UHASSELT
KNOWLEDGE IN ACTION

Maastricht University

Faculty of Sciences
School for Information Technology
Master of Statistics and Data Science

Master's thesis

Laplacian-P-Splines in Gamma Frailty Survival Model

Piotr Lewczuk

Thesis presented in fulfillment of the requirements for the degree of Master of Statistics and Data Science, specialization Biostatistics

SUPERVISOR :

Prof. dr. Christel FAES

dr. Oswaldo GRESSANI

2023
2024

Transnational University Limburg is a unique collaboration of two universities in two countries: the University of Hasselt and Maastricht University.

UHASSELT
KNOWLEDGE IN ACTION

www.uhasselt.be
Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek

Maastricht University

Faculty of Sciences
School for Information Technology
Master of Statistics and Data Science

Master's thesis

Laplacian-P-Splines in Gamma Frailty Survival Model

Piotr Lewczuk

Thesis presented in fulfillment of the requirements for the degree of Master of Statistics and Data Science, specialization Biostatistics

SUPERVISOR :
Prof. dr. Christel FAES
dr. Oswaldo GRESSANI

To my wife,
our son,
and memory of my parents

Contents

Abstract

BACKGROUND: Laplacian-P-splines (LPS) has been recently shown to deliver a fast and accurate Bayesian inference in (generalized) additive models, epidemic models, additive proportional odds models, and proportional survival models.
OBJECTIVES: This project aims at extending the LPS toolbox in survival models to build a method that makes explicit use of the gradient and the Hessian information resulting from Laplace approximations. In particular, it is applied to Gamma Shared Frailty survival model.
METHODS: Two algorithms were developed: a sampling-free LPS algorithm, and Metropolis-Adjusted Langevin Algorithm (MALA) within Gibbs Sampler. The two methods were derived algebraically, implemented in `R`, tested on 300 simulated datasets with different right-censoring ratios, compared to available frequentist function (`emfrail`), and applied on a real-life dataset from a randomized clinical trial of Interferon Gamma (IG) in Chronic Granulomatous Disease (CGD) in children.
RESULTS: The estimated B-spline coefficients and the regression parameters turned out to be reasonably precise and to have negligible bias on the simulated datasets. Also the estimates of the frailty variance were virtually identical to those of the frequentist method, though some discrepancies were observed in datasets with clusters of small sizes. In the CGD study, the two algorithms developed in this project and the `emfrail` function resulted in very similar point estimates and the 95% confidence/credible intervals of the treatment effect: -1.163 [-1.867; -0.460], -1.190 [-1.879; -0.484], and -1.052 [-1.660; -0.444], respectively, leading to the same conclusion of substantial effect of IG treatment on reducing hazard of recurrent serious infections. Similarly, very close point estimates and the intervals were obtained for female sex (-0.250 [-1.142; 0.642], -0.249 [-1.185; 0.623], and -0.227 [-1.003; 0.548], respectively), with no hazard-altering effect.
CONCLUSIONS: The two algorithms developed in this project reliably extend LPS methodology to Gamma Shared Frailty survival models.

Key words: *Bayesian statistics; Laplacian P-splines; Metropolis-adjusted Langevin algorithm; Gamma shared frailty model; Chronic granulomatous disease.*

✉ Piotr.Lewczuk@uk-erlangen.de

1 Introduction

Laplacian-P-splines (LPS) have recently been shown to be a powerful tool for inference in different model classes. The flexibility of P-spline smoothers, combined with Laplace approximations to selected posterior target distributions has opened up a modelling path that delivers a fast and accurate methodology for Bayesian inference in (generalized) additive models [22], epidemic models [23], additive proportional odds models [28], and survival models [21]. Key quantities associated to the Laplace approximation of a target distribution are the gradient and the Hessian as they are used in iterative algorithms to compute the mode of the posterior target. The gradient and Hessian information can also be used in Markov chain Monte Carlo (MCMC) algorithms to build powerful and enhanced sampling techniques such as Metropolis-Adjusted Langevin Algorithm (MALA).

This master thesis aims at extending the LPS and the MALA toolbox in survival models. It will combine three classes of statistical methodologies: (i) P-splines smoothers, (ii) Laplacian approximations to target posterior distributions, and (iii) shared frailty modelling, to depart from the Bayesian Cox Proportional Hazards (CPH) model developed recently [19]. LPS and MALA will be derived analytically, tested for plausibility in a set of simulations, and applied to analyze data from a study on chronic granulomatous disease (CGD) in children [24]. In the remaining part of this section, the three classes of statistical methods will be briefly introduced.

In its physical sense, a *spline* is a device made of flexible material which, when properly bent, can be used to draft an arbitrary curve on paper, wood or any kind of flat surface. Before onset of

computers, splines were used for creating designs by hand. To draw curves, draftsmen used long, thin, flexible strips of wood, plastic, or metal, held in place with lead weights. The elasticity of the spline material combined with the constraint of the control points, or knots, would cause the strip to take the shape that minimized the energy required for bending it between the fixed points, leading to the smoothest possible shape [2].

Mathematicians took over this idea and created a class of piecewise polynomial functions capable to reflect curves of complicated shapes. Many different types of splines arose, as reviewed in [38], with B-splines, introduced by Schoenberg, being one of the most widely used [37]. For a d-degree B-spline, $d+1$ polynomial segments are linked together at d inner knots, resulting in a basis function, b, which is positive on the domain made of $d+2$ knots and zero elsewhere. To approximate an unknown function f, a B-spline of degree $d=3$, *i.e.* consisting of cubic basis functions, can be used, such that f is modelled as a linear combination of K basis functions:

$$f(x) = \sum_{k=1}^{K} \theta_k b_k(x),$$

where θ_k are the spline parameters, referred to as *amplitudes*, and $b_k(x)$ is the k-th spline basis function (for more details, see [19]). Assembling K basis functions, $b_1(x), ..., b_K(x)$, for n observations into a $n \times K$ matrix, $\boldsymbol{B}$, the amplitude vector, $\boldsymbol{\theta}$, can be found by regressing the outcome variable, Y, onto the matrix $\boldsymbol{B}$. By applying least squares, this gives:

$$\widehat{\boldsymbol{\theta}} = (\boldsymbol{B}^T\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{y}.$$

B-splines characterize with one serious limitation, namely the shape of the resulting curve, and hence the degree how exact the model fits the data, strongly depend on the number of the basis functions, K. The more basis functions used, the more exactly the curve follows the data, which can quickly lead to overfitting. To overcome this problem, the idea of penalized splines (P-splines) was proposed [15]. P-splines allow (or even force) an arbitrarily large number of basis functions, counterbalancing potential overfitting by penalization based on finite differences between adjacent B-spline coefficients. Briefly, an r-th order $(K-r) \times K$ difference matrix, $\boldsymbol{D}_r$, is used along with a positive penalty parameter, λ, to control smoothness of the fit. Assuming first order difference, penalty matrix is:

$$\boldsymbol{D}_1 = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & -1 & 1 \end{bmatrix}.$$

Taking penalty into account, the solution of the least squares becomes:

$$\widehat{\boldsymbol{\theta}} = (\boldsymbol{B}^T\boldsymbol{B} + \lambda\boldsymbol{D}_r^T\boldsymbol{D}_r)^{-1}\boldsymbol{B}^T\boldsymbol{y}.$$

In the frequentist framework, the optimal smoothing parameter can be found, for example, by cross-validation. Penalized splines characterize with several advantages, as summarized in [14, 16], where Eilers and Marx also stated "P-splines are the ultimate smoothing tool.".

To illustrate the concept of smoothing splines and the role of penalty, let us reconsider a dataset modelled in one of the courses of the biostatistics program at Hasselt University[1]. Briefly, a publicly available dataset [1] contained - among others - the number of new daily deaths due to COVID-19 infection per one million European Union citizens in 2021 as a function of the percentage of fully

[1] Advanced Modelling Techniques, 2022. In the course, this dataset was analyzed with non-linear regression.

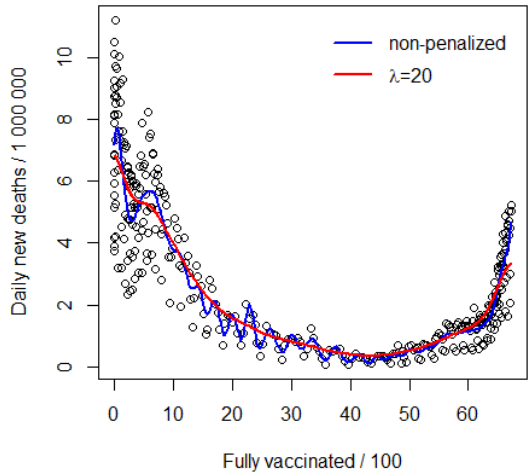


Figure 1: The number of new daily deaths due to COVID-19 infection per one million EU citizens in 2021 as a function of the percentage of fully vaccinated Europeans (dots), and two smoothing spline models with $K = 50$ cubic basis functions: a non-penalized model (blue curve) and a model with the penalty parameter $\lambda = 20$ (red curve).

vaccinated Europeans on the same day. Figure 1 presents the data along with two spline models, each with $K = 50$ cubic basis functions: the blue curve, resulting from a non-penalized approach, is wiggly and follows the data too close leading to potential overfitting. In contrast, the red curve, representing a penalized model with an arbitrarily chosen penalty parameter, $\lambda = 20$, still captures the data well but is much smoother.

P-splines can be integrated into the Bayesian framework by introducing a stochastic version of the difference penalty [29]. A diffuse prior on the initial amplitude is imposed, $p(\theta_1) \propto c$, followed by a random walk (of the first order) imposed on the following amplitudes: $p(\theta_k) = \theta_{k-1} + \varepsilon$, with $\varepsilon \sim \mathcal{N}(0,\ \lambda^{-1})$. Further, the penalty matrix can be defined, $\boldsymbol{\mathcal{P}} = \boldsymbol{D}_1^T \boldsymbol{D}_1 + \epsilon \boldsymbol{I}_K$, where a small ϵ added to the diagonal elements ensures that $\boldsymbol{\mathcal{P}}$ is full rank. Then it follows that the proper prior for the vector of amplitudes, conditional on the penalty, λ, becomes $p(\boldsymbol{\theta}|\lambda) \propto \lambda^{\frac{K}{2}} \exp(-0.5\lambda \boldsymbol{\theta}^T \boldsymbol{\mathcal{P}} \boldsymbol{\theta})$ [10, 19].

Very often functions we are dealing with, for example posterior distributions in a Bayesian context, become very complex and analytically untraceable, which calls for approximation strategies. In such scenarios, *Laplace approximation* is a very powerful tool (see [30] for English translation of the original paper published in 1774). Briefly, keeping in mind that the first derivative of a function equals zero at its mode, we observe (or, better to say, Laplace observed) that the second-order Taylor series expansion of a logarithmized function, $\ln p(\theta)$, around its mode, $\widehat{\theta}$, becomes:

$$\ln p(\theta) \approx \underbrace{\ln p(\widehat{\theta})}_{constant} + \frac{1}{2} \frac{d^2 \ln p(\theta)}{d\theta^2}\bigg|_{\theta=\widehat{\theta}} (\theta - \widehat{\theta})^2.$$

We also notice that the logarithm of the kernel of a Normal distribution for θ with mean μ and variance σ^2 is:

$$-\frac{1}{2\sigma^2}(\theta - \mu)^2,$$

from hence we conclude that the function in question, $p(\theta)$, can be approximated by a Normal distribution with mean $\mu = \widehat{\theta}$ and variance $\sigma^2 = -[\frac{d^2 \ln p(\theta)}{d\theta^2}]^{-1}$ evaluated at $\widehat{\theta}$.

To illustrate it, let us assume that the scalar parameter of interest, θ, follows a Gamma distribution, $\theta \sim \mathcal{G}(\alpha, \beta)$. Then the first and the second derivative of the log-density function become, respectively:

$$\frac{d \ln p(\theta)}{d\theta} = \frac{\alpha - 1}{\theta} - \beta$$
$$\frac{d^2 \ln p(\theta)}{d\theta^2} = -\frac{\alpha - 1}{\theta^2}.$$

By setting the first derivative equal to zero and solving for θ we obtain the well-known expression for the mode of a Gamma distribution, $\widehat{\theta} = (\alpha - 1)/\beta$, and – by evaluating the second derivative at the mode – the negative inverse of the approximation variance, provided that $\alpha \geq 1$ and $\beta > 0$. The Laplace approximation becomes then:

$$p(\theta) \approx p_G(\theta) = \mathcal{N}\Big(\frac{\alpha - 1}{\beta}, \frac{\alpha - 1}{\beta^2}\Big),$$

which is presented in Figure 2 for arbitrarily chosen $\alpha = 100$ and $\beta = 5$. Note that the above illustration has limitations, as the Gamma distribution has not the same support as the Normal distribution. As such, there will be nonzero probability mass on the negative real line with the Laplace approximation.

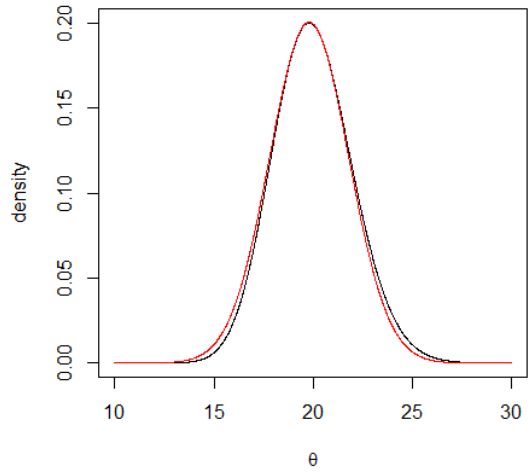


Figure 2: Illustration of a Gamma distribution ($\mathcal{G}(100,\ 5)$, black curve) and its Laplace approximation ($\mathcal{N}(19.8,\ 3.96)$, red curve).

Having briefly introduced P-splines smoothers and Laplacian approximations, we turn now to the concept of shared frailty. Assuming independent observations of a non-negative random variable time-to-event, T, the dependence between the survival time, $S(t)$, and the cumulative hazard function, $H(t)$, is given by $S(t) = \exp(-H(t))$. For n independent, right-censored observations, the likelihood function is given by (see, for example, p. 180 in [12]):

$$\mathcal{L} = \prod_{j=1}^{n} h(t_j)^{\widetilde{\delta}_j} S(t_j),$$

where $\widetilde{\delta}_j$ is an event indicator ($\widetilde{\delta}_j = 1$ if the event of interest is observed in the j-th case or 0 otherwise). Under CPH model [13], the baseline hazard function, $h_0(t)$, can be specified as exponent of a linear combination of cubic B-splines:

$$h_0(t) = \exp(\boldsymbol{\theta}^T \boldsymbol{b}(t)), \tag{1}$$

with $\boldsymbol{b}(\cdot)$ a cubic B-spline basis defined on $[0, t_{max}]$, where t_{max} is the maximal follow-up time (to event or censoring) observed in the dataset. From there, the following approximation becomes straightforward, and is necessary due to the fact that the integration in the cumulative hazard function, $H_0(t)$, has no analytic solution and needs to be approximated numerically on a grid, here assumed equidistant (for more details, see [19]):

$$H_0(t) = \int_0^t h_0(s)ds = \int_0^t \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s))ds \approx \sum_{m=1}^{m(t)} \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m))\Delta, \tag{2}$$

where $m(t)$ is the grid segment containing the value of time t, s_m is the time corresponding to the mid-point of that segment, and Δ is the length of the grid segments. With $\boldsymbol{z}_j$ vector of covariates for the j-th observation, and $\boldsymbol{\beta}$ vector of regression coefficients, likelihood in the CPH model can be written, utilizing B-splines specification for the hazard function, as:

$$\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, D) \approx \prod_{j=1}^{n} \Big(\exp(\boldsymbol{\theta}^T \boldsymbol{b}(t_j) + \boldsymbol{\beta}^T \boldsymbol{z}_j) \Big)^{\widetilde{\delta}_j} \times \exp\left\{ -\Big(\sum_{m=1}^{m(t_j)} \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m) + \boldsymbol{\beta}^T \boldsymbol{z}_j)\Delta \Big) \right\}. \tag{3}$$

All considerations so far have been developed under assumption that observation units are independent. As a matter of fact, very often this is not the case, for example when there exist some (latent) characteristics shared by groups of individuals or when events of interest occur repeatedly in the same individual. To accommodate for within-cluster homogeneity, a class of survival models, called *shared frailty models*, was developed (for an extended discussion see, for example pp. 345-380 in [12]). Consider, for illustrative example, a multicenter study of time to adverse effect after a new surgical procedure. Then it is plausible to assume that patients from one center will share a common characteristics due to being operated in this particular center in contrast to patients being operated in another center. Or we may be interested in time between placement of dental fillings and development of secondary caries. Then, again, it is plausible to assume that the observed time-to-events are not independent due to existence of patient-specific predispositions. A special scenario regards studies with recurrent appearance of the event of interest (say, time elapsing between episodes of migraine or seasonal flue) in a given subject. Although models dedicated to such datasets exist [4], they also may be analyzed by shared frailty model, under consideration that all time-to-events in a given subject share the same frailty.

CGD is a serious immunodeficiency condition of humans caused by a phagocytic malfunction due to mutations in genes encoding nicotinamide adenine dinucleotide phosphatase (NADPH) oxidase. Inability to generate reactive oxygen species brings about formation of systemic granuloma, considered pathognomonic. CGD manifests, primarily in children, with recurrent life-threatening infections [27]. It was first described in late 1950's in four children, of whom all eventually died, with syndrome consisting of chronic suppurative lymphadenitis, hepatosplenomegaly, pulmonary infiltrations, and eczematoid dermatitis [11]. Since then, several approaches have been more or less successfully undertaken to prevent recurrent infections, including antibacterial and antifungal prophylaxis. In early 1990's, a placebo-controlled, multicenter, randomized trial reported effectiveness and safety of Interferon Gamma (IG) treatment, which drastically reduced the hazard of recurrent infections [24].

2 Materials and Methods

2.1 From the conditional to the marginal log-likelihood, the gradient, and the Hessian

Consider n observations subgrouped into $\mathcal{I}$ mutually exclusive clusters such that an i-th cluster consists of n_i observations sharing the same frailty. Let us define a latent vector of the amplitudes and the regression coefficients, $\boldsymbol{\xi} := (\boldsymbol{\theta}^T, \boldsymbol{\beta}^T)^T$. Then, Eq. (3) can be rewritten for the contribution of the i-th cluster to the likelihood, conditional on the cluster-specific frailty, u_i:

$$\mathcal{L}_i(\boldsymbol{\xi} \mid u_i; D) \approx \prod_{j=1}^{n_i} \Big(\exp(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}))\ u_i\ \exp(\boldsymbol{\beta}^T \boldsymbol{z}_{ij}) \Big)^{\widetilde{\delta}_{ij}} \times \exp\left\{ -\Big(\sum_{m=1}^{m(t_{ij})} \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m))\Delta \Big)\ u_i\ \exp(\boldsymbol{\beta}^T \boldsymbol{z}_{ij}) \right\}, \tag{4}$$

where approximation comes from Eq. (2). Note that the baseline hazard, $h_0(t)$, is already written in terms of B-splines, resulting from Eq. (1). From there, marginal likelihood is obtained by integrating the frailty variable out:

$$\mathcal{L}_i(\boldsymbol{\xi}; D) = \int_0^\infty \prod_{j=1}^{n_i} \Big(\exp(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}))\ u\ \exp(\boldsymbol{\beta}^T \boldsymbol{z}_{ij}) \Big)^{\widetilde{\delta}_{ij}} \times \exp\left\{ -\Big(\sum_{m=1}^{m(t_{ij})} \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m))\Delta \Big)\ u\ \exp(\boldsymbol{\beta}^T \boldsymbol{z}_{ij}) \right\} f_U(u)\ du, \tag{5}$$

which, after slight rearrangement, gives:

$$\begin{aligned} \mathcal{L}_i(\boldsymbol{\xi}; D) &= \int_0^\infty \prod_{j=1}^{n_i} \Big(\exp(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij}) \Big)^{\widetilde{\delta}_{ij}} \\ &\quad \times u^{\widetilde{\delta}_{ij}}\ \exp\left\{ -\Big(\sum_{m=1}^{m(t_{ij})} \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij})\Delta \Big) u \right\} f_U(u)\ du \\ &= \int_0^\infty \underbrace{\exp\Big(\sum_{j=1}^{n_i} \widetilde{\delta}_{ij} (\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij}) \Big)}_{\textit{not depending on } u} \\ &\quad \times \underbrace{u^{\sum_{j=1}^{n_i} \widetilde{\delta}_{ij}}\ \exp\left\{ -\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij})\Delta \Big) u \right\}}_{\textit{kernel of a Gamma distribution}} f_U(u)\ du. \end{aligned} \tag{6}$$

Now, in order to compactly write the (log-)likelihood, the gradient, and the Hessian, we define a quantity:

$$\omega_{ijm} := \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij})\Delta, \tag{7}$$

which is specific for the j-th observation in the i-th cluster and the m-th segment on the time grid.

Gamma frailty model assumes that the frailty, U, is a random variable that follows a Gamma distribution with mean 1, $f_U(u) = \mathcal{G}_u(\gamma, \gamma)$. This means that the γ parameter represents inverse of

the variance of the frailty, $\gamma = 1/Var(U)$, and implies that the integrand in Eq. (6) is proportional to the product of $n_i + 1$ Gamma distributions, which is kernel of a Gamma distribution $\mathcal{G}(\widetilde{\alpha}_i, \widetilde{\beta}_i)$, with the parameters:

$$\widetilde{\alpha}_i = \gamma + \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} \tag{8}$$

$$\widetilde{\beta}_i = \gamma + \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm}, \tag{9}$$

and the constant, independent of the integration variable u:

$$\exp\Big[\sum_{j=1}^{n_i} \widetilde{\delta}_{ij}(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij})\Big] \; \frac{\gamma^\gamma}{\Gamma(\gamma)}. \tag{10}$$

This means that the contribution of the i-th cluster to the marginal likelihood is equal to the product of the above constant of the integration (Eq. (10)) and the inverse of the constant term of the $\mathcal{G}(\widetilde{\alpha}_i, \widetilde{\beta}_i)$ distribution:

$$\begin{aligned}
\mathcal{L}_i(\boldsymbol{\xi}, \gamma; D) = \Gamma(\gamma + \sum_{j=1}^{n_i} \widetilde{\delta}_{ij}) \\
&\times \exp\Big[\sum_{j=1}^{n_i} \widetilde{\delta}_{ij}(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij})\Big] \\
&\times \Big(\gamma + \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij})\Delta\Big)^{-(\gamma + \sum_{j=1}^{n_i} \widetilde{\delta}_{ij})} \\
&\times \gamma^\gamma \, \Gamma(\gamma)^{-1},
\end{aligned} \tag{11}$$

which, taking Eqs. (7) - (9) into account, we can express compactly:

$$\mathcal{L}_i(\boldsymbol{\xi}, \gamma; D) = \exp\Big[\sum_{j=1}^{n_i} \widetilde{\delta}_{ij}(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij})\Big] \frac{\gamma^\gamma}{\Gamma(\gamma)} \times \widetilde{\beta}_i^{-\widetilde{\alpha}_i} \Gamma(\widetilde{\alpha}_i). \tag{12}$$

From there, the i-th subgroup contribution to the marginal log-likelihood becomes:

$$\ell_i(\boldsymbol{\xi}, \gamma; D) = \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} \Big(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij}\Big) - \widetilde{\alpha}_i \ln(\widetilde{\beta}_i) + \underbrace{\gamma \ln(\gamma) - \ln\Big(\Gamma(\gamma)\Big) + \ln\Big(\Gamma(\widetilde{\alpha}_i)\Big)}_{\textit{terms depending on neither } \boldsymbol{\theta} \textit{ nor } \boldsymbol{\beta}}. \tag{13}$$

For the derivation of the gradient and the Hessian with respect to (wrt) the amplitudes and the regression coefficients in this section, this expression could be written omitting the terms depending only on γ; however, since we will also need the derivative wrt the γ parameter in the following sections, when it comes to the Metropolis Algorithm, we rather write Eq. (13) in full.

Now let us develop the gradient and the Hessian wrt the amplitudes and the regression coefficients, which we will need for the gradient and the Hessian of the posterior distribution of the latent vector. The element of the gradient, $\nabla \ell_i(\boldsymbol{\xi}, \gamma; D)$, wrt the k-th amplitude, θ_k, becomes:

$$\frac{\partial \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \theta_k} = \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} b_k(t_{ij}) - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}, \tag{14}$$

and the element of the gradient wrt the p-th regression coefficient, β_p, becomes:

$$\frac{\partial \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \beta_p} = \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} z_{ijp} - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} z_{ijp}\omega_{ijm}. \tag{15}$$

The Hessian, $\nabla^2 \ell_i(\boldsymbol{\xi}, \gamma; D)$, is a symmetric block matrix with the elements in the top-left block:

$$\frac{\partial^2 \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \theta_k \partial \theta_{k'}} = -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} b_k(s_m) b_{k'}(s_m)\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} b_{k'}(s_m)\omega_{ijm}\Big)\Big], \tag{16}$$

the elements in the bottom-right block:

$$\frac{\partial^2 \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \beta_p \partial \beta_{p'}} = -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} z_{ijp} z_{ijp'}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} z_{ijp}\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} z_{ijp'}\omega_{ijm}\Big)\Big], \tag{17}$$

and the elements in the off-diagonal top-right block (the bottom-left block being its transpose):

$$\frac{\partial^2 \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \theta_k \partial \beta_p} = -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} b_k(s_m) z_{ijp}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} z_{ijp}\omega_{ijm}\Big)\Big]. \tag{18}$$

A detailed derivation of the elements of the gradient and the Hessian is given in **Appendix 1**.

2.2 The priors

Second, after the likelihood, component of the Bayesian model are the priors. For our model, taking available literature into consideration [10, 21, 26], the following priors will be used:

$$(\boldsymbol{\xi} \mid \lambda) \sim \mathcal{N}_{dim(\boldsymbol{\xi})}(\mathbf{0}, \boldsymbol{Q}^{-1}) \tag{19}$$
$$(\lambda \mid \delta) \sim \mathcal{G}_\lambda(\nu/2, (\nu\delta)/2) \tag{20}$$
$$\gamma \sim \mathcal{G}_\gamma(\alpha_\gamma, \beta_\gamma) \tag{21}$$
$$\delta \sim \mathcal{G}_\delta(\alpha_\delta, \beta_\delta). \tag{22}$$

As the covariance matrix of the conditional prior distribution of the latent vector in Eq. (19), we will consider inverse of the following precision matrix:

$$\boldsymbol{Q}(\lambda) = \begin{bmatrix} \lambda\boldsymbol{\mathcal{P}} & 0 \\ 0 & \tau\boldsymbol{I}_P \end{bmatrix}, \tag{23}$$

with $\tau = 10^{-5}$. Top-left block is a $K \times K$ penalty matrix modified by a smoothing parameter, λ, which explains why the prior in Eq. (19) is dependent on λ; bottom-right block is a diagonal $P \times P$ precision matrix of the vector of regression coefficients $\boldsymbol{\beta}$. If not stated otherwise, the following constants will be used in the priors: $\alpha_\gamma = \beta_\gamma = \alpha_\delta = \beta_\delta = 10^{-4}$, and $\nu = 2$.

2.3 Laplace-approximated conditional posterior of the latent vector; the Newton-Raphson Algorithm

With the building blocks developed in the two previous sections, we are ready to follow to the conditional posterior distribution of the latent vector. By Bayes' theorem, the conditional posterior distribution of the latent vector $\boldsymbol{\xi}$ is given by:

$$\begin{aligned} p(\boldsymbol{\xi} \mid \lambda, \gamma, D) &\propto \mathcal{L}(\boldsymbol{\xi}, \gamma; D) \times p(\boldsymbol{\xi} \mid \lambda) \\ &\propto \exp\Big(\sum_{i=1}^{\mathcal{I}} \ell_i(\boldsymbol{\xi}, \gamma; D) - \frac{1}{2}\boldsymbol{\xi}^T \boldsymbol{Q} \boldsymbol{\xi}\Big). \end{aligned} \tag{24}$$

Next, we define the gradient and the Hessian of the log-likelihood as the respective sums of the contributions across all $\mathcal{I}$ subgroups:

$$\nabla \ell(\boldsymbol{\xi}) := \sum_{i=1}^{\mathcal{I}} \nabla \ell_i(\boldsymbol{\xi}, \gamma; D) \tag{25}$$

$$\nabla^2 \ell(\boldsymbol{\xi}) := \sum_{i=1}^{\mathcal{I}} \nabla^2 \ell_i(\boldsymbol{\xi}, \gamma; D), \tag{26}$$

where the contributions of an i-th subgroup to the gradient and the Hessian were developed previously (Eqs. (14) - (18)). Taking Eqs. (24) - (26) together, the gradient and the Hessian of the conditional log-posterior become, respectively:

$$\nabla \ln\ p(\boldsymbol{\xi} \mid \lambda, \gamma, D) = \nabla \ell(\boldsymbol{\xi}) - \boldsymbol{Q}\boldsymbol{\xi} \tag{27}$$

$$\nabla^2 \ln\ p(\boldsymbol{\xi} \mid \lambda, \gamma, D) = \nabla^2 \ell(\boldsymbol{\xi}) - \boldsymbol{Q}. \tag{28}$$

With those terms, the mode of conditional posterior distribution of the latent vector can be approximated, following the idea of Laplace, with the Newton-Raphson Algorithm (NRA):

$$\boldsymbol{\xi}^{(c+1)} = \boldsymbol{\xi}^{(c)} - \Big(\nabla^2 \ell(\boldsymbol{\xi})\Big|_{\boldsymbol{\xi}^{(c)}} - \boldsymbol{Q}\Big)^{-1} \Big(\nabla \ell(\boldsymbol{\xi})\Big|_{\boldsymbol{\xi}^{(c)}} - \boldsymbol{Q}\boldsymbol{\xi}^{(c)}\Big). \tag{29}$$

Briefly, to approximate the mode at the $(c+1)$-th iteration, the algorithm uses the c-th iteration approximation, and the first and the second derivative of the log-posterior distribution evaluated at this approximation. Upon convergence, which might be assessed by the Euclidean distance of the gradient from the origin, the NRA returns the mode, $\boldsymbol{\xi}^*(\lambda, \gamma)$, and the covariance matrix, $\boldsymbol{\Sigma}^*(\lambda, \gamma) = -\Big([\nabla^2 \ln\ p(\boldsymbol{\xi} \mid \lambda, \gamma, D)]^*\Big)^{-1}$, of the Laplace-approximated posterior distribution of the latent vector, both conditional on the penalty parameter and the parameter controlling the distribution of the

frailty variable. Taken together, Laplace approximation of the conditional posterior distribution of the latent vector becomes:

$$p_G(\boldsymbol{\xi} \mid \lambda, \gamma, D) = \mathcal{N}_{dim(\boldsymbol{\xi})}\Big(\boldsymbol{\xi}^*(\lambda,\gamma),\ \boldsymbol{\Sigma}^*(\lambda,\gamma)\Big). \tag{30}$$

2.4 Joint marginal posterior distribution of the hyperparameters; sampling-free LPS algorithm

In the next step, we optimize the hyperparameters λ and γ, which finalizes setting a sampling-free algorithm. Following Gressani *et al.* [23], we define the hyperparameter vector, $(\boldsymbol{\eta}^T, \delta)^T$, where $\boldsymbol{\eta} := (\lambda, \gamma)^T$. Then the joint marginal posterior becomes:

$$p(\boldsymbol{\eta}, \delta \mid D) \propto \frac{\mathcal{L}(\boldsymbol{\xi}, \gamma; D) p(\boldsymbol{\xi}|\lambda) p(\lambda|\delta) p(\delta) p(\gamma)}{p_G(\boldsymbol{\xi}|\lambda, \gamma, D)}, \tag{31}$$

where the approximated posterior distribution in the denominator is given by Eq. (30). From there, since $p(\lambda|\delta)$, $p(\delta)$, and $p(\gamma)$ are Gamma distributions, δ can be integrated out, leaving:

$$\begin{aligned} p(\boldsymbol{\eta} \mid D) &= \int_0^\infty p(\boldsymbol{\eta}, \delta \mid D)\, d\delta \\ &\propto \lambda^{\frac{K+\nu}{2}-1}\Big(\frac{\nu\lambda}{2} + \beta_\delta\Big)^{-(\frac{\nu}{2}+\alpha_\delta)} \gamma^{\alpha_\gamma - 1} \\ &\times \det(\boldsymbol{\Sigma}^*(\boldsymbol{\eta}))^{\frac{1}{2}} \exp\Big(\ell(\boldsymbol{\xi}^*(\boldsymbol{\eta}), \gamma; D) - \frac{\lambda}{2}\boldsymbol{\xi}^{*T}(\boldsymbol{\eta})\boldsymbol{Q}\boldsymbol{\xi}^*(\boldsymbol{\eta}) - \beta_\gamma\gamma\Big). \end{aligned} \tag{32}$$

To improve numerical stability, it is better to work with transformed variables: $\widetilde{\lambda} = \ln(\lambda)$, and $\widetilde{\gamma} = \ln(\gamma)$, which defines vector $\widetilde{\boldsymbol{\eta}} := (\widetilde{\lambda}, \widetilde{\gamma})^T$. Then, considering Jacobian of this transformation, given by $\exp(\widetilde{\lambda} + \widetilde{\gamma})$, approximated log-posterior becomes:

$$\begin{aligned} \ln p(\widetilde{\boldsymbol{\eta}} \mid D) \doteq\ & \frac{1}{2}\ln\det(\boldsymbol{\Sigma}^*(\widetilde{\boldsymbol{\eta}})) + \frac{\widetilde{\lambda}}{2}(K+\nu) + \alpha_\gamma\widetilde{\gamma} \\ & - (\frac{\nu}{2} + \alpha_\delta)\ \ln(\frac{\nu}{2}\exp(\widetilde{\lambda}) + \beta_\delta) \\ & + \ell(\boldsymbol{\xi}^*(\widetilde{\boldsymbol{\eta}}), \exp(\widetilde{\gamma}); D) - \frac{\exp(\widetilde{\lambda})}{2}\boldsymbol{\xi}^{*T}(\widetilde{\boldsymbol{\eta}})\boldsymbol{Q}\boldsymbol{\xi}^*(\widetilde{\boldsymbol{\eta}}) - \beta_\gamma\exp(\widetilde{\gamma}). \end{aligned} \tag{33}$$

Numerical optimization, for example with the Nelder-Mead Algorithm (NMA) [33], yields $\widetilde{\boldsymbol{\eta}}^* = (\widetilde{\lambda}^*, \widetilde{\gamma}^*)^T$. Note that every iteration in the NMA includes full NRA in order to obtain mode of $\boldsymbol{\xi}$ conditional on $\widetilde{\boldsymbol{\eta}}$ at the given NMA iteration, which makes the whole procedure computationally involved. After convergence of the NMA, $\widetilde{\boldsymbol{\eta}}^*$ is plugged into Eq. (30) to run the final NRA leading to the Maximum a Posteriori (MAP) estimate of the latent vector, $\widehat{\boldsymbol{\xi}} = \boldsymbol{\xi}^*(\widetilde{\boldsymbol{\eta}}^*)$, which gives the approximated posterior distribution in Eq. (30) conditional on the optimized $\widetilde{\boldsymbol{\eta}}^*$. This finalizes development of the sampling-free LPS algorithm, combining P-splines smoother and Laplace approximation:

$$p_G(\boldsymbol{\xi} \mid \widetilde{\boldsymbol{\eta}}^*, D) = \mathcal{N}_{dim(\boldsymbol{\xi})}\Big(\boldsymbol{\xi}^*(\widetilde{\boldsymbol{\eta}}^*),\ \boldsymbol{\Sigma}^*(\widetilde{\boldsymbol{\eta}}^*)\Big). \tag{34}$$

2.5 Conditional posterior distributions and the Metropolis-Adjusted Langevin Algorithm within the Gibbs Sampler

As an alternative approach, the Gibbs Sampler (GS) with a MALA step will be now developed, relying on the conditional posterior distributions of all parameters in the model. To set it up, the conditional posteriors for λ and δ are straightforward, since they belong to the Gamma family and can de derived analytically:

$$\begin{aligned} p(\lambda \mid \boldsymbol{\xi}, \delta, D) &\propto p(\boldsymbol{\xi} \mid \lambda)\, p(\lambda \mid \delta) \\ &\propto \underbrace{[\det(\boldsymbol{\Sigma}^*(\lambda,\gamma))]^{-1/2}}_{\propto \lambda^{\frac{K}{2}}} \exp\Big(-\frac{1}{2}\boldsymbol{\xi}^T(\boldsymbol{\Sigma}^*(\lambda,\gamma))^{-1}\boldsymbol{\xi}\Big) \times \lambda^{\frac{\nu}{2}-1}\exp(-\frac{\lambda\nu\delta}{2}) \\ &\propto \lambda^{\frac{\nu+K}{2}-1}\exp\Big(-\frac{1}{2}\lambda(\boldsymbol{\xi}^T\widetilde{\boldsymbol{\mathcal{P}}}\boldsymbol{\xi}+\nu\delta)\Big) \\ &\propto \mathcal{G}(\frac{\nu+K}{2},\ \frac{\boldsymbol{\xi}^T\widetilde{\boldsymbol{\mathcal{P}}}\boldsymbol{\xi}+\nu\delta}{2}) \end{aligned} \tag{35}$$

$$\begin{aligned} p(\delta \mid \lambda, D) &\propto p(\lambda \mid \delta)\, p(\delta) \\ &\propto \mathcal{G}(\frac{\nu}{2}+\alpha_\delta,\ \frac{\lambda\nu}{2}+\beta_\delta), \end{aligned} \tag{36}$$

where $\widetilde{\boldsymbol{\mathcal{P}}}$ in Eq. (35) is a square matrix of the dimension $\dim(\boldsymbol{\xi})$ with the $K \times K$ penalty matrix, $\boldsymbol{\mathcal{P}}$ (used also in Eq. (23)), as the top-left block and zeroes elsewhere. On the other hand, however, the conditional posterior for γ cannot be assigned to any known parametric family:

$$\begin{aligned} p(\gamma \mid \boldsymbol{\xi}, D) &\propto \mathcal{L}(\boldsymbol{\xi}, \gamma; D)\, p(\gamma) \\ &\propto \prod_{i=1}^{\mathcal{I}}\Big(\frac{\gamma^\gamma}{\Gamma(\gamma)} \times \widetilde{\beta}_i^{-\widetilde{\alpha}_i}\Gamma(\widetilde{\alpha}_i)\Big)\gamma^{\alpha_\gamma-1}\exp(-\gamma\beta_\gamma) \\ &\propto \prod_{i=1}^{\mathcal{I}}\Big(\widetilde{\beta}_i^{-\widetilde{\alpha}_i}\Gamma(\widetilde{\alpha}_i)\Big)\gamma^{\mathcal{I}\gamma+\alpha_\gamma-1}\Gamma(\gamma)^{-\mathcal{I}}\exp(-\gamma\beta_\gamma). \end{aligned} \tag{37}$$

Therefore, we propose Metropolis-within-Gibbs strategy [39], and following Gressani *et al.* [20] we will be sampling γ together with the elements of the latent vector, $\boldsymbol{\xi}$, in the MALA step. We define a vector $\boldsymbol{\zeta} := (\boldsymbol{\xi}^T, \gamma)^T = (\boldsymbol{\theta}^T, \boldsymbol{\beta}^T, \gamma)^T$ containing the amplitudes, the regression coefficients, and the parameter controlling the frailty variable. Then the conditional posterior for $\boldsymbol{\zeta}$ becomes:

$$\begin{aligned} p(\boldsymbol{\zeta} \mid \lambda, D) &\propto \mathcal{L}(\boldsymbol{\zeta}; D) \times p(\boldsymbol{\xi} \mid \lambda) \times p(\gamma) \\ &\propto \exp\Big(\ell(\boldsymbol{\zeta}; D) - \frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{Q}\boldsymbol{\xi}\Big) \times \gamma^{\alpha_\gamma-1}\exp(-\beta_\gamma\gamma) \\ &\propto \exp\Big(\ell(\boldsymbol{\zeta}; D) - \frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{Q}\boldsymbol{\xi} - \beta_\gamma\gamma\Big)\gamma^{\alpha_\gamma-1}. \end{aligned} \tag{38}$$

Since this sampling is performed from (multivariate) Normal distribution, it is convenient to transform γ parameter (which is strictly positive) to a new parameter, $\widetilde{\gamma} = \ln(\gamma)$, which lives on the entire real line. Correspondingly, we define $\widetilde{\boldsymbol{\zeta}} := (\boldsymbol{\theta}^T, \boldsymbol{\beta}^T, \widetilde{\gamma})^T$ and, taking into account the Jacobian of this transformation, $|d\gamma/d\widetilde{\gamma}| = \exp(\widetilde{\gamma})$, we obtain the conditional posterior:

$$p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D) \propto \exp\Big(\ell(\widetilde{\boldsymbol{\zeta}}; D) - \frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{Q}\boldsymbol{\xi} - \beta_\gamma\exp(\widetilde{\gamma}) + \alpha_\gamma\widetilde{\gamma}\Big). \tag{39}$$

Then $\ell(\widetilde{\boldsymbol{\zeta}}; D) = \sum_{i=1}^{\mathcal{I}} \ell_i(\widetilde{\boldsymbol{\zeta}}; D)$, and hence:

$$\ell(\widetilde{\boldsymbol{\zeta}}; D) \doteq \sum_{i=1}^{\mathcal{I}} \left\{ \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} \left(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij} \right) + \widetilde{\gamma} \exp(\widetilde{\gamma}) - \ln\Big(\Gamma(\exp(\widetilde{\gamma}))\Big) + \ln\Big(\Gamma(\breve{\alpha}_i)\Big) - \breve{\alpha}_i \ln(\breve{\beta}_i) \right\}. \quad (40)$$

For the implementation of this expression, we observe that the two "breved" terms, $\breve{\alpha}_i$ and $\breve{\beta}_i$, are algebraically identical to those in Eqs. (8) and (9), respectively (keeping in mind that $\gamma = \exp(\widetilde{\gamma})$), though expressed in terms of the transformed variable, $\widetilde{\gamma}$:

$$\breve{\alpha}_i = \exp(\widetilde{\gamma}) + \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} = \widetilde{\alpha}_i \quad (41)$$

$$\breve{\beta}_i = \exp(\widetilde{\gamma}) + \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm} = \widetilde{\beta}_i. \quad (42)$$

Assuming that the chain is currently in the c-th position, the candidate for the $\widetilde{\boldsymbol{\zeta}}$ vector for the $(c+1)$-th position is sampled from:

$$\widetilde{\boldsymbol{\zeta}}^{(prop)} \sim \mathcal{N}_{(K+P+1)}\Big(\widetilde{\boldsymbol{\zeta}}^{(c)} + 0.5\varrho \boldsymbol{\Sigma}_{\boldsymbol{L}} \nabla_{\widetilde{\zeta}} \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D)\Big|_{\widetilde{\boldsymbol{\zeta}}^{(c)}},\ \varrho \boldsymbol{\Sigma}_{\boldsymbol{L}} \Big), \quad (43)$$

where ϱ is a tuning parameter, developed in [23], helping to yield the optimal acceptance ratio of 0.57 [36], and the covariance matrix is given by

$$\boldsymbol{\Sigma}_{\boldsymbol{L}} = \begin{bmatrix} \boldsymbol{\Sigma}^*(\widetilde{\boldsymbol{\eta}}^*) & 0 \\ 0 & 1 \end{bmatrix}, \quad (44)$$

with $(K+P)$-dimensional covariance matrix $\boldsymbol{\Sigma}^*(\widetilde{\boldsymbol{\eta}}^*)$ as used in Eq. (34).

The gradient, $\nabla_{\widetilde{\zeta}} \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D)$, can be decomposed as:

$$\nabla_{\widetilde{\zeta}} \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D) = \Big(\nabla_{\boldsymbol{\theta}}^T \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D), \nabla_{\boldsymbol{\beta}}^T \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D), \frac{\partial \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D)}{\partial \widetilde{\gamma}} \Big)^T, \quad (45)$$

where the partial derivatives wrt the elements of $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ can be obtained from Eqs. (14), (15), (25) and (27), substituting $\exp(\widetilde{\gamma})$ for γ. The last element of the gradient, wrt $\widetilde{\gamma}$, becomes:

$$\frac{\partial \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D)}{\partial \widetilde{\gamma}} = \exp(\widetilde{\gamma}) \left(\mathcal{I}\Big(\widetilde{\gamma} + 1 - \digamma(\exp(\widetilde{\gamma})) \Big) + \sum_{i=1}^{\mathcal{I}} \Big(\digamma(\breve{\alpha}_i) - \ln(\breve{\beta}_i) - \frac{\breve{\alpha}_i}{\breve{\beta}_i} \Big) - \beta_\gamma \right) + \alpha_\gamma, \quad (46)$$

where $\digamma(.)$ is a digamma function. Detailed derivation is given in **Appendix 1**.

After sampling a candidate, $\widetilde{\boldsymbol{\zeta}}^{(prop)}$, the acceptance probability is given by:

$$\pi(\widetilde{\boldsymbol{\zeta}}^{(prop)}, \widetilde{\boldsymbol{\zeta}}^{(c)}) = \min \left\{ 1, \frac{p(\widetilde{\boldsymbol{\zeta}}^{(prop)} \mid D)}{p(\widetilde{\boldsymbol{\zeta}}^{(c)} \mid D)} \times \frac{q(\widetilde{\boldsymbol{\zeta}}^{(prop)}, \widetilde{\boldsymbol{\zeta}}^{(c)})}{q(\widetilde{\boldsymbol{\zeta}}^{(c)}, \widetilde{\boldsymbol{\zeta}}^{(prop)})} \right\}, \quad (47)$$

with the ratio of the proposal densities, $q(\widetilde{\boldsymbol{\zeta}}^{(prop)}, \widetilde{\boldsymbol{\zeta}}^{(c)})/q(\widetilde{\boldsymbol{\zeta}}^{(c)}, \widetilde{\boldsymbol{\zeta}}^{(prop)})$, derived in details in [23].

The candidate is accepted if $\pi(\widetilde{\boldsymbol{\zeta}}^{(prop)}, \widetilde{\boldsymbol{\zeta}}^{(c)}) > \widetilde{u}$, where $\widetilde{u}$ is sampled from continuous uniform distribution, $\mathcal{U}(0, 1)$, and rejected otherwise. For a better numerical stability, the calculation of the acceptance decision is performed after logarithmization. Taken together, Eqs. (35), (36), (43), and (47) form the MALA within the Gibbs Sampler. The box summarizes the developed algorithms.

(1) Optimize Eq. (30) to obtain λ^* and γ^* with the NMA; each NMA optimization step includes NRA (Eq. (29)) which, at convergence, returns $\boldsymbol{\xi}^*(\lambda, \gamma)$, *i.e.* the mode conditional on λ and γ of the current NMA-step.

(2) Run NRA once more using λ^* and γ^* obtained at the convergence of the NMA in the Step (1). This also gives the optimized Hessian, and hence the covariance matrix $\boldsymbol{\Sigma}^*(\widetilde{\boldsymbol{\eta}}^*)$ in Eq. (34).

(3) Set initial values for $\boldsymbol{\xi}^{(0)}$, $\lambda^{(0)}$, $\delta^{(0)}$, $\gamma^{(0)}$, and from hence $\widetilde{\gamma}^{(0)} = \ln(\gamma^{(0)})$ and $\widetilde{\boldsymbol{\zeta}}^{(0)} = (\boldsymbol{\xi}^{(0)^T}, \widetilde{\gamma}^{(0)})^T$.

(4) Sample a candidate $\widetilde{\boldsymbol{\zeta}}^{(prop)}$ with Eq. (43), making use of the gradient in Eq. (45) and the covariance matrix in Eq. (44) [which uses $\boldsymbol{\Sigma}^*(\widetilde{\boldsymbol{\eta}}^*)$ from the Step (2)]. Note that the elements of the gradient in Eq. (45) wrt $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ are algebraically identical to those in Eqs. (14) and (15) as $\gamma^{(c)} = \exp(\widetilde{\gamma}^{(c)})$; formally: $\left(\nabla^T_{\boldsymbol{\theta}} \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D), \nabla^T_{\boldsymbol{\beta}} \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D)\right)^T = \nabla \ln\ p(\boldsymbol{\xi} \mid \lambda, \gamma, D)$, compare Eqs. (45) and (27).

(5) Accept/reject the candidate using the Metropolis criterion. If accepted, $\widetilde{\boldsymbol{\zeta}}^{(c+1)} = \widetilde{\boldsymbol{\zeta}}^{(prop)}$, otherwise $\widetilde{\boldsymbol{\zeta}}^{(c+1)} = \widetilde{\boldsymbol{\zeta}}^{(c)}$.

(6) Sample $\lambda^{(c+1)}$ with Eq. (35), making use of $\boldsymbol{\xi}^{(c+1)}$ [which consists of the first $K + P$ elements of the $\widetilde{\boldsymbol{\zeta}}^{(c+1)}$ vector obtained in the Step (5)] and $\delta^{(c)}$. Sample $\delta^{(c+1)}$ with Eq. (36), making use of $\lambda^{(c+1)}$.

(7) Get back to the Step (4) until enough elements of the chain are sampled [iterate Steps (4) - (6)].

2.6 A small simulation study

The algorithms developed in the previous sections were implemented in `R` programming language (version 4.3.2 *Eye Holes*; R Core Team, 2023) with details of the implementation given in **Appendix 1**. To test the sampling-free LPS algorithm, 300 datasets were generated under three settings ($S = 100$ datasets per setting) with a function `simfrail` provided by Dr. Gressani[2] [18]. Each dataset consisted of 20 clusters, 20 observations per cluster, and contained three covariates with the true regression coefficients set to $\beta_1 = 1.0$, $\beta_2 = 0.1$, and $\beta_3 = -1.0$, respectively. The response variable,

[2]Not to be mixed with a function of the same name available in the `frailtySurv` package.

time-to-event, was drawn from the Weibull distribution, $f(t) = (a/b^a)t^{a-1}\exp(-(t/b)^a)$, with $a = 2.4$ and $b = 4$. The three settings were: (A) $\gamma = 5$ and right censoring rate (CR) 10%, (B) $\gamma = 3$ $CR = 10\%$, and (C) $\gamma = 5$ $CR = 30\%$. In addition, a simulation was performed with 100 unbalanced datasets of 200 clusters of size 1 and 100 clusters of size 2 and $CR = 30\%$ to mimic the unbalanced dataset of the CGD study, which is considered in the next section.

Each dataset was analyzed with the LPS algorithm and, to confirm plausibility of the results, with a widely-used frequentist method based on the Expectation-Maximization Algorithm (EMA), implemented in the `emfrail` function of the `frailtyEM` package [6]. Baseline survival was estimated with 30 cubic B-splines with second-order penalty on a grid of 300 equidistant segments covering $[0,\ t_{max}]$, where t_{max} was the largest event or censoring time for a given dataset. The NRA in Eq. (29) was considered converged at the c-th iteration if $\| \nabla \ln p(\boldsymbol{\xi}|\lambda,\gamma,D)|_{\boldsymbol{\xi}^{(c)}} \|_2 \leq 10^{-6}$, where $\|\cdot\|_2$ is the Euclidean norm. If not stated otherwise, the following initial values were used: $\lambda = 100$, $\gamma = 5$, and $\boldsymbol{\xi} = (0.8, ..., 0.8)^T$.

The frequentist properties of the Bayesian estimators of the regression coefficients, β_p, were assessed as follows. The empirical bias was defined as the average of the difference between the estimate and its true value across S replications:

$$Bias_{\widehat{\beta}_p} := S^{-1}\sum_{s=1}^{S}\Big(\widehat{\beta}_p^{(s)} - \beta_p\Big). \tag{48}$$

The empirical variance was defined as the sample variance across S replications:

$$Var_{\widehat{\beta}_p} := (S-1)^{-1}\sum_{s=1}^{S}\Big(\widehat{\beta}_p^{(s)} - \bar{\widehat{\beta}}_p\Big)^2. \tag{49}$$

The Mean Square Error (MSE) was calculated as the sum of the square of the bias and the variance. Finally, the ϕ-level coverage probability (for $\phi = 90\%$ and 95%) was calculated as the average of the indicator $\mathbb{I}(\cdot)$ that takes the value 1 if the constructed interval ($CI_{\phi,\,p}$) includes the true parameter and 0 otherwise:

$$CP_{\phi,\,p} := S^{-1}\sum_{s=1}^{S}\mathbb{I}\Big(\beta_p \in CI_{\phi,\,p}^{(s)}\Big). \tag{50}$$

MALA sampler was applied on a 100 simulated dataset consisting of 20 clusters, 20 observations per cluster and right-censoring of 10%. Before running MALA, in each dataset LPS step was performed to obtain the initial values for $\boldsymbol{\xi}^{(0)}$, $\lambda^{(0)}$, $\delta^{(0)}$, $\gamma^{(0)}$, and the covariance matrix, $\boldsymbol{\Sigma_L}$. The algorithm was iterated 10,000 times with burn-in of 1,000 iterations.

Finally, execution time of the algorithms was assessed in three scenarios, 10 simulated datasets per scenario (all with γ set to 5 and $CR = 10\%$): a dataset with 4 clusters of 100 observations, 20 clusters of 20 observations, and 40 clusters of 10 observations.

2.7 Recurrence of serious infections in a randomized trial of Interferon Gamma treatment for Chronic Granulotomous Disease

Next, the sampling-free LPS algorithm and the MALA sampler developed in the previous sections were applied to model the data from a controlled randomized clinical trial analyzed previously with frequentist methodology by *The International CGD Cooperative Study* [24]. The patient-level raw data of the trail are available in the literature [6, 17].

The trial setting is detailed elsewhere [24]. Briefly, 128 patients were randomized into an active arm ($n = 63$, mean age 14.3 ± 11.1 years, 81% males), treated with IG, and a control arm ($n = 65$,

mean age 15.0 ± 9.6 years, 82% males), treated with placebo. Initially, the primary endpoint was time, measured in days, between the randomization and the incidence of a serious infection. During the study, it turned out that several patients in both arms developed more than one incidence; in such cases, multiple outcome variables per patient were recorded, namely the number of days between the end of a previous infection and the beginning of the next. This leads to a hierarchical structure of the data, with the observations nested within patients, which are treated as clusters.

The main scientific interest focuses on the question whether treatment with subcutaneous injections of Interferon Gamma changes the hazard of serious infections, compared to treatment with placebo. Formally:

$$H_0 : \beta_{treat} = 0 \quad vs. \quad H_a : \beta_{treat} \neq 0, \tag{51}$$

where β_{treat} is the treatment effect, expressed as logarithm of the hazard ratio.

The outcome variable was modelled as a function of two covariates, the indicator of the treatment arm (1 if IG and 0 otherwise) and sex (1 if female). Baseline survival was estimated with 30 cubic B-splines with second-order penalty on a grid of 300 equidistant segments covering $[0,\ t_{max}]$, where t_{max} was the largest event or censoring time. The criterion for the NRA convergence and the initial values for λ and γ were as in those in the previous section. The initial latent vector of the amplitudes and the regression coefficients was $\boldsymbol{\xi} = (-3, ..., -3)^T$.

In the MALA part, the initial values were those obtained at the convergence of the LPS. The algorithm was run with 25,000 iterations with burn-in of 10,000 iterations.

3 Results

3.1 Results of the simulation study

Summary statistics of the LPS are presented in Table 1. The results clearly indicate that the averages of the estimated regression coefficients, $\widehat{\beta}_1$, $\widehat{\beta}_2$, and $\widehat{\beta}_3$, are very close to the target values, and that intra- and inter-dataset standard errors for particular estimates are virtually identical. On the other hand, there seems to exist discrepancies between the target values for the γ parameter and the estimates obtained in the simulations. This points at a potential bias and was explored further.

Distributions of the four parameters of interest are presented also in boxplots in Figure 3, with red horizontal bars indicating the target values; Table 2-4 show further frequentist statistics and comparisons of the outcomes of the LPS model with those from the EMA of the `emfrail` function. Also from those results it becomes obvious that the estimations of the regression parameters in both algorithms are virtually identical, acceptably precise, and have a negligible bias, which is reflected by very low MSE. Interestingly, however, both methods show bias of the estimates for the γ parameter, apparently more pronounced in the setting with a larger censoring rate [compare settings (A) and (C)]. Since this bias is similar in both methods, it may be concluded that it is not specific for the LPS algorithm developed in this study. NA entries for the γ parameters are a consequence of the setting with γ parameter included into the hyperparameters vector and not the latent vector, which leads to the LPS algorithm returning the point estimate but not the variance of the estimator.

Left panels of Figure 4 present direct comparison of the variances of the frailty ($1/\gamma$) obtained with `emfrail` function (horizontal axes) and the LPS algorithm (vertical axes). In all 300 datasets, the estimated parameters are virtually identical in both methods, although very dispersed across the datasets, which reconfirms plausibility of the LPS method and points once again that the bias of those estimates is not specific for either of the methods. Right panels of Figure 4 presents baseline survival curves obtained in the simulations (green curves), median curves (red), and the "true" curves calculated from the Weibull distribution.

Stability of the LPS algorithm for the γ parameter was tested on one dataset ($CR = 10\%$ $\gamma = 5$) with five different initial values for γ: 2, 5, 10, 20, and 50. The results at the convergence of the LPS algorithm were, respectively: 5.498, 5.508, 5.506, 5.509, and 5.504, suggesting acceptable stability of the algorithm for alterations of the initial values of γ.

Table 1: Results of the simulations. Presented are averages of one hundred simulations for each setting and their standard errors. CR, censoring rate.

Simulation setting	$\widehat{\gamma}$	$\widehat{\beta_1}$ (1.0)	$\widehat{\beta_2}$ (0.1)	$\widehat{\beta_3}$ (-1.0)	$\sigma(\widehat{\beta_1})$	$\sigma(\widehat{\beta_2})$	$\sigma(\widehat{\beta_3})$
(A) $CR = 10\%$, $\gamma = 5$	8.16	0.978	0.099	-0.978	0.069	0.056	0.069
	(7.01)	(0.061)	(0.056)	(0.075)	(0.003)	(0.003)	(0.003)
(B) $CR = 10\%$, $\gamma = 3$	4.04	0.970	0.089	-0.980	0.069	0.056	0.070
	(1.70)	(0.056)	(0.049)	(0.066)	(0.003)	(0.003)	(0.003)
(C) $CR = 30\%$, $\gamma = 5$	9.89	0.969	0.097	-0.975	0.076	0.063	0.077
	(7.15)	(0.070)	(0.062)	(0.079)	(0.004)	(0.003)	(0.003)

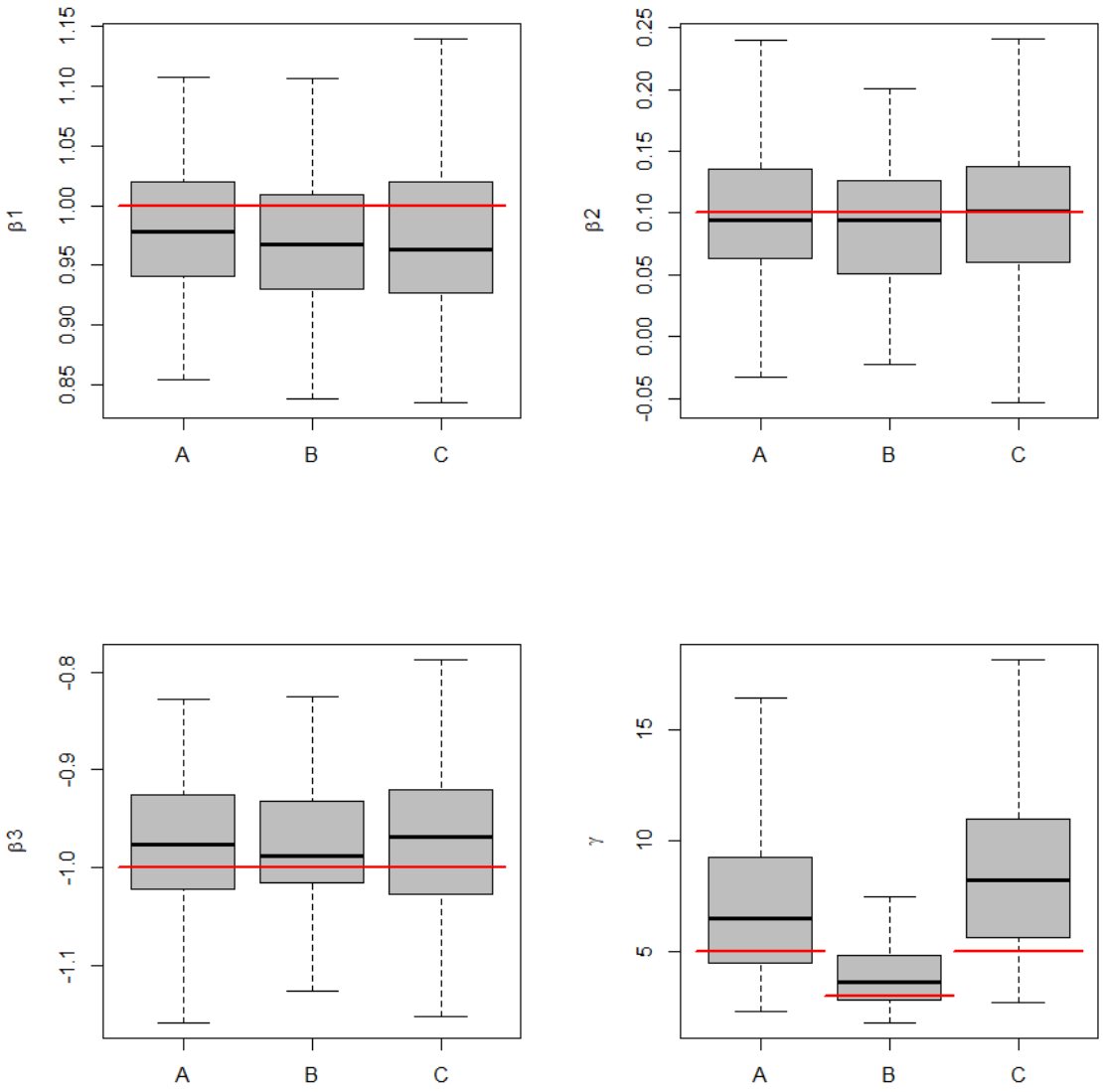


Figure 3: Results of the simulation of one hundred datasets per setting. Red bars indicate the target values. Settings: (A) $CR = 10\%$, $\gamma = 5$; (B) $CR = 10\%$, $\gamma = 3$; (C) $CR = 30\%$, $\gamma = 5$.

Table 2: Frequentist statistics of the simulations in the setting (A) $CR = 10\%$, $\gamma = 5$. MSE, Mean Square Error, CP_ϕ, ϕ Coverage Probability.

Parameter	$Bias^2$	Var	MSE	$CP_{90\%}$	$CP_{95\%}$
			LPS		
β_1	0.0005	0.0038	0.0042	93%	96%
β_2	0.0000	0.0032	0.0032	90%	94%
β_3	0.0005	0.0057	0.0061	86%	91%
γ	9.956	49.17	59.12	NA	NA
			emfrail		
β_1	0.0000	0.0040	0.0040	93%	98%
β_2	0.0000	0.0034	0.0034	89%	95%
β_3	0.0000	0.0060	0.0060	88%	93%
γ	12.02	58.15	70.17	NA	NA

Table 3: Frequentist statistics of the simulations in the setting (B) $CR = 10\%$, $\gamma = 3$. MSE, Mean Square Error, CP_ϕ, ϕ Coverage Probability.

Parameter	$Bias^2$	Var	MSE	$CP_{90\%}$	$CP_{95\%}$
			LPS		
β_1	0.0009	0.0031	0.0040	94%	96%
β_2	0.0001	0.0024	0.0025	90%	99%
β_3	0.0004	0.0044	0.0048	89%	95%
γ	0.917	2.905	3.822	NA	NA
			emfrail		
β_1	0.0000	0.0034	0.0034	94%	97%
β_2	0.0000	0.0025	0.0026	90%	99%
β_3	0.0000	0.0046	0.0046	89%	95%
γ	0.798	3.330	4.128	NA	NA

Table 4: Frequentist statistics of the simulations in the setting (C) $CR = 30\%$, $\gamma = 5$. MSE, Mean Square Error, CP_ϕ, ϕ Coverage Probability.

Parameter	$Bias^2$	Var	MSE	$CP_{90\%}$	$CP_{95\%}$
			LPS		
β_1	0.0010	0.0048	0.0058	91%	95%
β_2	0.0000	0.0039	0.0039	88%	93%
β_3	0.0006	0.0063	0.0069	90%	95%
γ	23.95	51.11	75.06	NA	NA
			emfrail		
β_1	0.0000	0.0052	0.0053	92%	96%
β_2	0.0000	0.0042	0.0042	86%	93%
β_3	0.0000	0.0068	0.0068	88%	91%
γ	27.58	62.65	90.23	NA	NA

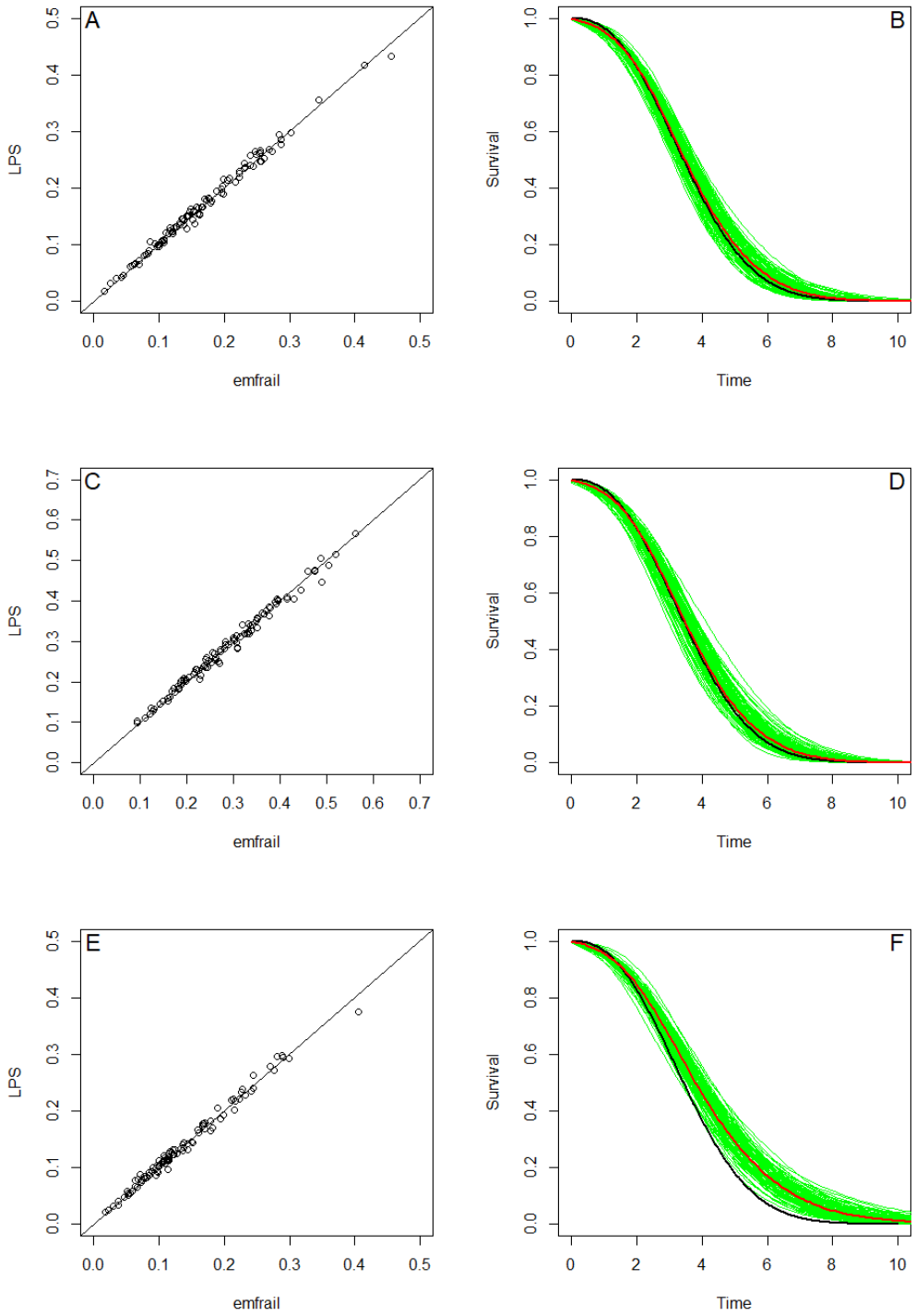


Figure 4: Left panels: Comparison of the frailty variance $(1/\gamma)$ in one hundred simulations per setting, with identity lines, $y = x$. Right panels: Baseline survival curves in the corresponding settings. Black curve represents the "true" survival from the Weibull distribution, $S_0(t) = \exp[-(t/b)^a]$; red curve represents the median survival curve. Panels: A & B, $CR = 10\%$, $\gamma = 5$; C & D, $CR = 10\%$, $\gamma = 3$; E & F, $CR = 30\%$, $\gamma = 5$.

Summary statistics of the posterior distributions from the MALA sampler are presented in Table 5. Proportion of the accepted proposals turned out to be, on average, almost ideally as expected (target, 0.57). Its small dispersion means that also the individual acceptance proportions, across the datasets, were close to the optimum. Similarly, averages of the regression coefficients were very close to the true values. In two datasets, outlying γ parameters were observed, and hence for that parameter we

report the median and the inter-quartile range instead of the average and the standard deviation. It is worth to emphasize that the dispersions of the regression coefficients (within- and between-datasets) are virtually the same as those in the LPS part (*cf.* SDs in Table 1 and Table 5).

Table 5: Posterior summary statistics of the MALA sampler (iterations 1,001:10,000). Presented are averages and standard deviations (median and inter- quartile range for γ) of 100 simulated datasets.

Parameter	Estimate	SD	$CP_{90\%}$	$CP_{95\%}$
Accept. ratio	0.572	0.010		
$\widehat{\beta}_1$	0.985	0.067	93%	95%
$\widehat{\beta}_2$	0.100	0.054	90%	92%
$\widehat{\beta}_3$	-0.989	0.060	94%	99%
$\sigma(\widehat{\beta}_1)$	0.071	0.003		
$\sigma(\widehat{\beta}_2)$	0.055	0.003		
$\sigma(\widehat{\beta}_3)$	0.071	0.003		
$\widehat{\gamma}$	6.111	[4.6; 9.1]	97%	100%

Table 6: Execution times of the algorithms. For the MALA sampler, 10,000 iterations were performed for each dataset.

Scenario	Aver. ($\pm$ SD) execution time [sec.]	
	LPS	MALA
4 clusters of 100 observations	4.5 (0.9)	174 (2.1)
20 clusters of 20 observations	11.3 (2.4)	212 (10.5)
40 clusters of 10 observations	39.8 (5.9)	368 (5.5)

Further, for illustrative purposes the traceplots (Figure 5) and the posterior distributions (Figure 6) are presented for one randomly chosen dataset. Traceplots of the regression parameters show a typical "thick-pen" shape, randomly exploring the whole domains without any particular pattern, whereas traceplots of the remaining parameters show some properties of the autocorrelation (particularly visible for λ on panel E) and/or occasional departures from their respective means (large peaks on panels D, E, and F). Posterior distributions show expected bell shapes for the regression parameters (panels A, B, and C) and characteristic skewed shapes of the Gamma distributions for the γ, λ, and particularly δ, parameters (panels D, E, and F, respectively). The latter parameter follows a posteriori virtually Exponential distribution, due to setting of the ν parameter equal to 2, which leads to the shape parameter in Eq. (36) practically equal to 1 (for $\nu = 4$, departure of the posterior distribution of δ from an Exponential distribution to more Gamma-looking distribution was observed, data not shown). Regarding estimations of the B-spline coefficients, Figure 7 presents the survival curve calculated with the the coefficients estimated in the MALA sampler, along with its 95% credible interval curves. Obviously, the "true" curve, resulting from the Weibull distribution, lies within this interval and, moreover, the survival curve from the LPS model of the same dataset virtually overlaps the MALA-curve.

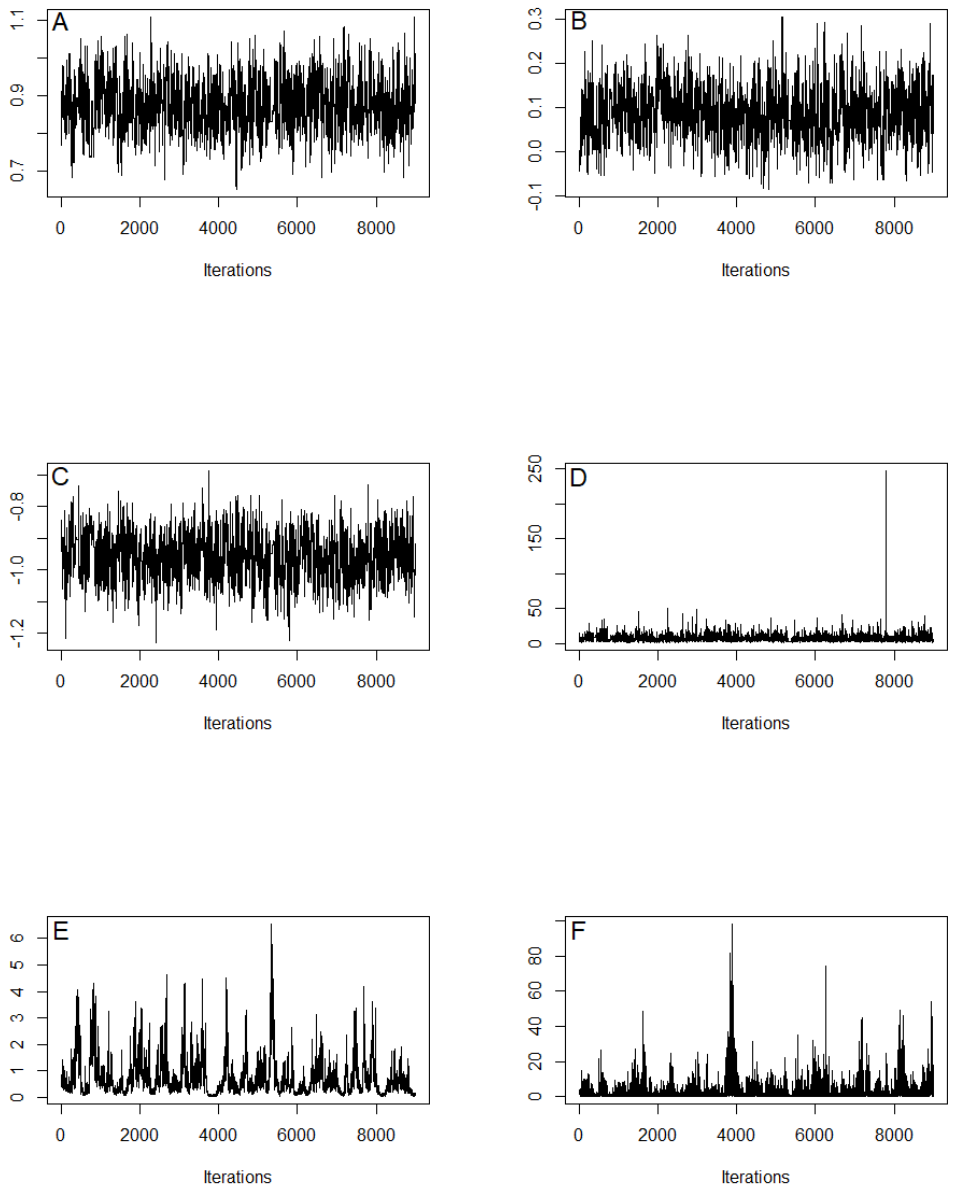


Figure 5: Traceplots of the parameters in the simulated dataset: β_1 (A), β_2 (B), β_3 (C), γ (D), λ (E), and δ (F).

Running times of the algorithms were assessed on a computer with a 2.4 GHz processor and 8 GB RAM, and are presented in Table 6. Assuming that 10,000 iterations of the MALA sampler are needed to reach the chain stability, the LPA method runs 10 – 40 times faster. For a given total size of a dataset, the execution times strongly depend on the number of the clusters and their sizes, with smaller number of larger clusters requiring much less time. Interestingly, there is also a non-negligible variability of the LPS run time across the datasets generated with the same scenario, which is reflected by the standard deviations within scenarios. The kernel of the implemented algorithms is a function computing the gradient and the Hessian; on average, execution of this kernel takes roughly 40ms, of which calculation of the gradient and the Hessian takes about 10ms and 30ms, respectively.

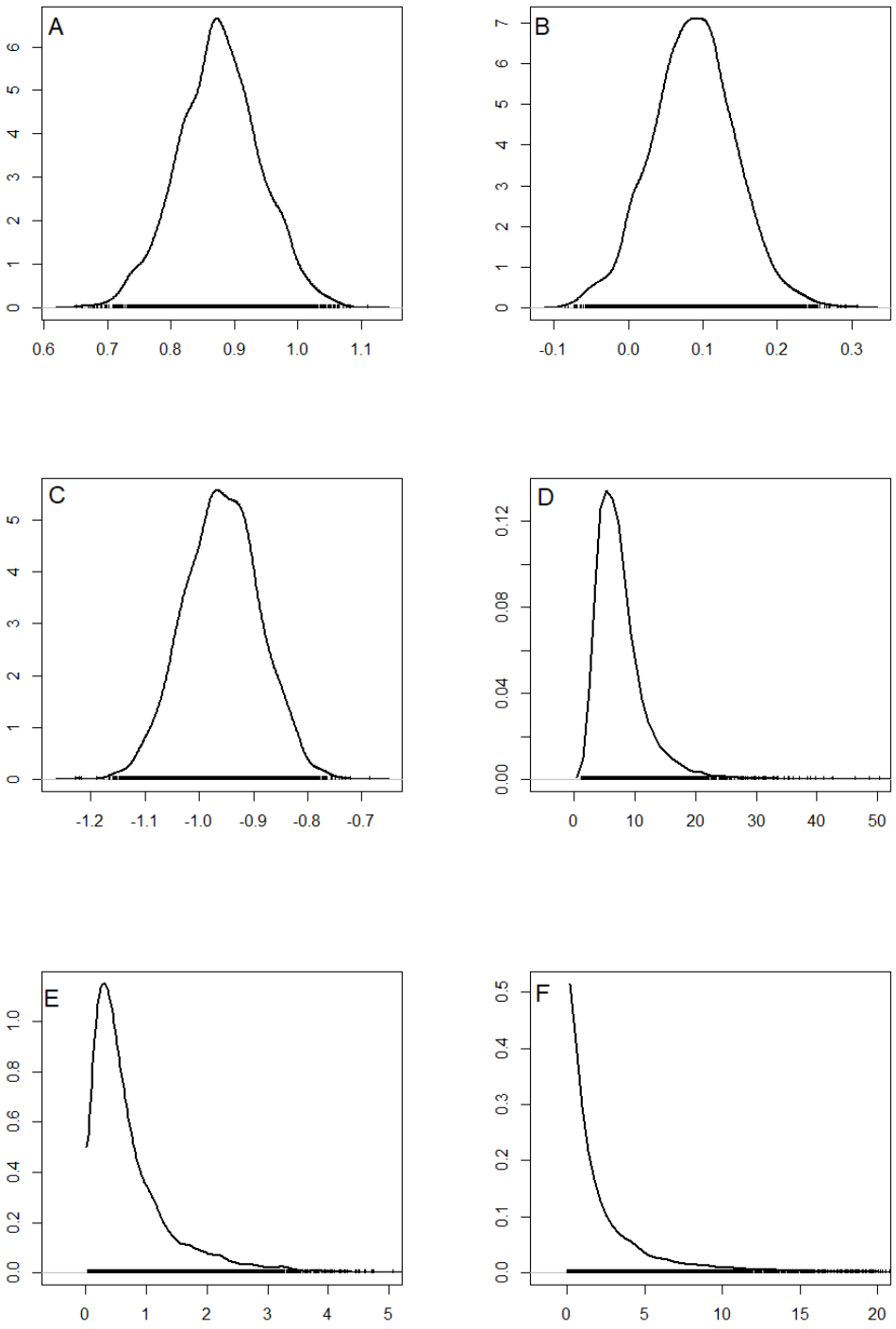


Figure 6: Posterior distributions: β_1 (A), β_2 (B), β_3 (C), γ (D), λ (E), and δ (F).

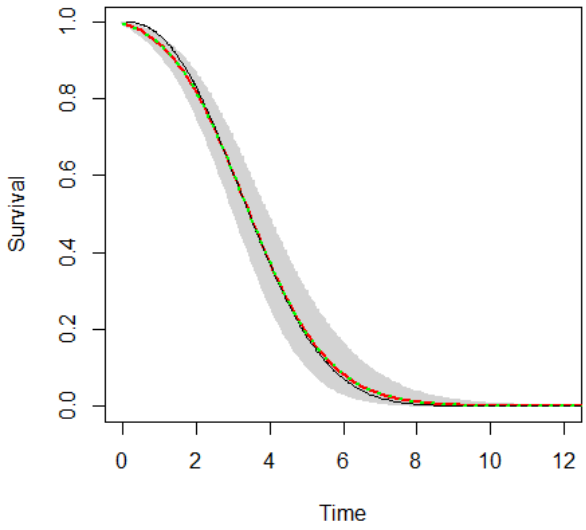


Figure 7: Baseline survival curves obtained from the MALA sampler (green dotted curve) and the LPS algorithm (red solid curve) in one simulated dataset (note that the two curves virtually overlap); grey area corresponds to the 95% credible interval of the LPS curve obtained with the delta method [20,21]. Black curve is the "true" survival curve from the Weibull distribution.

3.2 Results of the CGD study

Detailed descriptive statistics of the CGD study is given elsewhere [24]. From the perspective of the current modelling, it needs to be emphasized that the dataset is extremely unbalanced. In the treatment arm, for 49 subjects 1 observation was recorded, for 9 subjects 2 observations were recorded, for 4 subjects 3 observations were recorded, and for 1 subject 4 observations were recorded. In the placebo arm, for 35 subjects 1 observation was recorded, for 19 subjects 2 observations were recorded, for 4 subjects 3 observations were recorded, for another 4 subjects 4 observations were recorded, and 5, 6, and 8 observations were recorded for 1 subject each. This means that 66% of the clusters consisted of only one observation.

Regression coefficients of the treatment effect and the effect of sex, as well as the γ parameters and the λ parameter of the LPS and `emfrail` models are presented in Table 7. Both methods result in very similar regression coefficients, with the estimates from the LPS model about 10% larger (in absolute terms) compared to those from `emfrail`. Since this is also the case for the standard errors, it leads to almost identical z-statistics, and identical inferences: both algorithms indicate that treatment with Interferon Gamma leads to substantial decrease of the hazard of serious infections whereas sex does not play a role. Quantitative interpretation is also straightforward: under the LPS model, assuming it is correctly specified and holding sex covariate constant, patients on Interferon Gamma have, on average, $1 - \exp(-1.163) = 68.7\%$ lower hazard of a serious infection compared to those on placebo, which is equivalent to say that IG-treatment reduces the hazard of serious infection due to CGD three times. Since the 95% Posterior Credible Interval does not contain 0, we may further conclude that this hazard reduction is substantial.

Interestingly, the estimates of the γ parameter in `emfrail` and LPS differ by about factor two, which was not the case in the simulated datasets. To get insight into potential reasons for this

discrepancy, additional simulation was performed with 100 unbalanced datasets, each of 200 clusters of size 1 and 100 clusters of size 2 and $CR = 30\%$ to mimic the unbalanced character of the CGD dataset. Results of this simulation (presented in **Appendix 2**) show that in unbalanced datasets the correlation between γ parameters estimated by `emfrail` and LPS - although linear and very similar on average - may notably differ in individual datasets.

Regarding estimation of the amplitudes, Figure 8 presents Kaplan-Meier empirical estimates, together with the corresponding 95% confidence intervals of the two treatment arms, which essentially reproduce the estimates reported in the original paper (*cf.* Fig. 1 in [24]), as well as the survival curves calculated from the amplitudes obtained in the LPS model (left panel). For both arms, the estimated LPS curves lie within the 95% confidence intervals of the empirical estimators.

In the MALA sampler, the overall acceptance ratio was 0.574 (target, 0.57). It took much longer - compared to the simulated dataset in the previous section - for the chain to approach stationarity, which was however eventually achieved for all parameters (see Geweke's p-values). Traceplots of the model parameters (Figure 9) are similar to those of the previous section: the two of the regression parameters (panels A and B) do not show any particular pattern and no particular outlying observations. Traceplots of the γ, λ, and δ parameters (panels C, D, and E, respectively) show, again, some patterns of autocorrelation as well as outlying peaks. Posterior distributions of the regression parameters (panels A and B on Figure 10) are expectedly bell-shaped and close to symmetry. Posterior distributions of the γ, λ, and δ parameters (panels C, D, and E) are skewed, as they are expected to be, with the latter parameter following virtually Exponential distribution.

Table 8 reports summary statistics of the posterior parameters of the MALA sampler. The point estimate and the standard error of the effect of treatment are slightly larger (in absolute terms) compared to those of the LPS and `emfrail` models, which, however, leads to exactly the same inference: assuming the model is correctly specified, children on the IG-treatment have substantially lower hazard of a serious infection, whereas their sex does not play a hazard-altering role. It is also worth noting that the estimate of the γ parameter in the MALA sampler is somehow closer to that of the LPS model than to that of the `emfrail` model.

Table 7: LPS and emfrail models in the CGD study with all 128 subjects (203 observations); IG, Interferon gamma; CI, Credible Interval (in the LPS section) or Confidence Interval (in the emfrail section).

Parameter	Estimate	SE	z	95% CI
			LPS	
IG Treatment	−1.163	0.359	−3.240	[−1.867; −0.460]
Female sex	−0.250	0.455	−0.550	[−1.142; 0.642]
γ	0.580			
λ	1758			
			emfrail	
IG Treatment	−1.052	0.310	−3.389	[−1.660; −0.444]
Female sex	−0.227	0.396	−0.575	[−1.003; 0.548]
γ	1.218			

Table 8: Posterior summary statistics of the MALA sampler in the CGD study (iterations 10,001:25,000). HPDI, Highest Posterior Density Interval.

Parameter	Estimate	SE	95% HPDI	Geweke's p
IG Treatment	-1.190	0.365	$[-1.879; -0.484]$	0.96
Female sex	-0.249	0.463	$[-1.185; 0.623]$	0.93
γ	0.744	0.528	[0.222; 1.507]	0.92
λ	1822	1602	[87.8; 4941]	0.38
δ	0.001	0.002	[0.000; 0.005]	0.97

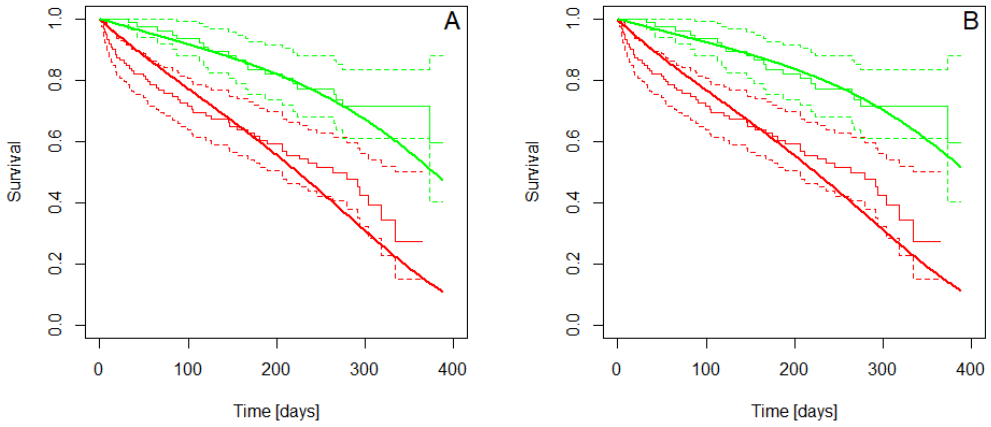


Figure 8: Empirical Kaplan-Meier estimates (solid polygonal chains) with the corresponding 95% confidence intervals (dashed) and the survival curves (solid curves) estimated with the amplitudes from: LPS (A), and MALA (B). Green color indicates the treatment arm and red color indicates the placebo arm.

As for estimation of the amplitudes, right panel of Figure 8 presents Kaplan-Meier empirical estimates, together with the corresponding 95% confidence intervals of the two treatment arms, and the survival curves obtained from the amplitudes estimated in the MALA sampler. Note that the model-based survival curves in the LPS (left panel) and in the MALA (right panel) are virtually identical.

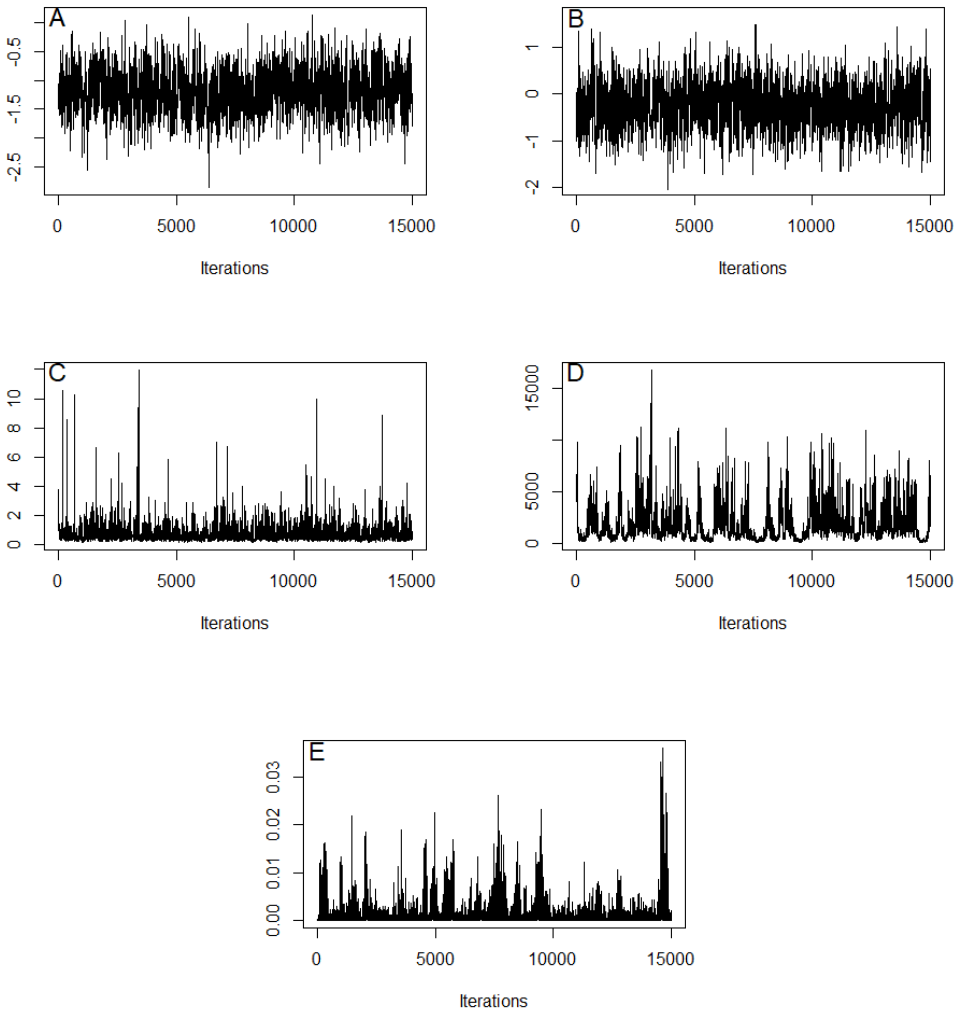


Figure 9: Traceplots of the parameters in the CGD study: IG treatment effect (A), female sex (B), γ (C), λ (D), and δ (E).

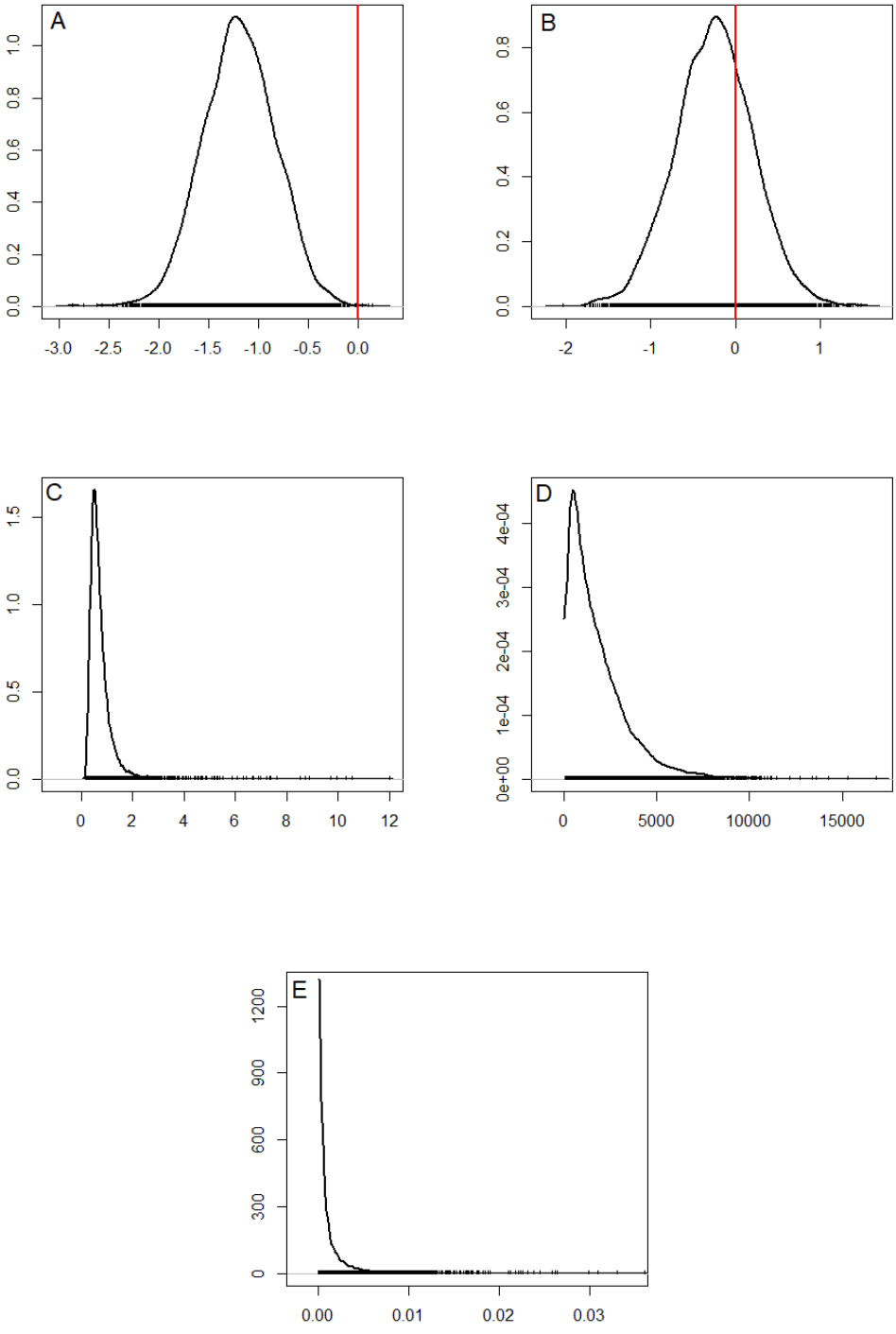


Figure 10: Posterior distributions of the parameters in the CGD study: IG treatment effect (A), female sex (B), γ (C), λ (D), and δ (E). Red bars on panels A and B indicate the *Null* hypotheses, $\beta_{(\cdot)} = 0$.

4 Discussion

In this study, we derived analytically a novel Bayesian model combining P-splines smoothers, Laplacian approximation, and gamma shared frailty survival. This resulted in two algorithms: a sampling-free Laplacian-P-Splines, and a Markov chain Monte Carlo Metropolis-Adjusted Langevin Algorithm within Gibbs Sampler. Both algorithms were implemented in `R`, tested on simulated datasets against an existing frequentist "competitor", and applied on a real-life randomized clinical trial.

The main motivation to extend the existing LPS toolbox [19] to shared frailty survival came from the study on the IG treatment in CGD in children, originally published in 1991 [24]. For proper treatment of the data as in this study, it is crucial to consider that the observed outcomes of interest (times-to-event) are not independent. Indeed, it is plausible to assume that times elapsing between incidences of a serious infection in a given patient are correlated, since some unobserved (latent) patient-specific characteristics may exist governing, for example, constitutive immunity strength of the patient, environmental factors, or perhaps quality of healthcare services given patient receives. It follows that those characteristics vary across patients, which leads to a hierarchical structure of the data with observations clustered within patients. In terms of survival analysis, the observations within a cluster (patient) share the same frailty.

Such datasets can be also analyzed with models based on event history and counting processes, with theoretical derivations based on stochasitic process known as *martingale* (see p. 431 in [12]). Other models for recurrent events include Anderson and Gill model [4] and Prentice, Williams and Peterson model [35], the former used in the original paper, under the assumption that the baseline hazard function is common to all recurrences and unaffected by the previous events. In line with Abrams *et al.* [3], we believe that this assumption might be questionable, as it might be postulated that recurrently appearing life-threatening infections may steadily worsen overall condition of a patient, leading to increased risk of infections in the future or, on contrary, infection may induce long-term immunity protecting a patient from another infection with the same agent. Therefore, association between recurrence times can be accounted for by adding a random frailty effect (see p. 436 in [12]). Abrams *et al.* postulate further extension to models taking into account the fact that individual frailties and correlations between them evolve in time [3].

In the process of model development, several decisions had to be made. First, we assumed a Gamma distribution for the frailty variable (with mean 1 and variance $1/\gamma$):

$$\begin{aligned} h_{ij}(t) &= u_i \exp(\boldsymbol{\beta}^T \boldsymbol{z}_{ij}) h_0(t) \\ U_i &\sim \mathcal{G}(\gamma, \gamma). \end{aligned} \tag{52}$$

This was driven by the fact that analytical integration of the conditional likelihood in Eq. (6) is only possible when the frailty is Gamma-distributed, although Hougaard noticed that Gamma-distributed frailty characterizes with a restriction that the dependence is most important for late events [25]. Another option, though without the convenient property of a closed-form integration, is a Normal distribution with mean 0 and variance σ_u^2:

$$\begin{aligned} h_{ij}(t) &= \exp(\boldsymbol{\beta}^T \boldsymbol{z}_{ij} + u_i) h_0(t) \\ U_i &\sim \mathcal{N}(0, \sigma_u^2). \end{aligned} \tag{53}$$

Further possible distributions are also described in the literature (for a recent review, see f.e. [7]).

Next, a decision needed to be taken, how to arrange the parameters of the model into a latent and a hyperparameter vectors. This is perhaps the crucial point in the whole model development, as it influences the strategy of the gradient and the Hessian derivations and, following from that, the code implementation. In this project, we placed the B-splines coefficients (amplitudes, $\boldsymbol{\theta}$) and the regression coefficients ($\boldsymbol{\beta}$) into the latent vector, and the remaining three parameters, the parameter

governing the frailty (γ), the parameter controlling the penalty (λ), and the hyperparameter for the distribution of the latter (δ) into the hyperparameter vector. This led to simpler gradient and the Hessian of the posterior distribution of the latent parameters since, assuming the Newton-Raphson Algorithm is used for the optimization of the latent vector, we needed the first and the second derivatives of the marginal likelihood in Eq. (13) only wrt $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$. For γ, only the first-order derivative was needed for the MALA part. As a consequence, however, our LPS algorithm provided only the point estimate but not the credible interval for γ (*cf.* *NA* entries in Table 2-4), which was also one of the arguments to develop MALA. It needs to be emphasized, however, that lacking precision estimates are not intrinsic limitation of the LPS itself but the consequence of the arrangement of the parameters into the latent and the hyperparameter vectors in this study. Additionally, decision had to be made regarding the variance of γ, needed in the covariance matrix of the proposal in Eq. (44). Following Gressani *et al.*, it was taken to be one, which may seem an arbitrary choice, which was, however, validated in multiple datasets and scenarios [23]. Further, at the stage of the optimization of the posterior hyperparameter, even after integrating δ out, the problem remains two-dimensional. Alternatively, the γ parameter could be included into the latent vector. This would require the second-order derivative of the marginal likelihood also wrt γ, assuming NRA is used, but it would provide the intervals for the γ parameter in the LPS algorithm, it would simplify the covariance matrix for the MALA part, and – perhaps most importantly – it would reduce optimization of the hyperparameter to a one-dimensional problem.

This is directly linked to the choice of the optimization algorithms. Having derived the gradient and the Hessian of the posterior latent vector in a closed form, the Newton-Raphson Algorithm was an obvious choice for its optimization. Another possibility would be, for example, the Levenberg-Marquardt Algorithm (LMA), which does not require the second-order derivatives and is more robust for the choice of the initial values [34]. Indeed, in our case finding appropriate starting vector took a while and even moderate alterations led to non-convergence. Robustness of the LMA comes, however, at the costs of slower code execution. Taken together, we believe that the setting where the γ parameter is treated as a hyperparameter, and for the latent vector the second-order derivatives are available in closed form (as in our study), the NRA is more appropriate, whereas including γ into the latent vector calls for the LMA. Both algorithms have an obvious performance advantage over the the Gradient Ascent Algorithm, which was not considered in this study.

Optimization of the hyperparameter, which was a two-dimensional problem in our case, was performed with the Nelder-Mead Algorithm (NMA) [33]. Briefly, the method relies on a *simplex*, which is a polytope with $n + 1$ vertices (in our case a triangle, since we optimize two parameters) in n dimensions. The simplex takes a series of steps moving the point where the function is lowest (for a maximization problem) through its opposite face in such a way that the volume of the simplex is preserved. This "movement" may resemble amoebic slither, which explains why the algorithm is also called "the amoeba method". The method uses only function values but not any derivative information, and hence is considered robust but slow (p. 164 in [9]). Another possibility is a quasi-Newton Broyden, Fletcher, Goldfarb and Shanno (BFGS) Algorithm (pp. 136-143 in [34]), which uses the derivative information, though does not perform matrix inverse, and hence is recommended "whenever one can reasonably assume that the objective function is smooth or at least differentiable" (Borchers in [9], p. 163). Both methods are available in the `optim()` function, which was used for the hyperparameter optimization, and indeed, we observed somehow faster performance of the BFGS algorithm compared to the NMA.

For the implementation stage, substantial effort was devoted to obtain a code that would be fast and efficient (after having achieved that it was beautiful, following the code optimization principle outlined by Armstrong [5]). The most involved part of the procedure are the calculations of the gradient (used in the LPS algorithm and in the MALA), and the Hessian (in the LPS). This is due

to the fact that they have to be repeated across all clusters in a dataset and in each iteration of the optimization algorithm, and hence may lead to an inefficient nested-loop process. Since it is known that `R` processes matrices and array objects [like `list()`] more efficiently than iterative `for` loops [31], wherever possible scalars were combined into vectors and vectors into matrices. This enabled replacement of the iterative calculations of the gradient and the Hessian across clusters by functions applied on array items, as outlined in **Appendix 1**. Efficacy of the object-oriented programming paradigm is obvious from the data in Table 6. For a given total size of a dataset (in our case, 400 observations), the execution is much faster when the dataset consists of small number of large clusters, compared to a dataset (of the same total size) with a large number of small clusters. Apparently, this is related to the abilities of the the current processors to efficiently perform matrix manipulations.

The main reason for the development of the LPS methods, and for their popularity, is because they are much faster compared to the traditional Bayesian strategy, relying on Markov chain samplers [19]. This is also the case in the current study, with the LPS algorithm delivering the outcome in seconds, rather, than minutes. Of course, the execution time of a sampler depends on its size, however, a running time of the MALA sampler in this project, when reduced to the time the LPS needs to complete (say, 5 seconds), would be sufficient only for about 500 iterations, which is obviously not enough. It needs to be stressed that the implementation of the LPS algorithm in this master's thesis is still not optimal, as the algorithm needs more time compared to the frequentist competitor. We believe that further optimization of the running time is possible. On the other hand, there seems to exist such real-life datasets, which the current – suboptimal – implementation of the LPS algorithm is able to analyze but which, for some reasons, cause problems for the frequentist method (*cf.* **Appendix 2**).

Simulation study is a very important part of a model development, as it allows critical testing whether algebraic derivations and their code implementation are correct (or at least plausible) under well controlled and modifiable conditions. In a way, this might be compared to a stage of laboratory tests in a biomedical project. In this study, the datasets were simulated using a novel, very flexible routine developed by Dr. Gressani [18], and tested simultaneously with a well-established frequentist function, based on the Expectation-Maximization Algorithm, `emfrail` [6]. In balanced datasets, the B-splines coefficients and the regression coefficients obtained with the algorithms developed here were in an excellent agreement with the results of the `emfrail` as well as with the results expected from the parameters of the simulations. Hence we may conclude that the LPS estimators are reasonably precise and characterize with negligible bias. Interesting case were the parameters of the frailty, γ, which differed from their "true values" expected from the parametrization of the datasets. This bias seems to be more pronounced in the setting with a larger censoring rate, which is explainable by the fact that censored observations contribute less information to the posterior distribution compared to the non-censored observations. Since this is a general phenomenon, affecting all estimation procedures, the correlation of the results of the LPS algorithm and the `emfrail`, again, practically follows an identity line. This also excludes that the observed bias is LPS-specific.

A different situation arises in a setting with unbalanced clusters, *i.e.* when a simulated dataset consists of many very small clusters (f.e. with 2 observations) accompanied by many single-observation clusters. Such datasets were generated and tested to understand better the results of the CGD study, which is of a similar nature. As presented in **Appendix 2**, in those datasets the correlation between LPS and `emfrail` estimates of the γ parameter is still linear and the estimates from both models are - on average - almost the same, but in individual datasets they may notably differ. We believe that this is explainable by the fact that, apart of 30% censoring, clusters with a single observation do not contribute posterior information on the variability of the frailty. In particular, they cannot provide information about within-cluster correlation. It also needs to be emphasized that the point

estimates of γ from all procedures are within the confidence interval of the `emfrail` as well as the credible interval of the MALA.

Intuition that the censoring rate influences estimated variances of the model parameters can be discussed in a more formal way. First, consider that the number of non-censored observations in an i-th cluster enters the estimation process via $\widetilde{\alpha}_i$ defined in Eq. (8). Next, observe that the elements of the i-th cluster's Hessian matrix in Eqs. (16) - (18) are directly proportional to the opposite of this quantity, and hence – holding all other variables constant – the larger the number of non-censored observations the larger $\widetilde{\alpha}_i$ and larger (in absolute terms) the elements on the diagonals of the Hessian matrices. This implies lower variances of the model parameters and explains lower standard errors of the regression coefficients in the setting (A), compared to (C), in the simulation experiment.

Initial values for the latent vector were moderately hard to find, and slight fluctuations of the initial latent vector often led to non-convergence of the underlying NRA. This is not surprising, and might be seen as an argument in favour of a different optimization algorithm, as already discussed. On the other hand, alterations of starting values for the hyperparameters did not influence optimization of the hyperparameter vector, which is also in agreement with the theory, as the NMA, used in this step, is considered robust [9].

Metropolis-Adjusted Langevin Algorithm was derived in this study as a method alternative to the LPS algorithm, and partially utilizing its results. The main reason to setting it, apart of confirming plausibility of the LPS algorithm, was that in our design the parameter controlling the frailty variability, γ, is allocated to the hyperparameter vector, and hence the LPS algorithm can provide its point estimate but not its variance. In contrast, MCMC returns approximate posterior distributions of all parameters of the model, from hence it is possible to calculate their point estimates as well as their variances. In contrast to the original algorithm developed by Metropolis *et al.* [32], its Langevin-adjusted version makes use of the information from the gradient of the (log-)posterior distribution [8]. Briefly, a new state of the MCMC is first proposed using *Langevin dynamics*, which evaluates the gradient of the target posterior at the current state of the chain, followed by acceptance/rejection decision with the Metropolis–Hastings algorithm (MHA). This drives the chain's random walk towards regions of higher posterior probability. As a matter of fact, in our algorithm the gradient of the target posterior distribution was used twice: first to adjust expectation of the proposal distribution (Eq. (43)) and next to optimize the acceptance probability by comparing the gradient of the posterior at the current state of the chain and at the proposed candidate (Eq. (47)). Further, in contrast to the "classic" MHA, MALA applied in this study makes use of the covariance matrix based on Laplace approximation, $\boldsymbol{\Sigma}^*(\widetilde{\boldsymbol{\eta}}^*)$ (*cf.* Eq. (44)), which sets a desirable correlation structure of the latent variables [20]. To further improve performance of the algorithm, a tuning parameter, ϱ, is adaptively calculated in each sampling step to yield the optimal acceptance rate of 0.57 [20, 23].

In the CGD study, we restricted our LPS and MALA modelling to two explanatory variables, the treatment and sex, to keep it close to the results reported with the `frailtyEM` package [6]. Needles to say, other predictors (age, height, weight, genotype, *etc.*) can be included into the model, too. The point estimates and the credible intervals for the treatment effect are in very good agreement across all algorithms used in the current study. All of them lead to exactly the same interpretation of IG reducing the hazard of serious infection by about 70%, as it was also reported in the original paper [24]. Indeed, in 1991 the U.S. Food and Drug Administration approved IG for treatment in CGD, having saved many lives since then. Insubstantial effect of sex, also consistently returned by all models, is also important result, as it allows the same clinical treatment of females and males.

This study is not without limitations. We believe that by looking at the data from the shared frailty perspective, we have accounted for the correlations across the multivariate outcomes for a given patient, however, our model assumes that those correlations are time-invariant. This is a seriously

limiting assumption; indeed, Abrams *et al.* postulate, in a frequentist framework and for a bivariate time-to-event setting, a model with the subject- and event-specific hazard function, which leads to a frailty variable U_{iq} (in our notation) for i-th cluster and q-th event [3]. From the perspective of their study, our model can be seen as a special case, with $U_{i1} = ... = U_{iQ} \equiv U_i$. To release this restriction, Abrams *et al.* propose *correlated frailty model* in a sense that correlations between times are allowed to differ from 1. Certainly, extension of our model into that direction could be an interesting further development.

References

[1] COVID-19 Data. https://github.com/owid/covid-19-data/tree/master/public/data. Accessed: 03.12.2021.

[2] Flat spline. https://en.wikipedia.org/wiki/Flat_spline. Accessed: 02.02.2024.

[3] S. Abrams, A. Wienke, and N. Hens. Modelling Time Varying Heterogeneity in Recurrent Infection Processes: An Application to Serological Data. *J R Stat Soc (C)*, 67(3):687–704, 2017.

[4] P.K. Andersen and R.D. Gill. Cox's Regression Model for Counting Processes: A Large Sample Study. *Ann Stat*, 10(4):1100–1120, 1982.

[5] J. Armstrong. https://blog.ndpar.com/2010/11/18/joe-armstrong-on-optimization/. Accessed: 21.03.2024.

[6] T.A. Balan and H. Putter. frailtyEM: An R package for estimating semiparametric shared frailty models. *J Stat Softw*, 90(7):1–29, 2019.

[7] T.A. Balan and H. Putter. A tutorial on frailty models. *Stat Methods Med Res*, 29(11):3424–3454, 2020.

[8] J. Besag. Comments on "Representations of knowledge in complex systems" by U. Grenander and MI Miller. *J R Stat Soc (B)*, 56:591–592, 1994.

[9] V.A. Bloomfield. *Using R for Numerical Analysis in Science and Engineering*. Taylor & Francis, 2014.

[10] A. Brezger and W.J. Steiner. Monotonic regression based on Bayesian P–splines: An application to estimating price response functions from store-level scanner data. *J Bus Econ Stat*, 26(1):90–104, 2008.

[11] R.A. Bridges, H. Berendes, and R.A. Good. A Fatal Granulomatous Disease of Childhood: The Clinical, Pathological, and Laboratory Features of a New Syndrome. *AMA J Dis Child*, 97(4):387–408, 1959.

[12] D. Collett. *Modelling Survival Data in Medical Research (3rd ed.)*. Chapman and Hall/CRC, New York, 2015.

[13] D.R. Cox. Regression Models and Life-Tables. *J R Stat Soc Series B Stat Methodol*, 34(2):187–202, 1972.

[14] P. Eilers and B. Marx. Why P-splines? https://psplines.bitbucket.io/Support/WhyPsplines.pdf. Accessed: 01.02.2024.

[15] P.H.C. Eilers and B.D. Marx. Flexible smoothing with B-splines and penalties. *Stat Sci*, 11(2):89–121, 1996.

[16] P.H.C. Eilers and B.D. Marx. *Practical Smoothing: The Joys of P-splines*. Cambridge University Press, Cambridge, 2021.

[17] T.R. Fleming and D.P. Harrington. *Counting Processes and Survival Analysis*. John Wiley and Sons Inc., New York, 2005.

[18] O. Gressani. https://github.com/oswaldogressani/Frailty. Accessed: 12.03.2024.

[19] O. Gressani. *Laplace Approximations and Bayesian P-splines for Statistical Inference*. PhD thesis, University of Louvain, 2020. Accessible at: https://greoswa.com/Oswaldo_PhD_eThesis_2020.pdf.

[20] O. Gressani, C. Faes, and N. Hens. Laplacian-P-splines for Bayesian inference in the mixture cure model. *Stat Med*, 41(14):2602–2626, 2022.

[21] O. Gressani and P. Lambert. Fast Bayesian inference using Laplace approximations in a flexible promotion time cure model based on P-splines. *Comput Stat Data Anal*, 124:151–167, 2018.

[22] O. Gressani and P. Lambert. Laplace approximations for fast Bayesian inference in generalized additive models based on P-splines. *Comput Stat Data Anal*, 154:107088, 2021.

[23] O. Gressani, J. Wallinga, C. L. Althaus, N. Hens, and C. Faes. Epilps: A fast and flexible Bayesian tool for estimation of the time-varying reproduction number. *PLoS Comput Biol*, 18(10):e1010618, 2022.

[24] International Chronic Granulomatous Disease Cooperative Study Group. A controlled trial of interferon gamma to prevent infection in chronic granulomatous disease. *N Engl J Med*, 324(8):509–516, 1991.

[25] P. Hougaard. Frailty models for survival data. *Lifetime Data Anal*, 1:255–273, 1995.

[26] A. Jullion and P. Lambert. Robust specification of the roughness penalty prior distribution in spatially adaptive Bayesian P-splines models. *Comput Stat Data Anal*, 51(5):2542–2558, 2007.

[27] A.A. Justiz-Vaillant, A.F. Williams-Persad, R. Arozarena-Fundora, D. Gopaul, S. Soodeen, O. Asin-Milan, R. Thompson, C. Unakal, and Akpaka P.E. Chronic Granulomatous Disease (CGD): Commonly associated pathogens, diagnosis and treatment. *Microorganisms*, 11(9):2233, 2023.

[28] P. Lambert and O. Gressani. Penalty parameter selection and asymmetry corrections to Laplace approximations in Bayesian P-splines models. *Stat Modelling*, 23(5-6):409–423, 2023.

[29] S. Lang and A. Brezger. Bayesian P-Splines. *J Comput Graph Stat*, 13(1):183–212, 2004.

[30] P.S. Laplace. Memoir on the Probability of the Causes of Events. *Stat Sci*, 1(3):364–378, 1986.

[31] N.S. Matloff. *Art of R programming*. No Starch Press, San Francisco, 2011.

[32] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J Chem Phys*, 21(6):1087–1092, 1953.

[33] J.A. Nelder and R. Mead. A Simplex Method for Function Minimization. *Comput J*, 7(4):308–313, 1965.

[34] J. Nocedal and S. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, New York, 2006.

[35] R.L. Prentice, B.J. Williams, and A.V. Peterson. On the regression analysis of multivariate failure time data. *Biometrika*, 68(2):373–379, 08 1981.

[36] G.O. Roberts and J.S. Rosenthal. Optimal scaling of discrete approximations to Langevin diffusions. *J R Stat Soc (B)*, 60(1):255–268, 1998.

[37] I.J. Schoenberg. Contributions to the problem of approximation of equidistant data by analytic functions. *Quart Appl Math*, 4:45–99 and 112–141, 1946.

[38] L. Schumaker. *Spline Functions: Basic Theory*. Cambridge University Press, Cambridge, 2007.

[39] L. Tierney. Exploring posterior distributions using Markov chains. In *Computing science and statistics: Proceedings of the 23rd symposium on the interface*, pages 563–570, 1991.

Appendix 1

From the log-likelihood (Eq. (13)), elements of the gradient wrt θ_k (Eq. (14)) can be derived as follows:

$$\begin{aligned}
\frac{\partial \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \theta_k} &= \frac{\partial}{\partial \theta_k}\Big(\sum_{j=1}^{n_i} \widetilde{\delta}_{ij}\Big(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij}\Big) - \widetilde{\alpha}_i \ln(\widetilde{\beta}_i)\Big) \\
&= \frac{\partial}{\partial \theta_k}\Big(\sum_{j=1}^{n_i} \widetilde{\delta}_{ij}\Big(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij}\Big)\Big) - \frac{\partial}{\partial \theta_k}\Big(\widetilde{\alpha}_i \ln(\widetilde{\beta}_i)\Big) \\
&= \sum_{j=1}^{n_i} \frac{\partial}{\partial \theta_k} \widetilde{\delta}_{ij}\Big(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij}\Big) - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \frac{\partial}{\partial \theta_k} \widetilde{\beta}_i \\
&= \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} b_k(t_{ij}) - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \frac{\partial}{\partial \theta_k} \underbrace{\Big(\gamma + \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm}\Big)}_{\widetilde{\beta}_i} \\
&= \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} b_k(t_{ij}) - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \Big[\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \frac{\partial}{\partial \theta_k} \underbrace{\Big(\exp(\boldsymbol{\theta}^T \boldsymbol{b}(s_m) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij})\Delta\Big)}_{\omega_{ijm}}\Big] \\
&= \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} b_k(t_{ij}) - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} b_k(s_m) \omega_{ijm}.
\end{aligned}$$

In a very similar way, we derive elements of the gradient wrt β_p (Eq. (15)):

$$\begin{aligned}
\frac{\partial \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \beta_p} &= \frac{\partial}{\partial \beta_p}\Big(\sum_{j=1}^{n_i} \widetilde{\delta}_{ij}\Big(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij}\Big) - \widetilde{\alpha}_i \ln(\widetilde{\beta}_i)\Big) \\
&= \frac{\partial}{\partial \beta_p}\Big(\sum_{j=1}^{n_i} \widetilde{\delta}_{ij}\Big(\boldsymbol{\theta}^T \boldsymbol{b}(t_{ij}) + \boldsymbol{\beta}^T \boldsymbol{z}_{ij}\Big)\Big) - \frac{\partial}{\partial \beta_p}\Big(\widetilde{\alpha}_i \ln(\widetilde{\beta}_i)\Big) \\
&= \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} z_{ijp} - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \frac{\partial}{\partial \beta_p}\Big(\gamma + \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm}\Big) \\
&= \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} z_{ijp} - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \frac{\partial}{\partial \beta_p} \omega_{ijm}\Big) \\
&= \sum_{j=1}^{n_i} \widetilde{\delta}_{ij} z_{ijp} - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} z_{ijp} \omega_{ijm}.
\end{aligned}$$

Elements of the Hessian matrix wrt the vector of the amplitudes (Eq. (16)) can be derived as follows:

$$
\begin{aligned}
\frac{\partial^2 \ell_i(\boldsymbol{\xi},\gamma;D)}{\partial\theta_k\partial\theta_{k'}} &= \frac{\partial}{\partial\theta_{k'}} \underbrace{\Big(\sum_{j=1}^{n_i}\widetilde{\delta}_{ij}b_k(t_{ij}) - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i}\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)}_{\frac{\partial \ell_i(\boldsymbol{\xi},\gamma,D)}{\partial\theta_k}} \\
&= -\frac{\partial}{\partial\theta_{k'}}\Big(\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i}\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big) \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\frac{\partial}{\partial\theta_{k'}}\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\frac{\partial}{\partial\theta_{k'}}\widetilde{\beta}_i\right\} \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\frac{\partial}{\partial\theta_{k'}}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\frac{\partial}{\partial\theta_{k'}}\widetilde{\beta}_i\right\} \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\frac{\partial}{\partial\theta_{k'}}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\frac{\partial}{\partial\theta_{k'}}\Big(\gamma + \sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})}\omega_{ijm}\Big)\right\} \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\frac{\partial}{\partial\theta_{k'}}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})}\frac{\partial}{\partial\theta_{k'}}\omega_{ijm}\Big)\right\} \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)b_{k'}(s_m)\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_{k'}(s_m)\omega_{ijm}\Big)\Big].
\end{aligned}
$$

Next, we derive elements of the Hessian wrt the vector of the regression coefficients (Eq. (17)):

$$
\begin{aligned}
\frac{\partial^2 \ell_i(\boldsymbol{\xi},\gamma;D)}{\partial\beta_p\partial\beta_{p'}} &= \frac{\partial}{\partial\beta_{p'}}\Big(\sum_{j=1}^{n_i}\widetilde{\delta}_{ij}z_{ijp} - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i}\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp}\omega_{ijm}\Big) \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp}\frac{\partial}{\partial\beta_{p'}}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp}\omega_{ijm}\Big)\frac{\partial}{\partial\beta_{p'}}\widetilde{\beta}_i\right\} \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp}\frac{\partial}{\partial\beta_{p'}}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp}\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})}\frac{\partial}{\partial\beta_{p'}}\omega_{ijm}\Big)\right\} \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp}z_{ijp'}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp}\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp'}\omega_{ijm}\Big)\Big].
\end{aligned}
$$

Finally, elements in "off-diagonal" block (Eq. (18)) can be obtained starting either from the first derivative wrt θ_k (Eq. (14)) or from the first derivative wrt β_p (Eq. (15)), with the first option followed here:

$$\begin{aligned}
\frac{\partial^2 \ell_i(\boldsymbol{\xi},\gamma;D)}{\partial\theta_k\partial\beta_p} &= \frac{\partial}{\partial\beta_p}\Big(\frac{\partial \ell_i(\boldsymbol{\xi};\gamma,D)}{\partial\theta_k}\Big) = \frac{\partial}{\partial\theta_k}\Big(\frac{\partial \ell_i(\boldsymbol{\xi};\gamma,D)}{\partial\beta_p}\Big) \\
&= \frac{\partial}{\partial\beta_p}\Big(\sum_{j=1}^{n_i}\widetilde{\delta}_{ij}b_k(t_{ij}) - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i}\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big) \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\frac{\partial}{\partial\beta_p}\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\frac{\partial}{\partial\beta_p}\widetilde{\beta}_i\right\} \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\frac{\partial}{\partial\beta_p}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})}\frac{\partial}{\partial\beta_p}\omega_{ijm}\Big)\right\} \\
&= -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)z_{ijp}\omega_{ijm}\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} b_k(s_m)\omega_{ijm}\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} z_{ijp}\omega_{ijm}\Big)\Big].
\end{aligned}$$

Partial derivative of the log-conditional posterior of the $\widetilde{\boldsymbol{\zeta}}$ vector wrt $\widetilde{\gamma}$ in the MALA step (Eq. (46)) is obtained as follows:

$$\frac{\partial \ln p(\widetilde{\boldsymbol{\zeta}} \mid \lambda, D)}{\partial\widetilde{\gamma}} = \frac{\partial}{\partial\widetilde{\gamma}}\Big(\underbrace{\ell(\widetilde{\boldsymbol{\zeta}};D)}_{cf.\ Eq.\ (40)} - \underbrace{\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{Q}\boldsymbol{\xi}}_{does\ not\ depend\ on\ \widetilde{\gamma}} - \beta_\gamma\exp(\widetilde{\gamma}) + \alpha_\gamma\widetilde{\gamma}\Big)$$

$$\begin{aligned}
&= \mathcal{I}\exp(\widetilde{\gamma})(\widetilde{\gamma}+1) - \mathcal{I}\exp(\widetilde{\gamma})F(\exp(\widetilde{\gamma})) + \sum_{i=1}^{\mathcal{I}}\exp(\widetilde{\gamma})F(\breve{\alpha}_i) \\
&- \exp(\widetilde{\gamma})\sum_{i=1}^{\mathcal{I}}\Big(\ln(\breve{\beta}_i) + \frac{\breve{\alpha}_i}{\breve{\beta}_i}\Big) - \beta_\gamma\exp(\widetilde{\gamma}) + \alpha_\gamma \\
&= \exp(\widetilde{\gamma})\left(\mathcal{I}\Big(\widetilde{\gamma}+1-F(\exp(\widetilde{\gamma}))\Big) + \sum_{i=1}^{\mathcal{I}}\Big(F(\breve{\alpha}_i) - \ln(\breve{\beta}_i) - \frac{\breve{\alpha}_i}{\breve{\beta}_i}\Big) - \beta_\gamma\right) + \alpha_\gamma.
\end{aligned}$$

◇ ◇ ◇

Implementation of the calculations of the log-likelihood, the gradient and the Hessian is much more straightforward when the derivatives are expressed in vector/matrix form, rather, than element-wise (as in the Eqs. (13) - (18)). This approach immediately translates to a code combining operations on matrices and `list()` objects, which is known to be very efficient in `R`.

The quantities defined in Eq. (7) can be grouped into an i-th cluster specific matrix, $\boldsymbol{\Omega_i}$, and those in Eqs. (8) and (9) can be calculated in the matrix notation, respectively:

$$\boldsymbol{\Omega_i} = \mathbf{exp}[\boldsymbol{B}(s_m)\boldsymbol{\theta}\mathbf{1}_{n_i}^T + \mathbf{1}_J(\boldsymbol{Z_i}\boldsymbol{\beta})^T] \odot \boldsymbol{\Delta_i}$$

$$\widetilde{\alpha}_i = \widetilde{\boldsymbol{\delta}}_i^T \mathbf{1}_{n_i} + \gamma$$

$$\widetilde{\beta}_i = \underbrace{\mathbf{1}_J^T \boldsymbol{\Omega_i} \mathbf{1}_{n_i}}_{grandsum} + \gamma,$$

where $\odot$ and $\mathbf{exp}[\cdot]$ denote element-wise (Hadamard) product and exponentiation, respectively, and $\mathbf{1}_{(\cdot)}$ are unit-vectors of the dimensions denoted in the subscripts. $J \times n_i$ matrix $\boldsymbol{\Delta_i}$ is constructed such way that for a j-th observation in the i-th cluster, the first $m(t_{ij})$ elements of its j-th column vector are equal Δ and the remaining elements are zeroes. Thus, $\boldsymbol{\Delta_i}$ controls approximation of the cumulative hazard for a given subject (*cf.* Eq. (2)) by setting irrelevant time points equal to zero and, simultaneously, it incorporates multiplication by the (constant) length of the grid segment, Δ. Further, by $\boldsymbol{I}_{(\cdot)}$ we denote an identity matrix of the dimension given in the subscript.

First derivative wrt the vector of amplitudes, $\boldsymbol{\theta}$, can be now written in the matrix notation:

$$\frac{\partial \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \boldsymbol{\theta}} = \boldsymbol{B_i}^T(t_{ij})\widetilde{\boldsymbol{\delta}}_i - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm} \boldsymbol{b}(s_m)$$

$$= \boldsymbol{B_i}^T(t_{ij})\widetilde{\boldsymbol{\delta}}_i - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \boldsymbol{B}^T(s_m)\boldsymbol{\Omega_i}\mathbf{1}_{n_i}.$$

First derivative wrt the vector of regression coefficients, $\boldsymbol{\beta}$, becomes:

$$\frac{\partial \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \boldsymbol{\beta}} = \boldsymbol{Z_i}^T\widetilde{\boldsymbol{\delta}}_i - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} \sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm} \boldsymbol{z}_{ij}$$

$$= \boldsymbol{Z_i}^T\widetilde{\boldsymbol{\delta}}_i - \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i} (\boldsymbol{\Omega_i}\boldsymbol{Z_i})^T \mathbf{1}_J.$$

The sumbatrix of the Hessian wrt $\boldsymbol{\theta}$ (the top-left $K \times K$ block) becomes:

$$\frac{\partial^2 \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} = -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm} \boldsymbol{b}(s_m)\boldsymbol{b}^T(s_m)\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm} \boldsymbol{b}(s_m)\Big)\Big(\sum_{j=1}^{n_i} \sum_{m=1}^{m(t_{ij})} \omega_{ijm} \boldsymbol{b}^T(s_m)\Big)\Big]$$

$$= \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big\{\boldsymbol{B}^T(s_m)\boldsymbol{\Omega_i}\mathbf{1}_{n_i}\mathbf{1}_{n_i}^T\boldsymbol{\Omega_i}^T\boldsymbol{B}(s_m) - \widetilde{\beta}_i\boldsymbol{B}^T(s_m)[\boldsymbol{\Omega_i}\mathbf{1}_{n_i}\mathbf{1}_J] \odot \boldsymbol{I_J}\boldsymbol{B}(s_m)\Big\}$$

$$= \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\boldsymbol{B}^T(s_m)\Big\{\boldsymbol{\Omega_i}\mathbf{1}_{n_i}(\boldsymbol{\Omega_i}\mathbf{1}_{n_i})^T - \widetilde{\beta}_i[\boldsymbol{\Omega_i}\mathbf{1}_{n_i}\mathbf{1}_J^T] \odot \boldsymbol{I_J}\Big\}\boldsymbol{B}(s_m),$$

where $[\boldsymbol{\Omega_i}\mathbf{1}_{n_i}\mathbf{1}_J^T] \odot \boldsymbol{I_J}$ is a $J \times J$ diagonal matrix with the sum of the column vectors of the $\boldsymbol{\Omega_i}$ matrix on the diagonal. Indeed, denote by $\mathrm{diag}(\boldsymbol{\Omega_{i(j)}})$, $j = 1, ..., n_i$, a $J \times J$ diagonal matrix with a j-th column of the $\boldsymbol{\Omega_i}$ matrix on the diagonal; then $\sum_{j=1}^{n_i} \mathrm{diag}(\boldsymbol{\Omega_{i(j)}}) = [\boldsymbol{\Omega_i}\mathbf{1}_{n_i}\mathbf{1}_J^T] \odot \boldsymbol{I_J}$.

The sumbatrix of the Hessian wrt $\boldsymbol{\beta}$ (the bottom-right $P \times P$ block) becomes:

$$\frac{\partial^2 \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} = -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{z}_{ij}\boldsymbol{z}_{ij}^T\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{z}_{ij}\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{z}_{ij}^T\Big)\Big]$$

$$= \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\boldsymbol{Z_i}^T\Big\{\boldsymbol{\Omega_i}^T\mathbf{1}_J\mathbf{1}_J^T\boldsymbol{\Omega_i} - \widetilde{\beta}_i[\boldsymbol{\Omega_i}^T\mathbf{1}_J\mathbf{1}_{n_i}^T] \odot \boldsymbol{I_{n_i}}\Big\}\boldsymbol{Z_i},$$

where $[\boldsymbol{\Omega_i}^T\mathbf{1}_J\mathbf{1}_{n_i}^T] \odot \boldsymbol{I_{n_i}}$ is a $n_i \times n_i$ diagonal matrix with the sum of the column vectors of the $\boldsymbol{\Omega_i}^T$ matrix on the diagonal.

The top-right $K \times P$ block becomes:

$$\frac{\partial^2 \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \boldsymbol{\theta} \partial \boldsymbol{\beta}^T} = -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{b}(s_m)\boldsymbol{z}_{ij}^T\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{b}(s_m)\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{z}_{ij}^T\Big)\Big]$$

$$= \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\boldsymbol{B}^T(s_m)\Big\{\boldsymbol{\Omega_i}\mathbf{1}_{n_i}\mathbf{1}_J^T\boldsymbol{\Omega_i} - \widetilde{\beta}_i\boldsymbol{\Omega_i}\Big\}\boldsymbol{Z_i}.$$

The bottom-left $P \times K$ block (which is transpose of the top-right block) becomes:

$$\frac{\partial^2 \ell_i(\boldsymbol{\xi}, \gamma; D)}{\partial \boldsymbol{\beta} \partial \boldsymbol{\theta}^T} = -\frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\Big[\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{z}_{ij}\boldsymbol{b}^T(s_m)\Big)\widetilde{\beta}_i - \Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{z}_{ij}\Big)\Big(\sum_{j=1}^{n_i}\sum_{m=1}^{m(t_{ij})} \omega_{ijm}\boldsymbol{b}^T(s_m)\Big)\Big]$$

$$= \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\left\{\boldsymbol{B}^T(s_m)\Big\{\boldsymbol{\Omega_i}\mathbf{1}_{n_i}\mathbf{1}_J^T\boldsymbol{\Omega_i} - \widetilde{\beta}_i\boldsymbol{\Omega_i}\Big\}\boldsymbol{Z_i}\right\}^T$$

$$= \frac{\widetilde{\alpha}_i}{\widetilde{\beta}_i^2}\boldsymbol{Z_i}^T\Big\{\boldsymbol{\Omega_i}^T\mathbf{1}_J\mathbf{1}_{n_i}^T\boldsymbol{\Omega_i}^T - \widetilde{\beta}_i\boldsymbol{\Omega_i}^T\Big\}\boldsymbol{B}(s_m).$$

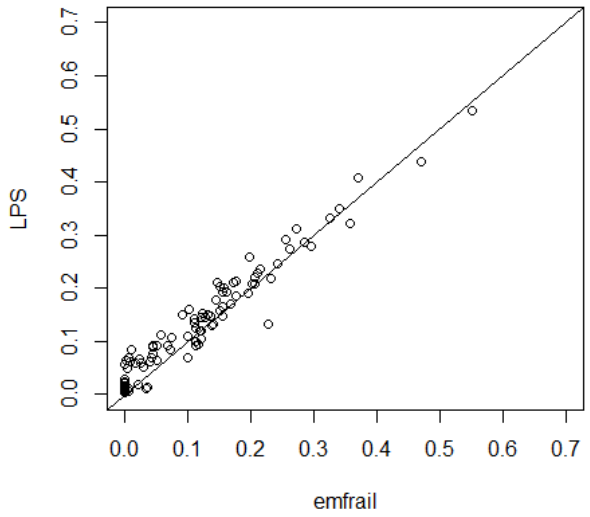
To approximate baseline hazard function, $h_0(t)$, in a *Null* model (a model without predictors), all elements of $\boldsymbol{Z_i}$ are simply set to 0 for all i.

The `R` code for the CGD part (LPS and MALA) is available at:

https://github.com/LewczukPiotr/LPS_and_MALA_in_CGD.

Appendix 2

This Appendix presents some auxiliary results for the discussion of the discrepant estimates, particularly of the γ parameter, in the LPS and the EMA models (the later implemented in the `emfrail` function and treated as a "competitor" in this thesis) in unbalanced datasets. In order to provide more insight, the simulation study was extended to 100 datasets more, each consisting of 200 clusters of size 1 and 100 clusters of size 2 with 30% right-censoring. This additional simulation is supposed to mimic unbalanced dataset of the CGD study. As Figure below indicates, in such unbalanced datasets the correlation between LPS and `emfrail` estimates of the γ parameter is linear and the estimates from both models are - on average - almost the same, however, in individual datasets they may notably differ. This does not seem to be the case in the balanced datasets, where the two algorithms always return very similar estimates (*cf.* left panels of Figure 4).



Appendix Figure 1: Comparison of the estimates of the frailty variance ($1/\gamma$) in one hundred additional simulations.

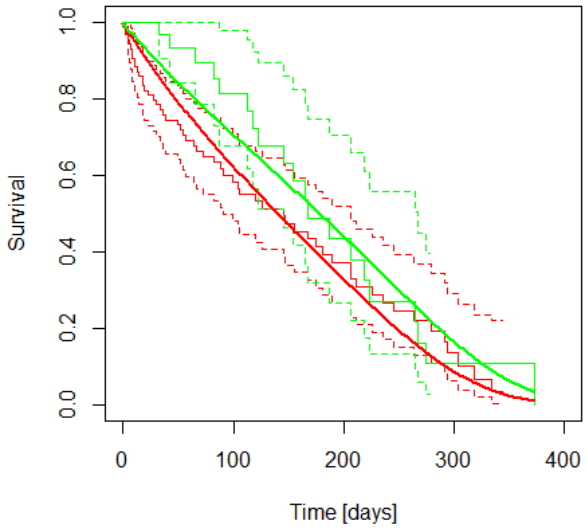
Next, three models were fitted with the data of the CGD study, including only on such patients that had at least two recorded observations (at least two incidences of a serious infection or one infection and one censored follow-up, $n = 44$), *i.e.* after excluding all clusters of size 1. Model M1 was fitted with the treatment indicator only, model M2 with the treatment indicator and sex, and model M3 with the treatment indicator, sex and age. Results of the LPS estimates of the three models are presented in Table below.

	M1			M2			M3		
Parameter	Estimate	SE	z	Estimate	SE	z	Estimate	SE	z
IG Treatment	−0.306	0.281	−1.092	−0.309	0.284	−1.158	−0.316	0.289	−1.095
Female sex				0.097	0.359	0.270	0.167	0.368	0.454
Age [yrs]							−0.022	0.015	−1.408
γ	10.87			9.232			7.757		
λ	1771			1717			1921		

Interestingly, the `emfrail` function failed to analyze this dataset (in all three models), reporting a message "frailty variance might be at the edge of the parameter space." This is certainly disappointing, as a comparison of the results in a real-life study would provide further important insights into the issue[3]. Without this additional information, we may at least conclude that the LPS estimates are plausible. In all three models, the effects of the treatment are negative (although lower, in absolute terms, compared to those in the modelling of the whole dataset). In the two models with sex (M2 and M3), this variable does not play a role, and in the model with age (M3), older children are less endangered with recurrence of serious infection, which is also known from the literature [24].

Finally, Figure below presents Kaplan-Meier empirical estimates, together with the corresponding 95% confidence intervals of the two treatment arms in the subgroup of patients with at least two observations, as well as the estimated survival curves obtained in the current LPS model. For both arms, the estimated LPS curves lie within the 95% confidence intervals of the Kaplan-Meier curves, which confirms plausibility of the LPS estimates of the amplitudes.

Taken together, we believe that clusters of size 1 do not contribute posterior information about the variation of the frailty, which leads to discrepant γ estimates between a frequentist and a Bayesian setting. It is hard to tell, which of the two settings provide a "correct" estimate.



Appendix Figure 2: Kaplan-Meier and LPS-estimated survival curves for the subset of the CGD patients with at least two observations.

[3]On the other hand, it might be concluded that the LPS algorithm is a "better" competitor, as it is able to analyze such dataset.

Acknowledgements

When I began studying biostatistics at the Hasselt University, I was already 50+. Now, when I am finalizing it, I am 50++ (or 60−, if you insist...) so, it was a tough job, and I succeeded only because many excellent people have been supporting and helping me through all those years. I am not able to mention all of them here and I apologize for it. Let me, however, send my cordial thanks to at least some of them.

My first and deepest expressions of gratitude go to my outstanding supervisors, Prof. Dr. Christel Faes and Dr. Oswaldo Gressani, for their inspirations, patience, and all explanations that made my walk through the abstract (for me) fields of Bayesian statistics less random.

I would like to extend my acknowledgements to the Authorities of the Hasselt University, for enabling me the studies. My most sincere thanks go to all teachers and instructors of the program of biostatistics; in particular, among many others, I would like to mention Prof. Dr. Geert Molenberghs, Prof. Dr. Geert Verbeke, Prof. Dr. Tomasz Burzykowski, Prof. Dr. Olivier Thas, and Prof. Dr. Anneleen Verhasselt. Furthermore, I am grateful to our very friendly and competent academic advisor, Mr. Michiel Vandenbempt, and the team coordinating the master's thesis course, with Dr. Sarah Vercruysse, Dr. Liesbeth Bruckers, and Mrs. Martine Machiels.

My boss, chairman of the Department of Psychiatry and Psychotherapy, Universitätsklinikum Erlangen, Prof. Dr. Johannes Kornhuber, deserves my most cordial gratitude for his continuous encouraging me and for his tolerance when I was too busy with my studies to do all my tasks on time. My wonderful team at the Laboratory for Clinical Neurochemistry and Neurochemical Dementia Diagnostics, with Christine Schödel, Ute Schulz, Angela Noureddine, Sabine Müller, Johanna Waedt, and Laura Haßler, was incredibly patient when their boss was running through the lab trying to solve a particularly interesting integral, instead of doing something useful. I am very thankful for that, you probably deserve a better boss.

The two most important persons in my life, my wife Dr. Barbara Sawicka-Lewczuk and our teenager son Paweł deserve special treatment for everything good that I have ever experienced. This goes much beyond my studies of biostatistics. This goes beyond everything I could express in words. I thank you for being with me and I thank God that I have you.

My parents, Dr. Ludmiła Gruszewska-Lewczuk (1937–2023) and Włodzimierz Lewczuk (1931–2008) are not here anymore in a physical sense. They have already gone in their longest journey, but I still feel a gentle touch of their support. I believe they are smiling at me from wherever they are, hopefully with proud and surely with love.