NORGES TEKNISK-NATURVITENSKAPELIGE
UNIVERSITET

Approximate Bayesian Inference for Multivariate Stochastic Volatility Models

by

Sara Martino

PREPRINT
STATISTICS NO. 1/2008

NORWEGIAN UNIVERSITY OF SCIENCE AND
TECHNOLOGY
TRONDHEIM, NORWAY

This preprint has URL http://www.math.ntnu.no/preprint/statistics/2008/S1-2008.pdf

Sara Martino has homepage: http://www.math.ntnu.no/∼martino
E-mail: martino@math.ntnu.no
Address: Department of Mathematical Sciences, Norwegian University of Science and Technology, N-7491
Trondheim, Norway.

Approximate Bayesian Inference for Multivariate Stochastic Volatility Models

Sara Martino
Department of Mathematical Sciences
NTNU, Norway

October 2007

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 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

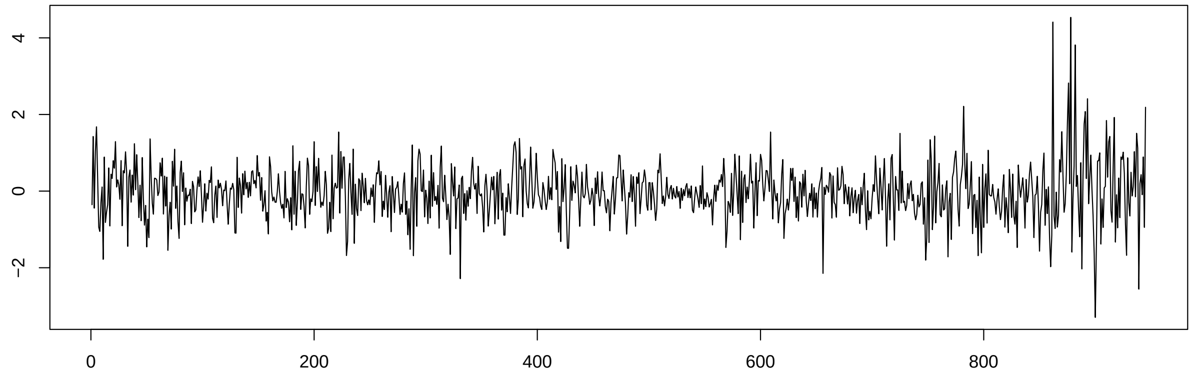


Figure 1: Log-returns of Pound-dollar daily exchange rate from October 1st, 1981 to June 28th,1985.

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, $\boldsymbol{y} = \{y_1, \dots, y_n\}$, 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 $\boldsymbol{h} = \{h_1, \dots, h_n\}$ is an auto regressive process of order 1 (AR1). The model is then defined as:

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

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

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 $\pi(h_1, \dots, h_n, \mu)$, 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

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

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

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:

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

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)

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

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)

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

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)

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

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) .

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

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 $(\phi_{11}, \phi_{12}, \phi_{22}, \rho_\eta, \tau_{\eta_1}, \tau_{\eta_2})$.

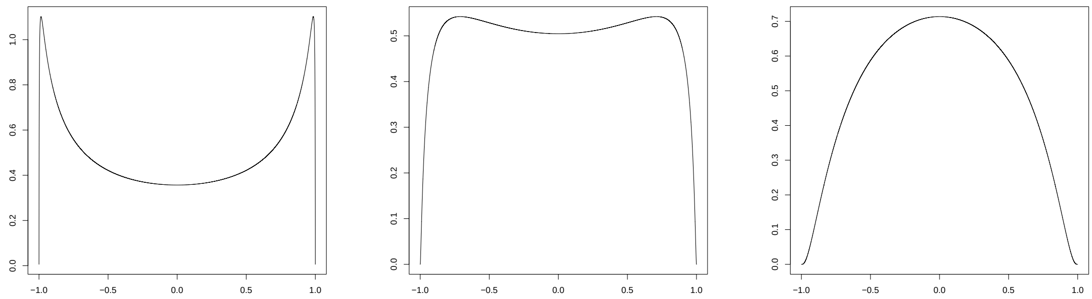
For computational reasons, it is convenient, when applying INLA, that all hyperparameters are defined over the whole real line. Hence, when the original parameters in the model are constrained, we consider a function of them.

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 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 $\rho_\eta = f(\rho^*_\eta)$
- $\tau_{\eta_i} \sim \text{Gamma}(0.25, 0.025)$ 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 $\boldsymbol{\theta}_2$:

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

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 hyperparameter 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 $i = N, \dots, 1$. Multiplying each side with x_j $j \geq i$, and taking expectation, we obtain

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

where $\mathbf{\Sigma}$ $(= \boldsymbol{Q}^{-1})$ 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.

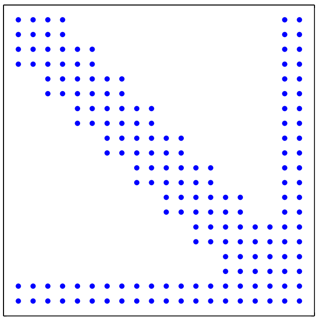


Figure 3: Non zero structure of the precision matrix for a bidimensional AR1 model with unknown mean

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 $t = 1 \dots, n_d$

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

where

$$\boldsymbol{C}_t = -\begin{bmatrix} \frac{\partial^2 g_t(\boldsymbol{x}_t)}{\partial x_{t1}^2} & \frac{\partial^2 g_t(\boldsymbol{x}_t)}{\partial x_{t1}\partial x_{t2}} \\ \frac{\partial^2 g_t(\boldsymbol{x}_t)}{\partial x_{t1}\partial x_{t2}} & \frac{\partial^2 g_t(\boldsymbol{x}_t)}{\partial x_{t2}^2} \end{bmatrix}_{\boldsymbol{x}_t = \boldsymbol{m}_t^0}$$

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

$$\begin{bmatrix} \boldsymbol{C}_1 & \boldsymbol{0} & & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C}_2 & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \vdots & & & & \\ \boldsymbol{0} & & \dots & \boldsymbol{C}_n & \boldsymbol{0} \\ \boldsymbol{0} & & & \dots & \boldsymbol{0} \end{bmatrix}, \tag{14}$$

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 $\boldsymbol{b}^T = (\boldsymbol{b}_1^T, \boldsymbol{b}_2^T, \dots, \boldsymbol{0})$. We obtain a Gaussian approximation with precision $\boldsymbol{Q}+$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.

Algorithm 1 Computing the Gaussian approximation $\pi_G(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$

```
 1: Given a value for $\boldsymbol{\theta}$ and an initial guess $\boldsymbol{m}^{(0)}$
 2: iter $= 0$, diff $= 10$
 3: while diff $> \alpha$ do
 4:     for $t = 1$ to $n$ do
 5:         Compute $\boldsymbol{b}_t$ and $\boldsymbol{C}_t$ using (13)
 6:     end for
 7:     Solve $(\boldsymbol{Q} + \text{diag}(\boldsymbol{C}))\boldsymbol{m}^{(1)} = \boldsymbol{b}$
 8:     Compute diff $=$ a distance measure between $\boldsymbol{m}^{(0)}$ and $\boldsymbol{m}^{(1)}$
 9:     Set $\boldsymbol{m}^{(0)} = \boldsymbol{m}^{(1)}$
10: end while
11: Return $\boldsymbol{x}_G = \boldsymbol{m}^{(0)}$ and $\boldsymbol{Q}_G = (\boldsymbol{Q} + \text{diag}(\boldsymbol{C}))$
```

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

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{y})\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \tag{15}$$

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

$$\widetilde{\pi}(\theta_m|\boldsymbol{y}) = \int \widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y}) d\boldsymbol{\theta}_{-m} \quad m = 1, \dots, M \tag{17}$$

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 $t = 1, 2, \dots$ and $i = 1, 2$. The general strategy is in Algorithm 2: first, select a set of configurations $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K\}$ 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 $x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y}$.

Algorithm 2 INLA strategy for computing $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$

```
Select a set $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K\}$
for $k = 1$ to $K$ do
    Compute $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$
    Compute $\widetilde{\pi}(x_{ti}|\boldsymbol{\theta}_k, \boldsymbol{y})$ as a function of $x_{ti}$
end for
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 $ti$
```

For Algorithm 2 to be operative we should first solve two tasks:

1. how to select a (possibly small) set of points $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K\}$
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.

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

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, $\boldsymbol{z} = \boldsymbol{0}$
    for dir in $\{-1, 1\}$ do
        while $\log\widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{0})|\boldsymbol{y}) - \log\widetilde{\pi}(\boldsymbol{\theta}(\boldsymbol{z})|\boldsymbol{y}) < \delta_\pi$ do
            $\boldsymbol{z} = \boldsymbol{z} + \texttt{dir} * \boldsymbol{1}_i$
            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
```

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 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.

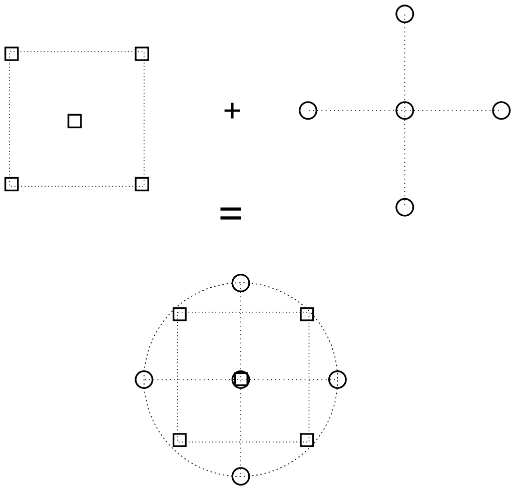


Figure 4: Location of points in a CCD design for $M = 2$. The squares are factorial points (design-points) and the circles are the additional "circle" points.

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 integration 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:

$$\begin{gathered} \log \widetilde{\pi}_{GG}(\boldsymbol{x}_{-ti}|x_{ti}, \boldsymbol{y}, \boldsymbol{\theta}_k)\Big|_{\boldsymbol{x}_{-ti}=\mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-ti}|x_{ti},\boldsymbol{\theta}_k,\boldsymbol{y})} \qquad \propto \\ \tfrac{1}{2}\log|\boldsymbol{Q}_{[-ti,-ti]} + \text{diag}(\boldsymbol{C}(x_{ti}, \boldsymbol{\theta}_k))| = f(x_{ti}) \end{gathered} \tag{23}$$

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 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 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 $(\boldsymbol{Q}_{[-ti,-ti]} + \text{diag}[\boldsymbol{C}(\mu_{ti}, \boldsymbol{\theta}_k)])$ can be computed at almost no extra costs as

$$f(\mu_{G_{ti}}) = \frac{1}{2}\log|\boldsymbol{Q}_{[-ti,-ti]} + \text{diag}[\boldsymbol{C}(\mu_{G_{ti}}, \boldsymbol{\theta}_k)]| = \frac{1}{2}\log|\boldsymbol{Q}_G| + \log\sigma_{G_{ti}} \tag{24}$$

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

$$x^s_{ti} = \frac{x_{ti} - \mu_{G_{ti}}(\boldsymbol{\theta}_k)}{\sigma_{G_{ti}}(\boldsymbol{\theta}_k)}$$

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

$\log \mathcal{B}(i, j)$	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

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

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

$$\widetilde{\pi}_G(\boldsymbol{\theta}|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{\theta}^*, \boldsymbol{\Sigma}) \tag{25}$$

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 interactive 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

$$\begin{aligned} y_t &= \exp(h_t/2)\epsilon_t, \quad t = 1,\dots,n, \quad \epsilon_t \sim t_7. \\ h_t &= 0.1 + 0.53(h_{t-1} - 0.1) + \eta_t, \quad t = 1,\dots,n, \quad \eta_t \sim \mathcal{N}(0, 1/2.3). \end{aligned} \tag{28}$$

The simulated time series is displayed in Figure 5. Note that the Student-t distribution allows for quite extreme values of the returns.

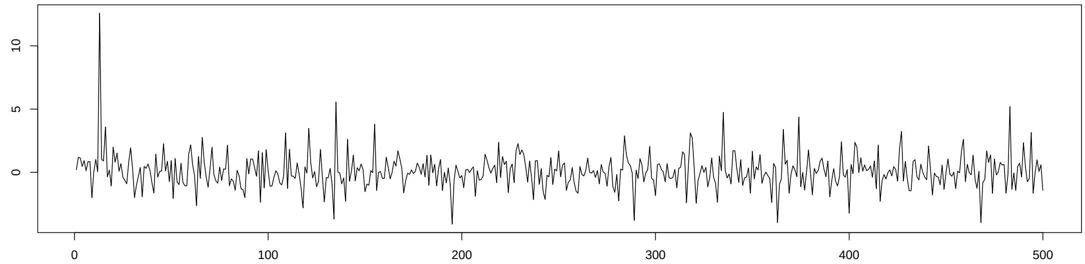
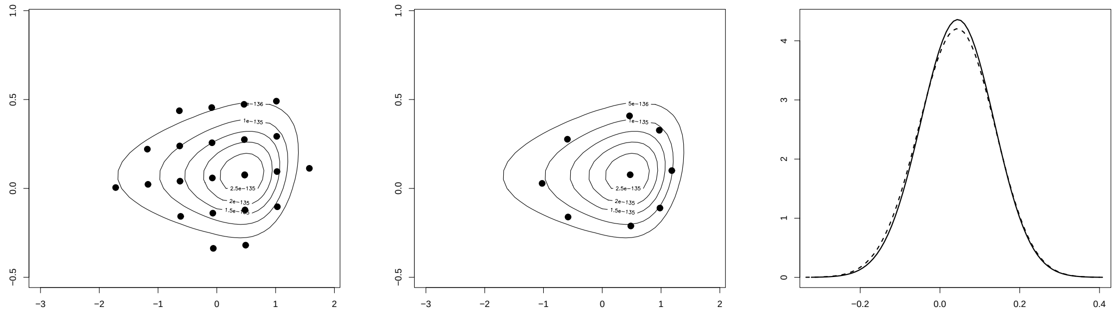


Figure 5: Time series of returns simulated from model (28)

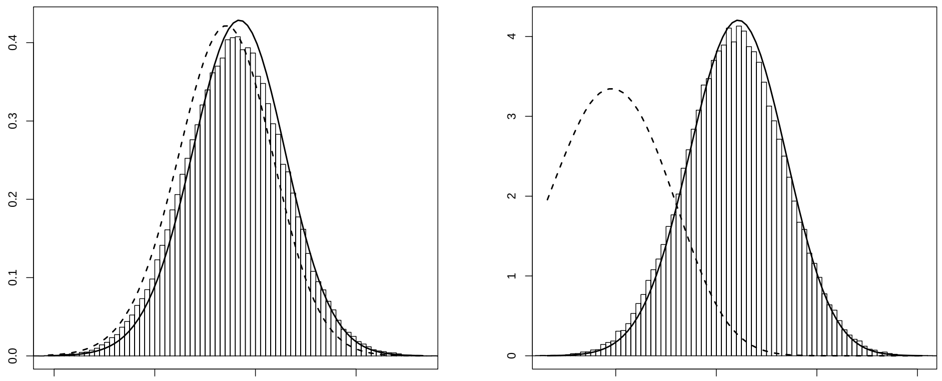
We first fit $\mathcal{M}_1$ to the simulated data. Following Algorithm 2, our first task is to locate a set of points in the hyperparameters space, $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1,\dots,\ \boldsymbol{\theta}_K\}$, where to compute $\widetilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$ 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.



(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)

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 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.

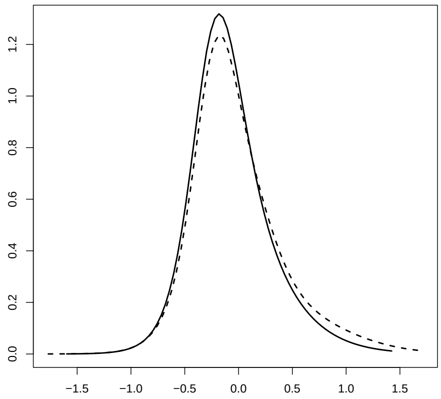
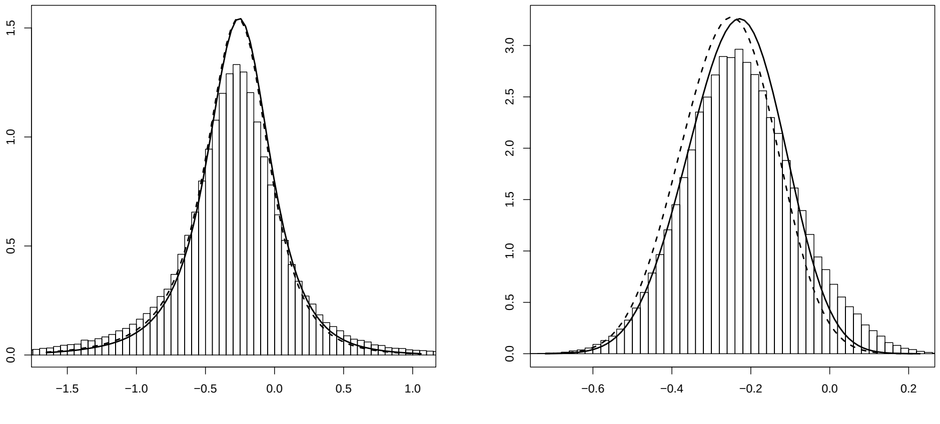


Figure 8: $\mathcal{M}_2$, simulated data example: approximation of $\pi(x_t|\boldsymbol{y})$ computed via the grid integration strategy (solid line) and the CCD integration strategy (broken line).

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 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.

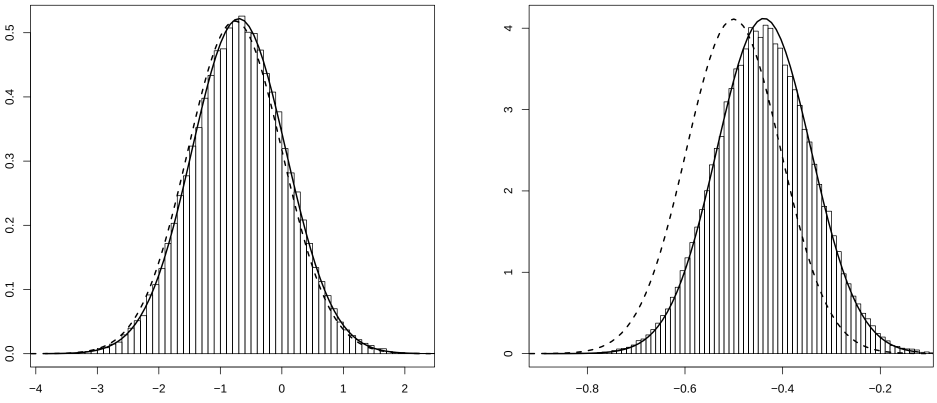


(a) $\pi(h_t|\boldsymbol{y})$

(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).

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).

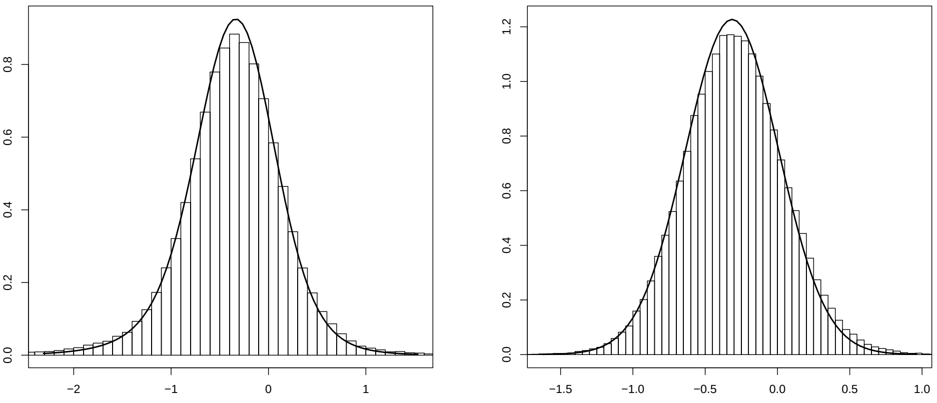
In our second experiment the hyperparameter vector $\boldsymbol{\theta}$ is random but only the first 50 data of the simulated time series are considered. Decreasing the number of data makes the MCMC algorithm run much faster and mix better. On the other side, the approximation problems are easier when 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 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 corresponds 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

	$\mathcal{M}_1$: 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.



(a) $\pi(h_t|\boldsymbol{y})$

(b) $\pi(\mu|\boldsymbol{y})$

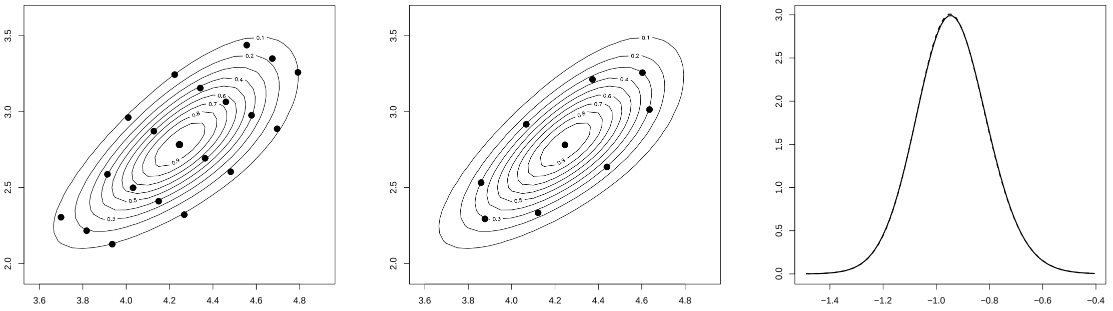
Figure 11: Simulated data, model $\mathcal{M}_2$ considering only 50 data: Gaussian approximation (broken line), improved approximation (solid line) and MCMC density estimate (histogram).

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.



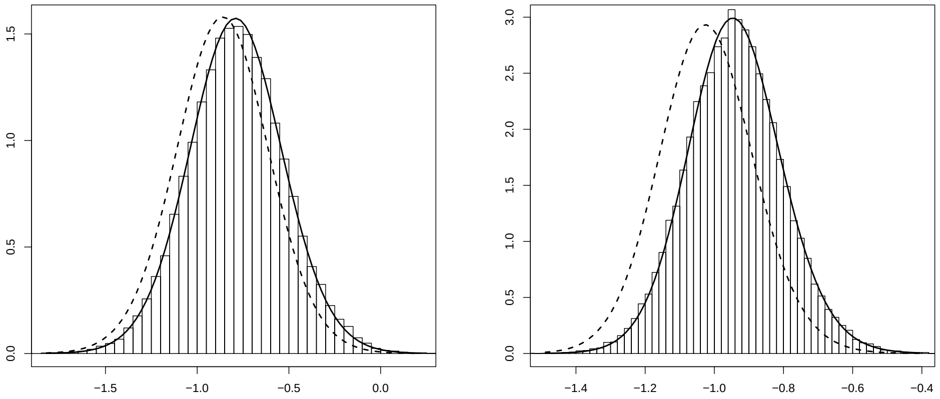
(a) Grid strategy

(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



(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).

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.

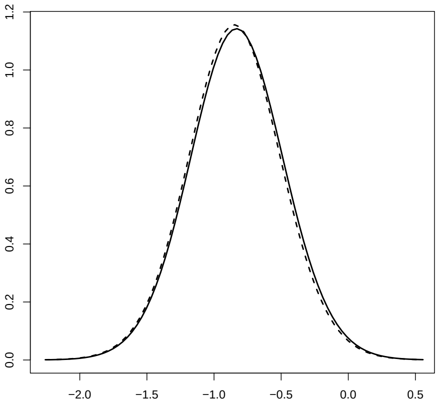
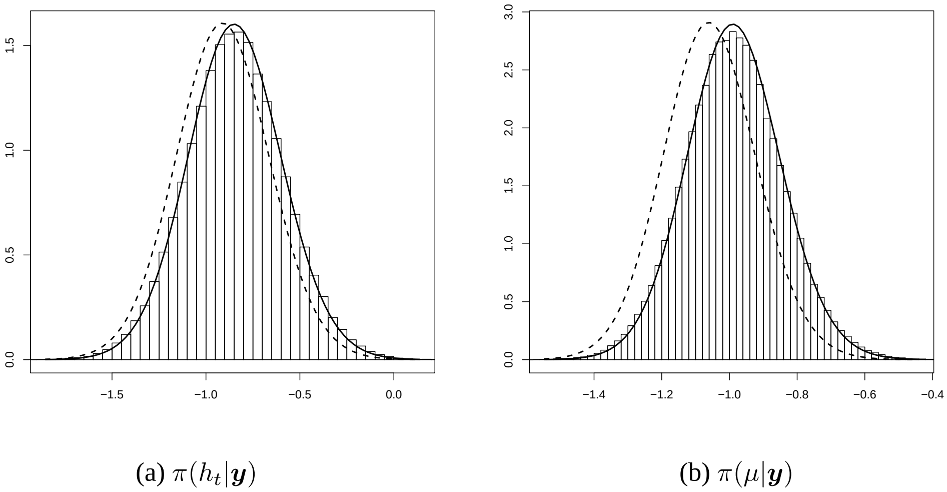


Figure 14: $\mathcal{M}_2$, real data example: approximation of $\pi(x_t|\boldsymbol{y})$ computed via the grid integration strategy (solid line) and the CCD integration strategy (broken line).

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 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.



(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).

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 approximations 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.

	$\mathcal{M}_1$: 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.

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 7, 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 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 $\widetilde{\pi}(\boldsymbol{y})$.

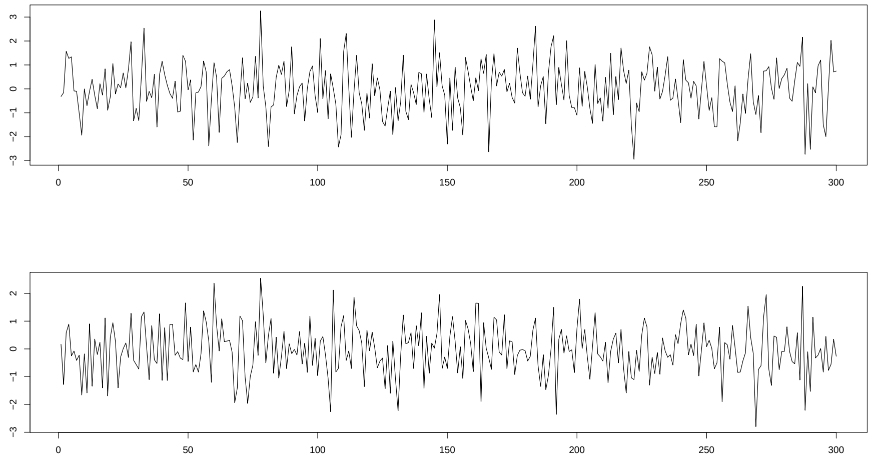


Figure 16: Simulated bivariate time series.

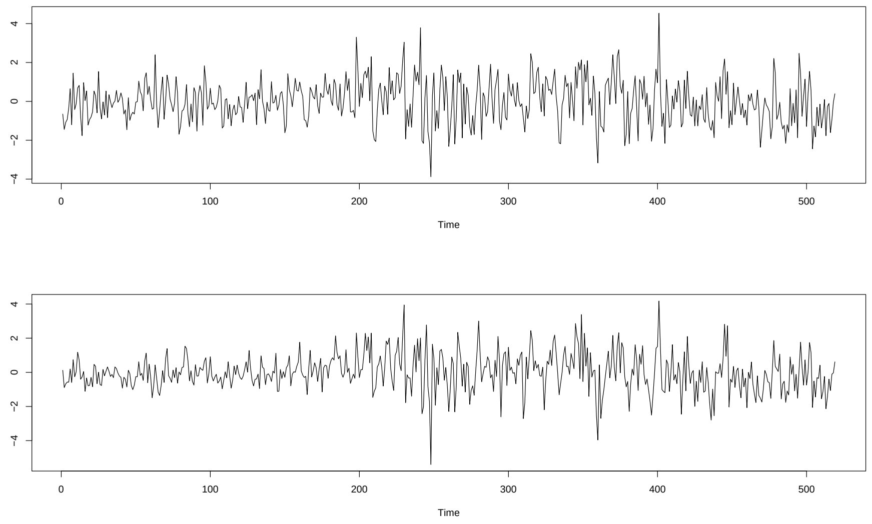


Figure 17: Time series for Australian/US Dollar (upper) and New Zeland/US Dollar (lower) exchange rate returns

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.

	N. of	Simulated Data		Real Data	
	hyperparam.	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.

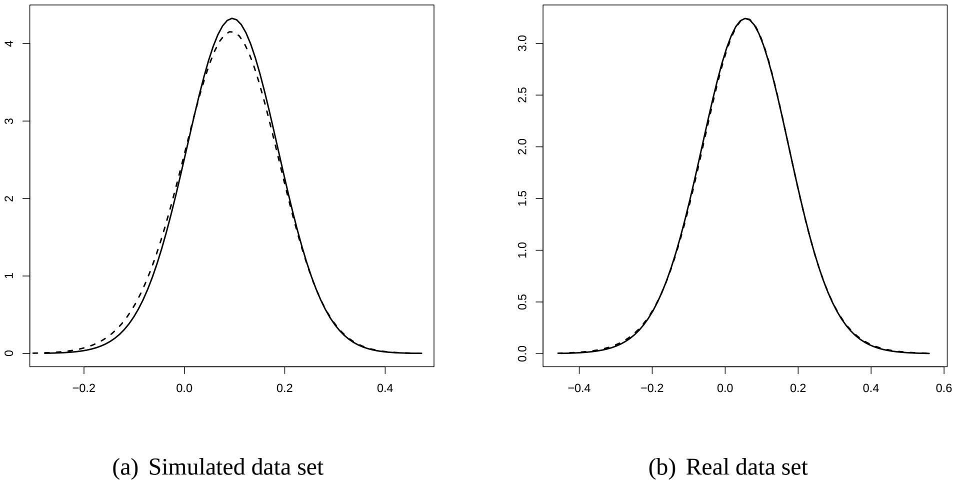
	Variance Equation						Mean Equation	
	$\log \kappa_1$	$\log \kappa_2$	ϕ_{12}	ρ^*_η	$\log \tau_{\eta 1}$	$\log \tau_{\eta 2}$	ρ^*_ϵ	ν^*
Model 1	3.998	4.174	-	-	3.188	2.700	-	-
	(0.333)	(0.351)	-	-	(0.449)	(0.505)	-	-
Model 2	3.391	3.588	-	-	3.792	2.803	1.993	-
	(0.566)	(0.631)	-	-	(0.538)	(0.731)	(0.097)	-
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.

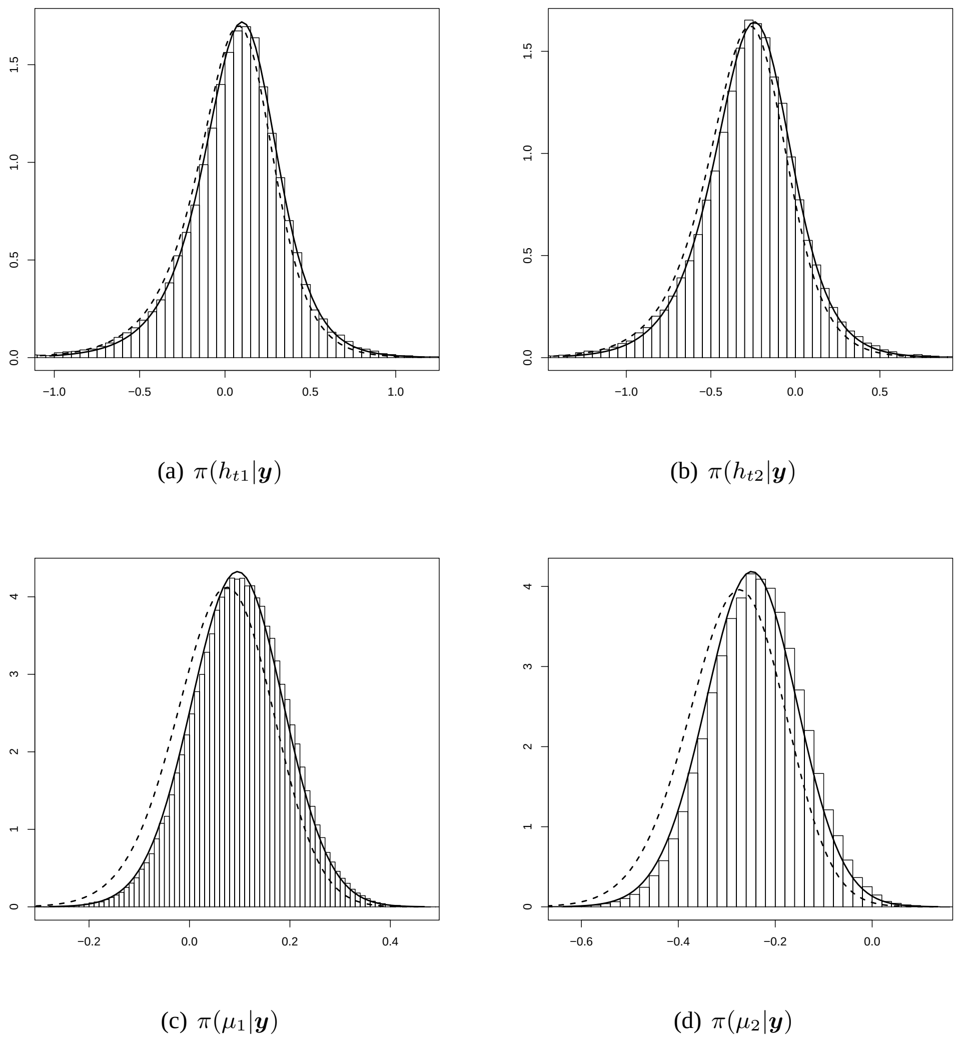


(a) Simulated data set

(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.



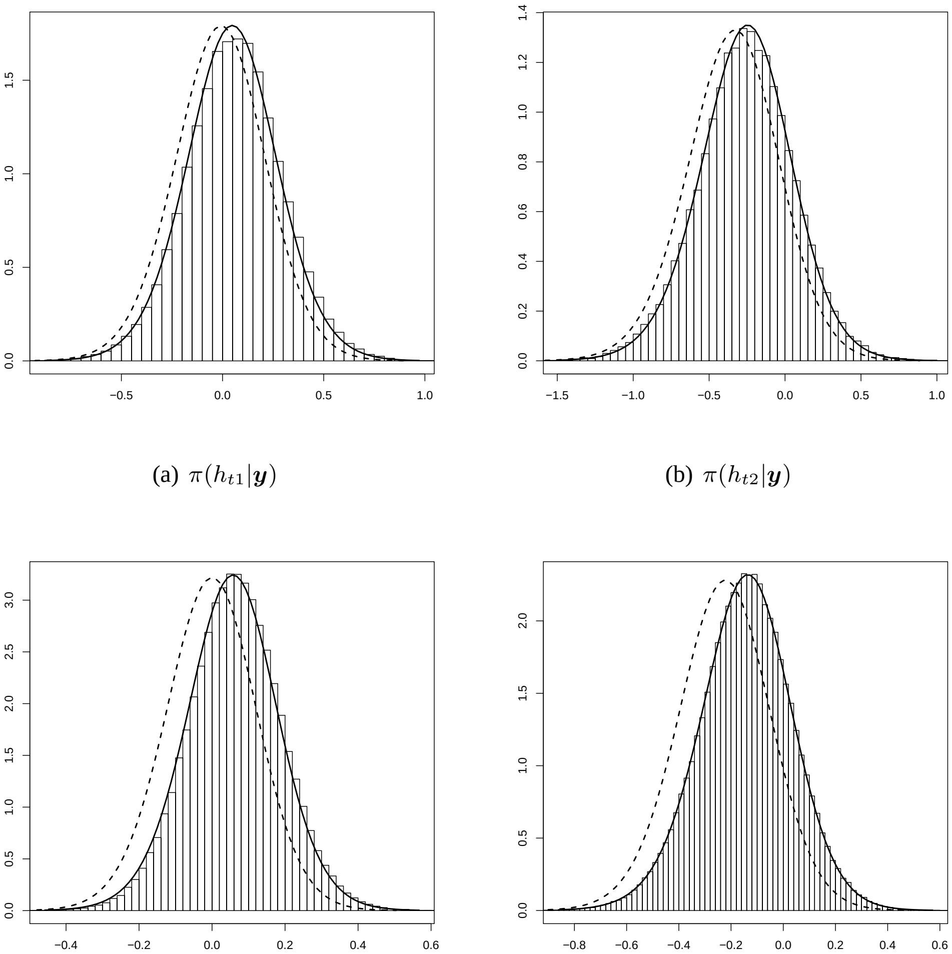
(a) $\pi(h_{t1}|\boldsymbol{y})$

(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) $\pi(\mu_1|\boldsymbol{y})$

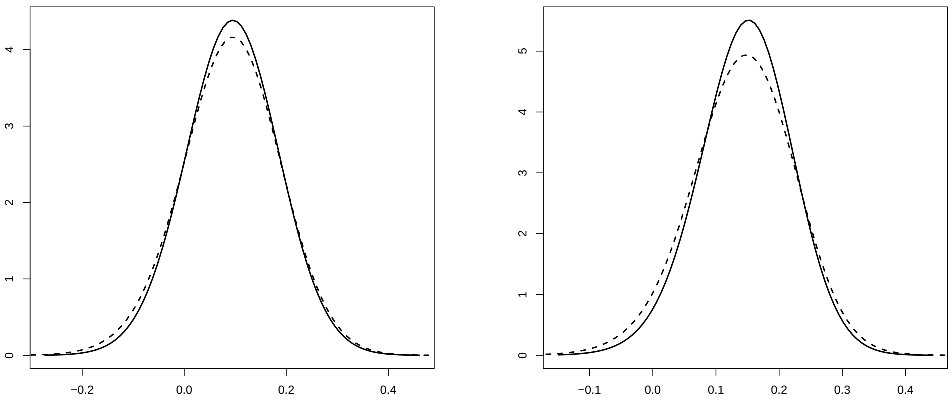
(d) $\pi(\mu_2|\boldsymbol{y})$

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.

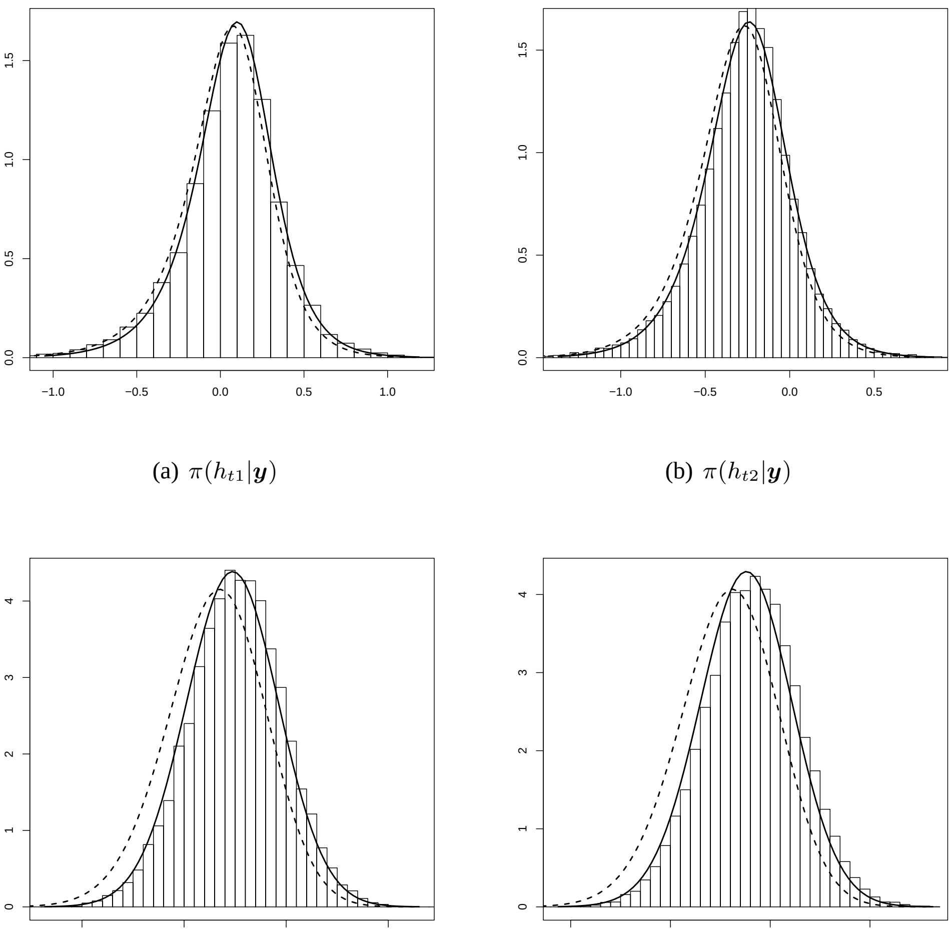


(a) Simulated data set

(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.



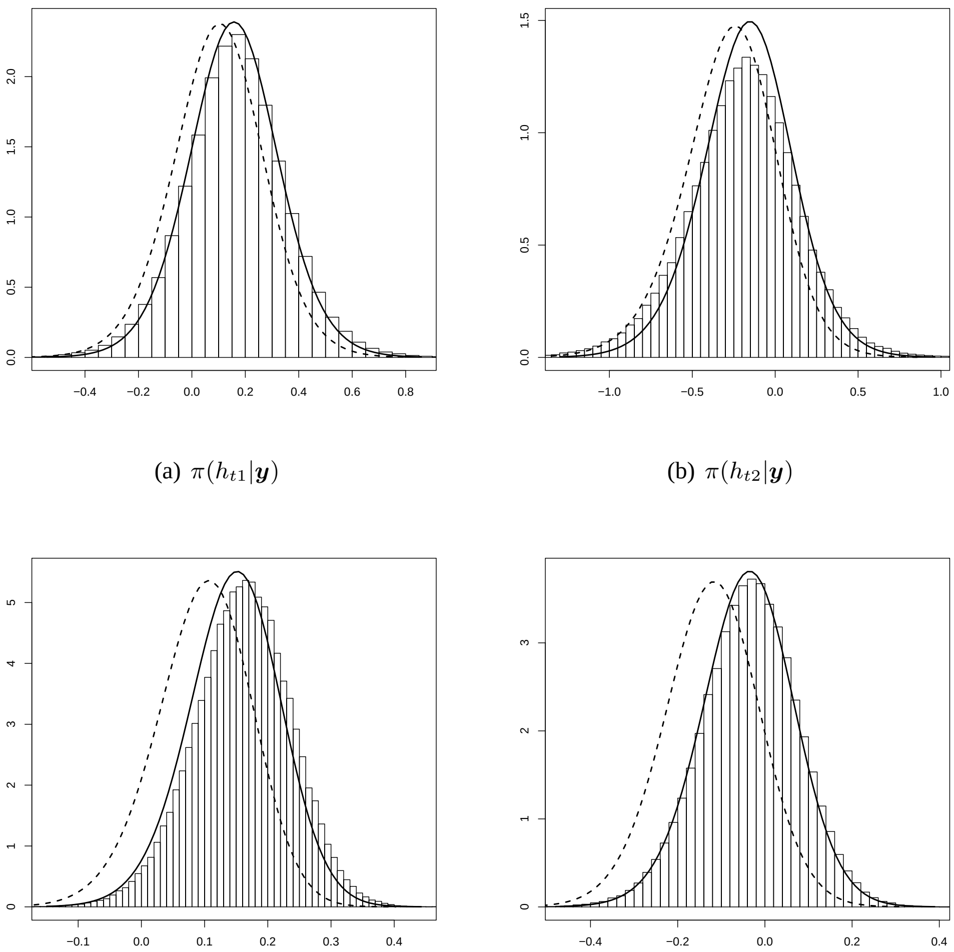
(a) $\pi(h_{t1}|\boldsymbol{y})$

(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) $\pi(\mu_2|\boldsymbol{y})$

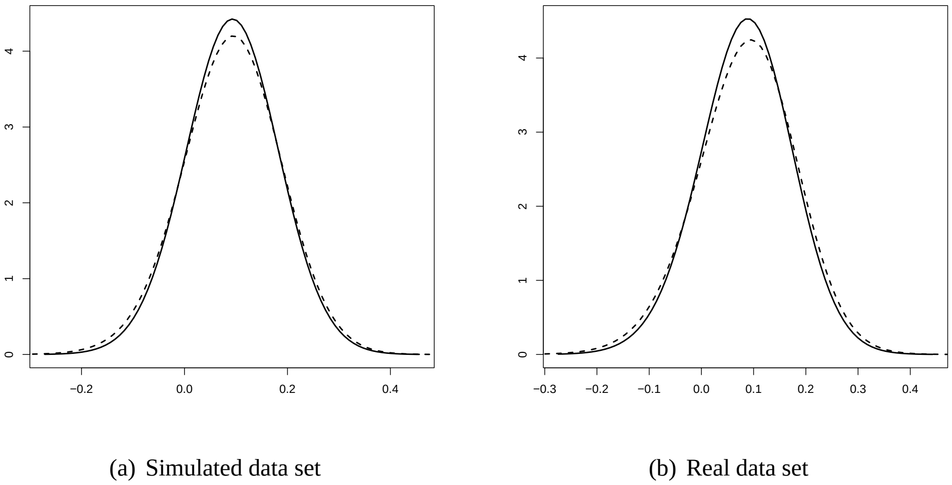
Figure 23: Australia/New Zeland data set, Model 2. Gaussian approximation (broken line) the improved approximation (solid line) and a MCMC based density estimate (histogram) for 4 nodes in the latent field.

6.3.3 Model 3 (MSV with Granger causality)

Model 3 adds one more hyperparameter by allowing the two latent time series to be interdependent. 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.



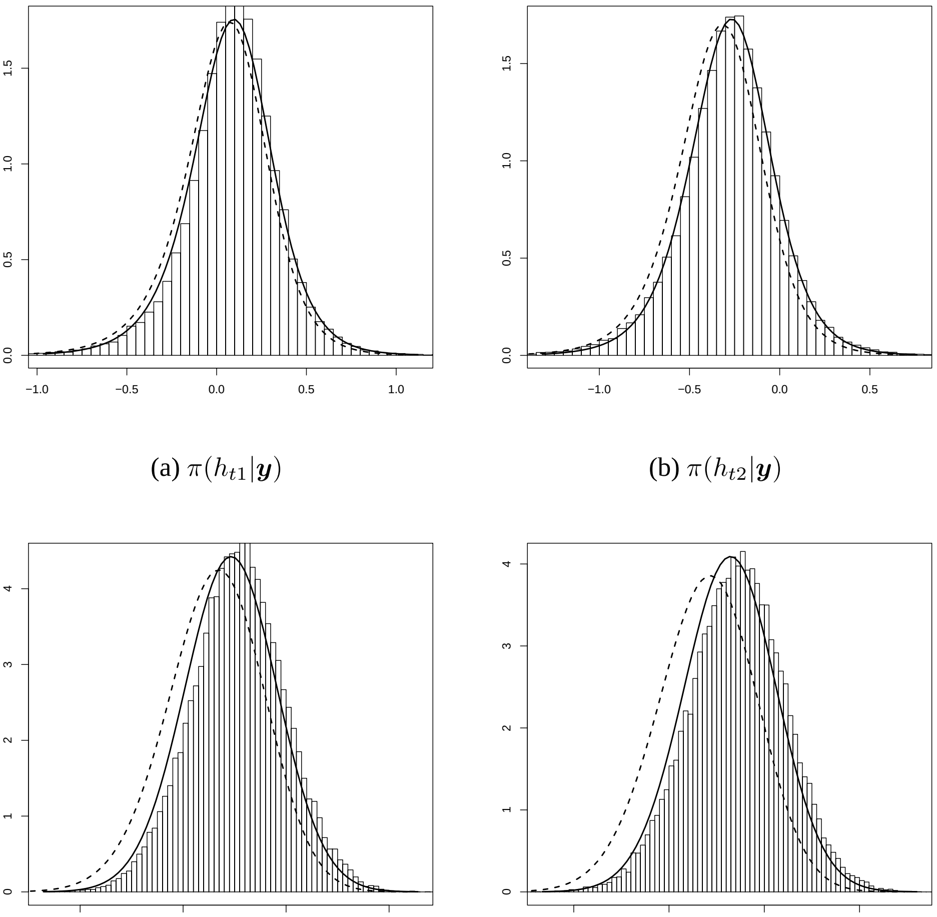
(a) Simulated data set

(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 the density, the small disagreement in skewness would make no difference in practice.



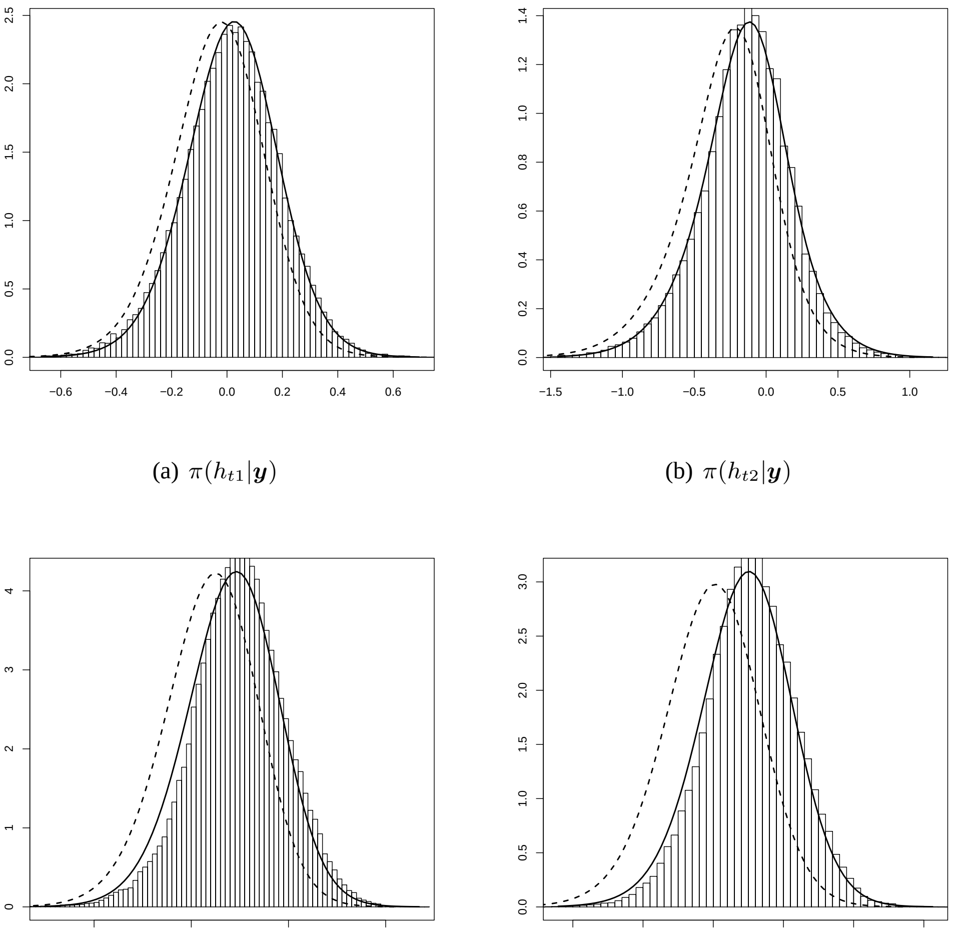
(a) $\pi(h_{t1}|\boldsymbol{y})$

(b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$

(d) $\pi(\mu_2|\boldsymbol{y})$

Figure 25: Simulated data set, Model 3. 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) $\pi(\mu_2|\boldsymbol{y})$

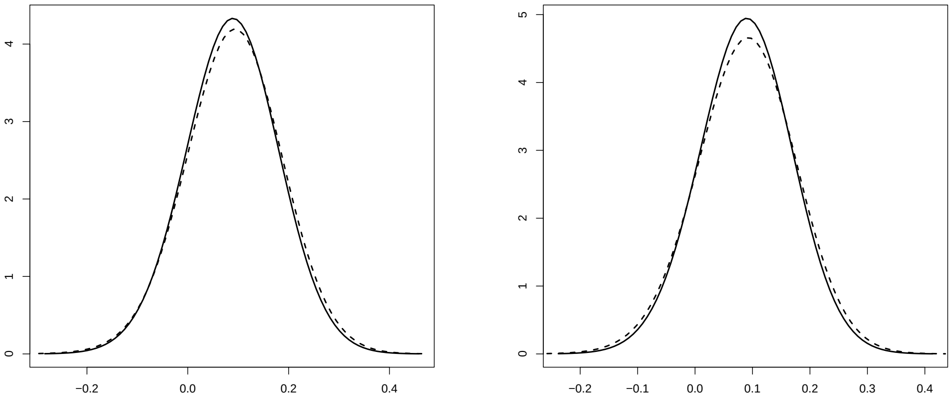
Figure 26: Australia/New Zeland data set, Model 3. Gaussian approximation (broken line) the improved approximation (solid line) and a MCMC based density estimate (histogram) for 4 nodes in the latent field.

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

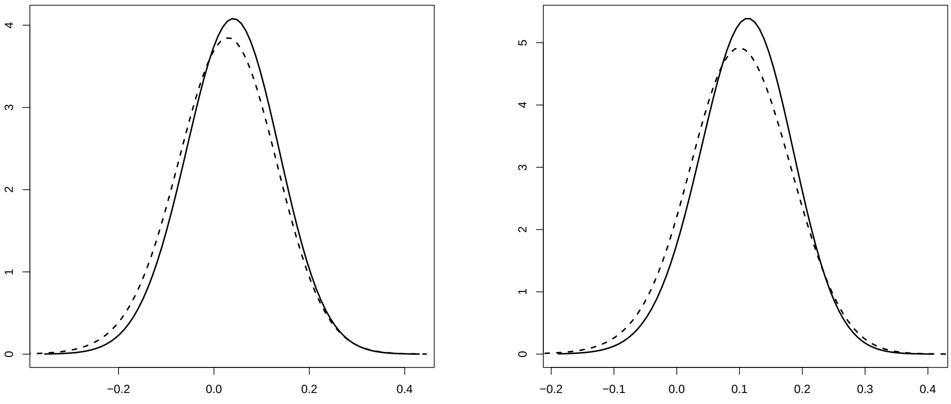
Figure 27: Model 4. Grid (solid line) and CCD (broken line) integration results.

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

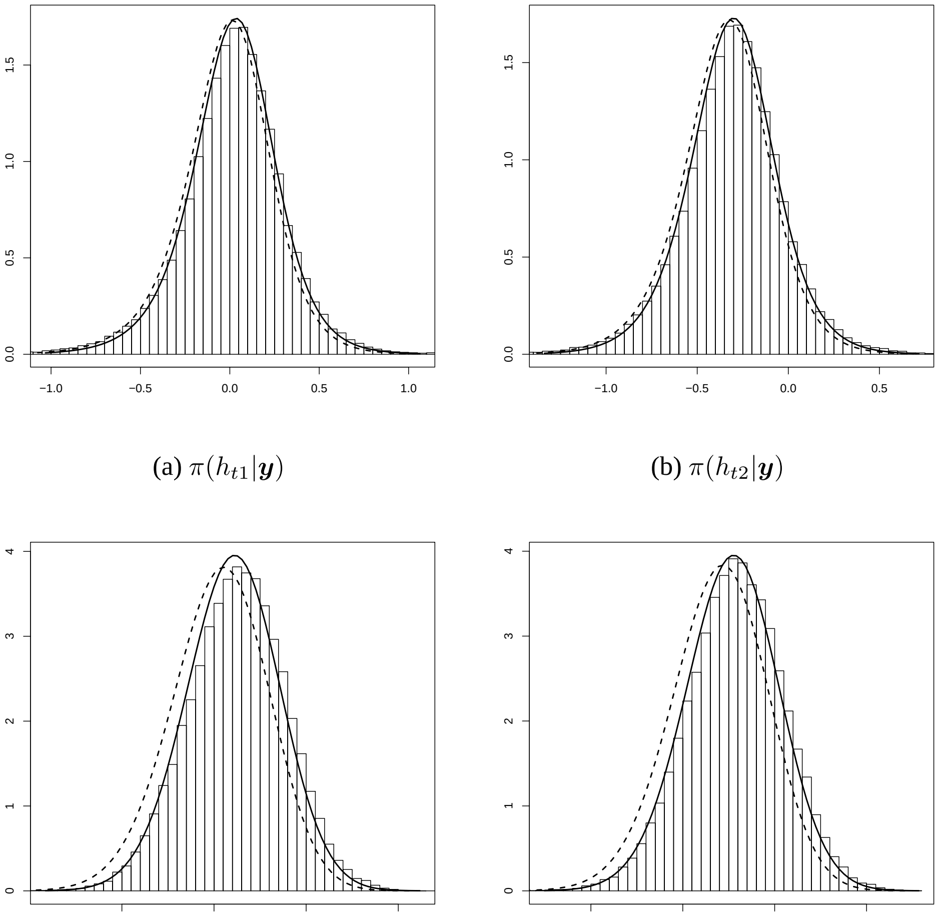
Figure 28: Model 5. Grid (solid line) and CCD (broken line) integration results.

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 running the MCMC algorithm long enough the approximation and the MCMC estimate would coincide also for the full data set.



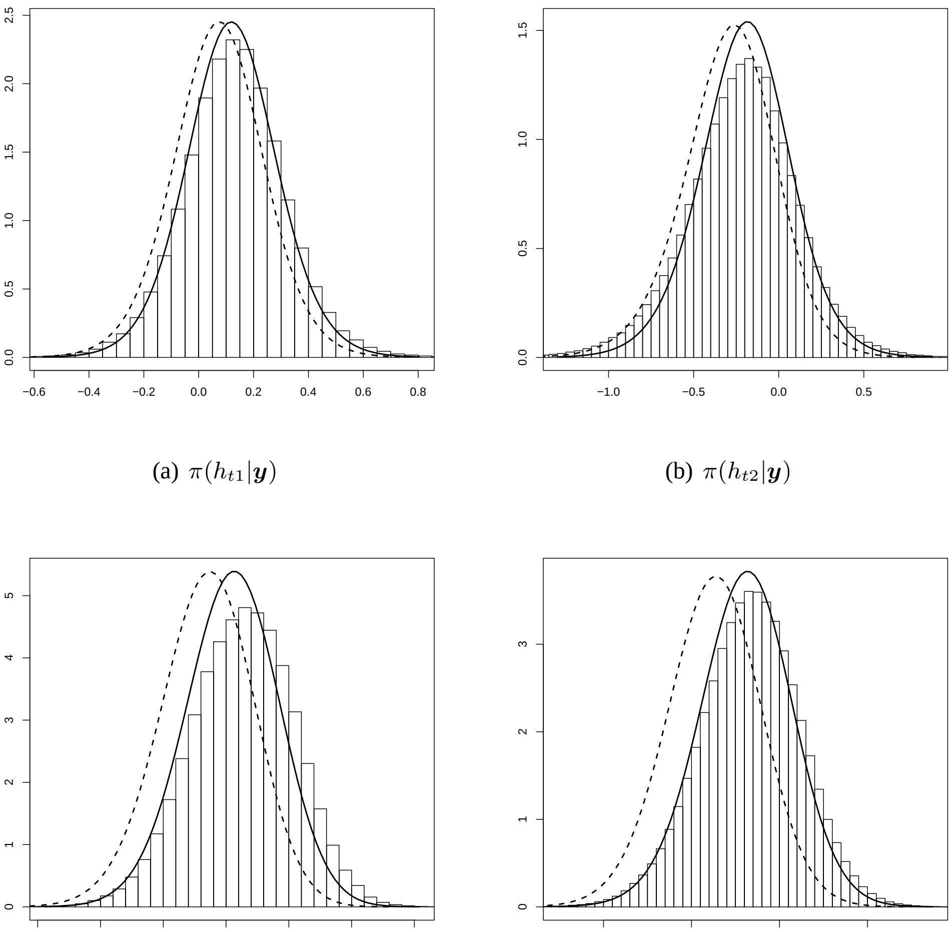
(a) $\pi(h_{t1}|\boldsymbol{y})$

(b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$

(d) $\pi(\mu_2|\boldsymbol{y})$

Figure 29: Simulated data set, Model 5. Gaussian approximation (broken line), improved approximation (solid line) and MCMC based density estimate (histogram).



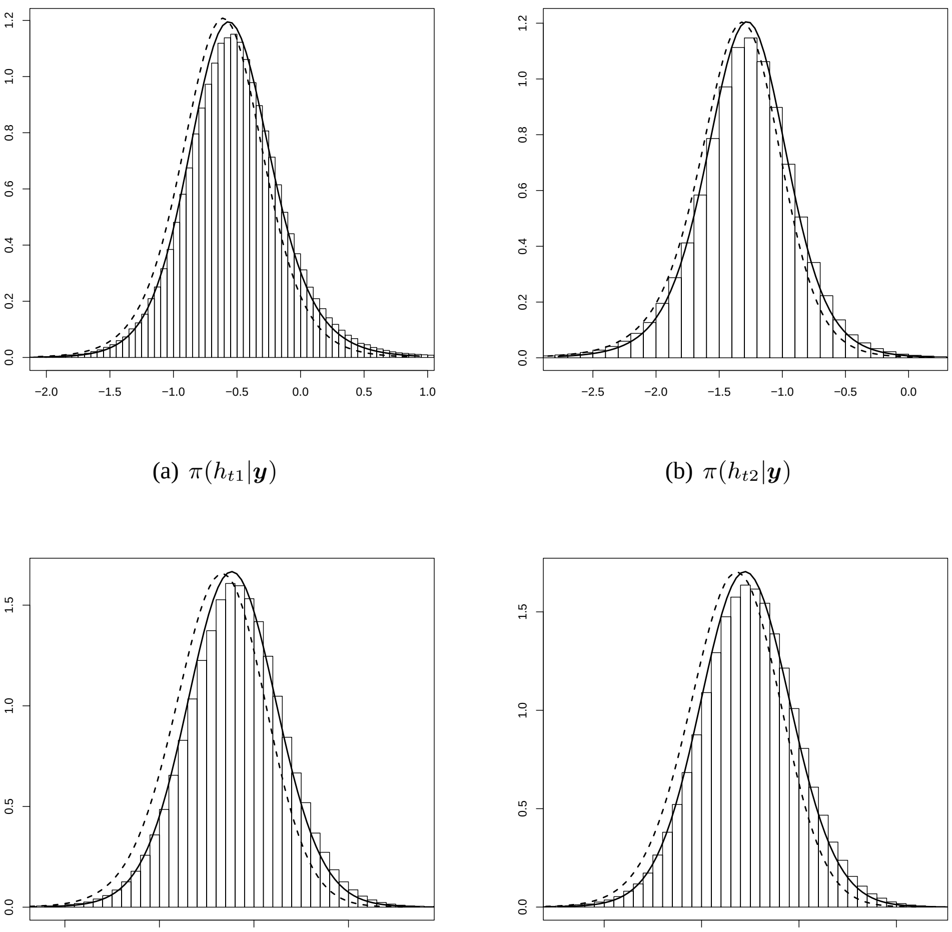
(a) $\pi(h_{t1}|\boldsymbol{y})$

(b) $\pi(h_{t2}|\boldsymbol{y})$

(c) $\pi(\mu_1|\boldsymbol{y})$

(d) $\pi(\mu_2|\boldsymbol{y})$

Figure 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	$\log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k) -$	$\log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k) -$
				$\max_k \log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k)$	$\max_k \log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k)$
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.

	$\log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k)$	$\log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k)$	Rank	$\log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k) -$	$\log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k) -$
				$\max_k \log \widetilde{\pi}_1(\boldsymbol{y}\|\mathcal{M}_k)$	$\max_k \log \widetilde{\pi}_2(\boldsymbol{y}\|\mathcal{M}_k)$
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.

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 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 $m = 1, \dots, M$. 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

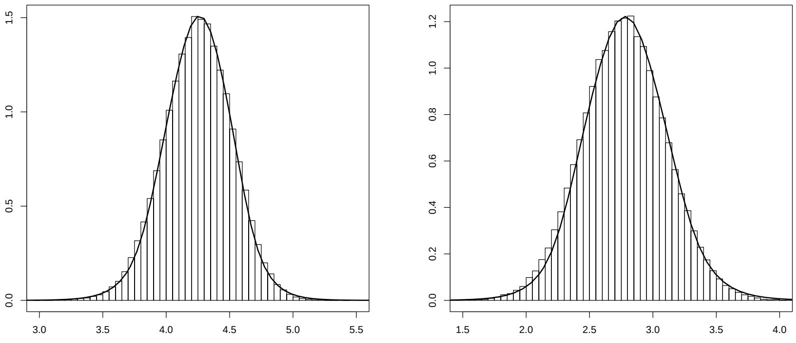
For not too high dimension of $\boldsymbol{\theta}$, it is possible to evaluate $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ on a regular grid and then use the resulting values to numerically compute the integral in (29). In order to locate the area of highest probability density we can use 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 $\{\theta_m^1, \dots, \theta_m^l\}$ with relative density $\{\widetilde{\pi}(\theta_m^1), \dots, \widetilde{\pi}(\theta_m^l)\}$. 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.

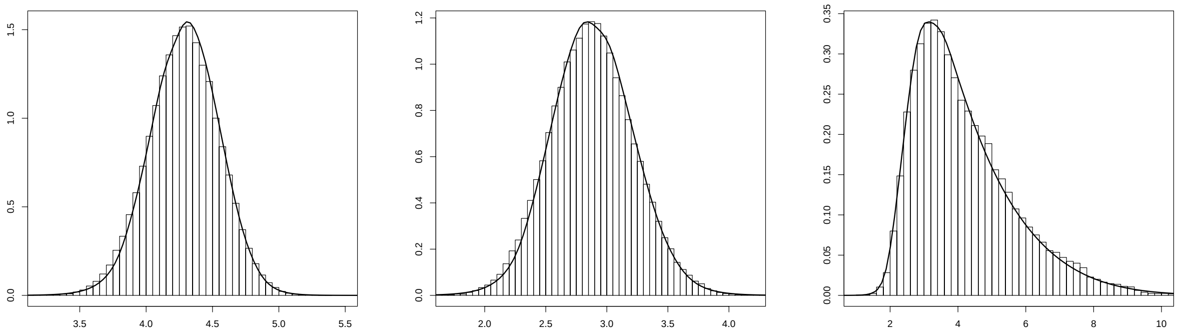
Unfortunately, in order to achieve precise approximations of the $\pi(\theta_m|\boldsymbol{y})$, especially in the tails, the grid has to be wider than the one used to compute $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$ and in some cases also finer. This means that we have to set the tuning parameter δ_π 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 $\pi(\theta_m|\boldsymbol{y})$, $m = 1, \dots, M$ 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.



(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.

Figure 32 displays the approximations for $\pi(\theta_m|\boldsymbol{y})$, $m = 1, \dots, M$ in model $\mathcal{M}_1$ compared 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.

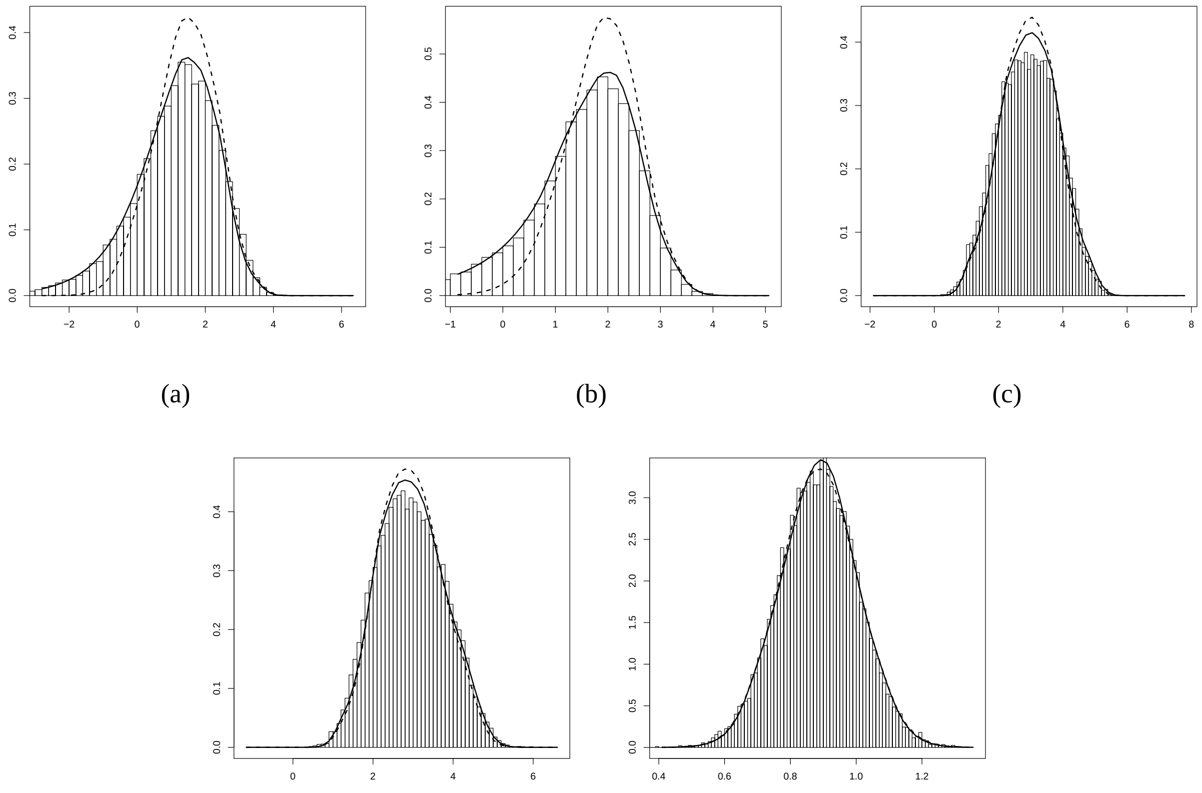
Unfortunately, computing the expression in (30) implies finding the maximum of the $(M-1)$ dimensional function $\widetilde{\pi}(\boldsymbol{\theta}_{-m}|\theta_m, \boldsymbol{y})$ for each value of θ_m. This operation, with increasing dimension of the hyperparameters space and of the latent field $\boldsymbol{x}$, 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 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.



(a) (b) (c)

(d) (e)

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.

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).

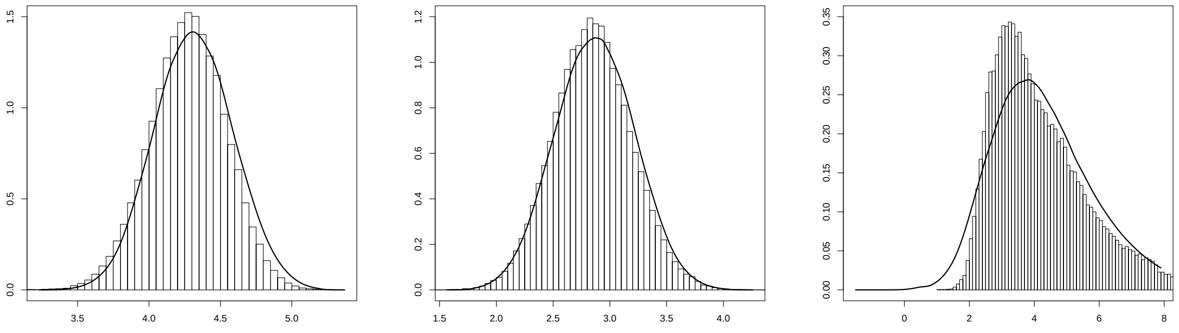
The main advantage of this approach is that, unlike the grid strategy presented in Section 7.1, it requires no extra computations of $\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ with respect to the computation of $\widetilde{\pi}(x_{ti}|\boldsymbol{y})$. In fact, the same evaluation points $\boldsymbol{\theta}_k$ 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.



(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.

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

$$f(\boldsymbol{\theta}) = \prod_{m=1}^{M} f_m(z_m(\boldsymbol{\theta})) \tag{31}$$

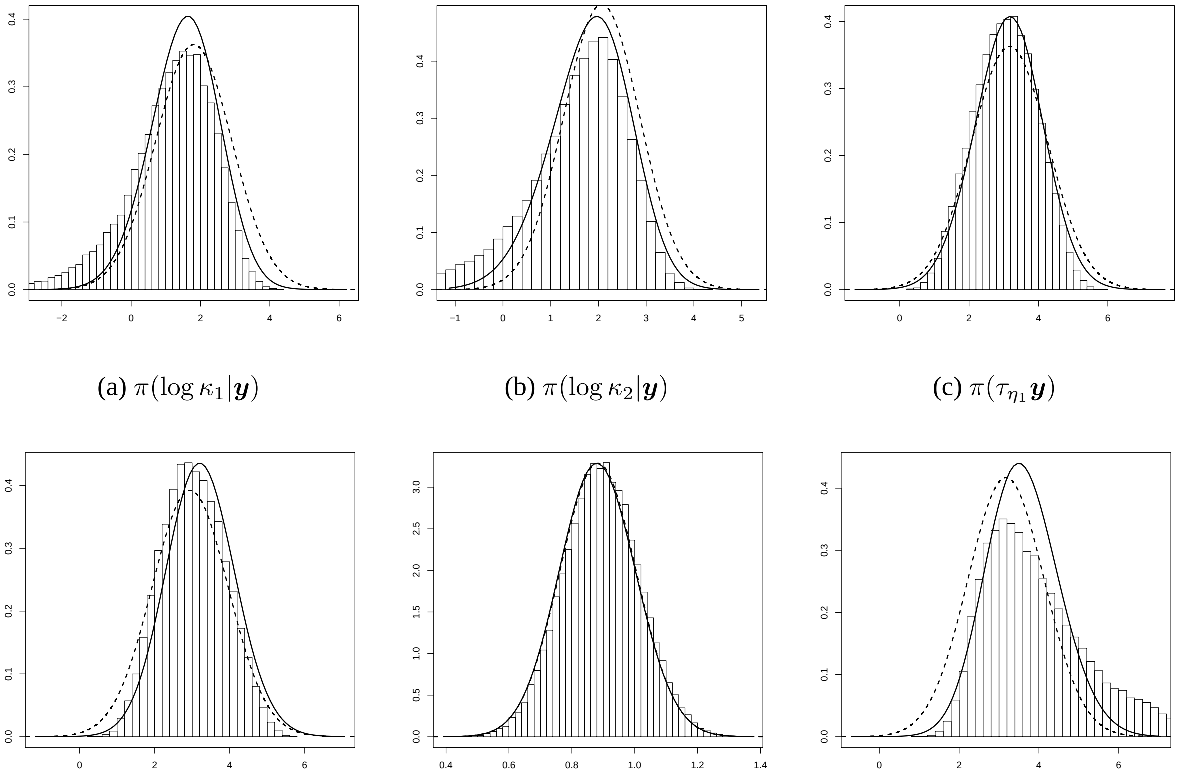
where

$$f_m(z) = \begin{cases} \exp\left(-\frac{1}{2(\sigma_{ccd}^{m+})^2} z^2\right) & \text{if} \quad z \geq 0 \\ \exp\left(-\frac{1}{2(\sigma_{ccd}^{m-})^2} z^2\right) & \text{if} \quad z < 0 \end{cases} \tag{32}$$

and σ_{ccd}^{m+} and σ_{ccd}^{m-}, $m = 1, \ldots, M$, are defined at page 16. 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 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 51. 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-}, $m = 1, \ldots, M$.



(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.

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 univariate SV model with leverage effect was first introduces by Harvey and Shephard (1996) and takes the form:

$$\begin{aligned} y_t &= \exp(h_t/2)\epsilon_t, \\ x_{t+1} &= \mu + \phi(h_t - \mu) + \sigma\eta_{t+1} \end{aligned} \tag{33}$$

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, $\text{Corr}(\epsilon_t, \eta_{t+1}) = \rho$, 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

$$\pi(y_t|\boldsymbol{x}, \boldsymbol{\theta}) = \pi(y_t|x_t, x_{t+1}, \boldsymbol{\theta}) = \mathcal{N}\left\{\frac{\rho}{\sigma}e^{x_t/2}[x_{t+1} - \mu + \phi(x_t - \mu)], e^{x_t}(1 - \rho^2)\right\} \tag{34}$$

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

$$\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) \propto \exp\left\{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \sum_{t=1}^{n_d} f_t(x_t, x_{t+1})\right\} \tag{35}$$

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 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

$$\log \widetilde{\pi}_{GG}(\boldsymbol{x}_{-t}|x_t,\boldsymbol{\theta}_k)\bigg|_{\boldsymbol{x}_{-t}=\mathrm{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-t}|x_t,\boldsymbol{\theta}_k)} \propto \frac{1}{2}\log|\boldsymbol{Q}^* + \mathrm{diag}\{\boldsymbol{c}(x_t,\boldsymbol{\theta}_k)\}| \tag{36}$$

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:

$$c_j(x_t,\boldsymbol{\theta}_k) = -\frac{\partial^2 \pi(y_j|x_j,\boldsymbol{\theta}_k)}{\partial x_j^2}\bigg|_{x_j=\mathrm{E}_{\widetilde{\pi}_G}(x_j|x_t,\boldsymbol{\theta}_k)}$$

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

$$d_j^3(x_t,\boldsymbol{\theta}_k) = \frac{\partial c_j(\boldsymbol{\theta}_k, x_t)}{\partial x_t} = -\frac{\partial^3 \pi(y_j|x_j,\boldsymbol{\theta}_k)}{\partial x_j^3}\bigg|_{x_j=\mathrm{E}_{\widetilde{\pi}_G}(x_j|x_t,\boldsymbol{\theta}_k)} \qquad \delta_j^t$$

Since for any matrix $\boldsymbol{M}$ we have that $\partial \log|\boldsymbol{M}| = \text{Trace}(\boldsymbol{M}^{-1}\partial\boldsymbol{M})$, then

$$\begin{aligned}\frac{d\log|\boldsymbol{Q}^{*}+\text{diag}(\boldsymbol{c})|}{dx_t} &= \text{Trace}\left\{[\boldsymbol{Q}^{*}+\text{diag}(\boldsymbol{c})]^{-1}\frac{d[\boldsymbol{Q}^{*}+\text{diag}(\boldsymbol{c})]}{dx_t}\right\} \\ &= \text{Trace}\left\{[\boldsymbol{Q}^{*}+\text{diag}(\boldsymbol{c})]^{-1}\text{diag}[\boldsymbol{d}^{3}(x_t,\boldsymbol{\theta}_k)]\right\} \\ &= \textstyle\sum_j \text{Var}(x_j|x_t)d_j^3(x_t,\boldsymbol{\theta}_k) \\ &= \textstyle\sum_j \sigma_{G_j}(\boldsymbol{\theta}_k)[1-\text{Corr}^2_{\pi_G}(x_t,x_j|\boldsymbol{\theta}_k)]\, d_j^3(x_t,\boldsymbol{\theta}_k)\end{aligned} \tag{37}$$

We have then

$$\begin{gathered}\log\widetilde{\pi}_{GG}(\boldsymbol{x}_{-t}|x_t,\boldsymbol{\theta}_k)\Bigg|_{\boldsymbol{x}_{-t}=\text{E}_{\widetilde{\pi}_G}(\boldsymbol{x}_{-t}|x_t,\boldsymbol{\theta}_k)} \approx \\ \tfrac{1}{2}\,x_t\,\textstyle\sum_j \sigma_{G_j}(\boldsymbol{\theta}_k)[1-\text{Corr}^2_{\pi_G}(x_t,x_j|\boldsymbol{\theta}_k)]\, d_j^3(x_t,\boldsymbol{\theta}_k)\end{gathered} \tag{38}$$

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 $i = 1,\dots,n$, the precision matrix of $x_{-i}|x_i$ is $\boldsymbol{Q}_{[-i,-i]}$. Moreover we have that

$$\pi(\boldsymbol{x}) = \pi(x_i)\pi(\boldsymbol{x}_{-i}|x_i) \propto \text{Var}(x_i)^{-1/2}|\boldsymbol{Q}_{[-i,-i]}|^{1/2}\exp\{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}\} \tag{40}$$

Comparing (39) and (40) we have that

$$\frac{1}{2}\log|\boldsymbol{Q}_{[-i,-i]}| = \frac{1}{2}\log|\boldsymbol{Q}| + \frac{1}{2}\log\text{Var}(x_i)$$

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 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

$$\begin{aligned}
\text{E}(y_t|x_t, x_{t+1}, \boldsymbol{\theta}) &= [x_{t+1} - \mu - \phi(x_t - \mu)] \frac{\rho e^{-x_t/2}}{\sigma(1-\rho^2)} \left[e^{-x_t} + \frac{\rho^2}{1-\rho^2} e^{-x_t}\right]^{-1} \\
&= [x_{t+1} - \mu - \phi(x_t - \mu)] \frac{\rho}{\sigma} e^{x_t/2}
\end{aligned}$$