show that the discrepancy between the two approximations is larger when we look at absolute values than when we look at the more interesting relative values.

The highest value of the marginal likelihood corresponds to Model 2, which is actually the model we simulated the data from. According to the marginal likelihood criteria, Model 4 receives practically the same support from the data as Model 2. The difference in log marginal likelihood between Model 2 and Model 1 is more than 20 indicating that some kind of dependence between the two time series is definitely present.

Results regarding the Australia/New Zeland data set are in Table 9. The model ranked as best is Model 4 which allows for correlations in both the returns and the volatilities. This agrees well with our prior idea that the economies of Australia and New Zeland are closely related. The difference in log marginal likelihood between Model 4 and Model 3, which is ranked as second best, is 1.8. Both these models imply interdependence in the returns process and in the latent volatility one. The difference being only in the nature of such interdependence.

The difference in log marginal likelihood between the best model (Model 4) and the two models which allow interdependence only in the returns process (Models 2 and 5) is over

	$\log \widetilde{\pi}_1(\boldsymbol{y}\vert\mathcal{M}_k)$	$\log \widetilde{\pi}_2(\boldsymbol{y}\vert\mathcal{M}_k)$	Rank	$\log \widetilde{\pi}_1(\boldsymbol{y}\vert\mathcal{M}_k) -$ $\max_k \log \widetilde{\pi}_1(\boldsymbol{y}\vert\mathcal{M}_k)$	$\log \widetilde{\pi}_2(\boldsymbol{y}\vert\mathcal{M}_k) -$ $\max_k \log \widetilde{\pi}_2(\boldsymbol{y}\vert\mathcal{M}_k)$
Model 1	-580.342	-585.131	5	-200.823	-198.045
Model 2	-385.995	-391.332	3	-6.476	-4.246
Model 3	-381.294	-388.942	2	-1.775	-1.856
Model 4	-379.519	-387.086	1	0.000	0.000
Model 5	-387.352	-392.612	4	-7.833	-5.526

Table 9: Australia/New Zeland data set: approximated value for $\log \pi(\boldsymbol{y}|\mathcal{M}_k)$ for the bivariate models computed via Gaussian approximation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and via numerical integration. In the third column is the ranking of the models according to the value of the marginal likelihood. The last two columns are the relative values of the marginal likelihood.

7. This implies very strong evidence against these two models.

Finally, Model 1, which assumes total independence between the two time series can definitely be rejected, its log marginal likelihood being more than 200 smaller that the one of Model 4.

Yu and Mayer (2006) fit all these five models, although with a different parametrisation, to the same data set. They rank the models using the deviation information criteria (DIC) obtaining the same ranking as we do here.

7 Approximating posterior marginals for the hyperparameters $\pi(\theta_m|\boldsymbol{y})$

In some cases one might be interested in investigating the marginal posterior distribution for the hyperparameters of the model, $\pi(\theta_m|\boldsymbol{y})$ for $m = 1, \ldots, M$. In Section 3.2 an approximation to the joint posterior $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$, is introduced. Moreover, in the examples in Section 6 we have seen that some information about the marginals $\pi(\theta_m|\boldsymbol{y})$ can be obtained by approximating the joint marginal for the hyperparameters $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ with a multivariate normal distribution with mean at the modal value $\boldsymbol{\theta}^*$ of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ and covariance matrix equal to the inverse of the negative Hessian matrix of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ computed at $\boldsymbol{\theta}^*$. This Gaussian approximation for $\pi(\theta_m|\boldsymbol{y})$ is quite rough, it does not take into account the skewness which often is present in the posterior density of the hyperparameters. In some cases we might, therefore, be interested in a more accurate approximation of $\pi(\theta_m|\boldsymbol{y})$.

Theoretically, given $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ the integral

$$\widetilde{\pi}(\theta_m|\boldsymbol{y}) = \int \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\, d\boldsymbol{\theta}_{-m} \tag{29}$$

can be computed numerically, thus providing the required approximation. In practice though, as all numerical integration problems, also this becomes more and more computational demanding with increasing dimension of $\boldsymbol{\theta}$.

In our experience, there seems to be no real "trick" to avoid the rather heavy computational procedures needed for evaluating $\widetilde{\pi}(\theta_m|\boldsymbol{y})$, which means that obtaining a precise approximation to the posterior marginals of the hyperparameters will always result in a time-consuming process.

In the following, we present different strategies to evaluate the integral in (29). Both strategies in Sections 7.1 and 7.2 give quite accurate results but require extra computations with respect to those used to approximate $\pi(x_{it}|\boldsymbol{y})$. The strategies in Section 7.3 instead, are intended to provide an approximation, not necessarily very accurate but still useful, by using quantities already computed when computing $\pi(x_{it}|\boldsymbol{y})$.

7.1 Numerical integration via regular grid

For not too high dimension of $\boldsymbol{\theta}$, it is possible to evaluate $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ on a regular grid and then use the resulting values to numerically compute the integral in (29). In order to locate the area of highest probability density we can use a strategy similar to that described in Algorithm 3 with two modifications. First the negative Hessian $\boldsymbol{H}$ is replaced by its diagonal. This because the rotation of the axis due to $\boldsymbol{V}$ in equation (18) is inconvenient when summing out the variables $\boldsymbol{\theta}_{-m}$. Using only the diagonal of $\boldsymbol{H}$ suppresses the rotation but maintains the scaling. Second, in order to obtain a regular grid of points we include all the fill in configurations whether or not condition (19) is fulfilled.

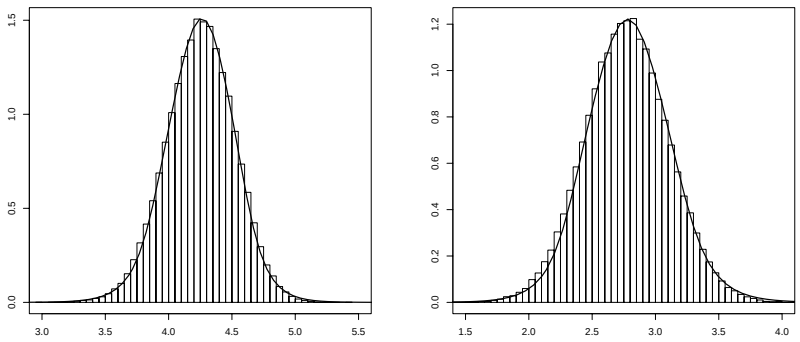
After having computed the value of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ for all points on the grid, by summing out the variables $\boldsymbol{\theta}_{-m}$, we obtain, for each dimension M, a series of points $\{\theta_m^1, \ldots, \theta_m^l\}$ with relative density $\{\widetilde{\pi}(\theta_m^1), \ldots, \widetilde{\pi}(\theta_m^l)\}$. We can then fit a spline to the values of the log-density in order to obtain a smooth estimate.

This is the strategy used in Rue and Martino (2006) and Rue et al. (2007) to approximate posterior marginals for hyperparameters, and has proved to give extremely accurate results when compared to those obtained by intensive MCMC runs, provided that the grid is wide and dense enough.

Unfortunately, in order to achieve precise approximations of the $\pi(\theta_m|\boldsymbol{y})$, especially in the tails, the grid has to be wider than the one used to compute $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ and in some cases also finer. This means that we have to set the tuning parameter δ_π to a higher value, lets say 5 and, in some cases, set δ_z to a value smaller than 1. This, together with the fact that we compute all fill in configurations, implies that with, increasing dimension of $\boldsymbol{\theta}$, the computation becomes soon very heavy. Moreover, computing approximations to $\pi(\theta_m|\boldsymbol{y})$ as described here, does not make use of the values of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ evaluated to compute $\pi(x_{ti}|\boldsymbol{y})$ using the grid strategy as described in Section 4.1.1, but implies additional computations.

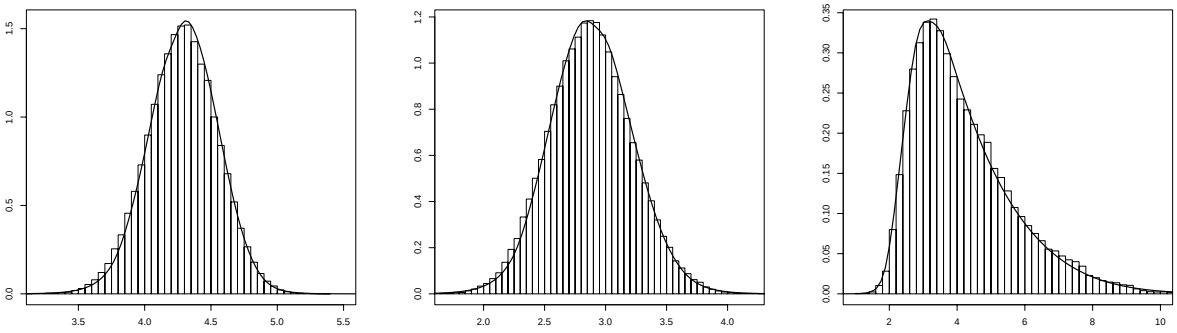
As examples of this strategy, we have approximated $\pi(\theta_m|\boldsymbol{y})$, $m = 1, \ldots, M$ for the two univariate models, $\mathcal{M}_1$ and $\mathcal{M}_2$ in Section 2.1, fitted to the pound-dollar exchange rate data set. The two models have respectively two and three hyperparameters.

Figure 32 displays the approximations for $\pi(\theta_m|\boldsymbol{y})$, $m = 1, \ldots, M$ in model $\mathcal{M}_1$ com-



(a) $\pi(\log \kappa|\boldsymbol{y})$ (c) $\pi(\log \tau|\boldsymbol{y})$

Figure 32: Posterior marginals for the hyperparameters in $\mathcal{M}_1$ fitted to the Pound/Dollar data set. The solid line is the approximation computed via the regular grid integration, the histogram is based on intensive MCMC run.



(a) $\pi(\log \kappa|\boldsymbol{y})$ (c) $\pi(\log \tau|\boldsymbol{y})$ (c) $\pi(\delta|\boldsymbol{y})$

Figure 33: Posterior marginals for the hyperparameters in $\mathcal{M}_2$ fitted to the Pound/Dollar data set. The solid line is the approximation computed via the regular grid integration, the histogram is based on intensive MCMC run.

pared with MCMC based density estimates, and Figure 33 displays model $\mathcal{M}_2$. The approximations and the MCMC-based estimates agree very well. The size of the grid used to compute $\widetilde{\pi}(\theta_m|\boldsymbol{y})$ is 70 for model $\mathcal{M}_1$ and 1300 for model $\mathcal{M}_2$. It is clear, then, that when the dimension of the hyperparameters space increases, this strategy for computing posterior marginals for the hyperparameters becomes soon really computational intensive.

7.2 Laplace approximation

An alternative way to evaluate $\widetilde{\pi}(\theta_m|\boldsymbol{y})$ is to use once more the Laplace approximation. The starting point is the identity:

$$\pi(\theta_m|\boldsymbol{y}) = \frac{\pi(\boldsymbol{\theta}|\boldsymbol{y})}{\pi(\boldsymbol{\theta}_{-m}|\theta_m, \boldsymbol{y})}.$$

We already have an approximation for $\pi(\boldsymbol{\theta}|\boldsymbol{y})$, then

$$\widetilde{\pi}(\theta_m|\boldsymbol{y}) \propto \left.\frac{\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})}{\widetilde{\pi}_G(\boldsymbol{\theta}_{-m}|\theta_m, \boldsymbol{y})}\right|_{\boldsymbol{\theta}_{-m}=\boldsymbol{\theta}^*_{-m}} \tag{30}$$

where $\boldsymbol{\theta}^*_{-m}$ is the modal configuration of $\widetilde{\pi}(\boldsymbol{\theta}_{-m}|\theta_m, \boldsymbol{y})$ for different values of θ_m and $\widetilde{\pi}_G(\boldsymbol{\theta}_{-m}|\theta_m, \boldsymbol{y})$ is a Gaussian approximation to $\widetilde{\pi}(\boldsymbol{\theta}_{-m}|\theta_m, \boldsymbol{y})$ built by matching the mode and the curvature at the mode. That is a Gaussian density with mean equal to $\boldsymbol{\theta}^*_{-m}$ and precision matrix equal to the negative of the Hessian matrix of $\widetilde{\pi}(\boldsymbol{\theta}_{-m}|\theta_m, \boldsymbol{y})$ computed at the mode.

In order to get a smooth approximation to $\pi(\theta_m|\boldsymbol{y})$ we can compute the quantity in (30) for a set of different values of θ_m and then fit a spline to the logarithm of the obtained values. The density needs then to be numerically normalised so that it integrates to one. The whole procedure has to be repeated for each of the marginal distribution $\pi(\theta_m|\boldsymbol{y})$ we are interested in.

The Laplace approximation as described above, gives quite accurate results when compared to density estimates obtained from intensive MCMC runs. As an example we have computed the marginal posterior densities for all the hyperparameters in Model 2 fitted to the simulated data set in Figure 16. The results are displayed in Figure 34. Here the Laplace approximation in (30) is shown as a solid line. The histograms are based on long (10^6) MCMC runs. In all cases the approximated densities agree almost perfectly with the estimated ones.

Unfortunately, computing the expression in (30) implies finding the maximum of the $(M-1)$ dimensional function $\widetilde{\pi}(\boldsymbol{\theta}_{-m}|\theta_m, \boldsymbol{y})$ for each value of θ_m. This operation, with increasing dimension of the hyperparameters space and of the latent field $\boldsymbol{x}$, might become very costly.

In order to simplify the computations we have tried to substitute, when computing (30), the conditional mode $\boldsymbol{\theta}^*_{-m}$ with the conditional mean $\mathrm{E}_G(\boldsymbol{\theta}_{-m}|\theta_m)$ computed from the Gaussian approximation $\widetilde{\pi}_G(\boldsymbol{\theta}|\boldsymbol{y})$ in equation (25). The conditional mean can be computed in no time thanks to the usual properties of the multivariate Gaussian distribution, therefore the computational time is reduced a lot. In fact, the only time-consuming operation left to perform is the computation of Hessian of $\widetilde{\pi}(\boldsymbol{\theta}_{-m}|\boldsymbol{y}, \theta_m)$ at $\mathrm{E}_G(\boldsymbol{\theta}_{-m}|\theta_m)$. This resembles what we have already done in Section 4.2.2 when computing the improved approximation for $\pi(x_{ti}|\boldsymbol{y})$. The idea of substituting the conditional mode with the conditional mean is based on the presupposition that the density of interest, $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ here and $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ in Section 4.2.2, is not "too far" from its Gaussian approximation built

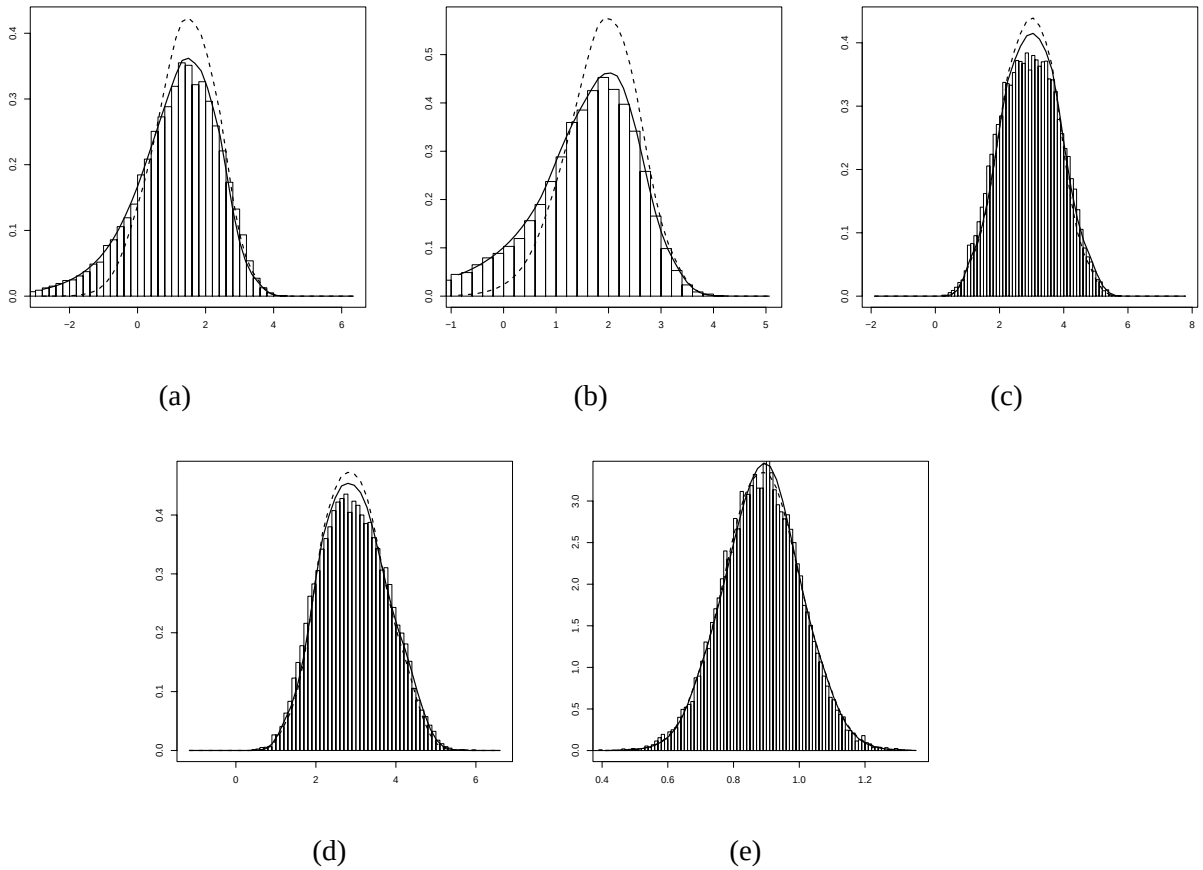


Figure 34: Posterior marginals for the hyperparameters in Model 2 fitted to the simulated data in Figure 16. The solid line is the Laplace approximation where (30) is computed at the conditional mode while the broken line is the Laplace approximation where (30) is computed at the conditional mean. The histogram is based on intensive MCMC run.

by matching the mode and the curvature at the mode. While this is essentially true for $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$, $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ can differ quite a lot from a Gaussian given also that the prior $\pi(\boldsymbol{\theta})$ is not Gaussian.

The results of approximating $\widetilde{\pi}(\theta_m|\boldsymbol{y})$ using (30) computed at the conditional mean instead of the conditional mode for Model 2 fitted to the simulated data set are displayed in Figure 34 as a broken line. Clearly the Laplace approximation computed at the conditional mean underestimates the skewness of the marginal posteriors when this is large.

7.3 Integration via an interpolating function

The procedures described in this section provide an approximation for $\pi(\theta_m|\boldsymbol{y})$ using values of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ already computed during the numerical integration of $\widetilde{\pi}(x_{it}|\boldsymbol{y})$ described in Section 4.1. The posterior marginals obtained are not necessarily accurate but provide the user with useful results.

When evaluating $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ using the grid integration strategy in Section 4.1.1 we compute the density $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$ for a certain number K of points. Although they cannot be directly used to compute $\widetilde{\pi}(\theta_m|\boldsymbol{y})$, these points carry information about the shape of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ in the area with highest density. We propose to use the K points in the grid to build a M-dimensional interpolating function $f(\boldsymbol{\theta})$. This can then be easily computed for any point inside the grid in order to numerically compute the integral in (29).

The main advantage of this approach is that, unlike the grid strategy presented in Section 7.1, it requires no extra computations of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ with respect to the computation of $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$. In fact, the same evaluation points $\boldsymbol{\theta}_k$ in the hyperparameters space, are used to compute all the posterior marginals in the model. Unfortunately building a $M-1$ dimensional interpolating function is not straight forward. We have implemented three different interpolating functions:

Function 1: Compute $f(\boldsymbol{\theta})$ as a weighted sum of the K values $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$, $k = 1, \ldots, K$, that is $f(\boldsymbol{\theta}) = \sum w_k \widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$. The weights w_k depend on the Euclidean distance of $\boldsymbol{\theta}$ from each $\boldsymbol{\theta}_k$.

Function 2: Compute $f(\boldsymbol{\theta})$, as the linear interpolation form the $M+1$ points nearest to $\boldsymbol{\theta}$.

Function 3: Compute $f(\boldsymbol{\theta})$, as the quadratic interpolation form the $M+1$ points nearest to $\boldsymbol{\theta}$. The curvature is assumed to be 1 as for the standard Gaussian density.

Function 1 seems to provide approximations which tends to be too smooth with respect to the real posterior densities while Function 2 and 3 can, sometimes, present spikes which make the numerical integration difficult. Moreover, when the dimension of $\boldsymbol{\theta}$ increases, not only computing the grid, but also computing $f(\boldsymbol{\theta})$ itself becomes expensive. In fact, computing any of the three functions described above requires visiting all the K points which constitutes the grid, and their number grows exponentially with M. Results obtained using Function 1 to interpolate the K points for the univariate Student-t ($\mathcal{M}_2$) model fitted to the Pound-Dollar data set, are displayed in Figure 35. Notice that the approximations, especially for $\pi(\nu^*|\boldsymbol{y})$ are too smooth.

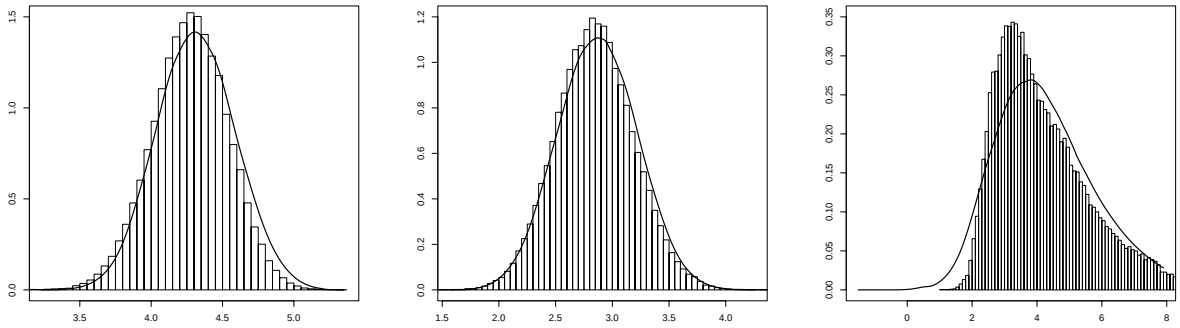
If the CCD strategy is used to compute $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ no grid on the hyperparameter space is available. Hence a different strategy has to be used. Let $\boldsymbol{z}(\boldsymbol{\theta}) = (z_1(\boldsymbol{\theta}), \ldots, z_M(\boldsymbol{\theta}))$ be the point in the $\boldsymbol{z}$-parametrisation defined in (18) corresponding to $\boldsymbol{\theta}$. We define the function $f(\boldsymbol{\theta})$ as

$$f(\boldsymbol{\theta}) = \prod_{m=1}^{M} f_m(z_m(\boldsymbol{\theta})) \tag{31}$$

where

$$f_m(z) = \begin{cases} \exp\left(-\frac{1}{2(\sigma_{ccd}^{m+})^2} z^2\right) & \text{if} \quad z \geq 0 \\ \exp\left(-\frac{1}{2(\sigma_{ccd}^{m-})^2} z^2\right) & \text{if} \quad z < 0 \end{cases} \tag{32}$$

and σ_{ccd}^{m+} and σ_{ccd}^{m-}, $m = 1, \ldots, M$, are defined at page 18. The function in (31) is not an interpolating function. It seems, however, to have some advantages over the three functions described above. First of all it is much faster to compute, regardless the dimension



(a) $\pi(\log \kappa_1|\boldsymbol{y})$ (b) $\pi(\log \kappa_2|\boldsymbol{y})$ (c) $\pi(\nu^*|\boldsymbol{y})$

Figure 35: Hyperparameters for the Student-t model fitted to the Pound-Dollar data set. The solid line is the approximation obtained via the interpolation Function 2 and the histogram is derived from a long MCMC run.

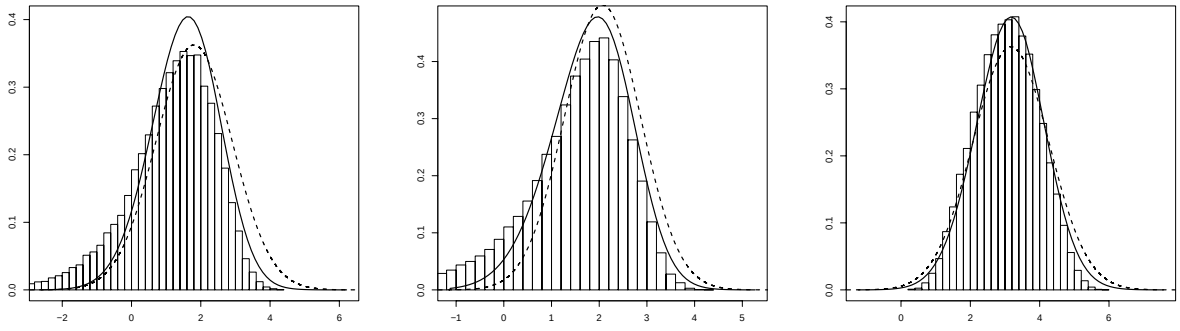
of $\boldsymbol{\theta}$, since it does not require visiting any other point in the hyperparameter space. Moreover, when the dimension of $\boldsymbol{\theta}$ is large we do not use the grid strategy for computing $\widetilde{\pi}(x_{it}|\boldsymbol{y})$ therefore the points constituting the grid are not available.

In Figure 36 we report the approximations for $\pi(\theta_m|\boldsymbol{y})$, $m = 1, \ldots, 6$ obtained using (31) for Model 5 fitted to the simulated data set. In the same Figure are also displayed the Gaussian approximations for $\pi(\theta_m|\boldsymbol{y})$ in (25), and an histogram derived from a long MCMC run. The approximations derived from (31) correct the Gaussian ones for locations and some skewness. Even though they are not extremely precise they still provide useful information about the marginals for the hyperparameters. The fact that this approximations are computed at almost no extra cost after having computed $\widetilde{\pi}(x_{it}|\boldsymbol{y})$ makes them valuable.

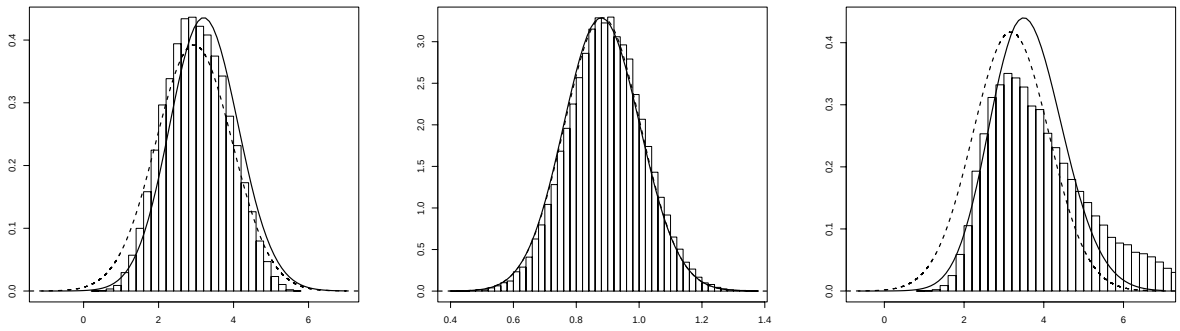
The approximations based on (31) seem to be more reliable than the one based on the interpolating functions described at page 56. They can also be computed when the grid integration strategy is used at the cost of computing the positive and negative "standard deviations" σ_{ccd}^{m+} and σ_{ccd}^{m-}, $m = 1, \ldots, M$.

8 Extension: asymmetric models

One feature often observed in financial studies is that volatility responds asymmetrically to positive and negative return shocks. Several explanations have have been proposed in the literature to explain the presence of such asymmetric relationship between volatility and returns. One of the most widely cited is due, to Black (1976) and Christie (1982) who suggest that the asymmetry reflects a change in financial leverage. In particular, the argument is that, when a firm experiences a positive (negative) return, it becomes less (more) risky, thus decreasing (increasing) its volatility. In other words there is a negative correlations between returns and volatility. This is known as *leverage* effect.



(a) $\pi(\log \kappa_1|\boldsymbol{y})$ (b) $\pi(\log \kappa_2|\boldsymbol{y})$ (c) $\pi(\tau_{\eta_1}\boldsymbol{y})$



(d) $\pi(\log \tau_{\eta_2}|\boldsymbol{y})$ (e) $\pi(\rho^*_\epsilon|\boldsymbol{y})$ (f) $\pi(\nu^*|\boldsymbol{y})$

Figure 36: Posterior marginals for the hyperparameters on Model 5 fitted to the simulated bivariate data set. The solid line is the approximation based on 31 while the broken line is the Gaussian approximation in (25). The histograms are based on intensive MCMC runs.

A univariate SV model with leverage effect was first introduces by Harvey and Shephard (1996) and takes the form:

$$\begin{array}{rcl} y_t & = & \exp(h_t/2)\epsilon_t, \\ x_{t+1} & = & \mu + \phi(h_t - \mu) + \sigma\eta_{t+1} \end{array} \tag{33}$$

where ϵ_t and η_{t+1} are standard Gaussian variables. The leverage effect is introduced by letting the two error processes to be negatively correlated. Formally, Corr$(\epsilon_t, \eta_{t+1}) = \rho$, with $\rho < 0$. Note that for asymmetric models we prefer the formulation in (33) over the one in (6), used in Jacquier et al. (2004). This is because in model (33) a shock at time t influences the volatility at time $t + 1$, while in model (6) a shock at time t influences the volatility at time t. The former being more logically appealing both from a theoretical and a empirical point of view, see Yu (2005). The SV model with leverage effect in (33) is estimated by quasi-likelihood method in Harvey and Shephard (1996) and by MCMC in Mayer and Yu (2000).

In this section we describe how it is possible to perform approximate inference using INLA for univariate SV models with correlated errors. We have not implemented the algorithms for such kind of models, therefore no example is presented.

The core of the INLA approach is the Gaussian approximation for the full conditional of the latent field $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ described in Section 3.1. In order to be able to write down such approximation we need to have an expression for the likelihood of each data point $\pi(y_t|\boldsymbol{x}, \boldsymbol{\theta})$. After some algebra ot can be showed that

$$\pi(y_t|\boldsymbol{x}, \boldsymbol{\theta}) = \pi(y_t|x_t, x_{t+1}, \boldsymbol{\theta}) = \mathcal{N}\left\{\frac{\rho}{\sigma}e^{x_t/2}[x_{t+1} - \mu + \phi(x_t - \mu)], e^{x_t}(1-\rho^2)\right\} \quad (34)$$

See Appendix for details on how to derive (34) from (33). Note that unlike the univariate models analysed on Section 2.1, here each data point y_t depends on two nodes of the latent field, namely, x_t and x_{t+1}. The prior distribution for the latent GMRF $\boldsymbol{x}$ is unchanged from Section 2.1. Hence, the full conditional reads

$$\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) \propto \exp\left\{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \sum_{t=1}^{n_d} f_t(x_t, x_{t+1})\right\} \quad (35)$$

where $f_t(x_t, x_{t+1}) = \log \pi(y_t|x_t, x_{t+1}, \boldsymbol{\theta})$. Similarly to what is done in Section 3.1, we can expand $f_t(x_t, x_{t+1})$ around the point (x_t^0, x_{t+1}^0) obtaining

$$f_t(x_t, x_{t+1}) \approx \text{Const} + (x_t, x_{t+1})\boldsymbol{b}_t - \frac{1}{2}(x_t, x_{t+1})\boldsymbol{C}_t(x_t, x_{t+1})^T.$$

where $\boldsymbol{C}_t$ is a 2×2 symmetric matrix and $\boldsymbol{b_t}$ a column vector if dimension 2. Both $\boldsymbol{b}_t$ and $\boldsymbol{C}_t$ are functions of the gradient and the Hessian matrix of $f_t(x_t, x_{t+1})$ computed at (x_t^0, x_{t+1}^0) and depend on the value of the hyperparameters vector $\boldsymbol{\theta}$. Let c_{ij}^t indicate the element ij of the matrix $\boldsymbol{C}_t$ and b_i^t indicate the ith element of vector $\boldsymbol{b}_t$, where $i, j = 1, 2$. Moreover let

$$\text{diag}(\boldsymbol{C}) = \begin{bmatrix} c_{11}^1 & c_{12}^1 & 0 & 0 & \dots & 0 \\ c_{21}^1 & c_{22}^1 + c_{11}^2 & c_{12}^2 & 0 & \dots & 0 \\ 0 & c_{21}^2 & c_{22}^2 + c_{11}^3 & c_{12}^3 & \dots & 0 \\ \vdots & & & & & \vdots \\ 0 & & & & \ddots & 0 \\ 0 & & & & \dots & 0 \end{bmatrix},$$

and

$$\boldsymbol{b}^T = [b_1^1, b_2^1 + b_1^2, b_2^2 + b_1^3, \dots, 0]$$

Here $\text{diag}(\boldsymbol{C})$ is a $N \times N$ matrix, where N is the dimension of the latent field $\boldsymbol{x}$ and $\boldsymbol{b}$ is a vector of length N. Similarly to what described in Section 3.1, we can build a Gaussian approximation to $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ with precision matrix $\boldsymbol{Q} + \text{diag}(\boldsymbol{C})$ and mean given by the solution of $(\boldsymbol{Q} + \text{diag}(\boldsymbol{C}))\boldsymbol{x}^* = \boldsymbol{b}$ where $\boldsymbol{x}^*$ is the modal configuration of $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$. Note that since x_t and x_{t+1} are neighbours in the graph of the latent field $\boldsymbol{x}$, the Gaussian approximation is a Gaussian Markov random field with respect to the same graph and therefore preserves the Markov properties of the prior distribution of the latent field $\boldsymbol{x}$.

Starting from the Gaussian approximation described above, it is possible to derive all the other algorithms necessary to implement the INLA approach also for SV models with correlated errors.

9 Discussion

The purpose of this report was to present one more class of models where Integrated Nested Laplace approximation, introduced in Rue et al. (2007) can be used. In this report we apply INLA to different bivariate stochastic volatility models obtaining approximations to the posterior marginals of the latent field. These approximations have been checked against very long runs of MCMC algorithms and appear to be extremely accurate. There are some cases where the approximations and the MCMC based estimates seem to disagree. We are confident that, in these cases the disagreement is mostly due to some MCMC error which, despite the long run, is still present in the sample.

The problems analysed in this report present a higher dimension of the hyperparameter vector $\boldsymbol{\theta}$ than those in Rue and Martino (2006) and Rue et al. (2007). Hence the grid integration scheme used in Rue and Martino (2006) and Rue et al. (2007) becomes too computationally expensive. We have, therefore, used a different integration procedure, named central composit design (CCD). This was introduced in Rue et al. (2007) but in this report we verify that in most cases it gives accurate results, despite the fact that the hyperparameter space is explored in a much cruder way.

In all examples considered here, we consider bivariate data and model latent field as a bivariate autoregressive model of order 1. It is, in principle, possible to generalise this model by allowing higher dimension of the data set and higher order of the autoregressive model. However, this would make not only the number of hyperparameters to increase, but also the structure of the precision matrix of the latent field to become more dense. This means, in turn, that the efficiency of INLA decreases. Anyway, efficiency problems would be present, for such complex models, also for MCMC based inference.

Computing approximations for the posterior marginals of hyperparameters $\pi(\theta_m|\boldsymbol{y})$, $m = 1, \ldots, M$ becomes harder when M grows. In this report we propose different solutions to this problem. There seems to be no real method to obtain accurate approximations for $\pi(\theta_m|\boldsymbol{y})$ in a cheap way. If accuracy $\widetilde{\pi}(\theta_m|\boldsymbol{y})$ is required, some additional computational time has to be invested in this task. Anyway,we describe fast solutions which give useful, though not extremely accurate, results.

Using INLA also the issue of model choice can be solved. An approximation for the marginal likelihood of the model can easily be derived and, for the class of models discussed here, the Bayes factor can be used for model comparison.

References

Andersen, T., Chung, H., and Sorensen, B. (1999). Efficient method of moments estimation of a stochastic volatility model: a monte carlo stody. *Journal of Econometrics*, 91:61–87.

Andersen, T. and Sorensen, B. (1996). Gmm estimation of stochastic volatility model: a monte carlo study. *Journal of Business and Economic Statistics*, 14:329–352.

Bauwens, L., Laurent, S., and Rombouts, J. (2006). Multivariate garch: A survey. *Journal of Applied Econometrics*, 21:79–109.

Black, F. (1976). Studies of stock market volatility changes. In *Proceedings of the American Statistical Association, Business and Economic Statistics Section*, pages 177–181.

Bollerslev, T. (1990). Modelling the coherence in shirt-run nominal exchange rates: a multivariate generalized arch approach. *Review of economics and statistics*, 72:498–505.

Box, G. E. P. and Wilson, K. B. (1951). On the experimental attainment of optimum conditions (with discussion). *Journal of the Royal Statistical Society, Series B*, 13(1):1–45.

Campbell, J. Y., Lo, A. W., and MacKilnay, A. C. (1997). *The econometrics of financial markets*. Princeton University press, Princeton, NJ.

Chib, S., Nardari, F., and Shepard, N. (2002). Markov chain Monte Carlo methods for stochastic volatility models. *Journal of Econometrics*, 108:281–316.

Christie, A. (1982). The stochastic behaviour of common stock variances: values leverage and interest rates effects. *Journal of Financial Economics*, 10:407–432.

Clyde, M. and George, E. I. (2004). Model uncertainty. *Statistical Science*, 19(1):81–94.

Cressie, N. A. C. (1993). *Statistics for spatial data*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. John Wiley & Sons Inc., New York. Revised reprint of the 1991 edition, A Wiley-Interscience Publication.

Danielsson, J. (1994). Multivariate stochastic volatility models: estimation with simulated maximum likelihood. *Journal of Econometrics*, 64:375–400.

Danielsson, J. (1998). Multivariate stochastic volatility models: estimation and comparison with vgarch models. *Journal of empirical finance*, 5:155–173.

Durbin, J. and Koopman, S. J. (1997). Monte Carlo maximum likelihood estimation for non-Gaussian state space models. *Biometrika*, 84(3):669–684.

Durbin, J. and Koopman, S. J. (2000). Time series analysis of non-Gaussian observations based on state space models from both classical and Bayesian perspectives (with discussion). *Journal of the Royal Statistical Society, Series B*, 62(1):3–56.

Eidsvik, J., Martino, S., and Rue, H. (2006). Approximate bayesian inference in spatial generalized linear mixed models. Statistics Report No. 2, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway.

Harvey, A. C., Ruiz, E., and Shephard, N. (1994). Multivariate stochastic variance models. *Review of economic studies*, 61:247–264.

Harvey, A. C. and Shephard, N. (1996). Estimation of an asymmetric stochastic volatility model for asset returns. *Journal of Business and Economic Statistics*, 14:429–434.

Hsiao, C. K., Huang, S. Y., and Chang, C. W. (2004). Bayesian marginal inference via cadidate's formula. *Statistics and Computing*, 14(1):59–66.

Ihaka, R. and Gentleman, R. (1996). R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299–314.

Jacquier, E., Polson, N., and Rossi, P. (2004). Bayesian analysis of stochastic volatility models with fat-tails and correlated errors. *Journal of Econometrics*, 122:185–212.

Jeffreys, H. (1961). *The theory of probability*. Oxford press.

Kadane, J. B. and Lazar, N. A. (2004). Methods and criteria fom model selection. *Journal of American Statistical Association*, 99(465):279–290.

Kass, R. E. and Vaidyatnatan, S. (1992). Approximate bayes factors and orthogonal parameters, with application to testing equality of two binomial proportions. *Journal of Royal Statistical Society, Series B*, 54(1):129–144.

Kim, S., Shephard, N., and Chib, S. (1998). Stochastic volatility: likelihood inference and comparison with arch models. *Review of Economic Studies*, 65:361–393.

Knorr-Held, L. and Rue, H. (2002). On block updating in Markov random field models for disease mapping. *Scandinavian Journal of Statistics*, 29(4):597–614.

Mayer, R. and Yu, J. (2000). Bugs for a bayesian analysis of stochastic volatility models. *Econometrics Journal*, 3:198–215.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*, volume 104 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Rue, H. and Martino, S. (2006). Approximate Bayesian inference for hierarchical Gaussian Markov random fields models. *Journal of Statistical Planning and Inference*, 137:3177–3192.

Rue, H., Martino, S., and Chopin, N. (2007). Approximate bayesian inference for latent gaussian models using integrated nested laplace approximations. Statistics Report No. 1, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway.

Rue, H., Steinsland, I., and Erland, S. (2004). Approximating hidden Gaussian Markov random fields. *Journal of the Royal Statistical Society, Series B*, 66(4):877–892.

Ruppert, D. (2004). *Statistics and Finance. An Introduction*. Springet texts in Statistics. Springer, New-York.

Sanchez, S. M. and Sanchez, P. J. (2005). Very large fractional factorials and central composite designs. *ACM Transactions on Modeling and Computer Simulation*, 15(4):362–377.

Shephard, N. and Pitt, M. K. (1997). Likelihood analysis of non-Gaussian measurement time series. *Biometrika*, 84(3):653–667.

Spiegelhalter, D. J., Thomas, A., Best, N. G., and Wilks, W. R. (2003). *WinBUGS User Manual (Version 1.4)*. MRC Biostatistics Unit, Cambridge, UK.

Taylor, S. J. (1986). *Modelling stochastic volatility*. John Wiley, Chichester.

Tierney, L. and Kadane, J. B. (1986). Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86.

Yu, J. (2005). On leverage in a stochastic volatility mdoels. *Journal of Econometrics*, 127:165–178.

Yu, J. and Mayer, R. (2006). Multivariate stochastic volatility models: Bayesian estimation and models comparison. *Econometric Reviews*, 25.

A Appendix

A.1 Linear expansion of $\log \pi_{GG}(\boldsymbol{x}_{-t}|x_t, \boldsymbol{\theta}_k)$

In a unidimensional problem, the log denominator of expression (22) is given by

$$\log \widetilde{\pi}_{GG}(\boldsymbol{x}_{-t}|x_t, \boldsymbol{\theta}_k)\Big|_{\boldsymbol{x}_{-t}=\mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-t}|x_t,\boldsymbol{\theta}_k)} \propto \frac{1}{2}\log|\boldsymbol{Q}^* + \mathrm{diag}\{\boldsymbol{c}(x_t, \boldsymbol{\theta}_k)\}| \tag{36}$$

where $\boldsymbol{Q}^*$ is the prior precision matrix of the GMRF $\boldsymbol{x}$ where the row and column number t have been removed, and $\boldsymbol{c}(x_t, \boldsymbol{\theta}_k)$ is the vector of minus the second derivative of the l0g-likelihood evaluated at $x_j = \mathrm{E}_{\widetilde{\pi}_G}(x_j|x_t, \boldsymbol{\theta}_k)$, that is:

$$c_j(x_t, \boldsymbol{\theta}_k) = -\frac{\partial^2 \pi(y_j|x_j, \boldsymbol{\theta}_k)}{\partial x_j^2}\Big|_{x_j=\mathrm{E}_{\widetilde{\pi}_G}(x_j|x_t,\boldsymbol{\theta}_k)}$$

Let δ^t indicate the derivative of the conditional mean $\mathrm{E}_{\widetilde{\pi}_G}(x_j|x_t, \boldsymbol{\theta}_k)$, then each x_j can be written as a function of x_t as

$$x_j = \mu_{G_j(\boldsymbol{\theta}_k)} + \delta_j^t(x_t - \mu_{G_t}(\boldsymbol{\theta}_k))$$

where $\boldsymbol{\mu}_G(\boldsymbol{\theta}_k)$ is the mean of the Gaussian approximation $\pi_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}_k)$.

We want to expand expression (36) around $x_t = \mu_{G_t}(\boldsymbol{\theta}_k)$. For this purpose we have to compute its first derivative. Let

$$d_j^3(x_t, \boldsymbol{\theta}_k) = \frac{\partial c_j(\boldsymbol{\theta}_k, x_t)}{\partial x_t} = -\frac{\partial^3 \pi(y_j|x_j, \boldsymbol{\theta}_k)}{\partial x_j^3}\Big|_{x_j=\mathrm{E}_{\widetilde{\pi}_G}(x_j|x_t,\boldsymbol{\theta}_k)} \delta_j^t$$

Since for any matrix $\boldsymbol{M}$ we have that $\partial \log|\boldsymbol{M}| = \mathrm{Trace}(\boldsymbol{M}^{-1}\partial\boldsymbol{M})$, then

$$\begin{array}{rcl} \frac{d\log|\boldsymbol{Q}^*+\mathrm{diag}(\boldsymbol{c})|}{dx_t} & = & \mathrm{Trace}\left\{[\boldsymbol{Q}^* + \mathrm{diag}(\boldsymbol{c})]^{-1}\frac{d[\boldsymbol{Q}^*+\mathrm{diag}(\boldsymbol{c})]}{dx_t}\right\} \\ & = & \mathrm{Trace}\left\{[\boldsymbol{Q}^* + \mathrm{diag}(\boldsymbol{c})]^{-1}\mathrm{diag}[\boldsymbol{d}^3(x_t, \boldsymbol{\theta}_k)]\right\} \\ & = & \sum_j \mathrm{Var}(x_j|x_t) d_j^3(x_t, \boldsymbol{\theta}_k) \\ & = & \sum_j \sigma_{G_j}(\boldsymbol{\theta}_k)[1 - \mathrm{Corr}^2_{\pi_G}(x_t, x_j|\boldsymbol{\theta}_k)]\, d_j^3(x_t, \boldsymbol{\theta}_k) \end{array} \tag{37}$$

We have then

$$\log \widetilde{\pi}_{GG}(\boldsymbol{x}_{-t}|x_t, \boldsymbol{\theta}_k)\Big|_{\boldsymbol{x}_{-t}=\mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-t}|x_t,\boldsymbol{\theta}_k)} \approx \frac{1}{2}\, x_t \, \sum_j \sigma_{G_j}(\boldsymbol{\theta}_k)[1 - \mathrm{Corr}^2_{\pi_G}(x_t, x_j|\boldsymbol{\theta}_k)]\, d^3_j(x_t, \boldsymbol{\theta}_k) \tag{38}$$

Note that the correlation between x_j and x_t, necessary to compute (38) is only available for some of the i's and t's since he marginal variances are computed using (11). The solution to this problem given by Rue et al. (2007) is to simply replace all non computed correlations with a default value, say 0.05.

For Gaussian data (36) is just a constant, so the term in (38) is the first order correction for non-Gaussian observations.

The first order expansion presented here depends from the fact that the matrix diag$\{\boldsymbol{c}\}$ is a diagonal matrix. The corresponding matrix for multidimensional models diag$\{\boldsymbol{C}\}$, defined in Section 3.1, instead, includes also some off diagonal terms, these make the computation of the derivative in (37) much more complex.

A.2 Determinant of $\boldsymbol{Q}_{[-i,-i]}$

For any GMRF $\boldsymbol{x}$, with precision matrix $\boldsymbol{Q}$ we have that

$$\pi(\boldsymbol{x}) \propto |\boldsymbol{Q}|^{1/2} \exp\{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}\} \tag{39}$$

From the basic properties of a Gaussian distribution we have that, for any index $i = 1, \ldots, n$, the precision matrix of $x_{-i}|x_i$ is $\boldsymbol{Q}_{[-i,-i]}$. Moreover we have that

$$\pi(\boldsymbol{x}) = \pi(x_i)\pi(\boldsymbol{x}_{-i}|x_i) \propto \mathrm{Var}(x_i)^{-1/2}|\boldsymbol{Q}_{[-i,-i]}|^{1/2} \exp\{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}\} \tag{40}$$

Comparing (39) and (40) we have that

$$\frac{1}{2}\log|\boldsymbol{Q}_{[-i,-i]}| = \frac{1}{2}\log|\boldsymbol{Q}| + \frac{1}{2}\log \mathrm{Var}(x_i)$$

A.3 Likelihood for asymmetric SV models

We can rewrite model (33) as

$$\begin{aligned} y_t &= \exp(x_t/2)\epsilon_t, \\ x_{t+1} &= \mu + \phi(x_t - \mu) + \sigma(\rho\epsilon_t + \sqrt{1-\rho^2}\omega_{t+1} \end{aligned}$$

with ω_{t+1} being a standard Gaussian and $\mathrm{Corr}(\epsilon_t, \omega_{t+1}) = 0$.

We want to compute the density $\pi(y_t|x_t, x_{t+1}, \boldsymbol{\theta})$. To start, notice that given the values of y_t and x_t, then $\epsilon_t = \exp(-x_t/2)\, y_t$ and

$$x_{t+1} = \mu + \phi(x_t - \mu) + \sigma \exp(-x_t/2) y_t + \sigma\sqrt{1-\rho^2}\omega_{t+1}$$

that is

$$x_{t+1}|x_t, y_t, \boldsymbol{\theta} \sim \mathcal{N}(\mu + \phi(x_t - \mu) + \sigma \exp(-x_t/2) y_t, \sigma\sqrt{1-\rho^2}). \tag{41}$$

Moreover we have

$$y_t|x_t\boldsymbol{\theta} \sim \mathcal{N}(0, \exp(x_t)). \tag{42}$$

We can write

$$\begin{aligned}
\pi(y_t|x_t, x_{t+1}, \boldsymbol{\theta}) &\propto \pi(y_t, x_t, x_{t+1}|\boldsymbol{\theta}) \\
&\propto \pi(x_t|\boldsymbol{\theta})\pi(y_t|x_t, \boldsymbol{\theta})\pi(x_{t+1}|x_t, y_t, \boldsymbol{\theta}) \\
&\propto \pi(y_t|x_t, \boldsymbol{\theta})\pi(x_{t+1}|x_t, y_t, \boldsymbol{\theta})
\end{aligned}$$

From (41) and (42) we have then

$$\begin{aligned}
\pi(y_t|x_t, x_{t+1}, \boldsymbol{\theta}) &\propto e^{x_t/2} \exp\left\{-\tfrac{e^{-x_t/2}}{2} y_t^2\right\} \exp\left\{-\tfrac{1}{2\sigma^2(1-\rho^2)}[x_{t+1} - \mu - \phi(x_t - \mu) - \sigma\rho e^{x_t/2} y_t]\right\} \\
&\propto \exp\left\{-\tfrac{1}{2}\left[e^{-x_t} + \tfrac{\rho^2}{1-\rho^2} e^{-x_t}\right] y_t^2 + [x_{t+1} - \mu - \phi(x_t - \mu)]\, \tfrac{\rho e^{-x_t/2}}{\sigma(1-\rho^2)} y_t\right\}
\end{aligned}$$

which is the core of a Gaussian density with

$$\text{Var}(y_t|x_t, x_{t+1}, \boldsymbol{\theta}) = \left[e^{-x_t} + \frac{\rho^2}{1-\rho^2} e^{-x_t}\right]^{-1} = (1-\rho^2) e^{x_t}$$

and

$$\begin{aligned}
\text{E}(y_t|x_t, x_{t+1}, \boldsymbol{\theta}) &= [x_{t+1} - \mu - \phi(x_t - \mu)]\, \frac{\rho e^{-x_t/2}}{\sigma(1-\rho^2)} \left[e^{-x_t} + \frac{\rho^2}{1-\rho^2} e^{-x_t}\right]^{-1} \\
&= [x_{t+1} - \mu - \phi(x_t - \mu)]\, \frac{\rho}{\sigma} e^{x_t/2}
\end{aligned}$$

Paper V

Implementing Approximate Bayesian Inference using Integrated Nested Laplace Approximation: a manual for the `inla` program.

Implementing Approximate Bayesian Inference using Integrated Nested Laplace Approximation: a manual for the `inla` program

Sara Martino and Håvard Rue
Department of Mathematical Sciences
NTNU, Norway

Abstract

This manual describes the `inla` program, a new instrument which allows the user to easily perform approximate Bayesian inference using integrated nested Laplace approximation (INLA). We describe the set of models which can be solved by the `inla` program and provide a series of worked out examples illustrating its usage in details. The Appendix contains a reference manual for the `inla` program.

1 Introduction

Integrated nested Laplace approximation (INLA) is a new approach to statistical inference for latent Gaussian Markov random field (GMRF) models introduced by Rue and Martino (2006) and Rue et al. (2007). It provides a fast, deterministic alternative to Markov chain Monte Carlo (MCMC) which, at the moment, is the standard tool for inference in such models. The main advantage of the INLA approach over MCMC is that it is much faster to compute; it gives answers in minutes and seconds where MCMC requires hours and days. The theory behind INLA is thoroughly described in Rue et al. (2007) and will not be repeated here.

In short, a latent GMRF model is a hierarchical model where, at the first stage we find a distributional assumption for the observables $\boldsymbol{y}$ usually assumed to be conditionally independent given some latent parameters $\boldsymbol{\eta}$ and, possibly, some additional parameters $\boldsymbol{\theta}_1$

$$\pi(\boldsymbol{y}|\boldsymbol{\eta}, \boldsymbol{\theta}_1) = \prod_j \pi(y_j|\eta_j, \boldsymbol{\theta}_1).$$

The latent parameters $\boldsymbol{\eta}$ are part of a larger latent random field $\boldsymbol{x}$, which constitutes the second stage of our hierarchical model. The latent field $\boldsymbol{x}$ is modelled as a GMRF with

precision matrix $\boldsymbol{Q}$ depending on some hyperparameters $\boldsymbol{\theta}_2$

$$\pi(\boldsymbol{x}|\boldsymbol{\theta}_2) \propto \exp\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{Q}(\boldsymbol{x}-\boldsymbol{\mu})\}$$

The third, and last, stage of the model consists of the prior distribution for the hyperparameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$.

The INLA approach provides a recipe for fast Bayesian inference using accurate approximations to $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and $\pi(x_i|\boldsymbol{y})$, $i = 0, \ldots, n-1$, i.e. the marginal posterior density for the hyperparameters and the posterior marginal densities for the latent variables. Different types of approximations are available, see Rue et al. (2007) for details. The approximate posterior marginals can then be used to compute summary statistics of interest, such as posterior means, variances or quantiles.

Computational speed is one of the most important components of the INLA approach, therefore special care has to be put in the implementation of the required algorithms. All procedures necessary to perform INLA are efficiently implemented in the GMRFLib library. This an open source library written in (ANSI) C and Fortran which is freely available on the web page `http://www.math.ntnu.no/~hrue/GMRFLib/`.

The `inla` program is a useful tool which allows the user to easily specify and solve a large class of models, using the algorithms in the GMRFLib library, without any need for C programming. The components of the model and the options for the INLA procedures are specified through a `ini` file. The `inla` program reads the `ini` file, then it builds and solves the model returning the required approximate posterior marginal densities and summary statistics.

The class of models which can be solved using the `inla` program is wide, covering *time series models*, *generalised additive models* (Hastie and Tibshirani, 1990), *generalised additive mixed models* (Lin and Zhang, 1999), *geoadditive models* (Kammand and Wand, 2003), *univariate volatility models* (Taylor, 1986). With the exception of univariate volatility models, the `inla` program supports a subset of the models supported by BayesX. BayesX is a software tool, developed in the University of Munich, for estimating structured additive regression models, Brezger et al. (2003).

In this tutorial we present the `inla` program and, through a series of worked out examples show the possible range of applications where approximate Bayesian inference using INLA can be useful. In Section 2 we discuss the class of models which can be defined and solved using the `inla` program. In Section 3 we describe the use of the `inla` program through a series of worked out examples of increasing complexity. The examples include all, but one, examples in Rue and Held (2005) and all examples in Rue et al. (2007), plus some more examples previously analysed with BayesX. The Appendix consists of a reference manual for the `inla` program.

2 Model description

The `inla` program supports hierarchical GMRF models of the following type

$$y_j|\eta_j, \boldsymbol{\theta}_1 \sim \pi(y_j|\eta_j, \boldsymbol{\theta}_1) \qquad j \in J \tag{1}$$

$$\eta_i = \sum_{k=0}^{n_f-1} f_k(c_{ki}) + \boldsymbol{z}_i^T \boldsymbol{\beta} + u_i \qquad i = 0, \ldots, n_\eta - 1 \tag{2}$$

where

- J is a subset of $\{0, 1, \ldots, n_\eta - 1\}$, that is, not necessarily all latent parameters $\boldsymbol{\eta}$ are observed through the data $\boldsymbol{y}$.
- $\pi(y_j|\eta_j, \boldsymbol{\theta}_1)$ is the likelihood of the observed data assumed to be conditional independent given the latent parameters $\boldsymbol{\eta}$, and, possibly, some additional parameters $\boldsymbol{\theta}_1$. The latent variable η_i enters the likelihood through a known link function, see Appendix A.1 for details.
- $\boldsymbol{u}$ is a vector of unstructured random effects of length n_η with i.i.d Gaussian priors with precision parameter λ_η:

$$\boldsymbol{u}|\lambda_\eta \sim \mathcal{N}(\boldsymbol{0}, \lambda_\eta \boldsymbol{I}) \tag{3}$$

- $f_k(c_{ki})$ is the effect of a generic covariate k which assumes value c_{ki} for observation i. The functions f_k, $k = 0, \ldots, n_f - 1$ comprise usual nonlinear effect of continuous covariates, time trends and seasonal effects, two dimensional surfaces, iid random intercepts and slopes and spatial random effects. The unknown functions, or more exactly the corresponding vector of function evaluations $\boldsymbol{f}_k = (f_{0k}, \ldots, f_{(m_k-1)k})^T$, are modelled as GMRFs given some parameters $\boldsymbol{\theta}_{f_k}$, that is

$$\boldsymbol{f}_k|\boldsymbol{\theta}_{f_k} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_k^{-1}) \tag{4}$$

- $\boldsymbol{z}_i$ is a vector of n_β covariates assumed to have a linear effect, and is $\boldsymbol{\beta}$ the corresponding vector of unknown parameters with independent zero-mean Gaussian prior with fixed precisions.

The full latent field, of dimension $n = n_\eta + \sum_{j=0}^{n_f-1} m_j + n_\beta$, is then

$$\boldsymbol{x} = (\boldsymbol{\eta}^T, \boldsymbol{f}_0^T, \ldots, \boldsymbol{f}_{n_f-1}^T, \boldsymbol{\beta}^T).$$

All elements of vector $\boldsymbol{x}$ are defined as GMRFs, hence $\boldsymbol{x}$ is itself a GMRF with density:

$$\pi(\boldsymbol{x}|\boldsymbol{\theta}_2) = \prod_{i=0}^{n_\eta-1} \pi(\eta_i|\boldsymbol{f}_0, \ldots, \boldsymbol{f}_{n_f-1}, \boldsymbol{\beta}, \lambda_\eta) \prod_{k=0}^{n_f-1} \pi(\boldsymbol{f}_k|\boldsymbol{\kappa}_{f_k}) \prod_{m=0}^{n_\beta-1} \pi(\beta_m) \tag{5}$$

where $\boldsymbol{\theta}_2 = \{\lambda_\eta, \boldsymbol{\theta}_{f_0}, \ldots, \boldsymbol{\theta}_{n_f-1}\}$ is a vector of unknown hyperparameters.

The last element in the definition of our hierarchical model is a prior distribution for the hyperparameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$. In the `inla` function all precisions are given a Gamma prior with parameters a and b so that the mean is a/b and the variance is a/b^2. See the Appendix for details about the prior distributions for all the hyperparameters of the model.

Many well known models from the literature can be written as special cases of (1) and (2)

- *Time series models*

  Time series models are obtained if $c_k = t$ represents time. The functions f_k can model nonlinear trends or seasonal effects

  $$\eta_t = f_{trend}(t) + f_{seasonal}(t) + \boldsymbol{z}_t^T \boldsymbol{\beta}$$

- *Generalised additive models (GAM)*

  A GAM model is obtained if $\pi(y_i|\eta_i, \boldsymbol{\theta}_l)$ belongs to an exponential family, c_k are univariate, continuous covariates and f_k are smooth functions.

- *Generalised additive mixed models (GAMM) for longitudinal data*

  Consider longitudinal data for individuals $i = 0, \ldots, n_i - 1$, observed at time points $t_0, t_1, \ldots$. A GAMM model extends a GAM by introducing individual specific random effects, i.e.

  $$\eta_{it} = f_0(c_{it0}) + \ldots, + f_{n_f-1}(c_{it(n_f-1))}) + b_{0i} w_{it0} + \cdots + b_{(n_b-1)i} w_{it(n_b-1)}$$

  where η_{it} is the predictor for individual i at time t, x_{itk}, $k = 0, \ldots, n_f - 1$,w_{itq}, $q = 0, \ldots, n_b - 1$ are covariate values for individual i at time t, and $b_{0i}, \ldots, b_{(n_b-1)i}$ is a vector of n_b individual specific random intercepts (if $w_{itq} = 1$) or slopes. The above model can be written in the general form in equation (2) by defining $r = (i, t)$, $c_{rj} = c_{itj}$ for $j = 0, \ldots, n_f - 1$ and $c_{r,(n_f-1)+q} = w_{itq}$, $f_{(n_f-1)+q}(c_{r,(n_f-1)+q}) = b_{qi} w_{itq}$ for $q = 0, \ldots, n_b$. In the same way GAMM's for cluster data can be written in the general form (2).

- *Geoadditive models*

  If geographical information for the observations in the data set are available, they might be included in the model as

  $$\eta_i = f_1(c_{0i}) + \cdots + f_{n_f-1}(c_{(n_f-1)i}) + f_{spat}(s_i) + \boldsymbol{z}_i^T \boldsymbol{\beta}$$

  where s_i indicates the location of observation i and f_{spat} is a spatially correlated effect. Models where one of the covariate represent the spatial effect have recently been coined geoadditive by Kammann and Wand (2003).

- *ANOVA type interaction model*

  The effect of two continuous covariate w and v can be modelled as

  $$\eta_i = f_1(w_i) + f_2(v_i) + f_{1|2}(w_i, v_i) + \dots$$

  where f_1 and f_2 are the main effects of the two covariates and $f_{1|2}$ is a two dimensional interaction surface. The above model can be written in the general form (2) simply by defining $c_{1i} = w_i$, $c_{2i} = v_i$, $c_{3i} = (w_i, v_i)$,

- *Univariate stochastic volatility model*

  Stochastic volatility models are time series models with Gaussian likelihood where it is the variance, and not the mean of the observed data, to be part of the latent GMRF model. That is

  $$y_i|\eta_i \sim \mathcal{N}(0, \exp(\eta_i))$$

  The latent field is then typically modelled as a autoregressive model of order 1.

3 Examples of application

In this section we present a series of worked out examples mostly taken from Rue and Held (2005), Rue et al. (2007) and from the BayesX web page. The aim is both to show the wide range of models which can be solved using the approximate Bayesian inference techniques presented in Rue et al. (2007), and to introduce the `inla` program which makes it possible for the user to apply the above mentioned approximation techniques, making use of the GMRFLib library, in an easy and painless way.

The only input required from the `inla` program is a `ini` file containing the description of the model, the location of the files where the data and the covariates are stored, and, possibly, some options to be passed to the underlying GMRFLib library. The `ini` file is organised in sections each of which either describes one element of the hierarchical model in equations (1) and (2), or specifies some global parameters for the underlying functions in the GMRFLib library. The user is required to specify the likelihood model for the data, the parameters for the prior distribution of the model hyperparameters $\boldsymbol{\theta}$, and to describe, one by one, all components of the latent GMRF $\boldsymbol{x}$ in (2). The `inla` program will then read the model specifications, build the joint probability distribution for the latent GMRF $\boldsymbol{x}$ in equation (5), compute approximations for the required posterior marginals and store the results in a user defined directory.

Before presenting the examples, we describe how the covariate values are stored in files. Each covariate has to be stored in a separate file. The format of the file depends on whether the covariate is assumed to have linear or non-linear effect:

Covariates with linear effect: The value of the covariate is simply stored in a file with n_η columns each row having the format:

$$i \quad z_i$$

where $i = 0, \ldots, n_\eta - 1$ and z_i is the value of the covariate for observation i.

Covariates with non-linear effect: Let $c \in \boldsymbol{C}$ and $\boldsymbol{C} = \{c^{(0)} < c^{(1)} < \cdots < c^{(idx)} < \cdots < c^{(m-1)}\}$. That is, covariate c takes one of the m values in the ordered vector $\boldsymbol{C}$. The file storing covariate c has n_η row, each with the following format:

$$i \quad (idx)_i$$

where $i = 0, \ldots, n_\eta - 1$ and $(idx)_i$ is the position of the observed value c_i in the vector $\boldsymbol{C}$. If the values in $\boldsymbol{C}$ are different from $0, 1, \ldots$, another file of m rows, is necessary to store the values of $\boldsymbol{C}$. A short example will be useful:

Example: *Let $n_\eta = 5$ and $\boldsymbol{C} = \{9, 10, 11\}$. Let the observed covariate values be $c_0 = 10$, $c_1 = 9$, $c_2 = 11$, $c_3 = 9$ and $c_4 = 10$. Then the covariate file will be as following*

$$\begin{array}{cc} 0 & 1 \\ 1 & 0 \\ 2 & 2 \\ 3 & 0 \\ 4 & 1 \end{array}$$

We would need also a file storing the values in $\boldsymbol{C}$:

$$\begin{array}{c} 9 \\ 10 \\ 11 \end{array}$$

Note that all indexes go from 0 to $n - 1$ and not from 1 to n.

We run each example in Section 3.1 on two different machines. The first, defined Machine 1, is a laptop with a Intel(R) Pentium(R) M processor 1.86GHz. The second one, defined Machine 2 is a Dell Poweredge 2950 equipped with two quad-core Itel Xeon 2.66GHz CPUs. For each of the examples we describe the model, the corresponding `ini` file and report some output results and the computation time for each of the two machines.

3.1 A simple time series: the Tokyo rainfall data

Our first example is a simple time series model, analysed, among others, in Rue and Held (2005, Sec. 4.3.4).

Example 1 *The number of occurrences of rainfall over 1 mm in the Tokyo area for each calendar year during two years (1983-84) are registered. It is of interest to estimate the underlying probability p_t of rainfall for calendar day t which is, apriori, assumed to change gradually over time. The likelihood model is binomial*

$$y_t|\eta_t \sim \textit{Bin}(n_t, p_t)$$

with logit link function

$$p_t = \frac{\exp(\eta_t)}{1+\exp(\eta_t)}.$$

The model for the latent variables can be written in the general form of equation (2) as

$$\eta_t = f(t)$$

where t is the observed time whose effect is modelled as a smooth function $f(\cdot)$. Following Rue and Held (2005), the random vector $\boldsymbol{f} = \{f_0, \ldots, f_{365}\}$ is assumed to have a circular random walk of order 2 (RW2) prior with unknown precision λ_f.

There is only one hyperparameter $\boldsymbol{\theta} = (\lambda_f)$ which we assign a Gamma(a, b) prior distribution with $a = 1$ and $b = 0.0001$.

Figure 1, panel (a), displays the observed frequencies of rain for the 366 time points. The `TOKYO.ini` file which defines the above model for the `inla` program is:

```
[The Tokyo-rainfall example]
type = problem
dir = results

[Unstruct-term]
type = unstruct
initial = 10
fixed = 1
n = 366

[data]
type = data
likelihood = binomial
filename = tokyo.rainfall.data

[latent-RW2]
type = ffield
covariates = time.covariate
n=366
model = rw2
parameters = 1.0  0.0001
cyclic = 1
quantiles =0.025  0.975
```

In the following we guide the reader, section by section, through the above `ini` file and explain what the different fields represent. We then briefly illustrate how to run the `inla` program and how and where the output is stored.

Each section of the `ini` file starts with a tag (in square brackets) which is simply a user defined name for the section itself. The order of the sections is not important. The field named ***type*** is contained in each section. It defines the role of the section in the problem specification and, consequently, determines also the nature of all other fields in the same section. There are six specifications for the ***type*** field, see Appendix A.1 for details.

The first section in our `ini` file, defined by ***type***=*problem*, specifies some global parameters. The options specified in this section are valid for the whole problem. Here, the directory where the results will be stored is defined (line 3).

The second section, defined by ***type***=*unstruct*, (lines 5-9), specifies the model for the unstructured term u_i in equation (2). The field ***n*** is required and indicates the length of the latent variable vector $\boldsymbol{\eta}$. The `inla` program requires a section of ***type***=*unstruct* to always be present, even in cases, like the example we are presenting here, where there is no unstructured random effect. We mimic the absence of unstructured random effect by declaring the precision λ_η to be fixed and not random (***fixed***=*1*), and the value of the log precision $\log \lambda_\eta$ to be high (***initial*** =*10*).

The following section, defined by ***type***=*data* (lines 11-14), specifies the model for the likelihood of the data $\pi(y_t|\eta_t)$ (line 13), and the name of the file where the data are stored (line 14). The format of the data file depends on the likelihood model, see Appendix A.1.2. For binomial likelihood it is as following:

$$t \quad n_t \quad y_t$$

where t is the data index going from 0 to $(n_d - 1) = 365$.

The last section, defined by ***type***= *ffield* (lines 16-23) specifies the model for the random vector $\boldsymbol{f}$. In this example we have a second order random walk (***model***=*rw2*) of length 366 (***n***=*366*) which is cyclical (***cyclic*** =*1*). We also specify here the parameters a and b for the Gamma prior for the precision parameter λ_f (line 21). We require the `inla` program to compute also the 0.025 and 0.975 quantiles for each of the posterior marginal densities in the latent RW2 field (line 23). The name of the file where the covariate values are stored (line 18) completes the model specification. In this case the covariate is just the observed time point. The covariate file consists of two identical columns with index going from 0 to 365.

$$\begin{matrix} 0 & 0 \\ 1 & 1 \\ 2 & 2 \end{matrix}$$

Once the `ini` file is ready, we can run the program using the following command line:

```
inla -v TOKYO.ini
```

The option `-v` (verbose) makes the program print out some more information about the

model while running. Only for this example, we reproduce the output of the `inla` program to make the reader familiar with it.

```
inla_build...
        number of sections=[4]
        parse section=[0] name=[the tokyo-rainfall example]
           type=[PROBLEM]
        inla_parse_problem...
                name=[the tokyo-rainfall example]
                use.derivaties=[1]
                store results in directory=[results]
                output:
                        kld=[1]
                        hyperparameters=[0]
                        summary=[1]
                        density=[1]
                        nquantiles=[0]  [ ]
                        npercentiles=[0]  [ ]
        parse section=[1] name=[unstruct-term] type=[UNSTRUCT]
        inla_parse_unstruct ...
                section=[unstruct-term]
                PRIOR->name=[GAMMA]
                PRIOR->PARAMETERS=[1, 0.001]
                initialise log_precision[10]
                fixed=[1]
                n=[366]
                compute=[0]
                output:
                        summary=[1]
                        density=[1]
                        nquantiles=[0]  [ ]
                        npercentiles=[0]  [ ]
        parse section=[2] name=[data] type=[DATA]
        inla_parse_data...
                tag=[data]
                likelihood=[BINOMIAL]
                file->name=[tokyo.rainfall.data]
                read n=[1098] entries from file=[tokyo.rainfall.data]
                        0/366  (idx,a,y) = (0, 2, 0)
                        1/366  (idx,a,y) = (1, 2, 0)

        parse section=[3] name=[latent-rw2] type=[FFIELD]
        inla_parse_ffield...
                section=[latent-rw2]
                model=[rw2]
                PRIOR->name=[GAMMA]
                PRIOR->PARAMETERS=[1, 0.0001]
                constr=[0]
                diagonal=[0]
                compute=[1]
                fixed=[0]
                read covariates from file=[time.covariate]
                read n=[732] entries from file=[time.covariate]
```

```
                file=[time.covariate] 0/366  (idx,y) = (0, 0)
                file=[time.covariate] 1/366  (idx,y) = (1, 1)

                n=[366]: use default locations, if required
                cyclic=[1]
                initialise log_precision[6.90776]
                output:
                        summary=[1]
                        density=[1]
                        nquantiles=[2]  [ 0.025 0.975 ]
                        npercentiles=[0]  [ ]
inla_build: check for unused entries in[TOKYO.ini]
inla_INLA...
        Size of full graph=[732]
        Found optimal reordering=[amd]
        List of hyperparameters:
                theta[0] = [Log_precision for latent-rw2]
Maximise marginal for hyperparam: log(density) = -331.7258  theta =
   9.335481
Maximise marginal for hyperparam: log(density) = -331.7258  theta =
   9.335732
Compute the Hessian using central differences and step_size[0.0001].
   Matrix-type [dense]
Maximise marginal for hyperparam: log(density) = -331.7258  theta =
   9.335732
         2.609340
Eigenvectors of the Hessian
        1.000000
Eigenvalues of the Hessian
        2.609340
Search: coordinate 0 direction -1
        config  0=[ -1] log(rel.dens)= -0.49, accept, compute, 0.10s
        config  1=[ -2] log(rel.dens)= -1.88, accept, compute, 0.10s
        config  2=[ -3] log(rel.dens)= -3.99, diff to large, stop
           searching
Search: coordinate 0 direction 1
        config  3=[  1] log(rel.dens)= -0.51, accept, compute, 0.10s
        config  4=[  2] log(rel.dens)= -2.13, accept, compute, 0.10s
        config  5=[  3] log(rel.dens)= -5.33, diff to large, stop
           searching
Fill-in computations
Maximise marginal for hyperparam: log(density) = -331.7258  theta =
   9.335732
        config  6=[  0] log(rel.dens)=  0.00, accept, compute, 0.10s
Combine the densities with relative weights:
        config  0/ 5=[ -1.00] weight = 0.614 adjusted weight = 0.616
        config  1/ 5=[ -2.00] weight = 0.152 adjusted weight = 0.171
        config  2/ 5=[  1.00] weight = 0.603 adjusted weight = 0.604
        config  3/ 5=[  2.00] weight = 0.119 adjusted weight = 0.134
        config  4/ 5=[  0.00] weight = 1.000 adjusted weight = 0.963
Done.
        store results in directory[results]
```

```
                store summary results
                    in[results/latent-rw2/summary.dat]
                store summary (gaussian) results
                    in[results/latent-rw2/summary-gaussian.dat]
                store marginals
                    in[results/latent-rw2/marginal-densities.dat]
                store marginal-densities (gaussian)
                    in[results/latent-rw2/marginal-densities-gaussian.dat]
                store (symmetric) kld's
                    in[results/latent-rw2/symmetric-kld.dat]
                store quantiles in[results/latent-rw2/quantiles.dat]
                store quantiles (gaussian)
                    in[results/latent-rw2/quantiles-gaussian.dat]
Time used:
        Preparations     :   0.011 seconds
        Approx inference :   2.361 seconds
        Output           :   2.122 seconds
        ---------------------------------
        Total            :   4.494 seconds
```

From the above output we can follow what the `inla` program does: it first reads the different sections, builds the model for the full latent field x, performs the INLA approximation and, finally, stores the results in the appropriate directories. The whole procedure takes less than 5 seconds on Machine 1 and about 2 seconds on Machine 2.

The results are stored in the the directory *results* . The program creates sub-directories to store separately results for each component of the model. In our Tokyo example we have two sub-directories:

- *unstruct −term/*
- *latent −rw2/*

The first one is an empty directory since by default the marginals for the unstructured term are not computed, see Appendix A.1.3. The second directory contains results for the latent RW2 model. The sub-directories where the results are stored are printed in the last part of the output of the `inla` function.

The default results consist of five files for each sub-directory created, namely:

- *marginal−densities−gaussian.dat*
- *summary−gaussian.dat*
- *marginal−densities.dat*
- *summary.dat*
- *symmetric−kld.dat*

Moreover we have two files containing the quantiles

- *quantiles −gaussian.dat*

- *quantiles . dat*

The names of the files are always the same for each sub-directory created. The files whose names ends with $-$*gaussian.dat* contain results obtained using the Gaussian approximation to approximate the density of $x_t|\boldsymbol{y}, \boldsymbol{\theta}$ (see Rue et al. (2007), Section 3.2.1) while the other files contain results obtained using one of the improved approximations for $x_t|\boldsymbol{y}, \boldsymbol{\theta}$ described in Rue et al. (2007), i.e. the Laplace approximation or its simplified version (default).

The file *symmetric$-$kld.dat* contains the (symmetric) Kullback-Leibler (KL) divergence between the Gaussian and the (simplified) Laplace approximation to the marginal posterior densities, which we have plotted in Figure 1, panel (b). In this example the divergence is larger for the winter months (November to February), when the observed frequencies are lower, but it stays always very low. Rue et al. (2007) propose to use the Kullback-Leibler distance to check the accuracy of the Gaussian approximation.

The "*summary*" files contain the mean and the standard deviation for each posterior density. There is one line for each node in the RW2 model and each line is structured as follows:

$$t \quad \mathrm{E}(x_t|\boldsymbol{y}) \quad \sigma(x_t|\boldsymbol{y})$$

Also in the "*quantiles*" files each line refers to one node and is structured as follows:

$$t \quad p(0) \quad x_t(0) \quad p(1) \quad x_t(1) \ldots$$

where $p(j)$ and $x_t(j)$ are such that Prob$(x_t < x_t(j)|\boldsymbol{y}) = p(j)$, $j = 0, 1, \ldots$. The number of columns in the "*quantiles*" files depends on how many quantile values the user choose to compute. In our example there are 5 columns.

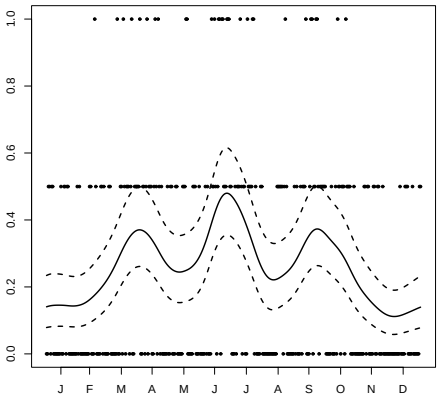
Figure 1, panel (a), displays the binomial frequencies and the approximated posterior mean with uncertainty bounds for the underlying probabilities p_t. The probability of rain is smaller in the winter months.

The "*marginal-densities*" files contain the approximated marginal posterior densities. Again each line refers to a different node in the RW2 model and the structure of each line is as follows

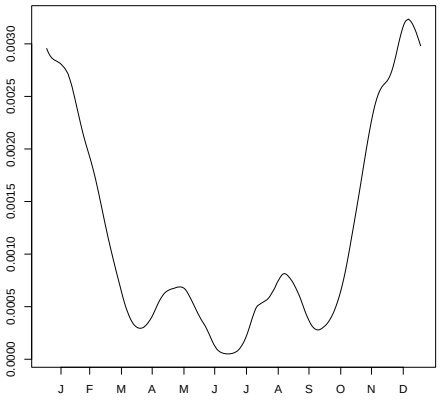
$$t \quad x_{t0} \quad \widetilde{\pi}(x_{t0}|\boldsymbol{y}) \quad x_{t1} \quad \widetilde{\pi}(x_{t1}|\boldsymbol{y}) \quad \ldots \quad x_{tK} \quad \widetilde{\pi}(x_{t(K-1)}|\boldsymbol{y})$$

where $(x_{t0}, x_{t1}, \ldots, x_{t(K-1)})$ are $K = 201$ selected values of the variable x_t and $(\widetilde{\pi}(x_{t0}), \widetilde{\pi}(x_{t1}), \ldots, \widetilde{\pi}(x_{t(K-1)}))$ are the corresponding values of the density. Figure 1 (right) displays the Gaussian approximation (broken line) and the simplified Laplace approximation (solid line) for the marginal posterior density of $x_{365}|\boldsymbol{y}$, this node is chosen for being the one for which the KL divergence is maximised. The following R code can be used to reproduce this figure

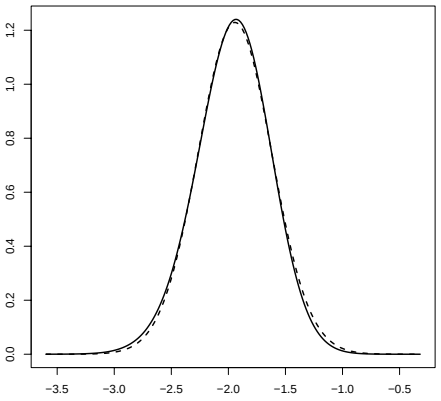
```
>plot(marginal[1,seq(2,403,2)],marginal[1,seq(3,403,2)],
type="l",lwd=2,ylab="",xlab="")
```



(a) Observed frequencies and fitted probabilities with uncertainty bounds



(b) KL-divergence between Gaussian and simplified Laplace approximation for $\pi(x_t|\boldsymbol{y})$



Gaussian (broken line) and simplified Laplace approximation (solid line) for $\pi(x_0|\boldsymbol{y})$

Figure 1: Results for the Tokyo rainfall example

```
>lines(gaus.marginal[1,seq(2,403,2)],gaus.marginal[1,seq(3,403,2)],
type="l",lwd=2,lty=2)
```

where `marginal` and `gaus.marginal` are $n_\eta \times (2K+1)$ matrices containing the data in the files *marginal-densities.dat* and *marginal-densities-gaussian.dat* respectively.

3.2 A time series with seasonal component: the drivers data

The second example is also taken from Rue and Held (2005, Sec 4.4.2). It is again a time series but here we decompose the latent variables η_t into a trend and a seasonal component.

Example 2 *The data consist in monthly counts of car drivers in Great Britain killed or seriously injured in car accidents from January 1969 to December 1984. The time series has $n_d = 192$ data points and exhibits a strong seasonal pattern. One of our goals is to predict the pattern of counts in the 12 month following the last observation.*

We assume the squared root of the counts y_t to be conditionally independent Gaussian random variables:

$$y_t|\eta_t, \lambda_y \sim \mathcal{N}(\eta_t, 1/\lambda_y), \quad t = 0, \ldots, n_d - 1$$

The conditional mean η_t is then a sum of a smooth trend and a seasonal component:

$$\eta_t = \textit{season}_t + \textit{trend}_t, \quad t = 0, \ldots, n_\eta - 1 \tag{6}$$

*We assume the vector **season** $= (\textit{season}_0, \ldots, \textit{season}_{n_\eta - 1})$ to follow the seasonal model in (3.58) of Rue and Held (2005), with length 12 and unknown precision $\lambda_{\textbf{season}}$, and the vector **trend** $= (\textit{trend}_0, \ldots, \textit{trend}_{n_\eta - 1})$ to follow a RW2 with unknown precision $\lambda_{\textbf{trend}}$.*

Note that we have that $n_\eta = n_d + 12 = 204$, since no observations y_t are available for $t = n_d, n_d + 1, \ldots, n_d + 11$. For prediction we are interested in the posterior marginals of $(\eta_{n_d}, \ldots, \eta_{n_d+11})$.

There are three hyperparameters in the model $\boldsymbol{\theta} = (\lambda_y, \lambda_{\textbf{season}}, \lambda_{\textbf{trend}})$ for which we choose the following prior distributions:

$$\begin{array}{lcl} \lambda_y & \sim & \textit{Gamma}(4, 4) \\ \lambda_{\textbf{season}} & \sim & \textit{Gamma}(1, 0.1) \\ \lambda_{\textbf{trend}} & \sim & \textit{Gamma}(1, 0.0005) \end{array}$$

See Rue and Held (2005) for more details.

The corresponding `DRIVERS.ini` file is as follows:

```
[Drivers data]
type = problem
dir = results-%d
quantiles = 0.025 0.975

[Unstruct]
type = unstruct
parameters = 1 0.0005
```

```
9  initial = 13
10 fixed = 1
11 n = 204
12 compute=1
13
14 [data]
15 type = data
16 likelihood = gaussian
17 filename = sqrt-drivers.dat
18 parameters = 4 4
19 initial = -2
20
21 [trend]
22 type = ffield
23 covariates = time.dat
24 n=204
25 model = rw2
26 parameters = 1 0.0005
27 initial = 7
28
29 [seasonal]
30 type = ffield
31 model = seasonal
32 covariates = time.dat
33 n = 204
34 season=12
35 parameters = 1 0.01
36 initial = 10
37
38 [INLA parameters]
39 type = INLA
40 gradient_finite_difference_step_len = 0.001
41 hessian_finite_difference_step_len = 0.001
```

We go briefly through the `ini` file ,section by section, highlighting the difference with the previous example.

- *[Drivers data]* section: specifying the quantiles in ***type***=*problem* section (line 4) , will make the program compute quantiles for all nodes in the latent field.
- *[Unstruct]* section: the precision is fixed to a high value (lines 9-12) to mimic the absence of an unstructured term in the model. Anyway, since our goal is to predict the expected counts we ask the program to compute posterior marginals for $\boldsymbol{\eta}$ as well (***compute***=*1*). Note that, even though in this section a model for the unstructured term u_i is specified, when the ***compute*** flag is turned on, the `inla`

program computes posterior marginals for vector $\boldsymbol{\eta}$. That is for the nodes in latent field which are directly linked to the observables $\boldsymbol{y}$.

- *[data]* section: for Gaussian likelihood the data file has the following format

$$t \quad w_t \quad y_t$$

where w_t are fixed weights, see Appendix A.1.2. Note that in this example the length of the observed data (194) is smaller than the length of the latent variables vector $\boldsymbol{\eta}$ (204).

- *[trend]* section: defines the RW2 model for the trend component. At line 26 we also define a starting value for $\log \lambda_{trend}$ for the optimiser.

- *[seasonal]* section: defines the model for the seasonal component of the model, the parameter ***season*** at line 34 defines the season length

- *[INLA parameters]*: this is an optional section, defined by ***type***=*INLA*, which specifies some parameters to be passed to the GMRFLib library, in this case we specify the step length for the numerical computation of the gradient and the Hessian of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ at its mode, see Appendix for details.

Building and solving the model takes about 10 seconds on Machine 1 and about 3 seconds on Machine 2.

Figure 2 displays the observed and expected counts in the squared root scale (together with 0.025 and 0.975 quantiles).

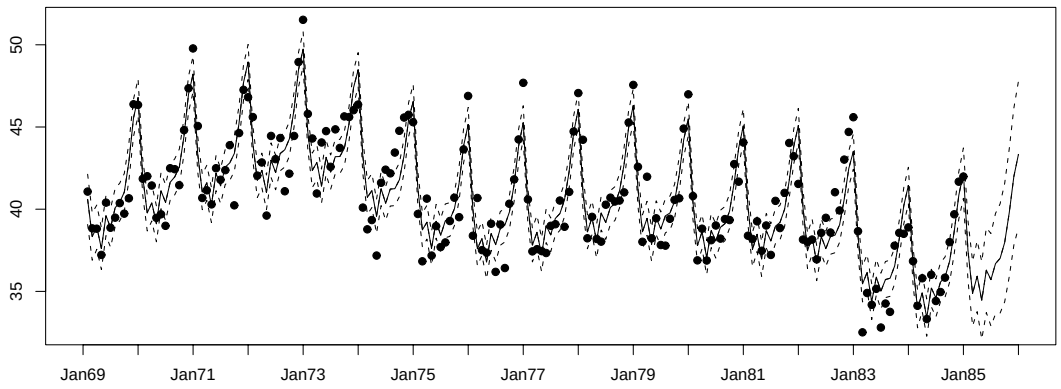


Figure 2: Observed and predicted counts (posterior mean within 0.025 and 0.975 quantiles) for the drivers data example without the seat belt covariate

We consider now a slight modification of Example 2 as discussed by Rue and Held (2005, Sec 4.2.2):

Example 2 cont. *On January 1983 wearing seat belt became compulsory. To check whether this law had an effect on the number of serious accidents we modify the model as follows:*

$$\eta_t = \begin{cases} \textit{season}(t) + \textit{trend}(t) & t = 0, \ldots, 168 \\ \textit{season}(t) + \textit{trend}(t) + \beta & t = 169, \ldots, 204. \end{cases}$$

We assign additional parameter β a Gaussian distribution with 0 precision, equivalent to a flat prior.

Modifying the `DRIVERS.ini` file to account for the extended model is really easy; it is enough to add a new section as below:

```
[belt]
type=linear
covariates = belt.dat
precision=0
```

The ***type***=*linear* parameter specifies that the new covariate has a lines effect, the file *belt.dat* is as follows

$$\begin{array}{cc} 0 & 0 \\ \vdots & \vdots \\ 168 & 0 \\ 169 & 1 \\ \vdots & \vdots \\ 203 & 1 \end{array}$$

Figure 3 displays the approximate posterior marginal density for β together with 0.025 and 0.975 quantiles. The 95% confidence region is well below 0 indicating a significant effect of the seat belt law in reducing the number of dead or injured drivers. Finally, the observed and expected counts in the squared root scale (together with 0.025 and 0.975 quantiles) for the model with the seat belt covariate are displayed in Figure 4, a slightly better fit of this model before and after January 1983 is visible.

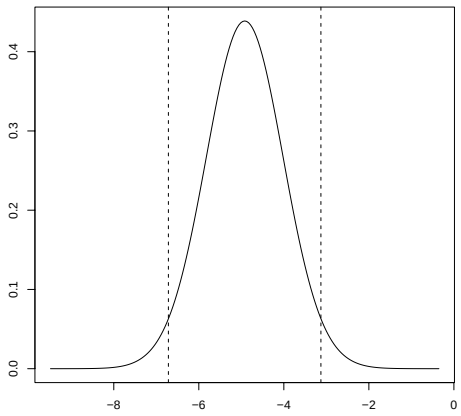


Figure 3: Approximate posterior marginal for parameter β with 0.025 and 0.975 quantiles

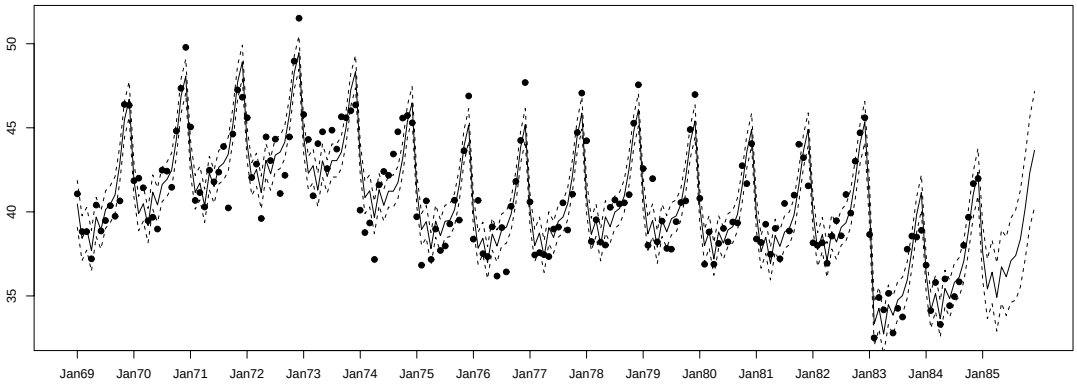


Figure 4: Observed and predicted counts (posterior mean within 0.025 and 0.975 quantiles) for the drivers data example with seat belt covariate

3.3 Stochastic volatility models

Stochastic volatility models are common models in financial time series analysis, lately much interest has been shown in developing efficient MCMC methods for such models, e.g. Shephard and Pitt (1997) and Chib et al. (2002). In the following example, we show how easily a univariate stochastic volatility model can be solved using the `inla` program. The example is taken from Rue et al. (2007) but the model is slightly modified here.

Example 3 *The data consist in 945 observed logarithms of the daily difference of the dollar-pound exchange rate from October 1st, to June 28th, 1985. The data are displayed in Figure 5, panel (a). We analyse this data set using a univariate stochastic volatility model (Taylor, 1986). The likelihood of the data, conditional on the latent variables is:*

$$y_t|\eta_t \sim \mathcal{N}(0, \exp(\eta_t)), \quad t = 0, \ldots, n_d - 1$$

and the model for the latent variables:

$$\eta_t = \mu + f_t \quad t = 0, \quad , n_\eta - 1$$

where μ is an unknown common mean with vague Gaussian prior and $\boldsymbol{f} = (f_0, \ldots, f_{n_\eta - 1})$ is modelled as an auto regressive process of order 1 (AR1) with persistence parameter $\phi \in (-1, 1)$ to ensure stationarity, and precision parameter λ_f.

The model has two hyperparameters, (λ_f, ϕ). We re-parametrise the persistence parameter ϕ as

$$\kappa = \textit{logit}\left(\frac{\phi + 1}{2}\right)$$

and assign the following prior distributions

$$\lambda_f \sim \textit{Gamma}(1, 0.0005)$$
$$\kappa \sim \mathcal{N}(0, 1/0.0001)$$

The `VOLATILITY.ini` file defining the model is the following:

```
[Standard Volatility]
type = problem
dir = results-%d

[Unstruct term]
type = unstruct
n = 1001
initial = 13
fixed = 1
compute=1

[Data]
```

```
13 type = data
14 likelihood = stochvol
15 filename = poundd.dat
16
17 [AR1]
18 type = ffield
19 model = ar1
20 covariates=time.dat
21 n=1001
22 prior0=gamma                    ;prior for the precision
23 initial0=3                      ;initial value for the log-precision
24 parameters0 = 1.0 0.0005        ;parameters for the Gamma prior of the
       precision
25
26 prior1=gaussian                 ;prior for \kappa
27 initial1=4                      ;initial value for \kappa
28 parameters1 = 0 0.0001          ;paramters for the Gaussian prior of
     \kappa
29
30 [Common mean]
31 type=linear
```

The likelihood for the stochastic volatility model is named *stochvol* (line 14) and the format of the data file is

$$t \quad y_t$$

As in Example 2, the precision for the unstructured term λ_η is fixed, but we compute the marginal posteriors distributions for the elements of vector $\boldsymbol{\eta}$.

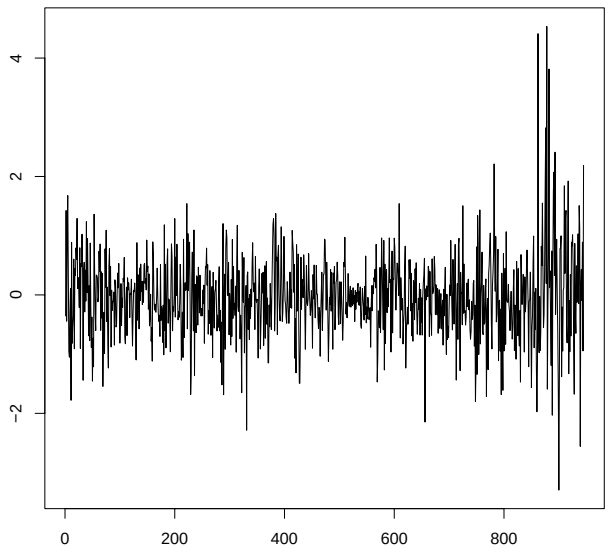
The AR1 model for $\boldsymbol{f}$ is defined in lines 17-28. Unlike all other models at the moment available for the *ffield* section, the AR1 has two hyperparameters, namely the precision parameter λ_f, and the transformed persistence parameter κ. Lines 22-24 specify the prior and the starting value for the precision parameter $\lambda_{\boldsymbol{f}}$, and lines 26-28 do the same for parameter κ.

The last section of the `ini` file describe the model for the common mean, the default value for the precision is used here.

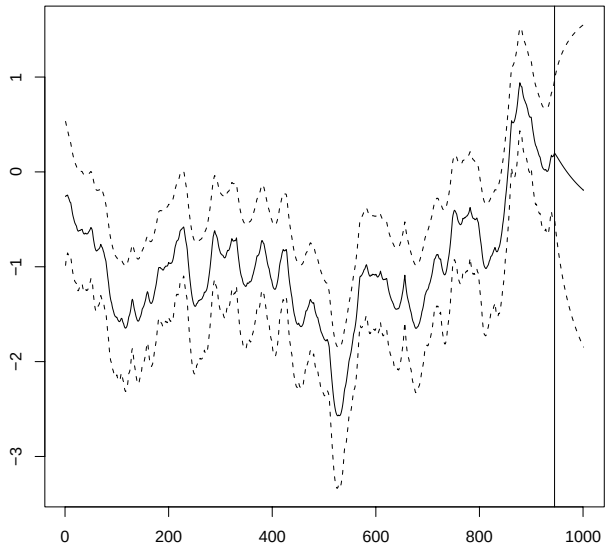
Note that the length of the data set n_d is 945 but we have set the length of the latent variable vector $\boldsymbol{\eta}$, to be $n_\eta = 1001$ (lines 7 and 21). In this way we obtain also predictions for the unobserved volatility for the 56 days following the last observation.

Building and running the model takes around 110 seconds on Machine 1 and 26 seconds on Machine 2.

Figure 5, panel (b), display the approximate posterior mean for the logarithm of the unobserved volatility, together with 0.025 and 0.975 posterior quantiles. The vertical line



(a) Log of the daily difference in the Pound/Dollar exchange rate



(b) Posterior mean of $\boldsymbol{\eta}$ together with 0.025 and 0.975 quantiles.

Figure 5: Data and results for the volatility model in Example 3

indicates the last observed data point.

3.4 Bayesian multiscale analysis for time series data

In the previous examples we were interested in the posterior marginals $\pi(x_i|\boldsymbol{y})$ where the uncertainty about the hyperparameter $\boldsymbol{\theta}$ is integrated out. We present here one example where it is important to be able to precisely estimate posterior marginals for a fixed value of the hyperparameter $\boldsymbol{\theta}$, that is $\pi(x_i|\boldsymbol{y}, \boldsymbol{\theta})$. The example is taken from Rue et al. (2007).

Example 4 *A signal is observed with noise and the goal of the analysis is to detect significant features and structures in the signal. Since some features might be visible only at some specific level of smoothing it is interesting to consider several levels of smoothing simultaneously. This is the idea behind the SIZer (Significant ZERo crossing of derivatives) methodology, see Chaudhuri and Marron (1999) and Erästö (2005).*

In our example the data are Gamma ray burst intensity, plotted in Figure 6 (panel (a)). The observations are assumed to be conditionally independent Poisson random variables

$$y(t_i)|\eta(t_i) \sim Po\{\exp(\eta(t_i)\} \quad i = 0, 1, \ldots$$

Where $\eta(t)$ is the underlying signal of interest. We assume $\eta(t)$ to be continuous with derivatives $\eta'(t)$, and level of smoothing κ. The derivative is said to be "significant positive" at time t if

$$\textit{Prob}(\eta'(t) > 0|\boldsymbol{y}, \kappa) > 1 - \alpha/2$$

with α being the level of significance. A similar definition holds for "significant negative".

We model $\eta(t)$ as an integrated Wiener process with precision κ which is Markov if augmented with derivatives (Wecker and Ansley, 1983), hence a discretely observed Wiener process observed in n time points is a GMRF of dimension $2n$, see Rue and Held (2005, Sec. 3.5). Our latent GMRF is then $\boldsymbol{x} = (\boldsymbol{\eta}, \boldsymbol{\eta}')$, that is the log-mean of the data augmented with its derivatives.

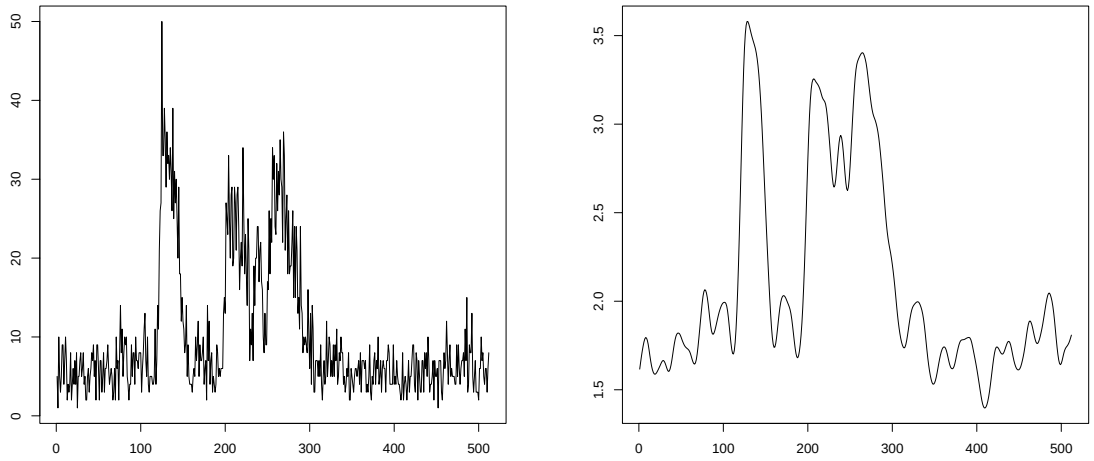
In this example the precision κ is fixed therefore there are no random hyperparameters in the model.

The file *BURST.ini* is as follows:

```
1  [Burst data example]
2  type = problem
3  dir = results-%d
4  smtp = GMRFLib_SMTP_BAND
5
6  [Poisson data]
7  type = data
8  likelihood = poisson
9  filename = burst.dat
10
11 [Unstruct term]
```

```
12 type = unstruct
13 n = 512
14 initial = 10
15 fixed = 1
16
17 [Smoother]
18 type = ffield
19 model = crw2
20 n = 512
21 covariates = covar.dat
22 initial = 7
23 fixed = 1
24 percentiles = 0
```

The ***smtp*** field in the *[Burst data example]* section (line 4) determines the type of solver for dealing with sparse matrices, in this case, since we know that the precision matrix of the problem is a band matrix, we can use the *GMRFLib_SMTP_BAND* solver which is optimal for band matrices.



(a) (b)

Figure 6: Multiscale analysis example: observed Gamma ray burst intensity (top) and posterior mean for the underlying signal $\eta(t)$ for level of smoothing given by $\log \kappa = 7$

Notice that all precision parameters are defined ***fixed*** in the `ini` file (lines 15 and 23). The log-precision of the *[Unstructured term]* section is fixed to a high value (line 14) again to mimic the absence of the unstructured component in the model, while the log-precision in the *[Smoother]* section is fixed to a user defined value, in this case $\log \kappa = 7$. This determines the level of smoothing in the result. The continuous time random walk model is defined in line 19. Note that even if the length of the smoother term is declared to

be 512 (line 20) the actual length of the output file is 1024 since the derivatives are also included. The derivatives constitutes the second half of the output file.

Since we are interested in checking where the derivatives are significantly positive or negative, we compute also the percentiles Prob$(x(t) < 0)$ for the smoother term (line 24). Figure 6 (panel (b)) displays the posterior mean of $\eta(t)$ for $\log \kappa = 7$. In Figure 7 the posterior mean of the derivatives $\eta'(t)$ is displayed. The band in the lower part of Figure 7 indicates where the derivatives are found to be significantly positive, negative or none.

The `inla` program runs in about 7 seconds on Machine 1 and about 2 seconds on Machine 2.

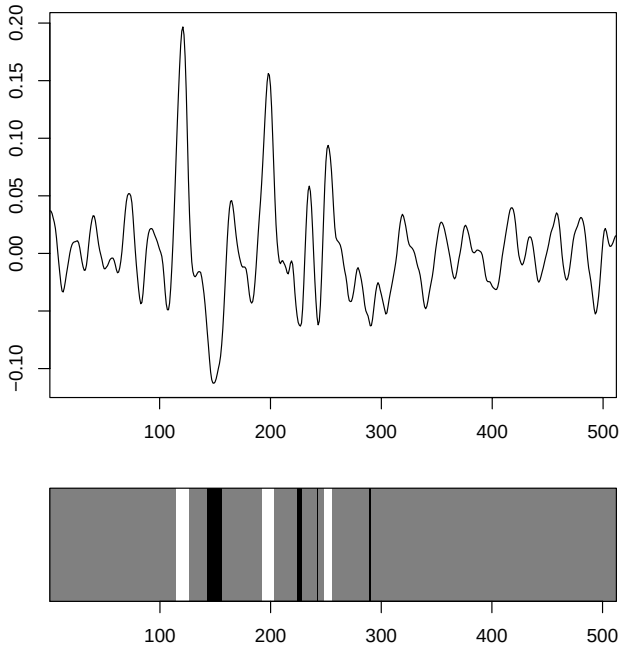


Figure 7: Multiscale analysis example: posterior mean of the derivatives $\eta'(t)$ is displayed. The band in the lower part of the figure indicates where the derivatives are found to be significantly positive (white), negative (black) or none (gray).

3.5 Disease mapping

Our next example is taken from (Rue and Held, 2005, Sec. 4.4.2). The data are collected over a spatial domain rather than over a time period. The data are georeferenced and we want to include the knowledge of the spatial location of the data in the model.

Each observed data y_i is linked to a spatial region $s \in \boldsymbol{S} = (0, \ldots, S-1)$, so that s_i indicates the region the ith data belongs to. A common way to introduce a spatially correlated effect is to assume that neighbouring sites are more alike than two arbitrary sites, therefore for a valid prior definition, a neighbourhood has to be defined for each site s. In geographical applications a common assumption is that two sites are neighbours if they share a common border.

Let $f_s(s_i)$ indicate the spatial effect. The prior model for $\boldsymbol{f}_s = (f(0), \ldots, f(s), \ldots, f(S-1))$ implemented in the `inla` program is a simple (but most often used) intrinsic GMRF model, see (Rue and Held, 2005, Ch. 3), defined as:

$$f_s(s)|f_s(s'), s \neq s', \lambda_s \sim \mathcal{N}(\frac{1}{n_s}\sum_{s \sim s'} f_s(s'), \frac{1}{n_s \lambda_s}) \tag{7}$$

where n_s is the number of neighbours of site s, $s \sim s'$ indicates that the two sites s and s' are neighbours. λ_s is the unknown precision parameter.

The neighbourhood structure has to be passed to the `inla` program through a file which describes the graph of the spatial component of the model. We describe the required format for such a file using a small example. Let the file *gra.dat*, relative to a small graph, be

```
1    5
2    0  1  1
3    1  2  0  2
4    2  3  1  3  4
5    3  1  2
6    4  1  2
```

Line 1 declares the total number of nodes in the graph, then, in lines 2-6 each node is described. For example, line 4 states that node 2 has 3 neighbours and these are nodes 1, 3 and 4. This is the same format used in the GMRFLib library.

Example 5 *The number of cases of oral cavity cancer is observed for a 5 year period (1986-1990) in the 544 districts of Germany. The goal of the analysis is to explore the spatial distribution of the data. The common approach is to assume that the data are conditionally independent Poisson counts*

$$y_i|\eta_i \sim \textit{Po}(E_i \exp(\eta_i)) \quad i = 0, \ldots, 543$$

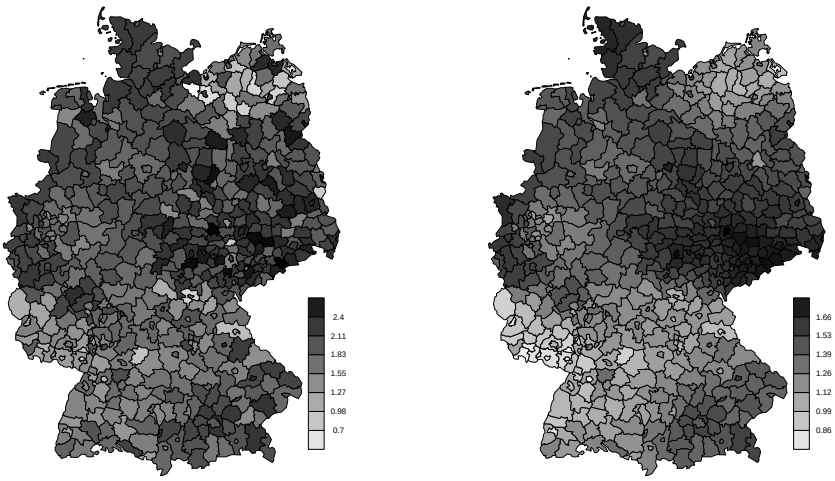
where E_i is a fixed quantity which accounts for number of people in district i, age distribution etc. The standardised mortality ratios y_i/E_i are displayed in Figure 8, panel (a).

The model for the latent variable η_i takes the following form

$$\eta_i = \mu + f_s(s_i) + u_i \tag{8}$$

where μ is th common mean, $\boldsymbol{f}_s$ is a spatially structured term and $\boldsymbol{u}$ is the unstructured term which accounts for non-observed variability. The prior model for $\boldsymbol{f}_s$ is the intrinsic GMRF in equation (7). We impose a sum-to-zero restriction on $\boldsymbol{f}_s$ ($\sum_s f(s) = 0$) to ensure identifiably of μ.

Following Rue and Held (2005), the two precision hyperparameters of the model (λ_u, λ_s) are both given Gamma priors with $a = 1$ and $b = 0.01$.



(a) (b)

Figure 8: Standardised mortality ratio for oral cavity cancer, panel (a) and estimated relative risks (posterior mean) of the spatial component $\exp(\boldsymbol{f}_s)$.

The `DISEASE-oral.ini` file describing the model for the `inla` program is:

```
[Oral-cavity cancer data]
type = problem
dir = results-for-oral-%d

[Unstruct]
type = unstruct
prior = gamma
parameters = 1 0.01
n = 544
```

```
10
11 [data]
12 type = data
13 likelihood = poisson
14 filename = oral.txt
15
16 [Spatial]
17 type = ffield
18 model = besag
19 covariates=spatial.covariate
20 parameters = 1 0.01
21 constraint = 1
22 graph = germany.gra
23
24 [Constant]
25 type = linear
```

The *[unstruct]* section (lines 5-9) defines the model for u_i. Unlike the previous examples, here there actually is an unstructured component, therefore in this case λ_η is not fixed.

The model for the spatial component of $f_s(\cdot)$ is defined in lines 16-22. The section is defined by ***type*** = *ffield* . The intrinsic GMRF model in equation (7) is named *besag* in the `inla` program. Line 21 defines the sum-to-zero constraint for $\boldsymbol{f}_s$. The graph of $\boldsymbol{f}_s$ is read from a file (line 22). The last section, lines 24-25 defines the model for the common mean μ. Figure 8, panel (b), displays the posterior mean of the spatial component $\exp(\boldsymbol{f}_s)$.

A different parametrisation would have been possible for the same model. Namely we could have dropped the common mean μ and the sum-to-zero constraint. Modifying the `ini` file to account for this other parametrisation is extremely easy; it is, in fact, sufficient to remove lines 24-25 defining the common mean and line 21 defining the constraint.

The `inla` program allows also the possibility to introduce a user defined model for some functions $f(\cdot)$ in equation (2). This is done in a ***type*** = *ffield* section specifying the field ***model*** = *generic*. The user then has to provide the precision function $\boldsymbol{Q}$, corresponding to the stochastic vector $\boldsymbol{f}$, in a file with the following format

$$i \quad j \quad \boldsymbol{Q}_{ij}$$

where i and j are the row and column index and $\boldsymbol{Q}_{ij}$ is the correponding element of the precision matrix. Only the non-zero elements of the precision matrix need to be stored in the file. For example, we could have stored the precision matrix corresponding to the spatial effect in (8) in a file, named `Qmat.dat`. We report the few first lines of such file:

```
1  0 0 1
2  0 11 -1
3  1 1 2
4  1 9 -1
```

The same model as in (8) can then be defined in a new `ini` file as following:

```
1  [Oral-cavity cancer - User defined Q matrix]
2  type = problem
3  dir = results-%1d
4
5  [Unstruct]
6  type = unstruct
7  prior = gamma
8  parameters = 1 0.01
9  n = 544
10
11 [data]
12 type = data
13 likelihood = poisson
14 filename = oral.txt
15
16 [Spatial]
17 type = ffield
18 model = generic
19 Qmatrix = Qmat.dat
20 rankdef = 1
21 covariates = spatial.covariate
22 parameters = 1 0.01
23 constraint = 1
24
25 [Constant]
26 type = linear
```

Notice that the only difference with respect to the `ini` file previously used is in the section *[Spatial]*. Here we declare ***model*** = *generic* and specify the file containing the $\boldsymbol{Q}$ function in line 19. The `inla` program then builds a graph based on the non-zero pattern of the specified precision matrix. The otional argument ***rankdef***, in line 20, specifies the rank deficiency of the precision matrix. For the intrinsic model in equation (7) the rank deficiency is 1.

3.6 Disease mapping with covariate

We present now an extension of the model in Example 5 which allows for adjusting the log-relative risk by a semi-parametric function of a covariate which is believed to influence the risk. The model is a Bayesian semiparametric model with an additional spatial effect. These kinds of models have been named "geoadditive models" in Kammann and Wand (2003). For an introduction to the subject see, for example, Fahrmeir and Tutz (2001). The example below is taken from Rue et al. (2007).

Example 6 *Larynx cancer mortality counts are observed in the 544 district of Germany from 1986 to 1990. As in Example 5 we assume the data to be conditionally independent Poisson random variables with mean $E_i \exp(\eta_i)$, where E_i is fixed and accounts for demographic variation, and η_i is the log-relative risk. Together with the counts, for each district, the level of smoking consumption c is registered.*

The model for η_i takes the following form

$$\eta_i = \mu + f_s(s_i) + f(c_i) + u_i \tag{9}$$

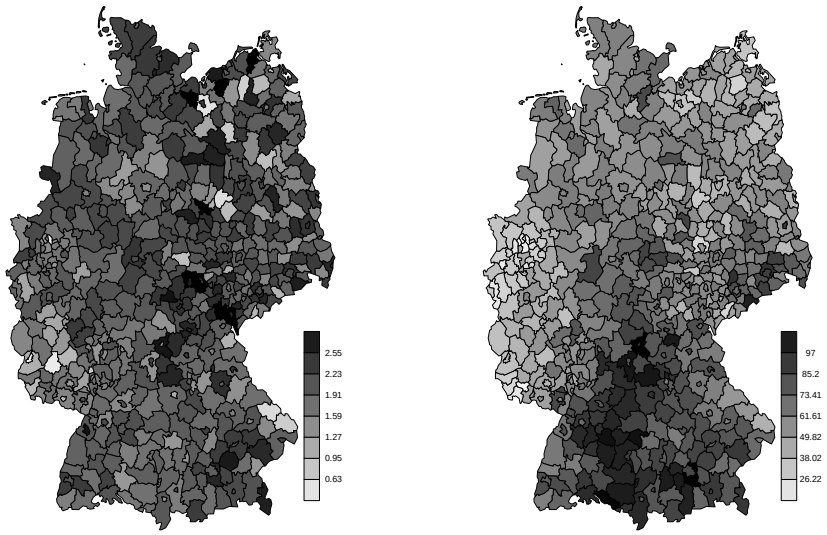
where, as in Example 5, $f_s(\cdot)$ is the spatial effect modelled according to (7), and u_i is the unstructured random effect. The remaining term in (9), $f(c_i)$, is the unknown effect of of the exposure covariate which assumes value c_i for observation i. The effect of covariate c is modelled as a smooth function $f(\cdot)$ parametrised as unknown values $\boldsymbol{f} = (f_0, \ldots, f_{m-1})^T$ at $m = 100$ equidistant values of c_i. We have scaled the covariate values so that they belong to the interval $[0, 10]$. The vector $\boldsymbol{f}$ is modelled with a second-order random walk (RW2) prior with unknown precision λ_f. A sum-to-zero constraint is imposed on $\boldsymbol{f}_s$ and $\boldsymbol{f}$ separate out the spatial effect and the effect of the covariate from the common mean μ.

The model has three hyperparameters $\boldsymbol{\theta} = (\lambda_s, \lambda_f, \lambda_\eta)$. Following Rue et al. (2007) we assign a vague Gamma prior to each element of $\boldsymbol{\theta}$.

In Figure 10 the standardised mortality ratios, y_i/E_i are displayed (panel (a)) together with the observed values of the covariate c (panel (b)).

The `DISEASE-COVARIATE.ini` file defining the model is the following:

```
[Disease mapping with covariate]
type = problem
dir = results-%d

[Unstruct term]
type = unstruct
n = 544
prior = gamma
initial=9
```



(a) (b)

Figure 9: Standardised mortality ratio for larynx cancer, panel (a) and observed covariate values, panel(b)

```
parameters = 1.0 0.00005

[Data]
type = data
likelihood = poisson
filename = larynx.dat

[Spatial]
type = ffield
model = besag
covariates=spatial-covariate.dat
prior = gamma
parameters = 1.0 0.00005
graph = germany.gra
constraint = 1
initial=3
diagonal = 0.001

[Covariate]
type =ffield
model = rw2
```

```
31 covariates = covariate.dat
32 locations=covariate.value
33 prior = gamma
34 parameters = 1 0.05
35 initial=9
36 diagonal = 0.00001
37 quantiles=0.025 0.975
38 constraint = 1
39
40 [Constant linear]
41 type = linear
42
43 [INLA param]
44 type = INLA
45 gradient_finite_difference_step_len = 0.001
46 hessian_finite_difference_step_len = 0.001
```

The section *[Spatial]* defines the model for the structured spatial component $\boldsymbol{f}_s$. We recognise the intrinsic GMRF model in line 19 and the graph file in line 23. The field ***diagonal*** at line 36 indicates a (small) number to be added to the diagonal of the precision matrix for $\boldsymbol{f}_s$ to ensure that it is positive definite.

The model for the semi-parametric function $\boldsymbol{f}$, which is the new feature introduced by this example, is defined in the section tagged *[Covariate]*. The file *covariate.value* declared in line 32 contains all values that the covariate c could assume, they are ordered from the lower to higher. In this case the file contains one sequence of numbers from 0 to 9.9 with step 0.1. The file *covariate.dat* contains information on which values of c is actually observed in each district. We report the first 5 lines of the file to better explain the format of such files

```
1 0 56
2 1 65
3 2 50
4 3 63
5 4 65
```

For example, line 3 tells us that for district 2 the observed value of the covariate c is the 50th element of the series in file *covariate.value*, that is 0.5.

In the last section, tagged *[INLA param]* we define the step length for the numerical computation of the gradient and Hessian of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ at the mode. This is necessary because the default values do not always ensure a positive definite Hessian matrix.

The computation time is about 30 seconds on Machine 1 and 15 seconds on Machine 2.

Figure 10 displays the posterior mean of the spatial effect $\boldsymbol{f}_s$ for all districts, while Figure 11, panel (a), displays the effect of the covariate c (posterior mean) within 2.5 and

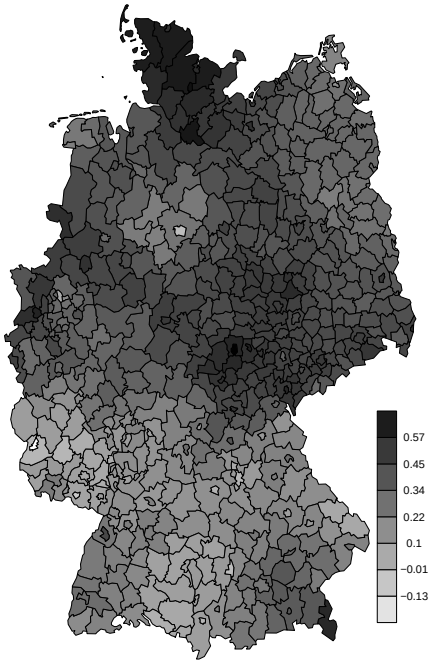


Figure 10: Posterior mean for the structured spatial effect $\boldsymbol{f}_s$

97.5% confidence intervals. The covariate effect is not too far from a linear effect. We might, therefore, want to run a modified version of the model in which the effect of c is modelled as a linear function, that is

$$\eta_i = \mu + f_s(s_i) + \beta c_i + u_i$$

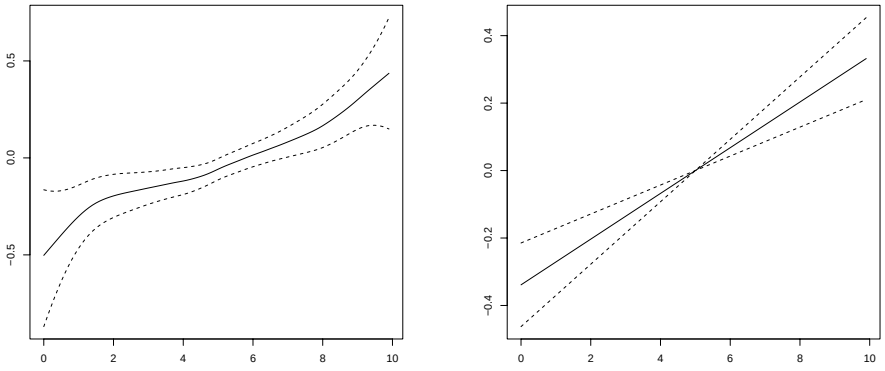
To modify the `DISEASE-COVARIATE.ini` file in order to fit the new model it is enough to delete the *[Covariate]* section, lines 28-38 and instead add the following section where β is defined.

```
1 [Covariate linear]
2 type=linear
3 covariates=covariate-linear.dat
```

The file *covariate−linear.dat* has the format

$$i \quad c_i$$

The computation time for the linear-effect model reduces to 11 seconds for Machine 1 and to 6 seconds on Machine 2. This is due to the fact that in the linear model both the



(a) (b)

Figure 11: Effect of the covariate. Panel (a) nonparametric model and panel (b) linear model: posterior mean within 2.5 and 97.5% confidence interval.

latent field $\boldsymbol{x}$ and the vectors of hyperparameters $\boldsymbol{\theta}$ are of lower dimensionality.

The estimated posterior mean for the slope parameter β is 0.0677 with posterior standard deviation 0.0126. Figure 11, panel (b), displays the linear effect of the covariate within 0.025 and 0.975 quantiles. To compute the quantiles for the regression line in Figure 11, panel (b), we have run the model described in the `DISEASE-COVARIATE.ini` file fixing the log precision of the RW2 model to a high value. In this way the RW2 is forced to be a straight line.

3.7 Mapping cancer incidence

We present a little more complicated example on the same line of examples 5 and 6. Instead of observing only one data point for each district, in the next example there are multiple observations sharing the same spatial location. Therefore, a possible unstructured spatial effect needs to be coded in a different way than in the two previous examples. The example is taken from Rue and Held (2005, Sec 4.3.5).

Example 7 *The data are incident cases of cervical cancer in the former East German Republic (GDR) from 1979, stratified by district and age group. Each cases was classified as pre-malignant (coded as 0) or malignant (coded as 1). For each of the* $n_d = 6\,690$ *cases in the data set, the age,* age_i*, and the district,* s_i*, of the patient are available. The age was categorised into 15 age groups.*

The data are assumed to be conditionally independent Bernoulli random variables:

$$y_i|\eta_i \sim \mathcal{B}(p_i) \quad i = 0, \ldots, n_d$$

with logit link function

$$p_i = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)}$$

The model for the latent variables is:

$$\eta_i = \mu + f(age_i) + f_s(s_i) + f_u(s_i)$$

where $f(age)$ *is the age group effect, modelled as a RW2 with precision parameter* λ_f*. The spatial effect of the district* s_i *is split into a spatially correlated part and an uncorrelated one. The spatially correlated element,* $f_s(\cdot)$*, is modelled as the intrinsic GMRF in equation (7) with given neighbouring structure. The uncorrelated part,* $f_u(\cdot)$*, is modelled as by a i.i.d Gaussian effect. Note that, in this model, the unstructured spatial effect* $f_u(\cdot)$*, does not coincide with the unstructured term* u_i *in equation (2), which was the case in Examples 5 and 6.*

There are three hyperparameters in the model $\boldsymbol{\theta} = (\lambda_f, \lambda_s, \lambda_u)$*. Following Rue and Held (2005), we assume a Gamma*$(1.0, 0.01)$ *prior distribution for* λ_s *and* λ_u *and a Gamma*$(1.0, 0.00005)$ *prior for* λ_f*. Moreover we impose a sum-to-zero constraint on both* $\boldsymbol{f}$ *and* $\boldsymbol{f}_s$

The file *CANCER−INCIDENCE.ini* defining the model is:

```
1 [Cancer incidence]
2 type = problem
3 dir  = results-%d
4
5 [Unstruct]
```

```
type = unstruct
n = 6690
initial = 15
fixed = 1

[Likelihood model]
type = data
likelihood = binomial
filename = cancer.dat

[Age classes]
type = ffield
model = rw2
covariates = age-group-cov.dat
n=15
constraint = 1
diagonal = 1.0e-4
parameters = 1 0.001
initial = 6.456745
quantiles=0.025 0.975

[Spatial]
type = ffield
model = besag
graph = ddr.gra
covariates = spatial-cov.dat
constraint = 1
diagonal = 1.0e-4
parameters = 1 0.0005
initial = 8.006793

[Spatial random effect]
type = ffield
model = iid
n = 216
parameters = 1 0.01
covariates = spatial-cov.dat
initial = 4.512093

[constant]
type = linear

[Parameters for INLA]
type = INLA
gradient_finite_difference_step_len = 0.01
hessian_finite_difference_step_len = 0.01
```

Note that while in Examples 5 and 6 the spatial unstructured component in the model was coded in the ***type***=*unstruc* section of the `ini` file, here, for the same purpose, we have to include a ***type***= *ffield* section where ***model***=*iid* (lines 37-43).

The model runs in about 90 seconds on Machine 1 and about 30 seconds on Machine 2.

In Figure 12 the posterior mean of the non-parametric effect of the age group within 2.5 and 97.5% confidence band is dispayed.

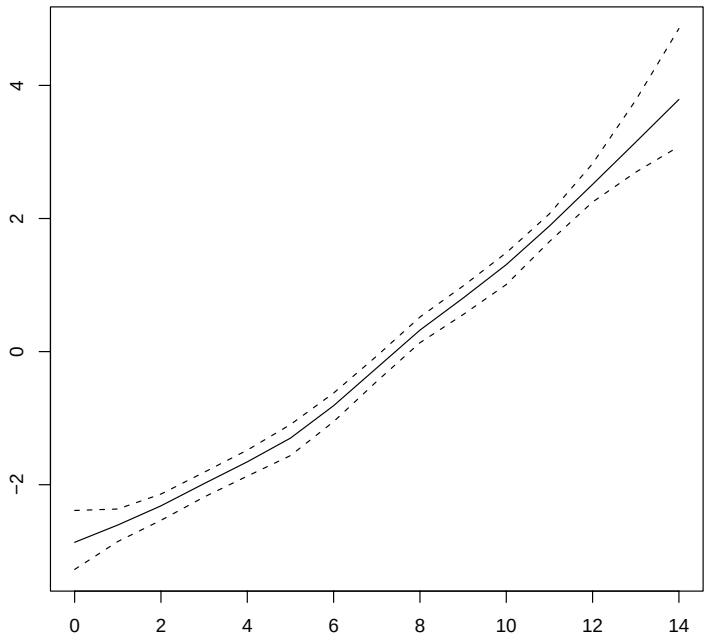


Figure 12: Nonparametric effect of age group. Posterior mean within 2.5 and 97.5% quantiles.

3.8 Geoadditive model: Munich rental guide

In this section we present a slightly more complex example of geoadditive models where we have a higher number of covariates in the data set. The example is taken from Rue and Held (2005, Sec. 4.2.1).

Example 8 - Munich rental guide

The response variable y_i is the rent (Euro per square meter) for a flat in Munich. There are three covariates to be included in the model: the spatial location (s_i), the floor space (size$_i$) and the year of construction (year$_i$). Moreover for each data point we have a set of indicator variables such as whether or not the flat has central heating, bathroom, a large balcony, etc. The data set consist in $n_d = 2\,035$ observations. There are 380 district in Munich, the floor size varies from 17 to 185 square meters and the year of construction goes from 1918 to 2001

The model for the data is:

$$y_i|\eta_i \sim \mathcal{N}(\eta_i, 1/\lambda_y)$$

with

$$\eta_i = \mu + f_s(s_i) + f_0(\text{size}_i) + f_1(\text{year}_i) + \boldsymbol{z}_i^T\boldsymbol{\beta} \tag{10}$$

where $f_s(\cdot)$ is the spatial effect modelled as the intrinsic GMRF in equation (7), $f_0(\cdot)$ is the non parametric effect of the floor size and $f_1(\cdot)$ is the non parametric effect of the year of construction. Both $f_0(\cdot)$ and $f_1(\cdot)$ are modelled as RW2 with unknown precision. The last term in (10) models the covariates assumed to have a linear effect. As usual we choose a Gaussian prior with known precision for the elements of vector $\boldsymbol{\beta}$. We impose a sum-to-zero constraint on $f_s(\cdot)$, $f_0(\cdot)$ and $f_1(\cdot)$.

The model has four hyperparameters $\boldsymbol{\theta} = (\lambda_y, \lambda_s, \lambda_0, \lambda_1)$. We assign each precision a Gamma$(1.0, 0.001)$ prior. In this example we approximate also the posterior marginals for the four hyperparameters $\boldsymbol{\theta}$.

In the following we report part of the `RENT.ini` file which defines the model. We have omitted the part defining most of the indicator variables since they are all defined in the same way.

```
[Rent in Munich]
type = problem
dir = results-%d
hyperparameters = 1

[Unstruct term]
type = unstruct
n = 2035
parameters = 1.0 0.001
```

```
initial = 10
fixed = 1

[Data]
type = data
likelihood = gaussian
filename = rent.dat
parameters = 1 0.001
initial = -1

[floor-size]
type =ffield
model = rw2
covariates = size-covariate.dat
locations = size-loc.dat
diagonal = 1.0e-6
initial = 7
constraint = 1
parameters = 1 0.001
quantiles = 0.25 0.975

[spatial]
type = ffield
model = besag
graph = munich.gra
covariates = spatial-covariate.dat
diagonal = 0.00001
constraint = 1
initial = 0.4
parameters = 1 0.001
compute=1

[year]
type =ffield
model = rw2
covariates = year-covariate.dat
locations = year-loc.dat
diagonal = 1.0e-6
initial = 7
constraint = 1
parameters = 1 0.001
quantiles = 0.25 0.975

[constant]
type = linear
precision = 0.01
```

```
56
57 [linear-beste.dat]
58 type = linear
59 covariates = beta-beste.dat
60 precision = 0.01
61
62 .
63 .
64 .
65
66 [INLA param]
67 type = INLA
68 int_strategy = GMRFLib_AI_INT_STRATEGY_CCD;
69 gradient_finite_difference_step_len = 0.01
70 hessian_finite_difference_step_len = 0.01
```

The flag ***hyperparameters*** in line 4 section is turned on to indicate that also posterior marginals for the hyperparameters have to be computed. The results are displayed in Figure 13 and they agree well with tho use found by Rue and Held (2005).

The new feature introduced in this example is the use of a different integration scheme to compute

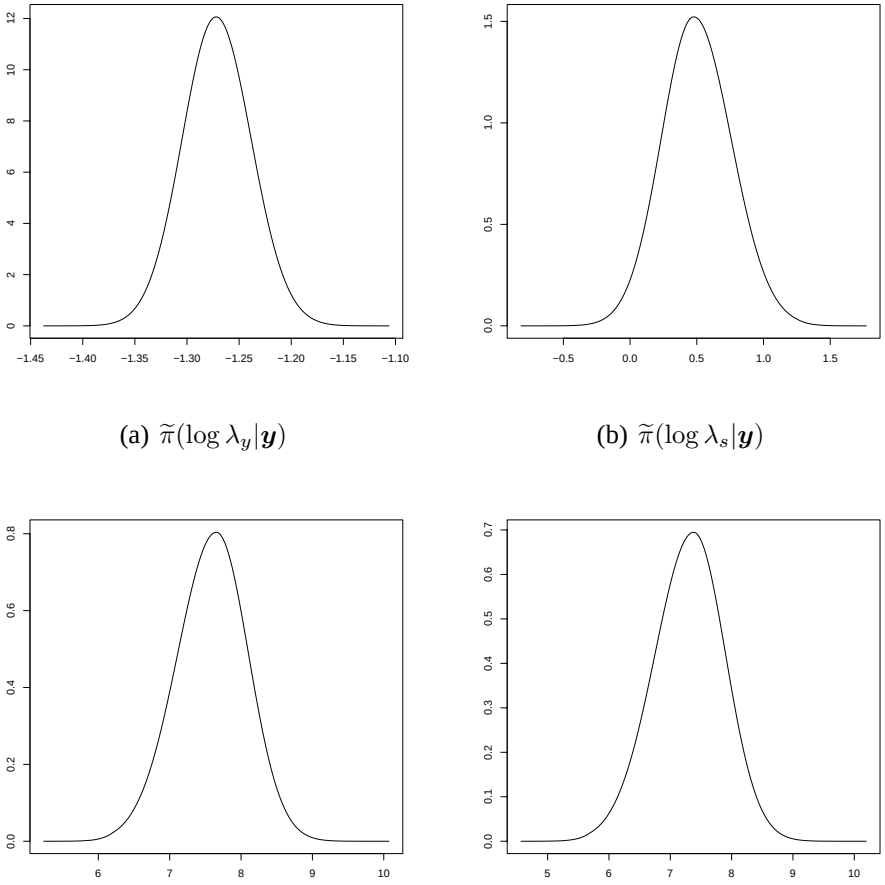
$$\widetilde{\pi}(x_i|\boldsymbol{y}) = \sum_k \widetilde{\pi}(x_i|\boldsymbol{y}, \boldsymbol{\theta}_k)\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})\Delta_k \tag{11}$$

When the dimension of the hyperparameters space grows, in fact, the grid integration scheme, which was used in all previous examples and which is the default choice in the `inla` program, soon becomes too computationally intensive. The central composit design (CCD) integration scheme, defined in line 68, is an alternative integration scheme which computes the integral in (11) using much less points, still providing useful results. Both integration schemes are described in Rue et al. (2007).

Figure 17, panels (a) and (b), displays the posterior mean, within 0.25 and 0.975 quantiles, of the effect of the floor size and the year of construction respectively.

To check the quality of the CCD integration scheme we run the model once more using the default grid scheme (to do so it is enough to delete line 67). The results are plotted in Figure 14 as dotted lines, they are indistinguishable from the CCD results despite the fact that the grid integration scheme used 115 evaluation points to compute the integral in (11) and the CCD one only 15.

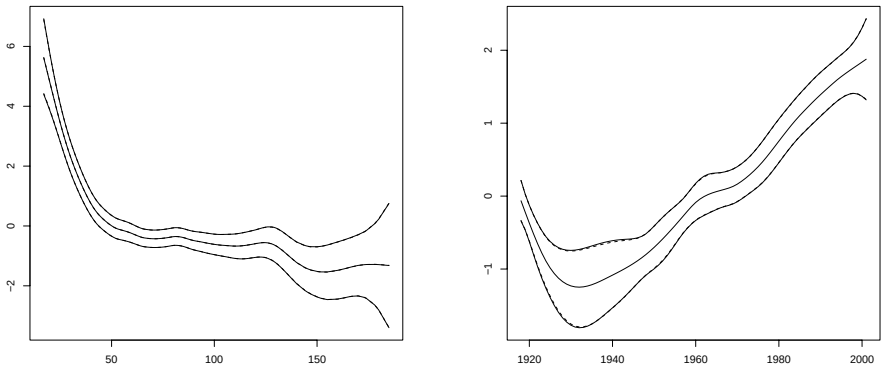
The computing time for this model on Machine 1 is of 80 seconds if we use the CCD scheme and 250 seconds using the grid scheme. On Machine 2 the computational time reduces to 30 seconds in the first case and 70 in the second case.



(a) $\widetilde{\pi}(\log \lambda_y|\boldsymbol{y})$ (b) $\widetilde{\pi}(\log \lambda_s|\boldsymbol{y})$

(c) $\widetilde{\pi}(\log \lambda_0|\boldsymbol{y})$ (d) $\widetilde{\pi}(\log \lambda_1|\boldsymbol{y})$

Figure 13: Munich rent example: approximate posterior marginals for the hyperparameter of the model.



(a) (b)

Figure 14: Munich rent example: semiparametric effect of the floor size (a) and of the year of construction (b). The posterior mean within 0.025 and 0.975 quantiles is displayed. The solid line is the result of the CCD integration scheme and the dotted line is the result of the grid integration scheme.

3.9 Geoadditive model: Zambia children undernutrition

The second example of geoadditive model with several covariates is from Kandala et al. (2001) and is one of the worked out examples in the BayesX web page.

Example 9 - Undernutrition of children in Zambia. *Undernutrition in children is measured determining the anthropometric status of the child relative to a reference standard. In our example undernutrition is measured by stunting, or inefficiency height for age, indicating chronic undernutrition. Stunting for a child i is determined using a Z score defined as*

$$Z_i = \frac{AI_i - MAI}{\sigma}$$

where AI refers to the child's anthropometric indicator, MAI refers to the median of the reference population and σ refers to the deviation of the standard population.

The main interest is on modelling the dependence of undernutrition on a set of covariates including the age of the child (age_i), the body mass index of the child's mother (bmi_i), the district the child lives in (s_i) and some further categorical covariates. The data set consists in $n_d = 4846$ observations. For more details about the data set see Kandala et al. (2001) and Kneib et al. (2004).

We assume the scores Z_i to be conditionally independent Gaussian random variables

$$Z_i|\eta_i \sim \mathcal{N}(\eta_i, 1/\lambda_y)$$

and

$$\eta_i = \mu + f_0(bmi_i) + f_1(agc_i) + f_s(s_i) + f_u(s_i) + \boldsymbol{z}_i^T\boldsymbol{\beta}$$

where $f_0(\cdot)$ and $f_1(\cdot)$ are the semi parametric effect of the mother's body mass index and the age of the child respectively. $f_s(\cdot)$ is the structured spatial effect of the district, $f_u(\cdot)$ is an unstructured spatial effect and $\boldsymbol{z}_i$ are a set of categorical covariates. We model the spatial structured effect $f_s(s_i)$ as the intrinsic GMRF in equation (7) and $f_0(\cdot)$ and $f_1(\cdot)$ as RW2. The unstructured spatial effect $f_u(s_i)$ is modelled by i.i.d. Gaussian random variables. We impose a sum-to-zero constraint for $f_s(\cdot)$, $f_0(\cdot)$ and $f_1(\cdot)$.

In this model there are five hyperparameters $\boldsymbol{\theta} = (\lambda_y, \lambda_s, \lambda_u, \lambda_0, \lambda_1)$ and we assign a vague Gamma prior distribution to each of them.

```
[Zambia model]
type = problem
dir = results_%d

[Unstruct term]
type = unstruct
n = 4846
prior = gamma
```

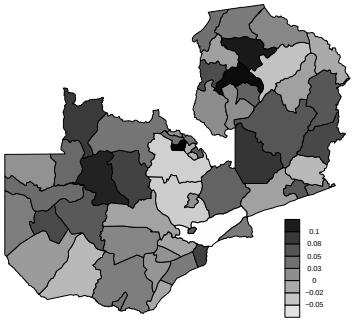
```
parameters = 1.0  0.005
initial = 10
fixed = 1

[Data]
type = data
likelihood = gaussian
filename = zambia.dat
parameters = 1  0.005
initial = 0.2

[spatial]
type = ffield
model = besag
graph = zambia.gra
covariates = spatial_covariate.dat
diagonal = 0.00001
constraint = 1
initial = 3.6
parameters = 1  0.005

[spatial unstruct]
type = ffield
model = iid
covariates = spatial_covariate.dat
n =57
diagonal = 0.00001
initial = 5.4
parameters = 1  0.005

[agc]
type = ffield
model = rw2
covariates = agc.dat
n=60
diagonal = 0.0001
constraint = 1
initial = 6.6
parameters = 1  0.005
quantiles = 0.025  0.975

[bmi]
type = ffield
model = rw2
covariates = bmi_covariate.dat
locations = bmi.location
```

```
diagonal = 0.00001
constraint = 1
initial = 6.2
parameters = 1 0.005
quantiles = 0.025 0.975

[beta]
type=linear

[rcw]
type=linear
covariates = rcw.dat

[edu1]
type=linear
covariates = edu1.dat

[edu2]
type=linear
covariates = edu2.dat

[sex]
type=linear
covariates = sex.dat

[tpr]
type=linear
covariates = tpr.dat

[INLA param]
type = INLA
int_strategy = GMRFLib_AI_INT_STRATEGY_CCD;
```

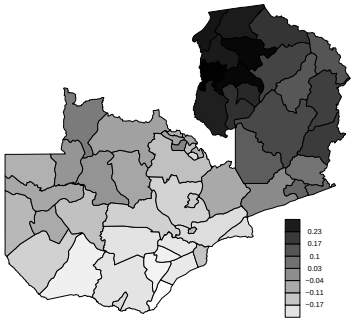
Also in this example we use the CCD integration scheme to compute the integral in (11).

In Figure 15, panels (a) and (b), the posterior mean of the unstructured and structured spatial effect is displayed. The effect of the age of the children is in Figure 15, panel (c). It shows a clear non linear pattern. The effect of the mother's body mass index (Figure 15, panel (d)) instead is more regular and could probably be substitute in the model formulation by a linear effect.

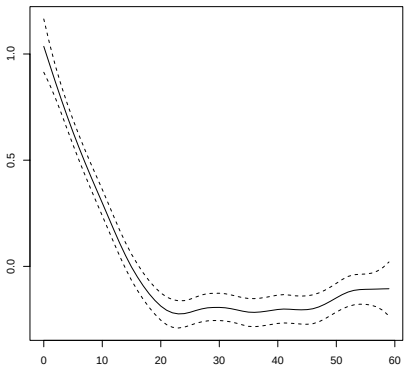
The computation time is about 4 minutes on Machine 1 and 1 minute on Machine 2.



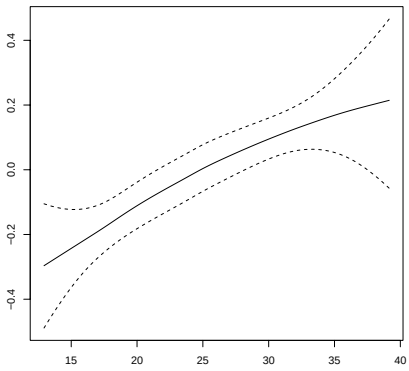
(a)



(b)



(c)



(d)

Figure 15: Results for the Zambia example. Panel (a) and (b) displays the posterior mean of the unstructured and structured spatial effect respectively. Panel (c) and (d) display the posterior mean, within 0.025 and 0.975 quantiles, of the age effect (c) and of the mother's body mass index (d)

3.10 Log-Gaussian Cox processes

The particular feature of our next example is that data are registered on a regular grid of dimension $n_{row} \times n_{col}$, where n_{row} is the number of row and n_{col} the number of columns. Unlike all the previous examples then, each data is identified by two indexes (i, j) indicating respectively the row and column the data point belongs to. This example is taken from Rue et al. (2007).

Example 10 *Log-Gaussian Cox processes (LGCP) are a class of models used for modelling spatial point processes, see for example Møller and Waagepetersen (2003). A LGCP is a Poisson point process.* $\boldsymbol{Y} \in W \subset \mathcal{R}^d$. *with random intensity function* $\lambda(\boldsymbol{\xi}) = \exp(Z(\boldsymbol{\xi}))$*, where* $Z(\boldsymbol{\xi})$ *is a Gaussian field and* $\boldsymbol{\xi} \in W$*. It is common practice to discretise the observation windows* W *into* $N = n_{row} \times n_{col}$ *disjoint cells* $\{s_{ij}\}$*with area* $|s_{ij}|$ *where* $i = 0, \ldots, n_{row} - 1$ *and* $j = 0, \ldots, n_{col} - 1$*.*

Let y_{ij} *be the observed number of occurrences of the realised point pattern within* s_{ij}*. Let* η_{ij} *be the random variable* $Z(\boldsymbol{\xi}_{ij})$*. The likelihood of the model is*

$$y_{ij}|\eta_{ij} \sim Po(|s_{ij}| \exp(\eta_{ij}))$$

while, as usual the latent variable vector $\boldsymbol{\eta}$ *is part of a larger GMRF.*

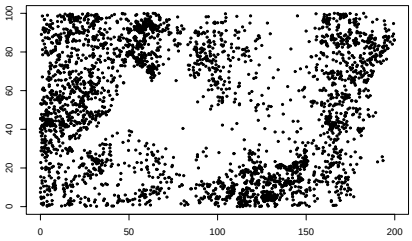
In this example, the data consist in the locations of a particular tropical tree species (Beilschmiedia pendula Lauraceae) registered in a 50-hectares plot in the tropical moist forest of Barro Colorado Island in central Panama. For more information about this study see Waagepetersen (2006). The 3605 tree locations are plotted in Figure 6, panel (a). We divide our region of interest into a 201×101 *regular grid, where each square pixel represent an area of 25 squares meters. Together with the data* y_{ij}*, we observe, the mean elevation and the mean norm of the gradient for each area on the grid. These covariates are believed to influence the behaviour of the tree under examination. A scaled version of these covariates is displayed in Figure 16, panels (b) and (c). The model for the latent variable* η_{ij} *is*

$$\eta_{ij} = \mu + \beta_1 alt_{ij} + \beta_2 grad_{ij} + f_s(s_{ij}) + u_{ij}$$

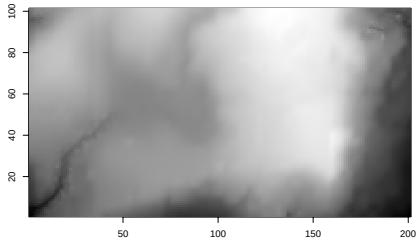
where alt_{ij} *and* $grad_{ij}$ *are the values for the two covariates at location* (i, j)*,* f_s *is the spatial structured effect of the location and* u_{ij} *is the unstructured random effect.*

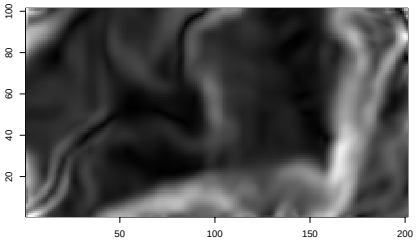
For the spatial structured term $\boldsymbol{f}_s$ *we use a second order polynomial intrinsic GMRF with unknown precision* λ_f*. See Rue and Held (2005, Sec 3.4.2) for a thorough definition of intrinsic GMRF models on a lattice. We use vague Gaussian priors for* μ*,* β_1 *and* β_2*. The unstructured terms* u_{ij} *are independent* $\mathcal{N}(0, 1/\lambda_u)$ *random variables. Notice that the latent field* $\boldsymbol{x} = (\boldsymbol{\eta}, \boldsymbol{f}_s, \mu, \beta_1, \beta_2)$ *in this example has dimension* $40\,605$*.*

The hyperparameters are $\boldsymbol{\theta} = (\lambda_f, \lambda_u)$ *are are assigned vague Gamma priors.*



(a)





(b) (c)

Figure 16: Data and covariate for the LGCP example: panel (a) displays locations for the 3065 trees, panel (b) displays the altitude and panel (c) the norm of the gradient.

```
[Tropical rainforest data]
type = problem
dir = results-%d

[Poisson data]
type = data
likelihood = poisson
filename = data-full.dat

[Unstruct term]
type = unstruct
nrow = 101
ncol = 201
initial = 0.4

[Spatial smoother]
type = ffield
covariates=spatial-full.dat
nrow=101
ncol=201
```

```
21 model = rw2d
22 constraint=1
23 initial=0.7
24
25 [Constant]
26 type = linear
27
28 [Altitude Covariate]
29 type = linear
30 covariates = altitude-full.dat
31
32 [Gradient Covariate]
33 type = linear
34 covariates = gradient-full.dat
35
36 [INLA parameters]
37 type = INLA
38 gradient_finite_difference_step_len = 0.001
39 hessian_finite_difference_step_len = 0.001
```

The data file *data−full.dat* has the following format

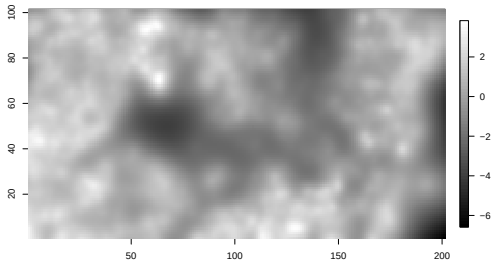
$$i \quad j \quad |s_{ij}| \quad y_{ij}$$

where $i = 0, \ldots, n_{row} - 1$ is the row index and $j = 0, \ldots, n_{col} - 1$ is the column index. Notice then, that for data observed on a grid the data file has four columns instead of three (see Appendix A.1.2). The data are stored by row, so that the first n_{row} lines of the data file refer to row 1, the second n_{row} lines to row 2 etc. The same also for the covariate files.

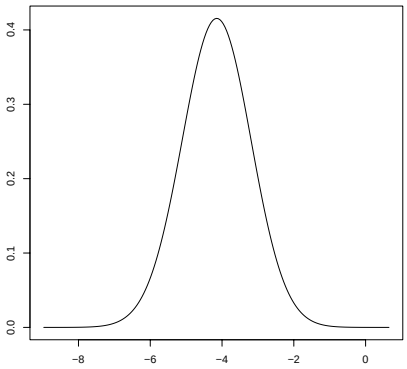
Notice also that it is required for the user to specify the number of rows and columns in the data set (lines 12-13 and 19-20). For grid observed data, the fields ***nrow*** and ***ncol*** substitute the field ***n*** which we have used in all previous examples. The prior model for the spatial effect is defined in line 21.

The results are displayed in Figure 17. Panel (a) shows the posterior mean of the structured spatial effect. Panels (b)-(d) show the posterior marginal distributions for the parameters μ, β_1 and β_2.

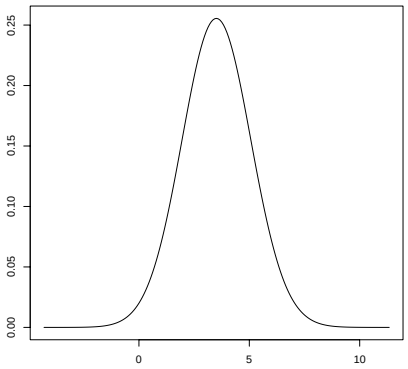
The graph of the full model for this example contains 40605 nodes, this makes the computation procedures heavier that for all other examples considered here. The computational time required to solve the model grows then to about 1 hour and 30 minutes on Machine 2. We have not run the model on Machine 1.



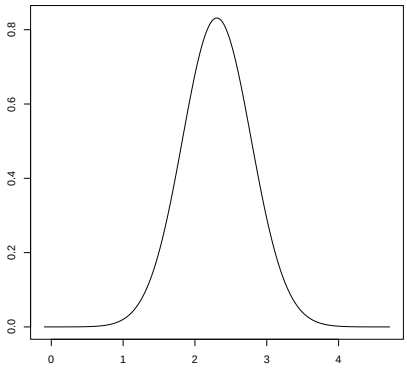
(a)



(b)



(c)



(d)

Figure 17: LGCP example: (a) posterior mean of the spatial effect $f_s(\cdot)$, (b)-(d) posterior marginals for μ, β_1 and β_2

3.11 A longitudinal study example - Forest health data

Our last example is a longitudinal study on forest health. The aim of the study if to identify potential factors influencing the health status of the trees. In addition to covariates characterising a tree and its stand, spatial and temporal information are also available. The example is taken from Kneib and Fahrmeir (2008), an earlier version of the data set is analysed in Kneib and Fahrmeir (2006).

Example 11 *The data have been collected annually in a visual forest health inventories between 1983 and 2004 in a northern Bavarian district. There are 83 observations plots within an area of around 15 squared kilometres.*

Every year, in some of the 83 observations plots the health status of the tree y_{it}, $i = 0, \ldots, 83$, $t = 0, \ldots, 21$, *is registered. Not all plots are observed every year, so the data set has in total* $n_d = 1796$ *observations. In the original data set there are 9 categories for tree health, anyway, here we consider only two: healthy or non-healthy. Together with the tree health status, several covariates are registered year after year at the different observation plot. All covariates are summarised in Table 1. Moreover the location of*

Covariate	Description
Age	age of the stand in years (continuous between 7 and 234 years)
elevation	elevation above the sea level (continuous, between 250 and 480 meters)
inclination	inclination of the terrain in percent (continuous between 0 and 1)
soil	depth of soil level (continuous, between 9 and 51 cm)
ph	ph-value in 0-2cm depth (continuous, between 3.28 and 5.05)
canopy	density of forest canopy in percent (continuous, between 0 and 1)
stand	type of stand (categorical, 3 categories)
fertilisation	fertilisation (categorical: yes or no)
humus	thickness of humus (categorical, 5 categories)
moisture	level of moisture (categorical, 3 categories)
saturation	base saturation (ordinal)

Table 1: Forest health data: description of covariates.

each registration plot s_i *is known. The spatial distribution of the locations is displayed in Figure 18.*

The likelihood of the data is binomial:

$$y_{it}|\eta_{it} \sim \textit{Bin}(p_{it})$$

with logit link

$$p_{it} = \frac{\exp(\eta_{it})}{1 + \exp(\eta_{it})} \quad i = 0, \ldots, 82, t = 0, \ldots, 21.$$

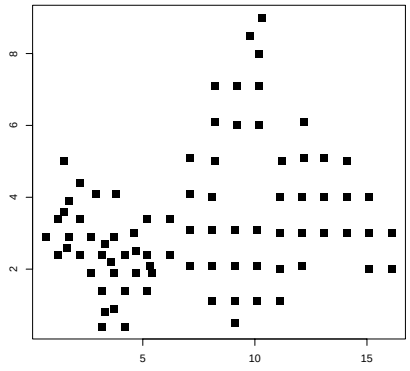


Figure 18: Forest health example: location of the 83 observation plots.

Following Kneib and Fahrmeir (2008) we model the latent variables as:

$$\eta_{it} = \mu + f_0(\textit{age}_{it}) + f_1(\textit{inclination}_i) + f_2(\textit{canopy}_{it}) + f_{\textit{time}}(t) + f_s(s_i) + f_u(s_i) + \boldsymbol{z}_{it}^T\boldsymbol{\beta} \quad (12)$$

where $f_0(\cdot)$, $f_1(\cdot)$, $f_2(\cdot)$ are the semiparametric effect of age of the tree, inclination and canopy of the location respectively, while $f_{\textit{time}}(\cdot)$ is the non parametric effect of time. Each semiparametric function is modelled as a RW2 with unknown precision parameter. The vector $\boldsymbol{z}_{it}^T$ includes all covariates in Table 2 not mentioned before which are assumed to have a linear effect. Finally $f_s(\cdot)$ and $f_u(\cdot)$ indicate the structured spatial effect and the unstructured one.

We models the spatial structured effect as the intrinsic GMRF in equation (7). We build the graph for such a model by considering two observation plots as neighbours if their distance is less than 1200 meters. The spatial unstructured effect is modelled as a series of uncorrelated Gaussian random variable.

We can cast the model in (12) in the general formulation in equation (2) by defining a new index $r = (i, t)$, $r = 0, \ldots, n_d - 1$, and rewriting the model as

$$\eta_r = \mu + f_0(\textit{age}_r) + f_1(\textit{inclination}_r) + f_2(\textit{canopy}_r) + f_{\textit{time}}(r) + f_s(s_r) + f_u(s_r) + \boldsymbol{z}_r^T\boldsymbol{\beta} \quad (13)$$

The above model has six precision hyperparameters $\boldsymbol{\theta} = (\lambda_0, \lambda_1, \lambda_2, \lambda_{\textit{time}}, \lambda_s, \lambda_u)$, each is given a vague Gamma prior.

We report part of the `ini` file which defines the model. We have omitted the definition of almost all covariates with linear effect.

```
[Forest damage]
type=problem
dir=results-%d
```

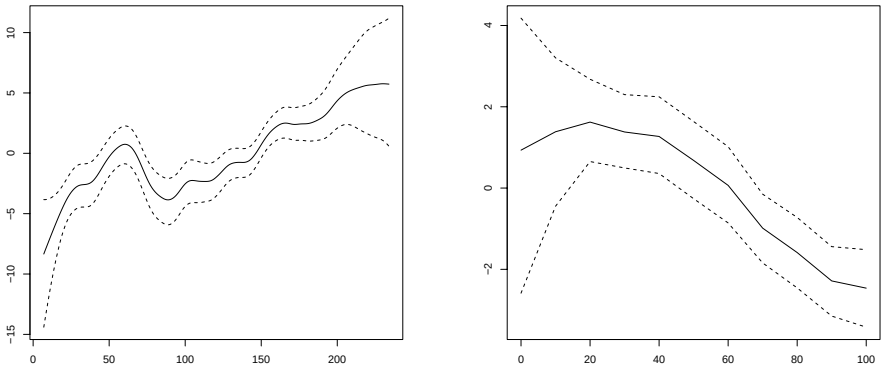
```
4
5  [unstruct term]
6  type=unstruct
7  n=1796
8  initial = 10
9  fixed=1
10
11 [Data]
12 type=data
13 likelihood=binomial
14 filename=damage.dat
15
16 [spatial]
17 type=ffield
18 model=besag
19 graph=forest.gra
20 covariates=spatial.covariate
21 diagonal = 0.00001
22 constraint = 1
23 initial = -3.346165
24 parameters = 1 0.001
25
26 [spatial-unstruct]
27 type=ffield
28 model=iid
29 n=83
30 covariates=spatial.covariate
31 diagonal = 0.00001
32 constraint = 1
33 initial = 7.324791
34 parameters = 1 1
35
36 [age]
37 type = ffield
38 model = rw2
39 covariates = age.covariate
40 locations=age.location
41 diagonal = 0.0001
42 constraint = 1
43 initial = 5.674807
44 parameters = 1 0.001
45 quantiles = 0.025 0.975
46
47 [canopy]
48 type = ffield
49 model = rw2
```

```
covariates = canopy.covariate
locations=canopy.location
diagonal = 0.0001
constraint = 1
initial = 13.763045
parameters = 1 0.001
quantiles = 0.025 0.975

[inclination]
type = ffield
model = rw2
covariates = inclination.covariate
n=47
diagonal = 0.0001
constraint = 1
initial = 6.422709
parameters = 1 0.001
quantiles = 0.025 0.975

[time]
type = ffield
model = rw2
covariates = year.covariate
locations=year.location
diagonal = 0.0001
constraint = 1
initial = 1.211905
parameters = 1 0.001
quantiles = 0.025 0.975

[common mean]
type=linear

[soil]
type = linear
covariates = soil.cov
.
.
.
[INLA parameters]
type = INLA
int_strategy = GMRFLib_AI_INT_STRATEGY_CCD;
gradient_finite_difference_step_len = 1.0e-2;
hessian_finite_difference_step_len = 1.0e-2;
```

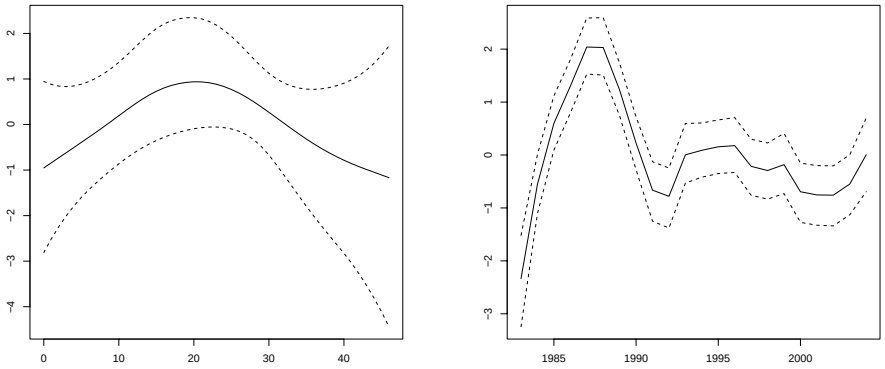
Notice that when using the `inla` program we treat all covariates, including space and time in the same way. All covariates files have the same structure.

Again we use the CCD strategy in order to integrate out the uncertainty about the hyperparameters $\boldsymbol{\theta}$. Given the high dimension of the hyperparameters space, the CCD strategy gives a much lower computation time if compared to the grid strategy. We have compared the results coming from the two integration strategies and the differences are irrelevant.

In Figure 19 the results about the semiparametric effects are displayed. The posterior mean is plotted within 0.025 and 0.975 posterior quantiles. The results agree very well with those found by Kneib and Fahrmeir (2008).



(a) Age of the tree (b) Canopy



(c) Inclination (d) Time

Figure 19: Results for the forest health example, semiparametric effect of covariates, posterior mean within 0.025 and 0.975 quantiles: age of the tree, panel (a), canopy, panel (b), inclination panel (c) and time panel (d).

The model runs in around 9 minutes on Machine 1 and around 4 minutes on Machine 2. Much of the time is used by the optimiser to find the maximum of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ and to

compute the Hessian at the modal configuration. When the hyperparameter space is high dimensional it is possible that the optimiser fails to succeed at a first attempt. The problem is usually solved by running the `inla` program again starting from different initial values for the hyperparameters. It is, usually, a good idea to start from the best configuration found during the previous run.

If one is interested in spatial prediction of tree health outside the observation plots, the spatial model in (7) is not very useful. We could instead use a second order random walk defined on a regular grid (Rue and Held, 2005, Sec 3.4.2) built as following. We divide the region of interest in $n_{row} \times n_{col}$ cells, with $n_{row} = 50$ and $n_{col} = 100$. We then build a new covariate file, *spatial −covariate−rw2.dat*, where, to each data point y_r are assigned two indexes n^r_{row} and n^r_{col} indicating its the location of the data on the $n_{row} \times n_{col}$ grid.

The code for the `ini` file substituting section *[spatial]* (lines 16-24) and *[spatial −unstruct]* (lines 26-34) is the following:

```
1 [ spatial ]
2 type=ffield
3 model=rw2d
4 covariates=spatial-covariate-rw2.dat
5 nrow=50
6 ncol=100
7 constraint=1
8 parameters = 1 0.001
9 initial= -1.570568
```

The new model has one hyperparameter less than the previous one since no spatial unstructured effect is present, but the number of nodes in the latent field $\boldsymbol{x}$ is increased, therefore running the new model will take longer time.

The results for the spatial effect in the new model is displayed in Figure 20. The non parametric effects of the other covariates do not change significantly.

Kneib and Fahrmeir (2008) propose to include in the model for the latent variable an interaction between the age of the tree and the calendar time, so that the model becomes:

$$\begin{aligned} \eta_{it} = \quad & \mu + f_1(\text{inclination}_i) + f_2(\text{canopy}_{it}) \\ & f_3(t, \text{age}_{it}) + f_s(s_i) + f_u(s_i) + \boldsymbol{z}_{it}^T \boldsymbol{\beta} \end{aligned} \tag{14}$$

where the spatial effect $f_s(\cdot)$ is modelled as in (7) and $f_4(\cdot)$ is the interaction effect between time and age of the tree modelled as a RW2d.

We can include the term $f_4(\cdot)$ in equation (14) in a similar way as we did earlier in this same example for the RW2d spatial effect. We just create a new covariate file, *year.age−covariate*, with the format

$$r \quad t \quad \text{age}_r$$

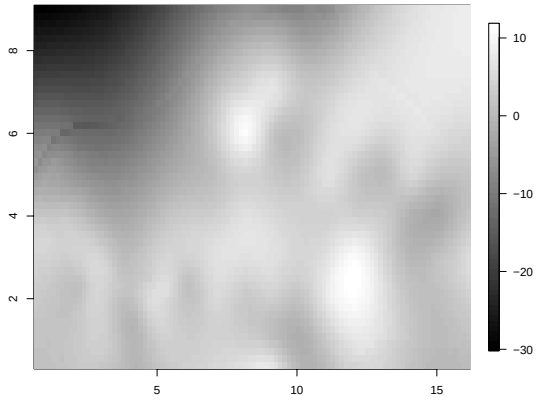


Figure 20: Posterior mean estimate for the spatial effect modelled as a RW2d

where both time and age are recorded, and delete from the `ini` on page 51 section *[age]* and *[time]* while adding the the following lines:

```
[year-age interaction]
type=ffield
model=rw2d
covariates=year.age-covariate
nrow=22
ncol=223
constraint=1
diagonal = 0.01
parameters = 1 0.01
initial= 2.025712
```

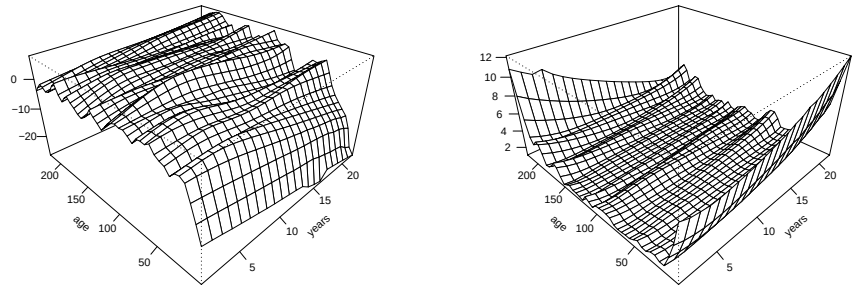
The new model has 5 hyperparameters and the total number of nodes in the latent field is 6939. We run the model on Machine 2 and the computation time was around 30 minutes using a CCD integration strategy.

The posterior mean and standard deviation of the interaction effect are displayed in Figure 21, panel (a) and (b) respectively.

References

Brezger, A., Kneib, T., and Lang, S. (2003). *BayesX: Software for Bayesian inference*. Department of statistics, University of Munich, version 1.1 edition. http://www.stat.uni-muenchen.de/∼lang/bayesx.

Chaudhuri, P. and Marron, J. S. (1999). SiZer for exploration of structures in curves. *Journal of the American Statistical Association*, 94:807–823.



(a) Posterior mean (b) Posterior standard deviation

Figure 21: Interaction effect between age of the tree and calendar time in Model (14). Panel (a) posterior mean, panel (b) posterior standard deviation.

Chib, S., Nardari, F., and Shepard, N. (2002). Markov chain Monte Carlo methods for stochastic volatility models. *Journal of Econometrics*, 108:281–316.

Erästö, P. (2005). *Studies in trend detection of scatter plots with visualization*. PhD thesis, Department of Mathematics and Statistics, University of Helsinki, Finland.

Fahrmeir, L. and Tutz, G. (2001). *Multivariate Statistical Modelling Based on Generalized Linear Models*. Springer-Verlag, Berlin, 2nd edition.

Hastie, T. J. and Tibshirani, R. J. (1990). *Generalized Additive Models*, volume 43 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Kammand, E. E. and Wand, M. P. (2003). Geoadditive models. *Journal of Royal Statistical Society C*, 52:1–18.

Kammann, E. E. and Wand, M. P. (2003). Geoadditive models. *Journal of the Royal Statistical Society, Series C*, 52(1):1–18.

Kandala, N. B., Lang, S., Klasen, S., and Fahrmeir, L. (2001). Semiparametric analysis of the socio-demographic and spatial determinants of undernutrition in two african countries. *Research in Official Statistics*, 1:81–100.

Kneib, T. and Fahrmeir, L. (2006). Structured additive regression for categorical space-time data: A mixed model approach. *Biometrics*, 62:109–118.

Kneib, T. and Fahrmeir, L. (2008). A space-time study on forest health. In Chandler, R. E. and Scott, M., editors, *Statistical Methods for Trend Detection and Analysis in the Environmental Sciences*. Wiley. (to appear).

Kneib, T., Lang, S., and Brezger, A. (2004). Bayesian semiparametric regression based on mcmc techniques: A tutorial. Technical report, Department of statistics, University of Munich.

Lin, X. and Zhang, D. (1999). Inference for generalized additive mixed models by using smoothing splines. *Journal of Royal Statistical Society B*, 61:381–400.

Martino, S. (2007). Approximate bayesian inference for multivariate stochastic volatility models. Technical report, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway.

Møller, J. and Waagepetersen, R. (2003). *Statistical inference and simulation for spatial point processes*, volume 100 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*, volume 104 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Rue, H. and Martino, S. (2006). Approximate Bayesian inference for hierarchical Gaussian Markov random fields models. *Journal of Statistical Planning and Inference*, 137:3177–3192.

Rue, H., Martino, S., and Chopin, N. (2007). Approximate bayesian inference for latent gaussian models using integrated nested laplace approximations. Statistics Report No. 1, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway.

Shephard, N. and Pitt, M. K. (1997). Likelihood analysis of non-Gaussian measurement time series. *Biometrika*, 84(3):653–667.

Taylor, S. J. (1986). *Modelling Stochastic volatility*. John Wiley.

Waagepetersen, R. P. (2006). An estimating function approach to inference for inhomogeneous Neyman-Scott processes. *Biometrics*, xx(xx):xx–xx. (to appear).

Wecker, W. E. and Ansley, C. F. (1983). The signal extraction approach to nonlinear regression and spline smoothing. *Journal of the American Statistical Association*, 78(381):81–89.

A Reference manual for the `inla` program

A.1 Structure of the `ini` file

The `ini` file describes the model and sets some additional parameters to be passed to the GMRFLib library. It divided in several sections. Each section starts with a tag written between squared brackets (*[tag]*) which is simply a user defined name for the section itself.

Each section contains the field ***type*** which determines the role of the section in the problem definition and also the structure of the section itself. The six different types of section are described in details below.

A.1.1 The *type=problem* section

This sections specifies some global parameters which are valid for the whole problem. It consists of the following fields:

dir: A string indicating the name of the directory where the results are stored. The directory is created when the `inla` program is run. The directory name can include `%d`

hyperparameters: A Boolean variable indicating whether or not to compute the marginals for the hyperparameters $\boldsymbol{\theta}$ of the model.

Default = 0

summary: A Boolean variable indicating whether or not to output a short summary of the posterior density for *all* the nodes in the GMRF $\boldsymbol{x}$. Currently the summary contains the posterior mean and standard deviation.

Default = 1

density: A Boolean variable indicating whether or not to output the marginal densities for *all* nodes in the latent GMRF $\boldsymbol{x}$.

Default = 1

quantiles: A list of maximum 10 quantiles, $p(0), p(1), \ldots$, to compute for each posterior marginal. The function returns, for each posterior marginal, the values $x(0), x(1), \ldots$ such that

$$\text{Prob}(X < x(p)) = p$$

Default: Empty

percentiles : A list of maximum 10 percentiles, $x(0), x(1), \ldots$, to compute for each posterior marginal. The function returns, for each posterior marginal, the probabilities $\text{Prob}(X < x(p))$.

Default: Empty

smtp: A string indicating which type of solver for sparse matrices should be used. The available choices are:

- *GMRFLib_SMTP_BAND* Lapack's band-solver. This is optimal for band matrices
- *GMRFLib_SMTP_TAUCS* The solver in the TAUCS-library. This is generic for all kind of sparse matrices.

Default: *GMRFLib_SMTP_TAUCS*

A.1.2 The *type=data* section

This section specifies the model for the likelihood of the data $\pi(y_i|\eta_i, \boldsymbol{\theta}_1)$ in equation (1). It consists of the following fields:

likelihood : A string indicating the name of the required likelihood model. The available choices are listed in Table 2.

prior: Prior distribution for the hyperparameters of the likelihood model $\boldsymbol{\theta}_1$. At the moment this is only used for the precision parameter λ_y of the *gaussian* likelihood which is assigned a Gamma(a, b) prior with mean a/b and variance a/b^2.

Default: *gamma*

initial : Initial value for $\log \lambda_y$.

parameters: Parameters a and b for the Gamma prior of the precision λ_y.

Default: $a = 1.0$ and $b = 0.001$

fixed: A Boolean variable indicating whether the hyperparameters of the likelihood model are fixed or random.

Default: 0

filename: The name of the file which contains the data for the model. The format of the file depends on the likelihood model chosen and is indicated in Table 2

A.1.3 The *type=unstruct* section

This section defines the model for the unstructured term u_i in equation (2). The `inla` program requires a section of ***type***=*unstruct* to always be present. It consists of the following fields:

prior: Name of the prior for the precision parameter λ_η. At the moment only the Gamma(a, b) prior is implemented.

Default: *gamma*

Model name	Distribution	Link function	Parameter $\boldsymbol{\theta}_1$	Input File format	Input File format (on a grid)
gaussian	$\mathcal{N}(\mu_i, \frac{\lambda_y^{-1}}{w_i})$	$\mu_i = \eta_i$	λ_y	$i \quad w_i \quad y_i$	$i \quad j \quad w_{ij} \quad y_{ij}$
poisson	$\text{Po}(E_i\lambda_i)$	$\lambda_i = \exp(\eta_i)$	-	$i \quad E_i \quad y_i$	$i \quad j \quad E_{ij} \quad y_{ij}$
binomial	$\text{Bin}(n_i, p_i)$	$p_i = \frac{\exp(\eta_i)}{(1+\exp(\eta_i))}$	-	$i \quad n_i \quad y_i$	$i \quad j \quad n_{ij} \quad y_{ij}$
stochvol	$\mathcal{N}(0, \sigma_i^2)$	$\sigma_i = \exp(\eta_i/2)$	-	$i \quad y_i$	-

Table 2: Likelihood models supported in the `inla` program.

parameters: Parameters a and b for the Gamma prior of the precision λ_η.

Default: $a = 1.0$ and $b = 0.001$

fixed: A Boolean variable indicating whether the precision parameter λ_η is fixed or random.

Default: 0.

initial : Starting value for $\log \lambda_\eta$

n: Length of the latent variable vector $\boldsymbol{\eta}$. Either ***n***, or ***nrow*** and ***ncol*** are required.

nrow: Number of rows of the latent variable vector $\boldsymbol{\eta}$. Either ***n***, or ***nrow*** and ***ncol*** are required.

ncol: Number of columns of the latent variable vector $\boldsymbol{\eta}$. Either ***n***, or ***nrow*** and ***ncol*** are required.

compute: A Boolean variable indicating whether or not the marginals for vector $\boldsymbol{\eta}$ have to be computed.

Default: 0 section

summary:A Boolean variable indicating whether or not to output a short summary of the posterior density for $\boldsymbol{\eta}$.

Default: ***compute***

density: A Boolean variable indicating whether or not to output the marginal densities for $\boldsymbol{\eta}$.

Default: ***compute***

quantiles: A list of maximum 10 quantiles, $p(0), p(1), \ldots,$ to compute for each node in $\boldsymbol{\eta}$.

Default: Empty

percentiles :A list of maximum 10 percentiles, $x(0), x(1), \ldots,$ to compute for each node in $\boldsymbol{\eta}$.

Default: Empty

A.1.4 The *type*= *ffield* type section

A section of ***type***= *ffield* specifies the model for one of the function f in equation (2). Hence, in a `ini` file there must be n_f sections of ***type***= *ffield* . Each ***type***= *ffield* section consists of the following fields:

model: A a string indicating the name of the chosen model. All available choices are listed in Table 3.

Model Type	Model Name	Parameters	Reference
Independent random noise	*iid*	precision λ_f	
Random Walk of order 1	*rw1*	precision λ_f	(Rue and Held, 2005, Ch. 3.3.1)
Random Walk of order 2	*rw2*	precision λ_f	(Rue and Held, 2005, Ch. 3.4.1)
First order Intrinsic GMRF on a irregular lattice	*besag*	precision λ_f	(Rue and Held, 2005, Ch. 3.3.2)
Continuous random walk	*crw2*	precision λ_f	(Rue and Held, 2005, Ch. 3.5)
Autoregressive of order 1 $x_t = \phi x_{t-1} + \epsilon_t$	*ar1*	precision λ_f $\kappa = \text{logit}\frac{\phi+1}{2}$	(Rue and Held, 2005, Ch. 1.1)
User defined precision matrix	*generic*	precision λ_f	(see Example 5)

Table 3: Models for the ***type***= *ffield* section implemented in the `inla` program.

prior: Name of the prior for the precision parameter λ_f. At the moment only the Gamma(a, b) prior is implemented (not in use if ***model***=*ar1*)

Default: *gamma*

parameters: Parameters a and b for the Gamma prior of the precision λ_f (not in use if ***model***=*ar1*)

Default: $a = 1.0$ and $b = 0.001$

initial : Starting value for $\log \lambda_f$ (not in use if ***model***=*ar1*)

prior0: Name of the prior for the precision parameter λ_f if ***model***=*ar1*. At the moment only the Gamma(a, b) prior is implemented

Default: *gamma*

prior1: Name of the prior for the precision parameter κ if ***model***=*ar1*. At the moment only the Gaussian$(0, \text{prec}_\kappa)$ prior is implemented

Default: *gaussian*

parameters0:Parameters a and b for the Gamma prior of the precision λ_f (**only** for ***model***=*ar1*)

Default: $a = 1.0$ and $b = 0.001$

parameters1: Parameter prec_κ for parameter κ (**only** for ***model***=*ar1*)

Default: $\text{prec}_k = 0.001$

initial0 : Starting value for $\log \lambda_f$ (**only** for ***model***=*ar1*)

initial1 : Starting value for κ (**only** for ***model***=*ar1*)

Hyperparameter	Prior distribution	Default param
Precision λ_f	Gamma(a, b) so that mean is a/b	$a = 1.0$ and $b = 0.001$
κ (only for AR1)	$\mathcal{N}(0, 1/\text{prec}_k)$	$\text{prec}_k = 0.001$

Table 4: Prior distributions for the hyperparameters

rankdef: A number indicating the rank deficiency of the user defined $\boldsymbol{Q}$ matrix (0nly used if ***model***=*generic*).

Default: 0.

fixed: A Boolean variable indicating whether the precision parameter λ_f is fixed or random.

Default: 0.

constraint: A Boolean variable indicating whether or not to impose a sum-to-zero constraint $\sum f_j = 0$

Default: 0.

diagonal: Additional constraint to add on the diagonal

Default: 0.

graph: The name of the file where the graph is stored (only if ***model***=*besag*)

n: Length m of vector $\boldsymbol{f}$. Only if ***model***=*rw1,rw2,crw2* and no ***locations*** is specified.

locations: The name of the file where the value of the covariate are stored, only if ***model***=*rw1,rw2* or *crw2*. If no file is specified the covariate are assumed to take values in $\{0, 1, \ldots, m-1\}$.

cyclic : A Boolean variable specifying whether the model is cyclical, only if ***model***=*rw1,rw2* and no ***locations*** is specified.

compute: A Boolean variable indicating whether or not the marginals for vector $\boldsymbol{f}$ have to be computed.

Default: 1

summary:A Boolean variable indicating whether or not to output a short summary of the posterior density for $\boldsymbol{f}$.

Default: ***compute***

density : A Boolean variable indicating whether or not to output the marginal densities for $\boldsymbol{f}$.

Default: ***compute***

quantiles : A list of maximum 10 quantiles, $p(0), p(1), \ldots$, to compute for each node in $\boldsymbol{f}$.

Default: Empty

percentiles :A list of maximum 10 percentiles, $x(0), x(1), \ldots$, to compute for each node in $\boldsymbol{f}$.

Default: Empty

A.1.5 The *type=linear* section

A section of ***type***=*linear* specifies the model for one of the element β_k of vector $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_{n_\beta - 1})$ in equation (2). Hence a `ini` file will contain n_β sections of ***type***=*linear*. Each section consists of the following fields:

covariates : Name of the file where covariate are stored. If empty, then all covariates are assumed to be 1.

precision : Fixed precision for the Gaussian prior distribution of β.

Default: 0.001

compute: A Boolean variable indicating whether or not the marginal for β_k has to be computed.

Default: 1

summary:A Boolean variable indicating whether or not to output a short summary of the posterior density for β_k.

Default: ***compute***

density : A Boolean variable indicating whether or not to output the marginal densities for β_k.

Default: ***compute***

quantiles : A list of maximum 10 quantiles, $p(0), p(1), \ldots$, to compute for each node in β_k.

Default: Empty

percentiles :A list of maximum 10 percentiles, $x(0), x(1), \ldots$, to compute for each node in β_k.

Default: Empty

A.1.6 The *type=INLA* section

This section is optional, it specifies parameters to be passed to the GMRFLib library. It is possible to specify here all parameters in the *GMRFLib_ai_param_tp* structure. We describe here the most used and useful ones, for more details see the on-line documentation for the GMRFLib library: `http://www.math.ntnu.no/~hrue/GMRFLib/doc/html/`

strategy : The strategy used to compute approximations to the posterior marginals $\pi(x_i|\boldsymbol{y}, \boldsymbol{\theta})$. The three main choice are:

- *GMRFLib_AI_STRATEGY_GAUSSIAN*: computes the Gaussian approximation
- *GMRFLib_AI_STRATEGY_MEANSKEWCORRECTED_GAUSSIAN*: computes the simplified Laplace approximation.
- *GMRFLib_AI_STRATEGY_ADAPTIVE*: Computes the full Laplace approximation.

The three approximation types are described in Rue et al. (2007).

Default: *GMRFLib_AI_STRATEGY_MEANCORRECTED_GAUSSIAN*

int_strategy : The strategy used to integrate out the hyperparameters $\boldsymbol{\theta}$ when computing $\widetilde{\pi}(x_i|\boldsymbol{y})$. There are two possible choices:

- *GMRFLib_AI_INT_STRATEGY_GRID* (or *grid*) : Use a grid strategy, slower and somehow more accurate.
- *GMRFLib_AI_INT_STRATEGY_CCD* (or *ccd*) : Use a central composite design strategy, faster and especially useful for problems with higher dimension of the hyperparameter vector $\boldsymbol{\theta}$.

Both strategies are described in Rue et al. (2007).

Default: *GMRFLib_AI_INT_STRATEGY_GRID*

dz : Step length for the integration procedure, only if ***int_strategy*** =*grid*.

Default: 1

diff_logdens : Only used if ***int_strategy*** = *grid*. Threshold for accepting a configuration.

Default: 2.5

skip_configurations : Only used if ***int_strategy*** = *grid*. Skip fill-in configuration larger than a non-accepted one.

Default: *GMRFLib_TRUE*

gradient_finite_difference_step_len (or ***h***): Step length to compute the gradient of $\widetilde{\pi}(\boldsymbol{\theta})$.

Default: 1.0e-4

hessian_finite_difference_step_len (or ***h***): Step length to compute the Hessian of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ at the mode.

Default: 1.0e-4

interpolator Type of interpolator used to compute marginals for each hyperparameter $\widetilde{\pi}(\theta_m|\boldsymbol{y})$, the available choices are:

- *GMRFLib_AI_INTERPOLATOR_AUTO*: Chose interpolation type based on the integration strategy.
  If ***int_strategy*** =*grid*, then choose *GMRFLib_AI_INTERPOLATOR_WEIGHTED_DISTANCE*.
  If ***int_strategy*** =*ccd*, then the choice is *GMRFLib_AI_INTERPOLATOR_CCD*
- *GMRFLib_AI_INTERPOLATOR_LINEAR*: Linear interpolation using the $(M + 1)$ nearest points, where M is the dimension of the hyperparameters space.
- *GMRFLib_AI_INTERPOLATOR_QUADRATIC*: Quadratic interpolation using the $(M+1)$ nearest points.
- *GMRFLib_AI_INTERPOLATOR_WEIGHTED_DISTANCE*: Linear interpolation using weighted distance.
- *GMRFLib_AI_INTERPOLATOR_CCD*: Special interpolation for the CCD integration scheme.

The interpolations are described in Martino (2007).

A.2 Format of the input files

There are five type of input files which can be read from the `inla` program: the data file, the covariate file, the covariate locations type, the graph file and the $\boldsymbol{Q}$-matrix file, each with its own format required. The formats have been already presented in different examples but are all collected here.

Data file The format of the data file depends on the likelihood model chosen and on whether the data are collected on a grid or not. The format of the data file is displayed in Table 2.

Covariate and location file Each covariate has to be stored in a separate file. The format of the file depends on whether the covariate is assumed to have linear or non-linear effect:

Covariates with linear effect: The value of the covariate is simply stored in a file with n_η columns each row having the format:

$$i \quad z_i$$

where $i = 0, \ldots, n_\eta - 1$ and z_i is the value of the covariate for node i.

Covariates with non-linear effect: Let $c \in \boldsymbol{C}$ and $\boldsymbol{C} = \{c^{(0)} < c^{(1)} < \cdots < c^{(idx)} < \cdots < c^{(m-1)}\}$. That is, covariate c takes one of the m values in the ordered vector $\boldsymbol{C}$. The file storing covariate c has n_η row, each with the following format:

$$i \quad (idx)_i$$

where $i = 0, \ldots, n_\eta - 1$ and $(idx)_i$ is the position of the observed value c_i in the vector $\boldsymbol{C}$. If the values in $\boldsymbol{C}$ are different from $0, 1, \ldots$ then another file (the locations file) of m rows, is necessary to store the values of $\boldsymbol{C}$. A short example will be useful:

Example: *Let* $n_\eta = 5$ *and* $\boldsymbol{C} = \{9, 10, 11\}$. *Moreover assume that the observed covariate values are* $c_0 = 10$, $c_1 = 9$, $c_2 = 11$, $c_3 = 9$ *and* $c_4 = 10$. *Then the covariate file will be as following*

$$\begin{matrix} 0 & 1 \\ 1 & 0 \\ 2 & 2 \\ 3 & 0 \\ 4 & 1 \end{matrix}$$

We would need also a file storing the values in $\boldsymbol{C}$*:*

$$\begin{matrix} 9 \\ 10 \\ 11 \end{matrix}$$

Graph file The graph file contains information on the neighbourhood structure of the spatial effect We describe the required format for such a file using a small example. Let the file ***graph***.*dat*, relative to a small graph, be

```
1  5
2  0  1  1
3  1  2  0  2
4  2  3  1  3  4
5  3  1  2
6  4  1  2
```

Line 1 declares the total number of nodes in the graph, then, in lines 2-6 each node is described. For example, line 4 states that node 2 has 3 neighbours and these are nodes 1, 3 and 4. This is the same format used in the GMRFLib library.

$\boldsymbol{Q}$**-matrix file** This file is only needed if the field ***model*** in a *ffield* -type section is defined as *generic*. The file should contain all non-zero entries of the user specified precision matrix $\boldsymbol{Q}$ in the following format

$$i \quad j \quad \boldsymbol{Q}_{ij}$$

where i and j are the row and column index and $\boldsymbol{Q}_{ij}$ is the corresponding entry in the precision matrix.

A.3 Some possible problems and solutions

1. The `inla` function checks that all entries in the `ini` file are used while building the models, so an error message like

   ```
   inla_build: [ZAMBIA.ini] contain[1] unused entries. PLEASE CHECK
   ```

   probably means that some of the fields in the `ini` file have been misspelled.

2. In our experience the most common problems with the `inla` function comes from the optimisation procedure and the numerical computation of the Hessian of $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ at the modal configuration.

   The optimiser might not converge, thus producing an error message like:

```
GMRFLib version 3.0-0-snapshot, has received error no [12]
        Reason     : The Newton-Reason optimiser did not converge
        Function   : GMRFLib_optimize_store
        File       : optimize.c
        Line       : 460
        RCSId      : $Id: optimize.c,v 1.37 2007/07/13 11:49:41
            hrue Exp $
```

Usually restarting the `inla` function assigning different starting values for the hyperparameters vector $\boldsymbol{\theta}$ (field ***initial***), will solve the problem.

3. Another error which might happen is that the computed numerical Hessian for $\log \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ in not positive definite. This produces the following error message:

```
GMRFLib version 3.0-0-snapshot, has received error no [2]
        Reason     : Matrix is not positive definite
        Message    : Condition 'gsl_vector_get(eigen_values,
            (unsigned int) i) > 0.0' is not TRUE
        Function   : GMRFLib_ai_INLA
        File       : approx-inference.c
        Line       : 2689
        RCSId      : $Id: approx-inference.c,v 1.372 2007/09/06
            21:38:26 hrue Exp $
```

To solve this problem it is usually enough to increase the step length used to numerically compute the Hessian and the gradient. These quantities can be re-defined in the ***type***=*INLA* section by using the parameters ***gradient_finite_difference_step_len*** and ***hessian_finite_difference_step_len*** .