

Case Studies in Bayesian Computation using INLA

Sara Martino and Håvard Rue

Abstract Latent Gaussian models are a common construct in statistical applications where a latent Gaussian field, indirectly observed through data, is used to model, for instance, time and space dependence or the smooth effect of covariates. Many well known statistical models, such as smoothing-spline models, space time models, semi-parametric regression, spatial and spatio-temporal models, log-Gaussian Cox models, and geostatistical models are latent Gaussian models. Integrated Nested Laplace approximation (INLA) is a new approach to implement Bayesian inference for such models. It provides approximations to the posterior marginals of the latent variables which are both very accurate and extremely fast to compute. Moreover, INLA treats latent Gaussian models in a general way, thus allowing for a great deal of automation in the inferential procedure. The `inla` program, bundled in the R library `INLA`, is a prototype of such black-box for inference on latent Gaussian models which is both flexible and user-friendly. It is meant to, hopefully, make latent Gaussian models applicable, useful and appealing for a larger class of users.

1 Introduction

Latent Gaussian models are an apparently simple but very flexible construct in statistical applications which covers a wide range of common statistical models spanning from (generalized) linear models, (generalized) additive models, smoothing spline models, state space models, semi-parametric regression, spatial and spatiotemporal models, log-Gaussian Cox processes and geostatistical and geosadditive models.

Sara Martino

Department of Mathematical Sciences, Norwegian University for Science and Technology, N-7491 Trondheim, Norway, e-mail: martino@math.ntnu.no

Håvard Rue

Department of Mathematical Sciences, Norwegian University for Science and Technology, N-7491 Trondheim, Norway, e-mail: hrue@math.ntnu.no

In these models, the latent Gaussian field serves as a flexible and powerful tool to model non-linear effects of covariates, group specific heterogeneity, as well as space and time dependencies among data.

Bayesian inference on latent Gaussian models is not straight forward since, in general, the posterior distribution is not analytically available. Markov Chain Monte Carlo (MCMC) techniques are, today, the standard answer to this problem and several ad hoc algorithms have been developed in recent years. Although in theory always possible to implement, MCMC algorithms applied to latent Gaussian models come with a wide range of problems in terms of convergence and computational time. Moreover, the implementation itself might often be problematic, especially for end users who might not be experts in programming.

Integrated Nested Laplace approximation (INLA) is a new tool for Bayesian inference on latent Gaussian models when the focus is on posterior marginal distributions [20]. INLA substitutes MCMC simulations with accurate, deterministic approximations to posterior marginal distributions. The quality of such approximations is extremely high, such that even very long MCMC runs could not detect any error in them. A detailed description of the INLA method and a thorough comparison with MCMC results can be found in [20].

INLA presents two main advantages over MCMC techniques. The first and most outstanding is computational. Using INLA results are obtained in seconds and minutes even for models with a huge dimensional latent field, where a well build MCMC algorithm would take hours or even days. This is also due to the fact that INLA is naturally parallelized, thus making it possible to exploit the new trend of having multi-core processors. The second, and not less important advantage, is that INLA treats latent Gaussian models in a unified way, thus allowing greater automation of the inference process. The core of the computational machinery, automatically adapts to any kind of latent field so that, from the computational point of view, it does not matter if we deal with, for example, spatial or temporal models. In practice INLA can be used almost as a black box to analyze latent Gaussian models.

A prototype of such program, `inla`, together with a user-friendly R interface (INLA library) is already available from the web-site www.r-inla.org. Its goal is to make the INLA approach available for a larger class of users. The hope is that near instant inference and simplicity of use will make latent Gaussian models more applicable, useful and appealing for the end user.

The purpose of this paper is to give an overview of models to which INLA is applicable. We will present a series of case studies ranging from generalized linear models to spatially varying regression models to survival models, solved using the INLA methodology through the `INLA` library. The structure of this article is as follows. Section 2 describes latent Gaussian models and their main features. Section 3 and Section 4 briefly introduce the INLA approach and the `INLA` library. In Section 5 three case studies are analyzed. They include a GLMM model with over-dispersion, different models for spatial analysis and a model for survival data. We end with a brief discussion in Section 6.

2 Latent Gaussian Models

Latent Gaussian models are hierarchical models which assume a n -dimensional Gaussian field $\mathbf{x} = \{x_i : i \in \mathcal{V}\}$ to be point-wise observed through n_d conditional independent data \mathbf{y} . Both the covariance matrix of the Gaussian field \mathbf{x} and the likelihood model for $y_i|\mathbf{x}$ can be controlled by some unknown hyperparameters θ . In addition, some linear constraints of the form $\mathbf{A}\mathbf{x} = \mathbf{e}$, where the matrix \mathbf{A} has rank k , may be imposed. The posterior then reads:

$$\pi(\mathbf{x}, \theta | \mathbf{y}) \propto \pi(\theta) \pi(\mathbf{x} | \theta) \prod_{i \in \mathcal{J}} \pi(y_i | x_i, \theta). \quad (1)$$

As the likelihood often is not Gaussian, this posterior density is not analytically tractable.

A slightly different point of view to look at latent Gaussian models is to consider structured additive regression models; these are a flexible and extensively used class of models, see for example [8] for a detailed account. Here, the observation (or response) variable y_i is assumed to belong to an exponential family where the mean μ_i is linked to a structured additive predictor η_i through a link-function $g(\cdot)$, so that $g(\mu_i) = \eta_i$. The likelihood model can be controlled by some extra hyperparameters θ_1 . The structured additive predictor η_i accounts for effects of various covariates in an additive way:

$$\eta_i = \beta_0 + \sum_{j=1}^{n_f} w_{ji} f^{(j)}(u_{ji}) + \sum_{k=1}^{n_\beta} \beta_k z_{ki} + \varepsilon_i. \quad (2)$$

Here, the $\{\beta_k\}$'s represent the linear effect of covariates \mathbf{z} . The $\{f^{(j)}(\cdot)\}$'s are unknown functions of the covariates \mathbf{u} . These can take very many different forms: non-linear effects of continuous covariates, time trends, seasonal effects, i.i.d.random intercepts and slopes, group specific random effects and spatial random effects can all be represented through the $\{f^{(j)}\}$'s functions. The w_{ij} are known weights defined for each observed data point. Finally, ε_i 's are unstructured random effects.

A latent Gaussian model is obtained by assigning $\mathbf{x} = \{\{f^{(j)}(\cdot)\}, \{\beta_k\}, \{\eta_i\}\}$ a Gaussian prior with precision matrix $\mathbf{Q}(\theta_2)$, with hyperparameters θ_2 . Note that we have parametrized the latent Gaussian field so that it includes the η_i 's instead of the ε_i 's, in this way some of the elements of \mathbf{x} , namely the η_i 's, are observed through the data \mathbf{y} . This is mainly due to the fact that the INLA library requires each data point y_i to be dependent on the latent Gaussian field only through one single element of \mathbf{x} , namely η_i . For this reason, a small random noise, ε_i , with high precision is always automatically added to the model. The definition of the latent model is completed by assigning the hyperparameters $\theta = (\theta_1, \theta_2)$ a prior distribution.

In this paper the latent Gaussian models are assumed to satisfy two basic properties: First, the latent Gaussian model \mathbf{x} , often of large dimension, admits conditional independence properties. In other words it is a latent Gaussian Markov random field (GMRF) with a sparse precision matrix \mathbf{Q} , [18]. The second property is that the di-

mension m of the hyperparameter vector θ is small, say ≤ 6 . These properties are satisfied by many latent Gaussian models in the literature. Exceptions exist, geostatistical models being the main one. INLA can still be applied to geostatistical model using different computational machinery or using a Markov representation of the Gaussian field (see [7] and the discussion contribution from Finn Lindgren in [20])

3 Integrated Nested Laplace Approximation

Integrated Nested Laplace approximation (INLA) is a new approach to statistical inference for latent Gaussian models introduced by [19] and [20]. In short, the INLA approach provides a recipe for fast Bayesian inference using accurate approximations to the marginal posterior density for the hyperparameters $\tilde{\pi}(\theta|\mathbf{y})$ and for the full conditional posterior marginal densities for the latent variables $\tilde{\pi}(x_i|\theta, \mathbf{y})$, $i = 1, \dots, n$. The approximation for $\pi(\theta|\mathbf{y})$ is based on the Laplace approximation [22], while for $\pi(x_i|\theta, \mathbf{y})$ three different approaches are possible: A Gaussian, a full Laplace and a simplified Laplace approximation. Each of these has different features, computing times and accuracy. The Gaussian approximation is the fastest to compute but there can be errors in the location of the posterior mean and/or errors due to the lack of skewness [19]. The Laplace approximation is the most accurate but its computation can be time consuming. Hence in [20] the simplified Laplace approximation is introduced. This is fast to compute and usually accurate enough.

Posterior marginals for the latent variables $\tilde{\pi}(x_i|\mathbf{y})$ are then computed via numerical integration as:

$$\begin{aligned}\tilde{\pi}(x_i|\mathbf{y}) &= \int \tilde{\pi}(x_i|\theta, \mathbf{y}) \tilde{\pi}(\theta|\mathbf{y}) d\theta \\ &\approx \sum_{k=1}^K \tilde{\pi}(x_i|\theta_k, \mathbf{y}) \tilde{\pi}(\theta_k|\mathbf{y}) \Delta_k\end{aligned}\tag{3}$$

Posterior marginals for the hyperparameters $\tilde{\pi}(\theta_j|\mathbf{y})$, $j = 1, \dots, m$ are computed in a similar way. The choice of the integration points θ_k can be done using two strategies: The first, more accurate but also time consuming, is to define a grid of points covering the area where most of the mass of $\tilde{\pi}(\theta|\mathbf{y})$ is located (GRID strategy). The second strategy, named central composite design (CCD strategy), comes from the design problem literature and consists in laying out a small amount of 'points' in a m -dimensional space in order to estimate the curvature of $\tilde{\pi}(\theta|\mathbf{y})$, see [20] for more details on both strategies. In [20] it is suggested to use the CCD strategy as a default choice. Such strategy is usually accurate enough for the computation of $\tilde{\pi}(x_i|\mathbf{y})$, while a GRID strategy might be necessary if one is interest in a accurate estimate of $\tilde{\pi}(\theta_j|\mathbf{y})$. The approximate posterior marginals obtained from such procedure can then be used to compute summary statistics of interest, such as posterior means, variances or quantiles.

INLA can also compute, as by-product of the main computations, other quantities of interest like Deviance Information Criteria (DIC) [21], marginal likelihoods and predictive measures as logarithmic score [11] and the PIT histogram [6], useful to detect outliers and to compare and validate models.

Different strategies to assess the accuracy of the various approximations for the densities $x_i|\theta, \mathbf{y}$ are described in [20]. The INLA approximations assumes the posterior distribution $\pi(\mathbf{x}|\theta, \mathbf{y})$ to be unimodal and fairly regular. This is usually the case for most real problem and data sets. INLA can, however, deal to some extent with the multimodality of $\pi(\theta|\mathbf{y})$, provided that the modes are sufficiently closed. See the discussion contributions of Ferreira and Hodges and the author's reply in [20].

Theory and practicalities about INLA are deeply analyzed in [20] and will not be repeated here. Loosely speaking we can say that INLA fully exploits all main features of the latent Gaussian models described in Section 2. Firstly, all computations are based on sparse matrix algorithms which are much faster than the corresponding algorithms for dense matrix. Secondly, the presence of the latent Gaussian field and the usual “well-behavior” of the likelihood function justify the accuracy of the Laplace approximation. Finally, the small number of hyperparameters θ makes the numerical integration in equation (3) computationally feasible.

4 The INLA package for R

Computational speed is one of the most important components of the INLA approach, therefore special care has to be put in the implementation of the required algorithms. All computations required by the INLA methodology are efficiently performed by the `inla` program, written in C based on the `GMRFLib`-library which includes efficient algorithms for sparse matrices [18]. Both the `inla` program and the `GMRFLib`-library, in addition, use the OpenMP (see <http://www.openmp.org>) to speed up the computations for shared memory machines, i.e. multicore processors, which are today standard for new computers.

Moreover, the `inla` program is bundled within a R library called `INLA` in order to ease its usage. The software is open-source and can be downloaded from the web site www.r-inla.org. It is running under Linux, MAC and Windows. On the same web site documentation and a large sample of applications are also provided.

5 Case studies

The following examples are meant to give an overview of the range of application of the INLA methodology. All examples are implemented using the `INLA` library on a dual-core 2.5GHz laptop. The R code is reported where it was considered helpful, the rest of the R code can be downloaded from the www.r-inla.org web site in the Download section.

5.1 A GLMM with over-dispersion

The first example is a generalized linear mixed model with binomial likelihood, where random effects are used to model within group extra variation. The data concern the proportion of seeds that germinated on each of $m = 21$ plates arranged in a 2×2 factorial design with respect to seed variety and type of root extract. The data set was presented by [5] and analyzed among other by [3], this example is also included in the WinBUGS/OpenBUGS manual [15]. In [9] the authors perform a comparison between the INLA and the maximum likelihood approach for this particular data set.

The sampling model is $y_i | \eta_i \sim \text{Binomial}(n_i, p_i)$ where, for plate $i = 1, \dots, m$, y_i is the number of germinating seeds (variable name (vn): r) and n_i the total number of seeds ranging from 4 to 81 (vn: n), and $p_i = \text{logit}^{-1}(\eta_i)$ is the unknown probability of germinating. To account for between plate variability, [3] introduce plate-specific random effects, and then fit a model with main and interaction effects:

$$\eta_i = \beta_0 + \beta_1 z_{1i} + \beta_2 z_{2i} + \beta_3 z_{1i} z_{2i} + f(u_i) \quad (4)$$

with z_{1i} and z_{2i} representing the seed variety (vn: $x1$) and type of root extract (vn: $x2$) of plate i . We assign β_k , $k = 0, \dots, 3$ vague Gaussian priors with known precision. Moreover, we assume $f(u_i) | \tau_u \sim \mathcal{N}(0, \tau_u^{-1})$, $i = 1, \dots, 21$, so that the general $f()$ function in Equation (2) here takes the simple form of i.i.d. random intercepts. To complete the model we assign the hyperparameter a vague Gamma prior $\tau_i \sim \text{Gamma}(a, b)$ with $a = 1$ and $b = 0.001$.

As explained in Section 2, when specifying the model in the INLA library a tiny random noise ε_i with zero mean and known high precision is always added to the linear predictor, so the latent Gaussian field for the current example is $\mathbf{x} = \{\eta_1, \dots, \eta_m, \beta_0, \dots, \beta_3, u_1, \dots, u_m\}$, while the vector of hyperparameters has dimension one, $\theta = \{\tau_u\}$.

To run the model using INLA two steps have to be taken. Firstly, the linear predictor of the model has to be specified as a `formula` object in R. Here the function `f()` is used to specify any possible form of the general $f()$ function in (2), in the current model the i.i.d. random effect is specified using `model="iid"`.

```
> formula = r ~ x1*x2 + f(plate, model="iid")
```

Secondly, the specified model can be run calling the `inla()` function.

```
> mod.seeds = inla(formula, data=Seeds, family="binomial",
+                 Ntrials=n)
```

The $a = 1$ and $b = 0.001$ parameters for the Gamma prior for τ_u are the default choice therefore there is no need to specify them. A different choice of parameters a and b can be specified as `f(plate, model="iid", param=c(a,b))`.

A `summary()` function is available to inspect results.

```
> summary(mod.seeds)
```

```

Fixed effects:
              mean      sd 0.025quant 0.975quant      kld
(Intercept) -0.554623 0.140843 -0.833317 -0.277369 5.27899e-05
x1           0.129510 0.243002 -0.326882  0.600159 3.97357e-07
x2           1.326120 0.199081  0.938824  1.725244 3.98097e-04
x1:x2        -0.789203 0.334384 -1.452430 -0.135351 3.35429e-05

Random effects:
Name      Model      Max KLD
plate     IID model   4e-05

Model hyperparameters:
              mean      sd      0.025quant 0.975quant
Precision for plate 1620.89 2175.49   104.63    7682.16

Expected number of effective parameters(std dev): 5.678(2.216)
Number of equivalent replicates : 3.698

```

Standard summary() output includes posterior mean, standard deviation, 2.5% and 97.5% quantiles both for the elements in the latent field and for the hyperparameters. Moreover, the expected number of effective parameters, as defined in [21], and the number of data points per expected number of effective parameter (Number of equivalent replicates) is also provided. These measure might be useful to assess the accuracy of the approximation, see [20] for more details. Briefly, a low number of equivalent replicates might flag a “difficult” case for the INLA approach.

The INLA library includes also a set of functions which post-process the marginal densities obtained by inla(). These functions allow to compute quantiles, percentiles, expectations of function of the original parameter, density of a particular value and also allow to sample from the marginal. As an example consider the following: The output of the inla() function provides us with posterior mean and standard deviation of the precision parameter τ_u . Assume that we are instead interested in the posterior mean and standard deviation of the variance parameter $\sigma_u^2 = 1/\tau_u$. This can be easily done by selecting the appropriate posterior marginal from the output of the inla() function:

```

prec.marg = mod.seeds$marginals.hyperpar$"Precision for plate"

and then using the function inla.expectation()

> m1 = inla.expectation(function(x) 1/x, prec.marg)
> m2 = inla.expectation(function(x) (1/x)^2, prec.marg)
> sd = sqrt(m2 - m1^2)
> print(c(mean=m1, sd=sd))
              mean      sd
0.001875261 0.002823392

```

Sampling from posterior densities can be also done using inla.rmarginal(). For example, a sample of size 1000 from the posterior $\tilde{\pi}(\beta_1|y)$ is obtained as follows:

```
> dens = mod.seeds$marginals.fixed$x1
> sample = inla.rmarginal(1000,dens)
```

More information about functions operating on marginals can be found by typing `?inla.marginal`.

5.2 Childhood undernutrition in Zambia: spatial analysis

In the second example we consider three different spatial models to analyze the Zambia data set presented in [14]. Here the authors study childhood undernutrition in 57 regions of Zambia. A total of $n_d = 4847$ observation are included in the data set. Undernutrition is measured by stunting, or inefficiency height for age, indicating chronic undernutrition. Stunting for child $i = 1, \dots, n_d$ is determined using a Z score defined as

$$Z_i = \frac{AI_i - MAI}{\sigma}$$

where AI refers to the child's anthropometric indicator, MAI refers to the median of the reference population and σ refers to the deviation of the standard population. In addition, the data set includes a set of covariates such as the age of the child (age_i), the body mass index of the child's mother (bmi_i), the district the child lives in (s_i) and four additional categorical covariates. For more details about the data set see [13] and [14].

We assume the scores Z_i (vn: `hazstd`) to be conditionally independent Gaussian random variables with unknown mean η_i and unknown precision τ_z . We consider three different models for the mean parameter η_i . The first is defined as:

$$\eta_i = \mu + \mathbf{z}_i^T \boldsymbol{\beta} + f_s(s_i) + f_u(s_i) \quad (5)$$

This model will be called MOD1. It assumes all six covariate to have a linear effect. Moreover, it contains a spatially unstructured component $f_u(s_i)$ (vn: `distr.unstruct`), which is i.i.d normally distributed with zero mean and unknown precision τ_u , and a spatially structured component $f_s(s_i)$ (vn: `district`) which is assumed to vary smoothly from region to region. To account for such smoothness $f_u(s_i)$ is modeled as an intrinsic Gaussian Markov random field with unknown precision τ_s , see [18]. This specification is also called a conditionally autoregressive (CAR) prior [1] and was introduced by [2]. To ensure identifiability of the mean μ , a sum-to-zero constrain must be imposed on the $f_s(s_i)$'s. The latent Gaussian field for this model is $\mathbf{x} = \{\mu, \{\beta_k\}, \{f_s(\cdot)\}, \{f_u(\cdot)\}, \{\eta_i\}\}$, while the hyperparameters vector is $\boldsymbol{\theta} = \{\tau_z, \tau_u, \tau_s\}$. Vague independent Gamma priors are assigned to each element in $\boldsymbol{\theta}$. When specifying the model in (5) using the INLA library, the type of smooth effect is specified using `model="iid"` for the unstructured spatial component and `model="besag"` for the structured one. Moreover, for the spatially structured term, a graph file (e.g. `"zambia.graph"`) containing

the neighborhood structure has to be specified. The structure of such graph file is described in [16]. The resulting model specification looks like:

```
> formula = hazstd ~ edu1 + edu2 + tpr + sex + bmi + agc +
+       f(district, model="besag", graph.file="zambia.graph") +
+       f(distr.unstruct, model="iid")
```

Note that in the INLA library a sum-to-zero constraint is the default choice for every intrinsic model. One requirement of the INLA library is that, each effect specified through a `f()` function in the formula should correspond to a different column in the data file, that's why the two column `district` and `distr.unstruct` are needed. Fitting the model is done by calling the `inla()` function:

```
> mod = inla(formula, family="gaussian", data=Zambia,
+       control.compute=list(dic=TRUE, cpo=TRUE))
```

The `dic=TRUE` flag makes the `inla()` function compute the model's deviance information criterion (DIC). This is a measure of complexity and fit introduced in [21] and used to compare complex hierarchical models. It is defined as:

$$\text{DIC} = \bar{D} + p_D$$

where \bar{D} is the posterior mean of the deviance of the model and p_D is the effective number of parameters. Smaller values of the DIC indicate a better trade-off between complexity and fit of the model.

The `cpo=TRUE` flag tells the `inla()` function to compute also some predictive measures for the observed y_i given all other observations. In particular the predictive density $\pi(y_i|y_{-i})$ (called `cpo`) and the probability integral transform $\text{PIT}_i = \text{Prob}(y_i^{\text{new}} < y_i|y_{-i})$ (called `pit`) are computed. These quantities can be useful to assess the predictive power of the model or to detect surprising observations. See [20] for details on how such quantities are computed. As noted in [12] the simplified Laplace approximation might, in some cases, not be accurate enough when computing predictive measures. The `inla()` function outputs a vector (which can be recovered as `mod$failure`) which contains values from 0 to 1 for each observation. The value 0 indicates that the computation of `cpo` and `pit` for the corresponding observation was computed without problems. A value greater than 0 instead, indicates that there were some computing problems and the predictive measures should be recomputed. See the FAQ section on www.r-inla.org if interested in the topic.

The posterior mean for the β parameters, together with standard deviations and quantiles are presented in Table 1. The `inla()` function returns the whole posterior density for such parameters, therefore, if needed other quantities of interest can also be computed. The posterior mean of smooth and unstructured spatial effects are displayed in Figure 1. The output of the `inla()` function also includes posterior marginals for the hyperparameters of the model and posterior marginals for the linear predictor, which are not displayed here. The value of the DIC for MOD1 is displayed in Table 2. To assess the predictive quality of the model the cross-validated

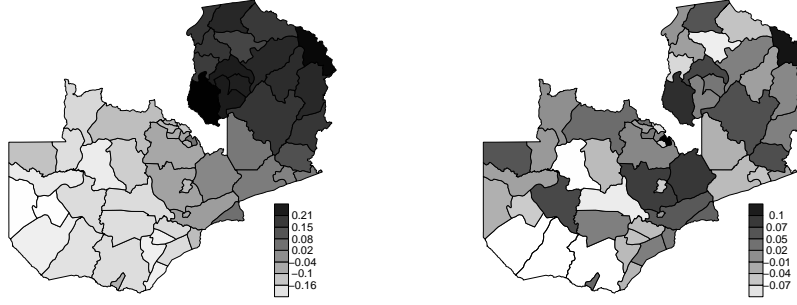


Fig. 1 Posterior mean for the smooth spatial effect (left) and posterior mean for the unstructured spatial effect (right) in MOD1

logarithmic score [11] can be used. It can be computed using the `inla()` output as:

```
> log.score = -mean(log(mod1$cpo))
```

And the resulting value is displayed in Table 2. A smaller value of the logarithmic score indicates a better prediction quality of the model. The `mod$failure` vector, in this case, contains only 0's so predictive quantities can be used without problems. A tool to assess the calibration of the model is to check the `pit` histogram. As suggested in [6], in fact, in a well calibrated model the `pit` values should have a uniform distribution. For MOD1 the `pit` histogram (not shown here but available on www.r-inla.org) doesn't show any sign on wrong calibration.

As discussed in Section 3 the default integration strategy in the `inla()` function is the CCD strategy. It is possible to choose a GRID strategy instead using the following call to `inla()`:

```
> mod = inla(formula, family="gaussian", data=Zambia,
+           control.inla = list(int.strategy = "grid"))
```

The computational time increases from the ca 9 seconds needed by the CCD integration to the ca 16 seconds while a comparison of the results coming from the two fit (not shown here) does not present any significant difference.

As discussed in [14] there are strong reasons to believe that the effect of the age of the child is smooth but not linear. To check such assumption we can modify the previous model as:

$$\eta_i = \mu + \mathbf{z}_i^T \beta + f_1(\text{age}_i) + f_s(s_i) + f_u(s_i) \quad (6)$$

This extended model will be called MOD2. Here $\{f_1(\cdot)\}$ follows an intrinsic second-order random-walk model with unknown precision τ_1 , see [18]. To ensure

Table 1 Posterior mean (standard deviation) together with 2.5% and 97.5% quantiles for the linear effect parameters in the three models for the Zambia data.

Model	Covariate	Mean(sd)	2.5% quant	97.5% quant
MOD1	μ	-0.010(0.100)	-0.207	0.187
	β_{agc}	-0.015(0.001)	-0.017	-0.013
	β_{edu1}	-0.061(0.027)	-0.114	-0.009
	β_{edu2}	0.227(0.047)	0.134	0.320
	β_{tpr}	0.113(0.021)	0.072	0.155
	β_{sex}	-0.059(0.013)	-0.086	-0.033
	β_{bmi}	0.023(0.004)	0.014	0.031
MOD2	μ	-0.412(0.096)	-0.602	-0.223
	β_{edu1}	-0.060(0.026)	-0.111	-0.009
	β_{edu2}	0.234(0.046)	0.145	0.324
	β_{tpr}	0.105(0.021)	0.064	0.145
	β_{sex}	-0.058(0.013)	-0.084	-0.033
	β_{bmi}	0.021(0.004)	0.013	0.029
MOD3	μ	-0.366(0.096)	-0.556	-0.178
	β_{edu1}	-0.061(0.026)	-0.112	-0.010
	β_{edu2}	0.232(0.046)	0.142	0.321
	β_{tpr}	0.107(0.023)	0.062	0.152
	β_{sex}	-0.059(0.013)	-0.084	-0.033

identifiability of μ , a sum-to-zero constraint must be imposed on $f_1(\cdot)$. The latent field is then $\mathbf{x} = \{\mu, \{\beta_k\}, \{f_s(\cdot)\}, \{f_u(\cdot)\}, \{f_1(\cdot)\}, \{\eta_i\}\}$ while the hyperparameter vector is $\theta = \{\tau_z, \tau_u, \tau_s, \tau_1\}$. The INLA specification of MOD2 differs from the previous one simply because now a `f()` function is used also to define the smooth effect of age.

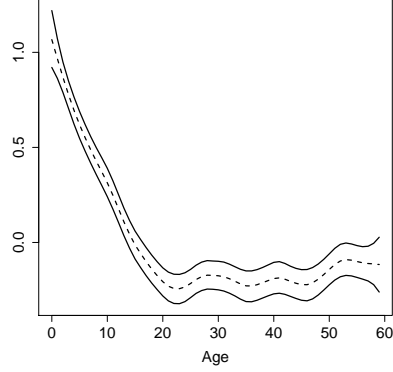
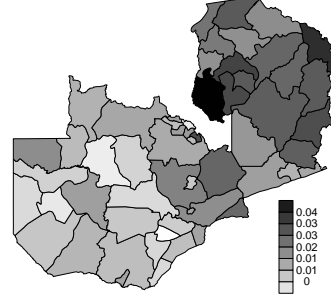
```
>formula = hazstd ~ edu1 + edu2 + tpr + sex + bmi +
+       f(agc, model="rw2") +
+       f(district, model="besag", graph.file="zambia.graph") +
+       f(distr.unstruct, model="iid")
```

The call to the `inla()` function is not changed. The estimated posterior mean and quantiles of the non-linear effect of age is plotted in Figure 2a and the non-linearity of the age effect is clear. The improvement obtained by using a more flexible model for the effect of the age covariate can be seen also from the decreased value of the DIC in Table 2. This second model results also to be more powerful as a prediction tool, as indicated by the decreased value of the logarithmic score in Table 2. Also for this model the predictive quantities are computed without problems and the `pit` distribution is close to uniform. The estimates of the other parameters in the model is reported in Table 1, the estimated spatial effects are similar to that in MOD1 and is not reported here. Results are similar to those obtained using BayesX, a program which performs a MCMC study [13].

As an alternative hypothesis to explain spatial variability one could imagine that the effect of one covariate, for example the mother's bmi, although being linear has a different slope for different regions. Again we assume this spatial variability to be

Table 2 DIC value and logarithmic score for the three model in the Zambia example.

	MOD1	MOD2	MOD3
mean of the deviance	13030.66	12679.90	12672.22
deviance of the mean	12991.61	12630.89	12610.68
effective number of parameters	39.04	49.01	61.53
DIC	13069.71	12728.92	12733.76
log Score	1.357	1.313	1.314

**Fig. 2a** Estimated effect of age (posterior mean together with 2.5% and 97.5% quantiles) using MOD2**Fig. 2b** Posterior mean for the space-varying regression coefficient in MOD3

smooth. This kind of models are known as space-varying regression models [10]. Using the notation in equation (2) the model (MOD3) can be written as:

$$\eta_i = \mu + \mathbf{z}_i^T \beta + f_1(\text{age}_i) + \text{bmi}_i f_2(s_i) \quad (7)$$

We assume here that the whole spatial variability is explained by the space varying regression parameter so that no other spatial effect is needed. Moreover we assume the age covariate to have a non linear effect. The model for $f_1(\cdot)$ is as in MOD2 while for $f_2(\cdot)$ we assume "besag" model, this time the sum-to-zero constraint is not necessary since there are no identifiability problem. Here the bmi covariate simply act as a known weight for the IGMRF $f_2(\cdot)$. The INLA specification of the model is as follows:

```
>formula = hazstd ~ edu1 + edu2 + tpr + sex +
+       f(age, model="rw2") +
+       f(district, bmi, model="besag",
+       graph.file="zambia.graph",
+       constr=FALSE)
```

The order of `district` and `bmi` in the second of the `f()` functions of the formula above is important since arguments are matched by position: the first argument is always the latent field and the second is always the weights. Note, moreover, that the sum-to-zero constraint has to be explicitly removed since, as said before, it is default for all intrinsic model. The resulting estimates for the space varying regression parameter are displayed in Figure 2b

The computing time (using the default CCD strategy) goes from a minimum of 9 seconds for MOD1 to a maximum of 14 seconds for MOD2.

5.3 A simple example of survival data analysis

Our last example comes from the survival analysis literature. A typical setting in survival analysis is that we observe the time point t at which the death of a patient occurs. Patients may leave the study (for some reason) before they die. In this case the survival time is said to be right censored, and t refers to the time point when the patient left the study. The indicator variable δ is used to indicate whether t refers to the death of the patient ($\delta = 1$) or to a censoring event ($\delta = 0$). The key quantity in modeling the probability distribution of t is the hazard function $h(t)$, which measures the instantaneous death rate at time t . We also define the cumulative hazard function $H(t) = \int_0^t h(s) ds$, implicitly assuming that the study started at time $t = 0$. A different starting time can also be considered and it is usually referred to as truncation time. The log-likelihood contribution from one patient is $\delta \log(h(t)) - H(t)$. A commonly used model for $h(t)$ is Cox's proportional hazard model [4], in which the hazard rate for the i^{th} patient is assumed to be on the form

$$h_i(t) = h_0(t) \exp(\eta_i), \quad i = 1, \dots, n.$$

Here, $h_0(t)$ is the baseline hazard function (common to all patients) and η_i is a linear predictor. In this example we shall assume that the baseline hazard belongs to the Weibull family: $h_0(t) = \alpha t^{\alpha-1}$ for $\alpha > 0$.

In [17] this model is used to analyze a data set on times to kidney infection for a set of $n_d = 38$ patients. The data set contains two observations per patient (the time to first and second recurrence of infection). In addition there are three covariates: age (continuous), sex (dichotomous) and type of disease (categorical, four levels), and an individual specific random effect (vn: ID), often named frailty: $u_i \sim N(0, \tau^{-1})$. Thus, the linear predictor becomes

$$\eta_i = \beta_0 + \beta_{sex} \text{sex}_i + \beta_{age} \text{age}_i + \beta_D \mathbf{x}_i + u_i \quad (8)$$

where $\beta_D = (\beta_2, \beta_3, \beta_4)$ and \mathbf{x}_i is a dummy vector coding for the disease type. Here we used a corner-point constraint imposing $\beta_1 = 0$.

Fitting a survival model using INLA is done using the following commands:

```
>formula = inla.surv(time,event) ~ age + sex + dis2 + dis3 +
```

```
+      dis4 + f(ID, model="iid")
>mod = inla(formula, family="weibull", data=Kidney)
```

Note that the function `inla.surv()` is needed to define the response variable of a survival model. This functions is used to define different censoring schemes such as right, left or interval censoring plus, possibly, truncation times. Including more complex effects in model (8) such as, for example, smooth effects of covariates or spatial effects can be done in exactly the same way as for the previous examples. Posterior means and standard deviations, together with quantiles, for the model parameters are shown in Table 3 and are similar to those obtained by Gibbs sampling via WinBUGS and by maximum likelihood. Fitting the model took less than 2 seconds.

Table 3 Posterior mean, standard deviation and quantiles for the parameters in the survival data example

	mean	sd	2.5%quant	97.5%quant
β_0	-4.809	0.954	-6.77	-3.103
β_{age}	0.003	0.016	-0.028	0.036
β_{sex}	-2.071	0.535	-3.180	-1.076
β_2	0.155	0.591	-1.007	1.346
β_3	0.679	0.595	-0.467	1.900
β_4	-1.096	0.863	-2.813	0.611
α	1.243	0.151	0.970	1.560
τ	2.365	1.647	0.586	6.976

The `inla()` function computes the posterior marginals for the hyperparameters $\theta = (\alpha, \tau)$ using only few points in the θ space (CCD strategy). As noted in Section 3 this might be not accurate enough. The function `inla.hyperpar()`, which takes as input the output of `inla()`, recomputes the marginals for the hyperparameters in a more accurate way, using a grid integration:

```
> hyperpar = inla.hyperpar(mod)
```

In cases where the dimension m of θ is large, computing posterior marginals using `inla.hyperpar()` can be time consuming. In this case, being $m = 2$, re-computing marginals for θ took less than 3 seconds. Figure 3 shows the two approximations for the hyperparameters in model 8. For the current example there is no special improvement coming from using the more accurate approximation computed by `inla.hyperpar()`.

The INLA library can also deal with exponential models for the baseline function $h_0(t)$. Semi-parametric models for $h_0(t)$ such as piecewise log-constant are, at the moment of writing, under study.

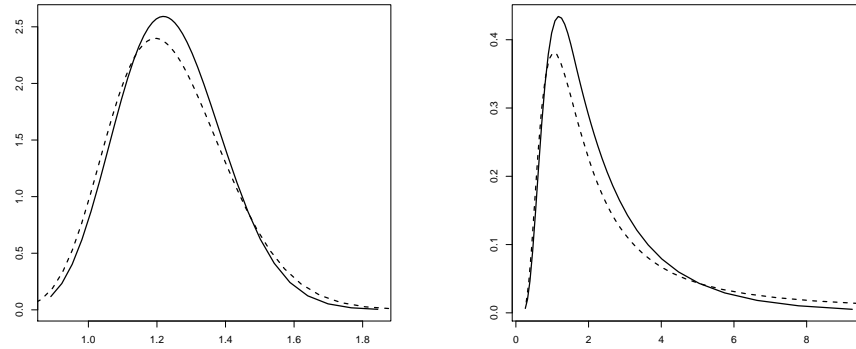


Fig. 3 Estimate of the posterior marginals for θ basic estimation (*solid line*) and improved one (*dashed line*). Left: posterior marginals for α . Right: posterior marginal for τ .

6 Discussion

As shown in this paper INLA is a powerful inferential tool for latent Gaussian models. The computational core of the `inla` program treats any kind of latent field in the same way thus behaving as a black-box. The available R interface `INLA`, can easily be handled by the user to obtain fast and reliable estimates. The large series of different options both for the approximations of the posterior marginals of the latent field, and for the exploration of the hyperparameter space may generate confusion in the novice user. On the other side, the default choices in the `INLA` library, offer usually a good starting point for the analysis.

The `INLA` library computes also quantities useful for model comparison, a feature that becomes important when the computational speed gives the possibility to fit several models to the same data set.

The `INLA` library contains also functions to process the posterior marginals obtained by the `inla()` function, so that it is possible to compute quantiles, percentiles or expectations of functions of the original random variable. Sampling from such posterior marginals is also possible.

References

1. Banerjee, S., Carlin, B., Gelfand, A.: Hierarchical Modeling and Analysis for Spatial Data. Chapman & Hall (2004)
2. Besag, J., York, J., Mollie, A.: Bayesian image restoration with two application in spatial statistics. *Annals of the Institute in Mathematical Statistics* **43**(1), 1-59. (1991)

3. Breslow, N.E. and Clayton D.G.: Approximate inference for Generalized Linear Mixed models. *Journal of the American Statistical Association*, **88**, 9-25. (1993)
4. Cox, D.R. : Regression models and life-tables. *Journal of the Royal Statistical Society, Series B*, **34**, 187-220 (1972)
5. Crowder, M.J. :Beta-Binomial ANOVA for Proportions. *Applied Statistics*, **27**, 34-37. (1978)
6. Czado, C., Gneiting, T. and Held, L.: Predictive model assessment for count data. *Biometrics* **65**, 1254-1261 (2009)
7. Eidsvik, J., Martino, S., Rue, H.: Approximate Bayesian Inference in spatial generalized linear mixed models. *Scandinavian Journal of Statistics*, **36** .(2009)
8. Fahrmeir, L., Tutz, G.: *Multivariate Statistical Modeling Based on Generalized Linear Models*. Springer Verlag, Berlin (2001)
9. Fong, Y., Rue H., and Wakefield J.: Bayesian inference for Generalized Linear Mixed Models, *Biostatistics*, doi:10.1093/biostatistics/kxp053 (2009)
10. Gamerman, D., Moreira, A. and Rue, H.:Space-varying regression models: specifications and simulation. *Computational Statistics & Data Analysis* **42** 513533 (2003)
11. Gneiting, T., Raftery, A.E.: Strictly proper scoring rules, prediction and estimation. *Journal of American Statistical Association, Series B*, **102**, 359-378. (2007)
12. Held, L., Schrodle, B., Rue, H.: Posterior and cross-validated predictive checks: A comparison of MCMC and INLA. In: Kneib, T., Tutz, G. (eds) *Statistical Modeling and Regression structures - Festschrift in Honour of Ludwig Fahrmeir*, Springer (2010)
13. Kneib, T., Lang, S. and Brezger, A.: Bayesian semiparametric regression based on MCMC techniques: A tutorial (2004)
14. Kandala, N. B., Lang, S., Klasen, S. and Fahrmeir, L.: Semiparametric Analysis of the Socio-Demographic and Spatial Determinants of Undernutrition in Two African Countries. *Research in Official Statistics*, **1**, 81100. (2001)
15. Lunn, D.J., Thomas, A., Best, N., and Spiegelhalter, D.: WinBUGS – a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, **10**:325–337, (2000)
16. Martino, S., Rue, H.: Implementing Approximate Bayesian inference using Integrated Nested Laplace Approximation: A manual for the INLA program. Technical report, Norwegian University of Science and Technology, Trondheim.
17. McGilchrist C.A., Aisbett C.W.: Regression with frailty in survival analysis, *Biometrics* **47**(2) 461-466, (1991).
18. Rue, H., Held, L.: *Gaussian Markov Random Fields: Theory and Applications*. Vol. 104 of *Monographs on Statistics and Applied Probability*, Chapman&Hall, London (2005)
19. Rue H., Martino S.: Approximate Bayesian Inference for Hierarchical Gaussian Markov Random Fields Models. *Journal of Statistical Planning and Inference*, **137** 3177-3192(2007)
20. Rue, H., Martino, S., Chopin, N.: Approximate Bayesian Inference for Latent Gaussian Models Using Integrated Nested Laplace Approximations (with discussion). *Journal of the Royal Statistical Society, Series B* **71**, 319-392 (2009)
21. Spiegelhalter, D., Best, N., Bradley, P., van der Linde, A.: Bayesian measure of model complexity and fit (with discussion). *Journal of the Royal Statistical Society, Series B*, **64**, 4, 583-639. (2002)
22. Tierney, L. and Kadane, J. B.: Accurate approximations for posterior moments and marginal densities. *J. Am. Statist. Ass.*, **81**, 82-86. (1986)