

Supplement to
“Bayesian Accelerated Failure Time Model
with Multivariate Doubly-Interval-Censored Data
and Flexible Distributional Assumptions”

ARNOŠT KOMÁREK and EMMANUEL LESAFFRE

Biostatistical Centre, Katholieke Universiteit Leuven,

Kapucijnenvoer 35, B-3000, Leuven, Belgium

E-mail: Arnost.Komarek@med.kuleuven.be

Emmanuel.Lesaffre@med.kuleuven.be

1 Introduction

This technical report acts as a supplement to the paper “Bayesian Accelerated Failure Time Model with Multivariate Doubly-Interval-Censored Data and Flexible Distributional Assumptions” (Komárek and Lesaffre, *Journal of the American Statistical Association*, accepted in 2007). The report presents mainly

- Directed acyclic graphs for considered models.
- Details on the implementation of the Markov chain Monte Carlo.
- More detailed results of the simulation study.
- Description on how to fit the Signal Tandmobiel[®] example using the R package `bayesSurv`.

To ease orientation, sections in the supplement have the numbers used in the paper. Further, when formula from the paper is referred it is done using the number used in the paper preceded with “P”, e.g., formula (1) from the paper is referred as (P.1) in the supplement.

4 Estimation

4.1 Prior distribution

In the following 2 sections, the prior distribution (P.6) and the related joint prior $p(\boldsymbol{\theta}, \boldsymbol{\psi})$ of the vector of parameters of main interest and allocation variables introduced in Section 4.2.2 will be fully written down, separately for **Model I** and **Model D**. Subsequently, directed acyclic graphs for the proposed model will be presented.

4.2 Prior distribution in Model I

The prior (P.6) in **Model I** is equal to

$$\begin{aligned}
 p(\boldsymbol{\theta}) &= p(\{v_{i,l}, u_{i,l}, t_{i,l}\}_{i,l}, \{\zeta_{i,l}, \varepsilon_{i,l}\}_{i,l}, \{d_i, b_i\}_i, \boldsymbol{\delta}, \boldsymbol{\beta}, \mathcal{G}_\zeta, \mathcal{G}_\varepsilon, \mathcal{G}_d, \mathcal{G}_b,) \quad (1) \\
 &= \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(v_{i,l} | u_{i,l}, t_{i,l}) \times p(u_{i,l} | \boldsymbol{\delta}, d_i, \zeta_{i,l}) \times p(t_{i,l} | \boldsymbol{\beta}, b_i, \varepsilon_{i,l}) \right\} \times \\
 &\quad \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(\zeta_{i,l} | \mathcal{G}_\zeta) \times p(\varepsilon_{i,l} | \mathcal{G}_\varepsilon) \right\} \times \prod_{i=1}^N \left\{ p(d_i | \mathcal{G}_d) \times p(b_i | \mathcal{G}_b) \right\} \times \\
 &\quad p(\mathcal{G}_\zeta) \times p(\mathcal{G}_\varepsilon) \times p(\mathcal{G}_d) \times p(\mathcal{G}_b) \times p(\boldsymbol{\delta}) \times p(\boldsymbol{\beta}).
 \end{aligned}$$

The joint prior for $\boldsymbol{\theta}$ and the vector of allocation variables $\boldsymbol{\psi}$ is equal to

$$p(\boldsymbol{\theta}, \boldsymbol{\psi}) = p(\{v_{i,l}, u_{i,l}, t_{i,l}\}_{i,l}, \{\zeta_{i,l}, \varepsilon_{i,l}\}_{i,l}, \{d_i, b_i\}_i, \boldsymbol{\delta}, \boldsymbol{\beta}, \mathcal{G}_\zeta, \mathcal{G}_\varepsilon, \mathcal{G}_d, \mathcal{G}_b, \{r_{i,l}^\zeta, r_{i,l}^\varepsilon\}_{i,l}, \{r_i^d, r_i^b\}_i) \quad (2)$$

$$\begin{aligned}
 &= \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(v_{i,l} | u_{i,l}, t_{i,l}) \times p(u_{i,l} | \boldsymbol{\delta}, d_i, \zeta_{i,l}) \times p(t_{i,l} | \boldsymbol{\beta}, b_i, \varepsilon_{i,l}) \right\} \times \\
 &\quad \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(\zeta_{i,l} | \mathcal{G}_\zeta, r_{i,l}^\zeta) \times p(r_{i,l}^\zeta | \mathcal{G}_\zeta) \times p(\varepsilon_{i,l} | \mathcal{G}_\varepsilon) \times p(r_{i,l}^\varepsilon | \mathcal{G}_\varepsilon) \right\} \times \\
 &\quad \prod_{i=1}^N \left\{ p(d_i | \mathcal{G}_d) \times p(r_i^d | \mathcal{G}_d) \times p(b_i | \mathcal{G}_b) \times p(r_i^b | \mathcal{G}_b) \right\} \times \\
 &\quad p(\mathcal{G}_\zeta) \times p(\mathcal{G}_\varepsilon) \times p(\mathcal{G}_d) \times p(\mathcal{G}_b) \times p(\boldsymbol{\delta}) \times p(\boldsymbol{\beta})
 \end{aligned}$$

where the terms into which the prior (2) is decomposed are described in Section 4.2 of the paper.

4.3 Prior distribution in Model D

The prior (P.6) in **Model D** is equal to

$$\begin{aligned}
 p(\boldsymbol{\theta}) &= p(\{v_{i,l}, u_{i,l}, t_{i,l}\}_{i,l}, \{\zeta_{i,l}, \varepsilon_{i,l}\}_{i,l}, \{d_i, b_i\}_i, \boldsymbol{\delta}, \boldsymbol{\beta}, \mathcal{G}_\zeta, \mathcal{G}_\varepsilon, \mathcal{G}_d, \mathcal{G}_b,) \quad (3) \\
 &= \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(v_{i,l} | u_{i,l}, t_{i,l}) \times p(u_{i,l} | \boldsymbol{\delta}, d_i, \zeta_{i,l}) \times p(t_{i,l} | \boldsymbol{\beta}, b_i, \varepsilon_{i,l}) \right\} \times \\
 &\quad \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(\zeta_{i,l} | \mathcal{G}_\zeta) \times p(\varepsilon_{i,l} | \mathcal{G}_\varepsilon) \right\} \times \prod_{i=1}^N p(d_i, b_i | \mathbb{D}) \times \\
 &\quad p(\mathcal{G}_\zeta) \times p(\mathcal{G}_\varepsilon) \times p(\mathbb{D}^{-1}) \times p(\boldsymbol{\delta}) \times p(\boldsymbol{\beta}).
 \end{aligned}$$

The joint prior for $\boldsymbol{\theta}$ and the vector of allocation variables $\boldsymbol{\psi}$ is equal to

$$\begin{aligned}
p(\boldsymbol{\theta}, \boldsymbol{\psi}) &= p(\{v_{i,l}, u_{i,l}, t_{i,l}\}_{i,l}, \{\zeta_{i,l}, \varepsilon_{i,l}\}_{i,l}, \{d_i, b_i\}_i, \boldsymbol{\delta}, \boldsymbol{\beta}, \mathcal{G}_\zeta, \mathcal{G}_\varepsilon, \mathcal{G}_d, \mathcal{G}_b, \{r_{i,l}^\zeta, r_{i,l}^\varepsilon\}_{i,l}) \quad (4) \\
&= \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(v_{i,l} \mid u_{i,l}, t_{i,l}) \times p(u_{i,l} \mid \boldsymbol{\delta}, d_i, \zeta_{i,l}) \times p(t_{i,l} \mid \boldsymbol{\beta}, b_i, \varepsilon_{i,l}) \right\} \times \\
&\quad \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(\zeta_{i,l} \mid \mathcal{G}_\zeta, r_{i,l}^\zeta) \times p(r_{i,l}^\zeta \mid \mathcal{G}_\zeta) \times p(\varepsilon_{i,l} \mid \mathcal{G}_\varepsilon) \times p(r_{i,l}^\varepsilon \mid \mathcal{G}_\varepsilon) \right\} \times \\
&\quad \prod_{i=1}^N p(d_i, b_i \mid \mathbb{D}) \times \\
&\quad p(\mathcal{G}_\zeta) \times p(\mathcal{G}_\varepsilon) \times p(\mathbb{D}^{-1}) \times p(\boldsymbol{\delta}) \times p(\boldsymbol{\beta}),
\end{aligned}$$

where the terms into which the prior (4) is decomposed are described in Sections 4.2 and 4.3 of the paper.

4.4 Posterior distribution

The joint posterior distribution of $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ is obtained from the Bayes' formula as

$$p(\boldsymbol{\theta}, \boldsymbol{\psi} \mid [\text{data}]) \propto p([\text{data}] \mid \boldsymbol{\theta}, \boldsymbol{\psi}) \times p(\boldsymbol{\theta}, \boldsymbol{\psi}), \quad (5)$$

where $p(\boldsymbol{\theta}, \boldsymbol{\psi})$ follows from (2) or (4), respectively, and $p([\text{data}] \mid \boldsymbol{\theta}, \boldsymbol{\psi})$ is given by (P.13), i.e.,

$$\begin{aligned}
p([\text{data}] \mid \boldsymbol{\theta}, \boldsymbol{\psi}) &= p([\text{data}] \mid \{u_{i,l}\}_{i,l}, \{v_{i,l}\}_{i,l}) \quad (6) \\
&= \prod_{i=1}^N \prod_{l=1}^{n_i} \left\{ p(u_{i,l}^L, u_{i,l}^U \mid u_{i,l}, \mathbf{C}_{i,l}) \times p(v_{i,l}^L, v_{i,l}^U \mid v_{i,l}, \mathbf{C}_{i,l}) \times p(\mathbf{C}_{i,l}) \right\}.
\end{aligned}$$

The terms into which the likelihood (6) is decomposed are described in Section 4.4 of the paper.

Figures 1–6 present directed acyclic graphs (DAG) representing graphically the posterior distribution in proposed models. The DAG's use the usual convention that square boxes represent fixed or observed quantities, circles unknown parameters, solid arrows stochastic dependencies and dashed arrows deterministic dependencies.

4.5 Markov chain Monte Carlo

This section presents description on how to update the model parameters in 1 iteration of the MCMC. Mainly, full conditional distributions needed to implement the Gibbs sampler are given. We also discuss the possibilities on how to sample from the full conditional distributions if these are not of the standard form.

4.5.1 Updating the time variables

As it is seen from the DAG's in Figures 1 and 2, the nodes $u_{i,l}$, $v_{i,l}$, and $t_{i,l}$ depend all deterministically on their parents. Consequently, the update of the time variables consists in using corresponding deterministic equations. In other words,

$$\begin{aligned}
p(v_{i,l} \mid \cdots) &= p(v_{i,l} \mid u_{i,l}, t_{i,l}) &= I[v_{i,l} = u_{i,l} + t_{i,l}] \\
p(u_{i,l} \mid \cdots) &= p(u_{i,l} \mid \boldsymbol{\delta}, d_i, \zeta_{i,l}) &= I[u_{i,l} = \exp(d_i + \boldsymbol{\delta}' \mathbf{z}_{i,l} + \zeta_{i,l})] \\
p(t_{i,l} \mid \cdots) &= p(t_{i,l} \mid \boldsymbol{\beta}, b_i, \varepsilon_{i,l}) &= I[t_{i,l} = \exp(b_i + \boldsymbol{\beta}' \mathbf{x}_{i,l} + \varepsilon_{i,l})]
\end{aligned}$$

4.5.2 Updating the error terms

The update of the ‘onset’ error terms $\zeta_{i,l}$ is fully deterministic provided the (i,l) th onset time $u_{i,l} = u_{i,l}^L = u_{i,l}^U$ is *uncensored*. The update of $\zeta_{i,l}$ consists then of using the AFT expression (P.1) with the current values of the parameters, i.e.,

$$p(\zeta_{i,l} \mid \dots) = I[\log(u_{i,l}) - d_i - \boldsymbol{\delta}'\mathbf{z}_{i,l}].$$

When the (i,l) th onset time is *censored* with an observed interval $[u_{i,l}^L, u_{i,l}^U]$, the full conditional distribution of $\zeta_{i,l}$ is truncated normal, namely

$$\zeta_{i,l} \mid \dots \sim \mathcal{N}\left(\alpha^\zeta + \tau^\zeta \mu_{r^\zeta}, (\sigma^\zeta \tau^\zeta)^2\right)$$

truncated on $\left[\log(u_{i,l}^L) - d_i - \boldsymbol{\delta}'\mathbf{z}_{i,l}, \log(u_{i,l}^U) - d_i - \boldsymbol{\delta}'\mathbf{z}_{i,l}\right]$.

A similar procedure is used when updating the ‘event’ error terms $\varepsilon_{i,l}$. It is useful to stress that for the update of $\varepsilon_{i,l}$ also the ‘onset’ error term $\zeta_{i,l}$ and subsequently also the true onset time $u_{i,l} = \exp(d_i + \boldsymbol{\delta}'\mathbf{z}_{i,l} + \zeta_{i,l})$ make a part of the condition when exploiting the full conditional distribution. This implies that the update of $\varepsilon_{i,l}$ is fully deterministic provided

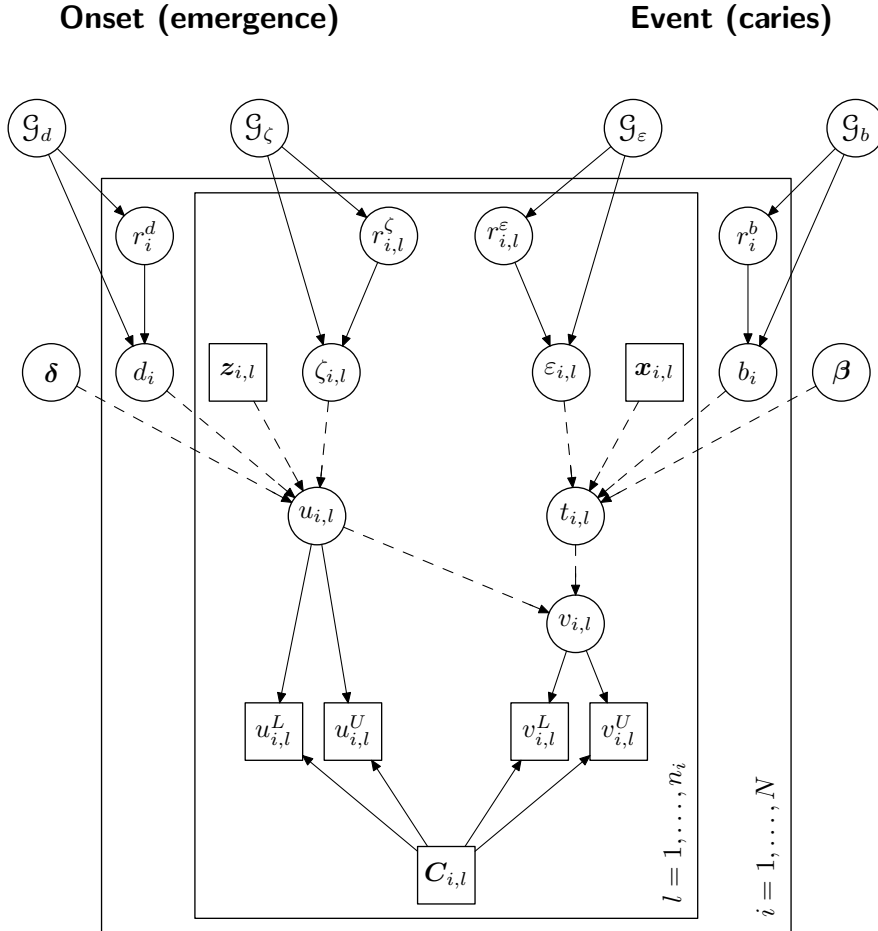


Figure 1: Directed acyclic graph for **Model I**. The structure of the nodes \mathcal{G}_ζ , \mathcal{G}_ε , \mathcal{G}_d , and \mathcal{G}_b is further exemplified in Figures 3–6.

the (i, l) th failure time $v_{i,l} = v_{i,l}^L = v_{i,l}^U$ is *uncensored*, irrespective whether the onset time is censored or not. The update of $\varepsilon_{i,l}$ consists then of using the AFT expression (P.2) with the current values of the parameters, i.e.,

$$p(\varepsilon_{i,l} \mid \cdots) = I[\log(v_{i,l} - u_{i,l}) - b_i - \beta' \mathbf{x}_{i,l}].$$

When the error term $\varepsilon_{i,l}$ corresponds to the *censored* failure time with an observed interval $[v_{i,l}^L, v_{i,l}^U]$ its update consists of the sampling from the full conditional distribution of $\varepsilon_{i,l}$ which is here a truncated normal distribution, namely

$$\varepsilon_{i,l} \mid \cdots \sim \mathcal{N}\left(\alpha^\varepsilon + \tau^\varepsilon \mu_{r_{i,l}^\varepsilon}, (\sigma^\varepsilon \tau^\varepsilon)^2\right)$$

truncated on $\left[\log(v_{i,l}^L - u_{i,l}) - b_i - \beta' \mathbf{x}_{i,l}, \log(v_{i,l}^U - u_{i,l}) - b_i - \beta' \mathbf{x}_{i,l}\right]$.

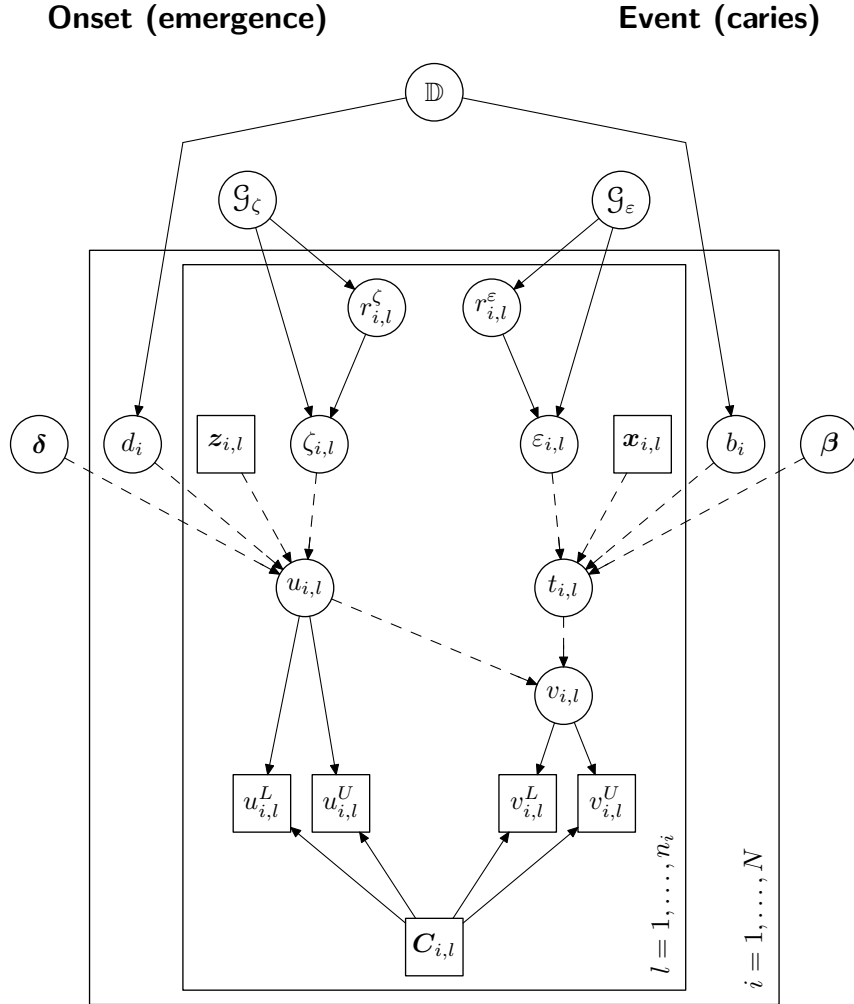


Figure 2: Directed acyclic graph for **Model D**. The structure of the nodes \mathcal{G}_ζ , \mathcal{G}_ε is further exemplified in Figures 3 and 4.

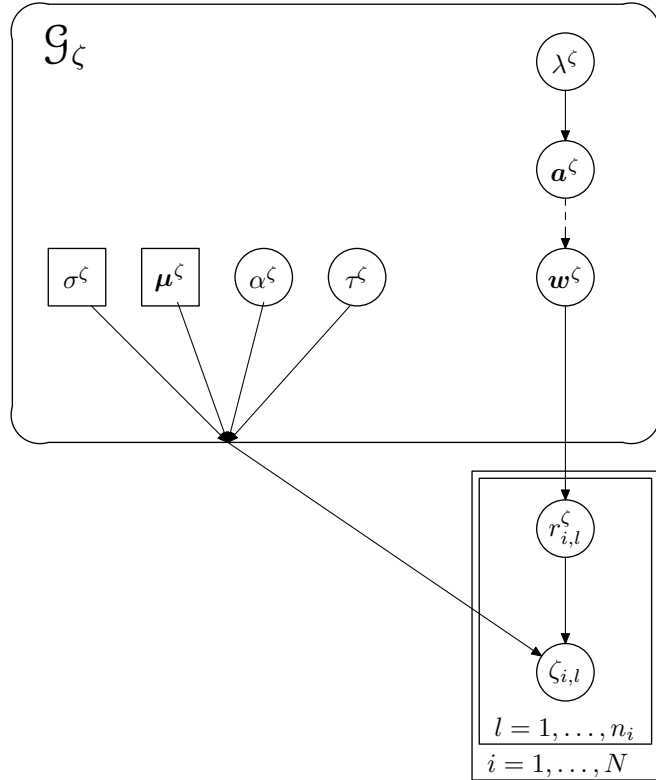


Figure 3: Directed acyclic graph for the penalized Gaussian mixture \mathcal{G}_ζ in both **Model I** and **Model D**.

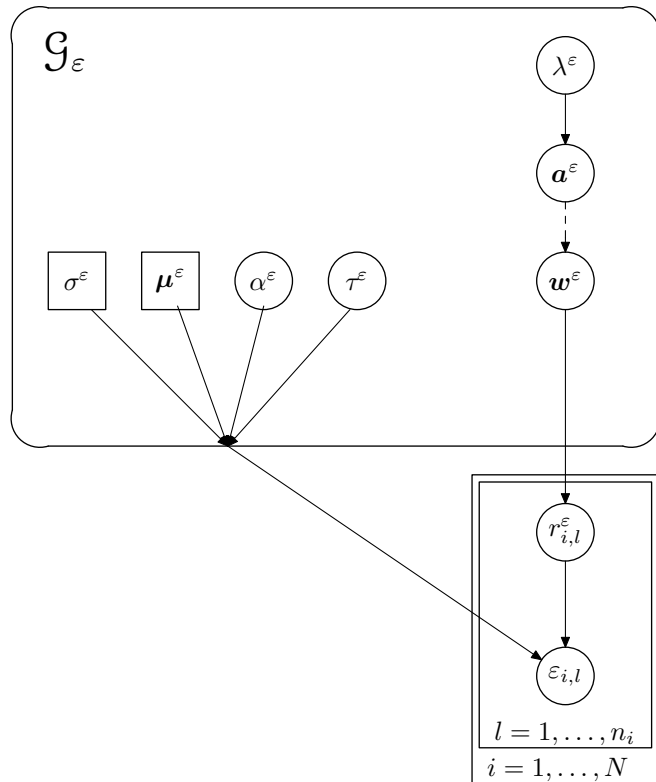


Figure 4: Directed acyclic graph for the penalized Gaussian mixture \mathcal{G}_ϵ in both **Model I** and **Model D**.

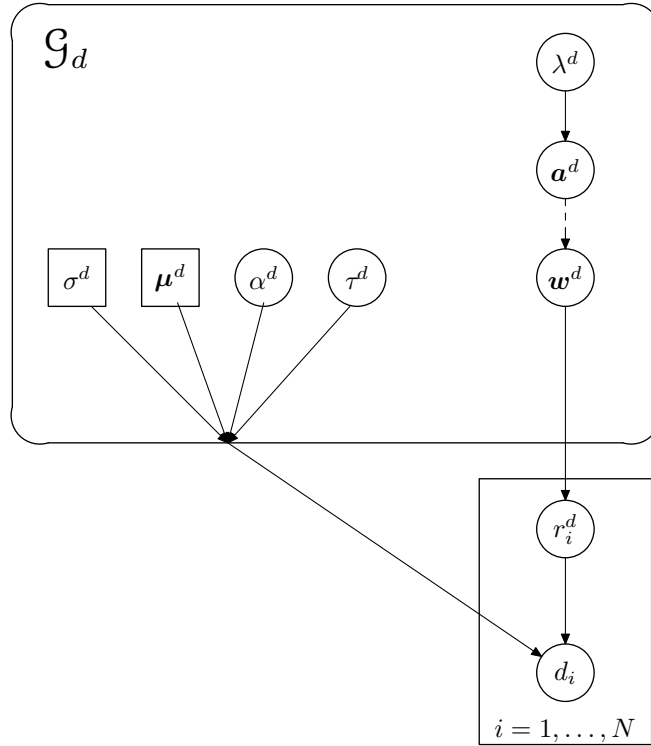


Figure 5: Directed acyclic graph for the penalized Gaussian mixture \mathcal{G}_d in **Model I**.

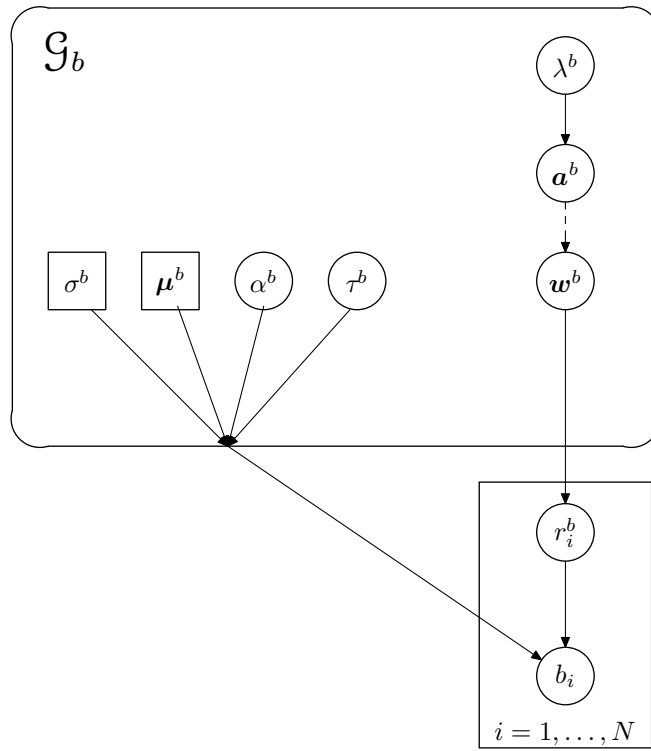


Figure 6: Directed acyclic graph for the penalized Gaussian mixture \mathcal{G}_b in **Model I**.

4.5.3 Updating the random effects in Model I

In **Model I**, the full conditional distributions for the random effects d_i and/or b_i are normal distributions, namely

$$\begin{aligned} d_i \mid \dots &\sim \mathcal{N}\left(\mathbb{E}(d_i \mid \dots), \text{var}(d_i \mid \dots)\right), \\ b_i \mid \dots &\sim \mathcal{N}\left(\mathbb{E}(b_i \mid \dots), \text{var}(b_i \mid \dots)\right), \end{aligned}$$

with

$$\begin{aligned} \mathbb{E}(d_i \mid \dots) &= \text{var}(d_i \mid \dots) \times \\ &\quad \left[(\sigma^d \tau^d)^{-2} \tau^d \mu_{r_i^d}^d + (\sigma^\zeta \tau^\zeta)^{-2} \sum_{l=1}^{n_i} \{ \log(u_{i,l}) - \alpha^\zeta - \boldsymbol{\delta}' \mathbf{z}_{i,l} - \tau^\zeta \mu_{r_{i,l}^\zeta}^\zeta \} \right], \\ \text{var}(d_i \mid \dots) &= \left\{ (\sigma^d \tau^d)^{-2} + n_i (\sigma^\zeta \tau^\zeta)^{-2} \right\}^{-1}, \end{aligned}$$

$$\begin{aligned} \mathbb{E}(b_i \mid \dots) &= \text{var}(b_i \mid \dots) \times \\ &\quad \left[(\sigma^b \tau^b)^{-2} \tau^b \mu_{r_i^b}^b + (\sigma^\varepsilon \tau^\varepsilon)^{-2} \sum_{l=1}^{n_i} \{ \log(t_{i,l}) - \alpha^\varepsilon - \boldsymbol{\beta}' \mathbf{x}_{i,l} - \tau^\varepsilon \mu_{r_{i,l}^\varepsilon}^\varepsilon \} \right], \\ \text{var}(b_i \mid \dots) &= \left\{ (\sigma^b \tau^b)^{-2} + n_i (\sigma^\varepsilon \tau^\varepsilon)^{-2} \right\}^{-1}. \end{aligned}$$

4.5.4 Updating the random effects in Model D

In **Model D**, the full conditional distributions for the bivariate random effects vector $(d_i, b_i)'$ is the bivariate normal distribution, namely

$$(d_i, b_i)' \mid \dots \sim \mathcal{N}_2\left(\mathbb{E}\{(d_i, b_i)' \mid \dots\}, \text{var}\{(d_i, b_i)' \mid \dots\}\right),$$

with

$$\begin{aligned} \mathbb{E}\{(d_i, b_i)' \mid \dots\} &= \text{var}\{(d_i, b_i)' \mid \dots\} \times \begin{pmatrix} (\sigma^\zeta \tau^\zeta)^{-2} \sum_{l=1}^{n_i} \{ \log(u_{i,l}) - \alpha^\zeta - \boldsymbol{\delta}' \mathbf{z}_{i,l} - \tau^\zeta \mu_{r_{i,l}^\zeta}^\zeta \} \\ (\sigma^\varepsilon \tau^\varepsilon)^{-2} \sum_{l=1}^{n_i} \{ \log(t_{i,l}) - \alpha^\varepsilon - \boldsymbol{\beta}' \mathbf{x}_{i,l} - \tau^\varepsilon \mu_{r_{i,l}^\varepsilon}^\varepsilon \} \end{pmatrix}, \\ \text{var}\{(d_i, b_i)' \mid \dots\} &= \left\{ \mathbb{D}^{-1} + n_i \begin{pmatrix} (\sigma^\zeta \tau^\zeta)^{-2} & 0 \\ 0 & (\sigma^\varepsilon \tau^\varepsilon)^{-2} \end{pmatrix} \right\}^{-1}. \end{aligned}$$

4.5.5 Updating the regression parameters

We assume that the prior distribution for the regression parameters $\boldsymbol{\delta}$ and $\boldsymbol{\beta}$ is a product of normals. That is, a priori

$$\boldsymbol{\delta} \sim \mathcal{N}(\boldsymbol{\nu}_\delta, \boldsymbol{\Psi}_\delta), \quad \boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{\nu}_\beta, \boldsymbol{\Psi}_\beta),$$

where matrices Ψ_δ and Ψ_β are diagonal. The full conditional distributions for both δ and β are (multivariate) normal distributions, namely

$$\begin{aligned}\delta \mid \dots &\sim \mathcal{N}\left(\mathbb{E}(\delta \mid \dots), \text{var}(\delta \mid \dots)\right), \\ \beta \mid \dots &\sim \mathcal{N}\left(\mathbb{E}(\beta \mid \dots), \text{var}(\beta \mid \dots)\right),\end{aligned}$$

with

$$\begin{aligned}\mathbb{E}(\delta \mid \dots) &= \text{var}(\delta \mid \dots) \times \left[\Psi_\delta^{-1} \boldsymbol{\nu}_\delta + (\sigma^\zeta \tau^\zeta)^{-2} \sum_{i=1}^N \sum_{l=1}^{n_i} \mathbf{z}_{i,l} \{ \log(u_{i,l}) - \alpha^\zeta - d_i - \tau^\zeta \mu_{r_{i,l}^\zeta}^\zeta \} \right], \\ \text{var}(\delta \mid \dots) &= \left\{ (\Psi_\delta^{-1} + (\sigma^\zeta \tau^\zeta)^{-2} \sum_{i=1}^N \sum_{l=1}^{n_i} \mathbf{z}_{i,l} \mathbf{z}'_{i,l}) \right\}^{-1}, \\ \mathbb{E}(\beta \mid \dots) &= \text{var}(\beta \mid \dots) \times \left[\Psi_\beta^{-1} \boldsymbol{\nu}_\beta + (\sigma^\varepsilon \tau^\varepsilon)^{-2} \sum_{i=1}^N \sum_{l=1}^{n_i} \mathbf{x}_{i,l} \{ \log(t_{i,l}) - \alpha^\varepsilon - b_i - \tau^\varepsilon \mu_{r_{i,l}^\varepsilon}^\varepsilon \} \right], \\ \text{var}(\beta \mid \dots) &= \left\{ (\Psi_\beta^{-1} + (\sigma^\varepsilon \tau^\varepsilon)^{-2} \sum_{i=1}^N \sum_{l=1}^{n_i} \mathbf{x}_{i,l} \mathbf{x}'_{i,l}) \right\}^{-1}.\end{aligned}$$

Note that the full conditional distributions for δ and β remain the same even if the prior covariance matrices Ψ_δ or Ψ_β are not diagonal.

4.5.6 Updating the covariance matrix of the random effects in Model D

We assume that the prior distribution for the inverse of the covariance matrix of the random effects in **Model D** is Wishart with df degrees of freedom and a scale matrix \mathbb{S} . That is, a priori

$$\mathbb{D}^{-1} \sim \text{Wishart}(df, \mathbb{S}).$$

The update of \mathbb{D}^{-1} consists of sampling from its full conditional distribution which is again the Wishart distribution, namely

$$\mathbb{D}^{-1} \mid \dots \sim \text{Wishart} \left(df + N, \left\{ \mathbb{S}^{-1} + \begin{pmatrix} \sum_{i=1}^N d_i^2 & \sum_{i=1}^N d_i b_i \\ \sum_{i=1}^N d_i b_i & \sum_{i=1}^N b_i^2 \end{pmatrix} \right\}^{-1} \right).$$

4.5.7 Updating the parameters related to the penalized Gaussian mixture

In a sequel, let $\mathcal{G} = \{\sigma, \boldsymbol{\mu}, \alpha, \tau, \mathbf{w}, \mathbf{a}, \lambda\}$ stand for $\mathcal{G}_\zeta, \mathcal{G}_\varepsilon, \mathcal{G}_d$, or \mathcal{G}_b , respectively. Further, let $y_{i^*}, i^* = 1, \dots, n$ be the current values of $\zeta_{i,l}, \varepsilon_{i,l}, d_i$, or b_i , respectively and $r_{i^*}, i^* = 1, \dots, n$ corresponding latent allocation variables. That is,

- For \mathcal{G}^ε we have

$$\begin{aligned}\{y_{i^*} : i^* = 1, \dots, n\} &= \{\varepsilon_{i,l} : i = 1, \dots, N, l = 1, \dots, n_i\}, \\ \{r_{i^*} : i^* = 1, \dots, n\} &= \{r_{i,l}^\varepsilon : i = 1, \dots, N, l = 1, \dots, n_i\}, \quad \text{and } n = \sum_{i=1}^N n_i.\end{aligned}$$

- For \mathcal{G}^ζ we have

$$\begin{aligned} \{y_{i^*} : i^* = 1, \dots, n\} &= \{\zeta_{i,l} : i = 1, \dots, N, l = 1, \dots, n_i\}, \\ \{r_{i^*} : i^* = 1, \dots, n\} &= \{r_{i,l}^\zeta : i = 1, \dots, N, l = 1, \dots, n_i\}, \quad \text{and } n = \sum_{i=1}^N n_i. \end{aligned}$$

- For \mathcal{G}^b we have

$$\begin{aligned} \{y_{i^*} : i^* = 1, \dots, n\} &= \{b_i : i = 1, \dots, N\}, \\ \{r_{i^*} : i^* = 1, \dots, n\} &= \{r_i^b : i = 1, \dots, N\}, \quad \text{and } n = N. \end{aligned}$$

- For \mathcal{G}^d we have

$$\begin{aligned} \{y_{i^*} : i^* = 1, \dots, n\} &= \{d_i : i = 1, \dots, N\}, \\ \{r_{i^*} : i^* = 1, \dots, n\} &= \{r_i^d : i = 1, \dots, N\}, \quad \text{and } n = N. \end{aligned}$$

Updating the transformed mixture weights

The full conditional of each element of \mathbf{a} is given by

$$\begin{aligned} p(a_j | \dots) &\propto \frac{\exp(N_j a_j)}{\left\{ \sum_{k=-K}^K \exp(a_k) \right\}^n} \times \exp \left[-\frac{\left\{ a_j - \mathbb{E}(a_j | \mathbf{a}_{-(j)}, \lambda) \right\}^2}{2 \text{var}(a_j | \mathbf{a}_{-(j)}, \lambda)} \right], \\ j &= -K, \dots, K, \end{aligned} \tag{7}$$

where N_j is the number of y_{i^*} for which the latent allocation variable r_{i^*} is equal to j , i.e.

$$N_j = \sum_{i^*=1}^n I[r_{i^*} = j].$$

Further, $\mathbb{E}(a_j | \mathbf{a}_{-(j)}, \lambda)$ and $\text{var}(a_j | \mathbf{a}_{-(j)}, \lambda)$ are the mean and the variance resulting from the IGMRF prior (P.12). For example, for the third order differences ($s = 3$), which have been used in the Signal Tandmobiel[®] example, we have

$$\begin{aligned} \mathbb{E}(a_j | \mathbf{a}_{-(j)}) &= \frac{a_{j-3} - 6a_{j-2} + 15a_{j-1} + 15a_{j+1} - 6a_{j+2} + a_{j+3}}{20}, \\ j &= -K + 3, \dots, K - 3, \end{aligned}$$

$$\mathbb{E}(a_{-K+2} | \mathbf{a}_{-(-K+2)}) = \frac{-3a_{-K} + 12a_{-K+1} + 15a_{-K+3} - 6a_{-K+4} + a_{-K+5}}{19},$$

$$\mathbb{E}(a_{K-2} | \mathbf{a}_{-(K-2)}) = \frac{-3a_K + 12a_{K-1} + 15a_{K-3} - 6a_{K-4} + a_{K-5}}{19},$$

$$\mathbb{E}(a_{-K+1} | \mathbf{a}_{-(-K+1)}) = \frac{3a_{-K} + 12a_{-K+2} - 6a_{-K+3} + a_{-K+4}}{10},$$

$$\mathbb{E}(a_{K-1} | \mathbf{a}_{-(K-1)}) = \frac{3a_K + 12a_{K-2} - 6a_{K-3} + a_{K-4}}{10},$$

$$\mathbb{E}(a_{-K} \mid \mathbf{a}_{-(-K)}) = 3a_{-K+1} - 3a_{-K+2} + a_{-K+3},$$

$$\mathbb{E}(a_K \mid \mathbf{a}_{-(K)}) = 3a_{K-1} - 3a_{K-2} + a_{K-3},$$

and

$$\begin{aligned} \text{var}(a_j \mid \mathbf{a}_{-(j)}) &= (20\lambda)^{-1}, \quad j = -K+3, \dots, K-3, \\ \text{var}(a_{-K+2} \mid \mathbf{a}_{-(-K+2)}) &= \text{var}(a_{K-2} \mid \mathbf{a}_{-(K-2)}) = (19\lambda)^{-1}, \\ \text{var}(a_{-K+1} \mid \mathbf{a}_{-(-K+1)}) &= \text{var}(a_{K-1} \mid \mathbf{a}_{-(K-1)}) = (10\lambda)^{-1}, \\ \text{var}(a_{-K} \mid \mathbf{a}_{-(K)}) &= \text{var}(a_K \mid \mathbf{a}_{-(K)}) = \lambda^{-1}. \end{aligned}$$

Distribution (7) is log-concave so we experimented both with the slice sampler of Neal (2003) as well as with the adaptive rejection sampling (ARS) method of Gilks and Wild (1992) to update the elements of \mathbf{a} . However, in our applications no method was found to be superior with respect to the performance of the MCMC. The results for the Signal Tandmobiell[®] were obtained using slice sampling.

Furthermore, it is seen that the full conditional distribution for each transformed mixture weight depends only on the weights of the neighboring mixture components. For a better performance of the MCMC, especially to decrease the autocorrelation of the sampled chain, it is thus advantageous to update in one iteration of the MCMC the transformed mixture weights in such an order that the full conditional of a we are updating does not depend on a which has just been updated. This is obtained, for example, using the following update order:

$$\dots \rightarrow a_0 \rightarrow a_{s+1} \rightarrow a_{2(s+1)} \rightarrow \dots \rightarrow a_1 \rightarrow a_{1+s+1} \rightarrow a_{1+2(s+1)} \rightarrow \dots$$

Updating the smoothing hyperparameter

Suppose the prior for the smoothing hyperparameter λ is Gamma, namely a priori

$$\lambda \sim \text{G}(h_1^\lambda, h_2^\lambda).$$

The full conditional distribution for λ is again Gamma, namely

$$\lambda \mid \dots \sim \text{G}\left(h_1^\lambda + \frac{2K+1-s+1}{2}, h_2^\lambda + \frac{1}{2}\mathbf{a}'\mathbb{P}'_s\mathbb{P}_s\mathbf{a}\right).$$

Updating the intercept of the PGM

Suppose the prior for the PGM intercept is normal, namely a priori

$$\alpha \sim \mathcal{N}(\nu_\alpha, \Psi_\alpha).$$

The full conditional distribution for α is again normal, namely

$$\alpha \mid \dots \sim \mathcal{N}\left(\mathbb{E}(\alpha \mid \dots), \text{var}(\alpha \mid \dots)\right),$$

with

$$\begin{aligned} \mathbb{E}(\alpha \mid \dots) &= \text{var}(\alpha \mid \dots) \times \left\{ \psi_\alpha^{-1} \nu_\alpha + (\sigma\tau)^{-2} \sum_{i^*=1}^n (y_{i^*} - \tau\mu_{\tau_{i^*}}) \right\}, \\ \text{var}(\alpha \mid \dots) &= \left\{ \psi_\alpha^{-1} + n(\sigma\tau)^{-2} \right\}^{-1}. \end{aligned}$$

Updating the scale of the PGM

Suppose the prior for the inverse of the variance of PGM is Gamma, namely a priori

$$\tau^{-2} \sim G(h_1^\tau, h_2^\tau).$$

The full conditional distribution of τ^{-2} has the form

$$p(\tau^{-2} | \dots) \propto (\tau^{-2})^{\xi_1-1} \exp\left(\xi_3\sqrt{\tau^{-2}} - \xi_2\tau^{-2}\right), \quad (8)$$

with

$$\begin{aligned} \xi_1 &= h_1^\tau + 0.5 n, \\ \xi_2 &= h_2^\tau + 0.5 \sigma^{-2} \sum_{i^*=1}^n (y_{i^*} - \alpha)^2, \\ \xi_3 &= \sigma^{-2} \sum_{i^*=1}^n \mu_{r_{i^*}} (y_{i^*} - \alpha). \end{aligned}$$

Distribution (8) is generally not log-concave so that the adaptive rejection sampling (ARS) method of Gilks and Wild (1992), successfully used in many situations when the full conditional distribution does not have a standard form, cannot be used here. However, it can easily be shown that the density (8) is always unimodal and a simpler version of the slice sampler of Neal (2003) can be used to update the parameter τ^{-2} in an MCMC run.

4.5.8 Updating the allocation variables

The full conditional for each allocation variable r_{i^*} , $i^* = 1, \dots, n$ is discrete with

$$\Pr(r_{i^*} = j | \dots) \propto w_j \exp\left\{-\frac{(y_{i^*} - \alpha - \tau\mu_j)^2}{2(\sigma\tau)^2}\right\}, \quad j \in \{-K, \dots, K\}.$$

4.5.9 Software

A software package, called `bayesSurv`, was written combining the R language (R Development Core Team, 2006) with programs in C++, and is available from the *Comprehensive R Archive Network* on <http://www.R-project.org>. Specifically, MCMC sampling is performed by the function `bayessurvreg3`, the estimates of the densities g_ζ , g_ε , g_d , $g(b)$ are obtained using the function `bayesGspline` and predictive survivor or hazard functions for specified combinations of covariates are computed using the function `predictive2`.

5 Simulation study

This section presents more detailed results of the simulation study described in the paper. Tables 1 and 2 show the results for the regression parameters. Tables 3 and 4 give the results for the variance components of the model.

Figures 7 and 8 show the fitted survivor densities for the onset part of the model for a combination of covariates $z_{i,l,1} = 0.5$ (median value) and $z_{i,l,2} = 1$. Figures 9 and 10 give the fitted survivor densities for the event part of the model for a combination of covariates $x_{i,l,1} = 0.5$ (median value) and $x_{i,l,2} = 1$. Corresponding fitted survivor functions are given in Figures 11 – 14.

Table 1: Results for the regression parameters from the **onset** part of the model. Mean, standard deviation and MSE ($\times 10^{-4}$) over the simulation.

	$\delta_1 = 0.200$		$\delta_2 = -0.100$	
$\tau^d/\tau^\zeta =$		MSE		MSE
τ^b/τ^ε	$\hat{\delta}_1$ (SD)	($\times 10^{-4}$)	$\hat{\delta}_2$ (SD)	($\times 10^{-4}$)
Scenario I				
(error \sim normal mixture, random effect \sim extreme value)				
5	0.199 (0.007)	0.56	-0.101 (0.004)	0.17
3	0.201 (0.008)	0.68	-0.100 (0.005)	0.20
2	0.198 (0.011)	1.30	-0.100 (0.006)	0.37
1	0.199 (0.014)	1.84	-0.100 (0.009)	0.76
1/2	0.200 (0.018)	3.14	-0.100 (0.010)	0.92
1/3	0.201 (0.019)	3.74	-0.101 (0.010)	1.02
1/5	0.198 (0.019)	3.51	-0.100 (0.010)	0.95
Scenario II				
(error \sim extreme value, random effect \sim normal mixture)				
5	0.200 (0.010)	0.93	-0.101 (0.005)	0.30
3	0.202 (0.015)	2.38	-0.101 (0.008)	0.72
2	0.200 (0.019)	3.44	-0.099 (0.011)	1.27
1	0.196 (0.029)	8.73	-0.099 (0.019)	3.45
1/2	0.194 (0.038)	14.46	-0.097 (0.025)	6.30
1/3	0.201 (0.041)	16.73	-0.099 (0.024)	5.90
1/5	0.203 (0.043)	18.12	-0.100 (0.020)	4.12

Table 2: Results for the regression parameters from the **event** part of the model. Mean, standard deviation and MSE ($\times 10^{-4}$) over the simulation.

	$\beta_1 = 0.300$		$\beta_2 = -0.150$	
$\tau^d/\tau^\zeta =$		MSE		MSE
τ^b/τ^ε	$\hat{\beta}_1$ (SD)	($\times 10^{-4}$)	$\hat{\beta}_2$ (SD)	($\times 10^{-4}$)
Scenario I				
(error \sim normal mixture, random effect \sim extreme value)				
5	0.302 (0.014)	2.12	-0.149 (0.008)	0.64
3	0.301 (0.032)	9.99	-0.149 (0.021)	4.47
2	0.298 (0.056)	30.55	-0.150 (0.034)	11.75
1	0.304 (0.054)	29.04	-0.148 (0.028)	7.55
1/2	0.301 (0.043)	18.07	-0.147 (0.031)	9.66
1/3	0.311 (0.058)	34.67	-0.150 (0.035)	11.88
1/5	0.299 (0.050)	25.11	-0.151 (0.031)	9.68
Scenario II				
(error \sim extreme value, random effect \sim normal mixture)				
5	0.298 (0.031)	9.40	-0.148 (0.016)	2.74
3	0.291 (0.040)	16.44	-0.152 (0.022)	4.99
2	0.306 (0.065)	42.02	-0.146 (0.036)	13.01
1	0.299 (0.103)	105.54	-0.149 (0.057)	32.60
1/2	0.304 (0.121)	144.59	-0.151 (0.070)	48.40
1/3	0.296 (0.126)	157.36	-0.146 (0.071)	50.06
1/5	0.308 (0.112)	125.51	-0.142 (0.065)	42.10

Table 3: Results for the scale parameters from the **onset** part of the model. Mean, standard deviation and MSE ($\times 10^{-4}$) over the simulation.

$\tau^d/\tau^\zeta =$ τ^b/τ^ε	τ^d			τ^ζ		
	True τ^d	$\hat{\tau}^d$ (SD)	MSE ($\times 10^{-4}$)	True τ^ζ	$\hat{\tau}^\zeta$ (SD)	MSE ($\times 10^{-4}$)
Scenario I (error \sim normal mixture, random effect \sim extreme value)						
5	0.310	0.341 (0.035)	21.20	0.062	0.062 (0.002)	0.04
3	0.300	0.324 (0.037)	19.13	0.100	0.100 (0.002)	0.04
2	0.283	0.283 (0.031)	9.31	0.141	0.141 (0.003)	0.08
1	0.224	0.219 (0.024)	5.85	0.224	0.223 (0.006)	0.39
1/2	0.141	0.143 (0.018)	3.24	0.283	0.283 (0.006)	0.31
1/3	0.100	0.103 (0.035)	12.23	0.300	0.301 (0.012)	1.35
1/5	0.062	0.110 (0.097)	116.93	0.310	0.325 (0.034)	13.74
Scenario II (error \sim extreme value, random effect \sim normal mixture)						
5	0.310	0.311 (0.009)	0.86	0.062	0.061 (0.003)	0.11
3	0.300	0.318 (0.112)	128.44	0.100	0.116 (0.099)	99.25
2	0.283	0.299 (0.141)	198.36	0.141	0.159 (0.126)	159.21
1	0.224	0.218 (0.011)	1.54	0.224	0.224 (0.012)	1.35
1/2	0.141	0.132 (0.021)	5.23	0.283	0.285 (0.013)	1.70
1/3	0.100	0.065 (0.037)	25.76	0.300	0.304 (0.013)	1.83
1/5	0.062	0.040 (0.030)	13.87	0.310	0.314 (0.015)	2.34

Table 4: Results for the scale parameters from the **event** part of the model. Mean, standard deviation and MSE ($\times 10^{-4}$) over the simulation.

$\tau^d/\tau^\zeta =$ τ^b/τ^ε	τ^b			τ^ε		
	True τ^b	$\hat{\tau}^b$ (SD)	MSE ($\times 10^{-4}$)	True τ^ε	$\hat{\tau}^\varepsilon$ (SD)	MSE ($\times 10^{-4}$)
Scenario I (error \sim normal mixture, random effect \sim extreme value)						
5	0.981	0.980 (0.393)	1532.34	0.196	0.202 (0.005)	0.60
3	0.949	0.987 (0.517)	2660.34	0.316	0.417 (0.160)	356.10
2	0.894	0.827 (0.065)	87.10	0.447	0.663 (0.217)	932.57
1	0.707	0.647 (0.046)	57.34	0.707	0.741 (0.090)	91.97
1/2	0.447	0.428 (0.039)	18.27	0.894	0.901 (0.017)	3.47
1/3	0.316	0.307 (0.037)	14.42	0.949	0.954 (0.018)	3.57
1/5	0.196	0.180 (0.037)	15.93	0.981	0.984 (0.018)	3.49
Scenario II (error \sim extreme value, random effect \sim normal mixture)						
5	0.981	0.971 (0.030)	9.67	0.196	0.202 (0.012)	1.73
3	0.949	0.941 (0.040)	16.72	0.316	0.325 (0.054)	29.79
2	0.894	0.884 (0.049)	24.54	0.447	0.532 (0.237)	626.75
1	0.707	0.671 (0.040)	28.78	0.707	0.886 (0.273)	1056.64
1/2	0.447	0.394 (0.092)	111.54	0.894	1.160 (0.230)	1228.67
1/3	0.316	0.079 (0.115)	695.18	0.949	1.286 (0.214)	1589.69
1/5	0.196	0.024 (0.026)	302.36	0.981	1.345 (0.235)	1873.27

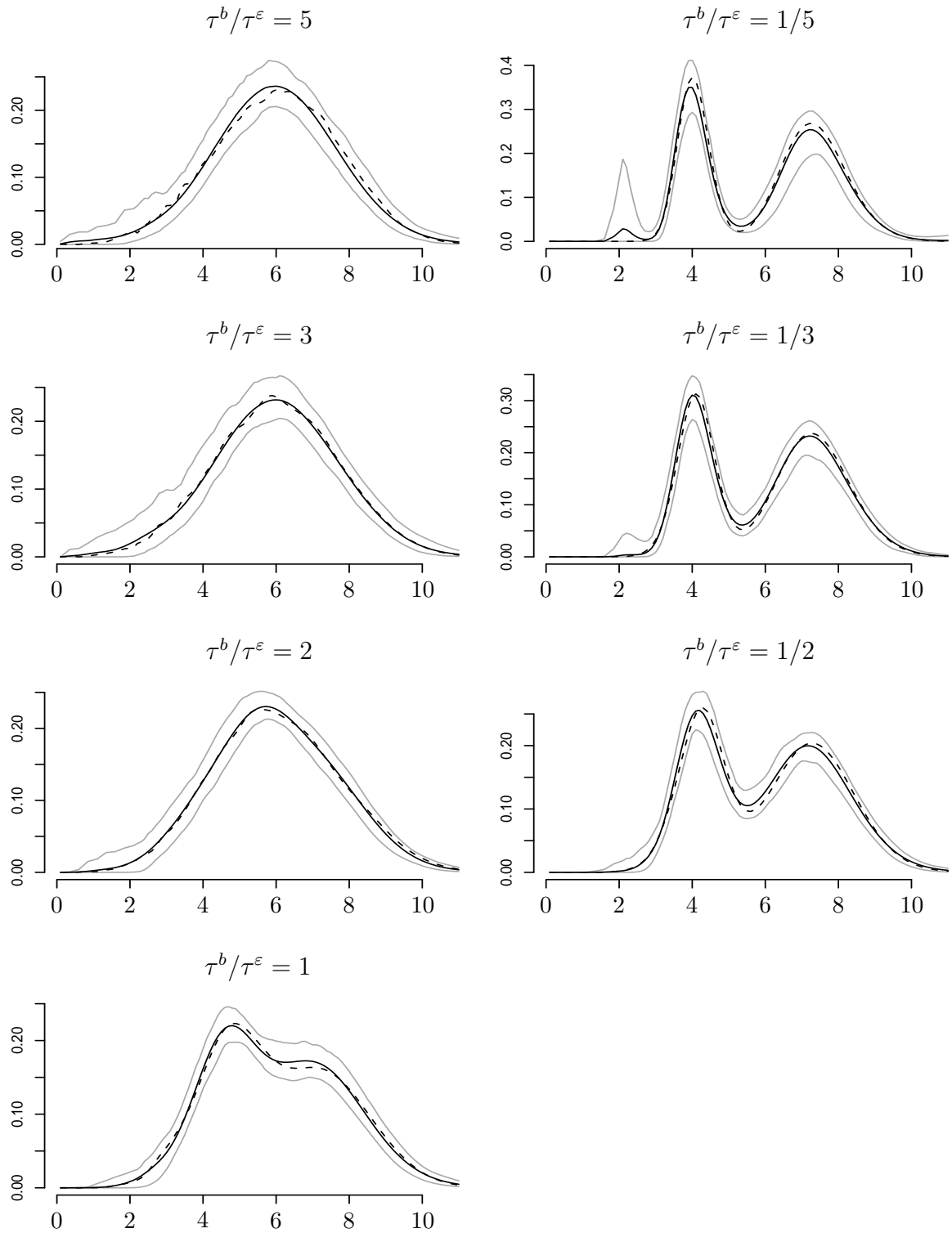


Figure 7: Results for the survivor density of the **onset** time, for the combination of covariates $z_{i,l} = (0.5, 1)'$, scenario I (error \sim normal mixture, random effect \sim extreme value). Solid line: average fitted survivor density, grey lines: 95% pointwise confidence band, dashed line: true survivor density..

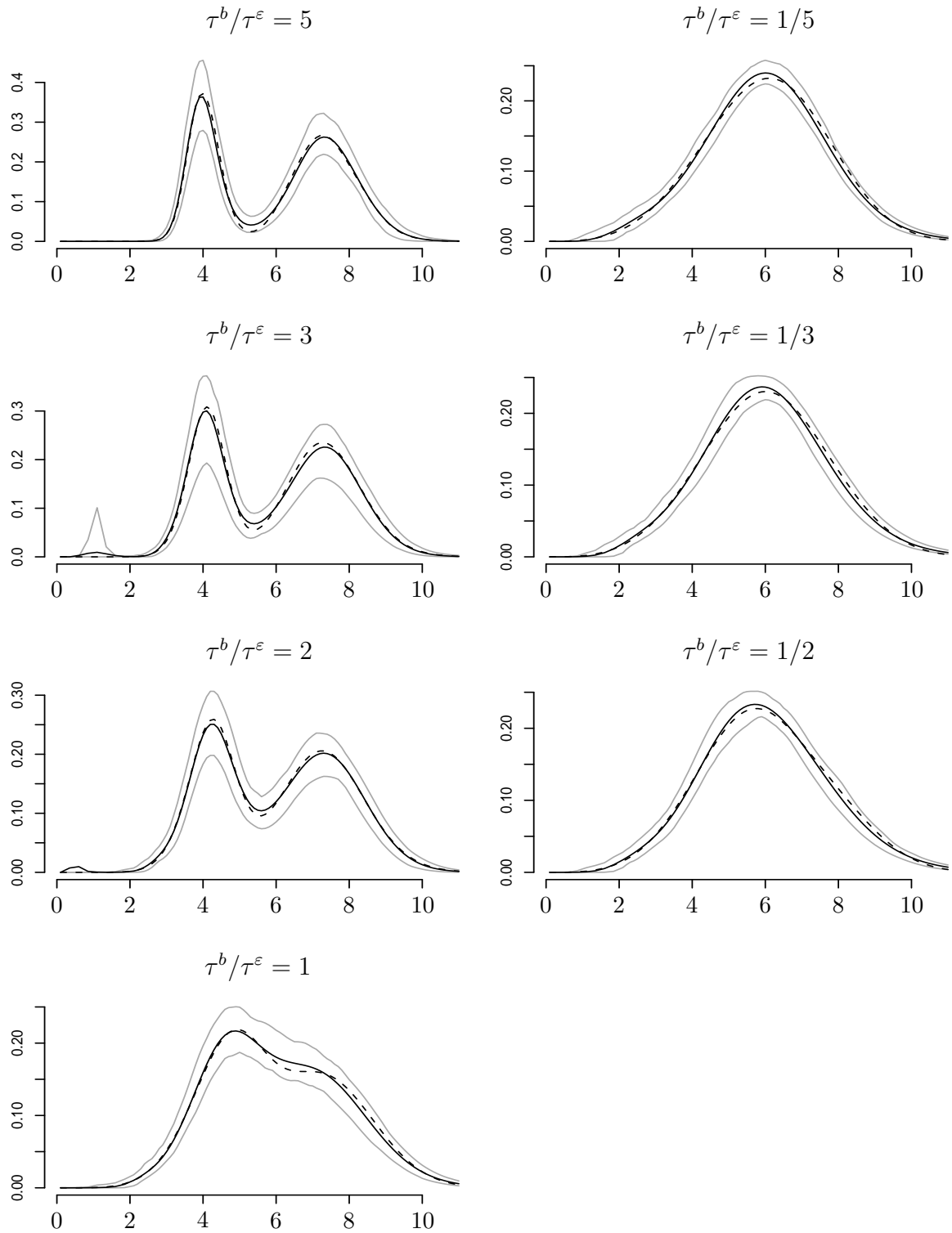


Figure 8: Results for the survivor density of the **onset** time, for the combination of covariates $\mathbf{z}_{i,l} = (0.5, 1)'$, scenario II (error \sim extreme value, random effect \sim normal mixture). Solid line: average fitted survivor density, grey lines: 95% pointwise confidence band, dashed line: true survivor density..

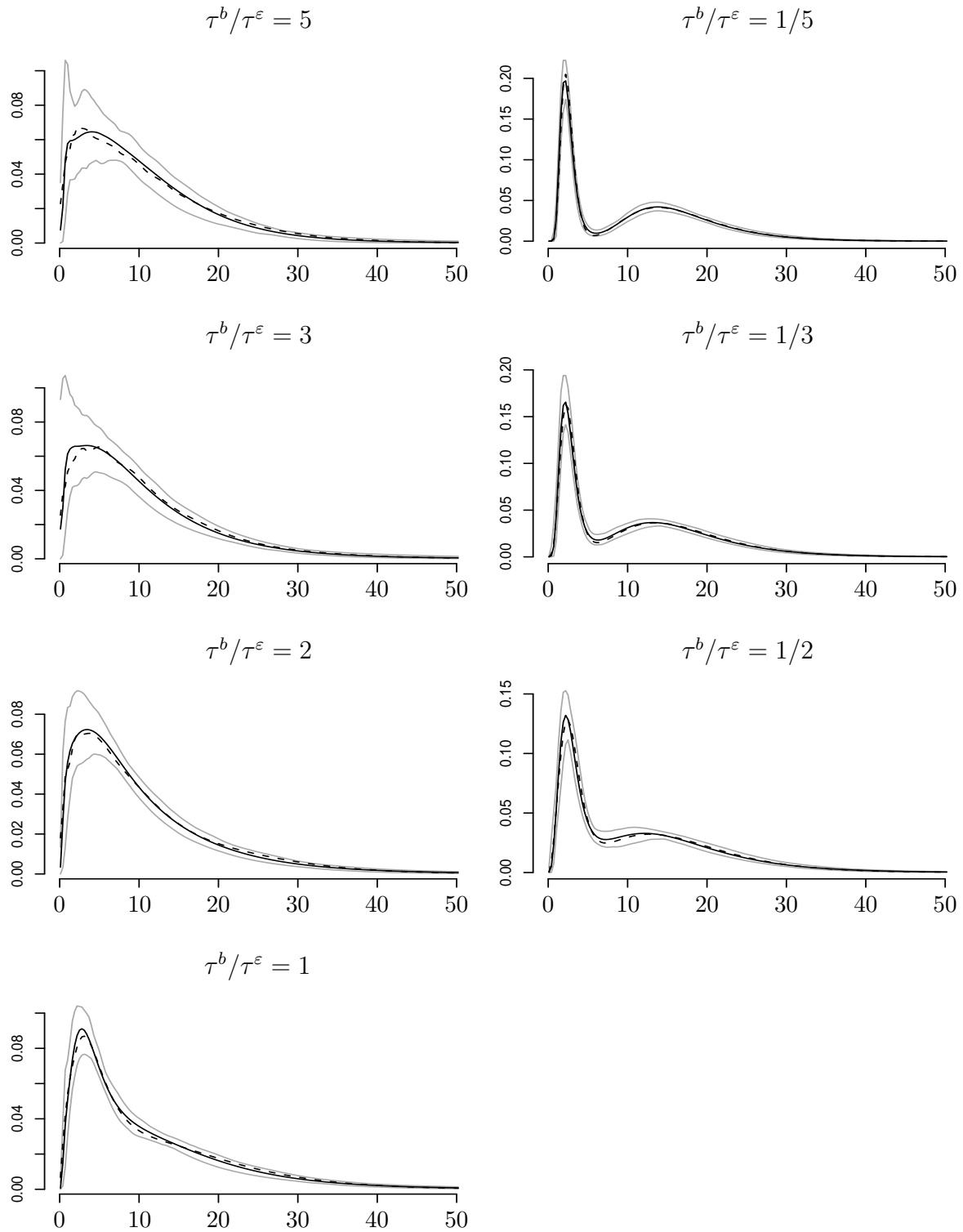


Figure 9: Results for the survivor density of the **event** time, for the combination of covariates $\mathbf{x}_{i,l} = (0.5, 1)'$, scenario I (error \sim normal mixture, random effect \sim extreme value). Solid line: average fitted survivor density, grey lines: 95% pointwise confidence band, dashed line: true survivor density..

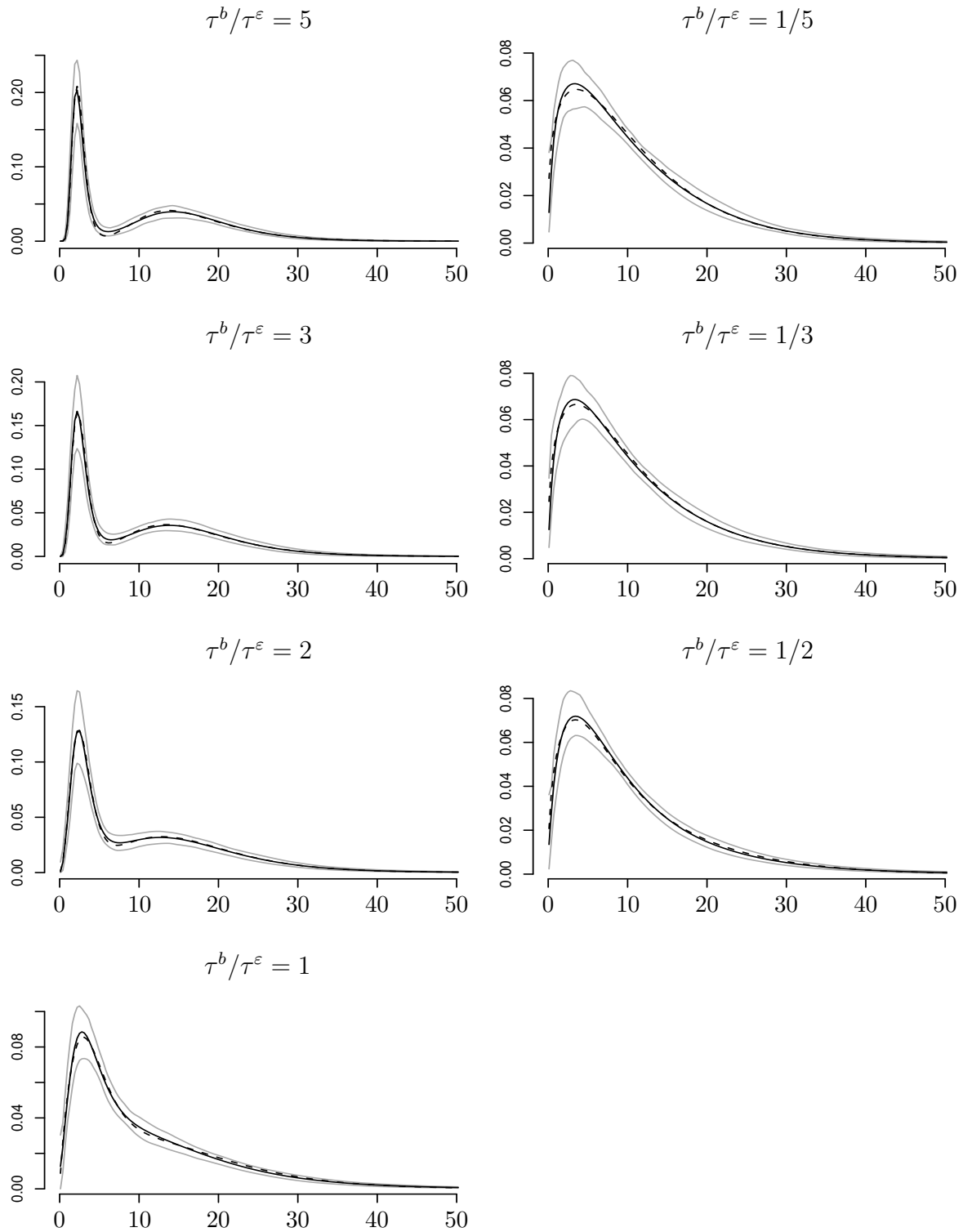


Figure 10: Results for the survivor density of the **event** time, for the combination of covariates $\mathbf{x}_{i,l} = (0.5, 1)'$, scenario II (error \sim extreme value, random effect \sim normal mixture). Solid line: average fitted survivor density, grey lines: 95% pointwise confidence band, dashed line: true survivor density..

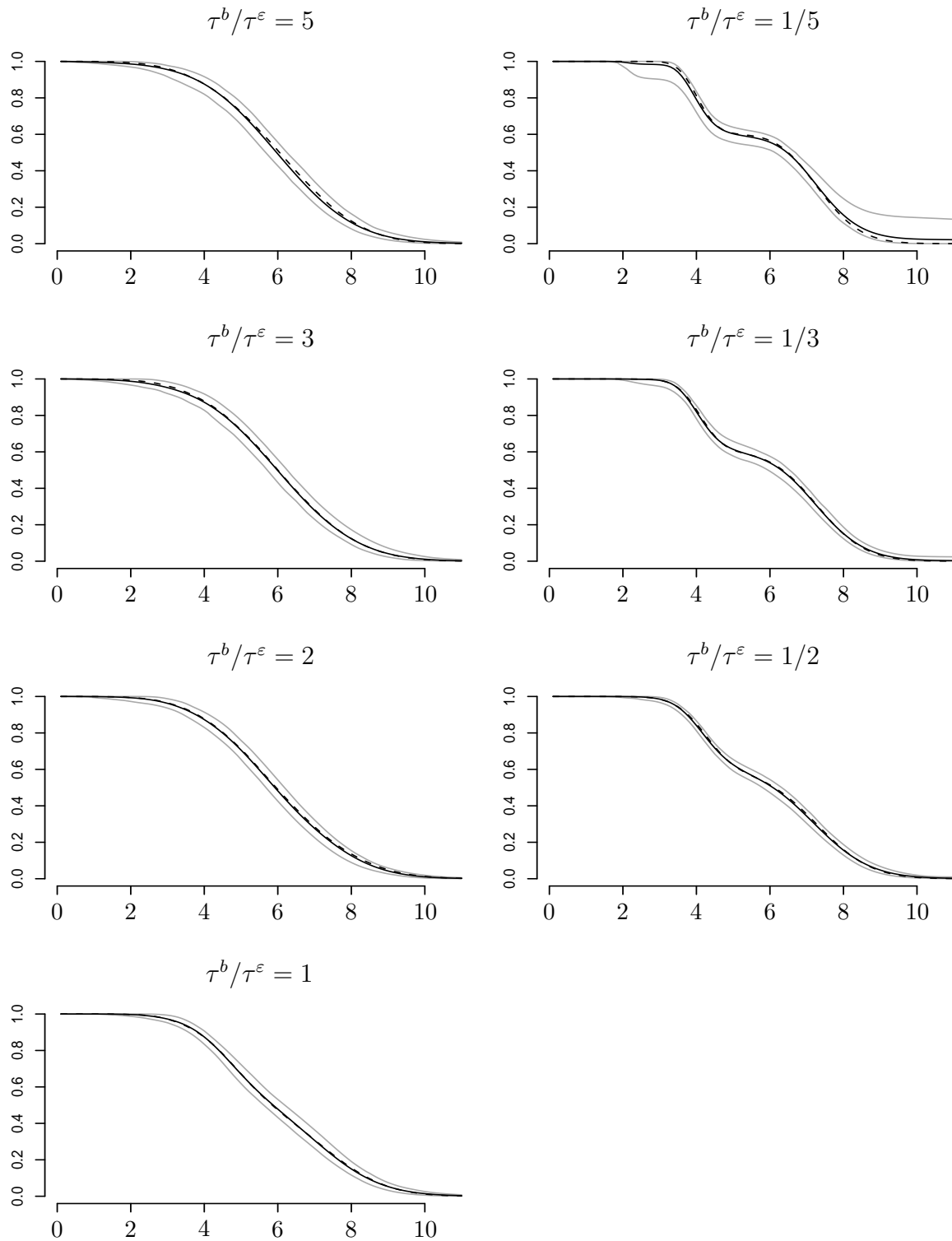


Figure 11: Results for the survivor function of the **onset** time, for the combination of covariates $\mathbf{z}_{i,l} = (0.5, 1)'$, scenario I (error \sim normal mixture, random effect \sim extreme value). Solid line: average fitted survivor function, grey lines: 95% pointwise confidence band, dashed line: true survivor function..

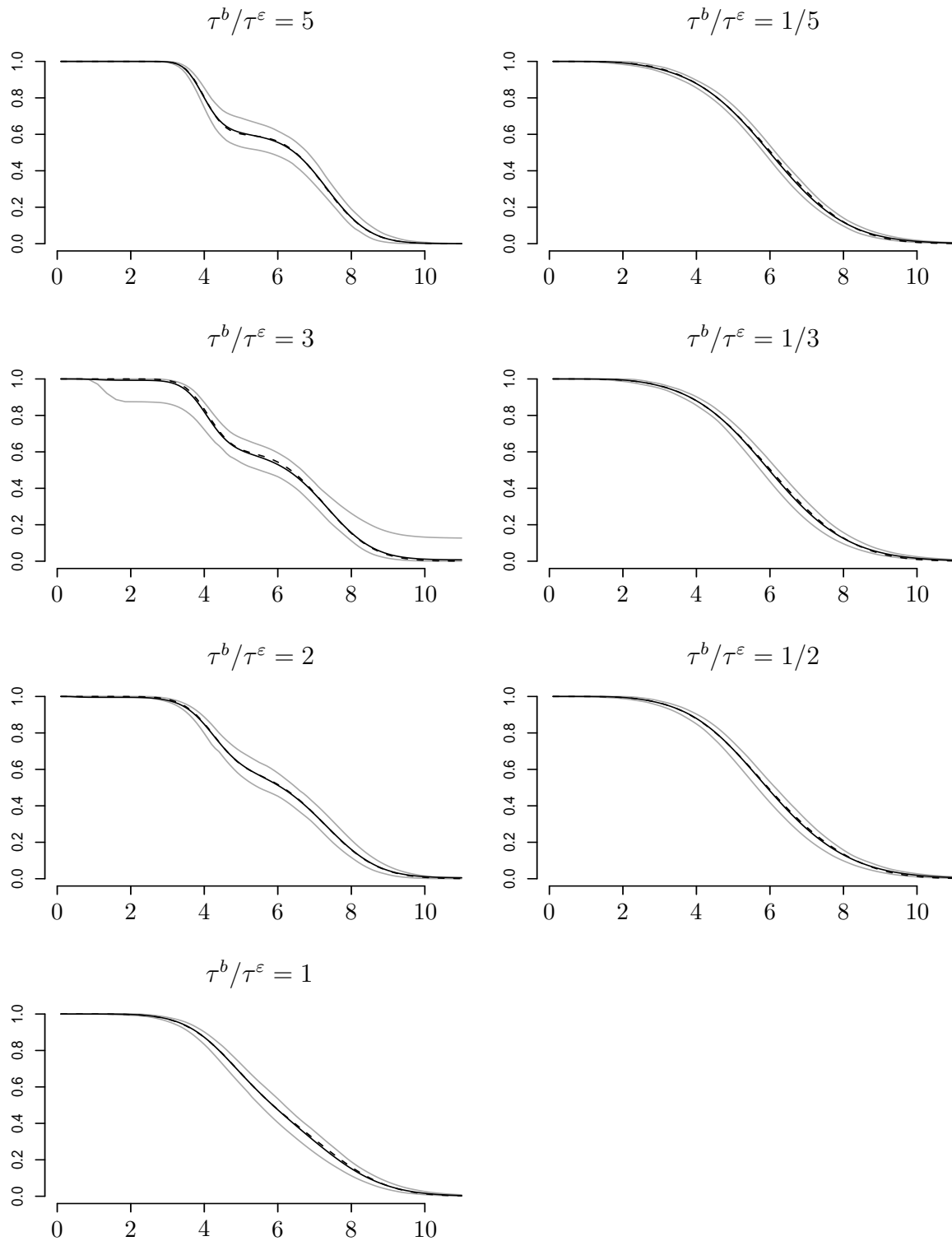


Figure 12: Results for the survivor function of the **onset** time, for the combination of covariates $\mathbf{z}_{i,l} = (0.5, 1)'$, scenario II (error \sim extreme value, random effect \sim normal mixture). Solid line: average fitted survivor function, grey lines: 95% pointwise confidence band, dashed line: true survivor function..

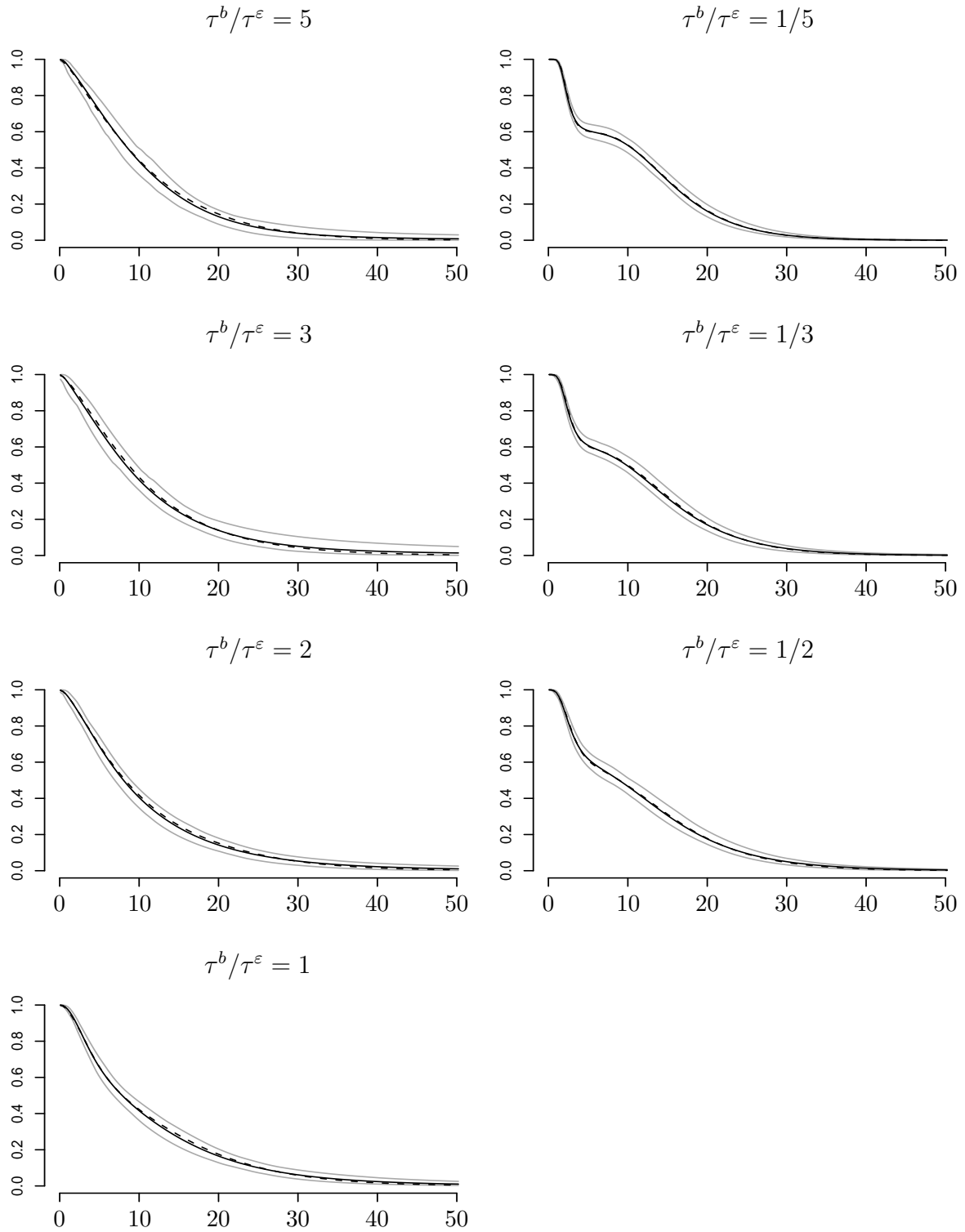


Figure 13: Results for the survivor function of the **event** time, for the combination of covariates $\mathbf{x}_{i,l} = (0.5, 1)'$, scenario I (error \sim normal mixture, random effect \sim extreme value). Solid line: average fitted survivor function, grey lines: 95% pointwise confidence band, dashed line: true survivor function..

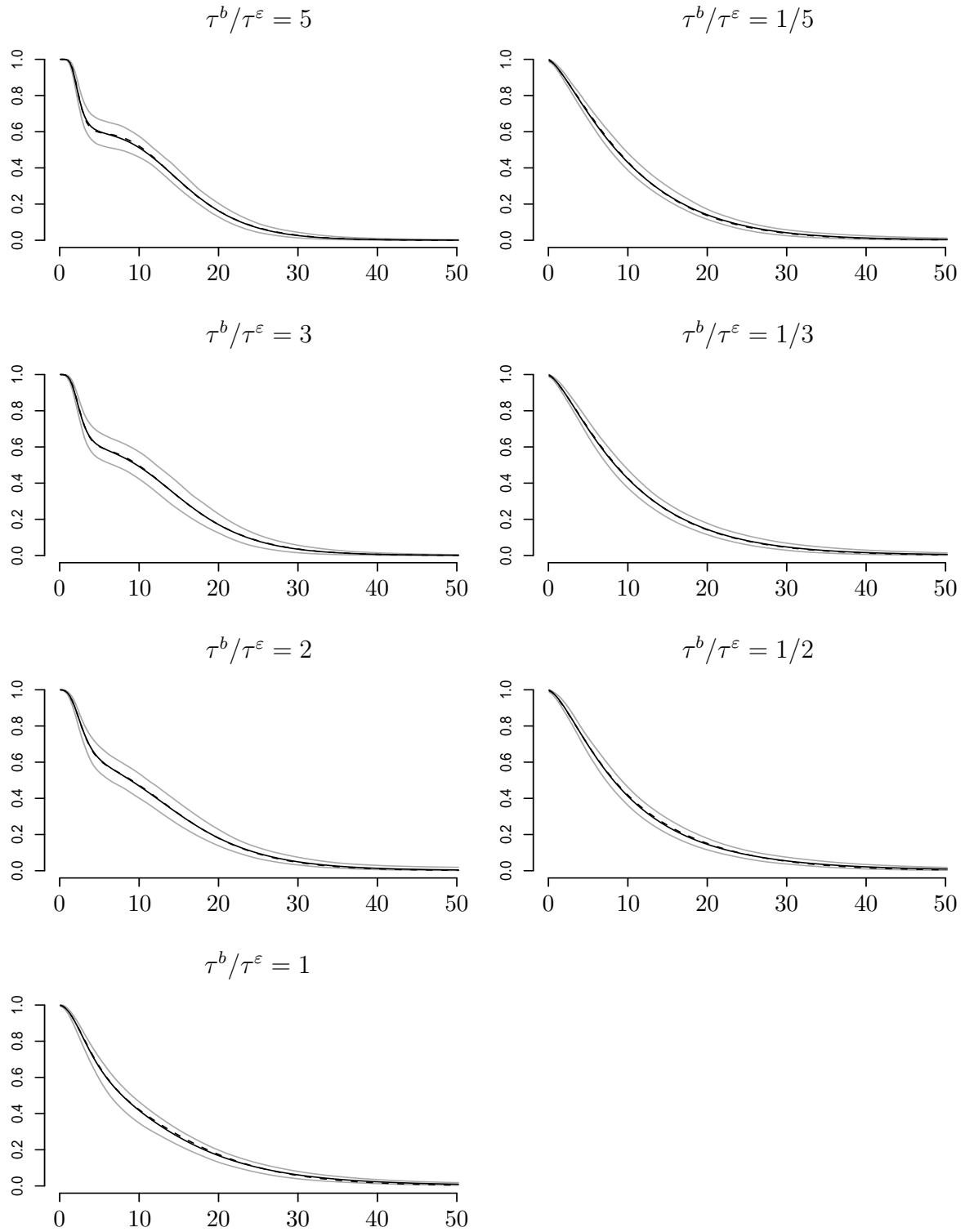


Figure 14: Results for the survivor function of the **event** time, for the combination of covariates $\mathbf{x}_{i,l} = (0.5, 1)'$, scenario II (error \sim extreme value, random effect \sim normal mixture). Solid line: average fitted survivor function, grey lines: 95% pointwise confidence band, dashed line: true survivor function..

6 Analysis of the Signal Tandmobiel[®] data

In this section we describe how the R package `bayesSurv` downloadable from *CRAN* on

<http://www.R-project.org>

can be used to obtain the results presented in the paper. We show in more detail how to perform the **Final Analysis** using **Model I** and **Model D**.

More detailed description of the functions from the `bayesSurv` package and their arguments can be found in the documentation to the package.

6.2 Data

The Signal Tandmobiel[®] data used in the paper are publicly available as a part of the package `bayesSurv`. Anyone can use these data for his/her own research work under the condition that each manuscript where the data are used is first approved by Prof. Emmanuel Lesaffre, Biostatistical Centre, Katholieke Universiteit Leuven, Kapucijnenvoer 35, B-3000 Leuven, Belgium, E-mail: emmanuel.lesaffre@med.kuleuven.be.

The data are loaded using

```
> library(bayesSurv)
> data(tandmobRoos)
```

Note that in the original data set, each row corresponds to 1 child. For the analysis, we need some rearrangement of the data such that each row corresponds to 1 tooth (standard data storage in the context of the linear mixed models). In a sequel, we will assume that the rearranged data are stored in a `data.frame` called `tandmob` whose structure is indicated in Table 5. The columns have the following meaning:

Idnr identification number of a child;

Tooth identification of the tooth. Note that it is assumed that the column `Tooth` has been coded as a categorical variable using the R function `factor`;

Ebeg, **Eend** lower and upper limit of the recorded emergence time (in years) after subtraction of 5 years of age (see Section 6 of the paper) and after adjustment of left-censored observations using the observed eruption stage (see Section 2 of the paper). Value of **Ebeg** equal to `NA` indicates that the emergence is considered as left-censored even after the adjustment for the eruption stage (teeth with the eruption stage P4).

For example, for child 1, the tooth 16 has emerged between 6.7 and 7.7 years of age, the tooth 26 has emerged before the age of 7.7 and the lower limit of the emergence is unknown;

Fbeg, **Fend** lower and upper limit of the recorded caries time (in years) after subtraction of 5 years of age (see Section 6 of the paper). Value of **Fend** equal to `NA` indicates right-censored caries times.

For example, for child 1, all 4 teeth were caries free at the last examination performed at the age of 11.2 years. For child 2, teeth 16, 36, 46 became decayed between 7.4 and 8.4 years of age;

Table 5: Structure of the R data.frame tandmob holding the data.

Idnr	Tooth	Ebeg	Eend	Fbeg	Fend	Girl	Brush	Seal	Plaque	Status
1	16	1.7	2.7	6.2	NA	1	0	1	0	0
1	26	NA	2.7	6.2	NA	1	0	1	0	0
1	36	1.7	2.7	6.2	NA	1	0	1	0	0
1	46	1.7	2.7	6.2	NA	1	0	1	0	0
2	16	NA	2.4	2.4	3.4	0	0	0	0	0
2	26	NA	2.4	3.4	4.3	0	0	0	1	0
2	36	NA	2.4	2.4	3.4	0	0	0	0	0
2	46	NA	2.4	2.4	3.4	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Girl binary variable indicating gender of a child ($0 = boys$, $1 = girls$);

Brush binary variable indicating frequency of brushing ($0 = less than once a day$, $1 = at least once a day$);

Seal binary variable indicating presence of sealants ($0 = no$, $1 = yes$);

Plaque binary variable indicating presence of plaque ($0 = no$, $1 = yes$);

Status binary variable indicating the status of the adjacent primary second molar ($0 = sound$, $1 = decayed or missing due to caries or filled$).

6.3 Model I

6.3.1 Specification of the prior distributions and initial values

For all penalized Gaussian mixtures involved in the model we are using 31 equidistant knots ($K = 15$), centered around $\mu_0 = 0$, where the distance between the 2 knots is equal to 0.3. The basis standard deviation σ is equal to 0.2, i.e. to 1.5 multiple of the distance between the 2 knots. Prior distribution for the PGM-related parameters are as follows

$$\begin{aligned} \alpha &\sim \mathcal{N}(0, 100), & \tau^{-2} &\sim \text{G}(1, 0.005), \\ \lambda &\sim \text{G}(1, 0.005). \end{aligned}$$

In the **Final analysis**, the regression parameter in the emergence part of the model is $\delta = (\delta_1, \dots, \delta_4)'$ and the regression parameter in the caries part of the model is $\beta = (\beta_1, \dots, \beta_8)'$. We are assuming the independent normal priors for the regression coefficients, namely

$$\begin{aligned} \delta_j &\sim \mathcal{N}(0, 100), & j &= 1, \dots, 4, \\ \beta_j &\sim \mathcal{N}(0, 100), & j &= 1, \dots, 8. \end{aligned}$$

Firstly, we specify some parameters of the penalized Gaussian mixtures which determines the distribution of the error term ζ in the emergence part of the model and prior choices for some of the PGM related parameters. Specified choices will be stored in the list `prior.zeta`:

```

> prior.zeta <- list(K=15, c4delta=1.5, order=3,
+   prior.intercept="normal", mean.intercept=0, var.intercept=100,
+   prior.scale="gamma", shape.scale=1, rate.scale=0.005,
+   prior.lambda="gamma", shape.lambda=1, rate.lambda=0.005)

```

Component `K` of the list `prior.zeta` gives the number of knots K on each side of the reference knot, component `c4delta` determines the distance between 2 consecutive knots expressed as a multiple of the basis standard deviation σ^ζ (to be specified later), component `order` gives the order of the penalty s . Further, components `prior.intercept`, `mean.intercept`, `var.intercept` give the prior distribution of the intercept α^ζ , components `prior.scale`, `shape.scale`, `rate.scale` give the prior distribution for the inverse variance τ^{-2} . Finally, components `prior.lambda`, `shape.lambda`, `rate.lambda` determine the prior distribution of the smoothing hyperparameter λ^ζ

Similarly, the PGM for the distribution of the error term ε , stored in the list `prior.eps` can be specified. Here, the same prior assumptions concerning the ζ and ε PGM's are made, so that

```

> prior.eps <- prior.zeta

```

Also the prior assumptions concerning the random effects d and b PGM's are the same as the prior assumptions concerning the ζ . We will store them in the lists `prior.d` and `prior.b`:

```

> prior.d <- prior.zeta
> prior.b <- prior.zeta

```

Note that the components `prior.intercept`, `mean.intercept`, `var.intercept` of the lists `prior.d` and `prior.b` will actually be ignored in a sequel since the random effects intercepts α^d and α^b are fixed to 0.

Secondly, we specify the prior distribution for the regression parameters δ and β and store it in the lists `prior.delta` and `prior.beta`:

```

> prior.delta <- list(mean.prior=rep(0, 4), var.prior=rep(100, 4))
> prior.beta <- list(mean.prior=rep(0, 8), var.prior=rep(100, 8))

```

To start the MCMC sampling, it is useful to give some reasonable initial values for the regression parameters, error intercepts and scales. These can be found, e.g., by fitting an AFT model without random effects, separately for emergence and caries using the standard R function `survreg` from the package `survival` with, e.g., normal error distribution. For the purpose of finding the initial values, doubly-censored caries times can be changed into single-censored times using, e.g., mid-point imputation of the emergence times. With this procedure, we have found the following initial values for the regression coefficients:

$$\begin{aligned} \delta_{init} &= (-0.01, 0.01, 0.02, -0.02), \\ \beta_{init} &= (0.03, 0, 0, -0.06, 0.35, -0.24, 0.07, -0.63). \end{aligned}$$

Also the initial values for the error intercepts α^ζ and α^ε are found from the `survreg` fit:

$$\alpha_{init}^\zeta = 0.40, \quad \alpha_{init}^\varepsilon = 2.16.$$

Finally, initial values for the scale parameters τ^ζ , τ^d , τ^ε , τ^b can be determined by a decomposition of the error scale found by the `survreg` fit. We use the following values:

$$\begin{aligned}\tau_{init}^\zeta &= 0.10, & \tau_{init}^\varepsilon &= 0.55, \\ \tau_{init}^d &= 0.05, & \tau_{init}^b &= 0.05.\end{aligned}$$

For the smoothing hyperparameters we use quite high initial values

$$\begin{aligned}\lambda_{init}^\zeta &= 3\,000, & \lambda_{init}^\varepsilon &= 3\,000, \\ \lambda_{init}^d &= 1\,000, & \lambda_{init}^b &= 1\,000,\end{aligned}$$

which express smoothness of the initial PGM's.

The initial values of above mentioned parameters will be stored in the lists `init.emerg` and `init.caries`. Also the value of the middle knots μ_0^ζ , μ_0^d , μ_0^ε , μ_0^b (`gamma` and `gamma.b` below), and values of the PGM basis standard deviations σ^ζ , σ^d , σ^ε , σ^b (`sigma` and `sigma.b` below) will be stored in these lists:

```
> init.emerg <- list(lambda=3000, intercept=0.40, scale= 0.10,
+   gamma=0, sigma=0.2,
+   beta= c(-0.01, 0.01, 0.02, -0.02),
+   lambda.b=1000, intercept.b=0.00, scale.b=0.05,
+   gamma.b=0.00, sigma.b=0.2)
> init.caries <- list(lambda=3000, intercept=2.16, scale=0.55,
+   gamma=0, sigma=0.2,
+   beta=c(0.03, 0, 0, -0.06, 0.35, -0.24, 0.07, -0.63),
+   lambda.b=1000, intercept.b=0.00, scale.b=0.05,
+   gamma.b=0.00, sigma.b=0.2)
```

The user can also specify the initial values for the PGM weights \mathbf{w} . If it is not done, the initial PGM is automatically determined by the `bayessurvreg3` function such that it corresponds as closely as possible to the normal distribution.

6.3.2 MCMC sampling

The core part of the analysis, MCMC sampling, is performed using the function `bayessurvreg3` in the following way:

```
> library(bayesSurv)
sampleI <- bayessurvreg3(
+   formula=Surv(Ebeg, Eend, type="interval2")~Tooth+Grl+cluster(Idnr),
+   random=~1,
+   formula2=Surv(Fbeg, Fend, type="interval2")~Tooth+Grl+Brush+Seal+
+   Plaque+Status+cluster(Idnr),
+   random2=~1,
+   prior=prior.zeta,
+   prior2=prior.eps,
+   prior.beta=prior.delta,
```

```

+ prior.beta2=prior.beta,
+ prior.b=prior.d,
+ prior.b2=prior.b,
+ nsimul=list(niter=250000, nthin=3, nburn=200000, nwrite=250),
+ store=list(a=TRUE, a2=TRUE, a.b=TRUE, a.b2=TRUE),
+ dir="/home/userAK/modelI",
+ data=tandmob)

```

Sampled chains are then found in the form of ASCII files having an extension `.sim` in the directory called `"/home/userAK/modelI"` and can be further worked out, e.g., using the R package `coda` Plummer et al. (2005).

6.3.3 Posterior summary

For example, data for Table 2 of the paper were obtained using the following commands:

```

> library(coda)
> itersInd <- read.table("/home/userAK/modelI/iteration.sim", header=TRUE)[,1]
> nlines <- length(itersInd)

```

Emergence: Regression

```

> ndelta <- 4
> delta <- matrix(scan("/home/userAK/modelI/beta.sim", skip=1, nlines=nlines),
+ ncol=ndelta, byrow=TRUE)
> colnames(delta) <- scan("/home/userAK/modelI/beta.sim",
+ what="character", nlines=1)
> exp.delta <- mcmc(exp.delta)
> summary(exp.delta)
> HPDinterval(exp.delta)

```

Emergence: Moments of the error and random effects distribution

```

> mixmoment.zeta <- matrix(scan("/home/userAK/modelI/mixmoment.sim",
+ skip=1, nlines=nlines), ncol=3, byrow=TRUE)
> colnames(mixmoment.zeta) <- scan("/home/userAK/modelI/mixmoment.sim",
+ what="character", nlines=1)
> mixmoment.d <- matrix(scan("/home/userAK/modelI/mixmoment_b.sim",
+ skip=1, nlines=nlines), ncol=3, byrow=TRUE)
> colnames(mixmoment.d) <- scan("/home/userAK/modelI/mixmoment_b.sim",
+ what="character", nlines=1)

> intcpt.zeta <- mixmoment.zeta[, "Mean.1"] + mixmoment.d[, "Mean.1"]
> scale.zeta <- sqrt(mixmoment.zeta[, "D.1.1"])
> scale.d <- sqrt(mixmoment.d[, "D.1.1"])
> baseline.emerg <- data.frame(intcpt=intcpt.zeta,

```

```

+ scale.error=scale.zeta, scale.random=scale.d)
> baseline.emerg <- mcmc(baseline.emerg)
> summary(baseline.emerg)
> HPDinterval(baseline.emerg)

```

Caries: Regression

```

> nbeta <- 8
> beta <- matrix(scan("/home/userAK/modelI/beta_2.sim", skip=1, nlines=nlines),
+ ncol=nbeta, byrow=TRUE)
> colnames(beta) <- scan("/home/userAK/modelI/beta_2.sim",
+ what="character", nlines=1)
> exp.beta <- mcmc(exp.beta)
> summary(exp.beta)
> HPDinterval(exp.beta)

```

Caries: Moments of the error and random effects distribution

```

> mixmoment.eps <- matrix(scan("/home/userAK/modelI/mixmoment_2.sim",
+ skip=1, nlines=nlines), ncol=3, byrow=TRUE)
> colnames(mixmoment.eps) <- scan("/home/userAK/modelI/mixmoment_2.sim",
+ what="character", nlines=1)
> mixmoment.b <- matrix(scan("/home/userAK/modelI/mixmoment_b2.sim",
+ skip=1, nlines=nlines), ncol=3, byrow=TRUE)
> colnames(mixmoment.b) <- scan("/home/userAK/modelI/mixmoment_b2.sim",
+ what="character", nlines=1)

> intcpt.eps <- mixmoment.eps[, "Mean.1"] + mixmoment.b[, "Mean.1"]
> scale.eps <- sqrt(mixmoment.eps[, "D.1.1"])
> scale.b <- sqrt(mixmoment.b[, "D.1.1"])
> baseline.caries <- data.frame(intcpt=intcpt.eps,
+ scale.error=scale.eps, scale.random=scale.b)
> baseline.caries <- mcmc(baseline.caries)
> summary(baseline.caries)
> HPDinterval(baseline.caries)

```

6.3.4 Posterior predictive hazard and survival function

To compute the predictive hazard and survival functions as shown in Figure 2 of the paper, we have to specify the combinations of covariates for which the hazard and survival functions would be computed (columns `Ebeg`, `Eend`, `Fbeg`, `Fend` can be filled arbitrarily):

```

> ncov <- 4
> tandmob.pred <- data.frame(Idnr=1:(4*ncov),
+ Tooth=factor(c(rep(16, ncov), rep(26, ncov), rep(36, ncov), rep(46, ncov)))),

```

```

+ Ebeg=rep(1, 4*ncov),
+ Eend=rep(1, 4*ncov),
+ Fbeg=rep(1, 4*ncov),
+ Fend=rep(1, 4*ncov),
+ Girl=rep(0, 4*ncov),
+ Brush=rep(c(1, 1, 0, 0), 4),
+ Seal=rep(c(1, 1, 0, 0), 4),
+ Plaque=rep(c(0, 0, 1, 1), 4),
+ Status=rep(c(0, 1, 0, 1), 4)

```

Computation of the values of predictive survival and hazard functions for the *caries* part of the model on the grid of values 0.10, 0.15, ..., 1.45, 1.5, 1.65, 1.80, ..., 5.85, 6.00 (years after emergence) is performed using the following code:

```

> predI <- predictive2(
+   Surv(Fbeg, Fend, type="interval2")~Tooth+Girl+Brush+Seal+
+                                     Plaque+Status+cluster(Idnr),
+   random~1,
+   grid=c(seq(0.1, 1.5, by=0.05), seq(1.65, 6, by=0.15)),
+   Gspline=list(dim=1, K=15),
+   quantile=c(0.025, 0.975), only.aver=FALSE,
+   dir="/home/userAK/modelI",
+   predict=list(density=FALSE, Surv=TRUE, hazard=TRUE, cum.hazard=FALSE),
+   extens="_2", extens.random="_b2", version=3,
+   data=tandmob.pred)

```

6.3.5 Estimates of the error and random effects densities

It is possible to obtain the estimates of the PGM's (densities of the error terms and random effects) as pointwise posterior means, see first 4 panels of Figure 15. Their values on a chosen grid can be computed using the following commands.

Emergence: Error

```

> grid.zeta <- seq(0.35, 0.55, length=100)
> gI.zeta <- bayesGspline(dir="/home/userAK/modelI", grid1=grid.zeta,
+   skip=0, by=1, nwrite=5000,
+   extens="", extens.adjust="_b",
+   only.aver=TRUE, version=30)

```

Caries: Error

```

> grid.eps <- seq(-1, 4, length=100)
> gI.eps <- bayesGspline(dir="/home/userAK/modelI", grid1=grid.eps,
+   skip=0, by=1, nwrite=5000,
+   extens="_2", extens.adjust="_b2",
+   only.aver=TRUE, version=30)

```

Emergence: Random effects

```
> grid.d <- seq(-0.6, 0.6, length=100)
> gI.d <- bayesGspline(dir="/home/userAK/modelI", grid1=grid.d,
+ skip=0, by=1, nwrite=5000,
+ extens="_b", extens.adjust="_b",
+ only.aver=TRUE, version=31)
```

Caries: Random effects

```
> grid.b <- seq(-2, 1.2, length=100)
> gI.d <- bayesGspline(dir="/home/userAK/modelI", grid1=grid.b,
+ skip=0, by=1, nwrite=5000,
+ extens="_b2", extens.adjust="_b2",
+ only.aver=TRUE, version=31)
```

6.4 Model I

6.4.1 Specification of the prior distributions and initial values

Specification of the prior distributions and initial values in **Model D** is similar like in **Model I**. We describe only differences.

1. The lists `prior.d` and `prior.b` are not specified.
2. Components `lambda.b`, `intercept.b`, `scale.b`, `gamma.b`, `sigma.b` of the lists `init.emerg` and `init.caries` do not have to be specified.
3. A new list, called `priorinit.Nb` giving prior specification and initial values for the bivariate normal distribution of the random effects is specified:

```
priorinit.Nb <- list(init.D=c(0.05^2, 0, 0.05^2),
+ df.Di=1.5,
+ scale.Di=100*c(1, 0, 1))
```

This means that a priori, the covariance matrix \mathbb{D} of random effects

$$\mathbb{D}^{-1} \sim \text{Wishart}(df, \mathbb{S}),$$

$$\text{with } df = 1.5, \quad \mathbb{S} = \begin{pmatrix} 100 & 0 \\ 0 & 100 \end{pmatrix}.$$

The initial value of \mathbb{D} is

$$\mathbb{D}_{init} = \begin{pmatrix} 0.05^2 & 0 \\ 0 & 0.05^2 \end{pmatrix}.$$

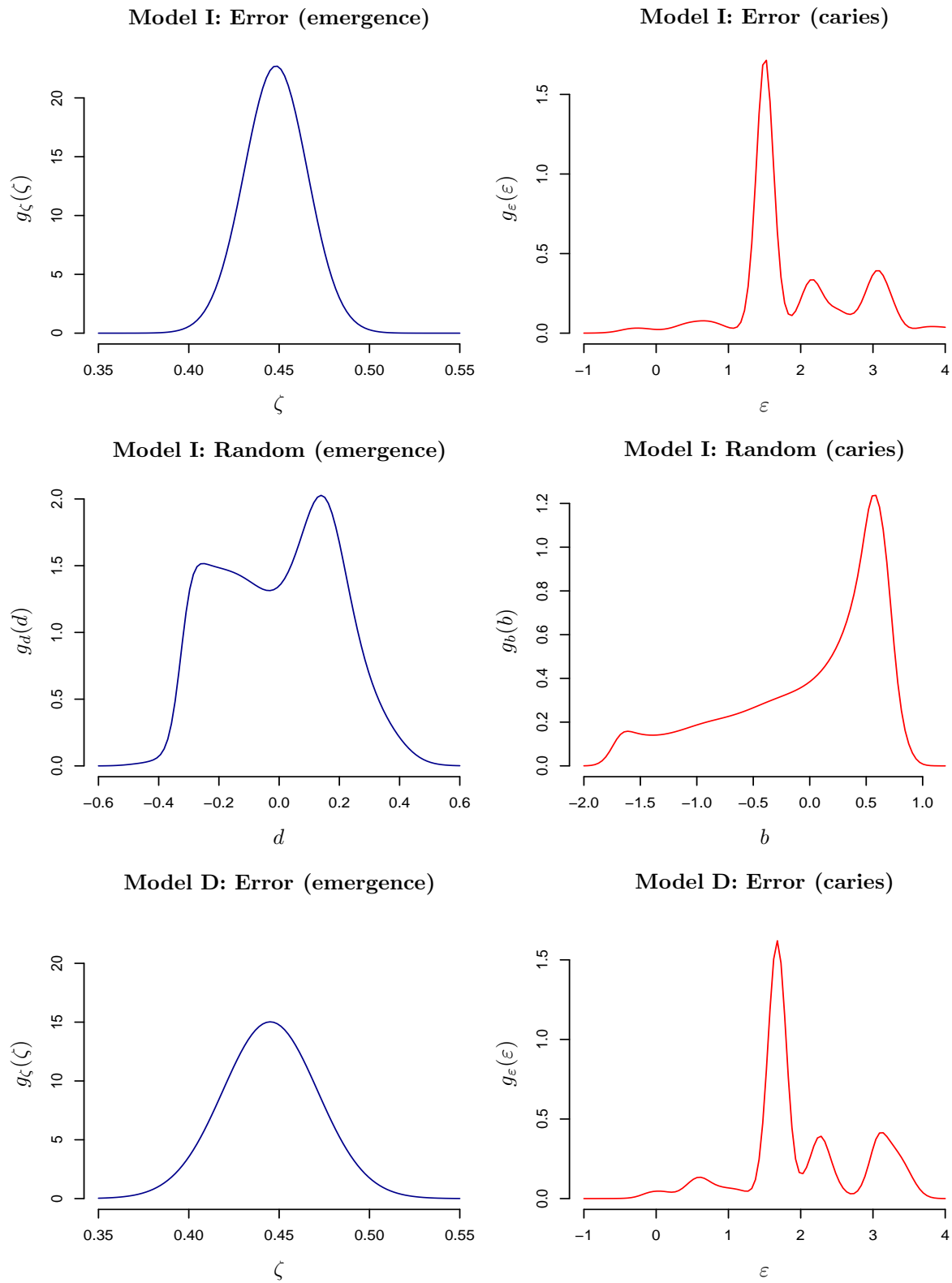


Figure 15: Signal Tandmobiél[®] data, **Final Analysis**. Estimates of the densities of the error terms and random effects in **Model I** and **Model D**.

6.4.2 MCMC sampling

The core part of the analysis, MCMC sampling, is performed using the function `bayessurvreg3` in the following way:

```
> library(bayesSurv)
sampleD <- bayessurvreg3(
+   formula=Surv(Ebeg, Eend, type="interval2")~Tooth+Girl+cluster(Idnr),
+   random=~1,
+   formula2=Surv(Fbeg, Fend, type="interval2")~Tooth+Girl+Brush+Seal+
+                                     Plaque+Status+cluster(Idnr),
+   random2=~1,
+   prior=prior.zeta,
+   prior2=prior.eps,
+   prior.beta=prior.delta,
+   prior.beta2=prior.beta,
+   priorinit.Nb=priorinit.Nb,
+   nsimul=list(niter=250000, nthin=3, nburn=200000, nwrite=250),
+   store=list(a=TRUE, a2=TRUE, a.b=TRUE, a.b2=TRUE),
+   dir="/home/userAK/modelD",
+   data=tandmob)
```

6.4.3 Posterior summary

Posterior summary for the regression parameters is obtained in the same way as in **Model I**. Summary for the quantities derived from the covariance matrix \mathbb{D} can be obtained in the following way.

```
> D.b <- matrix(scan("/home/userAK/modelD/D.sim", skip=1, nlines=nlines), ncol=4, byrow=TRUE)
> colnames(D.b) <- scan("/home/userAK/modelD/D.sim", what="character", nlines=1)
> moment.b <- data.frame(detD=D.b[, "det"],
+   SD.emerg=sqrt(D.b[, "D.1.1"]),
+   SD.caries=sqrt(D.b[, "D.2.2"]),
+   Rho=D.b[, "D.2.1"]/sqrt(D.b[, "D.1.1"]*D.b[, "D.2.2"]))
> moment.b <- mcmc(moment.b)
> summary(moment.b)
> HPDinterval(moment.b)
```

6.4.4 Posterior predictive hazard and survival function

Predictive hazard and survival functions based on **Model D** are computed as follows.

```
> predD <- predictive2(
+   Surv(Fbeg, Fend, type="interval2")~Tooth+Girl+Brush+Seal+
+                                     Plaque+Status+cluster(Idnr),
```

```

+ random~1,
+ grid=c(seq(0.1, 1.5, by=0.05), seq(1.65, 6, by=0.15)),
+ Gspline=list(dim=1, K=15),
+ quantile=c(0.025, 0.975), only.aver=FALSE,
+ dir="/home/userAK/modelI",
+ predict=list(density=FALSE, Surv=TRUE, hazard=TRUE, cum.hazard=FALSE),
+ extens="_2", version=32,
+ data=tandmob.pred)

```

6.4.5 Estimates of the error and random effects densities

Estimates of the PGM's specifying the error distributions (last 2 panels of Figure 15) are computed as indicated below.

Emergence: Error

```

> grid.zeta <- seq(0.35, 0.55, length=100)
> gD.zeta <- bayesGspline(dir="/home/userAK/modelD", grid1=grid.zeta,
+ skip=0, by=1, nwrite=5000,
+ extens="",
+ only.aver=TRUE, version=32)

```

Caries: Error

```

> grid.eps <- seq(-1, 4, length=100)
> gD.eps <- bayesGspline(dir="/home/userAK/modelD", grid1=grid.eps,
+ skip=0, by=1, nwrite=5000,
+ extens="_2",
+ only.aver=TRUE, version=32)

```

References

- GILKS, W. R. and WILD, P. (1992). Adaptive rejection sampling for Gibbs sampling. *Applied Statistics*, **41**, 337–348.
- NEAL, R. M. (2003). Slice sampling (with Discussion). *The Annals of Statistics*, **31**, 705–767.
- PLUMMER, M., BEST, N., COWLES, K., and VINES, K. (2005). *coda: Output analysis and diagnostics for MCMC*. URL <http://www-fis.iarc.fr/coda/>. R package version 0.9-5.
- R DEVELOPMENT CORE TEAM (2006). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org>. ISBN 3-900051-07-0.