Joint Posterior Inference for Latent Gaussian Models and extended strategies using INLA

Dissertation by

Cristian Chiuchiolo

In Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy

King Abdullah University of Science and Technology

Thuwal, Kingdom of Saudi Arabia

June, 2022

EXAMINATION COMMITTEE PAGE

The dissertation of Cristian Chiuchiolo is approved by the examination committee

Committee Chairperson: Prof. Håvard Rue
Committee Members: Prof. David Bolin, Prof. Ajay Jasra, Prof. Simon Wilson

ABSTRACT

Joint Posterior Inference for Latent Gaussian Models and extended strategies using INLA

Cristian Chiuchiolo

Bayesian inference is particularly challenging on hierarchical statistical models as computational complexity becomes a significant issue. Sampling-based methods like the popular Markov Chain Monte Carlo (MCMC) can provide accurate solutions, but they likely suffer a high computational burden. An attractive alternative is the Integrated Nested Laplace Approximations (INLA) approach, which is faster when applied to the broad class of Latent Gaussian Models (LGMs). The method computes fast and empirically accurate deterministic posterior marginal approximations of the model's unknown parameters. In the first part of this thesis, we discuss how to extend the software's applicability to a joint posterior inference by constructing a new class of joint posterior approximations, which also add marginal corrections for location and skewness. As these approximations result from a combination of a Gaussian Copula and internally pre-computed accurate Gaussian Approximations, we name this class Skew Gaussian Copula (SGC). By computing moments and correlation structure of a mixture representation of these distributions, we achieve new fast and accurate deterministic approximations for linear combinations in a subset of the model's latent field. The same mixture approximates a full joint posterior density through a Monte Carlo sampling on the hyperparameter set. We set highly skewed examples based on Poisson and Binomial hierarchical models and verify these new approximations using INLA and MCMC. The new skewness correction from the Skew Gaussian Copula is more consistent with the outcomes provided by the default INLA strategies. In the

last part, we propose an extension of the parametric fit employed by the Simplified Laplace Approximation strategy in INLA when approximating posterior marginals. By default, the strategy matches log derivatives from a third-order Taylor expansion of each Laplace Approximation marginal with those derived from Skew Normal distributions. We consider a fourth-order term and adapt an Extended Skew Normal distribution to produce a more accurate approximation fit when skewness is large. We set similarly skewed data simulations with Poisson and Binomial likelihoods and show that the posterior marginal results from the new extended strategy are more accurate and coherent with the MCMC ones than its original version.

ACKNOWLEDGEMENTS

First, I want to thank my advisor Håvard Rue for the valuable, incomparable experience here at King Abdullah University of Science and Technology (KAUST) in Saudi Arabia. I learned a lot from him during these past years, and I recall the first time I joined his Bayesian Computational Statistics & Modeling (BAYESCOMP) group, where we were just a few members. I had the opportunity to grow as a researcher and expert in the Bayesian statistic field by seeking any opportunity and learning from other experts in different fields. I firmly believe this achievement would not have been possible without his guidance from academic and human perspectives. A similar mention goes to my senior colleague Janet van Niekerk, a Research Scientist in our group who provided immeasurable assistance and support in transposing my ideas into the present written work. I also thank the committee members: Prof. David Bolin, Prof. Ajay Jasra and Prof. Simon Wilson for being part of my Ph.D. defense and for their recognition of my research work. I also thank my group teammates and the people I met both in the Statistics and other campus departments, mainly the many Italians studying here who joined me in several fun activities. It is unthinkable not to be grateful to my friends at home in Italy, who helped me go through all the challenges I overcame during this new academic path in my life, and my family, who supported such a drastic decision to live and study so far away from home for such a long time. In particular, my brother Marco has been a central pillar for the entire family by properly taking care of my parents and myself. So he deserves to be recognized as a contributor to my past and future achievements, and my Ph.D. milestone is one of them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 General Overview

Approximate Bayesian theory always seeks new methods to recover the underlying posterior density assumption of the model at a minimum cost. When using approaches that rely on sampling from proposal distributions such as Markov Chain Monte Carlo (MCMC), we often need to come to terms with possible convergence issues and poor speed performance due to model dimension and complexity (Robert and Casella (2011)). In particular, hierarchical models can suffer from slow convergence to reach accurate stationary results if the model parameters are highly correlated. Even block joint sampling strategies using Gibbs-Metropolis algorithms or auxiliary variable methods may not be enough to reduce such high computational time. In other cases, convergence may not even approach an end, making the inference unfeasible. However, a sub-class of hierarchical models ignores most of these issues. This is the class of Latent Gaussian models (LGMs) whose latent field structure, containing all the unobserved parameters, is assumed to be Gaussian distributed. Like generalized linear regression models, these mathematical models follow a similar additive structure in the linear predictor which can also allow more functional and complex structures such as splines, time series or spatial fields. The additive structure combined with the Gaussian prior assumption onto the latent field offers a natural way to encode data information into a precision matrix. Compared to dense covariance matrices, these precision matrices are sparse and provide good storage solutions. Numerical

linear algebra approaches efficiently take care of this sparsity, leading to a speed-up in the computations. The Integrated Nested Laplace Approximation (INLA) approach (Rue et al. (2009)) takes the most advantage of these features when applied to Latent Gaussian Models. Such methodology completely bypasses the computational burden and diagnostic needs of sampling-based strategies. The INLA algorithm is deterministic and offers a user-friendly R software interface that computes univariate marginal approximations for Latent Gaussian Models. In this work, we dig into extending the methodology to joint approximations as well. As a statistical software for Bayesian statistical analysis, INLA provides a unique algorithm for tackling Bayesian problems by relying on Gaussian and Laplace approximations. We can find many applications in different scientific fields: health data analyses in Alvaro-Meca et al. (2013); Li et al. (2012); Seppä et al. (2019); spline models applied in medicine in Bauer et al. (2016) or spatial/Spatio-temporal models in Beguin et al. (2012); Gómez-Rubio and Palmí-Perales (2019); Gómez-Rubio et al. (2021); Yuan et al. (2017); Meehan et al. (2019); Peluso et al. (2020); Pereira et al. (2021); measurement error models in Muff et al. (2015); modelling applications in Quiroz et al. (2015); Ferkingstad et al. (2017); Sørbye et al. (2019a) and air data in Dawkins et al. (2019); a functional data analysis in Yue et al. (2019) and time trends for related populations in Riebler et al. (2012a,b); environmental data applications in Huang et al. (2017) and Illian et al. (2012) with some genetics in Holand et al. (2013); dynamic and stochastic volatility models respectively in Ruiz-Cárdenas et al. (2012); Martino et al. (2010a); joint models and survival applications in Martino et al. (2010b); Van Niekerk et al. (2019); Rustand et al. (2020). The advancements of the R-INLA project (see `www.r-inla.org`) are remarkable and have defined new standards for Bayesian statistics problem solving by enlarging the realm of possibilities. A neverending constant development of the software allows expanding the researcher's toolbox with new features and more options. This thesis aims to offer a detailed introduction to the software and its methodology

while discussing new extensions that enhance the performance of INLA marginal and joint approximations.

1.2 Objectives and Contributions

This thesis lays down theoretical details for applying joint posterior inference onto Latent Gaussian Models using R-INLA software from Rue et al. (2009) by introducing accurate mixtures of posterior approximations with marginal skewness adjustments. When Gaussian assumptions do not hold, we require the approximations to the target true posteriors to be more flexible in modeling extremely skewed observations. We use available INLA strategies to build marginal and joint approximations that encode location and skewness corrections. In order to deal with more extreme settings characterized by highly skewed data, we construct new approximations by introducing Gaussian copulas and other Skew Normal family densities. While Chapter 2 serves as an entry point for a general introduction of the INLA methodology and its applications, the main findings and new built-in tools of this thesis are given in Chapter 3 and Chapter 4. Chapter 2 gives a detailed background of the computational aspects of INLA from its core structure up to the use of Laplace approximations for fitting posterior outcomes. Essential concepts are Gaussian Markov Random Fields (GMRFs) to represent the latent field structure of the model (see Rue and Held (2005)) in terms of precision matrices, whose sparse structure provides a computational speed-up as opposed to dense covariance matrices. In Chapter 3 we introduce the class of Skew Gaussian Copula densities to build joint posterior approximations for Latent Gaussian Models, which combine a Gaussian Copula on the latent field whose marginals are being adjusted for location and skewness. This class of joint approximations improves the internal Gaussian Approximation in INLA used to achieve accurate posterior marginal approximations of the unknown parameters in the model. When embedded into R-INLA, a mixture representation of Skew Gaussian Copula densities opens possibilities for additional accurate and fast approximations. In well-defined subsets of the latent field, we can construct deterministic approximations for posterior marginals and additive linear combinations by analytically computing and matching

moments of this mixture representation with the ones from Skew Normal densities. Similarly, the same mixture representation of Skew Gaussian Copula densities can be used to achieve a full joint posterior approximation of a Latent Gaussian Model by taking into account the hyperparameter uncertainty as well. To accomplish this task, we exploit an exact Monte Carlo sampling approach on a discrete representation of the hyperparameter joint posterior density since the mixture representation does not have a deterministic form. All the approximations we get are closer to the truth and consistent with the results provided by INLA default strategies. Chapter 4 adds a new extension to one of the most used strategies in INLA known as Simplified Laplace strategy (see Rue et al. (2009) for details and Wood (2020) for an alternative version). This default approach is efficient in most cases as it ensures fast and accurate posterior marginal approximations while avoiding the more costly full Laplace strategy. Such results are obtained by matching higher-order derivatives of a third-order Taylor expansion of the Laplace Approximation to those derived from Skew Normal distributions. The Skew Normal fit may suffer inaccuracies in more extreme settings than the more accurate full Laplace approximations, especially around the mode. We proposed to reduce this accuracy gap by using another Skew Normal family density: the Extended Skew Normal distribution. This new parametric choice satisfies similar Gaussian boundaries and tail properties proper of the Gaussian and Skew Normal density but provides an additional parameter to better model skewness. Then we can extend the Simplified strategy by fitting instead Extended Skew Normal distributions to a fourth-order expanded Laplace approximation. The results show that the new parametric assumption can recover extreme skewed posterior outcomes better than the default Simplified strategy, therefore, encouraging to seek more alternative model solutions (other improvements towards Laplace approximations can be seen in Ruli et al. (2014); Ruli and Ventura (2016); Ruli et al. (2016)).

Chapter 2

Introduction to INLA: Integrated Nested Laplace Approximations

This chapter introduces the standalone R software named INLA (Integrated Nested Laplace Approximations) from Rue et al. (2009), whose first official release dates back to 2008 as a result of Håvard Rue's work. The software provides a fast and reliable deterministic approach for tackling complex Bayesian problems in the context of a broad class of hierarchical models known as Latent Gaussian Models (LGMs). It quickly computes approximations for the univariate posterior marginals of the unknown model parameters using strategies based on Gaussian and Laplace Approximations. Section 2.1 provides some general info of the INLA software correlated with a few insights on its origins. A mathematical formulation of the hierarchical structure of the class of Latent Gaussian Models is presented in Section 2.2. This class contains many well-known statistical models, and some of them are used as applied examples for this thesis. Prior assumptions on the latent field components are discussed in Section 2.3 where we introduce the mathematical concept of Gaussian Markov Random Fields (GMRFs) (see Rue and Held (2005)) and the importance of using sparse precision matrices. Section 2.4 provides a general overview of the Laplace Approximation technique, which is extensively used in INLA to obtain accurate empirical approximations from a mixture representation of target posterior marginals. Then Section 2.5 describes mathematical and computational details behind the available strategies, which allow an Approximate Bayesian analysis of Latent Gaussian Models. This introduction permits grasping the main details and advantages of the INLA

methodology, offering a solid background for both understanding the new joint posterior inference tools in Chapter 3 and the new proposed extended INLA strategy in Chapter 4.

2.1 INLA in the Bayesian World

Since its official appearance on the global statistic community Rue et al. (2009), INLA has quickly become incredibly popular. We can summarise this aspect in two factors: a user-friendly interface freely available in the R language, whose code relies on a combination of C/C++ and R/Fortran lines, and a deterministic algorithm that achieves solutions at high speed and accuracy with proper scaling properties. We must underline this point: *speed* and *accuracy* make INLA what it is nowadays. Well-known sampling-based approaches like Monte Carlo (MC) or Markov Chain Monte Carlo (MCMC) tackle Bayesian inference problems that provide the highest accurate results under correct assumptions. Therefore they represent the most natural counterpart to INLA in this field of applications. Still, they lack speed performance compared to INLA when applied to hierarchical models. Some MCMC based available software are JAGS (Plummer et al. (2003)), STAN (Carpenter et al. (2015),Carpenter et al. (2017)), BayesX (Lang et al. (2005)) and NIMBLE (de Valpine et al. (2017)). These programs provide ways to build procedural MCMC algorithms that are still slower than the scientific community would like them to be. Bayesian computational researchers always seek new alternatives to efficiently approach most modeling problems quickly while exploiting the full computer capacity. INLA is the first embodiment of natural program performance as it avoids sampling and post-diagnostic analyses, unlike its MCMC counterparts. The first ideas came out when it was clear that most statistical models could be cast in a broad, unique class with an additive linear model structure on its linear predictor component that is Gaussian distributed. This is the class of *Latent Gaussian Models* (LGMs), where the unknown parameters

of the model belong to a latent Gaussian structure, meaning that most of these are unobserved and Gaussian distributed. Many widely used statistical models like GLM, GAM, GLMM, GAMM belong to this class. For example, we can fit splines, time series, measurement error, or spatial and spatio-temporal models by encoding part of their structure into random effects of the Latent Gaussian Model framework. Thus, we can think of INLA as the product of three main mathematical concepts:

- Latent Gaussian Models, LGMs
- Gaussian Markov Random Fields, GMRFs
- Laplace Approximations, LAs

The following sections will go through the details of each item above, with particular emphasis on their correlation and synergy. These three topics define the main assumptions and tools needed for INLA to provide comparable and appealing outcomes in a Bayesian inference analysis.

2.1.1 Historical background of INLA

Before moving on to the details behind the INLA methodology, we first want to provide some historical context for a deeper insight. While the entire project required more than ten years of work, the first ideas appeared around 2002-2004 when Håvard Rue and Leonard Held became aware of Latent Gaussian Models. In the period 2002-2005, Rue developed a C-library code named `GMRFLib` for the book Rue and Held (2005) where he wrote and applied many algorithms involving Gaussian Markov Random Fields (GMRFs) and precision matrix factorizations that would have later played a major role in the main INLA core. The algorithm's first working user-friendly implementation was entirely built on C code in 2007, with many hand-crafted input files of varying length and complexity. Arnoldo Frigessi, who was living in Oslo, suggested that an R interface would have eased the spreading of the software

throughout the statistical community. Between January and February 2008, Sara Martino wrote a working prototype for the R-INLA interface. The first official INLA paper explaining its main ideas appeared in 2009 in the Royal Statistical Society Journal Rue et al. (2009). A second INLA paper regarding the INLA outbreak on the geostatistic field with the SPDE approach was published in 2011 again in the Royal Statistical Society Journal Lindgren et al. (2011). New extensions of the INLA algorithm were published the same year by Simpson et al. (2011). Furthermore, new features came out in 2013 with Martins et al. (2013). Here, the aim was to summarize the original INLA work from 2009 by adding more insights into the interpolation algorithms used to approximate the hyperparameter posterior marginals. In these years, more papers and books about the R-INLA software became available: more extensions on Martins and Rue (2014), a measurement error model focus on Muff et al. (2015), the concept of Penalised Complexity priors on Simpson et al. (2017), criticisms and Bayesian model diagnostics on Ferkingstad et al. (2017), spatial and spatio-temporal models on Blangiardo et al. (2013) and way more. For more books, software-related papers, and examples, one can check the official R-INLA website on `www.r-inla.org`. A first INLA review correlated with examples was published in 2017 by Rue et al. (2017). In May 2018, the new PARDISO (Schenk and Gärtner (2004)) library project for doing high-performance computing was embedded into the R-INLA interface. For more details about the PARDISO library and its use in INLA see the link `https://pardiso-project.org/r-inla/`. PARDISO represents one of the most efficient libraries for doing parallel computations on large sparse matrices according to Gould et al. (2007). A second INLA review on the original SPDE approach was published in 2018 by Bakka et al. (2018). In the same year, an advanced spatial book on INLA-SPDE applications by Krainski et al. (2018) became available. The first application of PARDISO library in INLA appeared on a joint models paper in 2019 by Van Niekerk et al. (2019). The third review of R-INLA was

published in 2019 by Martino and Riebler (2019). Nowadays, the complete source code of the INLA algorithm counts more than 10^6 lines in R/C/C++ languages.

2.2 Class of Latent Gaussian Models

Latent Gaussian Models (LGMs) are hierarchical statistical models whose linear predictor $\boldsymbol{\eta}$ has an additive structure with respect to the unknown parameters. This formulation allows significant computational advantages and accurate outcomes when other assumptions are involved. We can write each linear predictor term as

$$\eta_i = \gamma_0 + \sum_{j=1}^{n_J} \gamma_j z_{ij} + \sum_{k=1}^{n_K} f_k(u_{ik}) + \epsilon_i \quad \text{for} \quad i = 1, \ldots, n \tag{2.1}$$

where η_i corresponds to response observation y_i through a link function $g(\mu_i) = \eta_i$ in terms of the mean component μ_i with n being the total number of data observations. From (2.1) we observe the nature of different covariates that can be part of the model structure: γ_0 refers to the overall intercept of the model, the $\{\gamma_j\}$'s describe the fixed and random coefficients assigned to each covariate z_{ij} while $\{f_k(u_{ik})\}$'s represent unknown defined linear or non-linear functions associated with the covariates $\{u_{ik}\}$ in terms of each random effect k. The terms n_J and n_K denote the fixed and random covariate dimension while ϵ_i is a Gaussian distributed noise error. Since the linear predictor belongs to the latent field $\boldsymbol{x}$ structure in (2.1) by construction, the overall dimension would be $N = n + n_P$ where $n_P = n_J + n_K + 1$ denotes the total number of unknown parameters in the model to be estimated. The Gaussian assumption on the unstructured noise term ϵ_i is required to avoid singularity issues of the covariance matrix in the computations (see Section 2.2.3 for a detailed example). An alternative is to avoid adding the noise term in the linear predictor structure by using cumulative sums. Briefly we consider $\boldsymbol{\eta} = \boldsymbol{f}_1 + \boldsymbol{f}_2 + \cdots + \boldsymbol{f}_l$, where each $\boldsymbol{f}_i$ is a vector of fixed or random parameters of the model with some well-defined precision matrix structures.

Then we construct new variables $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_l\}$ such that $\boldsymbol{v}_{i+1} - \boldsymbol{v}_i = \boldsymbol{f}_i$. By construction, the joint density of these new variables is $\pi(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_l) = \pi(\boldsymbol{v}_1)\pi(\boldsymbol{v}_2|\boldsymbol{v}_1) \cdot \cdots \cdot \pi(\boldsymbol{v}_l|\boldsymbol{v}_{l-1})$ with a joint precision structure given by blocks of sum of the precision matrices related to each component $\boldsymbol{f}_i$. Thus we can directly apply posterior inference on $\boldsymbol{f}_1$ and $\boldsymbol{\eta}$ as $\boldsymbol{v}_1 = \boldsymbol{f}_1$ and $\boldsymbol{v}_l = \boldsymbol{\eta}$ while the same does not apply on the other model components which are now encoded as differences of the new variables. Such a strategy avoids possible mathematical issues of the covariance matrix and reduces its dimensionality at the cost of making inference for some of the original components harder. It can be useful in settings where the interest lies on a few parameters only or the linear predictor itself while avoiding singularity issues (see an application in Sørbye et al. (2019b)). Soon a new version of INLA where the linear predictor is not part of the latent field anymore will be officially available together with improved applications of the Laplace Approximation (see van Niekerk and Rue (2021); van Niekerk et al. (2022)). Depending on the nature of the functions $f_k(u_{ik})$'s we can fit many different statistical models. Here is a list of some of them

- GLM, GLMM, GAM, GAMM
- Measurement error models
- Dynamic models
- Splines, Semiparametric regression models
- Log Gaussian Cox processes
- Stochastic volatility models
- Disease mapping models using Besag, BYM and other stuctures
- Geostatistics models
- Survival analysis models, Joint models

- Longitudinal models
- Spatial and Spatio-temporal models

This list clarifies the significant flexibility of Latent Gaussian Models, therefore, enabling INLA to be applied on many different statistical problems. The additive structure of the linear predictor in (2.1) where each fixed, random, functional component is Gaussian distributed is greatly beneficial to this model framework and its applications in INLA. Such formulation leads to a well-defined representation of the latent field $\boldsymbol{x}$ into a Gaussian Markov Random Field (GMRF) representation.

2.2.1 Hierarchical structure of Latent Gaussian Models

Latent Gaussian Models follow a general hierarchical structure which is appealing for a Bayesian analysis using INLA. Therefore we must provide an accurate formulation. We consider data $\{y_i\}_{i=1}^n$ and unknown parameters $\boldsymbol{\theta} = (\boldsymbol{\theta_y}, \boldsymbol{\theta_x})$ where $\boldsymbol{\theta_y}$ are the hyperparameters assumed for the model likelihood while $\boldsymbol{\theta_x}$ accounts for the hyperparameter set of the latent field $\boldsymbol{x}$. We can then define a Latent Gaussian Model through a three-stage hierarchical structure as follows

$$\begin{aligned} \boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta_y} &\sim \prod_{i=1}^{n} \pi(y_i|x_i, \boldsymbol{\theta_y}) \\ \boldsymbol{x}|\boldsymbol{\theta_x} &\sim \mathrm{N}(\mathbf{0}, \boldsymbol{Q^{-1}}(\boldsymbol{\theta_x})) \\ \boldsymbol{\theta} &\sim \pi(\boldsymbol{\theta}) \end{aligned} \tag{2.2}$$

The second stage in the hierarchy assumes the latent field $\boldsymbol{x}$ to be Gaussian distributed a priori with a zero mean. A more general mean, which can also depend on hyperparameters, can be easily added at a later stage. This assumption is one of the most important for the model formulation since it leads to fast computational

performance when GMRFs are used (see Chapter 4 for details). The latent field is a random vector that contains all the model's unknown parameters where each term is linked to one and only one observation y_i. From equation (2.1), the latent field is

$$\boldsymbol{x} = \{\boldsymbol{\eta}, \gamma_0, \gamma_1, \gamma_2, \ldots, f_1(\cdot), f_2(\cdot) \ldots\} \tag{2.3}$$

which is assumed to be Gaussian distributed with a sparse precision matrix $\boldsymbol{Q}$. Here we notice that the linear predictor $\boldsymbol{\eta}$ indeed is part of the latent field object together with all the other parameters. In Bayesian terms, we point out that a precision matrix is the inverse of the covariance matrix denoted as $\boldsymbol{\Sigma}$. The multivariate Gaussian prior on $\boldsymbol{x}$ is the result of assuming Gaussian priors for each parameter in (2.3). The linear predictor $\boldsymbol{\eta}$ belonging to the latent field $\boldsymbol{x}$ is part of a scheme that allows stable computations in INLA (see Section 2.2.3 for an example). Using precision matrices instead of their covariance counterpart is not a random choice as attractive properties sustain it. Sparse precision matrices allow easy to handle manipulation and fast factorizations therefore leading to speed up in the computations when the sparse structure is significant (details on Section 2.3.3). The same does not apply to covariance matrices which are generally dense. In high dimensional settings, the cost becomes incredibly high due to many operations and storage issues. Both Gaussian assumptions of the latent field and sparsity in its precision matrix represent important properties for fast and accurate inference in INLA. A summary of these assumptions is given below:

1. $|\boldsymbol{\theta}|$ must be small, less than 20. This set of hyperparameters can appear both in the likelihood and latent field, affecting the number of operations required by INLA to compute the solutions;

2. the latent field $\boldsymbol{x}$ has to be Gaussian distributed with a sparse precision matrix $\boldsymbol{Q}$. More precisely, we say that $\boldsymbol{x}$ is required to satisfy some well-defined condi-

tional independent Markov properties. Later we will encode all these properties into a Gaussian Markov Random Field (GMRF) representation;

3. there must exist conditional independence between the data $\boldsymbol{y}$ and the entire set of parameters given by $(\boldsymbol{x}, \boldsymbol{\theta})$. This means that we have a one to one correspondence between each data point y_i and latent field component x_i where most of them are not observed.

From (2.2), we can easily compute the joint posterior distribution of a Latent Gaussian Model as follows

$$\begin{aligned}\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) &\propto \pi(\boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta}_{\boldsymbol{x}}) \prod_{i=1}^{n} \pi(y_i|x_i, \boldsymbol{\theta}_{\boldsymbol{y}}) \\ &\propto \pi(\boldsymbol{\theta})|\boldsymbol{Q}(\boldsymbol{\theta}_{\boldsymbol{x}})|^{\frac{1}{2}} \exp\Big(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}(\boldsymbol{\theta}_{\boldsymbol{x}})\boldsymbol{x} + \sum_{i=1}^{n} \log[\pi(y_i|x_i, \boldsymbol{\theta}_{\boldsymbol{y}})]\Big) \qquad (2.4)\end{aligned}$$

This joint distribution is unknown and can be hard to handle in high dimensional settings which is generally the case as both n and N can be large. The main INLA scope is to construct posterior marginal approximations for $\boldsymbol{x}$ and $\boldsymbol{\theta}$) independently. These resulting univariate approximations are empirically accurate and fast to compute by employing built in strategies (see Section 2.5). By their deterministic nature, we can use these approximations to get any posterior outcome for a Bayesian analysis: moment summaries, credible intervals, model diagnostics, and more. If we need information from the joint posterior density above for specific combinations of the model parameters (functionals), INLA can also achieve a corresponding joint approximation by relying on an exact Monte Carlo sampling approach with skewness corrected marginals (see Chapter 3 for details).

2.2.2 Insight on precision matrix Q

First, we should introduce why precision matrices have many preferable properties to covariance ones. The sparsity pattern of a precision matrix boosts INLA efficiency in computations and allows better storage of data observations within a LGM context. Using a more straightforward example of the structure in (2.2), we can point out another favorable result derived from the $\boldsymbol{Q}$ matrix structure.

Definition 1 (**Joint Q matrix of a 2-stage model**)

Consider a first stage data model $\boldsymbol{y} \sim \mathrm{N}(\boldsymbol{x}, \boldsymbol{Q_y^{-1}})$ *and a second stage of the structure* $\boldsymbol{x} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{Q_x^{-1}})$. *Then the joint precision matrix of the bivariate random variable* $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y})$ *is*

$$\boldsymbol{Q_z} = \begin{bmatrix} \boldsymbol{Q_x} + \boldsymbol{Q_y} & -\boldsymbol{Q_y} \\ -\boldsymbol{Q_y} & \boldsymbol{Q_y} \end{bmatrix} \tag{2.5}$$

This simple example underlines an essential key point that INLA exploits a lot. Anytime a change occurs in one of the parameters in the above structure, we need to recompute the joint matrix $\boldsymbol{Q_z}$. However, according to (2.5), there is hardly anything to recompute in this case since only one element in the sum changes. This factor is heavily present in the INLA framework since the joint precision matrix of the latent field may depend on one or more hyperparameters. Therefore, each hyperparameter change or point evaluation leads to a recomputation of the matrix. Based on the LGM structure in (2.2), it is undoubtedly more efficient to work with precision matrices to keep computations at their minimum. On the contrary, the joint structure of covariance matrices is more cumbersome and requires more operations when recomputing all the terms. When looking at this example, we can recall a particular case of the Kalman-Filter or Kalman update algorithm as the whole process is recursively updated every time we add a new term in the structure.

2.2.3 Singularity Feature in the latent field

From Section 2.2.1, we know that the latent field $\boldsymbol{x}$ is the prior stage on the unobserved parameters of a Latent Gaussian Model. This field contains the set of all unknown parameters augmented with the linear predictor parameter $\boldsymbol{\eta}$ itself. This augmentation is redundant in the fitting process but allows a straightforward approximation of the posterior linear predictor marginals. By encoding this information into the latent field, we are purposely forcing the resulting covariance structure $\boldsymbol{\Sigma}$ to be singular since the linear predictor term $\boldsymbol{\eta}$ is a linear combination of the same parameters plugged in the field. To avoid this singularity, we add a Gaussian noise term $\boldsymbol{\epsilon}$ to the linear predictor structure keeping the original assumptions untouched. We can look more into what this entails by setting a toy example. Consider a generic LGM model with one covariate $\boldsymbol{z}$ and no hyperparameters $\boldsymbol{\theta}$. Its linear predictor term would be

$$\boldsymbol{\eta} = \beta_0 \mathbb{1} + \beta_1 \boldsymbol{z} + \boldsymbol{\epsilon} \tag{2.6}$$

with independent parameters β_0 and β_1 and $\boldsymbol{\epsilon}$ being the additional noise parameter. Both the entire set of parameters $(\beta_0, \beta_1, \boldsymbol{\epsilon})$ and latent field $\boldsymbol{x} = \{\boldsymbol{\eta}, \beta_0, \beta_1\}$ are Gaussian distributed by model assumption. The noise must be small to minimize its effect in the fitting process and posterior results. The hyperparameter τ_ϵ keeps track of the noise effect and is assumed to be fixed and equal to exp(15) by default. This singularity feature represents an excellent ploy in the INLA methodology. Adding a noise quantity into the latent field partially solves the singularity issue in the covariance structure where its determinant would be close to zero. The corresponding joint posterior distribution of the model is

$$\pi(\boldsymbol{x}|\boldsymbol{y}) \propto \pi(\beta_0, \beta_1) \prod_{i=1}^{n} \pi(\eta_i|\beta_0, \beta_1) \prod_{i=1}^{n} \pi(y_i|\beta_0, \beta_1) \tag{2.7}$$

Based on Rue et al. (2009), Rue et al. (2017), the joint precision matrix is

$$\boldsymbol{Q}_{\text{joint}} = \begin{bmatrix} \tau_\epsilon \boldsymbol{I} & \tau_\epsilon \boldsymbol{I}\boldsymbol{z} & \tau_\epsilon \boldsymbol{I}\mathbb{1} \\ \tau_\epsilon \boldsymbol{z}^{\boldsymbol{T}} \boldsymbol{I} & \tau_{\beta_1} + \tau_\epsilon \boldsymbol{z}^{\boldsymbol{T}} \boldsymbol{z} & \tau_\epsilon \boldsymbol{z}^{\boldsymbol{T}} \mathbb{1} \\ \tau_\epsilon \mathbb{1}^{\boldsymbol{T}} \boldsymbol{I} & \tau_\epsilon \boldsymbol{z}^{\boldsymbol{T}} \boldsymbol{I} & \tau_{\beta_0} + \tau_\epsilon \mathbb{1}^{\boldsymbol{T}} \mathbb{1} \end{bmatrix} \tag{2.8}$$

which has dimension $(n+2) \times (n+2)$ as the data dimensionality is n. By inverting $\boldsymbol{Q}_{\text{joint}}$ we get the joint covariance structure

$$\boldsymbol{\Sigma}_{\text{joint}} = \begin{bmatrix} \sigma^2_{\eta_1} & \sigma_{\eta_1\eta_2} & \sigma_{\eta_1\eta_3} & \dots & \sigma_{\eta_1\eta_n} & -z_1\sigma^2_{\beta_1} & \sigma^2_{\beta_0} \\ \dots & \sigma^2_{\eta_2} & \sigma_{\eta_2\eta_3} & \dots & \sigma_{\eta_2\eta_n} & -z_2\sigma^2_{\beta_1} & \sigma^2_{\beta_0} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \sigma^2_{\eta_n} & -z_n\sigma^2_{\beta_1} & \sigma^2_{\beta_0} \\ \dots & \dots & \dots & \dots & \dots & \sigma^2_{\beta_1} & 0 \\ \dots & \dots & \dots & \dots & \dots & 0 & \sigma^2_{\beta_0} \end{bmatrix} \tag{2.9}$$

where $\sigma^2_{\eta_i} = \sigma^2_\epsilon + z_i^2\sigma^2_{\beta_1} + \sigma^2_{\beta_0}$ and $\sigma_{\eta_i\eta_j} = z_i z_j \sigma^2_{\beta_1} + \sigma^2_{\beta_0}$, $\forall i, j$. The noise variance of the parameter ϵ is identified by σ^2_ϵ while $(\sigma^2_{\beta_0}, \sigma^2_{\beta_1})$ are the other parameter variances. Although the toy example could be generalized to more parameters and covariates, we can already recognize a proper block pattern in the $\boldsymbol{\Sigma}_{\text{joint}}$ matrix structure in (2.9)

- We define the first block as $\boldsymbol{\Sigma}_{\sigma_{\eta_i}, \sigma_{\eta_i\eta_j}}$ which represents the $n \times n$ sub-matrix structure related to the linear predictor term $\boldsymbol{\eta}$. The noise parameter σ^2_ϵ belongs to this block matrix only

- The second block is given by $\boldsymbol{\Sigma}_{\beta_0,\beta_1}$ whose 2×2 block dimension contains the variance contribution from the independent parameters (β_0, β_1)

- the noise parameter σ^2_ϵ only appears in the sub-matrix $\boldsymbol{\Sigma}_{\sigma_{\eta_i}, \sigma_{\eta_i\eta_j}}$ by σ_{η_i} definition $\forall i$

- The remaining block structure is described by $\boldsymbol{\Sigma}_{-\boldsymbol{z}\sigma^2_{\beta_1}, \sigma^2_{\beta_0}} = [-\boldsymbol{z}\sigma^2_{\beta_1}, \mathbb{1}\sigma^2_{\beta_0}]$ and

its transpose.

Since the block joint covariance structure is known, we can exactly compute its determinant. For this example we see that

$$Det(\mathbf{\Sigma}_{\text{joint}}) = Det(\mathbf{\Sigma}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}})Det(\mathbf{\Sigma}_{\beta_0,\beta_1} - \mathbf{\Sigma}^T_{-\boldsymbol{z}\sigma^2_{\beta_1},\sigma^2_{\beta_0}}\mathbf{\Sigma}^{-1}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}\mathbf{\Sigma}_{-\boldsymbol{z}\sigma^2_{\beta_1},\sigma^2_{\beta_0}}) \tag{2.10}$$

where $\mathbf{\Sigma}_{\beta_0,\beta_1} - \mathbf{\Sigma}^T_{-\boldsymbol{z}\sigma^2_{\beta_1},\sigma^2_{\beta_0}}\mathbf{\Sigma}^{-1}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}\mathbf{\Sigma}_{-\boldsymbol{z}\sigma^2_{\beta_1},\sigma^2_{\beta_0}}$ is the Schur complement of the matrix $\mathbf{\Sigma}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}$. The determinant of the Schur complement is strictly positive while $Det(\mathbf{\Sigma}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}})$ depends on the noise term σ^2_ϵ. This latter determinant represents the additional information encoded in the latent field and should be close to zero due to the noise variance σ^2_ϵ. In this case we see that

$$\begin{aligned} Det(\mathbf{\Sigma}^{(k)}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}) = (\sigma^2_\epsilon)^k + (\sigma^2_\epsilon)^{k-1}\Big[\sum_{i=1}^{k} z_i\sigma^2_{\beta_1} + k\sigma^2_{\beta_0}\Big] \\ + \mathcal{I}_{2\leq k\leq n}\Big\{(\sigma^2_\epsilon)^{k-2}\Big[(k-1)\sigma^2_{\beta_0}\sigma^2_{\beta_1}\sum_{i=1}^{k} z_i^2 - 2\sigma^2_{\beta_0}\sigma^2_{\beta_1}\sum_{j=1}^{k}\sum_{i<j} z_i z_j\Big]\Big\} \end{aligned} \tag{2.11}$$

where $k < n$ refers to a sub block of the original $n\times n$ matrix $\mathbf{\Sigma}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}$ and $\mathcal{I}_{2\leq k\leq n}\{\dots\}$ is an indicator function of terms appearing at $k \geq 2$. The notation $\mathbf{\Sigma}^{(k)}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}$ defines the sub-matrix of order k of the original $n \times n$ covariance matrix $\mathbf{\Sigma}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}$ related to the linear predictor. A simple proof to verify the result for this example can be obtained by induction on k.

Proof. First we underline two points

- the expression strongly depends on the noise parameter σ^2_ϵ which is close to zero
- the determinant of $\mathbf{\Sigma}^{(k)}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}$ by degree k is equal to the product of its eigenvalues

Then we proceed by induction. Assuming the expression to be true for any degree k, then we have

$$Det(\mathbf{\Sigma}^{(k)}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}) = \lambda_1\lambda_2\prod_{i=2}^{k}\sigma_\epsilon^2 = \lambda_1\lambda_2(\sigma_\epsilon^2)^{k-2}, \quad \text{for } k \geq 2 \tag{2.12}$$

where (λ_1, λ_2) are two eigenvalues strictly positive. It is straightforward to show that the cases $k = 1, 2$ are true. For $k = 1$ we obtain a block with one element only

$$\sigma_\epsilon^2 + z_1^2\sigma_{\beta_1}^2 + \sigma_{\beta_0}^2 \tag{2.13}$$

which is strictly positive and different from σ_ϵ^2. Hence, it provides the only eigenvalue λ_1 of $Det(\mathbf{\Sigma}^{(1)}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}})$ as defined in (2.12). Similarly, the case $k = 2$ ends up being

$$(\sigma_\epsilon^2)^2 + \sigma_\epsilon^2[z_1^2\sigma_{\beta_1}^2 + z_2^2\sigma_{\beta_1}^2 + 2\sigma_{\beta_0}^2] + \sigma_{\beta_1}^2\sigma_{\beta_0}^2(z_1 - z_2)^2 \tag{2.14}$$

which is strictly positive and different from σ_ϵ^2. As given in (2.12), this result corresponds to $Det(\mathbf{\Sigma}^{(2)}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}})$ defined as the product of the two eigenvalues of the matrix structure $\lambda_1\lambda_2$. To finalize the proof by induction, we need to verify that the formula is true for $k + 1$. For the case $k + 1$ with $k > 2$, we get

$$
\begin{aligned}
&(\sigma_\epsilon^2)^{k+1} + (\sigma_\epsilon^2)^k \left[\sum_{i=1}^{k+1} z_i \sigma_{\beta_1}^2 + (k+1)\sigma_{\beta_0}^2\right] + (\sigma_\epsilon^2)^{k-1}\left[k\sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{i=1}^{k+1} z_i^2 - 2\sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{j=1}^{k+1}\sum_{i<j} z_i z_j\right] \\
&= \sigma_\epsilon^2 \Bigg\{ (\sigma_\epsilon^2)^k + (\sigma_\epsilon^2)^{k-1}\left[\sum_{i=1}^{k} z_i \sigma_{\beta_1}^2 + z_{k+1}\sigma_{\beta_1}^2 + k\sigma_{\beta_0}^2 + \sigma_{\beta_0}^2\right] \\
&+ (\sigma_\epsilon^2)^{k-1}\Bigg[(k-1)\sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{i=1}^{k} z_i^2 + \sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{i=1}^{k} z_i^2 + k\sigma_{\beta_0}^2\sigma_{\beta_1}^2 z_{k+1} \\
&- 2\sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{j=1}^{k}\sum_{i<j} z_i z_j - 2\sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{i=1}^{k} z_i z_{k+1}\Bigg]\Bigg\} \\
&= \sigma_\epsilon^2 \Bigg\{ \lambda_1\lambda_2(\sigma_\epsilon^2)^{k-2} + (\sigma_\epsilon^2)^{k-1}\Big(z_{k+1}\sigma_{\beta_1}^2 + \sigma_{\beta_0}^2\Big) \\
&+ (\sigma_\epsilon^2)^{k-2}\left[\sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{i=1}^{k} z_i^2 + k\sigma_{\beta_0}^2\sigma_{\beta_1}^2 z_{k+1} - 2\sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{i=1}^{k} z_i z_{k+1}\right]\Bigg\} \\
&= \sigma_\epsilon^2 \Bigg\{ \left[\lambda_1\lambda_2 + \sigma_\epsilon^2\Big(z_{k+1}\sigma_{\beta_1}^2 + \sigma_{\beta_0}^2\Big) + \sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{i=1}^{k} z_i^2 + k\sigma_{\beta_0}^2\sigma_{\beta_1}^2 z_{k+1} - 2\sigma_{\beta_0}^2\sigma_{\beta_1}^2 \sum_{i=1}^{k} z_i z_{k+1}\right](\sigma_\epsilon^2)^{k-2}\Bigg\} \\
&= \sigma_\epsilon^2 \Bigg\{ \lambda_1^*\lambda_2^*(\sigma_\epsilon^2)^{k-2}\Bigg\} \\
&= \lambda_1^*\lambda_2^*(\sigma_\epsilon^2)^{k-1} \qquad (2.15)
\end{aligned}
$$

which exactly corresponds to $Det(\mathbf{\Sigma}^{(k+1)}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}})$ as defined in (2.12).

□

As soon as $k > 2$, the determinant quickly drops to zero since the noise variance term $\sigma_\epsilon^2 = 1/\exp(15)$ by default. Algebrically, σ_ϵ^2 represents the eigenvalue of $\mathbf{\Sigma}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}$ with algebric multiplicity equal to $k-2$. In general the determinant of $\mathbf{\Sigma}_{\text{joint}}$ would be of the form

$$
Det(\mathbf{\Sigma}_{\text{joint}}) = \tilde{\lambda}_1\tilde{\lambda}_2(\sigma_\epsilon^2)^{n-2} Det(\mathbf{\Sigma}_{\beta_0,\beta_1} - \mathbf{\Sigma}^T_{-\boldsymbol{z}\sigma_{\beta_1}^2,\sigma_{\beta_0}^2}\mathbf{\Sigma}^{-1}_{\sigma_{\eta_i},\sigma_{\eta_i\eta_j}}\mathbf{\Sigma}_{-\boldsymbol{z}\sigma_{\beta_1}^2,\sigma_{\beta_0}^2}) \qquad (2.16)
$$

with $\tilde{\lambda}_1$ and $\tilde{\lambda}_2$ being the two strictly positive eigenvalues related to the parameters (β_0, β_1) and $n-2$ eigenvalues exactly equal to $\sigma_\epsilon^2 \approx 0$. The expression in (2.11) underlines that the more linear predictors we consider, the more quickly we approach a determinant close to zero as the noise parameter σ_ϵ^2 scale all the terms involved. In this simple toy example, we expect singularity issues at dimension three, where the noise effect starts to kick in. More extended examples are possible by considering more than two parameters or more random structures, but we avoid it for simplicity. New improvements has been made towards this methodology which entirely avoids such issue by not adding the linear predictor component into the latent field (see van Niekerk et al. (2022)).

2.2.4 Conditional Independence

While adding a Gaussian noise term to the linear predictor eases inference in Latent Gaussian Models, sparsity of the precision matrix is another important feature that we must consider. We introduce the concept of conditional independence, which allows having useful Markov properties between the latent field terms. Through this conditional assumption and the multivariate Gaussian prior on the field, we can adequately encode the latent random vector of a Latent Gaussian Model as a Gaussian Markov Random Field (GMRF) object. A first probabilistic definition of the conditional independence is the following

Definition 2 (**Conditional Independence**)

Consider three random variables X, Y and Z with respective distributions $\pi(x)$, $\pi(y)$ and $\pi(z)$. Then

$$X \perp Y|Z \Leftrightarrow \pi(x, y|z) = \pi(x|z)\pi(y|z) \tag{2.17}$$

This reads as X and Y are conditionally independent given Z if and only if the joint distribution of (X, Y) given Z can be expressed as the product of the univariate

conditional distributions of X and Y given Z.

According to this definition, there exists a subtle link between covariance $\boldsymbol{\Sigma}$ and precision matrix $\boldsymbol{Q}$. Elements are marginally independent for covariance matrices, while the same are conditional independent for precision matrices. Hence, we cannot add much about the information stored in the covariance matrix structure. Instead it is possible to encode the conditional independent information onto precision matrix structures (see Section 2.3). We can verify the property above through the factorization theorem

Theorem 1 (**Factorization Criterion**)

Using the same notation of the previous definition we have

$$X \perp Y|Z \Leftrightarrow \pi(x, y, z) = f(x, z)g(y, z) \tag{2.18}$$

for some functions $f(\cdot)$,$g(\cdot)$ and for all Z with $\pi(z) > 0$.

The proof of the theorem comes from applying the conditional independence property. These results translate into Markov properties of the latent field structure $\boldsymbol{x}$ and lay the foundation of the whole GMRF theory. We will see that the sparsity structure of the precision matrix is partially induced by applying (2.17) and (2.18).

2.3 Gaussian Markov Random Fields (GMRFs)

With the multivariate Gaussian prior assumption of the latent field $\boldsymbol{x}$ and the conditional Markov properties onto its terms, we can now define a Gaussian Markov Random Field (GMRF). This mathematical object is summarised by a random vector $\boldsymbol{x} = (x_1, \ldots, x_N)^T$ whose distribution is Gaussian with a sparse precision matrix $\boldsymbol{Q}$. The Markov properties are strictly related to the conditional independence definition in (2.17) as

$$x_i \perp x_j | \boldsymbol{x}_{-ij} \tag{2.19}$$

for any well defined set $\{i, j\}$ of indexes where $\boldsymbol{x}_{-ij}$ defines the whole random vector except x_i and x_j. If these conditions apply, the GMRF must encode this conditional information into its structure. Although conditional independent elements cannot be translated into a covariance matrix structure as it contains marginal information, this does not apply for precision matrices $\boldsymbol{Q}$. This property assumes consistency as soon as we consider the following theorem

Theorem 2 (**Q sparsity**)

$$x_i \perp x_j | \boldsymbol{x}_{-ij} \Leftrightarrow Q_{ij} = 0 \tag{2.20}$$

where its proof is accomplished by using the factorization criterion in (2.18) for a fixed set $\{i, j\}$ (for more details, see Chapter 2 in Rue and Held (2005)). We can then extrapolate conditional independence for a pair of terms x_i and x_j from the precision matrix zero patterns and viceversa. A similar result applies to covariance matrices where a zero element translates into independence. However, independence is a strong assumption, while conditional independence is more flexible and interpretable. Now we can define a GMRF through its $\boldsymbol{Q}$-matrix representation

Definition 3 (**Gaussian Markov Random Field (GMRF)**)
A random vector $\boldsymbol{x} = (x_1, \ldots, x_N)^T \in \mathcal{R}^N$ *is referred as a GMRF with respect to the (undirected) graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ *with mean* $\boldsymbol{\mu}$ *and precision matrix* $\boldsymbol{Q} > 0$*, if and only if its density has the form*

$$\pi(\boldsymbol{x}) = (2\pi)^{-\frac{N}{2}} |\boldsymbol{Q}|^{\frac{1}{2}} \exp\Big(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{Q} (\boldsymbol{x} - \boldsymbol{\mu})\Big) \tag{2.21}$$

and

$$Q_{ij} = 0 \Leftrightarrow \{i, j\} \in \mathcal{E} \quad \textit{for all} \quad i \neq j \tag{2.22}$$

The (undirected) graph $\mathcal{G}$ is used to represent the conditional independency property of (2.17) in the GMRF. This graph representation is constituted by a set of *nodes* $\mathcal{V} = \{1, \ldots, N\}$ and a set of *edges* $\{i, j\}$ that belong to $\mathcal{E}$. If a pair of nodes $\{i, j\}$ belongs to $\mathcal{E}$ then there exists an (undirected) edge connecting both otherwise there is not. Additionally a graph $\mathcal{G}$ is fully connected if $\{i, j\} \in \mathcal{E}$ for all i,$j \in \mathcal{V}$ with $i \neq j$. Based on the conditional independence property we can derive a few important Markov properties with the most useful and general being the following (for the others see Chapter 2 in Rue and Held (2005))

Definition 4 (**Global Markov property**)

Define $\boldsymbol{x}$ *to be a GMRF with respect to the graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. *The* Global Markov property

$$\boldsymbol{x_A} \perp \boldsymbol{x_B} | \boldsymbol{x_C} \tag{2.23}$$

for all disjoint sets A,B and C where C separates A and B with A and B not empty sets.

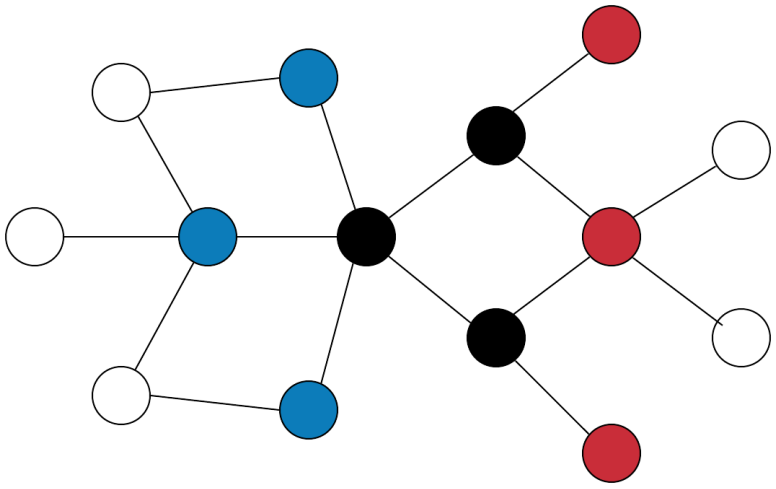


Figure 2.1: Common Graph illustration of the Global Markov property. The blue circles define the nodes of the subset A, the red ones the subset B while the black ones constitute the subset C.

Conditional Independence implies no path from one subset to another, with a third subset separating the first two. It also implies that several big blocks of elements in the precision matrix only contain zeros. This zero pattern is fundamental: if we know that some terms of the latent field $\boldsymbol{x}$, encoded in $\boldsymbol{Q}$, are zeros, then we can figure out a similar zero pattern for related neighboring elements as well. In this way, we consider a chain of operations that we do not need to apply since zero values do not add any valuable information to the whole model structure, which saves computational time. An example is given by time series models, which have large sparsity structures in their precision matrix representations due to the conditional independence property. Indeed, autoregressive models have severe sparse precision matrices. An autoregressive model of order p has a $(2p+1)$-diagonal precision matrix $\boldsymbol{Q}$ whose pattern allows fast computations in INLA.

2.3.1 Cholesky Factorization

Sparse matrices allow more accessible storage and faster computations by using the *Cholesky Factorization* algorithm. In most cases, we know that only $\mathcal{O}(N)$ of the N^2 terms in $\boldsymbol{Q}$ are non-zero, and we can take advantage of its sparsity by exploiting the respective zero pattern. The computations are faster on sparse matrices compared to dense ones since the cost for factorizing a dense matrix is generally $\mathcal{O}(N^3)$. A few noteworthy examples that belong to this category are

- Temporal GMRF models, $\mathcal{O}(N)$ cost with $\boldsymbol{Q}$ sparse
- Spatial GMRF models, $\mathcal{O}(N^{3/2})$ cost with $\boldsymbol{Q}$ sparse
- Spatio-Temporal GMRF models, $\mathcal{O}(N^2)$ cost with $\boldsymbol{Q}$ sparse

First we introduce the *Cholesky Decomposition* as follows

Definition 5 (**Cholesky Triangle** L)
If $\boldsymbol{A}$ *is a* $N \times N$ *symmetric positive definite (SPD) matrix, then there exists a unique* Cholesky Traingle $\boldsymbol{L}$ *such that* $\boldsymbol{L}$ *is a lower triangular matrix and*

$$\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^T \tag{2.24}$$

The computation of $\boldsymbol{L}$ *involves* $N^3/3$ *operations.*

This factorization represents the basic step for solving linear systems like $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ or $\boldsymbol{A}\boldsymbol{X} = \boldsymbol{B}$ or equivalently $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ or $\boldsymbol{X} = \boldsymbol{A}^{-1}\boldsymbol{B}$. Forward and Backward loop steps contribute to the following linear system of equations

Algorithm 1 Solving $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ with $\boldsymbol{A} > 0$

Input: A cholesky factorization $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^T$ with $\boldsymbol{L}$ being the Cholesky Triangle
Output: Return linear system solutions $\boldsymbol{x}$
1: Solve $\boldsymbol{L}\boldsymbol{v} = \boldsymbol{b}$
2: Solve $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{v}$

Step 1 is called *forward substitution* and provides the vector solution $\boldsymbol{v}$ by using a forward loop

$$v_i = \frac{1}{L_{ii}}\Big(b_i - \sum_{j=1}^{i-1} L_{ij} v_j\Big) \quad \text{for} \quad i = 1, \ldots, N \tag{2.25}$$

while Step 2 represents the *back substitution* and computes the vector solution $\boldsymbol{x}$ in a backward loop

$$x_i = \frac{1}{L_{ii}}\Big(v_i - \sum_{j=i+1}^{N} L_{ji} x_j\Big) \quad \text{for} \quad i = N, \ldots, 1 \tag{2.26}$$

Both looping operations cost $\mathcal{O}(N^2)$. We apply similar operations to the general case $\boldsymbol{AX} = \boldsymbol{B}$ as well. This version is used to deal with constrained GMRF where mean and covariance structures change accordingly (a constrained GMRF note is given on Chapter 3 for the joint Gaussian Approximation). We can also draw samples from a GMRF structure denoted by $\boldsymbol{x}$ using (2.24) and the operations listed in (1)

Algorithm 2 Sampling $\boldsymbol{x} \sim \mathrm{N}(\boldsymbol{\mu}, \boldsymbol{Q}^{-1})$

Input: A cholesky factorization $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^T$ with $\boldsymbol{L}$ being the Cholesky Triangle
Output: Return $\boldsymbol{x}$ where $\boldsymbol{x}$ can be a GMRF

1: Sample $\boldsymbol{z} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{I})$
2: Solve $\boldsymbol{L}^T\boldsymbol{v} = \boldsymbol{z}$
3: Compute $\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{v}$

If $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^T$ and $\boldsymbol{z} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{I})$, then $\boldsymbol{x}$ defined by $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ has covariance

$$\mathrm{Cov}(\boldsymbol{x}) = \mathrm{Cov}(\boldsymbol{L}^{-T}\boldsymbol{z}) = (\boldsymbol{L}\boldsymbol{L}^T)^{-1} = \boldsymbol{Q}^{-1} \tag{2.27}$$

We factorize the precision matrix and solve the linear system by sampling the process backward. Through the resulting factorization, we also compute the determinant of $\boldsymbol{Q}$, which corresponds to the determinant of the lower triangular matrix squared.

2.3.2 Numerical computations for sparse matrices

This section provides numerical results of precision matrix sparse computations in a GMRF framework. We use numerical sparse linear algebra algorithms to achieve fast factorizations at minimum cost. In Section 2.3.1 we summarised all these operations in two tasks

1. Compute the Cholesky Factorization in (2.24) for sparse precision matrix $\boldsymbol{Q}$
2. Employ the Forward and Backward algorithms in (1) to solve $\boldsymbol{Lv} = \boldsymbol{b}$ and $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ respectively

The second task is faster to accomplish than the first one as sparsity plays a significant role. The first task hides no particular trick since we avail the zero pattern of the precision matrix $\boldsymbol{Q}$ allowing the process to avoid computing the zero terms. Hence, we decompose each precision term into sums of their respective Cholesky lower triangular matrix as follows

$$Q_{ij} = \sum_{k=1}^{j} L_{ik}L_{jk}, \quad i \geq j \tag{2.28}$$

$$v_i = Q_{ij} - \sum_{k=1}^{j-1} L_{ik}L_{jk}, \quad i \geq j \tag{2.29}$$

These equations compute all terms within the $\boldsymbol{L}$ structure. The solutions are

- $L_{jj}^2 = v_j$ and $L_{ij}L_{jj} = v_i, \quad i > j$
- if we know $\{v_i\}$ for fixed j, then $L_{jj} = \sqrt{v_j}$ and $L_{ij} = \frac{v_i}{\sqrt{v_j}}$ for $i = j+1, \ldots, N$

This approach shows how to get the j^{th} column of the Cholesky Triangle $\boldsymbol{L}$ and, therefore, the whole lower triangular matrix structure. This process represents the first step of factorizing $\boldsymbol{Q}$ and take advantage of its sparsity structure as well. Still,

it is not clear if this decomposition of $\boldsymbol{Q}$ into $\boldsymbol{L}$ matrices has a real benefit to our computations. Thus, we consider a representation of $\boldsymbol{L}$ where its pattern determines each GMRF term of $\boldsymbol{x}$ as follows

Definition 6 (**Determination of x through L**)
Consider the decomposition $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^T$, *then the solution of* $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ *where* $\boldsymbol{z} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{I})$ *is* $\mathrm{N}(\boldsymbol{0}, \boldsymbol{Q}^{-1})$. *Since L is lower triangular then*

$$\begin{aligned} x_N &= \frac{1}{L_{NN}} z_N \\ x_{N-1} &= \frac{1}{L_{N-1,N-1}} (z_{N-1} - L_{N,N-1} x_N) \\ &\dots \end{aligned} \tag{2.30}$$

Since z_N is standard normal, the variance of a marginal distribution x_N is equal to the square inverse diagonal element L_{NN}. A similar result holds for expression x_{N-1} if we condition on the future term x_N of the process. By conditioning the previous term x_{N-1} on the future term x_N we are assuming x_N to be constant. Doing so brings interesting results: the conditional variance $\mathrm{Var}(x_{N-1}|x_N)$ is only related to the diagonal terms of $\boldsymbol{L}$ while the conditional mean $\mathrm{E}(x_{N-1}|x_N)$ is strictly linked to the off-diagonal elements $L_{N,N-1}x_N$. These results are summarised in the following theorem

Theorem 3 (**Alternative representation of a GMRF**)
Define $\boldsymbol{x}$ *to be a GMRF with respect to the labelled graph* $\mathcal{G}$, *with mean* $\boldsymbol{\mu} = \boldsymbol{0}$ *and precision matrix* $\boldsymbol{Q} > 0$. *We denote* $\boldsymbol{L}$ *as the Cholesky triangle of* $\boldsymbol{Q}$. *Then for* $i \in \mathcal{V}$, *we have*

$$\mathrm{E}(x_i|\boldsymbol{x}_{(i+1):N}) = -\frac{1}{L_{ii}} \sum_{j=i+1}^{N} L_{ji} x_j \tag{2.31}$$

$$\mathrm{Var}(x_i|\boldsymbol{x}_{(i+1):N}) = \frac{1}{L_{ii}^2} \tag{2.32}$$

The expressions above define a sequential representation of a zero-mean GMRF backward in time that can be seen as a backward autoregressive time series model given by

$$x_i|x_{i+1}, \ldots, x_N \sim \mathrm{N}\Big(-\frac{1}{L_{ii}} \sum_{j=i+1}^{N} L_{ji} x_j, \frac{1}{L_{ii}^2}\Big) \quad i = N, \ldots, 1 \tag{2.33}$$

The elements of $\boldsymbol{L}$ define conditional properties of the GMRF terms by conditioning to the future. Indeed, the off-diagonal elements of $\boldsymbol{L}$, conditioned on the higher-order terms of the GMRF $\boldsymbol{x}$, contribute to the conditional mean of the GMRF elements. In contrast, the diagonal elements are the only ones that appear in the conditional variance expression. If we have x_i and x_j conditional independent as in (2.17) (assuming x_j fixed), then we get $L_{ji} = 0$ according to the conditional expressions in the theorem above. Conditional independence is then a powerful property when applied on the distribution of x_i and the higher-order terms $\boldsymbol{x}_{(i+1):N}$ since it determines the zero pattern of $\boldsymbol{L}$. Also, theorem 3 provides another insight when considering the zero pattern of $\boldsymbol{L}$

Theorem 4 (**Zero Pattern of L**)

Define $\boldsymbol{x}$ *to be a GMRF with respect to* $\mathcal{G}$*, with mean* $\mathbf{0}$ *and precision matrix* $\boldsymbol{Q} > 0$*. Again we denote* $\boldsymbol{L}$ *as the Cholesky triangle of* $\boldsymbol{Q}$ *and define for* $1 \leq i < j \leq N$ *the set*

$$\mathrm{FU}(i,j) = \{i+1,\ldots,j-1,j+1,\ldots,N\} \tag{2.34}$$

which is the future of i except j, meaning that it contains all indexes higher than i but not j. Then

$$x_i \perp x_j | \boldsymbol{x}_{\mathrm{FU}(i,j)} \Leftrightarrow L_{ji} = 0 \tag{2.35}$$

Proof. To keep things simple, we assume $\boldsymbol{\mu} = 0$ and fix a pair of indexes $\{i,j\}$ such that $1 \leq i < j \leq N$. By using Theorem 3 we have that

$$\pi(\boldsymbol{x}_{i:N}) \propto \exp\Big(-\frac{1}{2}\sum_{k=i}^{N} L_{kk}^2 (x_k + \frac{1}{L_{kk}}\sum_{j=k+1}^{N} L_{jk}x_j)^2\Big) \tag{2.36}$$

$$= \exp\Big(-\frac{1}{2}\boldsymbol{x}_{i:N}^T \boldsymbol{Q}^{(i:N)} \boldsymbol{x}_{i:N}\Big) \tag{2.37}$$

where $Q_{ij}^{(i:N)} = L_{ii}L_{ji}$. Finally Theorem 2 leads to

$$x_i \perp x_j | \boldsymbol{x}_{\mathrm{FU}(i,j)} \Leftrightarrow L_{ii}L_{ji} = 0 \tag{2.38}$$

which is equivalent to the proof we need, that is, $L_{ji} = 0$ is the acceptable result since $L_{ii} > 0$ as $\boldsymbol{Q}^{(i:N)} > 0$. □

The implications of these results are important. If we know that L_{ji} is zero, then we do not need to compute it when factorizing the precision matrix $\boldsymbol{Q}$, and this saves computations. However, Theorem 4 does not provide any insight in detecting which terms of the Cholesky triangle $\boldsymbol{L}$ are zero a priori as it only shows conditional independence properties of the marginals x_i from i up to N. The most natural idea would be to compute $\boldsymbol{L}$ and check if L_{ji} term is effectively zero, but this would not be

very meaningful for our computational saving purposes. We need weaker properties to make the whole approach feasible. The Global Markov property in (2.23) is a natural candidate for this role. Theorem 4 describes numerical properties of the matrix terms, while the Markov property gives information of the underlying graph structure. There is no need to know the numerical values to apply these conditional Markov properties because the results are strictly related to the graph structure. We say here that for any identical graph structure, we get the same results while the numerical values encoded in that structure are free to change without affecting the computations. Through the following Corollaries, the Global Markov property ensures a sufficient criterion for checking if $L_{ji} = 0$ as stated by Theorem 4

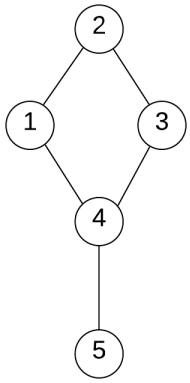
Corollary 1 (**Separate Future Set**)
If $\mathrm{FU}(i, j)$ *separates* $i < j$ *in* $\mathcal{G}$*, then* $L_{ji} = 0$.

Corollary 2 (**Neighbors Representation**)
If $i \sim j$ *then* $\mathrm{FU}(i, j)$ *does not separate* $i < j$.

The following two points summarise the whole methodology

1. use the Global Markov property to check if $L_{ji} = 0$
2. only compute the non-zero terms in the Cholesky triangle $\boldsymbol{L}$ and then apply the factorization $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^T$

The Cholesky triangle $\boldsymbol{L}$ inherits the non-zero pattern of the precision matrix $\boldsymbol{Q}$. We can show an example with the following graph (many more examples and applications appear on the GMRF book by Rue and Held (2005))



has set of nodes $\mathcal{V} = \{1, 2, 3, 4, 5\}$ and set of edges $\mathcal{E} = \{\{1, 2\}, \{1, 4\}, \{2, 3\}, \{3, 4\}, \{4, 5\}\}$. The graph provides all the information to compute the corresponding Cholesky $\boldsymbol{L}$ triangle

$$\boldsymbol{L} = \begin{pmatrix} L_{11} & & & & \\ L_{21} & L_{22} & & & \\ L_{31} & L_{32} & L_{33} & & \\ L_{41} & L_{42} & L_{43} & L_{44} & \\ L_{51} & L_{52} & L_{53} & L_{54} & L_{55} \end{pmatrix} \tag{2.39}$$

where the black cells denote non-zero terms while the red ones are unknown. Corollary 1 helps to fill in the information about these five red cells. By simply observing the structure above, the future sets are $\text{FU}(1, 3) = \{2, 4, 5\}$, $\text{FU}(2, 4) = \{3, 5\}$, $\text{FU}(1, 5) = \{2, 3, 4\}$, $\text{FU}(2, 5) = \{3, 4\}$ and $\text{FU}(3, 5) = \{4\}$. The only one that does not satisfy the Corollary is $\text{FU}(2, 4)$. Therefore, L_{24} term is the only non-zero element while all the others do not need to be computed as they are zero.

2.3.3 Band Matrices and Reordering

This section provides more insights into the numerical strategies used to speed up the computations with sparse precision matrices. We introduce two main concepts: *Band*

precision matrix structure and *Permutations* of the indexes. At the end of Section 2.3 we mentioned the precision matrix structure of an AR(p) time series model of order $p > 1$. In this example, the matrix $\boldsymbol{Q}$ has a $(2p+1)$-diagonal band of non-zero elements that makes the whole structure sparse. By construction its precision matrix $\boldsymbol{Q}$ is referred to a Band Matrix with *bandwidth* of degree p. By using Theorem 4 and Corollary 1 we can see that for $k > p$, the future set $\text{FU}(i, i+k)$ separates nodes i and $i+k$ so that the respective Cholesky triangle $\boldsymbol{L}$ has a lower triangular structure with the same lower bandwidth of order p. The bandwidth does not change after the factorization, and therefore we can state the following

Theorem 5 (**Bandwidth of L**)

Consider the precision matrix $\boldsymbol{Q} > 0$ being a $N \times N$ Band Matrix with bandwidth of degree p. Then the Cholesky triangle $\boldsymbol{L}$ from the Cholesky factorization $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^T$ has a (lower) bandwidth of degree p as well (a proof is given on Chapter 2 in Rue and Held (2005)).

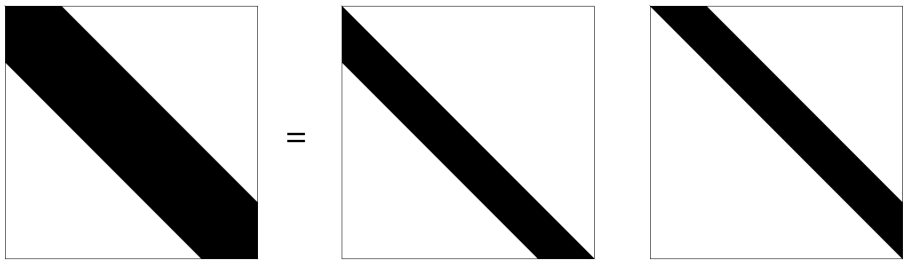


Figure 2.2: Graphical Band structures of the precision matrix $\boldsymbol{Q}$ and the Cholesky triangle $\boldsymbol{L}$ involved in the Cholesky factorization for an AR(p) process with bandwidth of degree p.

Here we use a modification of the Cholesky factorization applied on band matrices where we only use entries that satisfy $|i-j| \leq p$. For the autoregressive example, the cost reduces to $N(p^2+3p)$ when N is much higher than p so that it is linear in N compared to the usual $N^3/3$ operations. A loop from the diagonal up to each

column lets us take advantage of the band structure of $\boldsymbol{Q}$ which then extends to $\boldsymbol{L}$. Then we can run the loop on the diagonal within the p band. This band matrix structure has a relevant computational advantage as it is easy to handle and enlarges the sparsity structure of the matrix. Moreover, we can convert any sparse matrix into a band matrix, as reported in Theorem 5, by applying a reordering of the vertices. In practice, we construct an alternative representation of the same matrix by permuting the indexes while retaining its original structure and properties. Then we apply a permutation on these indexes and choose a new permuted matrix that most satisfies our desired bandwidth conditions. A summary of the process is given below

Definition 7 (**Reordered Band Matrix Q**)

Consider to select one of the $N!$ *possible permutations and define the corresponding permutation matrix* $\boldsymbol{P}$ *such that* $\boldsymbol{i}^{P^*} = \boldsymbol{P}\boldsymbol{i}$ *where* $\boldsymbol{i} = (1, \ldots, N)^T$ *is the new order of the vertices. Then we choose* $\boldsymbol{P}$*, if possible, such that*

$$\boldsymbol{Q}^{P^*} = \boldsymbol{P}\boldsymbol{Q}\boldsymbol{P}^T \tag{2.40}$$

is a Band Matrix with permutation P^* *of the indexes and a small bandwidth.*

The new permuted precision matrix $\boldsymbol{Q}^{P^*}$ will have a more sparse structure. As it appears from Definition 7, it is impossible to achieve an optimal permutation since there are too many combinations to explore when N is high. We can avoid such complexity by choosing a less optimal solution and solve $\boldsymbol{Q}\boldsymbol{\mu} = \boldsymbol{b}$ for a given ordering as follows

- Compute the new permuted problem $\boldsymbol{b}^{P^*} = \boldsymbol{P}\boldsymbol{b}$ where $\boldsymbol{b}$ is any new order of the vertices
- Solve $\boldsymbol{Q}^{P^*}\boldsymbol{\mu}^{P^*} = \boldsymbol{b}^{P^*}$
- Map the solution back with $\boldsymbol{\mu} = \boldsymbol{P}^T\boldsymbol{\mu}^{P^*}$

Basically, we apply the factorization in the new permutation setting, obtain the right-hand side of the permuted solution, and then compute the inverse permutation to get the first solution back. Other ideas can be used to maximize the speed up of this reordering scheme. For example, we can use the *Nested dissection reordering* where a small set of nodes is removed from the original graph structure producing multiple smaller, not connected sub-graphs of similar sizes. On this reordering task, two solvers are available in the `GMRFLib` C-library available in INLA: the Band Cholesky Factorization (BCF) with LAPACK-routines DPBTRF and DTBSV for the factorization and the forward/back-substitution, respectively, and the Gibbs-Poole-Stockmeyer algorithm for bandwidth reduction. More, we have the Multifrontal Supernodal Cholesky factorization (MSCF) implementation in the library TAUCS using the nested dissection reordering from the library METIS. Proper reorderings of preselected nodes and recursions on all the single graphs lead to significant reductions in the factorization cost. In the spatial setting this cost is generally of the order $\mathcal{O}(N^{3/2})$ with a fill-in cost of $\mathcal{O}(N\log(N))$. Existing software and built-in methods in the sparse linear algebra field are constantly updated to provide the best computational performance for sparse matrices. In R there are packages like `Matrix` that apply efficient embedding structures for sparse matrices while also choosing the best reordering in the background.

2.4 INLA Shell: Laplace Approximation

Latent Gaussian Models (LGMs) in Section 2.2.1 and Gaussian Markov Random Fields (GMRFs) in Section 2.3 are all we need to build empirically accurate Laplace approximations for the posterior marginals using INLA. The deterministic nature of the algorithm comes from the use of these approximations, which benefit from previous assumptions on the latent structure using GMRF priors and sparse matrices. The software does not rely on sampling from these GMRF structures but builds

reliable marginal posterior approximations for each unknown parameter of the model by using Laplace approximations. The Laplace mathematical technique sees its first appearance in 1774 by Pierre-Simon Laplace, who used it to approximate integrals of exponential functions of the form

$$\int_a^b \exp(mf(x))\,dx \tag{2.41}$$

where $f(x)$ is a generic twice differentiable function and a and b can assume any value. The method belongs to the field of asymptotic analysis, where we seek approximate solutions to parametric settings where the parameter tends to an asymptotic limit. The purpose is to analytically compute the integral of well-behaved unimodal functions $f(x)$ with strictly positive second-order derivative at the mode. Mostly, the integrand turns out to be pretty peaked as the dimension m approaches infinity so that a Taylor expansion can properly approximate it. In our statistical framework, we see that expansions up to second-order are sufficient to approximate the integrand of interest, which is close to a Gaussian density in most cases. The use of a Taylor expansion up to order two has indeed some advantages

- the polynomial result behaves as a quadratic function, which recalls a Gaussian distribution. By integrating it out, we would end up with just the normalization constant of the Gaussian density;

- the expansion represents a natural idea as the main bulk of the function concentrates around the mode. This smoothly decreases as we approach its tails, assuming no unusual pattern arises as we move far from the modal point. One can see that the tail behavior of Skew Normal family densities shows limiting Gaussian-like patterns as well (see Chapter 4);

- one can also argue that higher-order terms in the expansion would lead to more accurate approximations, with the error quickly approaching zero as m grows.

However, we would have more terms and cross-terms to evaluate therefore having more costly computations. Laplace Approximations in INLA involve expansions up to order two, which are perfectly accurate for posterior distributions that are close to a Gaussian one most of the times. INLA can even push more when this does not hold by stepping into a third-order expansion approximation (see Section 2.5.3).

The resulting approximations get more accurate as the dimension m increases. There are also some regularity conditions strictly related to the applied Taylor expansions, which affect the approximation method, but we will not discuss them here. We can now move on to some mathematical details using a generic example. The idea revolves around computing the integral of a function $f(x)$ by considering $g(x) = \log(f(x))$ and approximating the equivalent integral of $\exp(g(x))$. Computing the Laplace approximation for any integral $\int_a^b \exp(mf(x))\,dx$ is straightforward as soon as we apply a second-order Taylor expansion on $\exp(mf(x))$ around its mode x^*

$$\begin{aligned}\exp(mg(x)) &\approx \exp\Big(mg(x^*) + mg^1(x^*)(x-x^*) + \frac{m}{2}g^2(x^*)(x-x^*)^2\Big)\\ &\approx \exp(mg(x^*))\exp\Big(\frac{m}{2}g^2(x^*)(x-x^*)^2\Big)\end{aligned} \tag{2.42}$$

where $g^1(\cdot)$ and $g^2(\cdot)$ represent the first and second derivative of $g(x)$ respectively. The first derivative $g^1(\cdot)$ in the expansion vanishes since $g^1(x^*) = 0$. By integrating the expression in (2.42), we recognize the pattern of a Gaussian density with mean x^* and standard deviation $\sigma = \frac{1}{\sqrt{-mg^2(x^*)}}$. Then the resulting integral is

$$\int \exp(mg(x))\,dx \approx \exp(mg(x^*))\sqrt{2\pi\sigma^2} \tag{2.43}$$

which is proportional to a Gaussian normalizing constant. As $m \to \infty$, the approximation gets more accurate, and the error drops to zero. This resulting error is relative

and mostly comes from approximating the normalizing constant of the function. We can describe its relative nature as

$$\frac{\tilde{\mathcal{I}}(m)}{\mathcal{I}(m)} = 1 + \mathcal{O}(m^{-1}) \tag{2.44}$$

where $\tilde{\mathcal{I}}(m)$ is the approximated integral and $\mathcal{I}(m)$ represents its true value in terms of m observations, which are given. The relative error is of order $\mathcal{O}(m^{-1})$ which is $\sqrt{m}$ faster than simulation approaches like Monte Carlo (MC) or Markov Chain Monte Carlo (MCMC) where, instead, the error is additive

$$\tilde{\mathcal{I}}(m) = \mathcal{I}(m) + \mathcal{O}(m^{-\frac{1}{2}}) \tag{2.45}$$

In general, the m notation used for the Monte Carlo sampling approaches would refer to the samples used to construct the approximation and this can grow exponentially large if more accuracy is required. From the first application of Laplace Approximation in Tierney and Kadane (1986), the authors compute more terms in the expansion by reaching error terms of the order $\mathcal{O}(m^{-3})$. While this is more demanding and accurate, it is not needed in our context since we do not observe any relevant additional gain. INLA approximation strategies already ensure good trade-off between accuracy and computational cost in order to achieve ideal results.

2.5 Bayesian Computing with INLA

The R-INLA software is a standalone R program that allows the user to compute deterministic posterior marginal approximations for all parameters of a Latent Gaussian Model. We can speed up computations by introducing Gaussian Markov Random Fields with sparse precision matrices to model the latent field. Fast algorithms from the sparse linear algebra theory lead to the desired performance. Marginal posterior densities are approximated by Gaussian or Laplace Approximations depending on the

chosen strategy and model complexity. The resulting approximations are empirically correct, and we can write them as

$$\tilde{\pi}(\theta_j|\boldsymbol{y}) = \int \tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\, d\boldsymbol{\theta}_{-j}, \quad j = 1, \ldots, |\boldsymbol{\theta}| \tag{2.46}$$

$$\tilde{\pi}(x_i|\boldsymbol{y}) = \int \tilde{\pi}(x_i|\boldsymbol{\theta}, \boldsymbol{y})\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\, d\boldsymbol{\theta}, \quad i = 1, \ldots |\boldsymbol{x}| \tag{2.47}$$

where we assume the dimension $|\boldsymbol{\theta}|$ to be less than 20 for a low computational burden while $|\boldsymbol{x}|$ is equal to the entire model dimension N. Any Bayesian inference analysis mainly requires computing the expressions above. First, we need to efficiently evaluate each single integral in the approximations $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ and $\tilde{\pi}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ as their cost grows with increasing dimensions. As hinted by its acronym, INLA exploits nested Laplace Approximations and numerical integration to accomplish this complex task. The first approximation is applied on the posterior marginal of the hyperparameters $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ in (2.46) which does not generally require heavy computations since the hyperparameter dimension is assumed to be small. The second approximation is the Laplace Approximation on each conditional distribution $\tilde{\pi}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ in (2.47) which is univariate. Finally, we combine both approximated results to get the posterior marginals of the latent field. We can summarise the whole approach into three steps

- **STEP 1:** Compute the Laplace Approximation $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ through a grid exploration scheme of its multi-dimensional density. We avoid representing this density parametrically since it would be too costly. We employ explorative strategies to sufficiently represent the multivariate result by computing relevant mass probability points of the hyperparameter space (details are outlined in Section 2.5.1). Each univariate posterior marginal $\tilde{\pi}(\theta_j|\boldsymbol{y})$ for each j is obtained by interpolation.

- **STEP 2:** Compute the second Laplace Approximation $\tilde{\pi}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ for each i by us-

ing the pre-defined explorative scheme employed on Step 1 for $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. This part can be quite tricky since it can be potentially slow. Three different strategies are available in INLA to avoid these possible heavy computations based on the problem complexity. This is outlined in Section 2.5.2.

- **STEP 3:** Combine Step 1 and Step 2 to finally get the univariate posterior marginals $\tilde{\pi}(x_i|\boldsymbol{y})$ for each i through numerical integration.

One may think to apply the Laplace Approximation on both $\boldsymbol{x}$ and $\boldsymbol{\theta}$ jointly instead of relying on multiple, nested applications of the same technique. This idea is hardly feasible since the resulting multivariate density would be far from a Gaussian and involve many cross-correlated terms. For nearly Gaussian models with non-gaussian data, the Laplace approximation always provides the most accurate results. If the model is exactly Gaussian, we can instead use a Gaussian Approximation that grants the most accurate and fast results.

2.5.1 Applying Gaussian and Laplace Approximation

To get the marginal approximations (2.46) we first need to compute an approximation for $\pi(\boldsymbol{\theta}|\boldsymbol{y})$. Instead of trying to solve the integral, it is much better to apply conditional relations and write

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \tag{2.48}$$

The numerator in (2.48) involves the term $\pi(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{\theta})$ which can deviate considerably from a Gaussian density while the denominator is both unknown and not Gaussian distributed. This suggests that we cannot directly approximate the overall density, but instead, we need to rely on other methods. Hence we apply a joint approximation in $\boldsymbol{\theta}$-dimensional space on the denominator term so that we have

$$\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y}) \propto \frac{\pi(\boldsymbol{x}^*, \boldsymbol{\theta}|\boldsymbol{y})}{\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}^*|\boldsymbol{\theta}, \boldsymbol{y})}\bigg|_{\boldsymbol{x}^*=\boldsymbol{\mu}(\boldsymbol{\theta})} \tag{2.49}$$

where $\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}^*|\boldsymbol{\theta}, \boldsymbol{y})$ is the Gaussian Approximation obtained by matching the mode and curvature at the mode of the full density $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ after an iterative process (see Appendix A for details). Then we obtain the Laplace approximation for (2.49) by evaluating the ratio at the denominator mean $\boldsymbol{\mu}(\boldsymbol{\theta})$. Here we point out two observations:

- the approximation employs the mode instead of other summaries like mean or median to avoid mismatches due to possible extreme observations
- the computed modal configurations strictly depend on the hyperparameter set $\boldsymbol{\theta}$ after an evaluation strategy is applied on $\pi(\boldsymbol{\theta}|\boldsymbol{y})$

The Gaussian Approximation appearing in the denominator of (2.49) simplifies the Laplace Approximation problem. This is clear if we take a look at the nature of $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$. If we do not consider data $\boldsymbol{y}$ then we obtain the density $\pi(\boldsymbol{x}|\boldsymbol{\theta})$ which is Gaussian by GMRF construction. Therefore everything changes as soon as we plug data into $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$. Its resulting behaviour would be Gaussian if observations $\boldsymbol{y}$ are Gaussian and nearly Gaussian if they are not. We can also ease the approximation problem by using variance-stabilizing transformations for $\boldsymbol{\theta}$ such as log or logit functions for example. These transformations contribute to reduce skewness and get more Gaussian-like posterior densities with lighter tails. The way the Gaussian Approximation is applied to the full conditional density $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ of a Latent Gaussian Model is shown below

$$\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}) \propto \exp\Big(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}(\boldsymbol{\theta})\boldsymbol{x} + \sum_i^n \log(\pi(y_i|x_i,\boldsymbol{\theta}))\Big)$$
$$\approx \exp\Big(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}(\boldsymbol{\theta}))^T(\boldsymbol{Q}(\boldsymbol{\theta}) + \mathrm{diag}(\boldsymbol{c}(\boldsymbol{\theta})))(\boldsymbol{x}-\boldsymbol{\mu}(\boldsymbol{\theta}))\Big) \equiv \tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}) \tag{2.50}$$

where $\boldsymbol{\mu}(\boldsymbol{\theta})$ contains the location of the mode while $\boldsymbol{c}(\boldsymbol{\theta})$ contains the negative second derivative of the log-likelihood at the mode. The distribution $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ depends on two quantities: the Gaussian structure of the latent field $\boldsymbol{x}$ and some log-likelihood contribution from the data $\boldsymbol{y}$. By definition of the Gaussian Approximation, if the likelihood contribution term is Gaussian, we end up with a quadratic expression in the expansion, and the approximation is exact. In a general non-Gaussian case, the expansion error would propagate to the third-order or higher term representing skewness or higher-order moments. However, the resulting approximation will not be much far from the true probabilistic outcome since the error would be negligible. The Gaussian prior on latent field $\boldsymbol{x}$ is a strong assumption for a Latent Gaussian Model as it ensures accurate posterior approximations at negligible costs (see Chapter 4 for more details). Indeed, the prior considerably forces a Gaussian behavior onto $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ even before the data are considered into the likelihood contribution part. This term would mainly affect the mean and the marginal variance of $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ while slightly touching the skewness. If we condition the expression (2.50) on data $\boldsymbol{y}$, we only get an additional term on the diagonal of the precision matrix $\boldsymbol{Q}$, given by the term $\mathrm{diag}(c_i)$, which does not affect much the approximation and computations. The sparsity structure, or similarly the underlying graph, remains unscathed, making clear that the computational cost of considering data or not is the same. As a result, the data contribution does not affect the sparsity structure of $\boldsymbol{Q}$ or any existing dependency structure coming from the GMRF prior of the latent field. Moreover, the

mixed product terms, not appearing in the likelihood contribution term, only come from the GMRF prior, which improves the Gaussian Approximation accuracy.

2.5.2 Exploring the joint hyperparameter density

In this section we explain how INLA computes Laplace Approximations for the joint posterior density $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and its marginals. From the marginal expressions (2.46) and (2.47) we see that $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is the first density we need to approximate to obtain all the target posterior marginals of a Latent Gaussian Model. Since this joint density is assumed not to belong to a high dimensional space, we can approach its approximation from a non-parametric perspective avoiding possible heavy computations derived from a full parametric representation. Therefore we explore the approximation $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ by evaluating a set of points $\{\boldsymbol{\theta}_k, k = 1, \ldots, K\}$ that can properly represent its parameters range in the probability space. The number of points K determines the level of accuracy we assume to recover the initial density and is generally small in most cases. We can compute these points through a grid exploration scheme for the entire dimension of $\boldsymbol{\theta}$ by following a set of conditions. The whole scheme can be summarised as follows

1. compute the approximation $\log\{\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\}$ and locate its modal configuration $\boldsymbol{\theta}^*$ by applying a Quasi-Newton optimization strategy;

2. evaluate the Hessian matrix $\boldsymbol{H} > 0$ at the mode $\boldsymbol{\theta}^*$ using finite difference methods. Then define a new parameterization for the hyperparameters $\boldsymbol{\theta}$ as follows

   Definition 8 (**z-parameterization**)
   Define $\boldsymbol{\Sigma} = \boldsymbol{H}^{-1}$ *as the covariance matrix of* $\boldsymbol{\theta}$ *and* $\boldsymbol{\Sigma} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^T$ *the eigendecomposition of* $\boldsymbol{\Sigma}$. *The new z-parameterization is*

$$\boldsymbol{\theta}(\boldsymbol{z}) = \boldsymbol{\theta}^* + \boldsymbol{V}\boldsymbol{\Lambda}^{\frac{1}{2}}\boldsymbol{z} \tag{2.51}$$

where $\boldsymbol{z}$ are standardized and mutually orthogonal variables.

3. Compute the integration points $\{\boldsymbol{\theta}_k, k = 1, \ldots, K\}$ until the difference between the new modal evaluation $\log\{\tilde{\pi}(\boldsymbol{\theta}(\mathbf{0})|\boldsymbol{y})\}$ and the respective point $\log\{\tilde{\pi}(\boldsymbol{\theta_k}(\boldsymbol{z})|\boldsymbol{y})\}$ is below a certain threshold (a value of 6 is large enough to cover an acceptable range);

4. Use the points $\boldsymbol{\theta}_k$ to both evaluate the density $\log\{\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\}$ and then compute the respective marginals in (2.46) by interpolation to avoid computational slowdowns.

These steps define the aforementioned *grid exploration* phase of the joint hyperparameter density. This strategy ensures accurate results but can be computationally demanding since the cost grows exponentially with the dimension of $\boldsymbol{\theta}$. When this happens, we can turn to other two explorative strategies

- ***EB*:** *Empirical Bayes.* If the variability amongst the hyperparameters $\boldsymbol{\theta}$ does not affect the posterior information of the latent field $\boldsymbol{x}$ and $|\boldsymbol{\theta}|$ is large, then we can plug in the mode of $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ and do the integration. This chice avoids heavy computations due to the number of hyperparameters in the model while ensuring good level of accuracy in the results;

- ***CCD*:** *Central Composite Design.* This is the explorative scheme used by default in INLA when $|\boldsymbol{\theta}| > 2$. For $|\boldsymbol{\theta}| \leq 2$ the grid exploration strategy is employed by default. This scheme considerably reduces the computational cost by setting the integration problem into a design problem which exploits a response surface approach. Such method allows to compute a considerably lower amount of

integration points $\boldsymbol{\theta}_k$ that can still recover the target approximation (see Rue et al. (2009) and Gómez Rubio (2020) for more details on these strategies).

The z-parameterization provides a useful transposition of the $\boldsymbol{\theta}$-space into a Gaussian one therefore simplifying the exploration of $\log\{\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\}$ by detecting its highest bulk of probability around the mode. We define a specific threshold around the mode $\boldsymbol{\theta}^*$ and then compute a set of points in the new standardized Gaussian $\boldsymbol{z}$-space with $\boldsymbol{H}^{-1} = \boldsymbol{I}$ which well represent the entire density. The initial integration points $\{\boldsymbol{\theta}_k, k = 1, \ldots, K\}$ are recovered by reverting back to the $\boldsymbol{\theta}$-space using the relation in (2.51). Each posterior marginal approximation $\tilde{\pi}(\theta_j|\boldsymbol{y})$ in formula (2.46) is then computed by interpolation using the integration points $\boldsymbol{\theta}_k$. More precisely, we can fit a spline using the respective coordinates $\{\theta_{k(j)}, \log(\tilde{\pi}(\theta_{k(j)}|\boldsymbol{y}))\}$ with relative to the points $\theta_{k(j)}$ of the j^{th} marginal and then normalize the density (see Martino and Riebler (2019) for a detailed example). Details on how to build an efficient interpolation approach can be found in Martins et al. (2013). By default INLA exploits a numerical integration free algorithm for obtaining the posterior marginals of the hyperparameters. This method takes advantage of the pre-computed integration points on the approximation $\tilde{\pi}(\theta_{k(j)}|\boldsymbol{y})$ and speed up the computations by also correcting for skewness when $|\boldsymbol{\theta}|$ is high. INLA also allows the user to implement his own grid exploration scheme for the hyperparameter space but this requires advanced awareness from the user perspective.

2.5.3 Approximating the latent field marginals

Since we have now obtained the integration points $\boldsymbol{\theta}_k$ from the grid exploration scheme in Section 2.5.2, we can finally compute the remaining posterior approximations of the latent field in (2.47). These marginal approximations require a nested combination of Laplace Approximations of both the marginals $\pi(\theta_j|\boldsymbol{y})$ for $j = 1, \ldots, |\boldsymbol{\theta}|$ and the full conditionals $\pi(x_i|\boldsymbol{y}, \boldsymbol{\theta})$ for $i = 1, \ldots, |\boldsymbol{x}|$. For the Laplace Approximation of these

full conditional densities, we can follow a similar strategy used for (2.49). However, this hides some tricky issues since it involves multiple factorizations of the precision matrix structure $\boldsymbol{Q}(\boldsymbol{\theta})$, which depends on the hyperparameter set. A blind brute force approach can lead to slow computations since the precision matrix dimensionality can grow large, in the order of 10^3, 10^4, or more. Indeed its dimension depends on both data and latent parameter dimensions. For example, if we assume n observations and p parameters, then the dimension would be $n + p$ with both n and p being possibly really large. In this final section, we outline the three main available strategies in INLA for efficiently tackling this problem:

- ***Gaussian Approximation (GA).*** By exploiting the information from the joint Gaussian approximation $\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ we can compute each marginal approximation $\tilde{\pi}(x_i|\boldsymbol{y}, \boldsymbol{\theta})$ for each i through a Gaussian distribution with marginal mean $\mu_i(\boldsymbol{\theta})$ and marginal variance $\sigma_i^2(\boldsymbol{\theta})$. This task requires the computations of the respective marginal means and variances. The Gaussian Approximation is recognized to be the most accurate and fast among the available strategies except it does not correct for location and skewness.

- ***Laplace Approximation (LA).*** Similarly to the approximation in (2.49), we can again apply the Laplace approximation to $\pi(x_i|\boldsymbol{y}, \boldsymbol{\theta})$ to get

$$\tilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{y}, \boldsymbol{\theta}) \propto \frac{\pi(\boldsymbol{x}^*, \boldsymbol{\theta}|\boldsymbol{y})}{\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}^*_{-i}|x_i, \boldsymbol{y}, \boldsymbol{\theta})}\Big|_{\boldsymbol{x}^*_{-i}=\boldsymbol{\mu}_{-i}(x_i,\boldsymbol{\theta})} \tag{2.52}$$

  with $\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}^*_{-i}|x_i, \boldsymbol{\theta}, \boldsymbol{y})$ being the Gaussian Approximation with modal configuration $\boldsymbol{\mu}_{-i}(x_i, \boldsymbol{\theta})$. The Gaussian Approximation on the lower-dimensional density $\pi(\boldsymbol{x}_{-i}|x_i, \boldsymbol{y}, \boldsymbol{\theta})$ provides accurate results as they are well behaved and nearly Gaussian. Although the Laplace Approximation results are more accurate in more skewed settings than its Gaussian counterpart, it still suffers a not negligible slowdown in the computations as it requires N factorizations of

$(N-1)\times(N-1)$ matrices. While it does resolve the computational issue related to the factorizations of $\boldsymbol{Q}(\boldsymbol{\theta})$ for each integration point θ_k, it still suffers from the many optimization steps to locate the mode of each density $\pi(\boldsymbol{x}_{-i}|x_i, \boldsymbol{y}, \boldsymbol{\theta})$. We can save computations by computing an approximated mode using the conditional mean in terms of $\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ and then only consider the latent terms that actively provide a contribution to x_i. Some pre-defined criteria lead to a series of points used to approximate the outcomes with a Gaussian density and a cubic spline (details on Rue et al. (2009)). The resulting approximation would be of the form

$$\tilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{y}, \boldsymbol{\theta}) \propto N(\mu_i(\boldsymbol{\theta}), \sigma_i(\boldsymbol{\theta})) \exp(S(x_i)) \tag{2.53}$$

where $S(x_i)$ is a cubic spline involving polynomials of third order degree. The spline applies an interpolation of these points from the marginal latent variable to the log density difference between the Laplace Approximation and respective Gaussian Approximation. While this strategy is more expensive, this point by point spline interpolation corrects the Gaussian Approximation when this one is far from being accurate.

- ***Simplified Laplace Approximation (SLA).*** This strategy reduces the computational burden of the Laplace approximation strategy by keeping the accuracy as high as possible. The idea is to apply a Taylor expansion up to third order of $\tilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{y}, \boldsymbol{\theta})$ around the point $x_i = \mu_i(\boldsymbol{\theta})$ and use the resulting components of the expansion to correct $\tilde{\pi}_{\mathrm{G}}(x_i|\boldsymbol{y}, \boldsymbol{\theta})$ for location and skewness. The first and second order term exactly give $\tilde{\pi}_{\mathrm{G}}(x_i|\boldsymbol{y}, \boldsymbol{\theta})$ while the third order term provides a correction for skewness. The resulting approximation has the form

$$\tilde{\pi}_{\mathrm{SLA}}(x_i|\boldsymbol{y}, \boldsymbol{\theta}) \approx \exp\Big(-\frac{1}{2}x_i^2 + b_i(\boldsymbol{\theta})x_i + \frac{1}{6}c_i(\boldsymbol{\theta})x_i^3)\Big) \tag{2.54}$$

where $(b_i(\boldsymbol{\theta}), c_i(\boldsymbol{\theta}))$ induce more marginal corrections to mean and skewness. This expression does not define a proper density function since the third-order term is unbounded, but it still can be used to fit a Skew-Normal density (see Azzalini and Capitanio (2018)). In general terms, we define $S \sim \text{SN}(\xi, \omega, \alpha)$ with probability density function

$$f(s; \xi, \omega, \alpha) = \frac{2}{\omega}\phi\Big(\frac{s-\xi}{\omega}\Big)\Phi\Big(\alpha\frac{s-\xi}{\omega}\Big) \tag{2.55}$$

with α being the skewness parameter to account for skewness in the distribution. If $\alpha = 0$ then the distribution degenerates into a $\text{N}(\xi, \omega^2)$. In Chapter 4 we show how to construct the approximation (2.54) by matching the first and third-order terms $b_i(\boldsymbol{\theta})$ and $c_i(\boldsymbol{\theta})$ to the Skew-Normal moments and then propose a possible extension using an Extended Skew Normal distribution.

The approximation $\tilde{\pi}_{\text{SLA}}(x_i|\boldsymbol{y}, \boldsymbol{\theta})$ represents the default strategy in INLA since it offers the best deal between speed and accuracy. The other two strategies may provide better results if the posteriors are closely Gaussian or more accuracy is required. The computational complexity of these strategies is mainly related to multiple factorizations of the precision matrix of the latent field and linear system solutions while also taking into account the dimension of $\boldsymbol{\theta}$. In a spatial setting, the cost for computing each marginal $\tilde{\pi}_{\text{SLA}}(x_i|\boldsymbol{y}, \boldsymbol{\theta})$ would be $\mathcal{O}(N \log(N))$. Therefore the total cost for computing all N marginals would be $\mathcal{O}(N^2 \log(N))$ for each configuration point in the $\boldsymbol{\theta}$ space. Now that the full conditional approximations are available, we can compute the posterior marginals in (2.47) via numerical integration as

$$\tilde{\pi}(x_i|\boldsymbol{y}) \approx \sum_{k=1}^{K} \tilde{\pi}(x_i|\boldsymbol{y}, \boldsymbol{\theta}_k)\tilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})\Delta_k \tag{2.56}$$

where $\tilde{\pi}(x_i|\boldsymbol{y}, \boldsymbol{\theta}_k)$ is obtained though one of the previous strategy (GA, LA or SLA) and $\tilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})$ is computed with one of the explorative scheme described in Section

2.5.2. In a spatial setting, the cost for computing all posterior marginal above would be $\exp(|\boldsymbol{\theta}|) \times \mathcal{O}(N^2 \log(N))$. Depending on the likelihood nature, the equation follows a mixture of Gaussian or Skew-Normal densities with integration weights $\tilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})\Delta_k$. A similar structure will come back in Chapter 3 when discussing joint posterior density approximations for Latent Gaussian Models. The components Δ_k are area weights from the grid used for exploring $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$. These are equal to one since the grid structure is often constructed in an equidistant way to ease the density exploration but they can assume different values if the grid is more irregular. The entire source of the approximation error in (2.56) comes from the Laplace Approximations and the grid exploration scheme. If $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}_k)$ is Gaussian then each $\pi(x_i|\boldsymbol{y})$ can be computed exactly for each integration point $\boldsymbol{\theta}_k$ with the only source of error coming from the grid exploration phase.

Chapter 3

Joint Posterior Adjusted Inference for Latent Gaussian Models

The computational aspect of solving complex Bayesian problems plays a significant role in inferential analysis. Most strategies, such as Markov Chain Monte Carlo (MCMC) methods, rely on sampling from proposal distributions to retrieve the underlying truth but can be heavily slow in more complex settings. The Integrated Nested Laplace Approximation (INLA, Rue et al. (2009)) approach, whose methodology is outlined in Chapter 2, can achieve the same or better Monte Carlo accuracy by constructing marginal deterministic approximations for the posterior marginals of a Latent Gaussian Model. We can extend this methodology to a more accurate joint inference analysis by defining a new class of joint approximations. Section 3.1 introduces the class of Skew Gaussian Copula (SGC) joint approximations to the latent field components, where we combine a Gaussian Copula structure with marginal transformations that add location and skewness adjustments. Section 3.2 exploits a mixture representation of the new class and its moments to compute deterministic posterior approximations for linear combinations in a subset of the latent field. These approximations are fast to compute as they follow exact parametric assumptions. In Section 3.3 the same mixture representation of Skew Gaussian Copula (SGC) joint densities contribute to achieving an approximation for the full joint posterior density of a Latent Gaussian Model by employing an exact Monte Carlo sampling approach on the hyperparameter space. All these new approximations are then tested and compared with the accurate MCMC results obtained in JAGS (Plummer et al. (2003))

by using highly skewed simulated examples from Poisson and Binomial hierarchical models. We summarise the main findings of the simulations in Section 3.4. This chapter is based on the respective submitted paper in Chiuchiolo et al. (2021).

3.1 Class of Skew Gaussian Copula approximations

Marginal posterior inference for a Latent Gaussian Model (LGM) is easy to obtain in INLA using accurate approximations. On the contrary, joint inference for the same model is less straightforward as the density is unknown, and an accurate approximation is hard to achieve. This section introduces how we can approach a joint approximation by first extending the Gaussian Approximation to a more accurate version. We recall that the joint posterior density (2.4) is derived from a hierarchical mathematical formulation of the multivariate latent field $\boldsymbol{x}$ with dimension N and a set of hyperparameters $\boldsymbol{\theta}$ with dimension p. The posterior marginals for the latent field $\boldsymbol{x}$ are obtained as follows

$$\begin{aligned}\pi(\boldsymbol{x}|\boldsymbol{y}) &= \int \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})\, d\boldsymbol{\theta} \\ &= \int \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\pi(\boldsymbol{\theta}|\boldsymbol{y})\, d\boldsymbol{\theta} \end{aligned} \tag{3.1}$$

We now know that we can compute approximations of these marginals by using a nested Laplace scheme and then integrating out the uncertainty coming from the hyperparameter set $\boldsymbol{\theta}$ (as described in Section 2.5.3). Three are the main approximation strategies we can use in INLA: the Gaussian Approximation (GA), the Laplace approximation (LA) and the Simplified Laplace approximation (SLA). The Gaussian Approximation results to be the most attractive when the likelihood contribution is Gaussian or close to that shape. This thesis chapter will greatly focus on the joint Gaussian Approximation onto the full conditional density of the latent field $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$

which we can use to construct accurate joint posterior approximation for the joint density in (2.4) with marginal skewness adjustments. Appendix A provides details on the Gaussian Approximation, which constructs a multivariate Gaussian density with mean and correlation structure inherited by the GMRF prior information assumed on the latent field (see Chapter 2 for a GMRF definition). However, the joint Gaussian Approximation $\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ lacks accuracy in more extreme settings where Gaussian assumptions do not properly hold. Here we can introduce a new class of joint approximations that encodes corrections for location and skewness while retaining Gaussian Approximation properties in its multivariate definition. This class can handle more extreme outcomes when the deviation from a Gaussian gets larger.

3.1.1 General Formulation

While considering the Gaussian Approximation on the latent field $\boldsymbol{x}$, we define a set of new random variables $\tilde{\boldsymbol{x}} = \boldsymbol{h}(\boldsymbol{x}) = (h_1(x_1), \ldots, h_N(x_N))$ with well-defined transformations $\boldsymbol{h}(\boldsymbol{x})$ and envelope the entire joint object into a Gaussian Copula structure. A copula represents a mathematical object which links a joint multivariate cumulative distribution function to its univariate marginals, which are uniform by construction. This mathematical formulation helps to properly model the dependency relation amongst a sequence of random variables. More precisely, Sklar (1959) theorem (see Nelsen (1999)) states that any joint cumulative density function $H(\cdot)$ having marginals $G_1(\cdot), \ldots, G_N(\cdot)$ defines a copula C on generic random variables $X_1, \ldots, X_N$ such that

$$H(X_1, X_2, \ldots, X_N) = C(G_1(X_1), G_2(X_2) \ldots, G_N(X_N))$$

where C is unique as soon as all marginals $\{G_i\}_{i=1}^N$ are continuous. If we define new random variables $U_1, \ldots, U_N$ such that each $U_i = G(X_i)$, $\forall i$ then we can write the copula as

$$C(U_1, U_2, \ldots, U_N) = H(G_1^{-1}(U_1), G_2^{-1}(U_2) \ldots, G_N^{-1}(U_N))$$

by applying the *Probability Integral Transform* theorem (PIT) to each random variable U_i assuming each marginal inverse cumulative distribution function $G_i^{-1}(\cdot)$ exists. This method allows to flexibly model a joint distribution with any well defined and continuous marginal distributions $\{G_i\}_{i=1}^N$ and copula C and generate samples from random variables having these marginals (see Section 3.1.4 on a Gaussian Copula construction for our case). In our framework, the use of a Gaussian Copula entirely matches our needs of constructing an improved version of the Gaussian Approximation. As a result, we can write the new class of approximations as follows

$$\begin{aligned}\tilde{\pi}_{\text{SGC}}(\tilde{\boldsymbol{x}}|\boldsymbol{\theta}, \boldsymbol{y}) &\propto \tilde{\pi}_{\text{G}}(\boldsymbol{h}^{-1}(\tilde{\boldsymbol{x}})|\boldsymbol{\theta}, \boldsymbol{y})|\boldsymbol{J}_{\tilde{\boldsymbol{x}}}| \\ &\propto |\boldsymbol{J}_{\tilde{\boldsymbol{x}}}| \exp(-\frac{1}{2}[\boldsymbol{h}^{-1}(\tilde{\boldsymbol{x}}) - \boldsymbol{\mu}^*(\boldsymbol{\theta})]^T \boldsymbol{Q}^*(\boldsymbol{\theta})[\boldsymbol{h}^{-1}(\tilde{\boldsymbol{x}}) - \boldsymbol{\mu}^*(\boldsymbol{\theta})] \qquad (3.2)\end{aligned}$$

where $(\boldsymbol{\mu}^*(\boldsymbol{\theta}), \boldsymbol{Q^*(\theta)})$ are posterior summaries from the respective Gaussian approximation and $\boldsymbol{J}_{\tilde{\boldsymbol{x}}}$ is the Jacobian result from applying the transformation $\boldsymbol{h(\cdot)}$. We know that $\boldsymbol{\mu}^*(\boldsymbol{\theta})$ comes from matching the mode of $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ and $\boldsymbol{Q}^*(\boldsymbol{\theta}) = \boldsymbol{Q}(\boldsymbol{\theta}) + \text{diag}[\boldsymbol{c}(\boldsymbol{\theta})]$ where $\boldsymbol{Q}(\boldsymbol{\theta})$ is the original precision matrix of the latent field while $\boldsymbol{c}(\boldsymbol{\theta})$ contains the negative second derivative of the log-likelihood. Depending on the choice of the set of functions $\boldsymbol{h(\cdot)}$, the multivariate object in (3.2) can assume different shapes. The resulting joint density inherits the Gaussian Approximation structure while also adding corrections for skewness to its marginals. Such copula construction retains the dependency structure amongst the latent field terms and then constructs more skewed marginals depending on the chosen transformation. Based on this construction, we name this class of joint approximations *Skew Gaussian copula* (SGC). Not only we can retrieve joint Gaussian Approximations from this class, but we can freely apply

a transformation $\boldsymbol{h}(\cdot)$ that enables more accurate marginals in terms of location and skewness adjustments. These properties improve the existing limits of the Gaussian Approximation towards more extreme settings without affecting the latent field structure of the model. The choice of the transformation becomes a key point for defining an accurate class of joint densities with desirable properties for the approximation. There are indeed settings where Gaussian modeling assumptions can be poor due to lack of observations and information. For example, when considering count data with unbalanced low counts or a low number of successes in a binomial experiment. These observed datasets can lead to heavily skewed outcomes that must be properly accounted when doing inference.

3.1.2 Gaussian Approximation and Poisson Likelihood example

As an example, we consider Poisson observations $y_1, \ldots, y_n$ with means $\lambda_1, \ldots, \lambda_n$, a single covariate $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_n)$ and linear predictors $\eta_i = \log \lambda_i = \beta_0 + \beta_1 \xi_i$, $\forall i$ with (β_0, β_1) being the coefficients for the intercept and covariate respectively. Based on INLA methodology, the latent field is $\boldsymbol{x} = (\eta_1, \ldots, \eta_n, \beta_0, \beta_1)$ with $\boldsymbol{x} \sim N(\boldsymbol{0}, \boldsymbol{Q})$ where $\boldsymbol{Q}$ is a symmetric precision matrix with dimension $(n+2) \times (n+2)$. In this case we have $\pi(\boldsymbol{\theta}|\boldsymbol{y}) \propto 1$ since no hyperparameter is assumed in the model structure and

$$\begin{aligned} \pi(\boldsymbol{x}|\boldsymbol{y}) &\propto \pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x}) \\ &\propto \exp\Big(-\frac{1}{2}\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} + \sum_{i=1}^{n} [y_i \eta_i - \exp(\eta_i)]\Big) \end{aligned} \tag{3.3}$$

which represents the joint posterior density of the model as it is. Accordingly, the precision matrix $\boldsymbol{Q}$ is

$$\boldsymbol{Q} = \begin{pmatrix} \tau_\epsilon \boldsymbol{I} & \tau_\epsilon \boldsymbol{I}\boldsymbol{\xi} & \tau_\epsilon \boldsymbol{I}\mathbb{1} \\ & \tau_{\beta_0} + \tau_\epsilon \mathbb{1}^T\mathbb{1} & \tau_\epsilon \boldsymbol{\xi}^T\mathbb{1} \\ & & \tau_{\beta_1} + \tau_\epsilon \boldsymbol{\xi}^T\boldsymbol{\xi} \end{pmatrix} \tag{3.4}$$

where $\boldsymbol{I}$ is a $n \times n$ identity matrix, $\mathbb{1} = (1, 1, \ldots, 1)^T$ a $(n+2)$-dimensional unit vector and $(\tau_{\beta_0}, \tau_{\beta_1})$ are the precisions of the fixed parameters (β_0, β_1). The precision τ_ϵ that appears in the precision matrix structure (3.4) is related to a tiny Gaussian noise ϵ added to each linear predictor η_i to avoid singularity issues for $\boldsymbol{Q}^{-1}$ (more insights in Rue et al. (2017) and Section 2.2.3). By applying the approach outlined in Appendix A, we construct a Gaussian Approximation to the joint density in (3.3) which is denoted by $\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{y})$. This approximation defines a multivariate Gaussian density $N(\boldsymbol{x}^*, \boldsymbol{Q}^*)$ where $\boldsymbol{x}^* = (x_1^*, \ldots, x_n^*, x_{n+1}^*, x_{n+2}^*)$ is the mean summary resulting from matching the modal configuration of $\pi(\boldsymbol{x}|\boldsymbol{y})$ while $\boldsymbol{Q}^* = \boldsymbol{Q} + \mathrm{diag}(\boldsymbol{c})$ with

$$\boldsymbol{c} = \begin{pmatrix} -\frac{\partial^2}{\partial^2 \eta_1}[\sum_{i=1}^n y_i\eta_i - \exp(\eta_i)]|_{\eta_1 = x_1^*} \\ \vdots \\ -\frac{\partial^2}{\partial^2 \eta_n}[\sum_{i=1}^n y_i\eta_i - \exp(\eta_i)]|_{\eta_n = x_n^*} \\ -\frac{\partial^2}{\partial^2 \beta_0}[\sum_{i=1}^n y_i(\beta_0 + \beta_1\xi_i) - \exp(\beta_0 + \beta_1\xi_i)]|_{\beta_0 = x_{n+1}^*} \\ -\frac{\partial^2}{\partial^2 \beta_1}[\sum_{i=1}^n y_i(\beta_0 + \beta_1\xi_i) - \exp(\beta_0 + \beta_1\xi_i)]|_{\beta_1 = x_{n+2}^*} \end{pmatrix}, \tag{3.5}$$

which contains all the negative second derivatives of the log-likelihood in (3.3) evaluated at the modal points within $\boldsymbol{x}^*$ with respect to each latent parameter x_i. If we consider a setting with low count data modeled by a Poisson distribution, we may find the Gaussian Approximation not as accurate as we need. Therefore, we can exploit the Skew Gaussian Copula densities introduced in the previous section to deal with such applications. Since we expect the outcomes of the Poisson model to be skewed, we can choose a transformation $\boldsymbol{h}(\cdot)$ that adds more skewness adjustments on the marginals so that the joint approximation can properly recover the true un-

derlined posterior density. Skew Normal family distributions represent an efficient, natural choice. Through this class, we can still retain the dependency structure encoded in $\boldsymbol{Q}$ due to the use of the Gaussian copula. While Section 2.5 shows that marginal Bayesian posterior analysis is an easy and efficient hack in INLA, the same methodology may not be enough in more extreme contexts. When the hyperparameter information is strongly correlated to one or more parameters or is intrinsically part of the underlying likelihood model structure, we might need to turn to a joint posterior inference and use different tools such as the aforementioned Skew Gaussian Copula.

3.1.3 Gaussian Approximation and linear constraints

As described in Section 2.5, the Gaussian Approximation is fundamental to approximate the hyperparameter posterior marginal $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ with a Laplace Approximation. Moreover, the Gaussian Approximation is also part of the available INLA strategies to get fast and accurate marginal deterministic results. From (2.4) we see that the full conditional we need to approximate is

$$\pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}) \propto \exp\Big(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}(\boldsymbol{\theta})\boldsymbol{x} + \sum_{i=1}^{n}\log\pi(y_i|x_i,\boldsymbol{\theta})\Big) \tag{3.6}$$

with usual LGM assumptions on each likelihood point $\pi(y_i|x_i,\boldsymbol{\theta})$ and Gaussian latent field $\boldsymbol{x}$ with sparse precision matrix $\boldsymbol{Q}$. A simple but powerful trick to approximate such densities is to collect and evaluate the available information such that the resulting density resembles a Gaussian distribution. In this way, we get the respective Gaussian Approximation for (3.6) as

$$\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}) = (2\pi)^{-\frac{N}{2}}|\boldsymbol{Q}^*(\boldsymbol{\theta})|^{\frac{1}{2}}\exp\Big(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}(\boldsymbol{\theta}))^T\boldsymbol{Q}^*(\boldsymbol{\theta})(\boldsymbol{x}-\boldsymbol{\mu}(\boldsymbol{\theta}))\Big) \tag{3.7}$$

which corresponds to a multivariate Gaussian density with mean $\boldsymbol{\mu(\theta)}$ and precision matrix $\boldsymbol{Q}^*(\boldsymbol{\theta})$ with $\boldsymbol{Q}^*(\boldsymbol{\theta}) = \boldsymbol{Q}(\boldsymbol{\theta}) + \text{diag}(\boldsymbol{c}(\boldsymbol{\theta}))$ where $\boldsymbol{c}(\boldsymbol{\theta})$ contains the negative second derivatives of the log density evaluated at the mode of (3.6). Appendix A provides some details about the iterative Newton Raphson process to obtain these summaries. We can also construct a Gaussian Approximation that corrects for linear constraints of the form $\boldsymbol{Cx} = \boldsymbol{e}$ with $\boldsymbol{C}$ and $\boldsymbol{e}$ being a $N \times k$ matrix with rank k and a real vector respectively. For this setting, the previous mean summary of the approximation is replaced in the iterative process with the expected value of a sample drawn from a constrained Gaussian Markov Random Field (see Section 2.3). The new corrected mean is obtained by using the formula

$$\boldsymbol{x}^c = \boldsymbol{x} - \boldsymbol{Q^{-1}C^T}(\boldsymbol{CQ^{-1}C^T})^{-1}(\boldsymbol{Cx} - \boldsymbol{e}), \tag{3.8}$$

where $\boldsymbol{x}$ denotes an unconstrained GMRF sample. The expected value of the formula (3.8) is especially useful to approximate the conditional mode used for the Laplace Approximation strategy in Chapter 2. Linear constraints account for proper identifiability of the parameters and must be carefully encoded in the precision structure when constructing a joint posterior approximation to $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$. Blind use of the Gaussian approximation can lead to inaccuracies in cases where the likelihood assumptions are far from being Gaussian (Binomial or Poisson data, for example). Here the approximations can fail in recovering the true result. Ferkingstad and Rue (2015) propose and achieve more corrected approximations by constructing a more accurate Gaussian Approximation with marginal location and skewness adjustments.

3.1.4 Mathematical derivation of a Skew Gaussian Copula (SGC)

In this section, we finally describe how to construct a Skew Gaussian Copula joint density approximation by defining a proper Gaussian Copula for the joint Gaussian

Approximation density and a well defined transformation $\boldsymbol{h}(\cdot)$. From the Gaussian approximation in (3.7), we construct a new class of joint approximation densities by assuming a new latent random vector $\tilde{\boldsymbol{x}}$ and a set of marginal transformations $\tilde{\boldsymbol{x}} = \boldsymbol{h}(\boldsymbol{x})$ applied on the original latent field. Both the transformation and the Gaussian Copula contribute to construct the Skew Gaussian Copula joint approximation density. The copula retains the same dependency structure encoded in the precision matrix $\boldsymbol{Q}$ of the original latent field $\boldsymbol{x}$ according to its assumed GMRF formulation. The marginal transformations in $\boldsymbol{h}(\boldsymbol{x})$ are flexible as they can borrow features from the more accurate posterior marginal approximations computed in INLA. The transformation choice is parametric and defines one and only one joint approximation within the Skew Gaussian Copula class. The class then sees a new defined random latent vector $\tilde{\boldsymbol{x}} = (h_1(x_1), \ldots, h_N(x_N))$ through the vector function $\boldsymbol{h}(\cdot)$. Here we consider $\tilde{\boldsymbol{x}} \sim \boldsymbol{F}$ where $\boldsymbol{F} = (F_1, \ldots, F_i, \ldots, F_N)$ is a vector of cumulative distribution functions assumed for the Gaussian Copula construction. Since we want to encode skewness into the new joint approximation through the copula, we choose each F_i to be from a Skew Normal density (2.55). Such choice is particularly appealing with the way INLA achieves marginal densities through its Simplified Laplace strategy by using Skew Normal approximations. Accordingly, the standardized latent field is $\tilde{\boldsymbol{z}} = \frac{\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})}{\tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})}$ with $\tilde{\boldsymbol{z}} \sim \tilde{\boldsymbol{F}}$ such that $\tilde{\boldsymbol{F}}$ is the vector of cumulative distribution functions of standard Skew Normal random variables with improved mean $\tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})$ and standard deviation $\tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})$. Thus, we can now apply the Gaussian Copula and get explicit expressions for the set of transformations $\boldsymbol{h}(\cdot)$, to get a new joint approximation density to (3.7). We summarise the methodology step by step as follows:

1. Consider $\boldsymbol{z} = \frac{\boldsymbol{x} - \boldsymbol{\mu}(\boldsymbol{\theta})}{\boldsymbol{\sigma}(\boldsymbol{\theta})}$ to be the original standardized latent variables with respect to the Gaussian approximation $\tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ with respective posterior summaries $(\boldsymbol{\mu}(\boldsymbol{\theta}), \boldsymbol{\sigma}(\boldsymbol{\theta}))$.

2. Construct $\tilde{\boldsymbol{z}} = \boldsymbol{h}(\boldsymbol{z}) = \tilde{\boldsymbol{F}}^{-1}(\boldsymbol{\Phi}(\boldsymbol{z}))$ with $\boldsymbol{\Phi} = (\Phi_1, \ldots, \Phi_i, \ldots, \Phi_N)$ being the

vector of standard cumulative Gaussian distribution functions where $\boldsymbol{\Phi}_i = \Phi$, $\forall i$ by definition. Note that the joint cumulative distribution function of $\Phi_1(z_1), \ldots, \Phi_i(z_i), \ldots, \Phi_N(z_N)$ is exactly the Gaussian Copula.

3. Thus $\Phi(z_i) \sim U(0,1)$, $\forall i$ by using the *Probability Integral Transform* (PIT) theorem while its inverse application leads to $\tilde{\boldsymbol{z}} \sim \tilde{\boldsymbol{F}}$ meaning that

$$\begin{aligned} \tilde{\boldsymbol{x}} &= \tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})\boldsymbol{h}(\boldsymbol{z}) + \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta}) \\ &= \tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})\tilde{\boldsymbol{F}}^{-1}(\boldsymbol{\Phi}(\boldsymbol{z})) + \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta}) \end{aligned} \tag{3.9}$$

where $\tilde{\boldsymbol{x}} \sim \boldsymbol{F}$ as per prior assumption.

The use of Skew Normal marginals in $\tilde{\boldsymbol{F}}$ allows us to borrow the more accurate properties from the Simplified Laplace strategy and encode them into the resulting Skew Gaussian Copula joint approximation. Its marginal properties would then be adjusted for skewness, extending the method's modeling applicability. Furthermore, we get an explicit expression for the vector transformation $\boldsymbol{h}(\cdot)$ as

$$\boldsymbol{h}(\boldsymbol{z}) = \tilde{\boldsymbol{F}}^{-1}\Big[\boldsymbol{\Phi}\Big(\frac{\boldsymbol{x} - \boldsymbol{\mu}(\boldsymbol{\theta})}{\boldsymbol{\sigma}(\boldsymbol{\theta})}\Big)\Big] \tag{3.10}$$

and its inverse

$$\boldsymbol{z} = \boldsymbol{h}^{-1}(\tilde{\boldsymbol{z}}) = \boldsymbol{\Phi}^{-1}\Big[\tilde{\boldsymbol{F}}\Big(\frac{\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})}{\tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})}\Big)\Big] \tag{3.11}$$

by using the change of variable theorem. Since $\boldsymbol{h}(\cdot)$ is now parameterically known, we can write down a more precise Skew Gaussian Copula density of (3.2) with respect to the new latent field $\tilde{\boldsymbol{x}}$ as follows

$$\begin{aligned}\tilde{\pi}_{\text{SGC}}(\tilde{\boldsymbol{x}}|\boldsymbol{\theta},\boldsymbol{y}) &= \tilde{\pi}_{\text{G}}(\boldsymbol{g}(\tilde{\boldsymbol{x}})|\boldsymbol{\theta},\boldsymbol{y})|\boldsymbol{J}_{\boldsymbol{x}}| \\ &= (2\pi)^{-\frac{N}{2}}|\boldsymbol{Q}^{*}(\boldsymbol{\theta})|^{\frac{1}{2}}\exp\Big[-\frac{1}{2}(\boldsymbol{g}(\tilde{\boldsymbol{x}})-\boldsymbol{\mu}(\boldsymbol{\theta}))^{T}\boldsymbol{Q}^{*}(\boldsymbol{\theta})(\boldsymbol{g}(\tilde{\boldsymbol{x}})-\boldsymbol{\mu}(\boldsymbol{\theta}))\Big]|\boldsymbol{J}_{\boldsymbol{x}}|\end{aligned} \tag{3.12}$$

with $\boldsymbol{g}(\tilde{\boldsymbol{x}}) = \boldsymbol{h}^{-1}\Big(\frac{\tilde{\boldsymbol{x}}-\tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})}{\tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})}\Big)\boldsymbol{\sigma}(\boldsymbol{\theta}) + \boldsymbol{\mu}(\boldsymbol{\theta})$ and $|\boldsymbol{J}_{\boldsymbol{x}}|$ being the Jacobian determinant of the applied vectorial transformation,

$$\boldsymbol{J}_{\boldsymbol{x}} = \Big[\frac{\partial \boldsymbol{x}}{\partial \tilde{x}_1},\ldots,\frac{\partial \boldsymbol{x}}{\partial \tilde{x}_N}\Big] = \begin{pmatrix} \frac{\partial x_1}{\partial \tilde{x}_1} & 0 & \ldots & 0 \\ 0 & \frac{\partial x_2}{\partial \tilde{x}_2} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \ldots & 0 & \frac{\partial x_N}{\partial \tilde{x}_N} \end{pmatrix} \tag{3.13}$$

with $|\boldsymbol{J}_{\boldsymbol{x}}| = \Big|\prod_i \frac{\partial x_i}{\partial \tilde{x}_i}\Big|$ where $\frac{\partial x_i}{\partial \tilde{x}_i} \geq 0$, $\forall i$. Next we compute each differential component of $\boldsymbol{J}_{\boldsymbol{x}}$ by differentiating the inverse transformation in (3.11) with respect to each latent component $\tilde{x}_i$

$$\frac{\partial x_i}{\partial \tilde{x}_i} = \frac{\tilde{f}_i\Big(\frac{\tilde{x}_i-\tilde{\mu}_i(\boldsymbol{\theta})}{\tilde{\sigma}_i(\boldsymbol{\theta})}\Big)}{\phi\Big(\Phi^{-1}\Big[\tilde{F}_i\Big(\frac{\tilde{x}_i-\tilde{\mu}_i(\boldsymbol{\theta})}{\tilde{\sigma}_i(\boldsymbol{\theta})}\Big)\Big]\Big)} \tag{3.14}$$

and derive a full density representation of the Skew Gaussian Copula by plugging (3.12) and (3.14) into the same expression

$$\tilde{\pi}_{\text{SGC}}(\tilde{\boldsymbol{x}}|\boldsymbol{\theta},\boldsymbol{y}) = (2\pi)^{-\frac{N}{2}}|\boldsymbol{Q}^{*}(\boldsymbol{\theta})|^{\frac{1}{2}}\exp\Big[-\frac{1}{2}[\boldsymbol{t}(\tilde{\boldsymbol{x}})]^{T}\boldsymbol{Q}^{*}(\boldsymbol{\theta})[\boldsymbol{t}(\tilde{\boldsymbol{x}})]\Big]\prod_{i=1}^{N}\delta_i(\tilde{x}_i) \tag{3.15}$$

with $\boldsymbol{t}(\tilde{\boldsymbol{x}}) = \boldsymbol{\Phi}^{-1}\Big[\tilde{\boldsymbol{F}}\Big(\frac{\tilde{\boldsymbol{x}}-\tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})}{\tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})}\Big)\Big]\boldsymbol{\sigma}(\boldsymbol{\theta})$ and $\delta_i(\tilde{x}_i) = \frac{\tilde{f}_i\Big(\frac{\tilde{x}_i-\tilde{\mu}_i(\boldsymbol{\theta})}{\tilde{\sigma}_i(\boldsymbol{\theta})}\Big)}{\phi\Big(\Phi^{-1}\Big[\tilde{F}_i\Big(\frac{\tilde{x}_i-\tilde{\mu}_i(\boldsymbol{\theta})}{\tilde{\sigma}_i(\boldsymbol{\theta})}\Big)\Big]\Big)}$, $\forall i$. By choice of the vector transformation $\boldsymbol{h}(\cdot)$, the joint density in (3.15) deviates from the Gaus-

sian pattern of the Gaussian Approximation while maintaining the same precision structure in $\boldsymbol{Q}^*$. The Skew Normal assumption propagates to the marginal transformations $\boldsymbol{h}(\cdot)$ in (3.9) leading to a general form of the Skew Gaussian Copula joint approximation. Although other choices are possible, this representation appears to be the most coherent and efficient towards INLA methodology. As an example, if skewness is negligible, then we can assume an identity relation of the form $\tilde{\boldsymbol{z}} = \boldsymbol{z}$ and the expression in (3.9) simplifies into

$$\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})\boldsymbol{z} + \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta}) = \boldsymbol{x} - \boldsymbol{\mu}(\boldsymbol{\theta}) + \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta}), \tag{3.16}$$

which applies location adjustments by shifting the original mean of the Gaussian approximation by the improved mean term $\tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})$ of the Simplified Laplace marginals computed in INLA. We also point out that $\boldsymbol{\sigma}(\boldsymbol{\theta})$ and $\tilde{\boldsymbol{\sigma}}(\boldsymbol{\theta})$ are the same by construction of both the Gaussian and Simplified Laplace approximations. Correspondingly the density in (3.12) degenerates into a Gaussian Approximation with improved mean $\tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})$ and precision matrix $\boldsymbol{Q}^*(\boldsymbol{\theta})$

$$\tilde{\pi}_{\mathrm{IG}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y}) = (2\pi)^{-\frac{N}{2}} |\boldsymbol{Q}^*(\boldsymbol{\theta})|^{\frac{1}{2}} \exp\Big[-\frac{1}{2}(\boldsymbol{x} - \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta}))^T \boldsymbol{Q}^*(\boldsymbol{\theta})(\boldsymbol{x} - \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta}))\Big] \tag{3.17}$$

which we denote as Improved Gaussian Approximation (IG). Both the Gaussian Approximation and its Improved version in (3.17) belong to the Skew Gaussian Copula class. The improved case is straightforward while the standard one is obtained by assuming the set of cumulative marginals $\boldsymbol{F}$ for $\tilde{\boldsymbol{x}}$ to be the ones from the Gaussian approximation marginals $\tilde{\pi}_{\mathrm{G}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$, $\forall i$. Since a proper parametric density form is available, we can measure the entity of the corrections added to the new joint approximation compared to its Gaussian counterpart.

3.1.5 Skewness Correction Differential on the log-joint approximation

From (3.7) we know that the joint density of the Gaussian Approximation at the denominator evaluated at its mean point $\boldsymbol{x} = \boldsymbol{\mu}(\boldsymbol{\theta})$ is

$$\log \tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\Big|_{\boldsymbol{x}=\boldsymbol{\mu}(\boldsymbol{\theta})} = \frac{1}{2}|\boldsymbol{Q}^*(\boldsymbol{\theta})| - \frac{N}{2}\log(2\pi) \tag{3.18}$$

To measure the amount of skewness correction being added to the new transformation on $\tilde{\pi}_G(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$, we must evaluate the new class of joint approximations defined by (3.15) at the same mean point of the new latent random field $\tilde{\boldsymbol{x}}$. Similarly to (3.15),

$$\log \tilde{\pi}_{\mathrm{SGC}}(\tilde{\boldsymbol{x}}|\boldsymbol{\theta}, \boldsymbol{y})\Big|_{\tilde{\boldsymbol{x}}=\boldsymbol{\mu}(\boldsymbol{\theta})} = \frac{1}{2}\log|\boldsymbol{Q}^*(\boldsymbol{\theta})| - \frac{1}{2}[\boldsymbol{t}(\boldsymbol{\mu}(\boldsymbol{\theta}))]^T\boldsymbol{Q}^*(\boldsymbol{\theta})[[\boldsymbol{t}(\boldsymbol{\mu}(\boldsymbol{\theta}))]] + \sum_{i=1}^{N}\log\delta_i(\mu_i(\boldsymbol{\theta})) \tag{3.19}$$

Through the evaluated quantities in (3.18) and (3.19), we can account for the differential correction when using the new transformed joint density,

$$\begin{aligned}\Delta_{\boldsymbol{x},\tilde{\boldsymbol{x}}} &= \Big[\log \tilde{\pi}_{\mathrm{G}}(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y}) - \log \tilde{\pi}_{\mathrm{SGC}}(\tilde{\boldsymbol{x}}|\boldsymbol{\theta}, \boldsymbol{y})\Big]\Big|_{(\boldsymbol{x},\tilde{\boldsymbol{x}})=\boldsymbol{\mu}(\boldsymbol{\theta})} \\ &= \frac{1}{2}[\boldsymbol{t}(\boldsymbol{\mu}(\boldsymbol{\theta}))]^T\boldsymbol{Q}^*(\boldsymbol{\theta})[[\boldsymbol{t}(\boldsymbol{\mu}(\boldsymbol{\theta}))]] - \sum_{i=1}^{N}\log\delta_i(\mu_i(\boldsymbol{\theta}))\end{aligned} \tag{3.20}$$

Here we notice that the expression in (3.20) degenerates into

$$\Delta_{\boldsymbol{x},\tilde{\boldsymbol{x}}} = \frac{1}{2}(\boldsymbol{\mu}(\boldsymbol{\theta}) - \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta}))^T\boldsymbol{Q}(\boldsymbol{\theta})(\boldsymbol{\mu}(\boldsymbol{\theta}) - \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})) \tag{3.21}$$

if we consider the Skew Gaussian Copula with only mean corrected marginals in (3.17). Again, this particular case of the class happens when the transformations in $\boldsymbol{h}(\cdot)$ are

assumed to be an identity. The differential of the applied skewness correction in (3.21) provides an indicator of the mass of probability adjustment when we employ a Skew Gaussian Copula with Skew Normal marginals.

3.2 Posterior Approximations for Linear Combinations

From Section 3.1 we know that the Skew Gaussian Copula class provides a family of joint density approximations for densities of the form $\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ within a Latent Gaussian Model framework. We can also exploit the general form of this class with marginal skewness correction to get approximations for posterior marginals and linear combinations in a subset of the latent field. Unlike the Gaussian Approximation, the density of the new multivariate class of joint approximations in (3.15) does not have a known probabilistic form, but we can still compute its summaries through its Gaussian Copula construction. By computing the first three order moments of a mixture representation of Skew Gaussian Copula densities, we can obtain all the information we need to approximate the target posterior marginals for the latent field subset of interest by exploiting Skew Normal distributions. After manipulating and matching the moments with the respective Skew Normal ones, we get results that still retain structure and properties of the Skew Gaussian Copula class with corresponding Skew Normal marginal approximations. Section 3.2.1 focuses on the construction of such surrogate Skew Gaussian Copula from its mixture representation to approximate posterior marginal densities within a subset of the latent field $\boldsymbol{x}$. From this straightforward application, Section 3.2.2 extends the methodology to additive linear combinations $\boldsymbol{Ax}$ with similar assumptions.

3.2.1 Latent Field marginal approximations in a subset

Consider a subset of the latent field $\boldsymbol{x}$ defined by a set of indexes $S = \{i|i \in \{1, \ldots, N\}, |\boldsymbol{x}| = N\}$. We aim to compute an approximation for $\pi(\boldsymbol{x}_S|\boldsymbol{y})$ whose

density can be further decomposed into

$$\pi(\boldsymbol{x}_S|\boldsymbol{y}) = \int \pi(\boldsymbol{x}_S|\boldsymbol{\theta}, \boldsymbol{y})\pi(\boldsymbol{\theta}|\boldsymbol{y})\, d\boldsymbol{\theta} \tag{3.22}$$

Similarly to the Laplace Approximation strategy in INLA (see Section 2.5.3), we can construct an approximation for (3.22) by exploiting a mixture representation of the Skew Gaussian Copula class of joint densities introduced in Section 3.1

$$\tilde{\pi}(\boldsymbol{x}_S|\boldsymbol{y}) \approx \sum_{k=1}^{K} \tilde{\pi}_{\text{SGC}}(\boldsymbol{x}_S|\boldsymbol{\theta}_k, \boldsymbol{y}) w_k \tag{3.23}$$

where $w_k = \tilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y})\Delta_k$ with normalised weights as $\sum_{k=1}^{K} w_k = 1$. The joint densities in expression (3.23) are the full conditional Skew Gaussian Copula approximations with corresponding density in (3.15). We can extend the Gaussian Approximation applicability through this class of approximations by encoding skewness into its structure with Skew Normal marginal transformations. In a simpler case, we may consider the Improved Gaussian Approximation in (3.17) and apply a correction on the mean only. Thus we may approximate the entire object in (3.23) with a multivariate Gaussian distribution which still benefits from the Skew Gaussian Copula construction. However, this choice might be limiting for more extreme applications where the skewness plays a bigger role. Since the general version of the Skew Gaussian Copula does not have a proper shape, we can construct a surrogate of the same class by taking advantage of the mixture structure in (3.23). Because of the sum properties, we can easily compute the moments up to order three for each involved Skew Gaussian Copula density and combine them into one single object with similar properties. We integrate out the posterior summary of $\tilde{\pi}_{\text{SGC}}(\boldsymbol{x}_S|\boldsymbol{\theta}_k, \boldsymbol{y})$ with respect to each integration point $\boldsymbol{\theta}_k$ and then combine them all into one. Such algebraic manipulation leads to a new multivariate joint structure that still preserves the precision structure and properties of the original Skew Gaussian Copula but with newfound moments.

This approach can also be applied to linear combinations since we retain both the mean vector $\tilde{\boldsymbol{\mu}}(\boldsymbol{\theta})$ of the Skew Normal marginals and the covariance matrix structure $\boldsymbol{\Sigma}^*_{\boldsymbol{SS}}(\boldsymbol{\theta})$ within a sub-block indexed by S. Here the covariance matrix $\boldsymbol{\Sigma}^*_{\boldsymbol{SS}}(\boldsymbol{\theta})$ denotes the extracted solution within the assumed subset S of the linear system $\boldsymbol{Q\Sigma} = \boldsymbol{I}$. As $\boldsymbol{x}_S|\boldsymbol{y} \sim \tilde{\pi}(\boldsymbol{x}_S|\boldsymbol{y})$ by assumption, then we can calculate its moments in terms of a Skew Gaussian Copula as follows

$$\begin{aligned}
\mathrm{E}_{\mathrm{SGC}}[\boldsymbol{x}_S^p|\boldsymbol{y}] &= \int_S \boldsymbol{x}_S^p \tilde{\pi}(\boldsymbol{x}_S|\boldsymbol{y})\, d\boldsymbol{x}_S \\
&= \int_S \boldsymbol{x}_S^p \sum_{k=1}^{K} \tilde{\pi}_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{\theta}_k, \boldsymbol{y}) w_k\, d\boldsymbol{x}_S \\
&= \sum_{k=1}^{K} w_k \int_S \boldsymbol{x}_S^p \tilde{\pi}_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{\theta}_k, \boldsymbol{y})\, d\boldsymbol{x}_S \\
&= \sum_{k=1}^{K} w_k\, \mathrm{E}_{\mathrm{SGC}}[\boldsymbol{x}_S^p|\boldsymbol{\theta}_k, \boldsymbol{y}] \\
&= \sum_{k=1}^{K} w_k \tilde{\boldsymbol{\mu}}_S^{(p)}(\boldsymbol{\theta}_k) \qquad (3.24)
\end{aligned}$$

where $\tilde{\boldsymbol{\mu}}_S^{(j)}(\boldsymbol{\theta}_k)$ defines all the j^{th} marginal non central moments of the full conditional densities in (3.24). The moments up to order $p = 3$ construct a surrogate Skew Gaussian Copula joint approximation to $\pi(\boldsymbol{x}_S|\boldsymbol{y})$ with mean $\mathrm{E}_{\mathrm{SGC}}[\boldsymbol{x}_S|\boldsymbol{y}] = \tilde{\boldsymbol{\mu}}_S$, covariance structure $\boldsymbol{\Sigma}^*$ and Skew Normal marginals by matching the third order moment $\mathrm{E}_{\mathrm{SGC}}[\boldsymbol{x}_S^3|\boldsymbol{y}]$. Through the third order moment we get the skewness $\gamma_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{y})$ by using the formula

$$\gamma_{\mathrm{SGC}}(\boldsymbol{x}|\boldsymbol{y}) = \frac{\mathrm{E}_{\mathrm{SGC}}(\boldsymbol{x}_S^3|\boldsymbol{y}) - 3\,\mathrm{E}_{\mathrm{SGC}}(\boldsymbol{x}_S^2|\boldsymbol{y})\,\mathrm{E}_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{y}) + 2\,\mathrm{E}^3_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{y})}{[\mathrm{E}_{\mathrm{SGC}}(\boldsymbol{x}_S^2|\boldsymbol{y}) - \mathrm{E}^2_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{y})]^{\frac{3}{2}}} \qquad (3.25)$$

Since mean, variance, and skewness for each marginal are available, we can derive proper Skew Normal densities by simply matching their respective moments. Indeed, we can map each skewness component to the respective Skew Normal parameters denoted by location ξ, scale ω, and skewness index α. This task can be accomplished by using the method of moment estimation procedure (MME), which sets a three equations system based on matching the first three order moments (Ghorbanzadeh et al. (2017)). We can compute all the results by using the available δ-parameterization of the Skew Normal family as follows

Definition 9 (δ**-parameterization**)

Let $\{\xi_i, \omega_i, \alpha_i\}$ *be a set of parameters triplet of a Skew Normal distribution and* γ_i *the skewness for each marginal latent field term* x_i *with mean* μ_i *and variance* σ_i^2 *for* $i = 1, \ldots, N$*. Then we can analytically compute a new* $\tilde{\delta}_i$ *parameter*

$$\tilde{\delta}_i = \text{sign}(\gamma_i) \sqrt{\frac{\frac{\pi}{2}|\gamma_i|^{2/3}}{\left(\frac{4-\pi}{2}\right)^{2/3} + |\gamma_i|^{2/3}}} \tag{3.26}$$

in terms of the skewness γ_i*. Based on MME, we get*

$$\begin{aligned}
\tilde{\alpha}_i &= \frac{\tilde{\delta}_i}{\sqrt{1 - \tilde{\delta}_i^2}} \\
\tilde{\omega}_i &= \sqrt{\frac{\pi \sigma_i^2}{\pi - 2\tilde{\delta}_i^2}} \\
\tilde{\xi}_i &= \mu_i - \tilde{\omega}_i \tilde{\delta}_i \sqrt{\frac{2}{\pi}}
\end{aligned}$$

which is the δ*-parameterization for the initial triplet.*

Each marginal of the surrogate Skew Gaussian Copula joint approximation $\tilde{\pi}_{\text{SGC}}(\boldsymbol{x}_S|\boldsymbol{y})$ in the subset S is represented by a Skew Normal density derived from its respective system solution.

3.2.2 Extension to Linear Combinations $\boldsymbol{Ax}$

We can also apply Skew Gaussian Copula joint approximations to additive linear combinations involving latent components. Due to the additive structure, we expect the true posterior densities of these linear combinations to closely follow a Gaussian pattern as the subset dimension $|S|$ increases. Therefore the Skew Gaussian Copula class appears to be a proper natural candidate for constructing accurate approximations of their posterior densities. In this section we define an additive linear combination $l(\boldsymbol{x}) = \boldsymbol{Ax}$ as a vector of dimension M with $\boldsymbol{A}$ being a $M \times N$ matrix of indexes that generalizes the notation S into M additive linear combinations and $\boldsymbol{x}$ follows the distribution of a Skew Gaussian Copula as in (3.15). Its joint density is approximated by

$$\tilde{\pi}(\boldsymbol{Ax}|\boldsymbol{y}) \approx \sum_{k=1}^{K} \tilde{\pi}_{\text{SGC}}(\boldsymbol{Ax}|\boldsymbol{\theta}_k, \boldsymbol{y}) w_k \tag{3.27}$$

with Skew Gaussian Copula approximations applied on the linear combination vector $\boldsymbol{Ax}$. Similarly to (3.24) we can compute the moments for (3.27) by using

$$\mathrm{E}_{\text{SGC}}[(\boldsymbol{Ax})^p|\boldsymbol{y}] = \sum_{k=1}^{K} w_k \, \mathrm{E}_{\text{SGC}}[(\boldsymbol{Ax})^p|\boldsymbol{\theta}_k, \boldsymbol{y}] \tag{3.28}$$

where $(\boldsymbol{Ax})^p = \boldsymbol{A}^p\boldsymbol{x}^p$ in expression (3.28) denotes a power vector-wise evaluation of each component in the M-dimensional linear combination vector $\boldsymbol{Ax}$. Therefore the respective posterior moments up to order $p = 3$ are the following

$$\begin{aligned}
\mathrm{E}_{\mathrm{SGC}}[\boldsymbol{Ax}|\boldsymbol{y}] &= \boldsymbol{A}\sum_{k=1}^{K} w_k \,\mathrm{E}_{\mathrm{SGC}}[\boldsymbol{x}|\boldsymbol{\theta}_k, \boldsymbol{y}] = \boldsymbol{A}\sum_{k=1}^{K} w_k \tilde{\boldsymbol{\mu}}(\boldsymbol{\theta_k}) = \boldsymbol{A}\tilde{\boldsymbol{\mu}} \\
\mathrm{E}_{\mathrm{SGC}}[(\boldsymbol{Ax})^2|\boldsymbol{y}] &= \sum_{k=1}^{K} w_k \,\mathrm{Var}_{\mathrm{SGC}}(\boldsymbol{Ax}|\boldsymbol{\theta}_k, \boldsymbol{y}) + \sum_{k=1}^{K} w_k [\mathrm{E}_{\mathrm{SGC}}[\boldsymbol{Ax}|\boldsymbol{\theta}_k, \boldsymbol{y}]]^2 \\
&= \mathrm{diag}[\boldsymbol{A}\boldsymbol{\Sigma}^*\boldsymbol{A}^T] + [\boldsymbol{A}\tilde{\boldsymbol{\mu}}]^2 \qquad (3.29)
\end{aligned}$$

where $\tilde{\boldsymbol{\mu}}$ and $\boldsymbol{\Sigma}^*$ are the mean vector and covariance matrix of the Skew Gaussian Copula joint approximation after integrating out $\boldsymbol{\theta}$. Since we need to evaluate the third moment to get the skewness of the linear combination $\boldsymbol{Ax}$, we can use its central moment representation

$$\begin{aligned}
\mathrm{E}_{\mathrm{SGC}}[(\boldsymbol{Ax} - \boldsymbol{A}\tilde{\boldsymbol{\mu}})^3|\boldsymbol{y}] &= \sum_{k=1}^{K} w_k \,\mathrm{E}_{\mathrm{SGC}}[(\boldsymbol{Ax} - \boldsymbol{A}\tilde{\boldsymbol{\mu}}(\boldsymbol{\theta_k}))^3|\boldsymbol{\theta}_k, \boldsymbol{y}] \\
&= \boldsymbol{A}^3 \,\mathrm{E}_{\mathrm{SGC}}[(\boldsymbol{x} - \tilde{\boldsymbol{\mu}})^3|\boldsymbol{y}] \\
&= \boldsymbol{A}^3 \boldsymbol{\gamma}_{\mathrm{SGC}}(\boldsymbol{x}|\boldsymbol{y})[\mathrm{diag}(\boldsymbol{\Sigma^*})]^{\frac{3}{2}} \qquad (3.30)
\end{aligned}$$

which only depends on the central third-order moments of the latent field $\boldsymbol{x}$ since all the other mixed moments are zero (see applications in Phillips (2010) for more details on moments of a multivariate Gaussian distribution). Thus, we can evaluate the overall skewness for the linear combination vector $\boldsymbol{Ax}$ as

$$\boldsymbol{\gamma}_{\mathrm{SGC}}(\boldsymbol{Ax}|\boldsymbol{y}) = \frac{\mathrm{E}_{\mathrm{SGC}}[(\boldsymbol{Ax} - \boldsymbol{A}\tilde{\boldsymbol{\mu}})^3|\boldsymbol{y}]}{(\mathrm{diag}[\boldsymbol{A}\boldsymbol{\Sigma}^*\boldsymbol{A}^T])^{\frac{3}{2}}} \qquad (3.31)$$

Since the Gaussian copula structure remains unscathed except for the marginals, we can apply the same argument of (3.23) to the Skew Gaussian Copula in (3.27). The first two order non central moments in (3.29) provide information for the mean sum-

mary $\boldsymbol{A}\tilde{\boldsymbol{\mu}}$ and the new covariance structure $\boldsymbol{A}\boldsymbol{\Sigma}^*\boldsymbol{A}^T$ where $\boldsymbol{\Sigma}^*$ is the entire $N \times N$ covariance structure of the latent field $\boldsymbol{x}$. The third central order moment in (3.30) is not zero by Skew Gaussian Copula construction and provides skewness information for each marginal linear combination in $\boldsymbol{Ax}$. Cross covariance terms within $\boldsymbol{A}\boldsymbol{\Sigma}^*\boldsymbol{A}^T$ define the covariance amongst different linear combinations h and l and are accordingly computed as

$$\begin{aligned}\mathrm{Cov}_{\mathrm{SGC}}\Big[\sum_{i=1}^{N} A_{h,i}x_i, \sum_{i=1}^{N} A_{l,i}x_i\Big|\boldsymbol{y})\Big] &= \mathrm{E}_{\mathrm{SGC}}\Big[\sum_{i=1}^{N} A_{h,i}x_i \sum_{i=1}^{N} A_{l,i}x_i\Big|\boldsymbol{y})\Big] - \\ &- \mathrm{E}_{\mathrm{SGC}}\Big[\sum_{i=1}^{N} A_{h,i}x_i\Big|\boldsymbol{y})\Big]\,\mathrm{E}_{\mathrm{SGC}}\Big[\sum_{i=1}^{N} A_{l,j}x_i\Big|\boldsymbol{y})\Big] \\ &= \sum_{i=1}^{N} A_{h,i}A_{l,i}\Sigma^*_{i,i} + \sum_{i=1}^{N}\sum_{j=1}^{i-1}(A_{h,i}A_{l,j} + A_{h,j}A_{l,i})\Sigma^*_{i,j}\end{aligned} \tag{3.32}$$

The weighted moment structure in (3.24) and (3.28) provide the posterior summaries for the surrogate Skew Gaussian Copula which approximates the joint density $\pi(\boldsymbol{Ax}|\boldsymbol{y})$. As a result, the skewness adjustment is induced by approximating the marginals of the joint density with Skew Normal distributions.

3.2.3 Approximating two linear combinations jointly

In previous sections we introduced new tools to build deterministic approximations for posterior joint densities of the form $\pi(\boldsymbol{x}_S|\boldsymbol{y})$ and $\pi(\boldsymbol{Ax}|\boldsymbol{y})$ in a subset S of the latent field $\boldsymbol{x}$. Due to their deterministic nature, these approximations are fast to compute and construct a surrogate Skew Gaussian Copula object in a subset of the latent field (see an application on Appendix C). To fully understand the mathematical approach, we provide a two dimensional example where we construct surrogate Skew Gaussian Copula approximations $\tilde{\pi}_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{y})$ and $\tilde{\pi}_{\mathrm{SGC}}(\boldsymbol{Ax}|\boldsymbol{y})$. We consider a two

dimensional latent field $\boldsymbol{x} = (x_1, x_2)$, corresponding to $S = \{1, 2\}$, and matrix of indexes $\boldsymbol{A} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ to define linear combinations $\boldsymbol{Ax} = (x_1 + x_2, x_1 - x_2)$. Based on the formulas in (3.24), we assume that $\pi(\boldsymbol{x}_S|\boldsymbol{y})$ is approximated by a surrogate Skew Gaussian Copula with $\mathrm{E}_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{y}) = (\tilde{\mu}_1, \tilde{\mu}_2) = (1, 2)$, $\boldsymbol{\Sigma}^* = \begin{pmatrix} 2 & 1 \\ 1 & 5 \end{pmatrix}$ and skewness $\boldsymbol{\gamma}_{\mathrm{SGC}}(\boldsymbol{x}_S|\boldsymbol{y}) = (-0.4, 0.6)$. Then we construct a surrogate Skew Gaussian Copula approximation to $\pi(\boldsymbol{Ax}|\boldsymbol{y})$ with observations $\boldsymbol{y}$. From (3.29) and (3.30) we compute the moments

$$\begin{aligned} \mathrm{E}_{\mathrm{SGC}}[\boldsymbol{Ax}|\boldsymbol{y}] &= \boldsymbol{A}\tilde{\boldsymbol{\mu}} = (\tilde{\mu}_1 + \tilde{\mu}_2, \tilde{\mu}_1 - \tilde{\mu}_2) = (3, -1) \\ \boldsymbol{A}\boldsymbol{\Sigma}^*\boldsymbol{A}^T &= \begin{pmatrix} 9 & -3 \\ -3 & 5 \end{pmatrix} \\ \boldsymbol{\gamma}_{\mathrm{SGC}}(\boldsymbol{Ax}|\boldsymbol{y}) &= \frac{\boldsymbol{A}^3\boldsymbol{\gamma}_{\mathrm{SGC}}(\boldsymbol{x}|\boldsymbol{y})[\mathrm{diag}(\boldsymbol{\Sigma}^*)]^{\frac{3}{2}}}{(\mathrm{diag}[\boldsymbol{A}\boldsymbol{\Sigma}^*\boldsymbol{A}^T])^{\frac{3}{2}}} = (0.206, -0.701) \end{aligned} \tag{3.33}$$

By construction of the Skew Gaussian Copula, we approximate the corresponding marginals $\pi(x_1 + x_2|\boldsymbol{y})$ and $\pi(x_1 - x_2|\boldsymbol{y})$ from $\boldsymbol{Ax}$ using Skew Normal densities with location ξ, scale ω and skewness parameter α

$$\begin{aligned} \tilde{\pi}(x_1 + x_2|\boldsymbol{y}) &\approx SN(-0.107, 1.796, 1.217) \\ \tilde{\pi}(x_1 - x_2|\boldsymbol{y}) &\approx SN(4.633, 3.454, -3.233) \end{aligned} \tag{3.34}$$

The parameters for the expressions above are obtained by matching the moments in (3.33) to the δ-parameterization in Definition 9. The example shows that manipulating the posterior summaries of a Skew Gaussian Copula still leads to a similar surrogate of the same class. Thus we obtain a straightforward analytical approxima-

tion for each additive linear combination component in $\boldsymbol{Ax}$, therefore easing their inference analysis. Although this other joint approximation may lack accuracy in more complex settings, the available results are convincing and allow streamlined and fast marginal inference for additive linear combinations of the latent field.

3.3 Mixture of Skew Gaussian Copula densities

Marginal posterior inference is well established in the INLA methodology by using Laplace Approximations onto mixture representations of their latent posterior marginals (see Chapter 2 for general details). The Skew Gaussian Copula class introduced in Section 3.1 offers a new mathematical formulation to approximate the full joint density $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ by exploiting a Gaussian Copula and marginal skewness adjustments. Some features of the class go beyond the simple Gaussian Approximation and can be exploited to compute deterministic approximations for posteriors of linear combinations of the latent field as well (see Section 3.2). While these initial tools can properly account for these posterior problems, the same does not hold for the entire joint posterior density of a Latent Gaussian Model in (2.4). In general, its shape is unknown, and we cannot construct a deterministic approximation that satisfies any of the available INLA strategies. Even though no particular joint distribution can be assumed, we can still use the INLA methodology and new joint inference tools to construct a close approximation to the truth by using an accurate sampling Monte Carlo scheme. We recall the mixture representation of the posterior marginals in expression (2.56) and the new defined class of Skew Gaussian Copula densities $\tilde{\pi}_{\text{SGC}}(\tilde{\boldsymbol{x}}|\boldsymbol{\theta},\boldsymbol{y})$ which assumes a Gaussian Copula construction. Then we can write down a similar mixture density structure for the approximation of the overall joint $\pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})$ as

$$\tilde{\pi}(\tilde{\boldsymbol{x}},\boldsymbol{\theta}|\boldsymbol{y}) \propto \sum_k \tilde{\pi}_{\text{SGC}}(\tilde{\boldsymbol{x}}|\boldsymbol{\theta},\boldsymbol{y})\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\mathbb{1}_{[\boldsymbol{\theta}=\boldsymbol{\theta}_k]}\Delta_k \tag{3.35}$$

with $\{\boldsymbol{\theta}_k, k = 1, \ldots, K\}$ being the configuration points obtained from the grid exploration phase and Δ_k the area weights (see Section 2.5.2 for details). We interpret the approximation (3.35) as a *mixture of Skew Gaussian Copula distributions* where Skew Normal densities provide parametric assumptions for its marginals. This class of joint approximations represents the only source of contribution and error to the full joint approximation accuracy. For this reason, the resulting joint approximation does not carry the same accuracy of the posterior marginals $\tilde{\pi}(x_i|\boldsymbol{y})$ since it lacks a Laplace approximation step. As similar as it may appear to expression (2.56) for the marginal approximations, this joint approximation representation is achieved through an exact sampling Monte Carlo approach conditioned on the pre-computed grid points of the hyperparameter space obtained in (2.49). Using sampling may sound like a step back when the entire INLA philosophy relies on fast to compute deterministic approximations. However, as per sampling-based strategies like Markov Chain Monte Carlo methods, we are not blindly exploring the joint parameter space with proposal distributions. The method builds a joint posterior approximation of the latent field point by point using weighted pre-computed probabilities of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and samples from the respective Gaussian Copula used to define the new approximation. The following two-step scheme summarises how this is done:

- we draw samples from the entire hyperparameter set $\boldsymbol{\theta}$ using the pre-computed configuration points $\{\boldsymbol{\theta}_k, k = 1, \ldots, K\}$ in terms of their mass probability function, as shown in (3.35). This translates into sampling from a multinomial process where each $\boldsymbol{\theta}_k$ has a point-mass probability;

- for each sampled configuration point $\boldsymbol{\theta}_k$ a N-dimensional sample is drawn from $\tilde{\pi}_{\text{SGC}}(\tilde{\boldsymbol{x}}|\boldsymbol{\theta}, \boldsymbol{y})$ with weighted probability $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})\mathbb{1}_{[\boldsymbol{\theta}=\boldsymbol{\theta}_k]}$ that sum up to one $\forall k$.

The Skew Gaussian Copula class offers an efficient and accurate formulation to compute the joint posterior approximation in (3.35). We can always draw samples from

this multivariate density by Gaussian Copula construction, and we can do it fast. When considering exceptional cases of the class such as the Gaussian Approximation or its improved version in (3.17), the computations involved in the sampling approach are straightforward because of the main Gaussian structure. Things get more cumbersome when we employ the version of the Skew Gaussian Copula with Skew Normal marginal transformations. While the Gaussian Approximation was already part of the existing INLA implementation, the Skew Gaussian Copula required more coding to keep the computational process fast. We have accomplished such a task by developing a new fast strategy that combines accurate mappings and interpolation of the solutions (see Section 3.3.2).

3.3.1 Skew Normal Marginal Transformations

Differences between mean (3.16) and skewness (3.9) correction can be observed in Figure 3.1 where we assume several levels of skewness on the Skew Normal transformed latent marginals compared to the non transformed latent terms $\{x_1, \ldots, x_N\}$ where the transformation is an identity.

The straight 45 degrees black line represents the mean corrected standardized values z_i while the colored lines show the skewness correction under the quantile Skew-Normal transformation $\tilde{F}^{-1}()$ in (3.9). The intersection points p_l and p_r underline a marginal threshold to detect when the skewness effect changes the distribution. We see that no changes happen in the range $[p_l, p_r]$, which is $\approx [-1.5, 1.5]$. The mean correction described by the black line is straight simply because no transformation is applied in this case. Instead, the skewness correction colored lines underline different behaviors according to the skewness values in the upper left corner of the legend. Here we have chosen six different levels of skewness from -0.8 to 0.8: higher is the skewness, higher is the effect on the marginal latent distribution with a greater focus on the tails. Moreover, a positive (negative) skewness effect leads to mostly underestimating

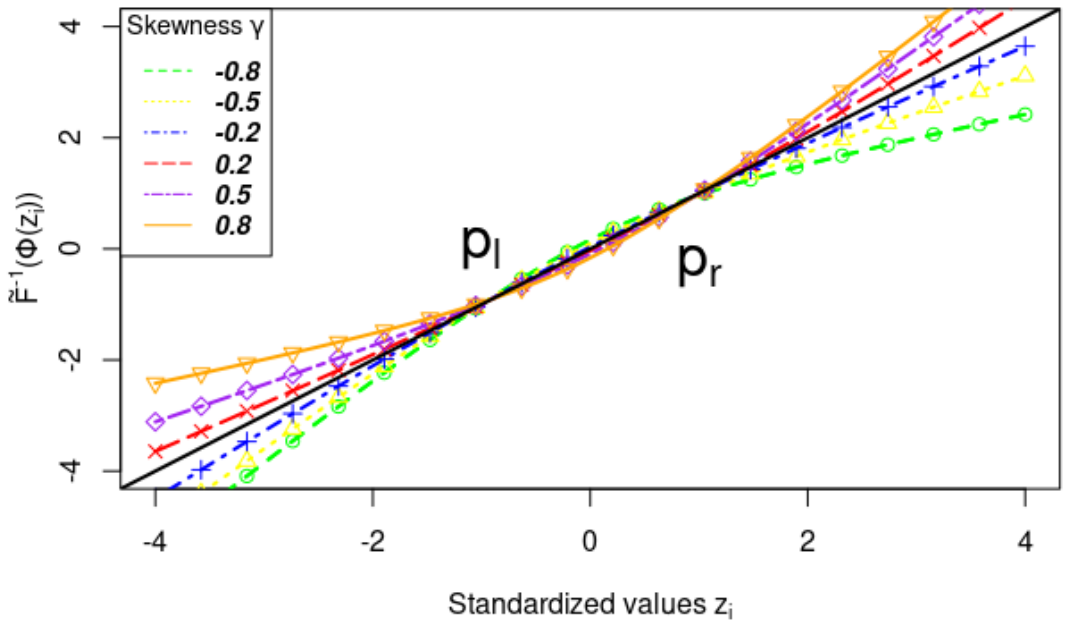


Figure 3.1: Standardized latent values z_i compared with the skewness corrected values $\tilde{x}_i$ through the quantile Skew Normal transformation $\tilde{F}^{-1}(\cdot)$ on the range $(-4, 4)$. The intersection points p_l and p_r are used as an exact threshold for detecting the skewness effect in the tails of the i^{th} marginal distribution.

(overestimating) the latent variables when the Improved Gaussian approximation in (3.17) is used. For example, if we evaluate $z_i = 3$ under marginal skewness equal to -0.8, we would end up with a corrected result below two instead of three by mean correction assumption. However, summaries like mean or median may not detect this effect since most of the correction effect arises in the distribution's tails. The deviation can be severe if the marginal skewness gets higher in absolute terms, and the plot confirms this pattern. Also, the computations of the transformation $\tilde{F}^{-1}()$ are not straightforward and can require several evaluations proportional to the dimension of the latent field $\boldsymbol{x}$. A solution to this computational burden is discussed in the next section.

3.3.2 Computational Strategy for the Skewness Correction

To properly apply a Skew Gaussian Copula approximation for the mixture representation of the joint posterior in (3.35), we need to consider the marginal skewness adjustments of the transformations (3.9) in our computational scheme. Both Gaussian and Improved Gaussian Approximations ignore this task due to their Gaussian nature and structure. In mathematical terms, our target requires to solve the following vectorial quantile equation a possibly large number of times

$$\tilde{F}^{-1}[\Phi(\boldsymbol{x})] = \boldsymbol{p}, \tag{3.36}$$

since the Skew Gaussian Copula joint approximation needs to be evaluated for each configuration point $\boldsymbol{\theta}_k$. The number of operations is then proportional to $N \cdot s$ where N denotes the dimension of the latent field $\boldsymbol{x}$ and s the pre-defined number of samples for constructing the joint posterior approximation in (3.35). This number can increase fast if both N and s are high. Therefore we must solve the equation efficiently to avoid any computational burden. Although the quantile equation in (3.36) does not have a closed-form solution, it is possible to compute exact evaluations through

an optimization step. In R language, an implementation of the solver is available in the function `qsn()` within the Skew Normal package `sn`. The algorithm mainly uses Newton Raphson steps or finite difference methods if the first option fails. The combination of Newton Raphson steps and evaluations of the Gaussian cumulative densities $\Phi(\cdot)$ can be heavy if $N \cdot s$ dimension is high. Also, the lack of vectorization for different skewness values is as critical as the slow optimization solver. Not even the multivariate version of the functions, as suggested by Azzalini and Capitanio (1999) can fix this issue since the required operations are marginal and independent, and more, a multivariate quantile version does not exist. Thus we handle the entire evaluation process through a different intuitive and fast approach that tabulates all the solutions and combines them in a well-defined scheme avoiding unnecessary computations. Proposition 1 summarises the new approach.

Proposition 1 (**Mapping and Interpolation two-way strategy**)
We assume to have access to all marginal skewness $\{\gamma_i, i = 1, \ldots, N\}$ *related to the latent field* $\boldsymbol{x} = \{x_i, i = 1, \ldots, N\}$ *and define the Skew Normal mapping function according to the* δ*-parameterization in Definition 9. Then*

1. *First a local cache of object files is being created within the local private INLA environment in R by assuming few useful initial constants;*

2. *Next the* δ*-parameterization is used to map each skewness* γ_i *into the respective Skew Normal triplet of parameters* $\{\xi_i, \omega_i, \alpha_i\}$*. We exploit these results to fit the correct marginal Skew Normal distribution for each* x_i*;*

3. *For each Skew-Normal marginal a fixed number of points is pre-computed and stored. Then we apply an interpolation process to compute all possible solutions with a pre-defined level of accuracy;*

4. *Finally we detect the correct interpolant by using a binary search algorithm in terms of each marginal skewness input value.*

The above procedure surpasses the limits of available methodologies by solving the vectorial quantile equation in (3.36) from a two-dimensional perspective where the input is a $N \times s$ dimensional object.

3.3.3 Speed Results of the new Quantile function

The numerical solutions to the vectorial problem in (3.36) can be easily achieved with R default packages. They ensure accurate results, but they are too slow for our needs. In this part of the thesis, we propose a new two-way strategy, summarised in Proposition 1, which speeds up these available computational strategies. The results already satisfy our speed requirements without relying on more efficient programming languages such as C/C++ or Python. Here we show a few details behind the new numerical approach with a final table of speed performance between the original R implementations and our alternative code. The constants in the first step of the strategy determine many important numerical details to ensure a good balance of accuracy and speed for the next iterations. We create fixed global constants to control the range, the number of evaluations, and digits accuracy for the required interpolants. The user is free to change these default options at his own will, but any minor adjustment can significantly affect accuracy and speed performances. In general, the default input values give accurate outcomes. For example, the built-in functions construct the interpolants by computing all possible skewness results with a precision up to the second digit. Increasing the number of digits leads to an increment in the number of interpolant functions to be computed. Although this enlarges the marginal fit accuracy of a negligible order, the computational burden becomes much heavier. By keeping these assumptions on the global constants, we can solve the initial problem by tabulating all possible solutions from the standard available quantile solver `qsn()` in R and then save all the interpolants into a cache environment. This part is only computed once and exists in the local R global environment session. Any

Table 3.1: Speed Time Results comparison between the standard `qsn()` function and the new strategy. Function evaluations based on 100 replications and $N = 10^6$ points. Function $\tilde{\boldsymbol{f}}_{\text{std}}$, $\tilde{\boldsymbol{F}}_{\text{std}}$ and $\tilde{\boldsymbol{F}}^{-\mathbf{1}}_{\text{std}}$ are respectively the pdf, cdf and quantile function of the Skew Normal distribution available in `qsn()`. Accordingly $\tilde{\boldsymbol{f}}_{\text{fast}}$, $\tilde{\boldsymbol{F}}_{\text{fast}}$ and $\tilde{\boldsymbol{F}}^{-\mathbf{1}}_{\text{fast}}$ relate to the new strategy.

	Min	**Mean**	**Max**
$\tilde{\boldsymbol{f}}_{\text{std}}$	1.51ms	2.23ms	7.32ms
$\tilde{\boldsymbol{f}}_{\text{fast}}$	1.27ms	2.1ms	6.1ms
$\tilde{\boldsymbol{F}}_{\text{std}}$	$6221\mu s$	$9144\mu s$	$19973\mu s$
$\tilde{\boldsymbol{F}}_{\text{fast}}$	$817\mu s$	$1243\mu s$	$4143\mu s$
$\tilde{\boldsymbol{F}}^{-\mathbf{1}}_{\text{std}}$	22.94s	25.27s	30.46s
$\tilde{\boldsymbol{F}}^{-\mathbf{1}}_{\text{fast}}$	1.61s	1.78s	3.18s

marginal skewness adjustment through the Skew Normal transformation is applied to the samples by calling the correct interpolant stored in the cache. Hence, this new computational approach is both automatic and fast. In Table 3.1, we compare the standard available solver `qsn()` and the new strategy in Proposition 1 under 100 replications. We also report the individual results for probability density function, cumulative density function, and quantile equation of the Skew Normal distribution performance of both implementations in R. The $\tilde{F}^{-1}$ notation in the table refers to $\tilde{F}^{-1}[\Phi(\boldsymbol{x})]$. In contrast, all the employed functions for the comparative analysis are computed using similar interpolants to avoid multiple Skew Normal density computations. The results are obtained through $N = 10^6$ evaluations of the latent field, and we can see that the new strategy achieves the biggest gain in speed. Indeed, the speed-up coming from the new quantile version is approximately 15 times faster than `qsn()` on average. Therefore we can undoubtedly state that the two-way strategy grants better computational advantages than available R solvers and should be preferred in this case.

3.4 Numerical Results using Simulations

This section compares the previously introduced joint posterior approximations based on the newly defined Skew Gaussian Copula class and its derivations. To verify the goodness of this class's new applied skewness corrections, we show the marginal results derived from these joint posterior approximation strategies. We set simulations from both a Poisson and Binomial likelihood in a Latent Gaussian Model framework to trigger more extreme skewed outcomes and underline the copula advantages. In this way, we can better show the accuracy and speed performance of the Skew Gaussian Copula object. We use Markov Chain Monte Carlo (MCMC) sampling approaches from JAGS (Plummer et al. (2003)) to get true, comparable results from the joint posterior. Then we compare these sampled outcomes to the ones obtained by the joint posterior approximation in (3.35) from INLA. Here we use both the Simplified Laplace and full Laplace strategy to point out the different levels of accuracy that affect the joint posterior result when more assumptions are used. More applications of the joint posterior approximation with location adjustments only are discussed in Seppä et al. (2019) or Wakefield et al. (2016). Similar marginal comparisons on the approximations for linear combinations are shown in Section 3.2 where we show the matching results with their sampling counterpart.

3.4.1 Joint Posterior Corrected Inference

To show the features of the Skew Gaussian Copula class, we set a comparative Bayesian analysis using both MCMC methods with JAGS and Laplace strategies with R-INLA. The applications are based on data simulations from Poisson and Binomial likelihoods within a hierarchical GLMM model framework. The hierarchical structure of the Poisson example is described as follows

$$\boldsymbol{y}|\boldsymbol{\mu} \sim \text{Poi}(\boldsymbol{\mu})$$

$$\boldsymbol{\mu} = \exp(\alpha + \boldsymbol{u})$$

$$\boldsymbol{u} \sim \text{N}_m(\boldsymbol{0}, \tau^{-1}\boldsymbol{I})$$

$$\alpha \sim \text{N}(0, 1000)$$

$$\tau \sim \text{Ga}(0.1, 0.1)$$

The data $\boldsymbol{y}$ have been simulated from a Poisson distribution with $N = 50$ observations and $m = 10$ randomized groups for the vector of random effect $\boldsymbol{u}$ and each one was simulated from a Gaussian distribution with standard deviation $\sigma = \sqrt{\tau^{-1}} = 1.5$ to trigger high marginal skewness. Similarly, we construct the Binomial example using a logit link function applied to its probability parameter p. As these simulation settings purposely show skewed posterior marginals, the mean corrected version of the Skew Gaussian Copula, or Improved Gaussian Approximations, may appear inaccurate in describing the outcomes. In this framework, we know that MCMC methods require long-run simulations to be reliable in terms of the Monte Carlo error. Instead, INLA relies on deterministic marginal posterior approximations, which are empirically accurate. Section 3.3 tells us that a full joint posterior inference of a Latent Gaussian Model in INLA is only possible through a sampling-based approach in terms of a mixture of Skew Gaussian Copula densities. The resulting joint posterior approximation benefits from the marginal corrections derived from the Skew Gaussian Copula class. The computational setup in the R language for both software is given below

- **JAGS:** simulating 6×10^6 samples for 20 independent Markov Chains with 10^2 iterations thinning and dropping 10^6 samples as burn-in. We handled the simulation in parallel by using our server at KAUST whose specs are: Intel(R) Xeon(R) Gold 6130 CPUs@2.10GHz with 512 Gb of RAM, two sockets with

16 cores, each with two threads per core. This parallel setting takes around 11 minutes to complete.

- **INLA:** simulating 10^5 samples from the joint sampler with mean and skewness correction. The mean corrected strategy takes around 4 seconds, while the skewness corrected one takes approximately 14 seconds on average. With 10^4 samples, the times are 0,6 and 1,5 seconds for both mean and skewness corrections. We have run these simulations on a Dell laptop whose specs are: Intel(R) Core(TM) i7-10710U CPU @ 1.10GHz with 16 Gb of RAM, one socket with 6 cores with two threads per core. This scheme is computationally heavy for the skewness corrected version of the Skew Gaussian Copula since it requires many evaluations. In most cases, there is no need to use so many samples as we observe less skewed marginals. Therefore, the computational difference between the two corrections is almost negligible. This average time is obtained under 100 function replications. The slow down would have been way more relevant if we used, for example, default R approaches (see Section 3.3.3).

Here we notice the substantial gain in speed with the INLA setup compared to JAGS. The joint INLA sampler is around 170 and 50 times faster for the mean and skewness corrected SGC approach. To keep things simple, we show results of a single linear predictor marginal for both the Poisson and Binomial GLMM example. We extract the marginal representation of these results from the joint posterior outcomes obtained from JAGS and R-INLA to verify the new skewness correction. In particular, Figures 3.2 and 3.4 show the marginal results for the chosen linear predictor marginal using the Simplified Laplace strategy in INLA. This strategy is preferable since the results can be computed fast with a slight accuracy reduction cost. The skewness corrected marginals almost precisely match the Simplified approximated marginals. In contrast, the mean corrected results are more inaccurate to INLA but closer to the MCMC samples produced by JAGS. The mean correction appears to match the

MCMC marginal outcome better around the mode at a detail level, but then it gets inconsistent as soon as we switch to the full Laplace strategy. Figures 3.6, 3.7, 3.8 and 3.9 show the same marginals results when fitting the model with the Laplace strategy in INLA. It can be clearly observed that the skewness corrected results from the joint follow quite closely the MCMC results while the mean corrected ones are farther away. Although the joint posterior approximation does not retain the same accuracy as the approximated Laplace marginals by construction, we still notice that the skewness corrected results produce more appropriate and coherent outcomes than the mean corrected ones and therefore should be preferred in a general application. As observed in Section 3.3.1, positive (negative) marginal skewness leads to an underestimation (overestimation) of the actual latent posterior results. The accurate matching of the marginal MCMC distributions, the approximated INLA marginals, and the skewness corrected marginals from the INLA joint sampler confirms this pattern. The marginal skewness adjustments of the Skew Gaussian Copula construction are more coherent with the marginal outcomes produced by the default INLA strategies. In such extreme settings, the skewness effect on the marginals should not be ignored since the deviation can propagate to the joint distribution. The results show that the general version of the Skew Gaussian Copula with skewness corrected marginals provides more consistent and generally accurate results when the marginal skewness is not negligible. Indeed, when used, the posterior marginal outcomes match the ones from the Simplified Laplace strategy. We can obtain even more accurate results using the full Laplace strategy, which never fails compared to MCMC. Appendix B shows both JAGS and INLA implementations of the hierarchical models of this section using R-INLA language.

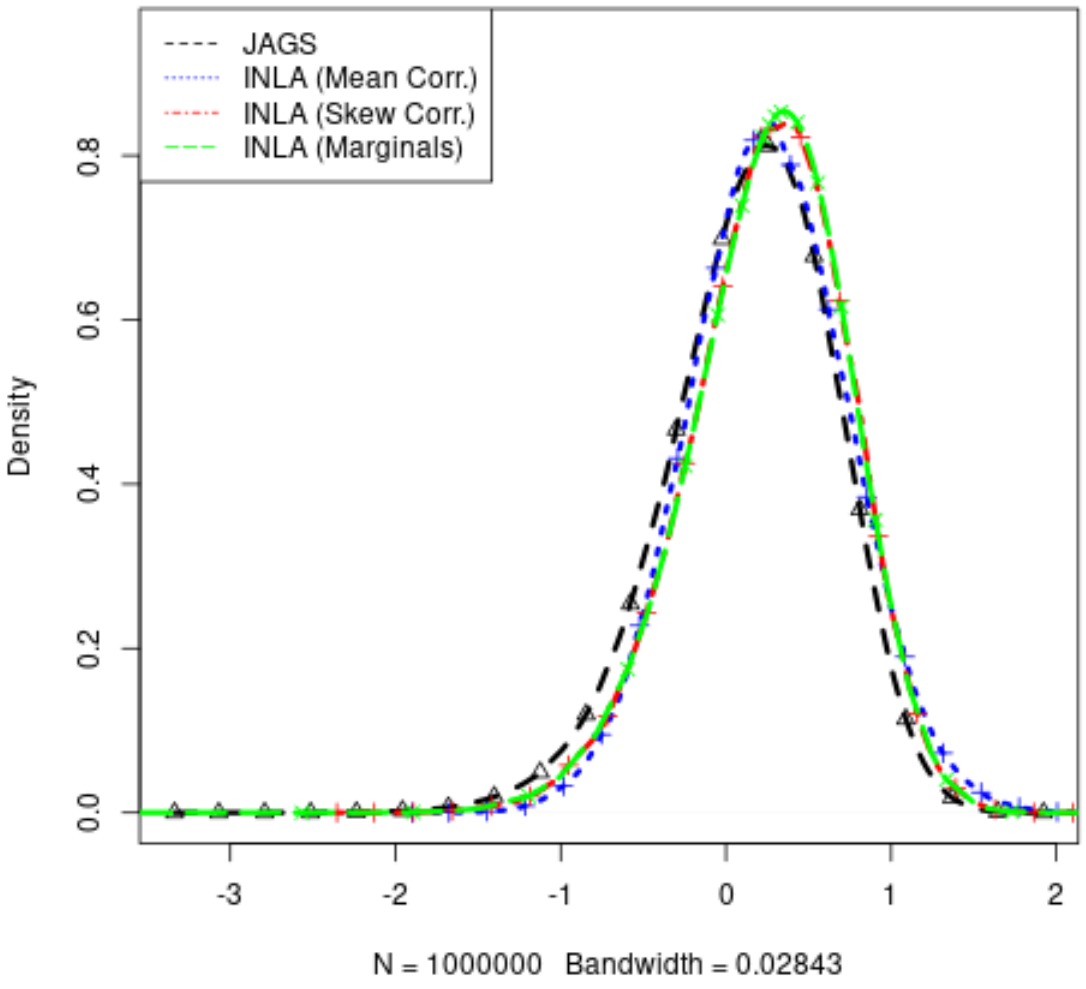


Figure 3.2: Posterior marginal representation for linear predictor η_9 of the Poisson GLMM model with marginal skewness is around -0.38 for all the configuration points. The curves display the outcomes from different strategies: posterior marginal from JAGS (black), mean corrected (blue) and skewness corrected (red) marginal from the SGC and the Simplified Laplace posterior marginal (green) computed by INLA.

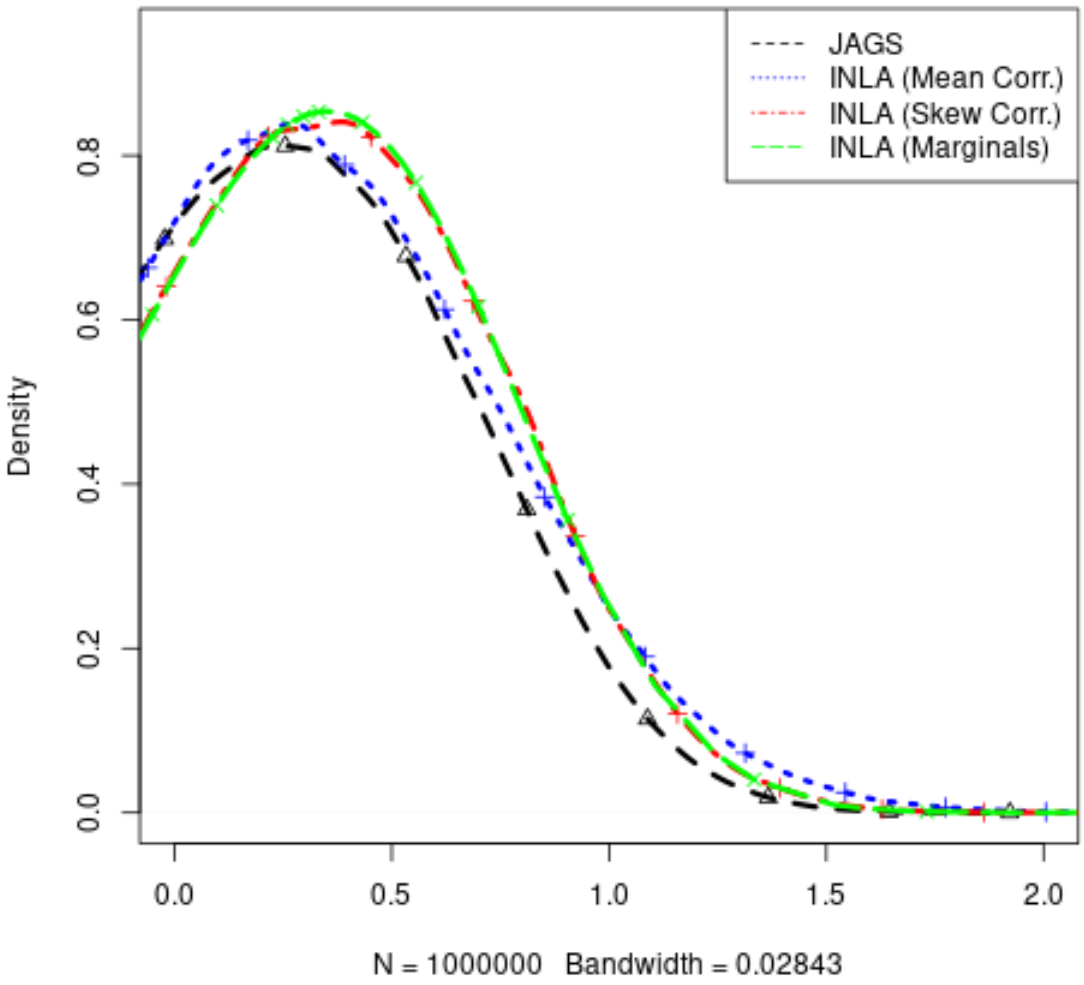


Figure 3.3: A focus on the right tail of the linear predictor η_9 of the Poisson GLMM model where the skewness propagation to the tail is more evident. The skewness corrected marginal (red) from the SGC totally matches with the Simplified Laplace marginal result (green).

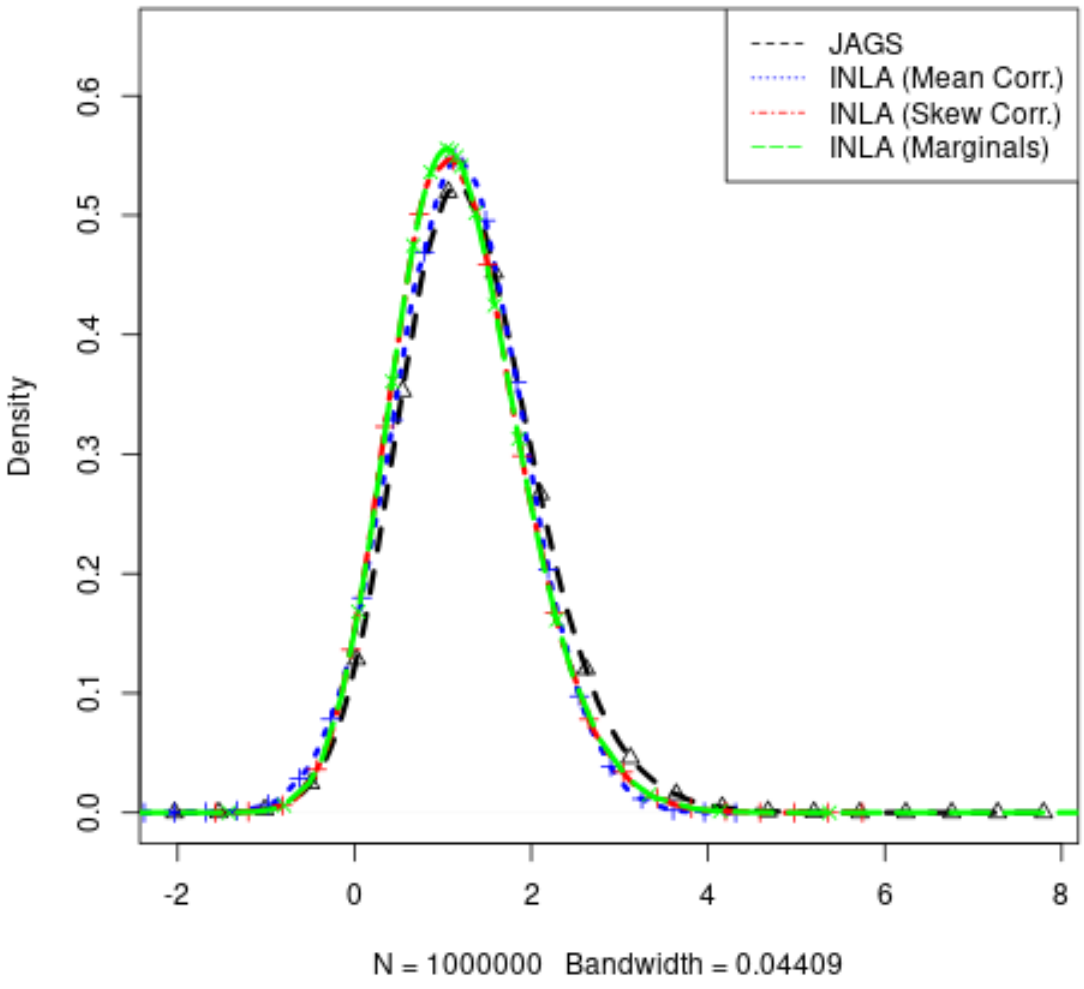


Figure 3.4: Posterior marginal representation for linear predictor η_{14} of the Binomial GLMM model with marginal skewness is around 0.38 for all the configuration points. The curves display the outcomes from different strategies: posterior marginal from JAGS (black), mean corrected (blue) and skewness corrected (red) marginal from the SGC and the Simplified Laplace posterior marginal (green) computed by INLA.

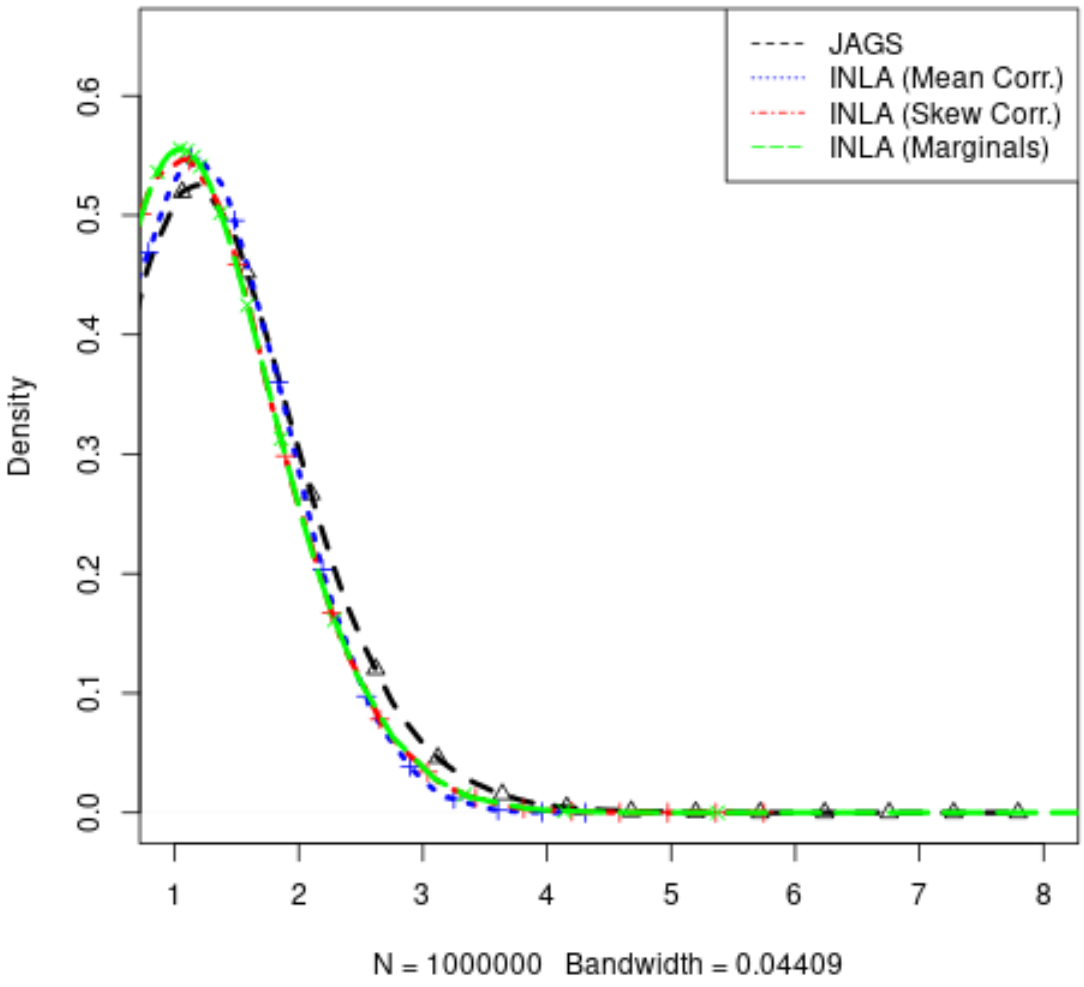


Figure 3.5: A focus on the right tail of the linear predictor η_{14} of the Binomial GLMM model where the skewness propagation to the tail is more evident. The skewness corrected marginal (red) from the SGC totally matches with the Simplified Laplace marginal result (green).

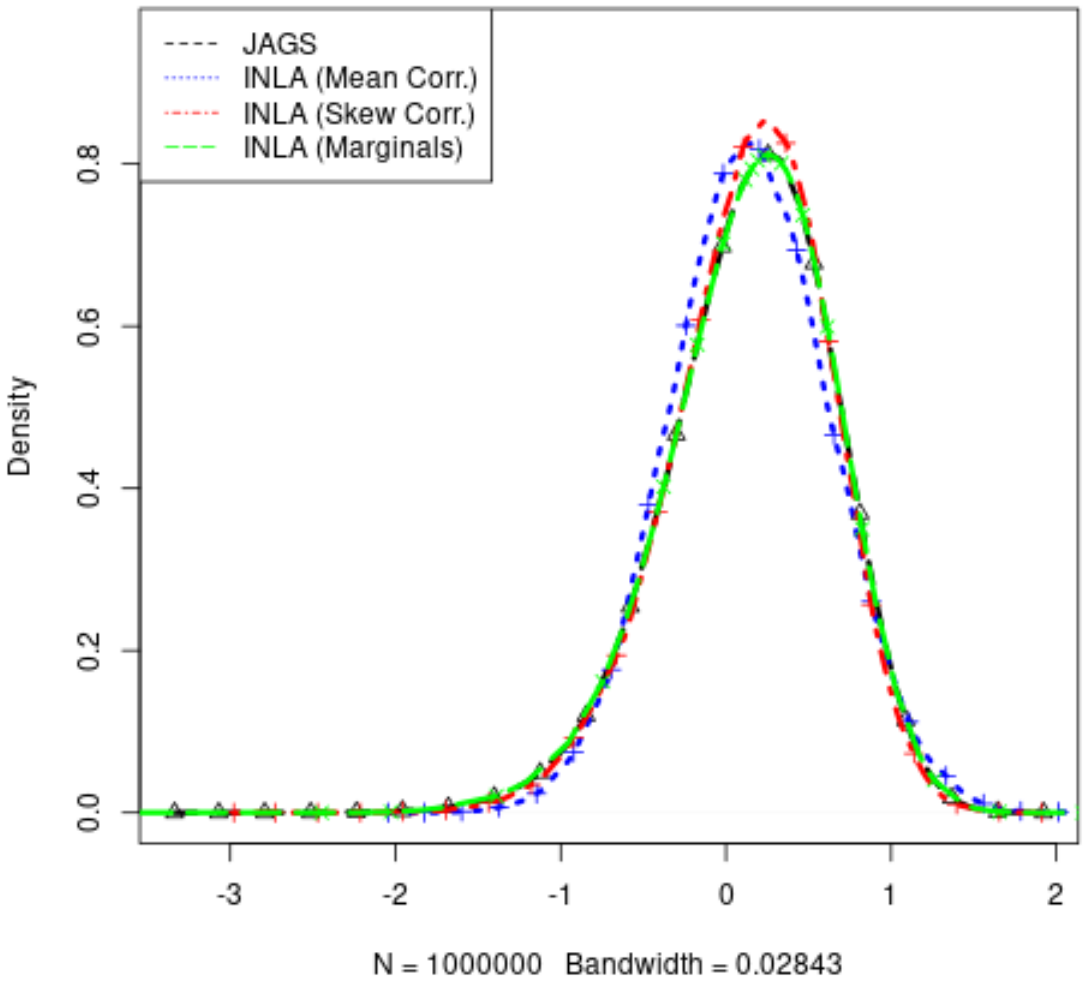


Figure 3.6: Posterior marginal representation for linear predictor η_9 of the Poisson GLMM model with marginal skewness is around -0.4 for all the configuration points. The curves display the outcomes from different strategies: posterior marginal from JAGS (black), mean corrected (blue) and skewness corrected (red) marginal from the SGC and the Laplace posterior marginal (green) computed by INLA.

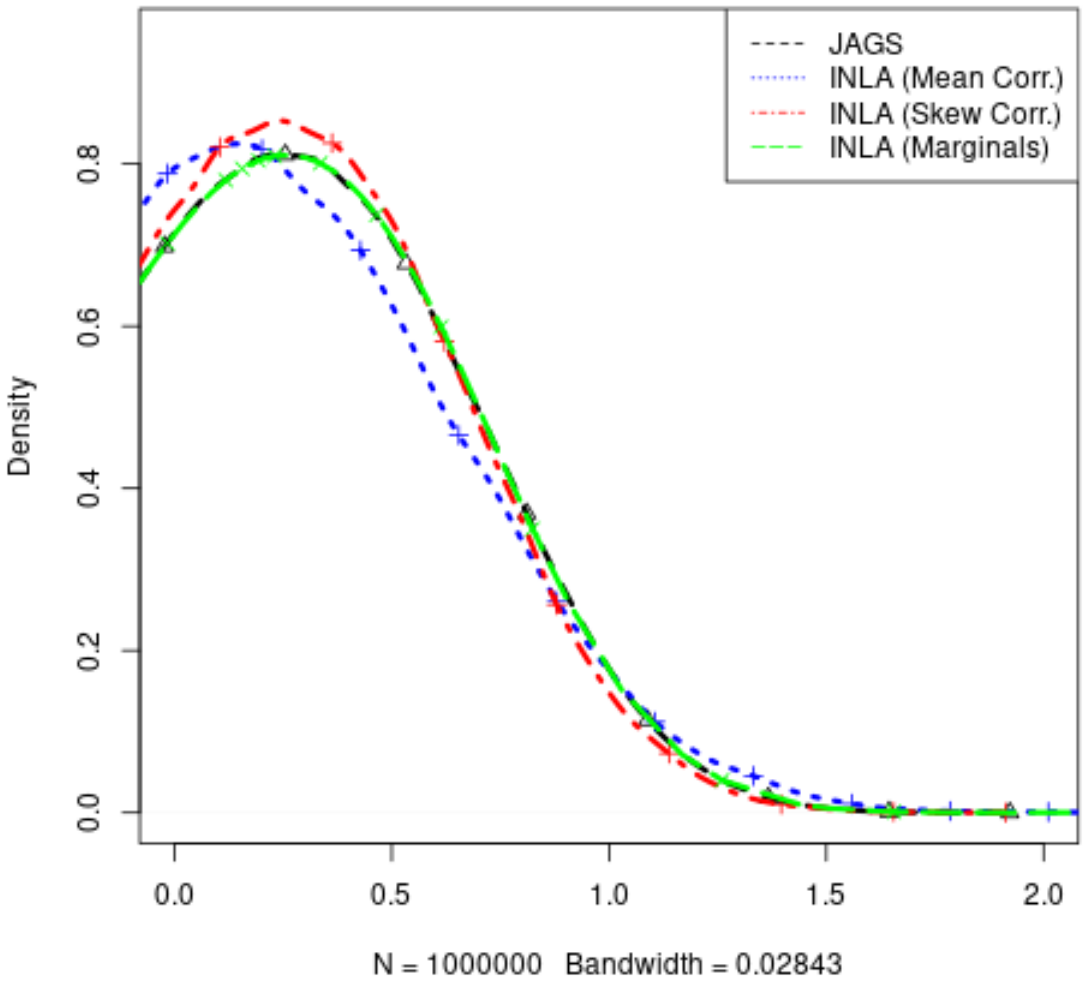


Figure 3.7: A focus on the right tail of the linear predictor η_9 of the Poisson GLMM model where the skewness propagation to the tail is more evident. The skewness corrected marginal (red) from the SGC closely matches with the Laplace (green) and MCMC marginal (black) result.

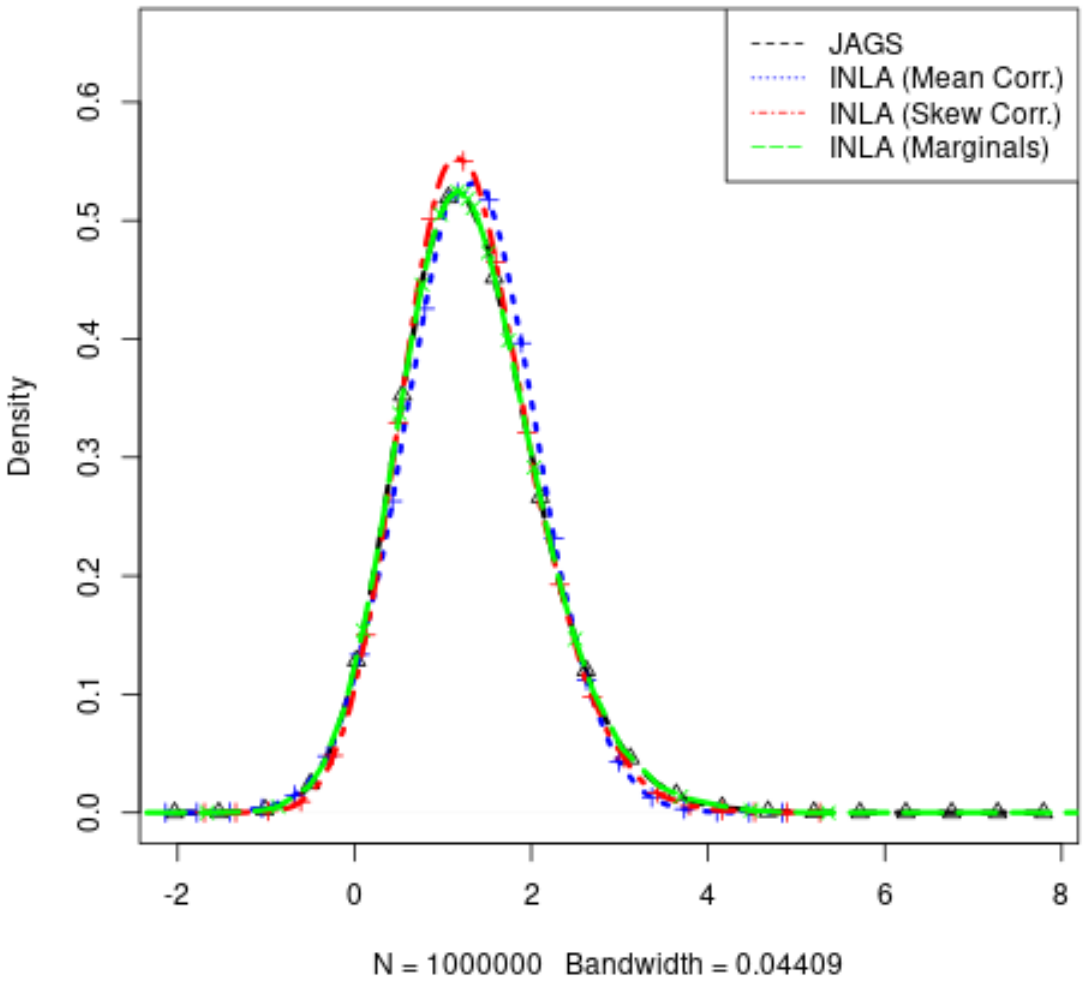


Figure 3.8: Posterior marginal representation for linear predictor η_{14} of the Binomial GLMM model with marginal skewness is around 0.33 for all the configuration points. The curves display the outcomes from different strategies: posterior marginal from JAGS (black), mean corrected (blue) and skewness corrected (red) marginal from the SGC and the Laplace posterior marginal (green) computed by INLA.

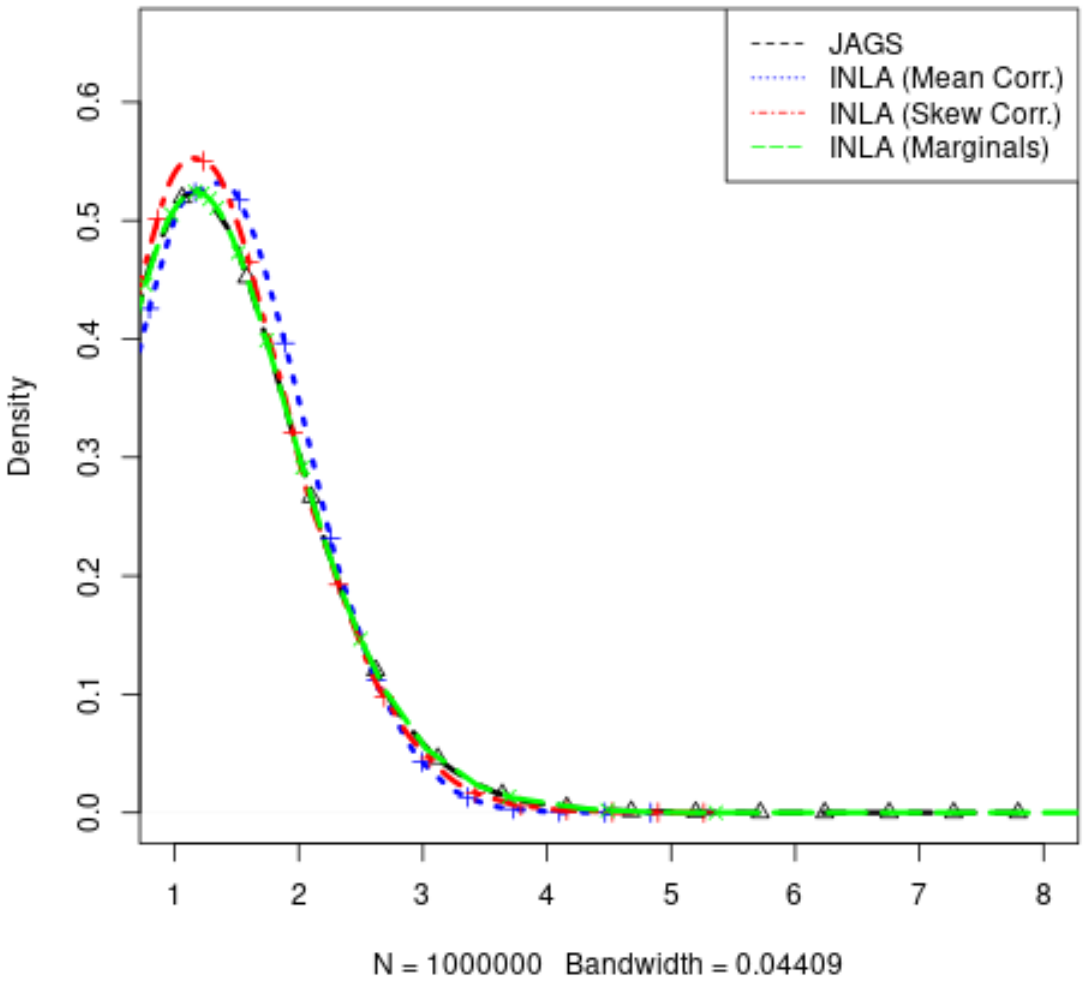


Figure 3.9: A focus on the right tail of the linear predictor η_{14} of the Binomial GLMM model where the skewness propagation to the tail is more evident. The skewness corrected marginal (red) from the SGC closely matches with the Laplace (green) and MCMC marginal (black) result.

3.4.2 Fast Inference for Linear Combinations

The Skew Gaussian Copula class can also be used to construct approximations for linear combinations of a subset of the latent field (see Section 3.2). Appendix C shows an application of new R-INLA tools that accomplish this task. Again we consider the Poisson hierarchical setting of Section 3.4.1 and construct four increasing additive linear combinations in terms of the linear predictors $(\eta_9, \eta_{10}, \eta_{11}, \eta_{12}, \eta_{13})$. We aim to explore the posterior approximations of the linear combinations $\pi(\eta_9 + \eta_{10}|\boldsymbol{y})$, $\pi(\eta_9 + \eta_{10} + \eta_{11}|\boldsymbol{y})$, $\pi(\eta_9 + \eta_{10} + \eta_{11} + \eta_{12}|\boldsymbol{y})$, $\pi(\eta_9 + \eta_{10} + \eta_{11} + \eta_{12} + \eta_{13}|\boldsymbol{y})$ by using both the sample-based joint posterior approximations in (3.35) and its surrogate version specified by (3.27) using moments. The joint posterior approximation obtained by sampling would be more accurate as it exploits the entirety of the information. Figure 3.10 shows a comparison between these two methods in terms of the resulting linear combination marginals. Here we notice no major difference between the two approximation strategies. Hence the surrogate approach by matching moments of a mixture of Skew Gaussian Copula densities can be a preferable choice since it avoids sampling and produces faster results. Posterior summaries for the linear combinations of the Poisson example are shown in Table 3.2 where we compare both approaches through Kullback Leibler distance using Monte Carlo (Hershey and Olsen (2007)). This divergence measure indicates how far the new surrogate approximation based on moment matching is from its true sampling counterpart. Despite the accuracy dispersion of a multivariate object, the KLD measure shows a 10^{-3} order precision for all marginal results. This means that the resulting approximation produces almost identical results to its more accurate sampling version. A simple example can emphasize this argument by considering two generic, well-defined probability density functions f and g that need to be compared. Their KLD is

$$\mathrm{KLD}(f||g) = \int f(t) \log\Big(\frac{f(t)}{g(t)}\Big)\, dt = \mathrm{E}_f\Big[\log\Big(\frac{f(t)}{g(t)}\Big)\Big]$$

We assume f to be a $\mathrm{N}(0,1)$ and g a $\mathrm{N}(\delta,1)$ with δ being a varying location constant. In this setting the computed KLD is

$$\mathrm{KLD}(f||g) = \int \frac{1}{\sqrt{2\pi}} \exp\Big(-\frac{t^2}{2}\Big)\Big[-\frac{t^2}{2} + \frac{(t-\delta)^2}{2}\Big]\, dt = \frac{\delta^2}{2}$$

Therefore a KLD equal to 10^{-3} means that the function g has location $\delta \approx 0.04$ which is really close to the truth. A $\delta \approx 0.01$ corresponds to a KLD equal to 10^{-4}. Similar conclusions apply for varying standard deviation as well. Even the plots show that the lack of accuracy in the surrogate version of the Skew Gaussian Copula is almost negligible compared to its sampling counterpart. Table 3.3 reports the computational time differences of using the joint sampler with a different number of samples and its deterministic surrogate version. From Table 3.3 we observe a considerable speed-up of the latter one when applied to linear combinations. Because of exact algebraic operations, this version is 1100 and 4200 times faster on average than 10^3 and 10^4 samples from the joint posterior sampling-based approach.

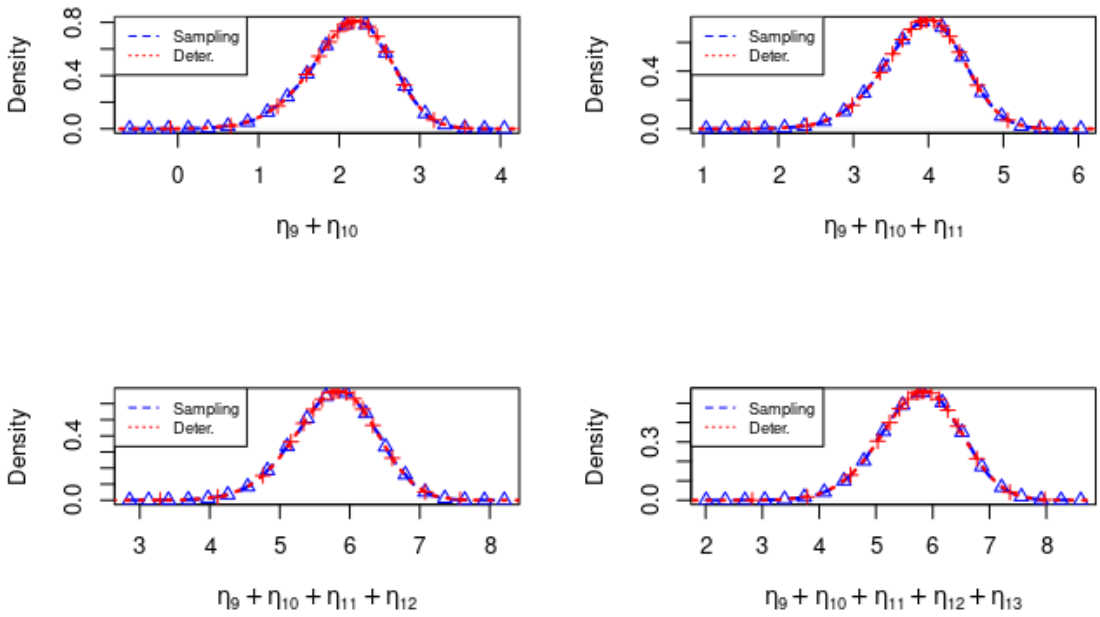


Figure 3.10: One dimensional comparison for all the linear combinations obtained from the joint posterior using 10^5 samples. Blue line marginal is obtained by sampling while red line represents the deterministic marginal result derived from the joint SGC class. All marginal linear combinations are skewed with marginal skewness $\gamma(\eta_9 + \eta_{10}|\boldsymbol{y}) = -0.33$, $\gamma(\eta_9 + \eta_{10} + \eta_{11}|\boldsymbol{y}) = -0.28$, $\gamma(\eta_9 + \eta_{10} + \eta_{11} + \eta_{12}|\boldsymbol{y}) = -0.21$ and $\gamma(\eta_9 + \eta_{10} + \eta_{11} + \eta_{12} + \eta_{13}|\boldsymbol{y}) = -0.18$.

Table 3.2: Posterior summaries and KLD evaluation for all one dimensional linear combinations in the Poisson hierarchical model.

Index	**Mean**	**Sd**	**0.025quant**	**0.5quant**	**0.975quant**	**Mode**	**kld**
$\eta_9 + \eta_{10}$	2.116	0.505	1.042	2.146	3.025	2.209	1.26×10^{-3}
$\eta_9 + \eta_{10} + \eta_{11}$	3.912	0.537	2.783	3.939	4.893	3.995	1.18×10^{-3}
$\sum_{i=9}^{12} \eta_i$	5.776	0.595	4.545	5.798	6.885	5.842	1.21×10^{-3}
$\sum_{i=9}^{13} \eta_i$	5.772	0.718	4.300	5.794	7.120	5.838	1.25×10^{-3}

Table 3.3: Speed Comparison between the joint deterministic algorithm and its sampling version using different sample sizes for computing all one dimensional linear combinations of the Poisson simulation. The performance results have been measured under 100 replications.

Method	**Min**	**Mean**	**Max**
no samples	0.18ms	0.36ms	0.66ms
10^3 samples	177ms	396ms	742ms
10^4 samples	1056ms	1524ms	2024ms

3.5 Discussion

INLA achieves great computational benefits by making approximate Bayesian inference on Latent Gaussian Models focusing on posterior marginal results. In Section 3.1, we also provided a general theory for reaching accurate approximations to the entire joint posterior density that can also encode corrections for both location and skewness. This was possible by combining a Gaussian Copula on the latent field and Skew Normal marginal transformations with borrowed information from the more accurate marginal approximations. We enclosed these joint approximations in a unique class named Skew Gaussian Copula, which applies the marginal adjustments we wanted while retaining the original correlation structure of the model. A mixture representation of this newfound class of approximations allowed a valuable range of possibilities for well approximating posteriors of functionals in heavily skewed settings. Fast and deterministic approximations for sets of additive linear combinations could be produced from a surrogate Skew Gaussian Copula object by matching moments of the mixture with Skew Normal ones. When the functional complexity increases or the analysis scope is larger, we could construct a more accurate joint approximation from the same mixture by using an exact sampling Monte Carlo scheme, including the hyperparameter uncertainty. These approximations granted more accurate and consistent outcomes with INLA when applied in contexts where the posterior is skewed while remaining computationally efficient. This methodology based on Skew Gaussian

copula joint densities has been appropriately embedded in R-INLA but can be computationally heavy in contexts where the grid is finer. If more configuration points for the hyperparameter set $\{\boldsymbol{\theta}_k, k = 1, \ldots, K\}$ are computed, then the additional computational burden becomes more evident. Additional strategies may be needed, such as distributing the computations amongst the configuration points within a parallel setting. Unlike its sampling counterpart, the surrogate approximation based on moment matching is not hindered by a large set of configuration points for the hyperparameter set and instantly produces inference. Because of its deterministic features and fast computations, this construction may be extended to mixed products of the same terms in the linear combination. However, distributions of products do not have a clear shape, and the Skew Normal assumption can likely fail in providing a good approximation. We can show that the product of more than one random variable largely deviates from Gaussianity and gets spikier and spikier. Figure 3.11 shows that even the Skew Normal density is not able to model the shape of a product appropriately. Unless the distributional properties of the non-linear functional are known almost exactly, the new strategies introduced in this chapter have limited use. Even though products of random variables are important in several statistical contexts such as factor models or interaction terms, sums of random variables are more popular, easy to handle, and widely used.

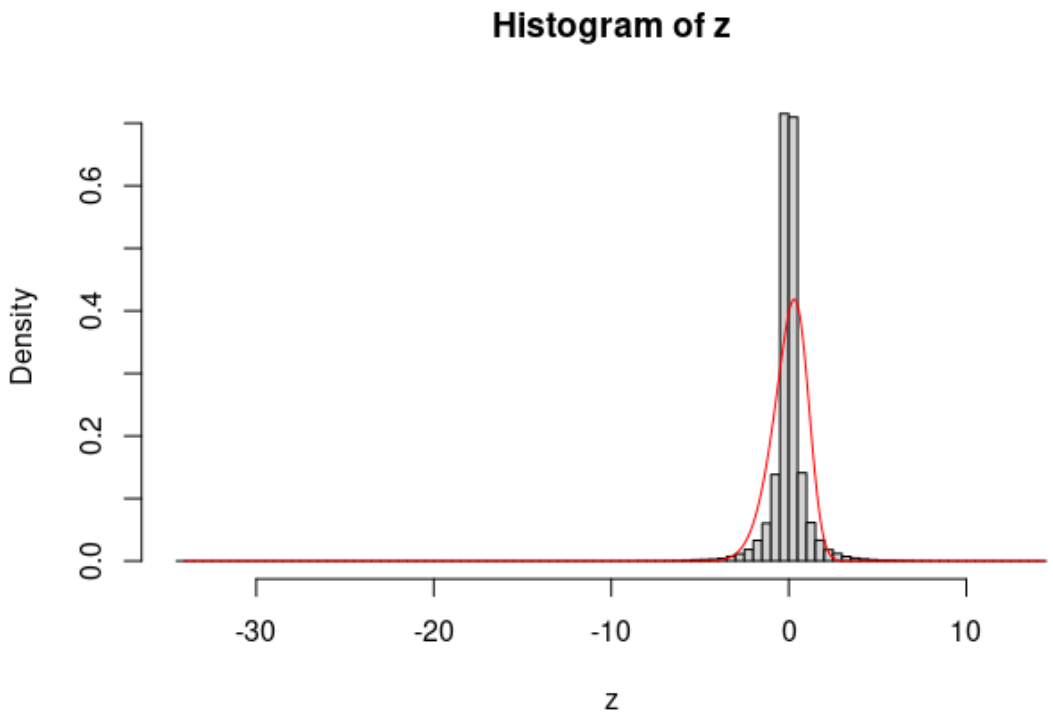


Figure 3.11: Comparative example of a tri-product linear combination of random variables. The histogram shows the true result of the product $z = xyk$ while the red line represents the corresponding Skew Normal adaptation using the moments information. The true result is indeed far from the Skew Normal fit.

Chapter 4

Extending the Simplified Laplace strategy

Latent Gaussian Models represent a broad class of hierarchical models that assume a Gaussian distribution on the latent field, containing all the unobserved parameters. Sampling-based methods such as Markov Chain Monte Carlo (MCMC) can be computationally costly when applied to this class of models. Alternative strategies have been proposed in the literature to overcome the limits of sampling approaches when trying to get marginal posterior distributions Ruli et al. (2014); Ruli and Ventura (2016); Ruli et al. (2016). The Simplified Laplace strategy in INLA appears to be one of the most efficient ways to obtain fast and accurate posterior inference using marginal approximations based on parametric assumptions. Section 4.1 introduces the Gaussian assumptions on the latent field and points out how this pattern propagates to both joint and marginal posterior densities. Approximations computed with the Simplified Laplace approach are particularly appealing for latent hierarchical structure as their tail behavior closely approaches the one from a Gaussian distribution. In Section 4.2 we introduce the details behind this strategy which produces posterior results by fitting Skew Normal distributions to a third-order expansion of a target Laplace approximation. The Skew Normal family contains many parametric skewed densities that can be natural candidates for this approximation task Azzalini and Capitanio (1999, 2003). Therefore we propose an extension of the approach by choosing a different parametric fit for the marginal posterior approximations, which requires matching an additional parameter: the Extended Skew Normal distribution Azzalini and Capitanio (2018); Canale (2011, 2015); Seijas-Macias et al. (2017);

Paulino Pérez-Rodriguez (2017). We provide a straightforward development and application for the new extension by ensuring the computational process remains fast and robust. In Section 4.3 we produce simulations based on skewed observations from Poisson and Binomial likelihood models and compare the posterior marginal results obtained by INLA and JAGS for the MCMC counterpart. The new extended strategy provides encouraging improvements towards its original version using Skew Normal densities. This chapter is based on the respective submitted paper in Chiuchiolo et al. (2022).

4.1 Latent Gaussian Assumption

INLA is extremely efficient when applied to Latent Gaussian Models in terms of providing fast and accurate posterior inference. Through Section 2.2 we extrapolate two main intrinsic assumptions that ensure such high accuracy performance in this hierarchical model framework: the log-likelihood contribution is log-concave in terms of its linear predictor, and the latent field follows a multivariate Gaussian distribution. Although these assumptions seem restrictive, many statistical models can be embedded into this structure. The Gaussian assumption greatly eases the posterior inference process and keeps the operations to their minimum. Moreover, these assumptions contribute to desirable tail properties that follow a Gaussian pattern. This section provides some simple proof of concept and examples to verify this concept.

4.1.1 A Gaussian Latent Field

Log concavity on the log-likelihood is a strong beneficial assumption as it enforces the distribution on the observed data to be nearly Gaussian when we assume conditional independence to each latent term. These assumptions underline that it is less fruitful to assume a statistical structure that goes far beyond a Gaussian distribution when dealing with Latent Gaussian Models. As an example, we first focus on

a Gaussian Latent field using the same notation for Latent Gaussian Models 2. By model assumptions, we consider a latent field structure $\boldsymbol{x} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{Q})$, with $\boldsymbol{Q}$ having marginal variances equal to 1 while not depending on any hyperparameter, and $\boldsymbol{y}|\boldsymbol{x} \sim \prod_{i=1}^{n} \pi(y_i|x_i)$ with n data observations. We also assume $|\pi(y_i|x_i)| < \tilde{C}_i$ where each likelihood density is a function of x_i bounded by a constant $\tilde{C}_i$ which is unique for each observation. The posterior distribution of the corresponding latent model is

$$\pi(\boldsymbol{x}|\boldsymbol{y}) \propto \pi(\boldsymbol{x})\pi(\boldsymbol{y}|\boldsymbol{x}) \leq \pi(\boldsymbol{x})\tilde{C} \tag{4.1}$$

where $\tilde{C} = \prod_i \tilde{C}_i$. Since this Gaussian bound exists for the latent joint density, we can question if a similar bound is preserved for each latent marginal. We will show that

$$\pi(\boldsymbol{x}|\boldsymbol{y}) \leq \tilde{C}\pi(\boldsymbol{x}) \Rightarrow \pi(x_i|\boldsymbol{y}) \leq \tilde{C}\pi(x_i) \tag{4.2}$$

where $\pi(x_i)$ is the respective i^{th} Gaussian marginal density from its multivariate counterpart $\pi(\boldsymbol{x})$. The above statement provides a reasonable justification to using Gaussian assumptions onto the latent field of a Latent Gaussian Model structure. This marginal implication can be shown in few steps. We define functions $g_i(x_i) = \log(\pi(y_i|x_i))$ and write the latent joint conditional density as

$$\pi(\boldsymbol{x}|\boldsymbol{y}) = G \exp\Big(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \sum_{i=1}^{n} g_i(x_i)\Big) \tag{4.3}$$

where G is the normalization constant. Each posterior latent marginal x_i is obtained by integrating out all the other latent components $\boldsymbol{x}_{-i}$

$$\begin{aligned}\pi(x_i|\boldsymbol{y}) &= \int_{\boldsymbol{x}_{-i}} \pi(\boldsymbol{x}|\boldsymbol{y})\, d\boldsymbol{x}_{-i} \\ &= \exp(g_i(x_i)) \int_{\boldsymbol{x}_{-i}} G \exp\Big(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}\Big) \exp\Big(\sum_{j\neq i} g_j(x_j)\Big)\, d\boldsymbol{x}_{-i} \end{aligned} \tag{4.4}$$

Since each $g_i(x_i)$ is bounded by our initial assumptions, then

$$\begin{aligned}\pi(x_i|\boldsymbol{y}) &\leq \tilde{C}_i \int_{\boldsymbol{x}_{-i}} G \exp\Big(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}\Big) \prod_{j\neq i} \tilde{C}_j\, d\boldsymbol{x}_{-i} \\ &\leq \tilde{C} \exp\Big(-\frac{1}{2} x_i^2 (\boldsymbol{Q}^{-1})_{ii}^{-1}\Big) \end{aligned} \tag{4.5}$$

which corresponds to (4.2). The notation $(\boldsymbol{Q}^{-1})_{ii}$ refers to the i^{th} marginal variance term $\boldsymbol{\Sigma}_{ii}$ derived from the covariance matrix $\boldsymbol{\Sigma} = \boldsymbol{Q}^{-1}$. The result (4.5) shows that the Gaussian distribution represents a natural bound for each marginal up to a constant. Distributions with a Gaussian-like behavior are the most natural choice for approximating posterior marginals. Their tails must follow a Gaussian behavior while the main bulk of the distribution is free to differ from a Gaussian density because of location shift and skewness. The Gaussian and Simplified Laplace strategies outlined in Section 2.5 represent an appropriate embodiment of this Gaussian feature since their application provides accurate marginal posterior approximations in most of the cases by exploiting nearly Gaussian distributions. Later we show that Skew Normal family distributions are natural candidates as their tail behavior approximately resembles the one from a Gaussian distribution. The Gaussian family framework provides enough good properties for allowing an accurate representation of Latent Models, and INLA fully takes advantage of this pattern (see Gaussian Markov Random Fields (GMRFs) details and sparsity in Section 2.3).

4.1.2 A Student-t Latent Field

The Gaussian latent field example of the previous section shows that the Gaussian assumption provides similar properties to the posterior marginals of a Latent Gaussian Model. In this context, we question if the same argument applies when assuming more heavy-tailed distribution like the Student-t distribution. Although it shows non-normal behavior, we can still cast this distribution into a Gaussian hierarchical structure. Borrowing a result from the literature, we know that if $z \sim t_\nu$ then $z|\lambda \sim \mathrm{N}(0, \lambda^{-1})$ with $\lambda \sim G(\frac{\nu}{2}, \frac{\nu}{2})$. This construction is named *scale mixture of normals* with λ being an auxiliary variable. Both the auxiliary variable and Student-t information contribute to define the known *hierarchical t-formulation* which leads to well-defined conditional properties. Considering the example in Rue and Held (2005), Chapter 4, we assume a Student-t latent field $\boldsymbol{x} \sim t_a$ and model its independent increments by using a RW1 model while also conditioning on some auxiliary variables $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_{n-1})$ with n data observations $\boldsymbol{y}$. Each data point y_i is Gaussian distributed with mean x_i, meaning that $y_i \sim \mathrm{N}(x_i, \tau_y^{-1})$ with a certain precision τ_y. By scale mixture assumption the RW1 model

$$\Delta x_i|\lambda \sim \mathrm{N}(0, \tau_x^{-1}\lambda_i^{-1}), \quad i = 1, \ldots, n-1 \tag{4.6}$$

is Gaussian distributed with precision τ_x if each λ_i is distributed as $G(\frac{a}{2}, \frac{a}{2})$. Then the Gaussian assumption on the conditional latent field $\boldsymbol{x}$ leads to the conditional density

$$\begin{aligned}
\pi(\boldsymbol{x}|\boldsymbol{\lambda}, \tau_x) &\propto \tau_x^{\frac{n-1}{2}} \Big(\prod_{i=1}^{n-1} \lambda_i\Big)^{\frac{1}{2}} \exp\Big(-\frac{\tau_x}{2} \sum_{i=1}^{n-1} \lambda_i (\Delta x_i)^2\Big) \\
&\propto \tau_x^{\frac{n-1}{2}} \Big(\prod_{i=1}^{n-1} \lambda_i\Big)^{\frac{1}{2}} \exp\Big(-\frac{\tau_x}{2} \boldsymbol{x}^T \boldsymbol{Q}_{\boldsymbol{\Delta}} \boldsymbol{x}\Big)
\end{aligned} \tag{4.7}$$

where $\boldsymbol{Q_\Delta} = \boldsymbol{D_x}^T \boldsymbol{D_\lambda} \boldsymbol{D_x}$ with $\boldsymbol{D_\lambda}$ being a $(n-1) \times (n-1)$ diagonal matrix with $\text{diag}(\boldsymbol{D_\lambda}) = \boldsymbol{\lambda}$ and $\boldsymbol{D_x}$ being a $(n-1) \times n$ matrix as follows

$$\boldsymbol{D_x} = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{bmatrix} \tag{4.8}$$

Under the above construction, we assume our likelihood to be bounded by constants $\tilde{K}_1, \ldots, \tilde{K}_n$, and end up with the following joint posterior relation

$$\pi(\boldsymbol{x}, \boldsymbol{\lambda}|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\lambda})\pi(\boldsymbol{\lambda}) \leq \pi(\boldsymbol{x}|\boldsymbol{\lambda})\pi(\boldsymbol{\lambda})\tilde{K} \tag{4.9}$$

where $\tilde{K} = \prod_i \tilde{K}_i$ is an overall constant. Then the full conditional is bounded as

$$\pi(\boldsymbol{x}|\boldsymbol{\lambda}, \boldsymbol{y}) \leq \tilde{K}\pi(\boldsymbol{x}|\boldsymbol{\lambda}) \tag{4.10}$$

Similarly to the Gaussian latent field case, we obtain bounds for the corresponding marginals of this RW1 example as

$$\pi(x_i|\lambda_i, \boldsymbol{y}) \leq \tilde{K}_i \exp\Big(-\frac{\tau_x \lambda_i (\Delta x_i)^2}{2} (\boldsymbol{Q_\Delta}^{-1})_{ii}^{-1}\Big), \tag{4.11}$$

that are again represented by Gaussian distributions. The inequalities (4.10) and (4.11) show that we have control on all possible full conditional densities of the model as they still preserve GMRF properties and therefore can still be bounded by Gaussian densities. The same does not apply to the marginals $\pi(x_i|\boldsymbol{y})$ which are still bounded by t-Student distributions. Non-normal latent field or likelihood assumptions add complexity in approximating posterior marginals from these hierarchical structures. However, the mixture representation of marginal posterior densities (2.56) entirely depends on full conditionals as we integrate out all the hyperparameters. Therefore,

both INLA parametric and non-parametric strategies will still give accurate marginal results, provided (4.11) applies in a non Gaussian latent field. Although the example allows interpretable posterior inference for a Student-t latent field, it still appears impractical in a computational setting due to the high number of hyperparameters in the model specification, which is equal to n.

4.2 Marginal Inference with an extended Simplified strategy

Both marginal and joint inference is possible in INLA by constructing accurate approximations to their respective posterior target. Chapter 2 goes through the details behind INLA methodology and its approximation strategies, while Chapter 3 gives insights on how to extend the user's toolbox to make inferences on joint posterior distributions as well. The INLA strategies provide different approaches for approximating the posterior marginal distributions of Latent Gaussian Model parameters using parametric and non-parametric assumptions depending on the user's needs. When the Gaussian argument of Section 4.1 holds for a normal likelihood, the Gaussian Approximation strategy is the fastest and most accurate amongst all. Instead, if the posterior truth is heavily skewed and accuracy is an issue, the full Laplace strategy provides more on-point approximations in a non-parametric way. However, the best deal is given by the Simplified Laplace strategy, which offers faster but slightly more inaccurate approximations fitting Skew Normal densities. In most cases, the inaccuracies of this strategy compared to the full Laplace one are negligible. This section of the thesis proposes boosting the Simplified Laplace accuracy by employing Extended Skew Normal distributions. Section 4.2.2 also underlines that such Skew Normal family densities naturally satisfy the appealing Gaussian bounds and tail behavior for Latent Gaussian Models. The extension enables the strategy to produce more accurate posterior outcomes while avoiding any additional computational cost.

4.2.1 The Extended Skew Normal distribution and its properties

In more extreme settings, the Gaussian assumptions may not be enough to model the observed data properly. The true posterior marginal distributions can be heavily skewed, and the approximations computed by INLA may not accurately detect their pattern. The R-INLA software efficiently accomplishes this task by modeling the observed skewness through the third moment of a Skew Normal distribution which is adapted on a third-order expansion of the Laplace approximation. While Skew Normal distributions can approximate the marginals through the Simplified Laplace strategy (see Section 2.5.3), the respective joint posterior density can be approximated by a Skew Gaussian Copula (see Section 3.1). Concerning marginal inference in INLA, we will show that a further extension of the Simplified strategy can model even more skewed results. The idea is to employ a fourth-order expansion and fit an Extended Skew Normal distribution, which still belongs to the Skew Normal family and satisfies similar properties (see Azzalini and Capitanio (1999) for more distributions in the Skew Normal family). We introduce some basic definitions and properties of this extended version of the Skew Normal distribution. We define $T \sim \text{ESN}(\xi, \omega, \alpha, \tau)$ to be an Extended Skew Normal random variable where its probability density function is

$$f(t; \xi, \omega, \alpha, \tau) = \frac{1}{\omega \Phi(\tau)} \phi\Big(\frac{t-\xi}{\omega}\Big) \Phi\Big(\tau\sqrt{\alpha^2+1} + \alpha\frac{t-\xi}{\omega}\Big) \tag{4.12}$$

with location parameter ξ, scale ω, skewness parameter α and hidden mean parameter τ (or truncation parameter as mentioned in Canale (2011); Azzalini and Capitanio (2018)) while $\phi(\cdot)$, $\Phi(\cdot)$ are respectively the probability and cumulative density function of a standard Gaussian. If $\tau = 0$ then the equation in (4.12) reverts back to a Skew Normal distribution. Important is the cumulant generating function of T

defined as

$$K(u) = \log M(u) = \xi u + \frac{1}{2}\omega^2 u^2 + \mathcal{C}_0(\tau + \delta\omega u) - \mathcal{C}_0(\tau) \tag{4.13}$$

with $M(u) = \mathrm{E}[e^{uT}]$ being the moment generating function with parameterization $\delta = \frac{\alpha}{\sqrt{1+\alpha^2}}$ and $\mathcal{C}_0(z) = \log 2\Phi(z)$. Through the use of $K(t)$, it is straightforward to compute the first four moments

$$\begin{aligned}
\mathrm{E}(T) &= \xi + \mathcal{C}_1(\tau)\omega\delta \\
\mathrm{Var}(T) &= \omega^2[1 + \mathcal{C}_2(\tau)\delta^2] \\
\gamma_1(T) &= \frac{\mathcal{C}_3(\tau)\delta^3}{(1 + \mathcal{C}_2(\tau)\delta^2)^{3/2}} \\
\gamma_2(T) &= \frac{\mathcal{C}_4(\tau)\delta^4}{(1 + \mathcal{C}_2(\tau)\delta^2)^2}
\end{aligned} \tag{4.14}$$

with γ_1, γ_2 being the standardized skewness and kurtosis. The $\mathcal{C}(\cdot)$ functions provide a simpler formulation for the expressions in (4.14) in terms of τ. Indeed, Azzalini and Capitanio (2018) shows that

$$\mathcal{C}_r(\tau) = \frac{\partial^r}{\partial \tau^r} \log 2\Phi(\tau) \tag{4.15}$$

with the first derivatives up to order $r = 5$ being

$$\mathcal{C}_1(\tau) = \frac{\phi(\tau)}{\Phi(\tau)}$$
$$\mathcal{C}_2(\tau) = -[\mathcal{C}_1(\tau)]^2 - \tau\mathcal{C}_1(\tau)$$
$$\mathcal{C}_3(\tau) = -\tau\mathcal{C}_2(\tau) - 2\mathcal{C}_1(\tau)\mathcal{C}_2(\tau) - \mathcal{C}_1(\tau)$$
$$\mathcal{C}_4(\tau) = -\tau\mathcal{C}_3(\tau) - 2\mathcal{C}_2(\tau) - 2\mathcal{C}_2^2(\tau) - 2\mathcal{C}_1(\tau)\mathcal{C}_3(\tau)$$
$$\mathcal{C}_5(\tau) = -3\mathcal{C}_3(\tau) - \tau\mathcal{C}_4(\tau) - 6\mathcal{C}_2(\tau)\mathcal{C}_3(\tau) - 2\mathcal{C}_1(\tau)\mathcal{C}_4(\tau) \tag{4.16}$$

We recover Skew Normal constants when $\tau = 0$. From the $\mathcal{C}$ function formulation, we see that $\mathcal{C}_1(0) = \sqrt{\frac{2}{\pi}}$, $\mathcal{C}_2(0) = -\frac{2}{\pi}$, $\mathcal{C}_3(0) = \sqrt{\frac{2}{\pi}}\frac{(4-\pi)}{\pi}$ and $\mathcal{C}_4(0) = -\frac{24}{\pi^2} + \frac{8}{\pi}$ which appear in the respective Skew Normal moments. Additionally, we can observe interesting patterns of these $\mathcal{C}$ functions in terms of τ values. From Figure 4.1 we get the following

- $C_1(\tau)$ has a linear pattern for negative values and quickly decays to zero as τ approaches zero towards the positive range side
- $C_2(\tau)$ assumes values in the range $(-1, 0)$ and follows a logistic pattern
- $C_3(\tau)$ assumes values in the range $(0, 0.3)$ and resembles a proper probability density function
- $C_4(\tau)$ assumes values in the range $(-0.2, 0.1)$ and quickly decays to zero as $\tau < -1$ and $\tau > 4$

In particular, the function $\mathcal{C}_3$ approximately satisfies all the required properties of a probability density function, the range is positive and the respective integral is close to one. Numerical integration shows that the integral is 0.9991876 with absolute error less than 8.1e-05 for values of τ within the range [-35, 35]. This simplifies the required implementation of the Extended Skew Normal distribution into the Simplifed strategy

as we observe no additional benefit in considering large values of τ (see Section 4.2.5 for insights on this matter). Both parameter τ and the $\mathcal{C}$ function patterns make the Extended Skew Normal distribution appealing for better modeling skewed posterior outcomes when using the Simplified Laplace strategy.

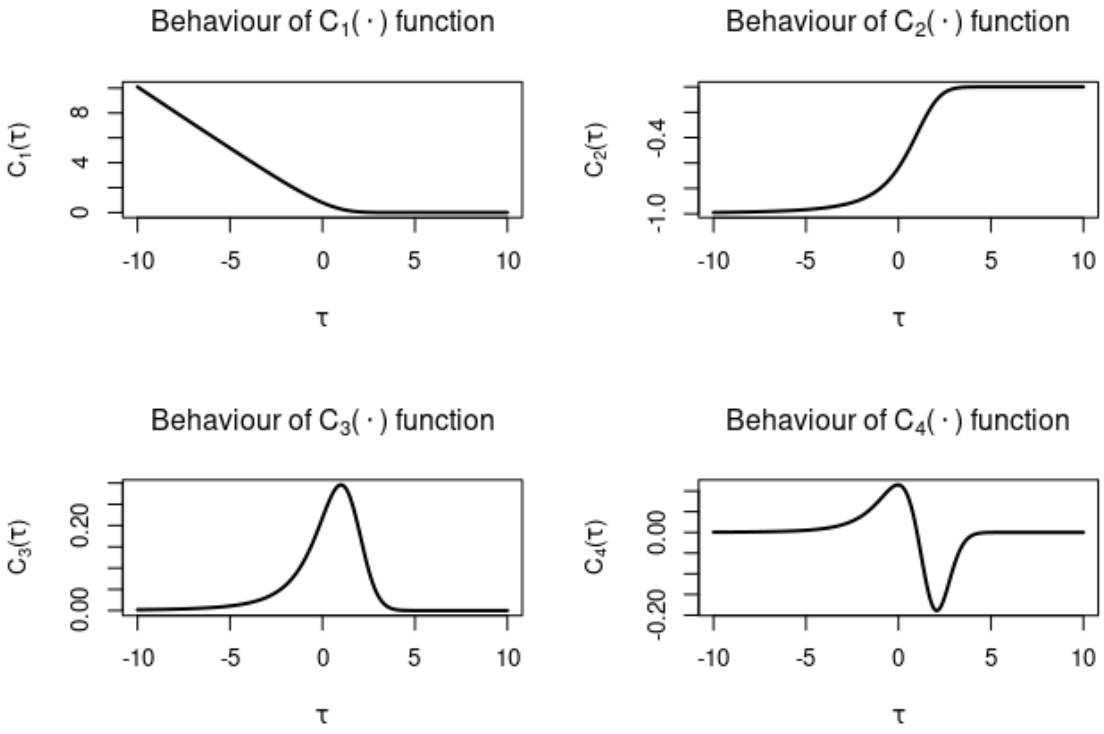


Figure 4.1: Plotting $\mathcal{C}$ functions of an Extended Skew Normal distribution up to order four with respect to the τ parameter with range values $[-10, 10]$.

On another note, we can also obtain a closed expression for the parameterization δ from (4.14) as follows

$$\delta = \text{sign}(\gamma_1)\sqrt{\frac{|\gamma_1|^{2/3}}{[\mathcal{C}_3(\tau)]^{2/3} - \mathcal{C}_2(\tau)|\gamma_1|^{2/3}}} \tag{4.17}$$

Moments can be used to construct accurate mappings to the parameters distribution, as we have seen for the Skew Normal density in Chapter 3. The same cannot be accomplished for the Extended Skew Normal distribution as there are two evident

issues:

- if we substitute equation (4.17) into the kurtosis one in (4.14), its resulting expression in terms of τ does not have a closed form solution
- the kurtosis is unbounded as its range is $[0, \infty)$ and this can lead to numerical issues or unreasonable outcomes as we do not have control on its range values

In the next sections we show that it is way easier and more efficient to follow a similar scheme adopted for the Simplified Laplace strategy by fitting Skew Normal distributions.

4.2.2 Tail behavior in the Skew Normal family densities

The Gaussian argument in Section 4.1 points out that a nearly Gaussian pattern can properly approximate the posterior marginals of a Latent Gaussian Model up to a constant. Skew Normal family densities are then appealing for the task since they naturally extend Gaussian densities by also modeling asymmetries. As the Simplified Laplace strategy in INLA provides accurate and fast approximations by using Skew Normal densities, we need to make sure that these distributions satisfy such Gaussian properties together with its extended version. Hence, we first consider log densities of a standard Skew Normal and Extended Skew Normal distribution

$$\begin{aligned}
\log f_{\text{SN}}(x;\alpha) &= \log(2) + \log(\phi(x)) + \log(\Phi(\alpha x)) \\
&= \log(2) + \log(\phi(x)) + \log\Big(\frac{1}{2} + \frac{1}{2}\operatorname{erf}\Big(\frac{\alpha x}{\sqrt{2}}\Big)\Big) \\
\log f_{\text{ESN}}(x;\alpha,\tau) &= -\log(\Phi(\tau)) + \log(\phi(x)) + \log(\Phi(\alpha x + \tau\sqrt{1+\alpha^2})) \\
&= -\log(\Phi(\tau)) + \log(\phi(x)) + \log\Big(\frac{1}{2} + \frac{1}{2}\operatorname{erf}\Big(\frac{\alpha x + \tau\sqrt{1+\alpha^2}}{\sqrt{2}}\Big)\Big)
\end{aligned} \tag{4.18}$$

with $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-z^2)\, dz$ being the error function. Although Gaussian boundaries are straightforward when considering that $\Phi(p) \leq 1$ for $p \in (-\infty, \infty)$, we need to check the tail behavior as well. Tails represent indeed an important aspect of the distribution as they model extreme observations. For accuracy purposes, we need to make sure that the Skew Normal family expressions in (4.18) show a similar tail Gaussian pattern. We can verify this aspect by computing series expansions of both log densities for the limiting cases $x \to \pm\infty$. An asymptotic expansion of the log Gaussian density $\phi(x)$ is straightforward and consists of one squared term. Skew Normal family densities add more complexity because of the $\Phi(\cdot)$ function term. Asymptotic expansion results for both Skew Normal family density tails are provided below, where we use $\nu = \tau\sqrt{1+\alpha^2}$. The results for the right tail are

$$\begin{aligned}
\log f_{\text{SN}}(x;\alpha)|_{x\to+\infty} &\approx -\frac{1}{2}x^2 + \exp\Big(-\frac{\alpha^2 x^2}{2}\Big)\Big(-\frac{1}{2}\frac{\sqrt{2}}{\alpha x\sqrt{\pi}} + \dots\Big) \\
\log f_{\text{ESN}}(x;\alpha,\tau)|_{x\to+\infty} &\approx -\frac{1}{2}x^2 + \exp\Big(-\frac{\alpha^2 x^2}{2} - \nu\alpha x\Big)\Big(-\frac{1}{2}\frac{\sqrt{2}\exp(-\frac{1}{2}\nu^2)}{\sqrt{\pi}\alpha x} + \dots\Big)
\end{aligned} \tag{4.19}$$

while for the left tail we have

$$\begin{aligned}
\log f_{\text{SN}}(x;\alpha)|_{x\to-\infty} &\approx -\frac{1}{2}x^2(1+\alpha^2) + \log\Big(-\frac{1}{\alpha x\sqrt{2\pi}}\Big) + \dots \\
\log f_{\text{ESN}}(x;\alpha,\tau)|_{x\to-\infty} &\approx -\frac{1}{2}(\alpha^2+1)x^2 + \nu x + \log\Big(-\frac{1}{2}\frac{\sqrt{2}\exp(-\frac{1}{2}\nu^2)}{\alpha x\sqrt{\pi}}\Big) + \dots
\end{aligned} \tag{4.20}$$

The expanded results in (4.19) show a sequence of higher-order terms that quickly approach zero as $x \to +\infty$. The right tail of both Skew Normal and Extended Skew Normal density gets more and more similar to the one expected from a Gaussian

distribution. Corresponding left tail results (4.20) for $x \to -\infty$ show a similar Gaussian pattern but with a slower decay. We can also recognize a log Gaussian density contribution with additional logarithmic terms coming from the expanded cumulative density $\Phi(\alpha x)$. Skew Normal family densities appear to be a natural choice for approximating Latent Gaussian posterior marginals as accurately as possible when outcomes slightly deviate from a Gaussian pattern.

4.2.3 Expanding the target posterior up to third order

This section offers a step by step description of the Simplified Laplace strategy, initially introduced in Chapter 2. First, we focus on the default approach, which constructs Skew Normal approximations based on the target distribution's third-order Taylor expansion. While going through the methodology, we also derive and apply a new way to compute an exact mode for the Skew Normal density by avoiding alternative approximations (see also Wood (2020)). Then we move on to the details of our new proposed extension using Extended Skew Normal distributions and how to get solutions efficiently. The default strategy consists of fitting a Skew Normal distribution to a third order Taylor expanded density of the form

$$\log(\pi(z)) = K - \frac{1}{2}z^2 + \tilde{\mu}z + \frac{1}{3!}\tilde{\gamma}_1 z^3 + \dots \tag{4.21}$$

where K is a constant, $(\tilde{\mu}, \tilde{\gamma}_1)$ are terms derived from the third order Taylor expansion of the Laplace Approximation evaluated at each Gaussian Approximation mean. The resulting density in (4.21) is $N(\tilde{\mu}, 1)$ up to second order while the third term $\tilde{\gamma}_1$ provides information of the third order derivative evaluated at the mode. We consider $R \sim \text{SN}(\xi, \omega, \alpha)$ with unknown location ξ, scale ω and skewness parameter α. Then we define a system of three equations to compute the respective parameter triplet $(\tilde{\xi}, \tilde{\omega}, \tilde{\alpha})$ to approximate the target density in (4.21). By matching the first two non central moments and the third derivative of the Skew Normal at the mode z^*, the

resulting system is

$$\begin{aligned} &\mathrm{E}(R) = \tilde{\mu} \\ &\mathrm{Var}(R) = 1 \\ &\frac{\partial^3}{\partial r^3} \log \pi(r; \xi, \omega, \alpha)\Big|_{r=z^*} = \tilde{\gamma}_1 \end{aligned} \tag{4.22}$$

However the mode z^* is not analytically available. From Appendix B in Rue et al. (2009), we can expand $\log \pi(r; \xi, \omega, \alpha)$ at its location point $r = \xi$ to compute an approximation to the mode as

$$r^* = \Big(\frac{\alpha}{\omega}\Big) \frac{\sqrt{2\pi} + 2\xi\big(\frac{\alpha}{\omega}\big)}{\pi + 2\big(\frac{\alpha}{\omega}\big)^2} \tag{4.23}$$

We evaluate the third derivative of the log Skew Normal density at the approximated mode (4.23). To allow an exact analytical result and fast computations, we consider a $\mathcal{C}$ function form of the third log derivative as $\frac{\partial^3}{\partial r^3} \log \pi(r; \xi, \omega, \alpha) = \mathcal{C}_3\Big(\frac{\alpha}{\omega}(r - \xi)\Big)\Big(\frac{\alpha}{\omega}\Big)^3$ and expand this expression at $\frac{\alpha}{\omega}$ around $\alpha = 0$. Only the third order term of the expansion is not zero and equal to $6\mathcal{C}_3(0)$. This results is then used to get the last equation of the system. But we can also compute the Skew Normal modal configuration through an interpolant between skewness and third log derivative results almost exactly. Figure 4.2 shows that the interpolation curve of these quantities is smooth and provides more precise results for the Skew Normal parameters. In most cases, we do not detect significant improvements but the new approach still makes the Simplified Laplace approximations slightly more accurate when non-negligible skewness is involved. Additionally, it simplifies the default INLA methodology avoiding computations for solving the system of equations.

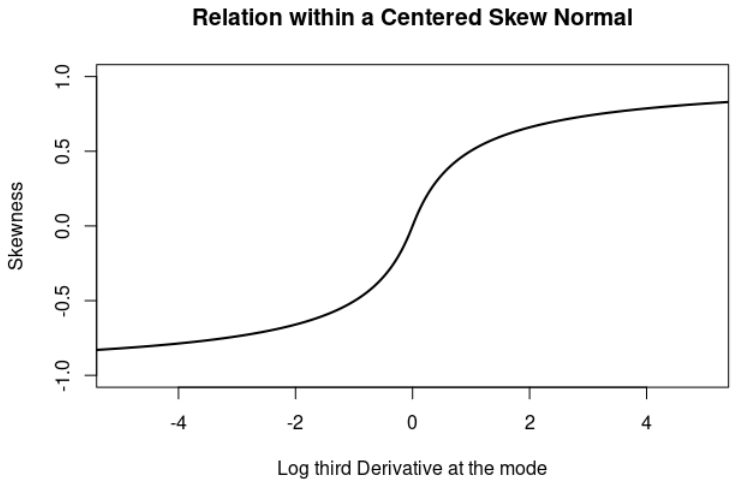


Figure 4.2: The curve describes the exact relation of skewness and log third derivative evaluated at the exact mode of a standard Skew Normal random variable for many possible values of skewness in the range (-1,1). The modes are computed by numerical optimization for maximum accuracy purposes.

Then we obtain the third equation in the system (4.22) as

$$\tilde{\gamma}_1 = \mathcal{C}_3(0)\Big(\frac{\alpha}{\omega}\Big)^3 \tag{4.24}$$

where the right side is exactly the resulting polynomial expansion of $\frac{\partial^3}{\partial r^3}\log\pi(r;\xi,\omega,\alpha)\Big|_{r=r^*}$ with $\mathcal{C}_3(\cdot)$ being the $\mathcal{C}$ function formulation derived from the Extended Skew Normal distribution. Using equation (4.24), we can directly solve the system (4.22) since α is a function of the sole scale parameter ω with $\mathcal{C}_3(0)$ being a constant (≈ 0.218). The solutions we get from the system are unique and exact and lead to fast Skew Normal approximations of the target distribution.

4.2.4 Expanding the target posterior up to fourth order

The target density in (4.21) revolves around a third order expansion but more terms can be considered. We can indeed extend the Simplified Laplace methodology by fitting an Extended Skew Normal distribution, introduced in Section 4.2.1, to a fourth

order expansion of the same target distribution. Similarly, we will need to solve a system of four equations since the new distribution has an additional parameter τ. The corresponding log density of (4.12) can be written in a $\mathcal{C}$ function formulation as

$$\log f(t;\xi,\omega,\alpha,\tau) = \log\Big[\frac{1}{\omega}\phi\Big(\frac{t-\xi}{\omega}\Big)\Big] + \mathcal{C}_0\Big(\tau\sqrt{1+\alpha^2} + \alpha\frac{t-\xi}{\omega}\Big) - \mathcal{C}_0(\tau) \tag{4.25}$$

If $\tau = 0$ the extended log density in (4.25) degenerates into a Skew Normal one. Moreover, the role of the hidden mean parameter becomes irrelevant when $\alpha = 0$ as the density reverts back to a Gaussian distribution with mean ξ and variance ω^2. From Seijas-Macias et al. (2017) and Azzalini and Capitanio (2018), we know that τ affects both skewness and kurtosis of the distribution when α is not zero. For this case we need log derivatives up to order four for the system

$$\begin{aligned}
\frac{\partial}{\partial t}\log f(t;\xi,\omega,\alpha,\tau) &= -\frac{t-\xi}{\omega^2} + \mathcal{C}_1\Big(\tau\sqrt{1+\alpha^2} + \frac{\alpha}{\omega}(t-\xi)\Big)\frac{\alpha}{\omega}\\
\frac{\partial^2}{\partial t^2}\log f(t;\xi,\omega,\alpha,\tau) &= -\frac{1}{\omega^2} + \mathcal{C}_2\Big(\tau\sqrt{1+\alpha^2} + \frac{\alpha}{\omega}(t-\xi)\Big)\Big(\frac{\alpha}{\omega}\Big)^2\\
\frac{\partial^3}{\partial t^3}\log f(t;\xi,\omega,\alpha,\tau) &= \mathcal{C}_3\Big(\tau\sqrt{1+\alpha^2} + \frac{\alpha}{\omega}(t-\xi)\Big)\Big(\frac{\alpha}{\omega}\Big)^3\\
\frac{\partial^4}{\partial t^4}\log f(t;\xi,\omega,\alpha,\tau) &= \mathcal{C}_4\Big(\tau\sqrt{1+\alpha^2} + \frac{\alpha}{\omega}(t-\xi)\Big)\Big(\frac{\alpha}{\omega}\Big)^4
\end{aligned} \tag{4.26}$$

Similar to the Skew Normal case, we do not have an analytical solution for the mode due to the intractable structure of the first log derivative in (4.26). Therefore we must rely on an expansion of the third log derivative at $t = \xi$ getting the new approximated mode

$$t^* = \Big(\frac{\alpha}{\omega}\Big)\frac{\mathcal{C}_1(\tau\sqrt{1+\alpha^2}) - \mathcal{C}_2(\tau\sqrt{1+\alpha^2})\xi\Big(\frac{\alpha}{\omega}\Big)}{1 - \mathcal{C}_2(\tau\sqrt{1+\alpha^2})\Big(\frac{\alpha}{\omega}\Big)^2} \tag{4.27}$$

which reverts back to (4.23) as $\tau = 0$. For this case, the interpolation shown in Figure 4.2 is challenging as we now need to match skewness with two free parameters. Therefore we decide to apply the modal approximation above which is still numerically accurate. Another existing numerical approximation for the mode is provided in Azzalini and Capitanio (2018) by using the centralized moments of Skew Normal family densities. Next we expand the third and fourth log derivatives of the Extended Skew Normal distribution at the mode (4.27) with respect to $\frac{\alpha}{\omega}$ around $\alpha = 0$. The results we get are the following

$$\begin{aligned} \frac{\partial^3}{\partial t^3} \log f(t; \xi, \omega, \alpha, \tau)\Big|_{t=t^*} &\approx \mathcal{C}_3(\tau)\Big(\frac{\alpha}{\omega}\Big)^3 \\ \frac{\partial^4}{\partial t^4} \log f(t; \xi, \omega, \alpha, \tau)\Big|_{t=t^*} &\approx \mathcal{C}_4(\tau)\Big(\frac{\alpha}{\omega}\Big)^4 \end{aligned} \tag{4.28}$$

that are available as functions of the scale parameter ω, the skewness parameter α and the hidden mean parameter τ. The final system of equations is obtained by matching the first two moments of the Extended Skew Normal random variable and its higher-order expanded log derivatives in (4.28) as follows

$$\begin{aligned} \xi + \omega\delta\mathcal{C}_1(\tau) &= \tilde{\mu} \\ \omega^2(1 + \mathcal{C}_2(\tau)\delta^2) &= 1 \\ \mathcal{C}_3(\tau)\Big(\frac{\alpha}{\omega}\Big)^3 &= \tilde{\gamma}_1 \\ \mathcal{C}_4(\tau)\Big(\frac{\alpha}{\omega}\Big)^4 &= \tilde{\gamma}_2 \end{aligned} \tag{4.29}$$

with $(\tilde{\gamma}_1, \tilde{\gamma}_2)$ being the third and fourth log derivatives evaluated at the mode derived from the target approximated posterior in (4.21). Finally we compute the solutions of the respective parameters by solving the system (4.29). However, we do not have a

straightforward access to a unique solution due to the two last non linear equations in τ. For the sake of efficiency, we construct an interpolant on τ to match its solutions with the equations in a reasonable range.

4.2.5 Computing τ solutions by interpolation

The Extended Skew Normal distribution can be another natural parametric choice for computing posterior marginals through the Simplified Laplace approach in INLA. The resulting system of equations (4.21) is hard to solve exactly. Therefore, we choose to work on a single expression depending on τ and construct an interpolant of its solutions. If we combine the last two equations of the system, we get the following

$$\frac{\tilde{\gamma}_2}{[\tilde{\gamma}_1]^{4/3}} = \frac{\mathcal{C}_4(\tau)}{[\mathcal{C}_3(\tau)]^{4/3}} \tag{4.30}$$

which heavily depends on high-order $\mathcal{C}$ functions. However we can see from Figure 4.3 that there exists a smooth behavior between the τ values and respective solutions of the ratio appearing on the right side of equation (4.30). Such one-to-one relation can be encoded into an interpolant without relying on more costly non-linear solvers (see `nleqslv` R package, for example).

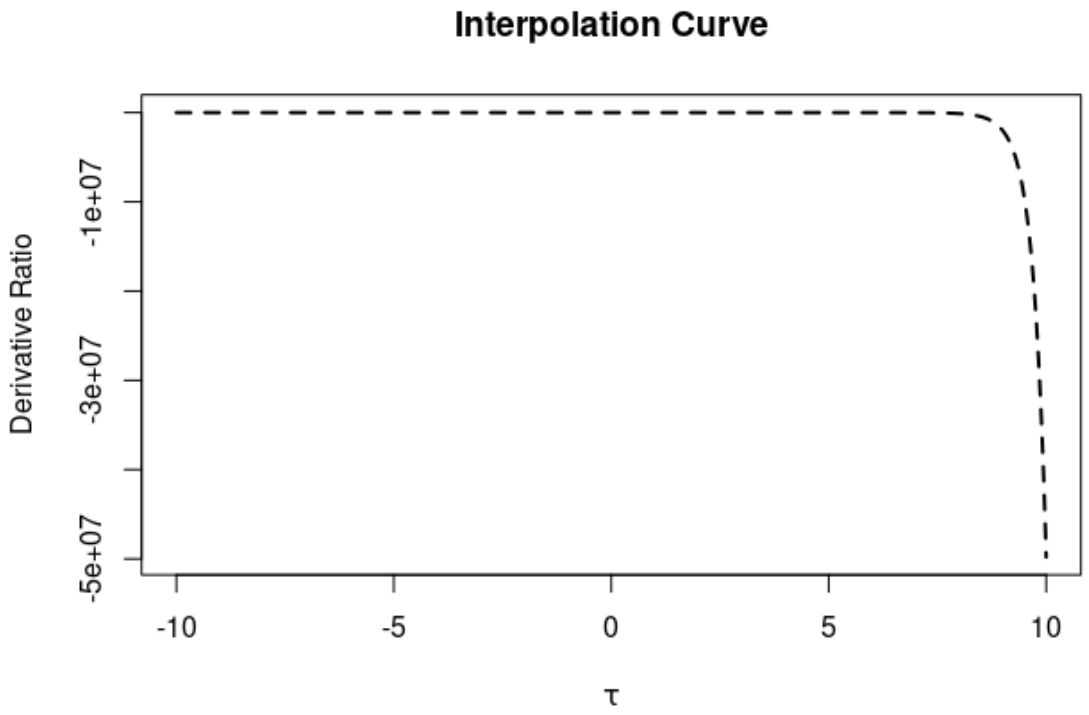


Figure 4.3: Relationship between the hidden mean parameter τ and the $\mathcal{C}$ function derivative ratio $\frac{\mathcal{C}_4(\tau)}{[\mathcal{C}_3(\tau)]^{4/3}}$ obtained from 4.30.

The interpolant ensures accurate and fast solutions within reasonable boundaries for τ. The derivative ratio is positively bounded from above as follows

$$-\infty < \frac{\mathcal{C}_4(\tau)}{[\mathcal{C}_3(\tau)]^{4/3}} < 2.4 \quad \text{with} \quad -\infty < \tau < \infty \tag{4.31}$$

which again underlines the Gaussian convergence bounds of the Extended Skew Normal distribution when $\tau \to \pm\infty$. In particular, Canale (2011) shows that these limiting Gaussian cases are $N(\xi, \omega^2)$ for $\tau \to \infty$ and $N(-\alpha|\tau|, \frac{1}{\sqrt{1-\delta^2}})$ for $\tau \to -\infty$. From Figure 4.1 we have seen that $\mathcal{C}_3(\tau)$ can approximately recover a probability density function with respect to the parameter τ. This pattern is useful for constructing a criterion that only picks reasonable solutions of τ with respect to the log derivative outcomes $(\tilde{\gamma}_1, \tilde{\gamma}_2)$. As a rule of thumb, we establish that a $|\tau| > 10$ value is already

far extreme and can lead to unlikely or unstable results produced by the interpolant. Such range leads to a probability space coverage equal to $\int_{-10}^{10} \mathcal{C}_3(\tau)\, d\tau \approx 0.99$ which is wide enough for our purposes. Moreover, a low value of $\tilde{\gamma}_1$ results in an unreasonable ratio outcome of 4.31 for the corresponding interpolant. When $\tilde{\gamma}_1$ approaches zero, the Extended Skew Normal density bends to a Gaussian one and the new approach gets unstable. All possible instabilities that can arise from an extreme solution of the system or an inaccurate interpolation are resolved by reverting back to the original Simplified Laplace approach in Section 4.2.3. In general, the resulting interpolant is accurate and does not add computational costs to the strategy. By solving the derivative ratio in (4.30), we can finally get solutions for the system of equations. Assuming $\tilde{\gamma}_1$ is not zero, we write $a^* = \mathcal{C}_3(\tilde{\tau})$ where $\tilde{\tau}$ is the result obtained by the interpolant as a solution of (4.30). We write the skewness parameter as $\tilde{\alpha} = \tilde{\omega} b^*$ with $b^* = (\frac{\tilde{\gamma}_1}{a^*})^{1/3}$ and get

$$\tilde{\omega} = \sqrt{\frac{-d^* + \sqrt{(d^*)^2 + 4c^*}}{2c^*}} \tag{4.32}$$

where $c^* = (b^*)^2(1 + \mathcal{C}_2(\tilde{\tau}))$ and $d^* = 1 - (b^*)^2$. If $\tilde{\tau}$ approaches 0 then we revert to a Skew Normal system of equations. Here we know that the location $\tilde{\xi}$ is given by

$$\tilde{\xi} = \tilde{\mu} - \tilde{\omega}\tilde{\delta}\mathcal{C}_1(\tilde{\tau}) \tag{4.33}$$

where $\tilde{\delta} = \frac{\tilde{\alpha}}{\sqrt{1+\tilde{\alpha}^2}}$. The last expression (4.33) gives the respective location parameter solution for the Extended Skew Normal system.

4.3 Posterior analysis using the Simplified Laplace strategy

Marginal posterior inference is particularly fast and accurate for hierarchical models within the class of Latent Gaussian Models when INLA is used. Section 4.1 discusses how Gaussian assumptions on the latent field positively affect the computed approx-

imations by forcing the posterior marginal densities to be accurately represented by Gaussian-like distributions. Both Simplified Laplace and full Laplace strategies are up to the task, with the latter being more accurate at the cost of some speed performance. In Ruli and Ventura (2016); Ruli et al. (2016) we can find more improvements to these approximations depending on the modeling application. A new direction for improving the Laplace Approximation sees the use of Variational Bayes corrections to the posterior mean (see van Niekerk and Rue (2021)). Section 4.2 proposes a new extension for the Simplified Laplace strategy to improve the accuracy of the results when the posterior marginals are heavily skewed. Instead of relying on Skew Normal approximations, we use Extended Skew Normal distributions, which still belong to the Skew Normal family and satisfy similar properties. Since this new strategy shines in contexts where the posterior marginals of the model are skewed, we again consider hierarchical formulations for Poisson and Binomial likelihood models by simulating datasets with different sample sizes and one single covariate with Gaussian prior to account for high marginal skewness. We then compare the posterior marginal results from each model by using the strategies in INLA and the Markov Chain Monte Carlo approach in JAGS (Plummer et al. (2003)). More in detail, the INLA strategies are the Simplified (SLA), the full Laplace (LA), and the Extended Simplified Laplace (ESLA).

4.3.1 Simulation Results

Here we show simulations for both Binomial and Poisson likelihoods focusing on different sample size dimensions denoted by n. This simulation setting allows to get skewed posterior marginals and shows less extreme patterns as soon as the sample size increases. For large values of n, we expect to observe a more prevalent Gaussian behavior in the posterior outcomes. The results for the two model scenarios are reported in the plots below with increasing order of n. Marginal outcomes of the

Binomial setting are in Figure 4.4, 4.5, 4.6 and 4.7 while the Poisson ones are shown in Figure 4.8, 4.9, 4.10 and 4.11. For low sample size n, we see that the ESLA strategy provides more accurate results around the mode. The full Laplace (LA) and MCMC methods report the most accurate results and do not differ in practice. ESLA posterior results appear closer to LA and MCMC than SLA strategy, where the mode is far off the more correct location. All employed strategies tend to show similar results as soon as the sample size n gets larger, with ESLA being slightly more accurate. A summary of the posterior modal configurations for different sample sizes is given on Tables 4.1 and 4.3, while interquartile ranges (IQR) are reported in Table 4.2 and 4.4. These simulations underline that ESLA strategy is preferable in more extreme settings where the skewness is high, especially around the mode. The extended methodology also preserves robustness as it is forced to revert to a standard Simplified strategy in less extreme cases.

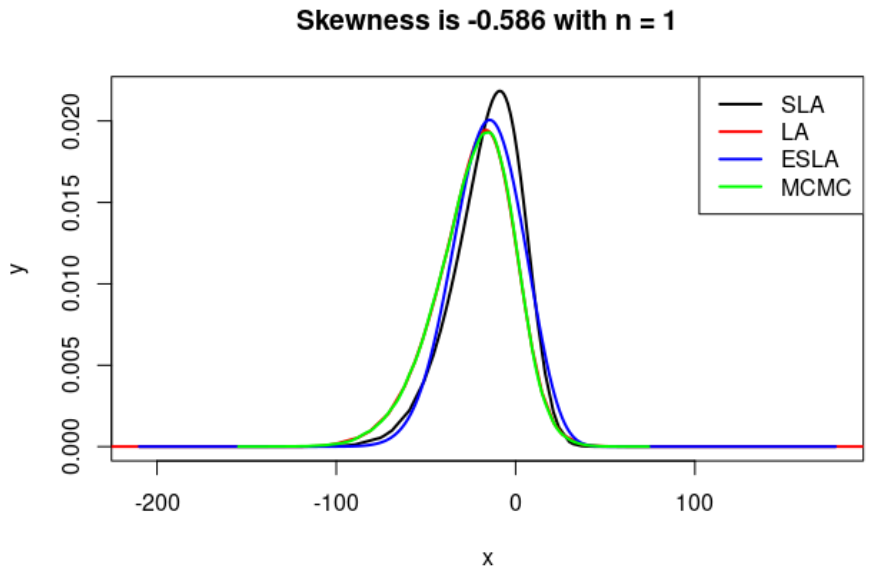


Figure 4.4: Comparative results between SLA (black line), LA (red line), ESLA (blue line) and MCMC (green line) strategies with $n = 1$ observations and a Bernoulli likelihood. Extreme negative skewness setting with minimum sample size. Since LA and MCMC strategies embody the posterior truth, we can observe that the SLA approach shows way less accuracy around the mode than its extended version denoted by ESLA. Tail behavior is similar for both SLA and ESLA and still appears to be slightly inaccurate in the left direction.

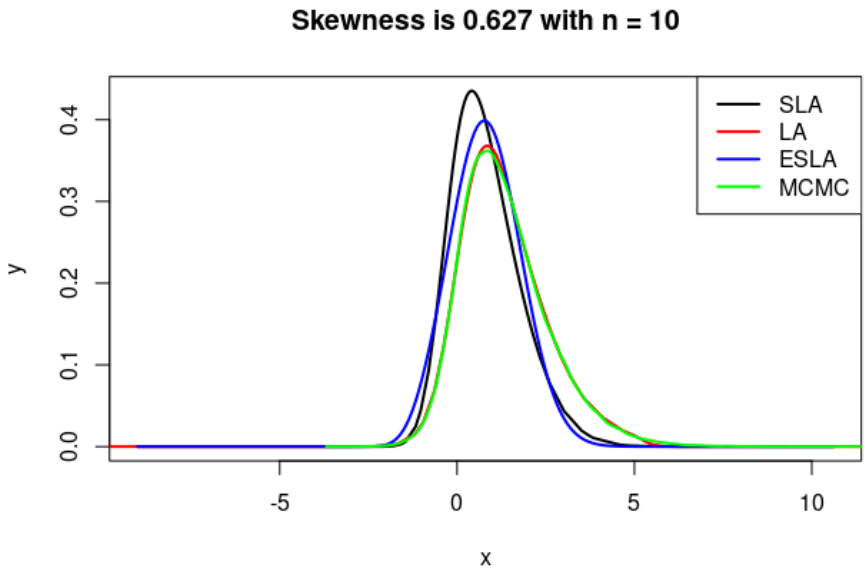


Figure 4.5: Comparative results between SLA (black line), LA (red line), ESLA (blue line) and MCMC (green line) strategies with $n = 10$ observations and a Bernoulli likelihood. Extreme positive skewness setting with small sample size. Since LA and MCMC strategies embody the posterior truth, we can see that the SLA approach shows way less accuracy around the mode than its extended version ESLA. Tail behavior is similar for both SLA and ESLA and still appears to be moderately inaccurate in the right direction.

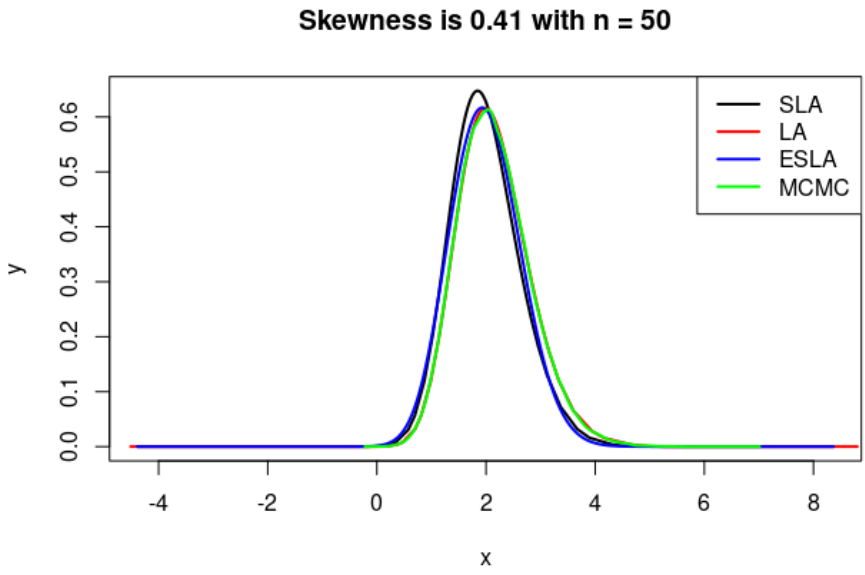


Figure 4.6: Comparative results between SLA (black line), LA (red line), ESLA (blue line) and MCMC (green line) strategies with $n = 50$ observations and a Bernoulli likelihood. Extreme positive skewness setting with moderate sample size. All employed strategies for this application show similar results except for the SLA methodology, which appears to be more inaccurate around the mode. Still, both SLA and ESLA suffer minor deviations in the right tail compared to LA and MCMC truth.

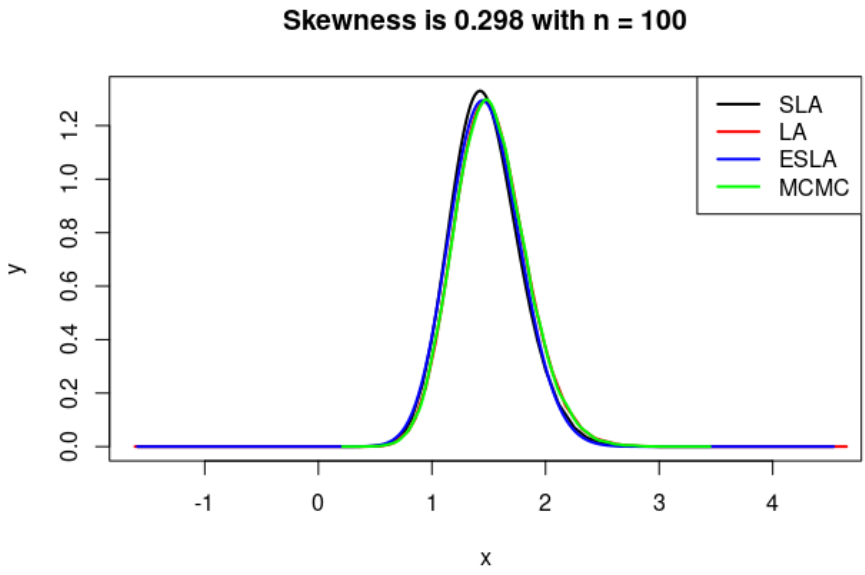


Figure 4.7: Comparative results between SLA (black line), LA (red line), ESLA (blue line) and MCMC (green line) strategies with $n = 100$ observations and a Bernoulli likelihood. High positive skewness setting with enough large sample size. All employed strategies for this application show similar results with minor deviations around the mode given by the SLA methodology. Large sample sizes tend to provide more stable expected results no matter the approximation strategy we use. Still, ESLA strategy is much closer to the true posterior results than SLA.

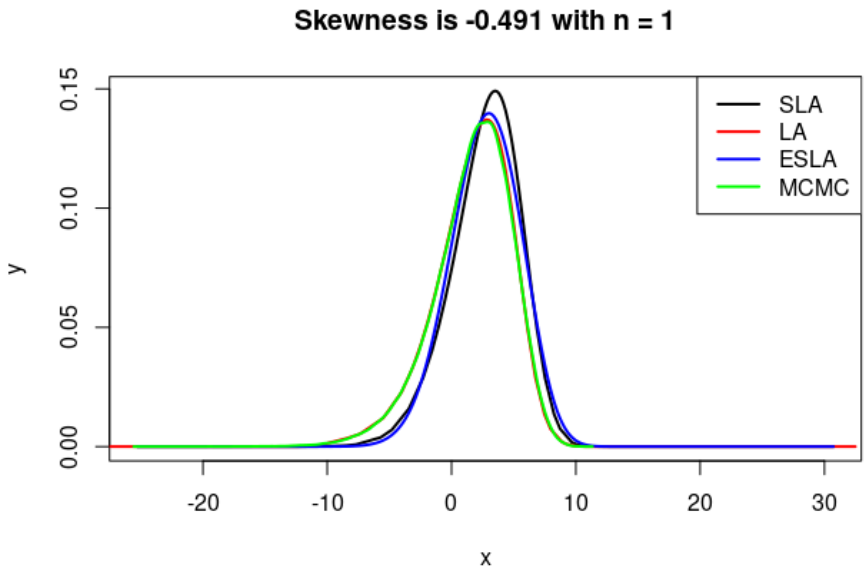


Figure 4.8: Comparative results between SLA (black line), LA (red line), ESLA (blue line) and MCMC (green line) strategies with $n = 1$ observations and a Poisson likelihood. Extreme negative skewness setting with minimum sample size. Since LA and MCMC strategies embody the posterior truth, we can observe that the SLA approach shows way less accuracy around the mode than its extended version denoted by ESLA. Unlike the Binomial case, tail behaviors for both SLA and ESLA closely match with no evident differences.

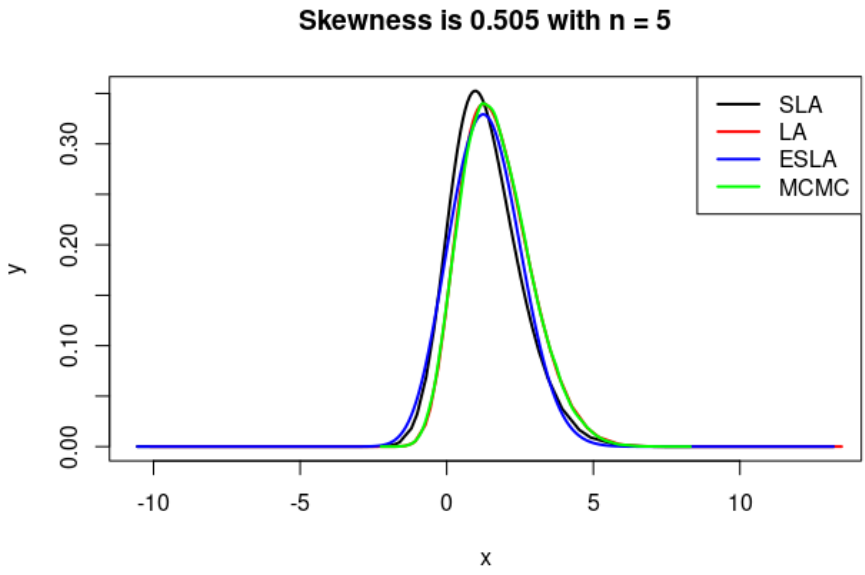


Figure 4.9: Comparative results between SLA (black line), LA (red line), ESLA (blue line) and MCMC (green line) strategies with $n = 5$ observations and a Poisson likelihood. Extreme positive skewness setting with small sample size. Since LA and MCMC strategies embody the posterior truth, we can see that the SLA approach shows way less accuracy around the mode than its extended version ESLA. Unlike the Binomial case, tail behaviors for both SLA and ESLA closely match with no evident differences.

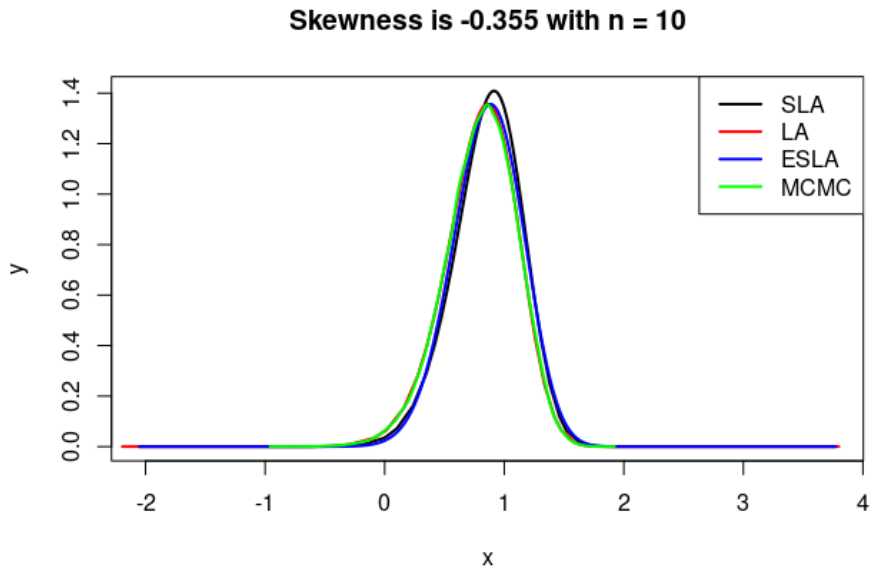


Figure 4.10: Comparative results between SLA (black line), LA (red line), ESLA (blue line) and MCMC (green line) strategies with $n = 10$ observations and a Poisson likelihood. High negative skewness setting with small sample size. All employed strategies for this application show similar results except for the SLA methodology, which appears to be more inaccurate around the mode. Still, both SLA and ESLA suffer minor deviations in the left tail compared to LA and MCMC truth.

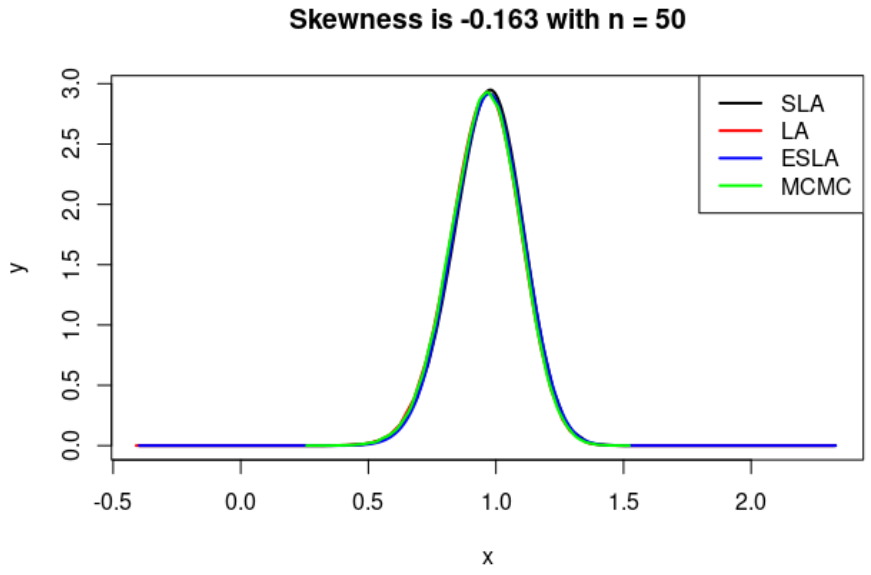


Figure 4.11: Comparative results between SLA (black line), LA (red line), ESLA (blue line) and MCMC (green line) strategies with $n = 50$ observations and a Bernoulli likelihood. Moderate negative skewness setting with enough large sample size. All employed strategies for this application closely converge to the same posterior result with no evident difference. Large sample sizes tend to provide more stable expected results no matter the approximation strategy we use.

Table 4.1: Binomial simulations for increasing sample sizes up to $n = 100$ and posterior mode evaluations using SLA, ESLA, LA and MCMC strategies. For low sample sizes, the modes derived from ESLA strategy are closer to the true ones from LA and MCMC approaches than the default SLA strategy. As the sample size n increases, we notice a decreasing pattern for the positive skewness sequence (apart from $n = 1$), with the mode values converging to the same result for all strategies. Overall, ESLA provides more coherent results to LA and MCMC, confidently representing the truth.

n	**Skew**	**Mode(SLA)**	**Mode(ESLA)**	**Mode(LA)**	**Mode(MCMC)**
1	-0.578	-8.979	-14.528	-16.581	-17.249
2	0.644	0.346	0.783	1.084	0.995
5	0.627	1.17	1.764	1.844	1.914
10	0.495	1.207	1.39	1.459	1.363
20	0.451	0.639	0.722	0.784	0.764
50	0.306	0.908	0.934	0.964	0.94
100	0.218	0.85	0.862	0.881	0.876

Table 4.2: Binomial simulations for increasing sample sizes up to $n = 100$ and posterior interquartile range (IQR) evaluations using SLA, ESLA, LA and MCMC strategies. The IQRs from both SLA and ESLA strategies get closer and closer to the truth provided by LA and MCMC posterior results as soon as the sample size increases. Although the difference is less relevant than the one from the respective mode in Table 4.1, ESLA grants more accurate results towards the truth than its simpler version SLA.

n	**Skew**	**IQR(SLA)**	**IQR(ESLA)**	**IQR(LA)**	**IQR(MCMC)**
1	-0.578	25.865	26.909	28.73	28.949
2	0.644	2.046	2.138	2.838	3.012
5	0.627	2.316	2.4	2.751	2.80
10	0.495	1.189	1.232	1.313	1.31
20	0.451	0.755	0.78	0.813	0.816
50	0.306	0.468	0.477	0.483	0.483
100	0.218	0.365	0.37	0.372	0.372

Table 4.3: Poisson simulations for increasing sample sizes up to $n = 100$ and posterior mode evaluations using SLA, ESLA, LA and MCMC strategies. For low sample sizes, the modes derived from ESLA strategy are closer to the true ones from LA and MCMC approaches than the default SLA strategy. As the sample size n increases, we notice a decreasing pattern for the negative skewness sequence (apart from $n = 2$), with the mode values converging to the same result for all strategies. Overall, ESLA provides more coherent results to LA and MCMC, confidently representing the truth.

n	**Skew**	**Mode(SLA)**	**Mode(ESLA)**	**Mode(LA)**	**Mode(MCMC)**
1	-0.446	0.972	0.905	0.87	0.886
2	0.496	-2.696	-2.195	-2.06	-1.87
5	-0.322	1.882	1.85	1.822	1.814
10	-0.311	0.796	0.78	0.767	0.763
20	-0.223	0.992	0.983	0.973	0.969
50	-0.179	1.109	1.106	1.103	1.108
100	-0.113	1.033	1.032	1.03	1.026

Table 4.4: Poisson simulations for increasing sample sizes up to $n = 100$ and posterior interquartile range (IQR) evaluations using SLA, ESLA, LA and MCMC strategies. The IQRs from both SLA and ESLA strategies get closer and closer to the truth provided by LA and MCMC posterior results as soon as the sample size increases. Although the difference is less relevant than the one from the respective mode in Table 4.3, ESLA grants more accurate results towards the truth than its simpler version SLA.

n	**Skew**	**IQR(SLA)**	**IQR(ESLA)**	**IQR(LA)**	**IQR(MCMC)**
1	-0.446	0.598	0.618	0.64	0.644
2	0.496	3.588	3.717	3.936	3.92
5	-0.322	0.492	0.5	0.5	0.5
10	-0.311	0.251	0.256	0.256	0.257
20	-0.223	0.224	0.227	0.227	0.227
50	-0.179	0.092	0.093	0.093	0.093
100	-0.113	0.082	0.083	0.083	0.083

4.4 Discussion

This last chapter of the thesis discussed how we could further extend the available Simplified Laplace strategy in INLA by considering more higher-order derivatives in the expansion and using a suitable parametric fit. Extended Skew Normal densities appeared to enhance the approximation strategy by modeling observed skewed outcomes more accurately than using Skew Normal distributions. As part of the

Skew Normal family, these extended distributions also satisfied the Gaussian bounds and tail properties discussed throughout Section 4.1. However, their four-parameter mathematical formulation was hard to handle as we could not achieve exact analytical solutions, unlike the Skew Normal setting. We solved this issue by constructing an interpolant for the hidden mean parameter solutions τ. To keep computations stable, we decided to bound the range of acceptable solutions with a well-defined rule of thumb. The whole approach was still computationally competitive towards its default implemented counterpart in the strategy while providing more accurate results around the mode. The new extended approach is flexible as it can quickly reproduce default method outcomes when inaccuracies or instabilities happen Canale (2011); Azzalini and Capitanio (2018). It also showed encouraging improvements in the simulation results consistent with the ones produced by the full Laplace strategy, which opens a new path of possibilities. Alternative skewed family distributions may be used to achieve even more accurate results to avoid using more costly strategies. Throughout this project, we also questioned if it could have been possible to encode the Extended Skew Normal distribution into the Skew Gaussian Copula class of Chapter 3 to improve the joint posterior inference as well. In Section 4.2.1 we already introduced a first reason why this could not be feasible in practice due to the lack of kurtosis bounds and mapping difficulties. This work has shown that Latent Gaussian assumptions allow us to easily extend INLA strategies as they do not impose too strict restrictions on the nature of approximations we can use. Parametric approximations are appealing from a computational perspective, but they are more likely to fail in recovering an exact truth in more extreme settings.

Chapter 5

Concluding Remarks

5.1 Summary

Marginal posterior inference broadly represents the main focus when applying statistical analyses from a Bayesian perspective. In the class of Latent Gaussian Models, we obtain fast and empirically accurate approximations of these marginals by using the Integrated Nested Laplace Approximation (INLA) computational approach Rue et al. (2009), which bypasses sampling-based methods, like the most used and well-known Markov Chain Monte Carlo algorithms. This thesis extensively avails of the INLA methodology to outline new advancements toward software development when tackling Bayesian joint and marginal problems that show extremely skewed patterns. After introducing some theoretical background and numerics behind the vast R-INLA methodology in Chapter 2, we start digging out possibilities to construct joint posterior approximations to enable a fast and accurate joint inference in Latent Gaussian Models, especially for non-Gaussian likelihoods. In Chapter 3 we introduce the new class of Skew Gaussian Copula joint approximation densities applied onto the full conditional posterior density of the latent field. By definition, we construct marginal Skew Normal transformations wrapped in a Gaussian Copula field structure to encode skewness adjustments within the approximated posterior outcomes. This class expands the R-INLA approximation toolkit box when the analysis revolves around joint inference, mainly when we deal with non-Gaussian behavior. Manipulation of a mixture of Skew Gaussian Copula densities across its moment's computations con-

tributes to fastly approximate posterior densities for marginals and linear additive combinations in a subset of the latent field (see Appendix C for an application in R-INLA). Later, we exploit the same mixture representation and an exact Monte Carlo sampling scheme onto the hyperparameter set to approximate the whole joint posterior density of a Latent Gaussian Model. This sampling version of the approximation is much more accurate and generally works for any mixed combination or functional of the model's parameters. In extreme frameworks where the likelihood contribution is far from Gaussian, we observe improvements in the new class of joint approximations and its deterministic, faster manipulative derivations. Computations for the Skew Gaussian Copula with marginal skewness adjustments can be heavy in extensive settings with a more dense grid of hyperparameter configuration points. A parallel approach amongst multiple cores may be preferable to avoid relevant slowdowns. In Chapter 4 we first discuss the underlying Gaussian prior assumptions of the latent field, showing that this pattern holds for both joint and marginal posterior densities derived from the same field. Indeed, these densities are bounded by the product of a Gaussian density and a constant, implying that extreme non-Gaussian assumptions are not needed to properly retrieve the main bulk of the true density. Amongst the available INLA strategies, the Simplified Laplace Approximation is the one that offers the best computational deal, at the cost of some accuracy, by using some parametric modeling fit. This strategy involves a Taylor expansion up to the third-order of full Laplace approximations and then fits Skew Normal distributions to approximate the true posterior densities parametrically. Approximation methodologies like INLA grow in popularity as they ease Bayesian inference analysis by using accurate low or high-order approximations. Related works towards this direction suggest that extensions are possible and bring benefit to the overall approach Ruli et al. (2014); Ruli and Ventura (2016); Ruli et al. (2016). On this topic, we propose to extend the parametric assumptions of the Simplified Laplace strategy by fitting an Extended Skew Normal

distribution to a fourth-order expansion. As we efficiently incorporate the additional parameter of this extended distribution in the approach, we can accurately model more extreme skewed outcomes into R-INLA. This thesis's new developments enlarge the software's capabilities to tackle Bayesian problems within a marginal and joint inference framework. Extreme observed skewed data behaviors now benefit from new modeling assumptions that can recover the truth more accurately than previously implemented features of the R-INLA program. Therefore, any user can exploit these new corrected tools with a user-friendly interface to applications where data scarcity or heavy asymmetric patterns affect the model.

5.2 Future Research Work

The advantages of the Skew Gaussian Copula class of approximation densities described in Chapter 3 allow the resulting joint posterior object to having more accurate corrected marginals for inference purposes. By combining a mixture of this class with a Monte Carlo approach, we can also derive a full joint posterior approximation of the model with computational cost proportional to the number of configuration points. Even though the sampling scheme to achieve the approximation is exact, each Skew Gaussian Copula density in the mixture must be evaluated for each pre-computed grid configuration point. In many applications, this number of points results in being minor, therefore not affecting the computations, but this can rapidly grow large if the dimension of $\boldsymbol{\theta}$ grows large or the grid is denser. In these settings, the computational burden of these corrected joint posterior approximations becomes more evident, and parallel strategies in terms of the configuration hyperparameter points may be required. INLA constantly evolves by encoding more and more features to deal with computational issues in contexts where the problems to solve are huge in dimension (see PARDISO library in Schenk and Gärtner (2004)). Some internal parallel strategies are already part of the Skew Gaussian Copula implementation to limit

these possible slowdowns. Additionally, we believe that a more accurate parametric assumption may be explored and used in the marginal transformations of the copula structure. In Chapter 4 we propose an extension for the Simplified Laplace Approximation strategy in INLA that applies Extended Skew Normal approximations for the posterior marginals of the model parameters. The idea could be further extended by relying on more complex parametric skewed families that involve several parameters to model their structure (Azzalini and Capitanio (1999)). If we can efficiently handle the numerical operations behind the parametric fit, then any modeling assumption is possible and can grant more accuracy in the results at no additional cost. As Bayesian research and computational advancements move forward, INLA becomes more and more appealing for statistical inference in many applied fields (biology, economy, physics, environmental statistics, geoscience, and more) due to its fast and streamlined features that now contain new tools to tackle extremely skewed data observations both in a marginal and joint inference framework.

REFERENCES

A. Alvaro-Meca, R. Akerkar, M. Alvarez-Bartolome, R. Gil-Prieto, H. Rue, and A. Gil de Miguel. Factors involved in health related transitions after curative resection for pancreatic cancer 10 year experience: A multi state model. *Cancer Epidemiology*, 37(1):91–96, 2013.

A. Azzalini and A. Capitanio. Statistical applications of the multivariate skew normal distribution. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):579–602, Aug 1999. ISSN 1467-9868. doi: 10.1111/1467-9868.00194. URL `http://dx.doi.org/10.1111/1467-9868.00194`.

Adelchi Azzalini and Antonella Capitanio. Distributions generated by perturbation of symmetry with emphasis on a multivariate skew t -distribution. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2):367–389, Apr 2003. ISSN 1369-7412. doi: 10.1111/1467-9868.00391. URL `http://dx.doi.org/10.1111/1467-9868.00391`.

Adelchi Azzalini and Antonella Capitanio. *The skew-normal and related families*. Cambridge Cambridge University Press, 2018.

H. Bakka, H. Rue, G. A. Fuglstad, A. Riebler, D. Bolin, J. Illian, E. Krainski, D. Simpson, and F. Lindgren. Spatial modelling with R-INLA: A review. *WIREs Computational Statistics*, 10:e1443(6), 2018. doi: 10.1002/wics.1443. (Invited extended review).

C. Bauer, J. Wakefield, H. Rue, S. Self, Z. Feng, and Y. Wang. Bayesian penalized spline models for the analysis of spatio-temporal count data. *Statistics in Medicine*, 35(11):1848–1865, 2016. doi: 10.1002/sim.6785.

J. Beguin, S. Martino, and H. Rue. Hierarchical analysis of spatially autocorrelated ecological data using integrated nested Laplace approximation. *Methods in Ecology and Evolution*, 3(5):921–929, 2012.

M. Blangiardo, M. Cameletti, G. Baio, and H. Rue. Spatial and spatio-temporal

models with R-INLA. *Spatial and Spatio-Temporal Epidemiology*, 3(December): 39–55, 2013.

David Bolin and Finn Lindgren. Excursion and contour uncertainty regions for latent gaussian models. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 77(1):85–106, 1 2015. ISSN 1369-7412. doi: 10.1111/rssb.12055.

David Bolin and Finn Lindgren. Calculating probabilistic excursion sets and related quantities using excursions. *Journal of Statistical Software, Articles*, 86(5):1–20, 2018. ISSN 1548-7660. doi: 10.18637/jss.v086.i05. URL `https://www.jstatsoft.org/v086/i05`.

Antonio Canale. Statistical aspects of the scalar extended skew-normal distribution. *Metron*, LXIX:279–295, 12 2011. doi: 10.1007/BF03263562.

Antonio Canale. A note on regions of given probability of the extended skew-normal distribution. *Communications in Statistics - Theory and Methods*, 44(12):2507–2516, 2015. doi: 10.1080/03610926.2013.788710. URL `https://doi.org/10.1080/03610926.2013.788710`.

Bob Carpenter, Matthew D Hoffman, Marcus Brubaker, Daniel Lee, Peter Li, and Michael Betancourt. The stan math library: Reverse-mode automatic differentiation in c++. *arXiv preprint arXiv:1509.07164*, 2015.

Bob Carpenter, Andrew Gelman, Matthew Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software, Articles*, 76(1):1–32, 2017. ISSN 1548-7660. doi: 10.18637/jss.v076.i01. URL `https://www.jstatsoft.org/v076/i01`.

Cristian Chiuchiolo, Janet van Niekerk, and Haavard Rue. Joint posterior inference for latent gaussian models with R-INLA, 2021. URL `https://arxiv.org/abs/2112.02861`.

Cristian Chiuchiolo, Janet van Niekerk, and Håvard Rue. An extended simplified laplace strategy for approximate bayesian inference of latent gaussian models using R-INLA, 2022. URL `https://arxiv.org/abs/2203.14304`.

Laura Dawkins, Daniel Williamson, Kerrie Mengersen, and Gavin Shaddick. 'where is the clean air?' a bayesian decision framework for personalised route selection using inla. 08 2019. doi: 10.13140/RG.2.2.22990.20804.

P. de Valpine, D. Turek, C.J. Paciorek, C. Anderson-Bergman, D. Temple Lang, and R. Bodik. Programming with models: writing statistical algorithms for general model structures with nimble. *Journal of Computational and Graphical Statistics*, 26:403–417, 2017. doi: 10.1080/10618600.2016.1172487.

E. Ferkingstad and H. Rue. Improving the INLA approach for approximate Bayesian inference for latent Gaussian models. *Electronic Journal of Statistics*, 9:2706–2731, 2015. doi: 10.1214/15-EJS1092.

E. Ferkingstad, L. Held, and H. Rue. Fast and accurate Bayesian model criticism and conflict diagnostics using R-INLA. *Stat*, 6(1):331–344, 2017. ISSN 2049-1573. doi: 10.1002/sta4.163. URL `http://dx.doi.org/10.1002/sta4.163`.

Dariush Ghorbanzadeh, Philippe Durand, and Luan Jaupi. Generating the skew normal random variable. In *Proceedings of the World Congress on Engineering*, volume 1, 2017.

Virgilio Gómez-Rubio and Francisco Palmí-Perales. Multivariate posterior inference for spatial models with the integrated nested laplace approximation. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 68(1):199–215, 2019.

Virgilio Gómez-Rubio, Roger S Bivand, and Håvard Rue. Estimating spatial econometrics models with integrated nested laplace approximation. *Mathematics*, 9(17): 2044, 2021.

N. I. M. Gould, J. A. Scott, and Y. Hu. A numerical evaluation of sparse direct solvers for the solution of large sparse symmetric linear systems of equations. 33 (2):Article No. 10, 2007.

Virgilio Gómez Rubio. *Bayesian Inference with INLA*. 02 2020. ISBN 9781138039872. doi: 10.1201/9781315175584.

J. R. Hershey and P. A. Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 4, pages IV–317–IV–320, 2007.

A. M. Holand, I. Steinsland, S. Martino, and H. Jensen. Animal models and integrated nested Laplace approximations. *G3: Genes—Genomics—Genetics*, 3(8): 1241–1251, 2013.

Jingyi Huang, Brendan Malone, Budiman Minasny, Alex Mcbratney, and John Triantafilis. Evaluating a bayesian modelling approach (inla-spde) for environmental mapping. *Science of The Total Environment*, 609:621–632, 12 2017. doi: 10.1016/j.scitotenv.2017.07.201.

J. B. Illian, S. H. Sørbye, H. Rue, and D. K. Hendrichsen. Using INLA to fit a complex point process model with temporally varying effects – a case study. *Journal of Environmental Statistics*, 3(7), 2012.

E. T. Krainski, V. Gómez-Rubio, H. Bakka, A. Lenzi, D. Castro-Camilio, D. Simpson, F. Lindgren, and H. Rue. *Advanced Spatial Modeling with Stochastic Partial Differential Equations using R and INLA*. CRC press, December 2018. Github version www.r-inla.org/spde-book.

Stefan Lang, Thomas Kneib, and Andreas Brezger. Bayesx: Analyzing bayesian structural additive regression models. *Journal of Statistical Software*, 14, 01 2005.

Y. Li, P. Brown, H. Rue, M. al-Maini, and P. Fortin. Spatial modelling of Lupus incidence over 40 years with changes in census areas. 61:99–115, 2012.

F. Lindgren, H. Rue, and J. Lindström. An explicit link between Gaussian fields and Gaussian Markov random fields: The SPDE approach (with discussion). 73(4): 423–498, 2011.

S. Martino, K. Aas, O. Lindqvist, L. R. Neef, and H. Rue. Estimating stochastic volatility models using integrated nested Laplace approximation. *The European Journal of Finance*, pages 1–17, 2010a.

S. Martino, R. Akerkar, and H. Rue. Approximate Bayesian inference for survival models. 28(3):514–528, 2010b.

Sara Martino and Andrea Riebler. Integrated nested laplace approximations (inla). *arXiv preprint arXiv:1907.01248*, 2019.

T. G. Martins and H. Rue. Extending INLA to a class of near-Gaussian latent models. 41(4):893–912, 2014.

T. G. Martins, D. Simpson, F. Lindgren, and H. Rue. Bayesian computing with INLA: New features. 67:68–83, 2013.

T. D. Meehan, N. L. Michel, and H. Rue. Spatial modeling of audubon christmas bird counts reveals fine-scale patterns and drivers of relative abundance trends. *Ecosphere*, 10(4), 2019. doi: 10.1002/ecs2.2707. Article e02707.

S. Muff, A. Riebler, H. Rue, P. Saner, and L. Held. Bayesian analysis of measurement error models using integrated nested Laplace approximations. 64(2):231–252, 2015.

Roger B Nelsen. *An introduction to copulas*. Springer, 1999.

Villaseñõr Paulino Pérez-Rodriguez, José A. Bayesian estimation for the centered parameterization of the skew-normal distribution. *Revista Colombiana de EstadÃstica*, 40:123 – 140, 01 2017. ISSN 0120-1751. URL `http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0120-17512017000100006&nrm=iso`.

Stefano Peluso, Antonietta Mira, Håvard Rue, Nicholas John Tierney, Claudio Benvenuti, Roberto Cianella, Maria Luce Caputo, and Angelo Auricchio. A bayesian spatiotemporal statistical analysis of out-of-hospital cardiac arrests. *Biometrical Journal*, 62(4):1105–1119, 2020.

Soraia Pereira, Kamil Feridun Turkman, Luis Correia, and Håvard Rue. Unemployment estimation: Spatial point referenced methods and models. *Spatial Statistics*, 41:100345, 2021.

Kem Phillips. R functions to symbolically compute the central moments of the multivariate normal distribution. *Journal of Statistical Software*, 33, 07 2010. doi: 10.18637/jss.v033.c01.

Martyn Plummer et al. Jags: A program for analysis of bayesian graphical models using gibbs sampling. In *Proceedings of the 3rd international workshop on distributed statistical computing*, volume 124, pages 1–10. Vienna, Austria., 2003.

Z. Quiroz, M. O. Prates, and H. Rue. A Bayesian approach to estimate the biomass of anchovies in the coast of Perú. 71(1):208–217, 2015.

A. Riebler, L. Held, and H. Rue. Estimation and extrapolation of time trends in registry data - Borrowing strength from related populations. 6(1):304–333, 2012a.

A. Riebler, L. Held, H. Rue, and M. Bopp. Gender-specific differences and the impact of family integration on time trends in age-stratified swiss suicide rates. 175(2): 473–490, 2012b.

Christian Robert and George Casella. A short history of markov chain monte carlo: Subjective recollections from incomplete data. *Statistical Science*, 26:102–115, 02 2011. URL https://arxiv.org/pdf/0808.2902.pdf.

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

H. Rue, S. Martino, and N. Chopin. Approximate Bayesian inference for latent Gaussian models using integrated nested Laplace approximations (with discussion). 71(2):319–392, 2009.

H. Rue, A. Riebler, S. H. Sørbye, J. B. Illian, D. P. Simpson, and F. K. Lindgren. Bayesian computing with INLA: A review. *Annual Reviews of Statistics and Its Applications*, 4(March):395–421, 2017. doi: 10.1146/annurev-statistics-060116-054045.

R. Ruiz-Cárdenas, E. T. Krainski, and H. Rue. Direct fitting of dynamic models using integrated nested Laplace approximations - INLA. 56(6):1808–1828, 2012.

Erlis Ruli and Laura Ventura. Higher-order bayesian approximations for pseudo-posterior distributions. *Communications in Statistics - Simulation and Computation*, 45(8):2863–2873, 2016. doi: 10.1080/03610918.2014.930902. URL https://doi.org/10.1080/03610918.2014.930902.

Erlis Ruli, Nicola Sartori, and Laura Ventura. Marginal Posterior Simulation via Higher-order Tail Area Approximations. *Bayesian Analysis*, 9(1):129 – 146, 2014. doi: 10.1214/13-BA851. URL https://doi.org/10.1214/13-BA851.

Erlis Ruli, Nicola Sartori, and Laura Ventura. Improved laplace approximation for marginal likelihoods. *Electronic Journal of Statistics*, 10:3986–4009, 01 2016. doi: 10.1214/16-EJS1218.

Denis Rustand, Janet van Niekerk, Haavard Rue, Christophe Tournigand, Virginie Rondeau, and Laurent Briollais. Bayesian estimation of two-part joint models for a longitudinal semicontinuous biomarker and a terminal event with r-inla: Interests for cancer clinical trial evaluation. *arXiv preprint arXiv:2010.13704*, 2020.

Olaf Schenk and Klaus Gärtner. Solving unsymmetric sparse systems of linear equations with pardiso. *Future Generation Computer Systems*, 20(3):475 – 487, 2004.

ISSN 0167-739X. doi: https://doi.org/10.1016/j.future.2003.07.011. URL http://www.sciencedirect.com/science/article/pii/S0167739X03001882. Selected numerical algorithms.

Antonio Seijas-Macias, Amílcar Oliveira, and Teresa Oliveira. The presence of distortions in the extended skew: normal distribution. In *Proceedings 2nd ISI Regional Statistics Conference*. ISI-RSC, 2017.

Karri Seppä, Håvard Rue, Timo Hakulinen, Esa Läärä, Mikko J Sillanpää, and Janne Pitkäniemi. Estimating multilevel regional variation in excess mortality of cancer patients using integrated nested laplace approximation. *Statistics in medicine*, 38 (5):778–791, 2019.

Daniel Simpson, Finn Lindgren, and Håvard Rue. Fast approximate inference with inla: the past, the present and the future. *arXiv preprint arXiv:1105.2982*, 2011.

Daniel Simpson, Håvard Rue, Andrea Riebler, Thiago G Martins, and Sigrunn H Sørbye. Penalising model component complexity: A principled, practical approach to constructing priors. *Statistical science*, 32(1):1–28, 2017.

S. H. Sørbye, E. Myrvoll-Nilsen, and H. Rue. An approximate fractional Gaussian noise model with $O(n)$ computational cost. *Statistics and Computing*, 29(4):821–833, 2019a. doi: 10.1007/s11222-018-9843-1. URL https://doi.org/10.1007/s11222-018-9843-1.

S. H. Sørbye, E. Myrvoll-Nilsen, and H. Rue. An approximate fractional Gaussian noise model with $O(n)$ computational cost. *Statistics and Computing*, 29(4):821–833, 2019b. doi: 10.1007/s11222-018-9843-1. URL https://doi.org/10.1007/s11222-018-9843-1.

L. Tierney and J. B. Kadane. Accurate approximations for posterior moments and marginal densities. 81(393):82–86, 1986.

Janet van Niekerk and Haavard Rue. Correcting the laplace method with variational bayes, 2021. URL https://arxiv.org/abs/2111.12945.

Janet Van Niekerk, Haakon Bakka, Håvard Rue, and Olaf Schenk. New frontiers in bayesian modeling using the inla package in r. *arXiv preprint arXiv:1907.10426*, 2019.

Janet van Niekerk, Elias Krainski, Denis Rustand, and Haavard Rue. A new avenue for bayesian inference with inla, 2022. URL `https://arxiv.org/abs/2204.06797`.

Jon Wakefield, Daniel Simpson, and Jessica Godwin. Spatial modeling, with application to complex survey data: Discussion of" model-based geostatistics for prevalence mapping in low-resource settings", by diggle and giorgi. *arXiv preprint arXiv:1608.03769*, 2016.

Simon N Wood. Simplified integrated nested laplace approximation. *Biometrika*, 107 (1):223–230, 2020.

Y. Yuan, F. E. Bachl, D. L. Borchers, F. Lindgren, J. B. Illian, S. T. Buckland, H. Rue, and T. Gerrodette. Point process models for spatio-temporal distance sampling data from a large-scale survey of blue whales. *Annals of Applied Statistics*, 11(4): 2270–2297, 2017.

Y. R. Yue, D. Bolin, H. Rue, and X. Wang. Bayesian generalized two-way ANOVA modeling for functional data using INLA. *Statistica Sinica*, 29(2):741–767, 2019. doi: https://doi.org/10.5705/ss.202016.0055.

APPENDICES

A The Gaussian Approximation

In Chapter 2 we introduced the Integrated Nested Laplace Approximation (INLA), which offers many approximation strategies to perform marginal posterior inference. For Gaussian likelihood models, the most accurate and fast is the Gaussian Approximation which is generally applied to densities of the form

$$\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}) \propto \exp\Big(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \sum_{i\in\mathcal{I}} g_i(x_i)\Big) \tag{A.1}$$

where $\boldsymbol{Q}$ is the precision matrix of the latent field $\boldsymbol{x}$ depending on hyperparameters $\boldsymbol{\theta}$ while $\mathcal{I}$ is an index set for data observations $\boldsymbol{y}$. By Latent Gaussian Model structure, the functions $g_i(x_i)$ refer to the log-likelihood densities $\log\{\pi(y_i|x_i,\boldsymbol{\theta})\}$. The Gaussian Approximation is obtained by matching the modal configuration and the curvature at the mode of $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$. This task is accomplished iteratively by using optimization methods like Newton-Raphson or Quasi-Netwon methods. We first apply a Taylor expansion up to second order on $g_i(x_i)$ around an initial guess $\mu_i^{(0)}$ and approximate the target functions $g_i(x_i)$ as

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

where (b_i, c_i) are constants related to $\mu_i^{(0)}$ in terms of the polynomial approximation. In the update step of the process, the new precision matrix becomes $\boldsymbol{Q}_{\mathbf{G}} = \boldsymbol{Q}+\text{diag}(\boldsymbol{c})$ with the modal configuration computed as the solution of the linear system $\boldsymbol{Q}_{\mathbf{G}}\boldsymbol{\mu}^{(1)} =$

$\boldsymbol{b}$. Then we continue the iterative update loop until we converge to a Gaussian distribution with, say, final mean $\boldsymbol{\mu}^*$ and final precision matrix $\boldsymbol{Q}^* = \boldsymbol{Q} + \text{diag}(\boldsymbol{c}^*)$. These quantities summarize the Gaussian Approximation $\tilde{\pi}_{\text{G}}(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ of (A.1), which result in a multivariate Gaussian distribution denoted as $N(\boldsymbol{\mu}^*, \boldsymbol{Q}^*)$. Markov conditional properties of the Gaussian Markov Random Field (GMRF) structure of $\boldsymbol{x}$ (see Section 2.3) are preserved since the functions $g_i(x_i)$ do not involve mixed product terms.

B R Codes to fit GLMMs in JAGS and INLA

Here we provide R language implementations using JAGS for the Markov Chain Monte Carlo strategy and INLA approach to fit a pair of Generalized Linear Mixed Models with Poisson and Binomial likelihood. Below, we propose implementing the Poisson model leaving the Binomial to the reader as few changes are needed. Both MCMC and INLA simulation approaches run in parallel on multiple system cores.

```
#load R2jags library

require(R2jags)


#Poisson model JAGS

model.Poi <- function() {
  for(i in 1:N) {
  y[i] ~ dpois(mu[i])
  log(mu[i]) <- alpha + re[grps[i]]
```

```
  }
  for(j in 1:M) {
    re[j] ~ dnorm(0, tau)
  }
  alpha ~ dnorm(0, 0.001)
  tau ~ dgamma(0.1, 0.1)

  sd <- 1/sqrt(tau)
}

#MCMC run in JAGS

mod.jags <- jags.parallel(data = c("y", "grps", "N", "M"),
                          n.chains = 20,
                          n.iter = 6000000,
                          n.burnin = 1000000,
                          n.thin = 100,
                          parameters=c("mu", "alpha", "re", "tau"),
                          inits=list(list(alpha = 1,
                                          re = rep(0, M),
                                          tau = 1.5)),
                          model.file = model.Poi)$BUGSoutput

#Dataframe for Poisson data in INLA

df.Poi <- data.frame(y, grps)
```

```
#INLA Poisson setup

mod.Poi <- inla(y ~ 1 + f(grps, hyper = list(
  prec = list(prior = "loggamma",
              param = c(0.1, 0.1),
              initial = log(1.5)))),
  data = df.Poi,
  family = "poisson",
  control.predictor = list(compute = TRUE),
  control.compute = list(config = TRUE,
                         return.marginals.predictor = TRUE),
  control.fixed = list(mean.intercept = 0,
                       prec.intercept = 0.001)
)
```

In this setting, we remark the user friendly implementation of the model setup in INLA compared to the respective one in JAGS. The control options `control.predictor = list(...)` and `control.compute = list(...)` in the `inla` main function call are mandatory for computing both the hyperparameter configuration points and the posterior linear predictor marginals of the model. The computed points allow to construct the joint posterior approximation $\tilde{\pi}(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ formulated in Chapter 3. This joint object is straightforward in R-INLA as we simply need to use the function `inla.posterior.sample` while MCMC strategies already compute an approximation for the joint posterior density by default.

```
samples = 1e04

joint.inla = inla.posterior.sample(n = samples,
                                   result = mod.Poi,
```

```
                                    skew.corr = TRUE)
```

The INLA function above constructs a joint posterior approximation of the Poisson model by drawing 10^4 independent samples to represent it properly. The sampling scheme is exact since we draw samples from a correct empirical approximation. The option `skew.corr = TRUE` enables the marginal skewness correction adjustments specifying that we are using the Skew Gaussian Copula formulation in (3.35). Some historical background on the function. It officially came out in 2013 as an INLA extension to approximate joint posterior densities by combining pre-existing features of the original implementation. The code was first created for the purpose of completing the excursion project on Latent Gaussian Models appeared on Bolin and Lindgren (2015) and Bolin and Lindgren (2018) within the R `excursions` package. This task was an easy hack, as the main work was mainly to store all grid configurations and combine the feature with the Gaussian Markov Random Field sampling function `inla.qsample`. This function became available starting May 2012 and was necessary for the joint project by David Bolin and Finn Lindgren. Some tutorials on this joint INLA posterior sampler can be found in Krainski et al. (2018) and Martino and Riebler (2019) which accurately explain how to derive and interpret correct functionals of the results for prediction purposes.

C Linear Combinations with R-INLA

In Section 3.2 we introduced how to construct joint posterior approximations for a set of linear combinations $\boldsymbol{Ax}$ where A is a matrix of indexes and $\boldsymbol{x}$ is the latent field of a Latent Gaussian Model. We exploit a surrogate Skew Gaussian Copula and its moments to derive linear combinations as marginals of the joint object. A Skew Normal distribution approximates each marginal. The new R-INLA tools that allow

to construct these new approximations are encoded in the functions `inla.tjmarginal` and `inla.1djmarginal`. These functions apply a fast post-process of the main INLA output after fitting the model of interest. The R code below shows an example by constructing approximations for two linear additive combinations in a Poisson model with three covariates.

```
# Poisson regression dataset (simulation)

nn = 50
p = 3
x1 <- rnorm(nn)
x2 <- rnorm(nn)
x3 <- rnorm(nn)
eta = 1+x1+x2+x3
y = rpois(nn, lambda = exp(eta))

data = data.frame(y = y, x1 = x1, x2 = x2, x3 = x3)

sel = list('(Intercept)'= 1, x1 = 1, x2 = 1, x3 = 1)

# Fitting the data into an INLA framework

mod.ex <- inla(y ~ 1+x1+x2+x3,
               family = "poisson",
               data = data,
               selection = sel,
               control.compute = list(config = TRUE))
```

```
# Getting deterministic approximations from the SGC object

Lin.idx = matrix(c(0, 1, 1, 0, 0, 1, 1, 1),
                   nrow = 2, ncol = p+1,  byrow = T)

Lin.idx
     [,1] [,2] [,3] [,4]
[1,]    0    1    1    0
[2,]    0    1    1    1

mod.det = inla.tjmarginal(jmarginal = mod.ex$selection,
                          A = Lin.idx)

$names
[1] "Lin:1" "Lin:2"

$mean
          [,1]
[1,] 1.971917
[2,] 3.013960

$cov.matrix
             [,1]         [,2]
[1,] 0.002613337 0.002072906
[2,] 0.002072906 0.004420322
```

```
$skewness
[1] 0.06482934 0.03272285

dm = inla.1djmarginal(jmarginal = mod.det)

# Plot the deterministic results

plot(dm$`Lin:1`, type = 'l')
plot(dm$`Lin:2`, type = 'l')
```

Through the `selection` list option in the main INLA function call, we can extract all the moments and correlation structure of the parameters of interest, here provided by the three model covariates x_1, x_2 and x_3. We focus on getting posterior approximations for the two specified linear combinations $x_1 + x_2$ and $x_1 + x_2 + x_3$ from the joint posterior structure of the model. First, we use the new function `inla.tjmarginal` to construct the surrogate Skew Gaussian Copula object structure with its moments and correlation matrix amongst the two linear combinations whose indexes belong to `Lin.idx`. Then we use the new function `inla.1djmarginal` to compute Skew Normal approximations for $\pi(x_1 + x_2|\boldsymbol{y})$ and $\pi(x_1 + x_2 + x_3|\boldsymbol{y})$. The results can be compared to the ones obtained from the sampling-based joint posterior approximation encoded in `inla.posterior.sample` function (see Appendix B for an application). More examples, insights of these new functions can be found on the tutorial vignette available

on the R-INLA official website.

D Submitted Papers and Book INLA Project

A list of submitted papers and projects that contributed to the realization of this thesis

- Cristian Chiuchiolo, Janet Van Niekerk and Håvard Rue (2021), "Joint Posterior Inference for Latent Gaussian Models with R-INLA". Submitted on Arxiv at `https://arxiv.org/abs/2112.02861` and at *Journal of Statistical Computation and Simulation* for review.

- Cristian Chiuchiolo, Janet Van Niekerk and Håvard Rue (2022), "An Extended Simplified Laplace strategy for Approximate Bayesian inference of Latent Gaussian Models using R-INLA". Submitted on Arxiv at `https://arxiv.org/abs/2203.14304` and at *Electronic Journal of Statistics* for review.

- Book INLA project using R-INLA software to be published with Wiley (*coming Jan 2023*). Tentative title is: *Temporal and Spatial Modeling of Landslide Hazards*. Authors are: Luigi Lombardo, Daniela Castro-Camilo, Thomas Opitz, Janet Van Niekerk, Cristian Chiuchiolo, Elias Teixeira Krainski, Hakan Tanyas, Anna Freni Sterrantino.