



HAL
open science

Tutorial in Joint Modeling and Prediction: A Statistical Software for Correlated Longitudinal Outcomes, Recurrent Events and a Terminal Event

Agnieszka Król, Audrey Mauguen, Yassin Mazroui, Alexandre Laurent, Stefan Michiels, Virginie Rondeau

► To cite this version:

Agnieszka Król, Audrey Mauguen, Yassin Mazroui, Alexandre Laurent, Stefan Michiels, et al.. Tutorial in Joint Modeling and Prediction: A Statistical Software for Correlated Longitudinal Outcomes, Recurrent Events and a Terminal Event. *Journal of Statistical Software*, 2017, 81 (3), 10.18637/jss.v081.i03 . hal-03940197

HAL Id: hal-03940197

<https://hal.science/hal-03940197v1>

Submitted on 16 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Tutorial in Joint Modeling and Prediction: A Statistical Software for Correlated Longitudinal Outcomes, Recurrent Events and a Terminal Event

Agnieszka Król
ISPED, INSERM U897
Université de Bordeaux

Audrey Mauguen
ISPED, INSERM U897
Université de Bordeaux

Yassin Mazroui
Université Pierre et Marie Curie
& INSERM, UMR S 1136, Paris

Alexandre Laurent
ISPED, INSERM U897
Université de Bordeaux

Stefan Michiels
INSERM U1018, CESP
& Gustave Roussy, Villejuif

Virginie Rondeau
ISPED, INSERM U897
Université de Bordeaux

Abstract

Extensions in the field of joint modeling of correlated data and dynamic predictions improve the development of prognosis research. The R package **frailtypack** provides estimations of various joint models for longitudinal data and survival events. In particular, it fits models for recurrent events and a terminal event (**frailtyPenal**), models for two survival outcomes for clustered data (**frailtyPenal**), models for two types of recurrent events and a terminal event (**multivPenal**), models for a longitudinal biomarker and a terminal event (**longiPenal**) and models for a longitudinal biomarker, recurrent events and a terminal event (**trivPenal**). The estimators are obtained using a standard and penalized maximum likelihood approach, each model function allows to evaluate goodness-of-fit analyses and provides plots of baseline hazard functions. Finally, the package provides individual dynamic predictions of the terminal event and evaluation of predictive accuracy. This paper presents the theoretical models with estimation techniques, applies the methods for predictions and illustrates **frailtypack** functions details with examples.

Keywords: dynamic prediction, frailty, joint model, longitudinal data, predictive accuracy, R, survival analysis.

1. Introduction

Joint models

Recent technologies allow the registration of greater and greater amount of data. In medical research different kinds of patient information are gathered over time together with clinical outcome data such as overall survival (OS). Joint models enable the analysis of correlated data of different types such as individual repeated data and clustered data together with OS. The repeated data may be recurrent events (e.g., relapses of a tumor, hospitalizations, appearance of new lesions) or a longitudinal outcome called biomarker (e.g., tumor size, prostate-specific antigen or CD4 cell counts). The correlated data that are not analyzed jointly with OS, are subjugated to a bias coming from ignoring the terminal event which is the competing event for the occurrence of repeated outcomes (not only does it preclude the outcomes from being observed but it also prevents them from occurring). On the other hand, the standard survival analysis for OS may lead to biased estimations, if the repeated data is considered as time-varying covariates or if it is completely ignored in the analysis.

For these correlated data one can consider joint models, e.g., a joint model for a longitudinal biomarker and a terminal event, which received most of the attention in the literature. This joint model estimates simultaneously the longitudinal and survival processes using the relationship via a latent structure of random effects (Wulfsohn and Tsiatis 1997). Extensions of these include, among others, models for multiple longitudinal outcomes (Hatfield, Boye, Hackshaw, and Carlin 2012), multiple failure times (Elashoff, Li, and Li 2008) and both (Chi and Ibrahim 2006). A review of the joint modeling of longitudinal and survival data was already given elsewhere (McCrink, Marshall, and Cairns 2013; Lawrence Gould *et al.* 2015; Asar, Ritchie, Kalra, and Diggle 2015).

Another option for the analysis of correlated data are joint models for recurrent events and a terminal event (Liu, Wolfe, and Huang 2004; Rondeau, Mathoulin-Pélissier, Jacqmin-Gadda, Brouste, and Soubeyran 2007). These models are usually called joint frailty models as the processes are linked via a random effect that represents the frailty of a subject (patient) to experience an event. They account for the dependence between two survival processes quantified by the frailty term, contrary to the alternative marginal approach (Li and Lagakos 1997), which does not model the dependence. Some extensions to joint frailty models include incorporation of a nonparametric covariate function (Yu and Liu 2011), inclusion of two frailty terms for the identification of the origin of the dependence between the processes (Mazroui, Mathoulin-Pélissier, Soubeyran, and Rondeau 2012), consideration of the disease-specific mortality process (Belot, Rondeau, Remontet, Roch, and CENSUR Working Survival Group 2014) and accommodation of time-varying coefficients (Yu, Liu, Bravata, and Williams 2014; Mazroui, Mauguén, Macgrogan, Mathoulin-Pélissier, Brouste, and Rondeau 2016). Finally, Mazroui, Mathoulin-Pélissier, Macgrogan, Brouste, and Rondeau (2013) proposed a model with two types of recurrent events following the approach of Zhao, Liu, Liu, and Xu (2012). A review of joint frailty models in the Bayesian context was given by Sinha, Maiti, Ibrahim, and Ouyang (2008).

Joint models for recurrent events and longitudinal data have received the least attention among the joint models so far. However, a marginal model based on the generalized linear mixed model was proposed by Efendi, Molenberghs, Njagi, and Dendale (2013). This model allows several longitudinal and time-to-event outcomes and includes two sets of random ef-

fects, one for the correlation within a process and between the processes, and the second to accommodate for overdispersion.

Finally, in some applications all the types of data, i.e., a longitudinal biomarker, recurrent events and a terminal event can be studied so that all of them are correlated to each other (in the following we call such models trivariate models). [Liu, Huang, and O'Quigley \(2008\)](#) proposed a trivariate model for medical cost data. The longitudinal outcomes (amount of medical costs) were reported at informative times of recurrent events (hospitalizations) and were related to a terminal event (death). The dependence via random effects was introduced so that the process of longitudinal measurements and the process of recurrent events were related to the process of the terminal event. A relationship between longitudinal outcomes and recurrent events was not considered. This relationship was incorporated into a trivariate model proposed by [Liu and Huang \(2009\)](#) for an application of an HIV study. In this parametric approach the focus was on the analysis of the associations present in the model and the effect of the repeated measures and recurrent events on the terminal event. [Kim, Zeng, Chambless, and Li \(2012\)](#) analyzed the trivariate data using the transformation functions for the cumulative intensities for recurrent and terminal events. Finally, [Król *et al.* \(2016\)](#) proposed a trivariate model in which all the processes were related to each other via a latent structure of random effects and applied the model to the context of a cancer evolution and OS.

Prediction in joint models

In the framework of joint models that consider a terminal event, one can be interested in predictions of the event derived from the model. As joint models include information from repeated outcomes, these predictions are dynamic and they change with the update of the observations. Dynamic predictive tools were proposed for joint models for longitudinal data and a terminal event ([Proust-Lima and Taylor 2009](#); [Rizopoulos 2011](#)), joint models for recurrent events and a terminal event ([Mauguen, Rachet, Mathoulin-Pélissier, MacGrogan, Laurent, and Rondeau 2013](#)) and trivariate models ([Król *et al.* 2016](#)).

For the evaluation of the predictive accuracy of a joint model few methods were proposed due to the complexity of the models in which the survival data are usually right-censored. The standard methods for the assessment of the predictive abilities derived from the survival analysis and adapted to the joint models context are the Brier score ([Proust-Lima and Taylor 2009](#); [Mauguen *et al.* 2013](#); [Król *et al.* 2016](#)) and the receiver operating characteristic (ROC) methodology ([Rizopoulos 2011](#); [Blanche, Proust-Lima, Loubère, Berr, Dartigues, and Jacqmin-Gadda 2015](#)). Finally, the expected prognostic cross-entropy (EPOCE), a measure based on information theory, provides a useful tool for the evaluation of predictive abilities of a model ([Commenges, Liquet, and Proust-Lima 2012](#); [Proust-Lima, Séne, Taylor, and Jacqmin-Gadda 2014](#)).

Software for joint models

Together with the theoretical development of the joint models, an increase of appropriate software is observed, mostly for standard frameworks. Among the available R ([R Core Team 2017](#)) packages, the joint analysis of a single longitudinal outcome and a terminal event can be performed using **JM** ([Rizopoulos 2010, 2017](#)) based on the likelihood maximization approach using an expectation-maximization (EM) algorithm, **JMbayes** ([Rizopoulos 2016b,a](#)) built in

the Bayesian context, and **joineR** (Philipson, Sousa, Diggle, Williamson, Kolamunnage-Dona, and Henderson 2017), a frequentist approach that allows a flexible formulation of the dependence structure. For the approach based on joint latent class models, i.e., joint models that consider homogeneous latent subgroups of individuals sharing the same biomarker trajectory and risk of the terminal event, the extensive package **lcmm** (Proust-Lima, Philipps, and Liquet 2017b; Proust-Lima, Philipps, Diakite, and Liquet 2017a) provides estimations based on the frequentist approach. Apart from the R packages, a **stjm** package (Crowther 2013) in Stata (StataCorp. 2015) uses maximum likelihood estimation and provides flexible methods for modeling the longitudinal outcome based on polynomials or splines. Another possibility for the analysis of the joint models for a longitudinal outcome and a terminal event is the SAS (SAS Institute Inc. 2013) macro **JMFit** (Zhang, Chen, Ibrahim, Boye, and Shen 2016). Joint models using nonlinear mixed effects models can be estimated using the **MONOLIX** software (Mbogning, Bleakley, and Lavielle 2015). The packages **JM**, **JMbayes** and **lcmm** also provide predictive tools and predictive accuracy measures: the EPOCE estimator in **lcmm**, ROC and AUC (area under the curve) measures and prediction error in **JM** and **JMbayes**.

For joint models for correlated events (recurrent event or clustered event data) and a terminal event the available statistical software is limited. Among the available R packages, **joint.Cox** (Emura 2017) provides the estimations using penalized likelihood under the copula-based joint Cox models for time to clustered events (progression) and a terminal event. Trivariate joint models proposed by Liu and Huang (2009) were performed in SAS using the procedure NLMIXED.

The R package **frailtypack** (Rondeau *et al.* 2017) fits several types of joint models. Originally developed for the shared frailty models for correlated outcomes it has been extended into the direction of joint models. These extensions include the joint model for one or two types of recurrent events and a terminal event, the joint model for a longitudinal biomarker and a terminal event and the trivariate model for a longitudinal biomarker, recurrent events and a terminal event. Moreover, **frailtypack** includes prediction tools, such as concordance measures adapted to shared frailty models and predicted probability of events for the joint models. The package is available from the Comprehensive R Archive Network (CRAN) at <https://CRAN.R-project.org/package=frailtypack>.

Characteristics of a previous version of the package, such as estimation of several shared and standard joint frailty models using penalized likelihood, were already described elsewhere (Rondeau, Mazroui, and Gonzalez 2012). This work focuses on an overview and the developments of joint models included in the package (models for recurrent/clustered events and a terminal event, models for multivariate recurrent events and a terminal event, models for longitudinal data and a terminal event and models for longitudinal data, recurrent events and a terminal event) and the prediction tools accompanied by predictive accuracy measures. Finally, several options available for the models (e.g., two correlated frailties in the model for recurrent events and a terminal event, left-censoring for the longitudinal data) will be presented with the appropriate examples. A practical guide to different types of models included in the package along with available options is presented in a schematic table in Appendix A.

This article firstly presents joint models with details on estimation methods (Section 2) and predictions of an event in the framework of these joint models (Section 3). In Section 4, the **frailtypack** package features are detailed. Section 5 contains some examples on real datasets and finally, Section 6 concludes the paper.

2. Models for correlated outcomes

2.1. Bivariate joint frailty models for two types of survival outcomes

Joint model for recurrent events and terminal event

We denote by $i = 1, \dots, N$ the subject and by $j = 1, \dots, n_i$ the rank of the recurrent event. For each subject i we observe the time of the terminal event $T_i = \min(T_i^*, C_i)$ with T_i^* the true terminal event time and C_i the censoring time. We denote the observed recurrent times $T_{ij} = \min(T_{ij}^*, C_i, T_i^*)$ with T_{ij}^* the real time of the recurrent event. Thus, for each rank j we can summarize the information with a triple $\{T_{ij}, \delta_{ij}, \delta_i\}$, where $\delta_{ij} = I_{\{T_{ij}=T_{ij}^*\}}$ and $\delta_i = I_{\{T_i=T_i^*\}}$. Let $r_0(\cdot)$ and $\lambda_0(\cdot)$ be baseline hazard functions related to the recurrent and terminal events, respectively. Let \mathbf{X}_{Rij} and \mathbf{X}_{Ti} be two vectors of covariates associated with the risk of the related events. Let β_R and β_T be constant effects of the covariates whereas $\beta_R(t)$ and $\beta_T(t)$ denote a time-dependent effect. Finally, a frailty u_i is a random effect that links the two risk functions and follows a given distribution \mathcal{D} and α is a parameter allowing more flexibility. The hazard functions are defined by (Liu *et al.* 2004; Rondeau *et al.* 2007):

$$\begin{cases} r_{ij}(t|u_i) = u_i r_0(t) \exp(\mathbf{X}_{Rij}^\top \beta_R) = u_i r_{ij}(t) & \text{(recurrent event)} \\ \lambda_i(t|u_i) = u_i^\alpha \lambda_0(t) \exp(\mathbf{X}_{Ti}^\top \beta_T) = u_i^\alpha \lambda_i(t) & \text{(terminal event)} \end{cases} \quad (1)$$

where the frailty terms u_i are *iid* (independent and identically distributed) and follow either the Gamma distribution with unit mean and variance θ ($\theta > 0$), i.e., $u_i \sim \mathcal{G}(\frac{1}{\theta}, \frac{1}{\theta})$, or the log-normal distribution, i.e., $u_i = \exp(v_i) \sim \ln\mathcal{N}(0, \sigma_v^2)$. The parameter α determines direction of the association (if found significant) between the processes.

For a given subject we can summarize all the information with $\Theta_i = \{\mathbf{T}_i^{(1)}, T_i, \boldsymbol{\delta}_i^{(1)}, \delta_i\}$, where $\mathbf{T}_i^{(1)} = \{T_{ij}, j = 1, \dots, n_i\}$ and $\boldsymbol{\delta}_i^{(1)} = \{\delta_{ij}, j = 1, \dots, n_i\}$. Finally, we are interested to estimate $\boldsymbol{\xi} = \{r_0(\cdot), \lambda_0(\cdot), \beta_R, \beta_T, \theta, \alpha\}$.

In some cases, e.g., long follow-up, effects of certain prognostic factors can be varying in time. For this reason a joint model with time-dependent coefficients was proposed by Mazroui *et al.* (2016). The coefficients $\beta_R(t)$ and $\beta_T(t)$ are modeled with regression B -splines of order q and m interior knots. The general form of an estimated time-dependent coefficient $\widehat{\beta}(t)$ is

$$\widehat{\beta}(t) = \sum_{j=-q+1}^m \widehat{\zeta}_j B_{j,q}(t),$$

where $B_{j,q}(t)$ is the basis of B -splines calculated using a recurring expression (De Boor 2001). Therefore, for every time-varying coefficient, $m + q$ parameters ζ_j , $j = -q + 1, \dots, m$ need to be estimated. It has been shown that quadratic B -splines, i.e., $q = 3$, with a small number of interior knots ($m \leq 5$) ensure stable estimation (Mazroui *et al.* 2016). The pointwise confidence intervals for $\beta(t)$ can be obtained using the Hessian matrix.

The application of time-varying coefficients is helpful if there is a need to verify the proportional hazard (PH) assumption. Using a likelihood ratio (LR) test, the time-dependency of a covariate can be examined. A model with the time-dependent effect and a model with the constant effect are compared and the test hypotheses depend on whether the covariate is related to one of the events or both. If the time-dependency is tested for both events

the null hypothesis is $H_0 : \beta_R(t) = \beta_R, \beta_T(t) = \beta_T$ and the alternative hypothesis is $H_1 : \beta_R(t) \neq \beta_R, \beta_T(t) \neq \beta_T$. The LR statistic has a χ^2 distribution of degree $k(m+q-1)$, where k is the number of events to which the covariate is related.

The LR test can also be used to verify whether a covariate with the time-dependent effect is significant for the events. In this case, a model with the time-varying effect covariate is compared to a model without this covariate. Again, the null hypotheses depend on the events considered. If the covariate is associated to both events, the null hypothesis $H_0 : \beta_R(t) = 0, \beta_T(t) = 0$ is tested against $H_1 : \beta_R(t) \neq 0, \beta_T(t) \neq 0$. The LR statistic has a χ^2 distribution of degree $k(m+q)$.

Joint model for two clustered survival events

An increasing number of studies favor the presence of clustered data, especially multi-center or multi-trial studies. In particular, meta-analyses are used to calculate surrogate endpoints in oncology. However, the clustering creates some heterogeneity between subjects, which needs to be accounted for. Using the above notations, the clustered joint model is similar to the model (1) but here, the index j ($j = 1, \dots, n_i$) represents a subject from cluster i ($i = 1, \dots, N$) and the cluster-specific frailty term u_i is shared by the subjects of a given group. Thus, the model can be written as (Rondeau, Pignon, and Michiels 2015):

$$\begin{cases} r_{ij}(t|u_i) = u_i r_0(t) \exp(\mathbf{X}_{Rij}^\top \boldsymbol{\beta}_R) = u_i r_{ij}(t) & \text{(time to event 1)} \\ \lambda_{ij}(t|u_i) = u_i^\alpha \lambda_0(t) \exp(\mathbf{X}_{Tij}^\top \boldsymbol{\beta}_T) = u_i^\alpha \lambda_{ij}(t) & \text{(time to event 2)} \end{cases},$$

and we assume the frailty terms u_i to be *iid* Gamma distributed. The events can be chosen arbitrarily but it is assumed that the event 2 impedes the process of the event 1. Usually the event 2 is death of a patient and the other is an event of interest (e.g., surrogate endpoint for OS) such as time to tumor progression or progression-free survival.

The interest of using the joint model for the clustered data stems from the fact that it considers the dependency between the survival processes and respects that event 2 is a competitive event for event 1. The frailty term u_i is common for a given group and represents the clustered association between the processes (at the cluster level) as well as the intra-cluster correlation.

The package **frailtypack** includes clustered joint models for two survival outcomes in presence of semi-competing risks (time-to-event, recurrent events are not allowed). To ensure identifiability of the models, it is assumed that $\alpha = 1$. For these models, only a Gamma distribution of the frailty term is implemented in the package.

Joint model for recurrent events and a terminal event with two frailty terms

In model (1) the frailty term u_i reflects the inter- and intra-subject correlation for the recurrent event as well as the association between the recurrent and the terminal events. In order to distinguish the origin of dependence, two independent frailty terms $\mathbf{u}_i = (u_i, v_i)$ can be considered (Mazroui *et al.* 2012):

$$\begin{cases} r_{ij}(t|u_i) = u_i v_i r_0(t) \exp(\mathbf{X}_{Rij}^\top \boldsymbol{\beta}_R) = u_i v_i r_{ij}(t) & \text{(recurrent event)} \\ \lambda_i(t|u_i) = u_i \lambda_0(t) \exp(\mathbf{X}_{Ti}^\top \boldsymbol{\beta}_T) = u_i \lambda_i(t) & \text{(terminal event)} \end{cases}, \quad (2)$$

where $v_i \sim \Gamma(\frac{1}{\eta}, \frac{1}{\eta})$ ($\eta > 0$) specific to the recurrent event rate and $u_i \sim \Gamma(\frac{1}{\theta}, \frac{1}{\theta})$ ($\theta > 0$) specific to the association between the processes. The variance of the frailty terms represents

the heterogeneity in the data, associated with unobserved covariates. Moreover, a high value of the variance η indicates strong dependence between the recurrent events and a high value of the variance θ indicates that the recurrent and the terminal events are strongly dependent. The individual information is equivalent to Θ_i from model (1) and the parameters to estimate are $\xi = \{r_0(\cdot), \lambda_0(\cdot), \beta_R, \beta_T, \theta, \eta\}$.

In the package **frailtypack** for model (2), called also general joint frailty model, only the Gamma distribution is allowed for the random effects.

2.2. Multivariate joint frailty model

One of the extensions to the joint model for recurrent events and a terminal event (1) is the consideration of different types of recurrence processes and time-to-event data. Two types of recurrent events are taken into account in a multivariate frailty model proposed by Mazroui *et al.* (2013). The aim of the model is to analyze dependencies between all types of events. The recurrent event times are defined by $T_{ij}^{(1)}$ ($j = 1, \dots, n_i^{(1)}$) and $T_{ij}^{(2)}$ ($j = 1, \dots, n_i^{(2)}$) and both processes are censored by the terminal event T_i . The joint model is expressed using the recurrent events and terminal event hazard functions:

$$\begin{cases} r_{ij}^{(1)}(t|u_i, v_i) = r_0^{(1)}(t) \exp(u_i + \mathbf{X}_{Rij}^{(1)\top} \beta_R^{(1)}) = \exp(u_i) r_{ij}^{(1)}(t) & \text{(rec. event 1)} \\ r_{ij}^{(2)}(t|u_i, v_i) = r_0^{(2)}(t) \exp(v_i + \mathbf{X}_{Rij}^{(2)\top} \beta_R^{(2)}) = \exp(v_i) r_{ij}^{(2)}(t) & \text{(rec. event 2)} \\ \lambda_i(t|u_i, v_i) = \lambda_0(t) \exp(\alpha_1 u_i + \alpha_2 v_i + \mathbf{X}_{Ti}^\top \beta_T) = \exp(\alpha_1 u_i + \alpha_2 v_i) \lambda_i(t) & \text{(terminal event)} \end{cases} \quad (3)$$

with vectors of regression coefficients $\beta_R^{(1)}, \beta_R^{(2)}, \beta_T$ and covariates $\mathbf{X}_{Rij}^{(1)}, \mathbf{X}_{Rij}^{(2)}, \mathbf{X}_{Ti}$ for the first recurrent event, the second recurrent event and the terminal event, respectively. The frailty terms $\mathbf{u}_i = (u_i, v_i)^\top$ explain the intra-correlation of the processes and the dependencies between them. For these correlated random effects the multivariate normal distribution of dimension 2 is considered:

$$\mathbf{u}_i = \begin{pmatrix} u_i \\ v_i \end{pmatrix} \sim \mathcal{N} \left(\mathbf{0}, \begin{pmatrix} \sigma_u^2 & \rho \sigma_u \sigma_v \\ \rho \sigma_u \sigma_v & \sigma_v^2 \end{pmatrix} \right).$$

The variances σ_u^2 and σ_v^2 explain the within-subject correlation between occurrences of the recurrent event of type 1 and type 2, respectively. The dependency between the two recurrent events is explained by the correlation coefficient ρ and the dependency between the recurrent event 1 (event 2) and the terminal event is assessed by the term α_1 (α_2) in case of significant variance σ_u^2 (σ_v^2).

Here, for each subject i we observe $\Theta_i = \{\mathbf{T}_i^{(1)}, \mathbf{T}_i^{(2)}, T_i, \boldsymbol{\delta}_i^{(1)}, \boldsymbol{\delta}_i^{(2)}, \delta_i\}$, where $\mathbf{T}_i^{(l)} = \{T_{ij}^{(l)}, j = 1, \dots, n_i^{(l)}\}$ and $\boldsymbol{\delta}_i^{(l)} = \{\delta_{ij}^{(l)}, j = 1, \dots, n_i^{(l)}\}$, $l = 1, 2$ and the parameters to estimate are $\xi = \{r_0^{(1)}(\cdot), r_0^{(2)}(\cdot), \lambda_0(\cdot), \beta_R^{(1)}, \beta_R^{(2)}, \beta_T, \sigma_u^2, \sigma_v^2, \rho, \alpha_1, \alpha_2\}$.

2.3. Bivariate joint model with longitudinal data

Here, instead of recurrent events we consider a longitudinal biomarker. For subject i we observe an l_i -vector of longitudinal measurements $\mathbf{y}_i = \{y_i(t_{ik}), k = 1, \dots, l_i\}$. Again, the true terminal event time T_i^* impedes the longitudinal process and the censoring time does not stop it but values of the biomarker are no longer observed. For each subject i we observe

$\Theta_i = \{\mathbf{y}_i, T_i, \delta_i\}$. A random variable $Y_i(t)$ representing the biomarker is expressed using a linear mixed effects model and the terminal event time T_i^* using a proportional hazards model. The sub-models are linked by the random effects $\mathbf{u}_i = \mathbf{b}_i^\top$:

$$\begin{cases} Y_i(t) = m_i(t) + \epsilon_i(t) = \mathbf{X}_{L_i}(t)^\top \boldsymbol{\beta}_L + \mathbf{Z}_i(t)^\top \mathbf{b}_i + \epsilon_i(t) & \text{(biomarker)} \\ \lambda_i(t|\mathbf{b}_i) = \lambda_0(t) \exp(\mathbf{X}_{T_i}^\top \boldsymbol{\beta}_T + h(\mathbf{b}_i, \boldsymbol{\beta}_L, \mathbf{Z}_i(t), \mathbf{X}_{L_i}(t))^\top \boldsymbol{\eta}_T) & \text{(death)} \end{cases}, \quad (4)$$

where $\mathbf{X}_{L_i}(t)$ and \mathbf{X}_{T_i} are vectors of fixed effects covariates. Coefficients $\boldsymbol{\beta}_L$ and $\boldsymbol{\beta}_T$ are constant in time. Measurements errors $\epsilon_i(\cdot)$ are *iid* normally distributed with mean 0 and variance σ_ϵ^2 . We consider a q -vector of the random effects parameters $\mathbf{b}_i = (b_0, \dots, b_{q-1})^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{B}_1)$ associated to covariates $\mathbf{Z}_i(t)$ and independent from the measurement error. In the **frailtypack** package, the maximum size of the matrix \mathbf{B}_1 is 3. The relationship between the two processes is explained via a function $h(\mathbf{b}_i, \boldsymbol{\beta}_L, \mathbf{Z}_i(t), \mathbf{X}_{L_i}(t))$ and quantified by coefficients $\boldsymbol{\eta}_T$. This possibly multivariate function represents the prognostic structure of the biomarker $m_i(t)$ (we assume that the measurement errors are not prognostic for the survival process) and in the **frailtypack** these are either the random effects \mathbf{b}_i or the current biomarker level $m_i(t)$. The structure of dependence is chosen *a priori* and should be designated with caution as it influences the model in terms of fit and predictive accuracy.

We consider that the longitudinal outcome $y_i(t_{ik})$ can be subject to a quantification limit, i.e., some observations, below a level of detection s cannot be quantified (left-censoring). We introduce this in the vector \mathbf{y}_i which includes l_i^o observed values \mathbf{y}_i^o and l_i^c censored values \mathbf{y}_i^c ($l_i = l_i^o + l_i^c$). The aspect of the left-censored data is handled in the individual contribution from the longitudinal outcome to the likelihood. The individual observed outcomes are $\Theta_i = \{\mathbf{y}_i^o, \mathbf{y}_i^c, T_i, \delta_i\}$ (in case of no left-censoring \mathbf{y}_i^c is empty and \mathbf{y}_i^o is equivalent to \mathbf{y}_i) and the parameters to estimate are $\boldsymbol{\xi} = \{\lambda_0(\cdot), \boldsymbol{\beta}_L, \boldsymbol{\beta}_T, \mathbf{B}_1, \sigma_\epsilon^2, \boldsymbol{\eta}_T\}$.

2.4. Trivariate joint model with longitudinal data

The trivariate joint model combines the joint model for recurrent events and a terminal event (model (1)) with the joint model for longitudinal data and a terminal event (4). We consider the longitudinal outcome Y observed in discrete time points $\mathbf{y}_i = \{y_i(t_{ik}), k = 1, \dots, l_i\}$, times of the recurrent event T_{ij} ($j = 1, \dots, n_i$) and the terminal event time T_i that is an informative censoring for the longitudinal data and recurrences. The respective processes are linked to each other via a latent structure. We define multivariate functions $g(\cdot)$ and $h(\cdot)$ for the associations between the biomarker and the recurrent events and between the biomarker and the terminal event, respectively. For the link between the recurrent events and the terminal event we use a frailty term v_i from the normal distribution. This distribution was chosen here to facilitate the estimation procedure involving multiple numerical integration. Interpretation of v_i should be performed with caution as the dependence between the recurrences and the terminal data is partially explained by the random effects of the biomarker trajectory. We define the model by (Król *et al.* 2016):

$$\begin{cases} Y_i(t) = m_i(t) + \epsilon_i(t) = \mathbf{X}_{L_i}(t)^\top \boldsymbol{\beta}_L + \mathbf{Z}_i(t)^\top \mathbf{b}_i + \epsilon_i(t) & \text{(biomarker)} \\ r_{ij}(t|v_i, \mathbf{b}_i) = r_0(t) \exp(v_i + \mathbf{X}_{R_{ij}}^\top \boldsymbol{\beta}_R + g(\mathbf{b}_i, \boldsymbol{\beta}_L, \mathbf{Z}_i(t), \mathbf{X}_{L_i}(t))^\top \boldsymbol{\eta}_R) & \text{(recurrent event)}, \\ \lambda_i(t|v_i, \mathbf{b}_i) = \lambda_0(t) \exp(\alpha v_i + \mathbf{X}_{T_i}^\top \boldsymbol{\beta}_T + h(\mathbf{b}_i, \boldsymbol{\beta}_L, \mathbf{Z}_i(t), \mathbf{X}_{L_i}(t))^\top \boldsymbol{\eta}_T) & \text{(terminal event)} \end{cases} \quad (5)$$

where the regression coefficients $\boldsymbol{\beta}_L, \boldsymbol{\beta}_R, \boldsymbol{\beta}_T$ are associated to possibly time-varying covariates $\mathbf{X}_{L_i}(t), \mathbf{X}_{R_{ij}}, \mathbf{X}_{T_i}$ for the biomarker, recurrent events and the terminal event (baseline

prognostic factors), respectively. The strength of the dependencies between the processes is quantified by α and vectors $\boldsymbol{\eta}_R$ and $\boldsymbol{\eta}_T$. The vector of the random effects $\mathbf{u}_i = (\mathbf{b}_i^\top, v_i)^\top$ of dimension $q = q_1 + 1$ follows the multivariate normal distribution:

$$\mathbf{u}_i = \begin{pmatrix} \mathbf{b}_i \\ v_i \end{pmatrix} \sim \mathcal{N} \left(\mathbf{0}, \begin{pmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \sigma_v^2 \end{pmatrix} \right),$$

where the dimension of the matrix \mathbf{B}_1 in the **frailtypack** package is allowed to be maximum 3. It is convenient if one assume for the biomarker a random intercept and slope:

$$Y_i(t) = m_i(t) + \epsilon_i(t) = \mathbf{X}_{L_i}(t)^\top \boldsymbol{\beta}_L + b_{i0} + b_{i1} \times t + \epsilon_i(t)$$

and then b_{i0} represents the heterogeneity of the baseline measures of the biomarker among the subjects and b_{i1} the heterogeneity of the slope of the biomarker's linear trajectory among the subjects. For non-linear trajectories, one can use parametric functions to assume two slopes of time, e.g., initial drop and long-term regrowth of tumor size under a treatment.

As for model (4) we allow the biomarker to be left-censored, again with \mathbf{y}_i^o the observed outcomes and \mathbf{y}_i^c the undetected ones. For individual i we observe then $\boldsymbol{\Theta}_i = \{\mathbf{y}_i^o, \mathbf{y}_i^c, \mathbf{T}_i^{(1)}, T_i, \delta_i^{(1)}, \delta_i\}$ and the interest is to estimate $\boldsymbol{\xi} = \{r_0(\cdot), \lambda_0(\cdot), \boldsymbol{\beta}_L, \boldsymbol{\beta}_R, \boldsymbol{\beta}_T, \mathbf{B}_1, \sigma_v^2, \sigma_\epsilon^2, \alpha, \boldsymbol{\eta}_R, \boldsymbol{\eta}_T\}$.

2.5. Estimation

Maximum likelihood estimation

Estimation of a model's parameters $\boldsymbol{\xi}$ is based on the maximization of the marginal log-likelihood derived from the joint distribution of the observed outcomes that are assumed to be independent from each other given the random effects. Let \mathbf{u}_i represent random effects of a joint model (\mathbf{u}_i can be a vector, for models (2), (3), (4), (5) or a scalar, for model (1)) and $f_{\mathbf{u}_i}(\mathbf{u}_i; \boldsymbol{\xi})$ the density function of the distribution of \mathbf{u}_i . The marginal individual likelihood is integrated over the random effects and is given by:

$$L_i(\boldsymbol{\Theta}_i; \boldsymbol{\xi}) = \int_{\mathbf{u}_i} f_{\mathbf{y}_i|\mathbf{u}_i}(\mathbf{y}_i|\mathbf{u}_i; \boldsymbol{\xi})^{\gamma_L} f_{T_i^{(1)}|\mathbf{u}_i}(\mathbf{T}_i^{(1)}, \delta_i^{(1)}|\mathbf{u}_i; \boldsymbol{\xi})^{\gamma_{R(1)}} f_{T_i^{(2)}|\mathbf{u}_i}(\mathbf{T}_i^{(2)}, \delta_i^{(2)}|\mathbf{u}_i; \boldsymbol{\xi})^{\gamma_{R(2)}} \times f_{T_i|\mathbf{u}_i}(T_i, \delta_i|\mathbf{u}_i; \boldsymbol{\xi}) f_{\mathbf{u}_i}(\mathbf{u}_i; \boldsymbol{\xi}) d\mathbf{u}_i, \quad (6)$$

where indicators γ . are introduced so that the likelihood is valid for all the joint models (1–5). Therefore, $\gamma_L = 1$ in case of models (4) and (5) and 0 otherwise, $\gamma_{R(1)} = 1$ in case of the models (1), (3) and (5) and 0 otherwise, $\gamma_{R(2)} = 1$ in case of the model (3) and 0 otherwise. The conditional density of the longitudinal outcome $f_{\mathbf{y}_i|\mathbf{u}_i}$ is the density of the l_i -dimensional normal distribution with mean $\mathbf{m}_i = \{m_i(t_{ik}), k = 1, \dots, l_i\}$ and variance $\sigma_\epsilon^2 \mathbf{I}_{m_i}$. In case of left-censored data, we observe l_i^o outcomes \mathbf{y}_i^o and l_i^c outcomes \mathbf{y}_i^c are left-censored (below the threshold s). Then $f_{\mathbf{y}_i|\mathbf{u}_i}$ can be written as a product of the density for the observed outcomes (normal with mean $m_i(t_{ik})$ and variance σ_ϵ^2) and the corresponding cumulative distribution function (cdf) $F_{\mathbf{y}_i|\mathbf{u}_i}$ for the censored outcomes:

$$f_{\mathbf{y}_i|\mathbf{u}_i}(\mathbf{y}_i|\mathbf{u}_i; \boldsymbol{\xi}) = \prod_{k=1}^{l_i^o} f_{y_i^o(t_{ik})|\mathbf{u}_i}(y_i^o(t_{ik})|\mathbf{u}_i; \boldsymbol{\xi}) \prod_{k=1}^{l_i^c} F_{y_i^c(t_{ik})|\mathbf{u}_i}(s|\mathbf{u}_i).$$

Modeling of the risk of an event can be performed either in a calendar timescale or in a gap timescale. This choice depends on the interest of the application, whether the focus is on time from a fixed date, e.g., beginning of a study, date of birth (calendar timescale) or on time between two consecutive events (gap timescale). The calendar timescale is often preferred in analyses of disease evolution, for instance, in a randomized clinical trial. In this approach, the time of entering the interval for the risk of experiencing the j th event is equal to the time of the $(j - 1)$ th event, an individual can be at risk of having the j th event only after the time of the $(j - 1)$ th event (delayed entry). On the other hand, in the gap timescale, after every recurrent event, the time of origin for the consecutive event is renewed to 0. In the medical context, this timescale is less natural than the calendar timescale, but it might be considered if the number of events is low or if the occurrence of the events does not importantly affect the subject's condition.

For the recurrent part of the model, the individual contribution from the recurrent event process of type l ($l = 1, 2$) is given by the contribution from the right-censored observations and the event times and in the calendar timescale it is:

$$f_{T_i^{(l)}|\mathbf{u}_i}(\mathbf{T}_i^{(l)}, \boldsymbol{\delta}_i^{(l)}|\mathbf{u}_i; \boldsymbol{\xi}) = \prod_{j=1}^{n_i^{(l)}} \left(r_{ij}^{(l)}(T_{ij}^{(l)}|\mathbf{u}_i; \boldsymbol{\xi}) \right)^{\delta_{ij}^{(l)}} \exp \left(- \int_{T_{i(j-1)}^{(l)}}^{T_{ij}^{(l)}} r_{ij}^{(l)}(t|\mathbf{u}_i; \boldsymbol{\xi}) dt \right).$$

For the gap timescale the lower limit of the integral is 0 and the upper limit is the gap between the time of the $(j - 1)$ th event and the j th event (Duchateau, Janssen, Kezic, and Fortpied 2003). In case of only one type of recurrent events, $r_{ij}^{(l)}$ is obviously r_{ij} . Similarly, the individual contribution from the terminal event process is:

$$f_{T_i|\mathbf{u}_i}(T_i, \delta_i|\mathbf{u}_i; \boldsymbol{\xi}) = (\lambda_i(T_i|\mathbf{u}_i; \boldsymbol{\xi}))^{\delta_i} \exp \left(- \int_0^{T_i} \lambda_i(t|\mathbf{u}_i; \boldsymbol{\xi}) dt \right).$$

For all the analyzed joint models the marginal likelihood (Equation 6) does not have an analytic form and integration is performed using quadrature methods. If a model includes one random effect that follows the Gamma distribution, the Gauss-Laguerre quadrature is used for the integral. The integrals over normally distributed random effects can be approximated using the Gauss-Hermite quadrature.

For the multivariate joint frailty model (3), the bivariate joint model for longitudinal data and a terminal event (4) and the trivariate joint model (5) it is required (except for the bivariate joint model with a random intercept only) to approximate multidimensional integrals. In this case, the standard non-adaptive Gauss-Hermite quadrature, that uses a specific number of points, gives accurate results but often can be time consuming and thus alternatives have been proposed. The multivariate non-adaptive procedure using fully symmetric interpolatory integration rules proposed by Genz and Keister (1996) offers advantageous computational time but in case of datasets in which some individuals have few repeated measurements, this method may be moderately unstable. Another possibility is the pseudo-adaptive Gauss-Hermite quadrature that uses transformed quadrature points to center and scale the integrand by utilizing estimates of the random effects from an appropriate linear mixed effects model (this transformation does not include the frailty in the trivariate model, for which the standard method is used). This method enables using less quadrature points while preserving the estimation accuracy and thus lead to a better computational time (Rizopoulos 2012).

Parametric approach

We propose to estimate the hazard functions using cubic M -splines, piecewise constant functions (PCF) or the Weibull distribution. In case of PCF and the Weibull distribution applied to the baseline hazard functions, the full likelihood is used for the estimation. Using the PCF for the baseline hazard function $h_0(t)$ (of recurrent events or a terminal event), an interval $[0, \tau]$ (with τ the last observed time among N individuals) is divided into n_{int} subintervals in one of two manners: equidistant so that all the subintervals are of the same length or percentile so that in each subinterval the same number of events is observed. Therefore, the baseline hazard function is expressed by $h_0(t) = \sum_{i=1}^{n_{\text{int}}} I_{\{t \in (t_{i-1}, t_i)\}} c_i$, with $c_i \geq 0$. In the approach of PCF the crucial point is to choose the appropriate number of the intervals so that the estimated hazard function could capture enough the flexibility of the true function. The other possibility in the parametric approach is to assume that the baseline hazard function $h_0(t)$ comes from the Weibull distribution with the shape parameter $a > 0$ and the scale parameter $b > 0$. Then the baseline hazard function is defined by $h_0(t) = (at^{a-1})/b^a$. This approach is convenient given the small number of parameters to estimate (only two for each hazard function) but the resulting estimated functions are monotone and this constraint, in some cases, might be too limiting.

Semi-parametric approach based using penalized likelihood

In the semi-parametric approach, with regard to expected smooth baseline hazard functions, the likelihood of the model is penalized by terms depending on the roughness of the functions (Joly, Commenges, and Lettenneur 1998). Cubic M -splines, polynomial functions of 3rd order, are combined linearly to approximate the baseline hazard functions (Ramsay 1988).

For the estimated parameters $\boldsymbol{\xi}$ the full log-likelihood, $l_{\text{joint}}(\boldsymbol{\xi}) = \sum_{i=1}^N \ln L_i(\boldsymbol{\Theta}_i, \boldsymbol{\xi})$ is penalized in the following way:

$$\begin{aligned} pl(\boldsymbol{\xi}) &= l_{\text{joint}}(\boldsymbol{\xi}) - \kappa_1 \int_0^\infty r_0''(t)^2 dt - \kappa_2 \int_0^\infty \lambda_0''(t)^2 dt, & \text{for model (1), (2), (5)} \\ pl(\boldsymbol{\xi}) &= l_{\text{joint}}(\boldsymbol{\xi}) - \kappa_1 \int_0^\infty r_0^{(1)''}(t)^2 dt - \kappa_2 \int_0^\infty r_0^{(2)''}(t)^2 dt - \kappa_3 \int_0^\infty \lambda_0''(t)^2 dt, & \text{for model (3) .} \\ pl(\boldsymbol{\xi}) &= l_{\text{joint}}(\boldsymbol{\xi}) - \kappa_1 \int_0^\infty \lambda_0''(t)^2 dt, & \text{for model (4)} \end{aligned}$$

The positive smoothing parameters (κ_1 , κ_2 and κ_3) provide a balance between the data fit and the smoothness of the functions. Both the full and penalized log-likelihood are maximized using the robust Marquardt algorithm (Marquardt 1963), a mixture between the Newton-Raphson and the steepest descent algorithm.

Goodness-of-fit

For verification of model assumptions, residuals are used as a standard statistical tool for visual assessment. In the context of the survival data (recurrent and terminal events) these are the martingale residuals and in the context of the longitudinal data, both, the residuals conditional on random effects and the marginal residuals are often used.

The martingale residuals model whether the number of observed events is correctly predicted by a model. Principally, it is based on the counting processes theory and for subject i and time t they are defined as the difference between the number of events of subject i until t and the Breslow estimator of the cumulative hazard function of t . Let $N_i(t)$ be the counting process of the event of type p (recurrent or terminal), \mathbf{u}_i represent the random effects and the

process's intensity $\zeta_i^{(p)}(t|\mathbf{u}_i) = \mathbf{u}_i \zeta_0^{(p)}(t) \exp(\mathbf{X}_{pi}(t)^\top \boldsymbol{\beta}_p) = \mathbf{u}_i \zeta_i^{(p)}(t)$ with $\zeta_0^{(p)}(t)$ the baseline risk and $\mathbf{X}_{pi}(t)$ the prognostic factors. The martingale residuals can be expressed by (for details, see [Commenges and Rondeau 2000](#)):

$$M_i(t) = N_i(t) - \widehat{\mathbf{u}}_i \int_0^t W_i(s) \widehat{\zeta}_i^{(p)}(s) ds, \quad (7)$$

where $W_i(t)$ is equal to 1 if the individual is at risk of the event at time t and 0 otherwise. The martingale residuals are calculated at T_i , that is at the end of the follow-up (terminal event). The assessment of the model is performed visually, the mean of the martingale residuals at a given time point should be equal to 0.

For the longitudinal data estimated in the framework of linear mixed effects models, there exist marginal residuals averaged on the population level defined by $\mathbf{R}_i^{(m)} = \mathbf{y}_i - \mathbf{X}_{Li}^\top \widehat{\boldsymbol{\beta}}_L$ and conditional residuals that are subject-specific, $\mathbf{R}_i^{(c)} = \mathbf{y}_i - \mathbf{X}_{Li}^\top \widehat{\boldsymbol{\beta}}_L - \mathbf{Z}_i^\top \widehat{\mathbf{b}}_i$. These raw residuals are recommended for checking homoscedasticity of the conditional and marginal variance. For verification of the normality assumption and detection of outlying observations, the Cholesky residuals are more adapted as they represent decorrelated residuals and are defined by:

$$\mathbf{R}_i^{(m)*} = \widehat{\mathbf{U}}_i^{(m)} \mathbf{R}_i^{(m)}, \quad \mathbf{R}_i^{(c)*} = \widehat{\mathbf{U}}_i^{(c)} \mathbf{R}_i^{(c)},$$

where the raw residuals are multiplied by the upper-triangular matrices $\widehat{\mathbf{U}}_i^{(m)}$ and $\widehat{\mathbf{U}}_i^{(c)}$ obtained by the Cholesky decomposition of the variance-covariance matrices:

$$\begin{aligned} \mathbf{V}_{\mathbf{R}_i^{(m)}} &= \widehat{\mathbf{V}}_i - \mathbf{X}_{Li} (\sum_{i=1}^N \mathbf{X}_{Li}^\top \widehat{\mathbf{V}}_i^{-1} \mathbf{X}_{Li})^{-1} \mathbf{X}_{Li}^\top = \widehat{\mathbf{U}}_i^{(m)\top} \widehat{\mathbf{U}}_i^{(m)}, \\ \mathbf{V}_{\mathbf{R}_i^{(c)}} &= \widehat{\sigma}_\epsilon^2 \mathbf{I}_{n_i} \widehat{\mathbf{V}}_i^{-1} \mathbf{V}_{\mathbf{R}_i^{(m)}} \widehat{\mathbf{V}}_i^{-1} \widehat{\sigma}_\epsilon^2 \mathbf{I}_{n_i} = \widehat{\mathbf{U}}_i^{(c)\top} \widehat{\mathbf{U}}_i^{(c)}, \end{aligned}$$

where \mathbf{V}_i is the marginal variance-covariance matrix of the longitudinal outcome \mathbf{y}_i , equal to $\mathbf{Z}_i \mathbf{B}_1 \mathbf{Z}_i^\top + \sigma_\epsilon^2 \mathbf{I}_{n_i}$ and of dimension $l_i \times l_i$. The marginal Cholesky residuals should be approximately normally distributed and thus, their normal Q-Q plot allows to verify the assumption of normality.

Both, for the calculation of the martingale residuals and the residuals of the longitudinal outcome, the values of the random effects are necessary. For this purpose, the empirical Bayes estimators of \mathbf{u}_i are calculated using the formula for the posterior probability function:

$$f(\mathbf{u}_i | \boldsymbol{\Theta}_i; \widehat{\boldsymbol{\xi}}) = \frac{f(\boldsymbol{\Theta}_i | \mathbf{u}_i; \widehat{\boldsymbol{\xi}}) f(\mathbf{u}_i; \widehat{\boldsymbol{\xi}})}{f(\boldsymbol{\Theta}_i; \widehat{\boldsymbol{\xi}})} \propto f(\boldsymbol{\Theta}_i | \mathbf{u}_i; \widehat{\boldsymbol{\xi}}) f(\mathbf{u}_i; \widehat{\boldsymbol{\xi}}).$$

For the joint models, this expression does not have an analytical solution and the numerical computation is applied that finds the \mathbf{u}_i that maximizes $f(\mathbf{u}_i | \boldsymbol{\Theta}_i; \widehat{\boldsymbol{\xi}})$:

$$\widehat{\mathbf{u}}_i = \arg \max_{\mathbf{u}_i} f(\mathbf{u}_i | \boldsymbol{\Theta}_i; \widehat{\boldsymbol{\xi}}),$$

and this is obtained using the Marquardt algorithm.

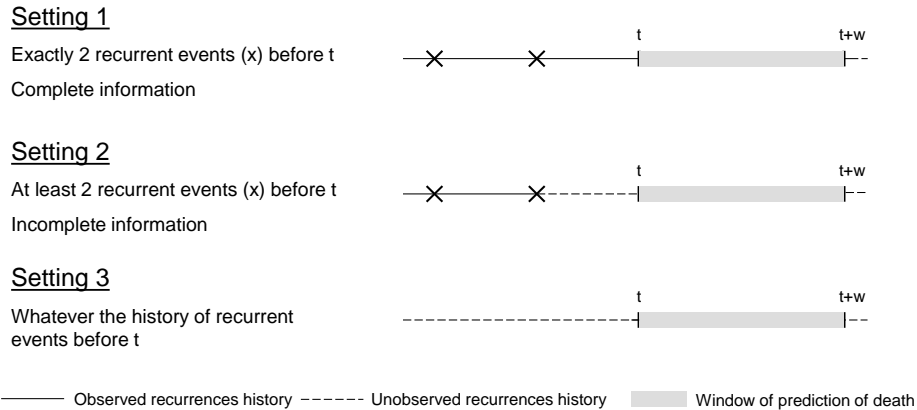


Figure 1: Three settings to take into account patient history of recurrent events in prediction. Illustration with two recurrent events.

3. Prediction of risk of the terminal event

Specific predictions can be obtained in the framework of joint modeling. Prediction consists of estimating the probability of an event at a given time $t + w$ knowing the available information at prediction time t . Using a joint model, it is possible to estimate the probability of having the terminal event at time $t + w$ given the history of the individual (occurrences of the recurrent events and/or the measurements of the longitudinal biomarker) prior to t . For the joint model for recurrent events and a terminal event (1), three settings of prediction were developed by Mauguen *et al.* (2013). In the first one, all the available information is accounted for, and we consider that this information is complete. In the second setting, all the known information is accounted for, however we consider that this information may be incomplete. Finally, in the third setting, recurrence information is not accounted for and only covariates are considered. All the settings are represented in Figure 1. Here, we focus on the first setting, for which we present the predictions and the measures of predictive accuracy but in the package all the three settings are implemented for the joint models for recurrent and terminal events.

For the joint models with a longitudinal outcome a complete history of the biomarker is considered (Król *et al.* 2016). Thus, for model (4) the individual’s history is the whole observed trajectory of the biomarker and for model (5) it is the the whole observed trajectory of the biomarker and all the observed occurrences of the recurrent event (see Figure 2).

The proposed prediction can be performed for patients from the population used to develop the model, but also for “new patients” from other populations. This is possible as the probabilities calculated are marginal, i.e., the conditional probabilities are integrated over the distribution of the random effects. Thus, values of patients’ frailties are not required to estimate the probabilities of the event. However, it should be noted that the predictions include individual deviations via the estimated parameters of the random effects’ distribution in a joint model.

We denote by t the time at which the prediction is made and by w the window of prediction. Thus, we are interested in the probability of the event (death) at time $t + w$, knowing what happened before time t . The general formulation of predicted probability of the terminal event conditional on random effects and patient’s history is:

$$P_i(t, t + w; \boldsymbol{\xi}) = P(T_i^* \leq t + w | T_i^* > t, \mathcal{F}_i(t), \mathbf{X}_i; \boldsymbol{\xi}), \quad (8)$$

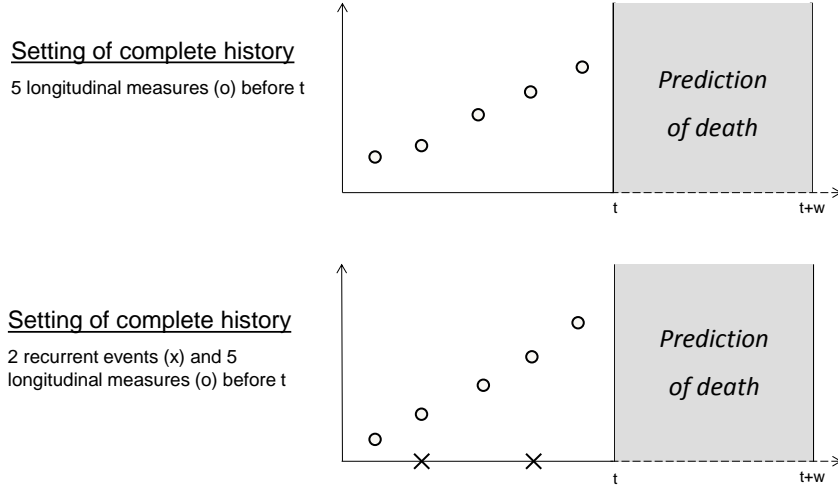


Figure 2: The possible prediction settings including the longitudinal data and considering the whole information available. The top setting corresponds to the model (4) and the bottom setting to the model (5).

where \mathbf{X}_i are all the covariates included in the model, $\mathcal{F}_i(t)$ corresponds to the complete repeated data of patient i observed until time t . We define the complete history of recurrences $\mathcal{H}_i^{J^{(l)}}(t) = \{N_i^{R^{(l)}}(t) = J^{(l)}, T_{i1}^{(l)*} < \dots < T_{iJ}^{(l)*} \leq t\}$ ($N_i^{R^{(l)}}(t)$ is the counting process of the recurrent events, $l = 1, 2$ for the two types of recurrent events, and in case of only one type of recurrent events in the models the index (l) is omitted) and the history of the biomarker $\mathcal{Y}_i(t) = \{y_i(t_{iK}), t_{i1} < \dots < t_{iK} < t\}$. Therefore, the individual's history $\mathcal{F}_i(t)$ depends on the model considered and is equal to:

$$\begin{aligned} \mathcal{F}_i(t) &= \mathcal{H}_i^J(t), & \text{for model (1), (2)} \\ \mathcal{F}_i(t) &= \{\mathcal{H}_i^{J^{(1)}}(t), \mathcal{H}_i^{J^{(2)}}(t)\}, & \text{for model (3)} \\ \mathcal{F}_i(t) &= \mathcal{Y}_i(t), & \text{for model (4)} \\ \mathcal{F}_i(t) &= \{\mathcal{H}_i^J(t), \mathcal{Y}_i(t)\}, & \text{for model (5)} \end{aligned}$$

and for the recurrent events we assume $T_{i0}^* = 0$ and $T_{i(J+1)}^* > t$. For the estimated probabilities, confidence intervals are obtained by the Monte Carlo (MC) method, using the 2.5th and 97.5th percentiles of the obtained distribution (percentile confidence interval).

3.1. Brier score

In order to validate the prediction ability of a given model, a prediction error is proposed using the weighted Brier score (Gerds and Schumacher 2006; Mauguen *et al.* 2013). It measures the distance between the prediction (probability of event) and the actual outcome (dead or alive). Inverse probability weighting corrects the score for the presence of right censoring. At a given horizon of prediction $t + w$, the error of prediction is calculated by:

$$\text{Err}_{t+w} = \frac{1}{N_t} \sum_{i=1}^{N_t} [I(T_i^* > t + w) - (1 - \hat{P}_i(t, t + w; \hat{\boldsymbol{\xi}}))]^2 \hat{w}_i(t + w, \widehat{G}_N), \quad (9)$$

where the weights $\widehat{w}_i(t+w, \widehat{G}_n)$ are defined by:

$$\widehat{w}_i(t+w, \widehat{G}_n) = \frac{I(T_i^* \leq t+w)\delta_i}{\widehat{G}_N(T_i^*)/\widehat{G}_N(t)} + \frac{I(T_i^* > t+w)}{\widehat{G}_N(t+w)/\widehat{G}_N(t)},$$

and $\widehat{G}_N(t)$ is the Kaplan-Meier estimate of the population censoring distribution at time t , N_t is the number of patients at risk of the event (alive and uncensored).

Direct calculations of the Brier score are not implemented in the package **frailtypack** but using predictions $\widehat{P}_i(t, t+w; \widehat{\boldsymbol{\xi}})$, the Brier score can be obtained using weights $\widehat{w}_i(\cdot)$ function **pec** from the package **pec** (Gerds 2016) (for details, see the colorectal example in Section 5.3). For an internal validation (on a training dataset, i.e., used for estimation) of the model as a prediction tool for new patients, a k -fold cross-validation is used to correct for over-optimism. In this procedure, the joint model estimations are performed k times on $k-1$ partitions from the random split and the predictions are calculated on the left partitions. Finally, the Brier score can be calculated using predictions for all individuals at several time points to obtain cross-validated prediction error curves.

3.2. EPOCE

Another method to evaluate a model's predictive accuracy is the EPOCE estimator that is derived using prognostic conditional log-likelihood (Commenges *et al.* 2012). This measure is adapted both for external data and then the mean prognosis observed loss (MPOL) is computed, as well as for the training data using the approximated cross-validated prognosis observed loss (CVPOL_a).

This measure of predictive accuracy is the risk function of an estimator of the true prognostic density function $f_{T^*|\mathcal{F}(t), T^* \geq t}$, where $\mathcal{F}(t)$ denotes the history of repeated measurements and/or recurrent events until time t . Using information theory this risk is defined as the expectation of the loss function, that is the estimator derived from the joint model $f_{T^*|\mathcal{F}(t), T^* \geq t}$ conditioned on $T^* > t$ and this can be written as $\text{EPOCE}(t) = \mathbf{E}(-\ln(f_{T^*|\mathcal{F}(t), T^* > t}) | T^* > t)$. In case where the model is evaluated on the training data the approximated leave-one-out CVPOL_a is defined by:

$$\text{CVPOL}_a(t) = -\frac{1}{N_t} \sum_{i=1}^{N_t} F_i(\widehat{\boldsymbol{\xi}}_i, t) + N \text{Trace}(\mathbf{H}^{-1} \mathbf{K}_t), \quad (10)$$

where \mathbf{H}^{-1} is the inverted Hessian matrix of the joint log-likelihood (Equation 6), $\mathbf{K}_t = \frac{1}{N_t(N-1)} \sum_{i=1}^N I_{\{T_i > t\}} \widehat{\mathbf{v}}_i \widehat{\mathbf{d}}_i^\top$ with $\widehat{\mathbf{v}}_i = \frac{\partial F_i(\boldsymbol{\xi}, t)}{\partial \boldsymbol{\xi}}|_{\widehat{\boldsymbol{\xi}}}$ and $\widehat{\mathbf{d}}_i = \frac{\partial l_i(\boldsymbol{\Theta}, \boldsymbol{\xi})}{\partial \boldsymbol{\xi}}|_{\widehat{\boldsymbol{\xi}}}$. The individual contribution to the log-likelihood of a terminal event at t defined for the individuals that are still at risk of the event at t can be written as:

$$F_i(\widehat{\boldsymbol{\xi}}, t) = \ln \left(\frac{\int_{\mathbf{u}_i} f_{\mathcal{Y}_i(t)|\mathbf{u}_i}(\mathcal{Y}_i(t)|\mathbf{u}_i; \widehat{\boldsymbol{\xi}})^{\gamma_L} \prod_{k=1}^l f_{\mathcal{H}_i^{J(k)}(t)|\mathbf{u}_i}(\mathcal{H}_i^{J(k)}(t)|\mathbf{u}_i; \widehat{\boldsymbol{\xi}})^{\gamma_R^{(k)}} f_{T_i|\mathbf{u}_i}(T_i, \delta_i|\mathbf{u}_i; \widehat{\boldsymbol{\xi}}) f_{\mathbf{u}_i}(\mathbf{u}_i) d\mathbf{u}_i}{\int_{\mathbf{u}_i} f_{\mathcal{Y}_i(t)|\mathbf{u}_i}(\mathcal{Y}_i(t)|\mathbf{u}_i; \widehat{\boldsymbol{\xi}})^{\gamma_L} \prod_{k=1}^l f_{\mathcal{H}_i^{J(k)}(t)|\mathbf{u}_i}(\mathcal{H}_i^{J(k)}(t)|\mathbf{u}_i; \widehat{\boldsymbol{\xi}})^{\gamma_R^{(k)}} S_i^t(T_i|\mathbf{u}_i; \widehat{\boldsymbol{\xi}}) f_{\mathbf{u}_i}(\mathbf{u}_i) d\mathbf{u}_i} \right)$$

S_i^t is the survival function of the terminal event for individual i . Again, l denotes the number of types of recurrent events included in the model.

In case of external data $MPOL(t)$ is expressed by the first component of the sum in $CVPOL_a(t)$ (Equation 10). Indeed, the second component corresponds to the statistical risk introduced to $CVPOL_a$ in order to correct for over-optimism coming from the use of the approximated cross-validation. The comparison of the models using EPOCE can be done visually by tracking the 95% intervals of difference in EPOCE.

The EPOCE estimator has some advantages compared to the prediction error, in particular for the training data the approximated cross-validation technique is less computationally demanding than the crude leave-one-out cross-validation method for the prediction error, which is important in case of joint models when time of calculation is often long (Proust-Lima *et al.* 2014).

4. Modeling and prediction using the R package `frailtypack`

4.1. Estimation of joint models

In the package `frailtypack` there are four different functions for the estimation of joint models, one for each model. The joint models for recurrent events and a terminal event (1) (and models for two clustered survival outcomes) as well as the general joint frailty models (2) are estimated with function `frailtyPenal`. The joint models for two recurrent events and a terminal event (3) are estimated with `multivPenal`. The estimation of joint models for a longitudinal outcome and a terminal event (4) is performed with `longiPenal`. Finally, function `trivPenal` estimates trivariate joint models (5). All these functions make calls to compiled Fortran codes programmed for computation and optimization of the log-likelihood. In the following, we detail each of the joint model functions.

`frailtyPenal` *function*

```
frailtyPenal(formula, formula.terminalEvent, data, recurrentAG = FALSE,
  cross.validation = FALSE, jointGeneral, n.knots, kappa, maxit = 300,
  hazard = "Splines", nb.int, RandDist = "Gamma", betaknots = 1,
  betaorder = 3, initialize = TRUE, init.B, init.Theta, init.Alpha, Alpha,
  init.Ksi, Ksi, init.Eta, LIMparam = 1e-3, LIMlogl = 1e-3, LIMderiv = 1e-3,
  print.times = TRUE)
```

Argument `formula` is a two-sided formula for a survival object ‘Surv’ from the `survival` package (Therneau 2017) and it represents the recurrent event process (the first survival outcome for the joint models in case of clustered data) with the combination of covariates on the right-hand side, the indication of a grouping variable (with term `cluster(group)`) and the indication of the variable for the terminal event (e.g., `terminal(death)`). It should be noted that the function `cluster(x)` is different from that included in the package `survival`. In both cases it is used for the identification of the correlated groups but in `frailtypack` it indicates the application of the frailty model and in `survival`, a GEE (generalized estimating equations) approach is used, without random effects. Argument `formula.terminalEvent` requires the combination of covariates related to the terminal event on the right-hand side. The name of the `data.frame` containing the variables used in the function should be specified in the

argument `data`. The logical argument `recurrentAG` indicates whether the calendar timescale for recurrent events or clustered data with the counting process approach of Andersen and Gill (1982) (`TRUE`) or the gap timescale (`FALSE`; by default) is to be used. The argument for the cross-validation `cross.validation` is not yet implemented for the joint models; thus its logical value must be `FALSE`. The smoothing parameters κ for a joint model can be chosen by first fitting suitable shared frailty and Cox models with the cross-validation method.

The general joint frailty models (2) can be estimated if argument `jointGeneral` is `TRUE`. In this case, the additional gamma frailty term is assumed and the α parameter is not considered. These models can be applied only with the Gamma distribution for the random effects.

The type of approximation of the baseline hazard functions is defined by argument `hazard` and can be chosen among `"Splines"` for semiparametric functions using equidistant intervals, `"Splines-per"` using percentile intervals, `"Piecewise-equi"` and `"Piecewise-per"` for piecewise constant functions using equidistant and percentile intervals, respectively and `"Weibull"` for the parametric Weibull baseline hazard functions. If either `"Splines"` or `"Splines-per"` is chosen for the baseline hazard functions, arguments `kappa` for the positive smoothing parameters and `n.knots` should be given with the number of knots chosen between 4 and 20 which corresponds to `n.knots+2` splines functions for approximation of the baseline hazard functions (the same number for hazard functions for both outcomes). If `"Percentile-equi"` or `"Percentile-per"` is chosen for the approximation, argument `nb.int` should be given with a 2-element vector of numbers of time intervals (1–20) for the two baseline hazard functions of the model.

Argument `RandDist` represents the type of the random effect distribution, either `"Gamma"` for the Gamma distribution or `"LogN"` for the normal distribution (log-normal joint model). If it is assumed that α in model (1) is equal to zero, argument `Alpha` should be set to `"None"`.

In case of time dependent covariates, arguments `betaknots` and `betaorder` are used for the number of inner knots used for the estimation of B -splines (1, by default) and the order of B -splines (3 for quadratic B -splines, by default), respectively.

The rest of the arguments are allocated for the optimization algorithm. Argument `maxit` declares the maximum number of iterations for the Marquardt algorithm. For a joint nested frailty model, i.e., a model that allows joint analysis of recurrent and terminal events for hierarchically clustered data, argument `initialize` determines whether the parameters should be initialized with estimated values from the appropriate nested frailty models. Arguments `init.B`, `init.Theta`, `init.Eta` and `init.Alpha` are vectors of initial values for regression coefficients, variances of the random effects and for the α parameter (by default, 0.5 is set for all the parameters). Arguments `init.Ksi` and `Ksi` are defined for joint nested frailty models and correspond to initial values for the flexibility parameter and the logical value indicating whether to include this parameter in the model or not, respectively. The convergence thresholds of the Marquardt algorithm are for the difference between two consecutive log-likelihoods (`LIMlog1`), for the difference between the consecutive values of estimated coefficients (`LIMparam`) and for the small gradient of the log-likelihood (`LIMderiv`). All these threshold values are 10^{-3} by default. Finally, argument `print.times` indicates whether to print the iteration process (the information note about the calculation process and time taken by the program after terminating), the default is `TRUE`.

The function `frailtyPenal` returns objects of class `'jointPenal'` if joint models are estimated. It should be noted that using this function for univariate models, shared frailty

models (Rondeau *et al.* 2012) and Cox models can be applied as well and they result in objects of class ‘`frailtyPenal`’. For both classes methods `print()`, `summary()` and `plot()` are available.

multivPenal function

```
multivPenal(formula, formula.Event2, formula.terminalEvent, data,
  initialize = TRUE, recurrentAG = FALSE, n.knots, kappa, maxit = 350,
  hazard = "Splines", nb.int, print.times = TRUE)
```

This function allows to fit the multivariate frailty models (3). Argument `formula` must be a two-sided formula for a ‘`Surv`’ object, corresponding to the first type of the recurrent event (no interval-censoring is allowed). Arguments `formula.Event2` refer to the second type of the recurrent event and `formula.terminalEvent` to the terminal event, and are equal to linear combinations related to the respective events. The rest of the arguments is analogical to `frailtyPenal`. Arguments `n.knots` (values between 4 and 20) and `kappa` must be vectors of length 3 for each type of event, first for the recurrent event of type 1, second for the recurrent event of type 2 and third for the terminal event. The function `multivPenal` return objects of class ‘`multivPenal`’ with `print()`, `summary()` and `plot()` methods available.

longiPenal function

```
longiPenal(formula, formula.LongitudinalData, data, data.Longi, random, id,
  intercept = TRUE, link = "Random-effects", left.censoring = FALSE, n.knots,
  kappa, maxit = 350, hazard = "Splines", nb.int, init.B, init.Random,
  init.Eta, method.GH = "Standard", n.nodes, LIMparam = 1e-3, LIMlogl = 1e-3,
  LIMderiv = 1e-3, print.times = TRUE)
```

In this function for the joint analysis of a terminal event and a longitudinal outcome, argument `formula` refers to the terminal event and the left-hand side of the formula must be a ‘`Surv`’ object and the right-hand side indicate the explanatory variables. Argument `formula.LongitudinalData` is equal to the sum of fixed explanatory variables for the longitudinal outcome. For the model, two datasets are required: `data` containing information on the terminal event process and `data.Longi` with data related to longitudinal measurements. Random effects associated to the longitudinal outcome are defined with `random` using the appropriate names of the variables from `data.Longi`. If a random intercept is assumed, the option "1" should be used. For a random intercept and slope, arguments `random` should be equal to a vector with elements "1" and the name of a variable representing time points of the biomarker measurements. At the moment, more complicated structures of the random effects are not available in the package. The name of the variable representing the individuals in `data.Longi` is indicated by `id`. The logical argument `intercept` determines whether a fixed intercept should be included in the longitudinal part or not (default is `TRUE`). Two types of subject-specific link function can be selected and are defined with the argument `link`. The default option "Random-effects" represents the link function of the random effects \mathbf{b}_i , otherwise the option is "Current-level" for the link function of the current level of the longitudinal outcome $m_i(t)$.

The initial values of the estimated parameters can be indicated by `init.B` for the fixed covariates (a vector of values starting with the parameters associated to the terminal event and then for the longitudinal outcome, interactions in the end of each component), `init.Random` for the vector of elements of the Cholesky decomposition of the covariance matrix of the random effects \mathbf{u}_i and `init.Eta` for the regression coefficients associated to the link function.

There are three methods of the Gaussian quadrature to approximate the integrals to choose from. The default "Standard" corresponds to the non-adaptive Gauss-Hermite quadrature for multidimensional integrals. The other possibility is the pseudo-adaptive Gaussian quadrature ("Pseudo-adaptive") (Rizopoulos 2012). Finally, the multivariate non-adaptive Gaussian quadrature using the algorithm implemented in a Fortran subroutine HRMSYM is indicated by "HRMSYM" (Genz and Keister 1996). The number of the quadrature nodes (`n.nodes`) can be chosen among 5, 7, 9, 12, 15, 20 and 32 using this argument (default is 9).

The function `longiPenal` returns objects of class 'longiPenal' with `print()`, `summary()` and `plot()` methods available.

trivPenal function

```
trivPenal(formula, formula.terminalEvent, formula.LongitudinalData, data,
  data.Longi, random, id, intercept = TRUE, link = "Random-effects",
  left.censoring = FALSE, recurrentAG = FALSE, n.knots, kappa, maxit = 300,
  hazard = "Splines", nb.int, init.B, init.Random, init.Eta, init.Alpha,
  method.GH = "Standard", n.nodes, LIMparam = 1e-3, LIMlogl = 1e-3,
  LIMderiv = 1e-3, print.times = TRUE)
```

The function for the trivariate joint model comprises three formulas for each type of process. The first two arguments are analogous to `frailtyPenal`, argument `formula`, referring to recurrent events, is a two-sided formula for a 'Surv' object on the left-hand side and covariates on the right-hand side (with indication of the variable for the terminal event using method `terminal`) and argument `formula.terminalEvent` represents the terminal event and is equal to a linear combination of the explanatory variables. Finally, argument `formula.LongitudinalData` as in function `longiPenal` corresponds to the longitudinal outcome indicating the fixed effect covariates. The rest of the arguments are detailed in the descriptions of functions `frailtyPenal` and `longiPenal`. The function `trivPenal` returns objects of class 'trivPenal' with `print()`, `summary()` and `plot()` methods available.

4.2. Prediction

The current increase of interest in the joint modeling of correlated data is often related to the individual predictions that these models offer. Indeed, calculating the probabilities of a terminal event given a joint model results in precise predictions that consider the past of an individual. Moreover, there exist statistical tools that evaluate a model's capacity for these dynamic predictions. In the package **frailtypack** we provide the `prediction` function for dynamic predictions of a terminal event in a finite horizon, the `epoce` function for evaluating predictive accuracy of a joint model and the `Diffepoce` function for comparing the accuracy of two joint models.

Predicted probabilities with `prediction` function

In the package it is possible to calculate the prediction probabilities for the Cox proportional hazard, shared frailty (for clustered data, [Rondeau et al. 2012](#)) and joint models. Among the joint models the predictions are provided for the standard joint frailty models (recurrent events and a terminal event), for the joint models for a longitudinal outcome and a terminal event and for the trivariate joint models (a longitudinal outcome, recurrent events and a terminal event). These probabilities can be calculated for a given prediction time and window or for a vector of times, with varying prediction time or varying window. For the shared frailty models for clustered events, marginal and conditional on a specific cluster predictions can be calculated and for the joint models only the marginal predictions are provided. Finally, for the joint frailty models the predictions are calculated in three settings: given the exact history of recurrences, given the partial history of recurrences (the first J recurrences) and ignoring the past recurrences. For the joint models with a longitudinal outcome (bivariate and trivariate) only the predictions considering the patient's complete history are provided. For all the aforementioned predictions the following function is proposed:

```
prediction(fit, data, data.Longi, t, window, group, MC.sample = 0)
```

Argument `fit` indicates the, a `'frailtyPenal'`, `'jointPenal'`, `'longiPenal'` or `'trivPenal'` object. The data with individual characteristics for predictions must be provided in the data frame `data` with information on the recurrent events and covariates related to recurrences and the terminal event and in case of `'longiPenal'` and `'trivPenal'` objects the data frame `data.Longi` needs to contain repeated measurements and covariates related to the longitudinal outcome. These two datasets must refer to the same individuals for which the predictions will be calculated. Moreover, the names and the types of variables should be equivalent to those in the dataset used for estimation. The details on how to prepare correctly the data are presented in appropriate examples (Section 5).

Argument `t` is a time or vector of times for predictions and `window` is a horizon or vector of horizons. The function calculates the probability of the terminal event between a time of prediction and a horizon (both arguments are scalars), between several times of prediction and a horizon (`t` is a vector and `window` a scalar) and finally, between a time of prediction and several horizons (`t` is a scalar and `window` a vector of positive values).

For all the predictions, confidence intervals can be calculated using the MC method with `MC.sample` number of samples (maximum 1000). If the confidence bands are not to be calculated argument `MC.sample` should be equal to 0 (the default).

Predictive accuracy measure with `epoce` and `Diffepoce` functions

Predictive ability of joint models can be evaluated with function `epoce` that computes the estimators of EPOCE, MPOL and $CVPOL_a$. For a given estimation, the evaluation can be performed on the same data and then both MPOL and $CVPOL_a$ are calculated, as well as on a new dataset but then only MPOL is calculated as the correction for over-optimism is not necessary. The call of the function is:

```
epoce(fit, pred.times, newdata = NULL, newdata.Longi = NULL)
```

with `fit` an object of class `'jointPenal'`, `'longiPenal'` or `'trivPenal'`, `pred.times` a vector of time for which the calculations are performed. In case of external validation, new

datasets `newdata` and `newdata.Longi` should be provided (`newdata.Longi` only in case of ‘`longiPenal`’ and ‘`trivPenal`’ objects). However, the names and types of variables in `newdata` and `newdata.Longi` must be the same as in the datasets used for estimation.

To compare the predictive accuracy of two joint models fit to the same data but possibly with different covariates, the simple comparison of obtained values of EPOCE can be enhanced by calculating the 95% tracking interval of difference between the EPOCE estimators. For this purpose we propose the function:

```
Diffepoce(epoce1, epoce2)
```

where `epoce1` and `epoce2` are objects inheriting from the ‘`epoce`’ class.

5. Illustrating examples

The package **frailtypack** provides various functions for models for correlated outcomes and survival data. The Cox proportional hazard model, the shared frailty model for recurrent events (clustered data), the nested frailty model, the additive frailty model and the joint frailty model for recurrent events and a terminal event have already been illustrated elsewhere (Rondeau and Gonzalez 2005; Rondeau *et al.* 2012).

In this section we focus on extended models for correlated data presented in Section 2 using three datasets: `readmission`, `dataMultiv` and `colorectal`. Although the joint frailty model has already been presented in the literature, we illustrate its usage as the form of the function has developed in the meantime.

5.1. Example on dataset `readmission` for joint frailty models

Dataset `readmission` comes from a rehospitalization study of patients after a surgery and diagnosed with colorectal cancer (Gonzalez *et al.* 2005; Rondeau *et al.* 2012). It contains information on times (in days) of successive hospitalizations and death (or last registered time of follow-up for right-censored patients) counting from date of surgery, and patient characteristics: type of treatment, sex, Dukes’ tumoral stage, comorbidity Charlson’s index and survival status. The dataset includes 403 patients with 861 rehospitalization events in total. Among the patients 112 (28%) died during the study.

Standard joint frailty model

We adapt the joint model for recurrent events and a terminal event using the gap timescale from the example given in Rondeau *et al.* (2012). The model `modJoint.gap` is defined:

```
R> library("frailtypack")
R> data("readmission", package = "frailtypack")
R> modJoint.gap <- frailtyPenal(Surv(time, event) ~ cluster(id) + dukes +
+   charlson + sex + chemo + terminal(death),
+   formula.terminalEvent = ~ dukes + charlson + sex + chemo,
+   data = readmission, n.knots = 8, kappa = c(2.11e+08, 9.53e+11))
```

This model includes Dukes’ stage, Charlson’s index, sex and treatment as covariates for both hospitalizations and death. The frailties are from the Gamma distribution (default option)

and the baseline hazard functions are approximated by splines with 8 knots and the smoothing parameters for the penalized log-likelihood are $2.11e + 8$ for the recurrent part and $9.53e + 11$ for the terminal part. To find the optimal number of knots, we first fitted the model with a small number of knots (`n.knots = 4`) and increased the number of knots until the graph of the baseline hazard functions was not changing importantly anymore. The smoothing parameters are obtained from a shared frailty and Cox model with respectively recurrent and terminal event as the outcome using the cross-validation method.

With this model, it is found that chemotherapy is a prognostic factor only on death with a positive association ($HR = 2.99$, $p < 0.001$). Both Charlson's index (Index ≥ 3 vs. Index 0) and Dukes' stage (Stage C and D vs. Stages A, B) are positively related to the recurrent and terminal events. A detailed description of the output for the standard joint frailty models are presented in [Rondeau *et al.* \(2012\)](#).

To verify whether the model predicts correctly the number of observed events, we represent the martingale residuals for both events against the follow-up time. These residuals in a well adjusted model should have a mean equal to 0 and thus a smoothing curve added to a graph should be approximately overlapping with the horizontal line $y = 0$. The following code produces the plots given in Figure 3:

```
R> plot(aggregate(readmission$t.stop, by = list(readmission$id),
+ FUN = max)[2][, 1], modJoint.gap$martingale.res, ylab = "",
+ xlab = "time", main = "Rehospitalizations", ylim = c(-4, 4))
R> lines(lowess(aggregate(readmission$t.stop, by = list(readmission$id),
+ FUN = max)[2][, 1], modJoint.gap$martingale.res, f = 1), lwd = 3,
+ col = "grey")
R> plot(aggregate(readmission$t.stop, by = list(readmission$id),
+ FUN = max)[2][, 1], modJoint.gap$martingaledeath.res, ylab = "",
+ xlab = "time", main = "Death", ylim = c(-2, 2))
R> lines(lowess(aggregate(readmission$t.stop, by = list(readmission$id),
+ FUN = max)[2][, 1], modJoint.gap$martingaledeath.res, f = 1), lwd = 3,
+ col = "grey")
```

For the rehospitalization process the mean of residuals is approximately 0 with the smooth curve close to the line $y = 0$, but in case of death this tendency is deviated by relatively higher values for short follow-up times. This may suggest, that the model may have underestimated the number of deaths in the early follow-up period. The identified individuals of which the residuals result in non-zero mean, have short intervals between their rehospitalization and death (1 day). Indeed, the removal of these individuals (50 patients) results in residuals with the mean close to 0 all along the follow-up period (plot not shown here).

The package **frailtypack** provides also the estimation of the random effects. The vector `frailty.pred` from a 'jointPenal' object contains the individual empirical Bayes estimates. They can be graphically represented for each individual with additional information on number of events (point size) to identify the outlying data.

```
R> plot(1:403, modJoint.gap$frailty.pred, xlab = "Id of patients",
+ ylab = "Frailty predictions for each patient", type = "p", axes = FALSE,
+ cex = as.vector(table(readmission$id)), pch = 1, ylim = c(-0.1, 7),
+ xlim = c(-2, 420))
```

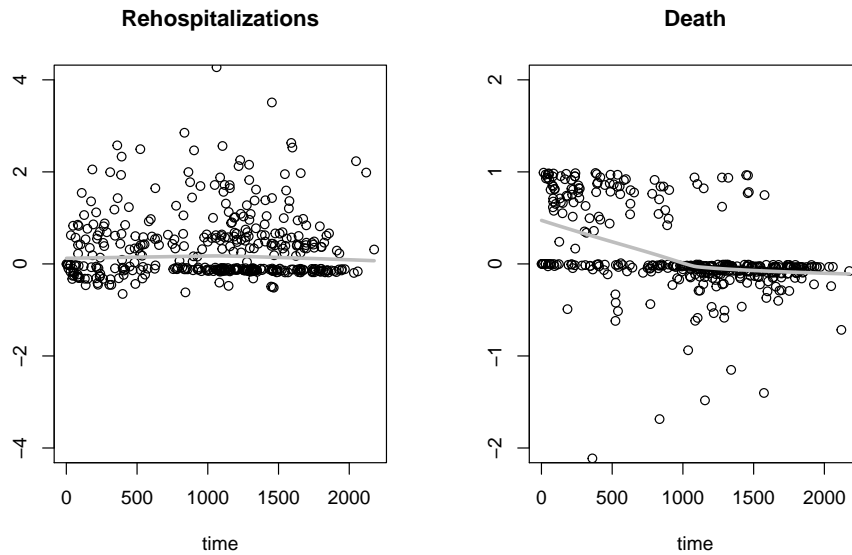


Figure 3: Martingale residuals for rehospitalizations and death against the follow-up time (in days). The grey line corresponds to a smooth curve obtained with `lowess`.

```
R> axis(1, round(seq(0, 403, length = 10), digit = 0))
R> axis(2, round(seq(0, 7, length = 10), digit = 1))
```

Figure 4 shows the values of frailty prediction for each patient with an association to the number of events (the bigger the point, the greater the number of rehospitalizations). The frailties tend to have bigger values if the number of events of a given individual is high. From the plot it can be noticed that there is an outlying frailty suggesting verification of the follow-up of the concerned individual.

Time-varying coefficients

In the framework of standard joint frailty models, it is possible to fit the models with time-varying effects of prognostic factors. Using function `timedep` in a formula of `frailtyPenal`, the time-dependent coefficients can be estimated using B -splines of order q (option `betaorder`) with m interior knots (option `betaknots`). In the example of the `readmission` dataset we are interested in verifying whether the variable `sex` has a time-varying effect on both recurrent and terminal events. Thus, we fit a model equivalent to `modJoint.gap` but with time dependent effects assuming quadratic B -splines and 3 interior knots:

```
R> modJoint.gap.timedep <- frailtyPenal(Surv(time, event) ~ cluster(id) +
+   dukes + charlson + timedep(sex) + chemo + terminal(death),
+   formula.terminalEvent = ~ dukes + charlson + timedep(sex) + chemo,
+   data = readmission, n.knots = 8, kappa = c(2.11e+08, 9.53e+11),
+   betaorder = 3, betaknots = 3)
```

In the result, using the method `print` the estimated values of parameters with time-constant effects and graphics of log-hazard ratios for time-dependent variables for each event are obtained (Appendix B, Figure 10). For rehospitalizations, we found firstly a protective effect

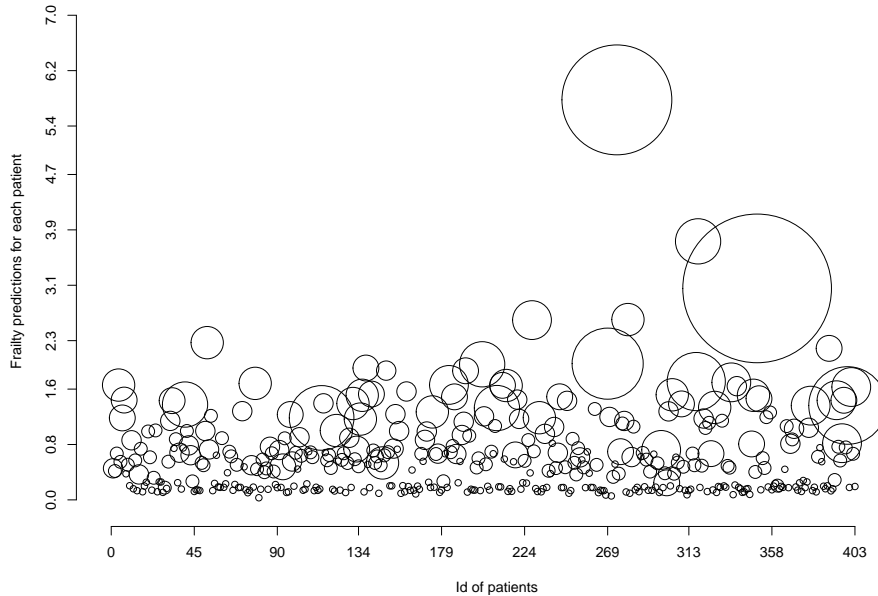


Figure 4: Individual prediction of the frailties. The size of points corresponds to number of an individual's recurrent events.

for females ($\beta(t) < 0$) and later an increased risk ($\beta(t) > 0$). For death, at the beginning, the effect of sex was weakening the risk but shortly became non-influential ($\beta(t)$ around 0).

The PH assumption for the variable sex can be checked using the LR test. We compare two models: `modJoint.gap` (related to the null hypothesis that the effect is constant in time: $H_0 : \beta_R(t) = \beta_R, \beta_T(t) = \beta_T$) and `modJoint.gap.timedep` (related to the alternative hypothesis of time-varying effects: $H_1 : \beta_R(t) \neq \beta_R, \beta_T(t) \neq \beta_T$):

```
R> LR.statistic <- -2 * modJoint.gap$logLik +
+ 2 * modJoint.gap.timedep$logLik
R> p.value <- signif(1 - pchisq(LR.statistic, df = 10), 5)
```

Given the obtained p value = 0.049, the PH assumption for the variable sex is not satisfied (at the level 0.05). Next, we check whether sex with time-dependent effects is an influential prognostic factor. Thus, again, we use the LR test to compare two models: `modelJoint.gap.nosex` without the covariate sex (model related to the null hypothesis: $H_0 : \beta_R(t) = 0, \beta_T(t) = 0$) and `modelJoint.gap.timedep` (related to the alternative hypothesis: $H_1 : \beta_R(t) \neq 0, \beta_T(t) \neq 0$):

```
R> modJoint.gap.nosex <- frailtyPenal(Surv(time, event) ~ cluster(id) +
+   dukes + charlson + chemo + terminal(death), formula.terminalEvent = ~
+   dukes + charlson + chemo, data = readmission, n.knots = 8,
+   kappa = c(2.11e+08, 9.53e+11))
R> LR.statistic <- -2 * modJoint.gap.nosex$logLik +
+ 2 * modJoint.gap.timedep$logLik
R> p.value <- signif(1 - pchisq(LR.statistic, df = 12), 5)
```

The test shows that the variable `sex` has a significant time-varying effect (p value < 0.001). In this example, the variable `sex` is found significant for both, the PH model and the non-PH model, but we showed that this variable does not satisfy the PH assumption.

Dynamic predictions

Using the joint model `modJoint.gap` we can calculate predicted probabilities of death using `prediction`. These predictions may serve as a tool to compare two exemplars of patients with different histories of recurrences but the same values of the prognostic factors to study the effect of the events on the survival (Mauguen *et al.* 2013). On the other hand, the influence of some explanatory covariates can be examined for patients that are considered to have the same history of recurrences. Here, we aim at evaluating the predictive effect of the Dukes' stage on survival considering the history of hospitalizations. We compare the predicted risk of death for two patients having the same characteristics (men with Charlson's index 0 and the chemotherapy treatment) and having two hospitalizations: 1 and 1.5 year after their surgeries. We set the time of prediction to 2 years and calculate the probability in a time window of 3 years (we apply a moving window with a step of 0.5 years). Patient 1 has Dukes' stage A and patient 2, Dukes' stage D. We focus on the predicted probabilities regarding the complete history of recurrences (Equation 8) and compare the results with those obtained using the incomplete history and ignoring the history.

To prepare data for the predictions, we start with an empty data frame with the variables of interest and the covariates:

```
R> datapred <- data.frame(time = 0, event = 0, id = 0, dukes = 0,
+   charlson = 0, sex = 0, chemo = 0)
R> datapred[, 4:7] <- lapply(datapred[, 4:7], as.factor)
R> levels(datapred$dukes) <- c(1, 2, 3)
R> levels(datapred$charlson) <- c(1, 2, 3)
R> levels(datapred$sex) <- c(1, 2)
R> levels(datapred$chemo) <- c(1, 2)
```

Patient 1 with Dukes' stage A has two observed hospitalizations at the 365th and 548th day after the surgery:

```
R> datapred[1, ] <- c(365, 1, 1, 1, 1, 1, 2)
R> datapred[2, ] <- c(548, 1, 1, 1, 1, 1, 2)
```

Patient 2 has the hospitalizations observed in the same times as Patient 1 but is assumed to have Dukes' stage B:

```
R> datapred[3, ] <- c(365, 1, 2, 3, 1, 1, 2)
R> datapred[4, ] <- c(548, 1, 2, 3, 1, 1, 2)
```

We calculate the predictions for both patients:

```
R> pred <- prediction(modJoint.gap, datapred, t = 730,
+   window = seq(1, 1096, 183), MC.sample = 500)
R> plot(pred, conf.bands = TRUE)
```

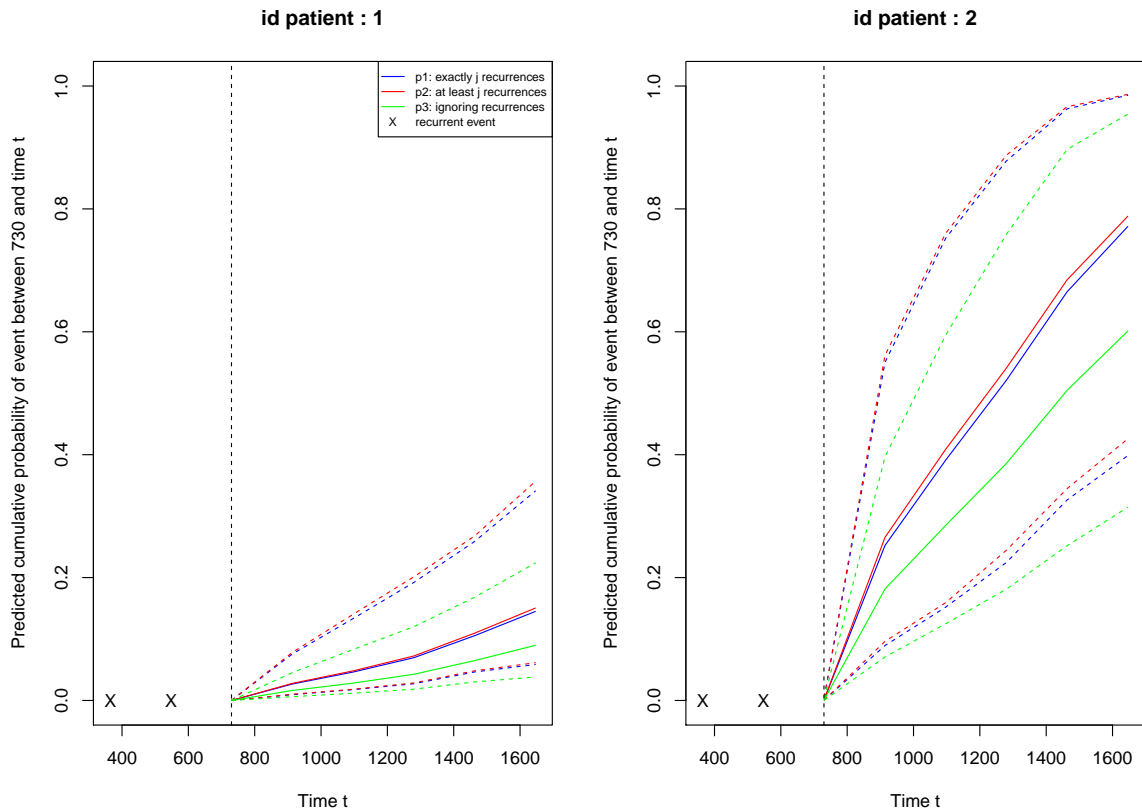



Figure 5: Predicted probabilities of death for two patients sharing the same history of recurrences and characteristics except of the Dukes' stage. Patient 1 has Dukes' stage A and patient 2, Dukes' stage D. Dashed lines represent the MC confidence intervals.

In the result, three types of predictions are calculated for 6 time horizons. As it has been already observed from the estimates of the model, an increased Dukes' stage (D) is positively associated with death. Figure 5 compares the predicted probabilities of death in the three settings. The predictions using the exact number of recurrences (2 hospitalizations) and at least 2 recurrences are very close to each other and are higher than the risk obtained without the information on recurrent events. Indeed, the significant estimate of variance of the frailty θ and significant positive estimate of α imply the positive association between the process of recurrences and death. Finally, all the predicted probabilities are higher for the patient with Dukes' stage D compared to the patient with Dukes' stage A.

Joint frailty model for clustered data

The joint models for clustered survival data can be estimated using again the `frailtyPenal` function. A dataset should include information on two survival outcomes for individuals from several groups. This model is presented using the `readmission` dataset with artificially created clusters on individuals. The first survival event will be the first observed rehospitalization and the second event, death. The framework of semi-competing risks is used here, thus individuals' follow-up stops at time of the rehospitalization, death or in case when none of these events are observed, the censoring time. We consider 6 clusters defined by a new variable `group`:

```
R> readmission <- transform(readmission, group = id %% 6 + 1 )
R> readm_cluster <- subset(readmission,
+   (t.start == 0 & event == 1) | event == 0)
```

New dataset `readm_cluster` includes clusters with 97 to 107 individuals per group. For definition of a clustered joint model two inner functions are required in `frailtyPenal`, `num.id` for the individual level and `cluster` for the groups:

```
R> joi.clus <- frailtyPenal(Surv(t.start, t.stop, event) ~ cluster(group) +
+   num.id(id) + dukes + sex + chemo + terminal(death),
+   formula.terminalEvent = ~ dukes + sex + chemo, data = readm_cluster,
+   n.knots = 8, kappa = c(1.e+10, 1.e+10), recurrentAG = TRUE,
+   Alpha = "None")
```

In the result, the estimates of prognostic factors for both types of events are obtained. The estimate of the variance of the frailty term indicates whether, at the cluster level, the processes are associated with each other and measures the heterogeneity between individuals (intra-cluster correlation). In the given example the estimate of the variance θ is significantly different from 0 (p value = 0.037), thus there is a positive association between the risk of hospitalizations and death via the non-observed frailty.

In case of the joint frailty models for clustered data it should be noted that sufficient amount of information must be provided, i.e., number of observations per cluster. Otherwise, given the complexity of the model, convergence might not be attained. The parameter α is assumed to be equal to 1 as these models are defined in the framework of semi-competing risks and not of recurrent events.

General joint frailty model

To estimate the general frailty model, argument `jointGeneral` must be equal to `TRUE` in function `frailtyPenal`. We applied this model to the original `readmission` dataset assuming two independent frailty terms using the following code:

```
R> modJoint.general <- frailtyPenal(Surv(time, event) ~ cluster(id) +
+   dukes + charlson + sex + chemo + terminal(death),
+   formula.terminalEvent = ~ dukes + charlson + sex + chemo,
+   data = readmission, jointGeneral = TRUE, n.knots = 8,
+   kappa = c(2.11e+08, 9.53e+11))
```

In the output of the function, estimations of the variances of both random effects are given. For the analyzed example, the estimated variance θ of the frailty u_i associating recurrent events and death indicates strong relationship between the processes ($\hat{\theta} = 0.68$, p value < 0.001). Moreover, the estimate of η implies small but significant dependence between the recurrent event gap times explained by the frailty v_i ($\hat{\eta} = 0.01$, p value < 0.001). This information complements the inference from the standard joint frailty model because it separates the correlation linked to the recurrent events with correlation between the recurrent events and the terminal event.

5.2. Example on dataset `dataMultiv` for multivariate joint frailty model

For the following example we use a generated dataset for 800 individuals from model (3) with 2 types of recurrent events and a terminal event. The random effects are assumed to follow the normal distribution $u_i \sim \mathcal{N}(0, 0.5)$, $v_i \sim \mathcal{N}(0, 0.5)$ and the correlation coefficient $\rho = 0.5$. The coefficients for the random effects are $\alpha_1 = \alpha_2 = 1$. The baseline hazard functions $r_0^{(1)}(t)$, $r_0^{(2)}(t)$ and $\lambda_0(t)$ follow the Weibull distribution and the time for right censoring is fixed at 5. The generated data includes 1652 observations. For a detailed description of the generation scenario see Mazroui *et al.* (2012).

The dataset includes the individuals' times of events with variables indicating the type of event: `INDICREC` for the recurrent event of type 1 (local recurrences), `INDICMETA` for the recurrent event of type 2 (metastases) and `INDICDEATH` for the censoring status (death). Additionally there are 3 binary covariates `v1`, `v2` and `v3`.

Multivariate frailty model

We consider the multivariate frailty model for the exemplary dataset `dataMultiv` to study jointly local recurrences, metastases and death for patients diagnosed with cancer. To define the model, three formulas must be defined in the function with additional indication on variables including status of the second recurrent event (`event2`) and of the terminal event (`terminal`), both included in the first formula. All the baseline hazard functions must be of the same type (Weibull, splines or piecewise constant). We fit the model as follows (computational time 54 minutes on a personal computer with an Intel Core i7 3.40 GHz processor and 8 GB RAM running Windows 7):

```
R> data("dataMultiv", package = "frailtypack")
R> modMultiv.spli <- multivPenal(Surv(TIMEGAP, INDICREC) ~ cluster(PATIENT) +
+   v1 + v2 + event2(INDICMETA) + terminal(INDICDEATH), formula.Event2 = ~
+   v1 + v2 + v3, formula.terminalEvent = ~v1, data = dataMultiv,
+   n.knots = c(8, 8, 8), kappa = c(1, 1, 1), initialize = FALSE)
```

Option `initialize` indicates whether to initialize the parameters (including parameters for the baseline hazard functions) using the estimates of separate models: shared frailty models (for the two types of recurrent events) and a Cox proportional hazard model (for the terminal event). The output of function `multivPenal` is presented below:

```
R> modMultiv.spli
```

Call:

```
multivPenal(formula = Surv(TIMEGAP, INDICREC) ~ cluster(PATIENT) +
+   v1 + v2 + event2(INDICMETA) + terminal(INDICDEATH), formula.Event2 = ~
+   v1 + v2 + v3, formula.terminalEvent = ~v1, data = dataMultiv,
+   initialize = FALSE, n.knots = c(8, 8, 8), kappa = c(1, 1, 1))
```

Multivariate joint gaussian frailty model for two survival outcomes
and a terminal event
using a Penalized Likelihood on the hazard function

Recurrences 1:

```
-----
      coef exp(coef) SE coef (H) SE coef (HIH)      z      p
v1 0.565676  1.76064   0.111603   0.111638 5.06863 4.0068e-07
v2 0.631891  1.88117   0.106534   0.106519 5.93138 3.0040e-09
```

Recurrences 2:

```
-----
      coef exp(coef) SE coef (H) SE coef (HIH)      z      p
v1 0.837140  2.309752   0.127631   0.127554 6.55905 5.4152e-11
v2 -0.641487  0.526509   0.127111   0.127075 -5.04668 4.4956e-07
v3 0.312774  1.367212   0.118103   0.118057 2.64832 8.0892e-03
```

Terminal event:

```
-----
      coef exp(coef) SE coef (H) SE coef (HIH)      z      p
v1 0.367778  1.44452   0.0987691   0.0984928 3.72362 0.00019639
```

Parameters associated with Frailties:

```
theta1 : 0.523131 (SE (H): 0.537725 ) p = 0.16531
theta2 : 0.25968 (SE (H): 0.966704 ) p = 0.39411
alpha1  : 0.54705 (SE (H): 0.111603 ) p = 9.4993e-07
alpha2  : 0.595186 (SE (H): 0.106534 ) p = 2.3125e-08
rho     : 0.738084 (SE (H): 0.0987691 )
```

penalized marginal log-likelihood = -594.7

LCV = the approximate likelihood cross-validation criterion
in the semi parametric case = 0.477466

n= 1318

n recurrent events of type 1= 518 n subjects= 800

n recurrent events of type 2= 334

n terminal events= 636

number of iterations: 16

Exact number of knots used: 8 8 8

Value of the smoothing parameters: 1 1 1

The output presents the results for prognostic factor estimates for each type of event. The estimates of parameters associated with the random effects are given by the variance of the frailty related to the first type of the recurrent events and the association with the terminal event (`theta1`), the variance of the frailty related to the second type of the recurrent events and the association with the terminal event (`theta2`) and the correlation coefficient between the frailties (`rho`). The sign and strength of the dependency between the recurrent event of type 1 (2) and the terminal event is represented by `alpha1` (`alpha2`). In the analyzed example, both `theta1` and `theta2` are not significantly different from 0, thus there are no dependencies between the processes explained by the non-observed factors.

5.3. Example on dataset colorectal for models with longitudinal data

Datasets `colorectal` and `colorectal.Longi` represent a random selection of 150 patients from the multi-center randomized phase III clinical trial FFCD 2000-05 of patients diagnosed with metastatic colorectal cancer not amenable to curative intent surgery. The trial was conducted between 2002 and 2007 in France by Fédération Francophone de Cancérologie Digestive (FFCD; Ducreux *et al.* 2011). The data contains a follow-up of tumor size measurements (sum of the longest diameters of target lesions) and times of apparition of new lesions as recurrent events. Moreover, some baseline characteristics (age, WHO performance status and previous resection), treatment arm (combination vs. sequential) and time of death (or last observed time for a right-censored individual) are included in the data. Dataset `colorectal` provides information on the recurrent event and death and dataset `colorectal.Longi` on the measurements of tumor size. A total of 906 tumor size measurements and 289 of recurrences were recorded for the patients included. Among them, 121 died during the study.

The variable `tumor.size` in `colorectal.Longi` is the transformed sum of the longest diameters (SLD^*) of an individual's target lesions measured during a visit ($SLD^* = (SLD^{0.3} - 1)/0.3$). The status of new lesions occurrence is registered in `new.lesions` in dataset `colorectal`. In this dataset start of time interval `time0` (0 or time of previous recurrence) and time of event `time1` (recurrence or censoring by terminal event) represent information for times of apparition of new lesions and for death (or right censoring).

We provide extracts of `colorectal` and `colorectal.Longi` datasets to guide the users how to prepare suitable datasets for joint models with longitudinal data using the package. These functions require datasets in long format (one row per observation and usually several rows per individuals) and only in case of data in `longiPenal` the long format is one row per individual (as it contains information on the terminal event only). In the example of the `colorectal` dataset this is represented as follows:

```
R> data("colorectalLongi", package = "frailtypack")
R> head(colorectalLongi, 10)
```

	id	year	tumor.size	treatment	age	who.PS	prev.resection
1	1	0.0000000	5.2276794	S	60-69 years	0	No
2	1	0.2131147	4.4926205	S	60-69 years	0	No
3	1	0.4590164	4.6000876	S	60-69 years	0	No
4	1	0.6311475	4.5333227	S	60-69 years	0	No
5	2	0.0000000	3.0454011	C	>69 years	0	No
6	2	0.1639344	1.3919052	C	>69 years	0	No
7	2	0.2814208	1.2063562	C	>69 years	0	No
8	2	0.4316940	1.2063562	C	>69 years	0	No
9	2	0.5846995	0.9462067	C	>69 years	0	No
10	2	0.7377049	1.9353985	C	>69 years	0	No

```
R> data("colorectal", package = "frailtypack")
R> head(colorectal, 5)
```

	id	time0	time1	new.lesions	treatment	age	who.PS
1	1	0.0000000	0.7095890	0	S	60-69 years	0

2	2	0.0000000	1.2821918	0	C	>69 years	0
3	3	0.0000000	0.5245902	1	S	60-69 years	1
4	3	0.5245902	0.9207650	1	S	60-69 years	1
5	3	0.9207650	0.9424658	0	S	60-69 years	1
		prev.resection	state	gap.time			
1		No	1	0.70958904			
2		No	1	1.28219178			
3		No	0	0.52459017			
4		No	0	0.39617486			
5		No	1	0.02170073			

Joint model for longitudinal data and a terminal event

Firstly, we estimate the bivariate joint model for longitudinal data and a terminal event (4). We consider a left-censored biomarker, i.e., the transformed tumor size measurements are not observed below a threshold -3.33 (which corresponds to 'zero' measures in the nontransformed data). The value of the smoothing parameter κ is chosen using a corresponding reduced model, i.e., a Cox model for the terminal event. For a model with a random intercept and slope for the biomarker and the link function being the random effects of the biomarker we use the following form of the `longiPenal` function using the `colorectalLongi` dataset and a subset of `colorectal` containing only the information on the terminal event (`colorectalSurv`):

```
R> colorectalSurv <- subset(colorectal, new.lesions == 0)
R> modLongi <- longiPenal(Surv(time1, state) ~ age + treatment + who.PS +
+   prev.resection, tumor.size ~ year * treatment + age + who.PS,
+   colorectalSurv, data.Longi = colorectalLongi, random = c("1", "year"),
+   id = "id", link = "Random-effects", left.censoring = -3.33,
+   n.knots = 8, kappa = 0.93, method.GH = "Pseudo-adaptive", n.nodes = 7)
R> modLongi
```

Call:

```
longiPenal(formula = Surv(time1, state) ~ age + treatment + who.PS +
  prev.resection, formula.LongitudinalData = tumor.size ~ year *
  treatment + age + who.PS, data = colorectalSurv,
  data.Longi = colorectalLongi, random = c("1", "year"), id = "id",
  link = "Random-effects", left.censoring = -3.33, n.knots = 8,
  kappa = 0.93, method.GH = "Pseudo-adaptive", n.nodes = 7)
```

Joint Model for Longitudinal Data and a Terminal Event

Parameter estimates using a Penalized Likelihood on the hazard function
and assuming left-censored longitudinal outcome

Association function: random effects

Longitudinal outcome:

	coef	SE	coef (H)	SE	coef (HIH)	z	p
--	------	----	----------	----	------------	---	---

Intercept	3.023301	0.187402	0.187402	16.132720	<1e-16
year	-0.303137	0.135874	0.135874	-2.231016	2.5680e-02
treatmentC	0.048672	0.213962	0.213962	0.227480	8.2005e-01
age60-69 years	-0.017022	0.168788	0.168788	-0.100847	9.1967e-01
age>69 years	-0.294602	0.131867	0.131867	-2.234076	2.5478e-02
who.PS1	0.106981	0.116945	0.116945	0.914795	3.6030e-01
who.PS2	0.739937	0.175651	0.175651	4.212532	2.5252e-05
year:treatmentC	-0.634622	0.183656	0.183656	-3.455500	5.4927e-04

	chisq	df	global	p
age	6.90318	2	0.031700	
who.PS	18.31983	2	0.000105	

Terminal event:

	coef	exp(coef)	SE	coef (H)	SE	coef (HIH)	z
age60-69 years	-0.226278	0.797497	0.243003	0.243003	0.243003	-0.931171	
age>69 years	-0.100919	0.904006	0.223801	0.223801	0.223801	-0.450933	
treatmentC	-0.090393	0.913572	0.198843	0.198843	0.198843	-0.454597	
who.PS1	-0.116788	0.889774	0.218351	0.218351	0.218351	-0.534864	
who.PS2	0.802405	2.230899	0.258303	0.258303	0.258303	3.106452	
prev.resectionYes	-0.225770	0.797902	0.193418	0.193418	0.193418	-1.167264	

	P
age60-69 years	3.5177e-01
age>69 years	6.5204e-01
treatmentC	6.4940e-01
who.PS1	5.9274e-01
who.PS2	1.8935e-03
prev.resectionYes	2.4310e-01

	chisq	df	global	p
age	0.867823	2	0.64800	
who.PS	12.897239	2	0.00158	

Components of Random-effects covariance matrix B1:

Intercept	1.972036	-0.519749
year	-0.519749	0.943576

Association parameters:

	coef	SE	z	p
Intercept	0.3203428	0.0848201	3.776733	0.0001589
year	0.0432225	0.1668878	0.258991	0.7956400

Residual standard error: 0.954238 (SE (H): 0.027079)

```

penalized marginal log-likelihood = -1684.01
Convergence criteria:
parameters = 7.91e-05 likelihood = 3.97e-05 gradient = 1.4e-08

```

```

LCV = the approximate likelihood cross-validation criterion
      in the semi parametrical case      = 1.62312

```

```

n= 150
n repeated measurements= 906
Percentage of left-censored measurements= 3.75 %
Censoring threshold s= -3.33
n events= 121
number of iterations: 17

```

```

Exact number of knots used: 8
Value of the smoothing parameter: 0.93
Gaussian quadrature method: Pseudo-adaptive with 7 nodes

```

On average, the tumor size significantly decreases in time in interaction with the treatment, this effect is more important in the C arm (-0.63 , p value = 0.001). However, there is no effect of treatment arm on the risk of death (HR = 0.91, p value = 0.65). The age of patients at baseline does not have any effect neither on the tumor size nor on the risk of death. The performance status WHO 2 evaluated before the treatment is a prognostic factor both for the tumor size (0.74, p value < 0.001) and overall survival (HR = 2.23, p value = 0.001). It should be noted that the model is fitted on a subset of the original trial in which, using the data of all patients, some of the prognostic effects were found to be different (Król *et al.* 2016).

The processes are linked together via the random intercept and slope of the longitudinal trajectory. This association is significant for the random intercept implying that with the increase of individual deviation from the population average tumor size, the risk of death increases as well ($\hat{\eta}_{t1} = 0.32$, p value < 0.001).

To verify the goodness-of-fit of the model, the estimated baseline hazard function, martingale residuals for the terminal event and residuals for the longitudinal outcome can be plotted using the following code (results not shown here):

```

R> plot(modLongi, main = "Hazard function")
R> plot(aggregate(colorectalSurv$time1, by = list(colorectalSurv$id),
+ FUN = max)[2][, 1], modLongi$martingaleddeath.res, ylab = "",
+ xlab = "time", main = "Martingale Residuals - Death",
+ ylim = c(-4.2, 4.2))
R> lines(lowess(aggregate(colorectalSurv$time1, by = list(colorectalSurv$id),
+ FUN = max)[2][, 1], modLongi$martingaleddeath.res, f = 1), lwd = 3,
+ col = "grey")
R> qqnorm(modLongi$marginal_chol.res, main = "Marginal Cholesky residuals",
+ xlab = "")
R> qqline(modLongi$marginal_chol.res)
R> plot(modLongi$pred.y.cond, modLongi$conditional.res, xlab = "Fitted",

```

```
+   ylab = "Conditional residuals",
+   main = "Conditional Residuals vs. Fitted Values")
```

Next, we explore predictiveness of this model comparing it with a model in which the link function is represented by the current level of the biomarker $m_i(t)$. Thus, we fit a bivariate model with the same covariates and characteristics but with the option `link = "Current-level"` (model `modLongi2`). Then, for times of prediction between 1.2 and 2.5 years we calculate the EPOCE estimator. We compare the models by plotting the values of $CVPOL_a$ and the tracking intervals for the differences between the models.

```
R> modLongi2 <- longiPenal(Surv(time0, time1, state) ~ age + treatment +
+   who.PS + prev.resection, tumor.size ~ year * treatment + age + who.PS,
+   colorectalSurv, data.Longi = colorectalLongi, random = c("1", "year"),
+   id = "id", link = "Current-level", left.censoring = -3.33,
+   n.knots = 8, kappa = 0.93, method.GH = "Pseudo-adaptive", n.nodes = 7)
R> time <- seq(1.2, 2.4, 0.1)
R> epoce <- epoce(modLongi, time)
R> epoce2 <- epoce(modLongi2, time)
R> diff <- Diffepoce(epoce, epoce2)
R> plot(temps, epoce$cvpol, ylab = "CVPOL", xlab = "time", pch = 6,
+   col = "darkcyan", type = "b", ylim = c(0.4, 1.0))
R> points(temps, epoce2$cvpol, pch = 15, col = "brown3", type = "b",
+   lty = 2)
R> legend(1.9, 1, legend = c("modLongi", "modLongi2"), pch = c(6, 15),
+   col = c("darkcyan", "brown3"), bty = "n", cex = 0.7, lty = c(1, 2))
R> plot(diff)
```

The results of $CVPOL_a$ are presented in Figure 6. Model `modLongi` with the random effects as the link function has better predictive abilities than model `modLongi2` with the current level of the biomarker as the link function until 1.9 year. After this time point, the tendency inverses. However, these difference are significant only at 2.4 years of treatment. Using method `AtRisk` for ‘`epoce`’ objects, we may verify for how many subjects at risk, $CVPOL_a$ is calculated at this prediction time (`epoce$AtRisk`). The significant difference between the models is found for the time point for which the number of subjects considered is relatively small (25). Thus, we can conclude that the models are close to each other and the choice of the link function does not strongly influence the predictiveness.

The comparison of the predictive abilities of bivariate models can be useful for the choice of the appropriate trivariate model. If we are interested in the prediction of death it is important to choose a model that provides the best predictive abilities for survival. For this reason, we choose a model with such a structure of the link function that provides the best predictive abilities, for both bivariate and trivariate models. First, we fit appropriate bivariate models (without the recurrent part) with different link structures and compare their predictive abilities. Then, we fit the trivariate model with the same link structure as in the bivariate model with the best predictive abilities. For the `colorectal` dataset the differences in $CVPOL_a$ are not of great importance, thus we apply a trivariate model with the random effects of the biomarker as the link function, as this model is less computationally intensive.

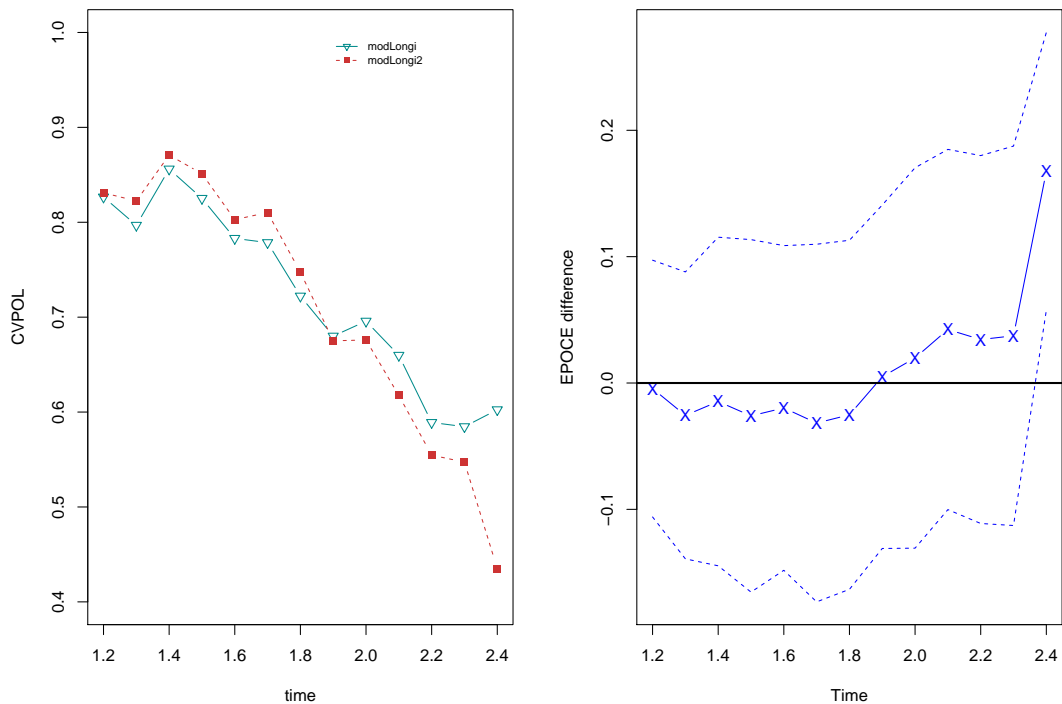


Figure 6: Estimated cross-validated EPOCE computed from the joint models until 2.5 years (left plot) and differences in the estimates with 95% tracking intervals (right plot).

Joint model for longitudinal data, recurrent events and a terminal event

The package **frailtypack** allows the estimation of the trivariate models for a longitudinal biomarker, recurrent events and a terminal event. In the dataset **colorectal**, the recurrent events are represented by the appearance of new lesions during the treatment. Usually in clinical trials the size of new lesions is not registered and thus, their burden cannot be simply added to the measure of the tumor size of the target lesions. However, it is of interest to add the information on new lesions to a model as it influences overall survival. A trivariate model can be a solution for such a goal and can be implemented using function **trivPenal**.

We fit a model with the calendar timescale for recurrent events and the baseline hazard functions approximated by splines with 8 knots. The smoothing parameter values are found from the separate models (shared frailty model and Cox model). A random intercept and a random slope and left-censoring (threshold $s = -3.33$) are assumed for the biomarker. The pseudo-adaptive Gaussian quadrature with 9 nodes is chosen for calculation of integrals.

Firstly, we find initial values for the covariates. For the longitudinal outcome and the terminal event we use the estimates from the bivariate model, **modLongi**. The covariates related to the recurrent events are initialized using the results of a shared frailty model for the appearance of new lesions:

```
R> modShared <- frailtyPenal(Surv(time0, time1, new.lesions) ~ cluster(id) +
+   age + treatment + who.PS, data = colorectal, recurrentAG = TRUE,
+   n.knots = 8, cross.validation = TRUE, kappa = 1000, RandDist = "LogN")
```

Then, using argument **init.B** we fit the trivariate model with appropriate initial values.

In `init.B`, the vector of initial values must follow the order: covariates for the recurrent events, terminal event and then biomarker (interactions at the end of each component). The trivariate model due to its complexity is computationally intensive but using the pseudo-adaptive quadrature method the time of estimation is reduced (40 minutes on a personal computer with an Intel Core i7 3.40 GHz processor and 8 GB RAM running Windows 7). The model is defined as follows:

```
R> modTrivariate <- trivPenal(Surv(time0, time1, new.lesions) ~ cluster(id) +
+   age + treatment + who.PS + terminal(state), formula.terminalEvent = ~
+   age + treatment + who.PS + prev.resection, tumor.size ~ year *
+   treatment + age + who.PS, data = colorectal, data.Longi =
+   colorectalLongi, random = c("1", "year"), id = "id", link =
+   "Random-effects", left.censoring = -3.33, recurrentAG = T, n.knots = 7,
+   kappa = c(0.01, 0.7), method.GH = "Pseudo-adaptive", n.nodes = 7,
+   init.B = c(-0.18, -0.22, -0.24, -0.22, 0.35, -0.23, -0.10, -0.09, -0.12,
+   0.80, -0.23, 3.02, -0.30, 0.05, -0.02, -0.29, 0.11, 0.74, -0.63))
R> modTrivariate
```

Call:

```
trivPenal(formula = Surv(time0, time1, new.lesions) ~ cluster(id) +
  age + treatment + who.PS + terminal(state), formula.terminalEvent =
  ~age + treatment + who.PS + prev.resection, formula.LongitudinalData =
  tumor.size ~ year * treatment + age + who.PS, data = colorectal,
  data.Longi = colorectalLongi, random = c("1", "year"), id = "id",
  link = "Random-effects", left.censoring = -3.33, recurrentAG = TRUE,
  n.knots = 7, kappa = c(0.01, 0.7), init.B = c(-0.18, -0.22, -0.24, -0.22,
    0.35, -0.23, -0.1, -0.09, -0.12, 0.8, -0.23, 3.02, -0.3,
    0.05, -0.02, -0.29, 0.11, 0.74, -0.63),
  method.GH = "Pseudo-adaptive", n.nodes = 7)
```

Calendar timescale

Trivariate Joint Model for Longitudinal Data, Recurrent Events and a Terminal Event Parameter estimates using a Penalized Likelihood on the hazard functions and assuming left-censored longitudinal outcome
Association function: random effects

Longitudinal outcome:

```
-----
              coef SE coef (H) SE coef (HIH)          z          p
Intercept      2.958884  0.187694    0.186822 15.764364    <1e-16
year           -0.282884  0.134833    0.134692 -2.098035  3.5902e-02
treatmentC     0.102857  0.215248    0.214932  0.477853  6.3275e-01
age60-69 years  0.013893  0.167638    0.167392  0.082874  9.3395e-01
age>69 years   -0.272310  0.131498    0.131356 -2.070835  3.8374e-02
who.PS1        0.120872  0.116505    0.116452  1.037480  2.9951e-01
```

who.PS2	0.760518	0.175385	0.175368	4.336273	1.4492e-05
year:treatmentC	-0.664321	0.185322	0.185215	-3.584687	3.3748e-04

	chisq	df	global	p
age	6.49994	2	0.038800	
who.PS	19.25227	2	0.000066	

Recurrences:

	coef	exp(coef)	SE coef	(H)	SE coef	(HIH)	z
age60-69 years	-0.272822	0.761228	0.272202	0.268207	-1.002279		
age>69 years	-0.298435	0.741979	0.250761	0.249414	-1.190117		
treatmentC	-0.290762	0.747693	0.219868	0.216165	-1.322440		
who.PS1	-0.008178	0.991856	0.263673	0.257756	-0.031015		
who.PS2	0.690831	1.995374	0.284015	0.279991	2.432378		

	P
age60-69 years	3.1621e-01
age>69 years	2.3400e-01
treatmentC	1.8602e-01
who.PS1	9.7526e-01
who.PS2	1.5000e-02

	chisq	df	global	p
age	1.74210	2	0.419	
who.PS	6.70329	2	0.035	

Terminal event:

	coef	exp(coef)	SE coef	(H)	SE coef	(HIH)	z
age60-69 years	-0.251916	0.777310	0.543611	0.522412	-0.463413		
age>69 years	-0.320350	0.725895	0.468031	0.461305	-0.684463		
treatmentC	-0.104732	0.900566	0.409034	0.389566	-0.256046		
who.PS1	0.521182	1.684016	0.598282	0.574802	0.871131		
who.PS2	2.154820	8.626335	0.632457	0.608890	3.407060		
prev.resectionYes	-0.071743	0.930770	0.390296	0.379218	-0.183818		

	P
age60-69 years	6.4307e-01
age>69 years	4.9368e-01
treatmentC	7.9792e-01
who.PS1	3.8368e-01
who.PS2	6.5667e-04
prev.resectionYes	8.5416e-01

	chisq	df	global	p
age	0.509673	2	0.775000	
who.PS	14.065163	2	0.000883	

Components of Random-effects covariance matrix B1:

```
Intercept  1.983247 -0.526577
year       -0.526577  0.955918
```

Recurrent event and longitudinal outcome association:

```
          coef      SE      z      p
Asso:Intercept  0.1367926  0.0965283  1.417125  0.15645
Asso:year       -0.0673937  0.1339954 -0.502955  0.61500
```

Terminal event and longitudinal outcome association:

```
          coef      SE      z      p
Asso:Intercept  0.8456297  0.289808  2.917896  0.003524
Asso:year       -0.0574471  0.234541 -0.244934  0.806510
```

Residual standard error: 0.139813 (SE (H): 0.026971)

Frailty parameter for the association between recurrent events and terminal event:

```
sigma square (variance of Frailties): 0.555577 (SE (H): 0.405782)
                                         p = 0.085476
alpha (for terminal event): 2.62514 (SE (H): 0.250761 ) p = <1e-16
```

penalized marginal log-likelihood = -1789.36

Convergence criteria:

parameters = 0.000107 likelihood = 0.000626 gradient = 6.33e-06

LCV = the approximate likelihood cross-validation criterion
in the semi parametric case = 1.53133

n subjects= 150

n repeated measurements= 906

Percentage of left-censored measurements= 3.75 %

Censoring threshold s= -3.33

n recurrent events= 139

n terminal events= 121

number of iterations: 20

Exact number of knots used: 7

Value of the smoothing parameters: 0.01 0.7

Gaussian quadrature method: Pseudo-adaptive with 7 nodes

In the output of the fitted model, we obtain the estimates for the covariates related to the

three types of processes: longitudinal outcome, recurrences and terminal event. The treatment effect is not significant neither for the risk of appearance of new lesions nor for the risk of death. As in the bivariate model, the tumor size decreases on average more in the treatment arm C (-0.66 , p value < 0.001). We find the baseline WHO performance status 2 having a significant effect on the average population biomarker, the risk of new lesions appearance and death. The coefficients for the association between the biomarker and the recurrences are found not significant and for the link between the biomarker and death only the coefficient related to the random intercept is significantly different from 0 (0.85 , p value $= 0.004$). The variance of the frailty term is found to be not significant.

As for the bivariate model `modLongi`, goodness-of-fit of the trivariate model can be evaluated using the martingale residuals for the recurrent and terminal events and the residuals related to the biomarker. For this purpose, the following code can be used (Figure 11, Appendix B):

```
R> plot(modTrivariate, main = "Hazard functions")
R> plot(modTrivariate, type = "Survival", main = "Survival functions")
R> plot(aggregate(colorectal$time1, by = list(colorectal$id),
+ FUN = max)[2][, 1], modTrivariate$martingaledeath.res, ylab = "",
+ xlab = "time", main = "Martingale Residuals - Death",
+ ylim = c(-4.2, 4.2))
R> lines(lowess(aggregate(colorectal$time1, by = list(colorectal$id),
+ FUN = max)[2][, 1], modTrivariate$martingaledeath.res, f = 1), lwd = 3,
+ col = "grey")
R> plot(aggregate(colorectal$time1, by = list(colorectal$id),
+ FUN = max)[2][, 1], modTrivariate$martingale.res, ylab = "",
+ xlab = "time", main = "Martingale Residuals\n - Recurrences",
+ ylim = c(-4.2, 4.2))
R> lines(lowess(aggregate(colorectal$time1, by = list(colorectal$id),
+ FUN = max)[2][, 1], modTrivariate$martingale.res, f = 1), lwd = 3,
+ col = "grey")
R> qqnorm(modTrivariate$marginal_chol.res,
+ main = "Marginal Cholesky residuals" , xlab = "")
R> qqline(modTrivariate$marginal_chol.res)
R> plot(modTrivariate$pred.y.cond, modTrivariate$conditional.res,
+ xlab = "Fitted", ylab = "Conditional residuals",
+ main = "Conditional Residuals \n vs. Fitted Values")
```

The martingale residuals for both recurrences and death processes show slight signs of skewness, their means seem to decrease in time. Using the marginal Cholesky residuals and the conditional residuals we find that the model does not fit the longitudinal data very well.

We compare the trivariate model `modTrivariate` with the bivariate model `modLongi` in terms of predictive accuracy for OS using the Brier score. Firstly, we define the prediction time and horizons and calculate the predictions for the bivariate and trivariate models:

```
R> predtime <- 1.0
R> window <- seq(0.1, 1.5, 0.1)
R> fwindow <- predtime + window
R> pred_bivariate <- prediction(modLongi, data = colorectalSurv,
```

```
+ data.Longi = colorectalLongi, predtime, window)
R> pred_trivariate <- prediction(modTrivariate, data = colorectal,
+ data.Longi = colorectalLongi, predtime, window)
```

We prepare data for the Brier score computations: `data_surv` including data on survival (variables of time and state), `survj_biv` and `survj_tri` for predicted probabilities of survival of subjects that are alive at time of prediction using the bivariate and trivariate model, respectively:

```
R> data_surv <- colorectalSurv[, c(1, 3, 9)]
R> predictions_bivariate <- as.matrix(pred_bivariate$pred)
R> survj_biv <-
+ as.matrix(predictions_bivariate[data_surv$time1 >= predtime, ])
R> survj_biv <- 1 - cbind(0, survj_biv)
R> predictions_trivariate <- as.matrix(pred_trivariate$pred)
R> survj_tri <-
+ as.matrix(predictions_trivariate[data_surv$time1 >= predtime, ])
R> survj_tri <- 1 - cbind(0, survj_tri)
```

Prediction error curves can be obtained by calculating the Brier score for several time points. Using the joint model estimated with **frailtypack**, the Brier score can be computed using the package **pec** (version 2.4.9, available at <https://CRAN.R-project.org/src/contrib/Archive/pec/>) and its modified function `pec` included in the supplementary material. The following code is used to calculate and plot the Brier score for the bivariate and trivariate models:

```
R> library("pec")
R> library("prodlim")
R> source("ipcw_2-9_modif.R")
R> source("pecMethods_2-9_modif.R")
R> BrierScore <- pec(list("Bivariate" = survj_biv, "Trivariate" = survj_tri),
+ formula = Surv(time1, state) ~ 1,
+ data = data_surv[data_surv$time1 >= predtime, ],
+ cens.model = "marginal", data.cens = data_surv, exact = FALSE,
+ times = fwindow, ptime = predtime, reference = FALSE)
R> plot(fwindow, BrierScore$AppErr$Bivariate[-1], pch = 6, main = "",
+ ylab = "Prediction error", xlab = "Years", ylim = c(0, 0.3),
+ xlim = c(0.9, 2.7), col = "black", type = "l", lwd = 2, axes = FALSE,
+ lty = 1)
R> points(fwindow, BrierScore$AppErr$Trivairate[-1], pch = 15, col = "blue",
+ type = "l", lwd = 2, lty = 2)
R> legend(2.1, 0.3, legend = c("modLongi", "modTrivariate"), bty = "n",
+ cex = 1.1, lty = c(1, 2), lwd = 2, col = c("black", "blue"))
R> axis(1, at = seq(0.9, 2.7, by = 0.3))
R> axis(2, at = seq(0, 0.3, by = 0.1))
R> abline(v = 1, lty = 2, lwd = 2, col = "gray35")
```

Figure 7 represents the prediction error curves for both models. In short prediction horizons, values of the Brier score of the models are very close to each other. After around 1.6 years

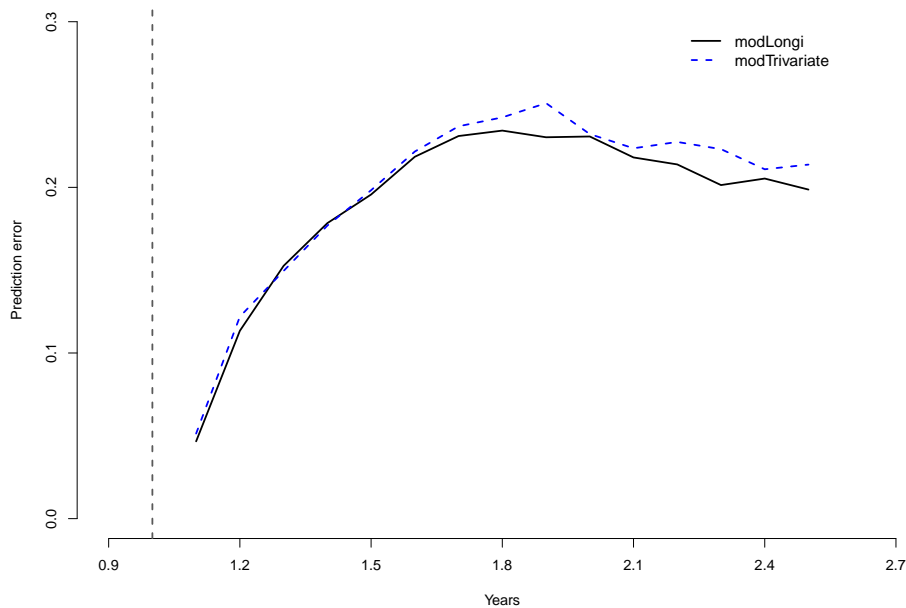


Figure 7: Error of prediction at $t = 1$ year and varying window w from 0.1 to 1.5. `modLongi` – tumor size and death, `modTrivariate` – tumor size, occurrence of new lesions and death.

the bivariate model has better predictive accuracy than the trivariate model. However, the results are presented for the apparent error calculated using the data from the estimation. For verification of the models for predictions for new patients (not used in the data for the estimation) a cross-validation procedure would be required.

Dynamic predictions

Both, for the bivariate models for longitudinal data and a terminal event and for the trivariate models, the package provides individual predictions of the terminal event. To create patients' profiles, two datasets must be provided, one including the history of the biomarker and the values of covariates at measurement times (`dataLongi`) and a second one with the information on covariates related to the terminal event (`data`). In case of the trivariate model, this dataset should also include the history of recurrences.

For the example of the `colorectal` dataset, we create two profiles of patients that differ from each other by the trajectory of the tumor size. The first patient has a progressive disease with a tumor size increasing in time and the second patient a response to the treatment with a diminishing tumor. Both patients have the same baseline characteristics. In the data creation step, the types of variables must be appropriately defined and coherent with the dataset used for estimation. Firstly, we prepare the data for the bivariate model setting:

```
R> datapredj_longi <- data.frame(id = 0, year = 0, tumor.size = 0,
+   treatment = 0, age = 0, who.PS = 0, prev.resection = 0)
R> datapredj_longi$treatment <- factor(datapredj_longi$treatment,
+   levels = 1:2)
R> datapredj_longi$age <- factor(datapredj_longi$age, levels = 1:3)
R> datapredj_longi$who.PS <- factor(datapredj_longi$who.PS, levels = 1:3)
```

```
R> datapredj_longi$prev.resection <- factor(datapredj_longi$prev.resection,
+     levels = 1:2)
```

For Patient 1 we assume 5 measurements indicating increasing tumor burden:

```
R> datapredj_longi[1, ] <- c(1, 0, 1.2, 2, 1, 1, 1)
R> datapredj_longi[2, ] <- c(1, 0.3, 1.1, 2, 1, 1, 1)
R> datapredj_longi[3, ] <- c(1, 0.6, 1.4, 2, 1, 1, 1)
R> datapredj_longi[4, ] <- c(1, 0.9, 2.2, 2, 1, 1, 1)
R> datapredj_longi[5, ] <- c(1, 1.5, 3.0, 2, 1, 1, 1)
```

On the contrary, Patient 2 is assumed to have a decreasing size of the tumor:

```
R> datapredj_longi[6, ] <- c(2, 0, 1.2, 2, 1, 1, 1)
R> datapredj_longi[7, ] <- c(2, 0.3, 1.1, 2, 1, 1, 1)
R> datapredj_longi[8, ] <- c(2, 0.5, 0.7, 2, 1, 1, 1)
R> datapredj_longi[9, ] <- c(2, 0.7, 0.3, 2, 1, 1, 1)
R> datapredj_longi[10, ] <- c(2, 0.9, 0.1, 2, 1, 1, 1)
```

Next, for both patients we prepare the data with information on covariates included in the survival part in the bivariate model:

```
R> datapredj <- data.frame(id = 0, treatment = 0, age = 0, who.PS = 0,
+     prev.resection = 0)
R> datapredj$treatment <- factor(datapredj$treatment, levels = 1:2)
R> datapredj$age <- factor(datapredj$age, levels = 1:3)
R> datapredj$who.PS <- factor(datapredj$who.PS, levels = 1:3)
R> datapredj$prev.resection <- factor(datapredj$prev.resection, levels = 1:2)
R> datapredj[1, ] <- c(1, 2, 1, 1, 1)
R> datapredj[2, ] <- c(2, 2, 1, 1, 1)
```

We calculate the estimated probabilities of the terminal event given that the patients are alive at time of prediction 1 year and a horizon varying from 0.5 to 2.5 years. We compare the predicted risk of death for the patients by plotting and smoothing the estimations. Additionally, the 95% MC confidence intervals are calculated in order to facilitate the interpretation.

```
R> pred.joint <- prediction(modLongi, datapredj, datapredj_longi, 1.0,
+     seq(0.5, 2.5, 0.2), MC.sample = 500)
R> plot(pred.joint, conf.bands = TRUE)
```

The left graph in Figure 8 presents the dynamic predictions for the patients. The patient with a decreasing tumor size (profile 2) has lower probability of death than the patient with the tumor size that increases during the treatment (profile 1). However, considering the MC confidence intervals, this difference is not significant. Thus, in the analyzed example, the biomarker itself does not influence the risk of death significantly. It is of interest if addition of the history of the recurrent event would increase the difference between the profiles. For this purpose, we modify the data `datapredj` by adding the history of recurrences and calculate the analogous dynamic predictions using the trivariate model `modTrivariate`. We assume that patient 1 experiences the occurrence of new lesions twice and patient 2 only once.

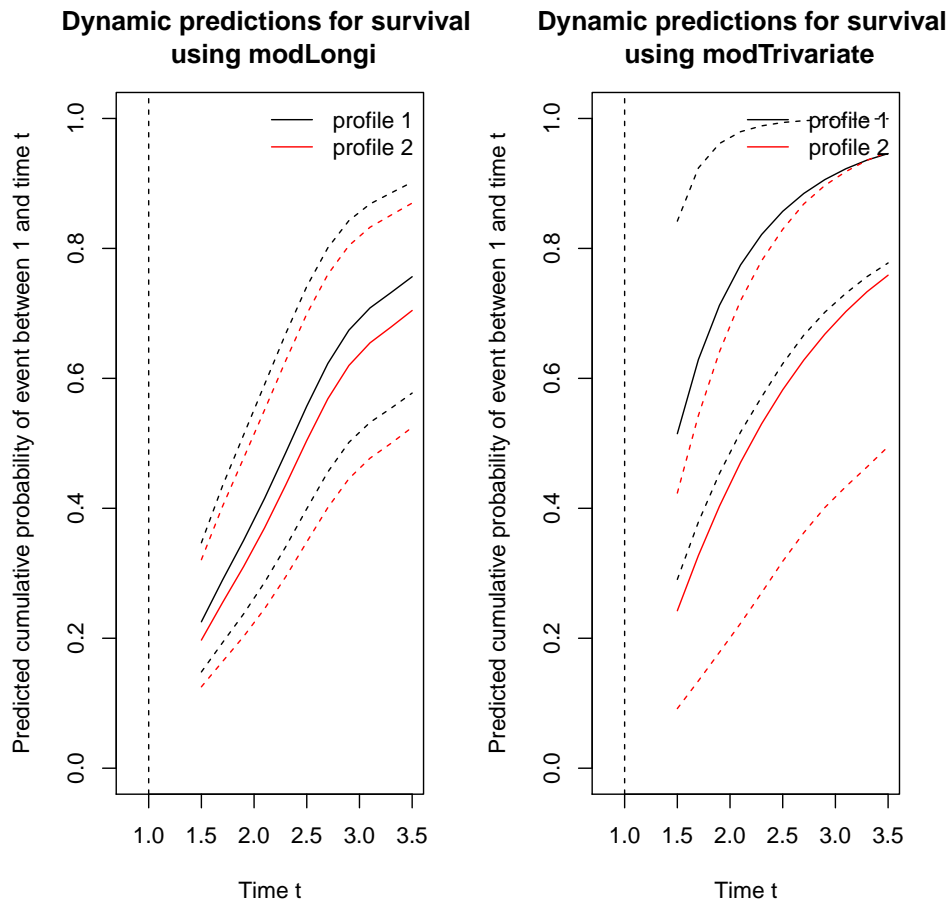


Figure 8: Predicted probabilities of death for two patients sharing the same characteristics and different history using information from the biomarker only (on the left) and the biomarker and recurrent events (on the right). The prediction time is set to 1 year and the horizon changed from 0.5 to 2.5 years. Dashed lines represent the MC confidence intervals.

```
R> datapredj <- data.frame(time0 = 0, time1 = 0, new.lesions = 0, id = 0,
+   treatment = 0, age = 0, who.PS = 0, prev.resection = 0)
R> datapredj$treatment <- factor(datapredj$treatment, levels = 1:2)
R> datapredj$age <- factor(datapredj$age, levels = 1:3)
R> datapredj$who.PS <- factor(datapredj$who.PS, levels = 1:3)
R> datapredj$prev.resection <- factor(datapredj$prev.resection,
+   levels = 1:2)
R> datapredj[1, ] <- c(0, 0.4, 1, 1, 2, 1, 1, 1)
R> datapredj[2, ] <- c(0.4, 1.2, 1, 1, 2, 1, 1, 1)
R> datapredj[3, ] <- c(0, 0.5, 1, 2, 2, 1, 1, 1)
```

Then, we calculate the predictions and plot the results with the 95% MC confidence intervals:

```
R> pred.joint2 <- prediction(modTrivariate, datapredj, datapredj_longi, 1.0,
+   seq(0.5, 2.5, 0.2), MC.sample = 500)
R> plot(pred.joint2, conf.bands = TRUE)
```

The right graph in Figure 8 shows the dynamic predictions of death using the trivariate model. As in the case of the bivariate model, patient 1 has an increased probability of death compared to patient 1. Moreover, the difference between the patients is more accentuated than in the bivariate model which is not able to include information on the history of recurrences. By considering the information on the appearance of new lesions, the predicted probabilities are influenced, although the differences between the patients are not significant according to the MC confidence intervals.

6. Conclusions

Joint models are now a well recognized statistical tool for a complex analysis of correlated data. They provide unbiased estimates compared to the univariate models. The package **frailtypack** provides several functions for applications of joint models for a longitudinal outcome, recurrent events and a terminal event. Methods of estimation applied in the package produce reliable results proven by simulations for all the implemented models (Rondeau *et al.* 2007, Mazroui *et al.* 2012, Król *et al.* 2016). Moreover, each function is furnished with several options for better adjustment of a model and accompanied by tools helping with the diagnostic of a model (residuals, individual predictions of random effects) and comparison with other models (LCV, AIC).

The increasing interest in individual predictions of death in the clinical perspective motivated the implementation of model-based dynamic predictions. Function **prediction** allows the users to calculate the estimated probability of death given the history of a patient. The history, depending on a model, can be a part of the information on observed recurrences, complete information on observed recurrences, complete past measurements of a biomarker or the entire available history of the recurrences and biomarker. Graphical representation of the predictions is a useful tool for comparing probability of death of patients, e.g., with the same characteristics but different past of recurrences and/or biomarker.

Further developments of the **frailtypack** package will concern the extensions of the existent functions. These extensions will be related to the random effects (distributions, number of random effects for the longitudinal part), association functions for models with a biomarker (provide more forms to choose from), stratification (increase the number of strata) and dynamic predictions for the joint model with two types of recurrent events and a terminal event. Moreover, in order to increase the variety of possible application, it will be of interest to implement left truncation and interval-censoring in the proposed functions.

In the near future, we will develop dynamic predictions of a recurrent event. The conditional probability of a new recurrence occurring in a finite time horizon given the history of the observed events and/or the biomarker and given that an individual is alive at the prediction time, will constitute a useful tool for clinicians, e.g., to explore the chances of developing a new relapse given a patient's characteristics.

Acknowledgments

The authors thank the Fédération Francophone de Cancérologie Digestive and Gustave Roussy for sharing the data of the FFCD 2000-05 trial supported by an unrestricted Grant from Sanofi. The authors thank as well the IReSP for financing the project AAP 2012 Plan Cancer

MULTIPLE, the Institut National du Cancer for funding the research of A. Mauguén and the École des Hautes Études en Santé Publique for funding the research of A. Król.

References

- Andersen PK, Gill RD (1982). “Cox’s Regression Model for Counting Processes: A Large Sample Study.” *The Annals of Statistics*, **10**(4), 1100–1120.
- Asar O, Ritchie J, Kalra PA, Diggle P (2015). “Joint Modelling of Repeated Measurement and Time-to-Event Data: An Introductory Tutorial.” *International Journal of Epidemiology*, **44**(1), 334–344.
- Belot A, Rondeau V, Remontet L, Roch G, CENSUR Working Survival Group (2014). “A Joint Frailty Model to Estimate the Recurrence Process and the Disease-Specific Mortality Process without Needing the Cause of Death.” *Statistics in Medicine*, **33**(18), 3147–3166. doi:10.1002/sim.6140.
- Blanche P, Proust-Lima C, Loubère L, Berr C, Dartigues JF, Jacqmin-Gadda H (2015). “Quantifying and Comparing Dynamic Predictive Accuracy of Joint Models for Longitudinal Marker and Time-to-Event in Presence of Censoring and Competing Risks.” *Biometrics*, **71**(1), 102–113. doi:10.1111/biom.12232.
- Chi YY, Ibrahim JG (2006). “Joint Models for Multivariate Longitudinal and Multivariate Survival Data.” *Biometrics*, **62**(2), 432–445. doi:10.1111/j.1541-0420.2005.00448.x.
- Commenges D, Liqueur B, Proust-Lima C (2012). “Choice of Prognostic Estimators in Joint Models by Estimating Differences of Expected Conditional Kullback-Leibler Risks.” *Biometrics*, **68**(2), 380–387. doi:10.1111/j.1541-0420.2012.01753.x.
- Commenges D, Rondeau V (2000). “Standardized Martingale Residuals Applied to Grouped Left Truncated Observations of Dementia Cases.” *Lifetime Data Analysis*, **6**(3), 229–235. doi:10.1023/a:1009637608473.
- Crowther MJ (2013). *stjm: Stata Module to Fit Shared Parameter Joint Models of Longitudinal and Survival Data*. URL <http://econpapers.repec.org/software/bocbocode/s457502.htm>.
- De Boor C (2001). *A Practical Guide to Splines*. Springer-Verlag, Berlin.
- Duchateau L, Janssen P, Kezic I, Fortpied C (2003). “Evolution of Recurrent Asthma Event Rate over Time.” *Journal of the Royal Statistical Society C*, **52**(3), 355–363. doi:10.1111/1467-9876.00409.
- Ducreux M, Malka D, Mendiboure J, Etienne PL, Texereau P, Auby D, Rougier P, Gasmi M, Castaing M, Abbas M, Michel P, Gargot D, Azzedine A, Lombard-Bohas C, Geoffroy P, Denis B, Pignon JP, Bedenne L, Bouché O (2011). “Sequential Versus Combination Chemotherapy for the Treatment of Advanced Colorectal Cancer (FFCD 2000-05): An Open-Label, Randomised, Phase 3 Trial.” *The Lancet Oncology*, **12**(11), 1032–1044. doi:10.1016/s1470-2045(11)70199-1.

- Efendi A, Molenberghs G, Njagi EN, Dendale P (2013). “A Joint Model for Longitudinal Continuous and Time-to-Event Outcomes with Direct Marginal Interpretation.” *Biometrical Journal*, **55**(4), 572–588. doi:10.1002/bimj.201200159.
- Elashoff RM, Li G, Li N (2008). “A Joint Model for Longitudinal Measurements and Survival Data in the Presence of Multiple Failure Types.” *Biometrics*, **64**(3), 762–771. doi:10.1111/j.1541-0420.2007.00952.x.
- Emura T (2017). **joint.Cox**: *Penalized Likelihood Estimation under the Joint Cox Models Between TTP and OS for Meta-Analysis*. R package version 2.12, URL <https://CRAN.R-project.org/package=joint.Cox>.
- Genz A, Keister BD (1996). “Fully Symmetric Interpolatory Rules for Multiple Integrals over Infinite Regions with Gaussian Weight.” *Journal of Computational and Applied Mathematics*, **71**(2), 299–309. doi:10.1016/0377-0427(95)00232-4.
- Gerds TA (2016). **pec**: *Prediction Error Curves for Risk Prediction Models in Survival Analysis*. R package version 2.4.9, URL <https://CRAN.R-project.org/package=pec>.
- Gerds TA, Schumacher M (2006). “Consistent Estimation of the Expected Brier Score in General Survival Models with Right-Censored Event Times.” *Biometrical Journal*, **48**(6), 1029–1040. doi:10.1002/bimj.200610301.
- Gonzalez JR, Fernandez E, Moreno V, Ribes J, Merce P, Navarro M, Cambray M, Borrás JM (2005). “Sex Differences in Hospital Readmission among Colorectal Cancer Patients.” *Journal of Epidemiology and Community Health*, **59**(6), 506–511. doi:10.1136/jech.2004.028902.
- Hatfield LA, Boye ME, Hackshaw MD, Carlin BP (2012). “Multilevel Bayesian Models for Survival Times and Longitudinal Patient-Reported Outcomes with Many Zeros.” *Journal of the American Statistical Association*, **107**(499), 875–885. doi:10.1080/01621459.2012.664517.
- Joly P, Commenges D, Lettenneur L (1998). “A Penalized Likelihood Approach for Arbitrarily Censored and Truncated Data: Application to Age-Specific Incidence of Dementia.” *Biometrics*, **54**(1), 185–194. doi:10.2307/2534006.
- Kim S, Zeng D, Chambless L, Li Y (2012). “Joint Models of Longitudinal Data and Recurrent Events with Informative Terminal Event.” *Statistics in Bioscience*, **4**(2), 262–281. doi:10.1007/s12561-012-9061-x.
- Król A, Ferrer L, Pignon JP, Proust-Lima C, Ducreux M, Bouché O, Michiels S, Rondeau V (2016). “Joint Model for Left-Censored Longitudinal Data, Recurrent Events and Terminal Event: Predictive Abilities of Tumor Burden for Cancer Evolution with Application to the FFCD 2000-05 Trial.” *Biometrics*, **72**(3), 907–916. doi:10.1111/biom.12490.
- Lawrence Gould A, Boye ME, Crowther MJ, Ibrahim JG, Quartey G, Micallef S, Bois FY (2015). “Joint Modeling of Survival and Longitudinal Non-Survival Data: Current Methods and Issues. Report of the DIA Bayesian Joint Modeling Working Group.” *Statistics in Medicine*, **34**(14), 2181–2195. doi:10.1002/sim.6141.

- Li QH, Lagakos SW (1997). “Use of the Wei-Lin-Weissfeld Method for the Analysis of a Recurring and a Terminating Event.” *Statistics in Medicine*, **16**(8), 925–940. doi:10.1002/(sici)1097-0258(19970430)16:8<925::aid-sim545>3.3.co;2-u.
- Liu L, Huang X (2009). “Joint Analysis of Correlated Repeated Measures and Recurrent Events Processes in the Presence of Death, with Application to a Study on Acquired Immune Deficiency Syndrome.” *Journal of the Royal Statistical Society C*, **58**(1), 65–81. doi:10.1111/j.1467-9876.2008.00641.x.
- Liu L, Huang X, O’Quigley JM (2008). “Analysis of Longitudinal Data in the Presence of Informative Observational Times and a Dependent Terminal Event, with Application to Medical Cost Data.” *Biometrics*, **64**(3), 950–958. doi:10.1111/j.1541-0420.2007.00954.x.
- Liu L, Wolfe RA, Huang X (2004). “Shared Frailty Models for Recurrent Events and a Terminal Event.” *Biometrics*, **60**(3), 747–756. doi:10.1111/j.0006-341x.2004.00225.x.
- Marquardt DW (1963). “An Algorithm for Least-Squares Estimation of Nonlinear Parameters.” *Journal of the Society for Industrial and Applied Mathematics*, **11**(2), 431–441. doi:10.1137/0111030.
- Mauguen A, Rachet B, Mathoulin-Pélissier S, MacGrogan G, Laurent A, Rondeau V (2013). “Dynamic Prediction of Risk of Death Using History of Cancer Recurrences in Joint Frailty Models.” *Statistics in Medicine*, **32**(30), 5366–5380. doi:10.1002/sim.5980.
- Mazroui Y, Mathoulin-Pélissier S, Macgrogan G, Brouste V, Rondeau V (2013). “Multivariate Frailty Models for Two Types of Recurrent Events with a Dependent Terminal Event: Application to Breast Cancer Data.” *Biometrical Journal*, **55**(6), 866–884. doi:10.1002/bimj.201200196.
- Mazroui Y, Mathoulin-Pélissier S, Soubeyran P, Rondeau V (2012). “General Joint Frailty Model for Recurrent Event Data with a Dependent Terminal Event: Application to Follicular Lymphoma Data.” *Statistics in Medicine*, **31**(11–12), 1162–1176. doi:10.1002/sim.4479.
- Mazroui Y, Mauguen A, Macgrogan G, Mathoulin-Pélissier S, Brouste V, Rondeau V (2016). “Time-Varying Coefficients in a Multivariate Frailty Model: Application to Breast Cancer Recurrences of Several Types and Death.” *Lifetime Data Analysis*, **22**(2), 191–215. doi:10.1007/s10985-015-9327-y.
- Mbogning C, Bleakley K, Lavielle M (2015). “Joint Modeling of Longitudinal and Repeated Time-to-Event Data Using Nonlinear Mixed-Effects Models and the SAEM Algorithm.” *Journal of Statistical Computation and Simulation*, **85**(8), 1512–1528. doi:10.1080/00949655.2013.878938.
- McCrink LM, Marshall AH, Cairns KJ (2013). “Advances in Joint Modelling: A Review of Recent Developments with Application to the Survival of End Stage Renal Disease Patients.” *International Statistical Review*, **81**(2), 249–269. doi:10.1111/insr.12018.
- Philipson P, Sousa I, Diggle P, Williamson P, Kolamunnage-Dona R, Henderson R (2017). **joineR**: *Joint Modelling of Repeated Measurements and Time-to-Event Data*. R package version 1.2.1, URL <https://CRAN.R-project.org/package=joineR>.

- Proust-Lima C, Philipps V, Diakite A, Liqueur B (2017a). **lcmm**: *Extended Mixed Models Using Latent Classes and Latent Processes*. R package version 1.7.8, URL <https://CRAN.R-project.org/package=lcmm>.
- Proust-Lima C, Philipps V, Liqueur B (2017b). “Estimation of Extended Mixed Models Using Latent Classes and Latent Processes: The R Package **lcmm**.” *Journal of Statistical Software*, **78**(2), 1–56. doi:10.18637/jss.v078.i02.
- Proust-Lima C, Sène M, Taylor JMG, Jacqmin-Gadda H (2014). “Joint Latent Class Models for Longitudinal and Time-to-Event Data: A Review.” *Statistical Methods in Medical Research*, **23**(1), 74–90. doi:10.1177/0962280212445839.
- Proust-Lima C, Taylor JMG (2009). “Development and Validation of a Dynamic Prognostic Tool for Prostate Cancer Recurrence Using Repeated Measures of Posttreatment PSA: A Joint Modeling Approach.” *Biostatistics*, **10**(3), 535–549.
- Ramsay JO (1988). “Monotone Regression Splines in Action.” *Statistical Science*, **3**(4), 425–461.
- R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
- Rizopoulos D (2010). “**JM**: An R Package for the Joint Modelling of Longitudinal and Time-to-Event Data.” *Journal of Statistical Software*, **35**(9), 1–33. doi:10.18637/jss.v035.i09.
- Rizopoulos D (2011). “Dynamic Predictions and Prospective Accuracy in Joint Models for Longitudinal and Time-to-Event Data.” *Biometrics*, **67**(3), 819–829. doi:10.1111/j.1541-0420.2010.01546.x.
- Rizopoulos D (2012). *Joint Models for Longitudinal and Time-to-Event Data: With Applications in R*. Chapman & Hall/CRC, Boca Raton.
- Rizopoulos D (2016a). **JMbayes**: *Joint Modeling of Longitudinal and Time-to-Event Data under a Bayesian Approach*. R package version 0.8-0, URL <https://CRAN.R-project.org/package=JMbayes>.
- Rizopoulos D (2016b). “The R Package **JMbayes** for Fitting Joint Models for Longitudinal and Time-to-Event Data Using MCMC.” *Journal of Statistical Software*, **72**(7), 1–45. doi:10.18637/jss.v072.i07.
- Rizopoulos D (2017). **JM**: *Joint Modeling of Longitudinal and Survival Data*. R package version 1.4-7, URL <https://CRAN.R-project.org/package=JM>.
- Rondeau V, Gonzalez JR (2005). “**frailtypack**: A Computer Program for the Analysis of Correlated Failure Time Data Using Penalized Likelihood Estimation.” *Computer Methods and Programs in Biomedicine*, **80**(2), 154–164. doi:10.1016/j.cmpb.2005.06.010.
- Rondeau V, Gonzalez JR, Mazroui Y, Mauguen A, Krol A, Diakite A, Laurent A (2017). **frailtypack**: *General Frailty Models: Shared, Joint and Nested Frailty Models with Prediction*. R package version 2.12.6, URL <https://CRAN.R-project.org/package=frailtypack>.

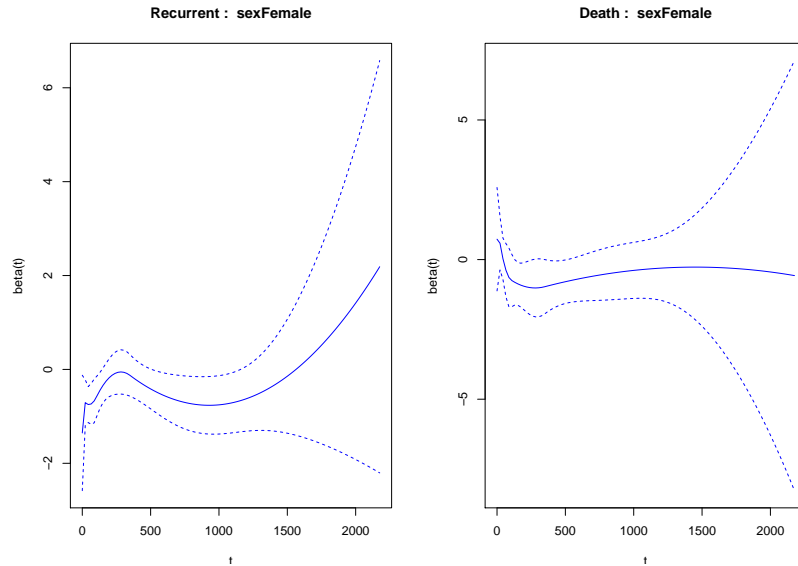
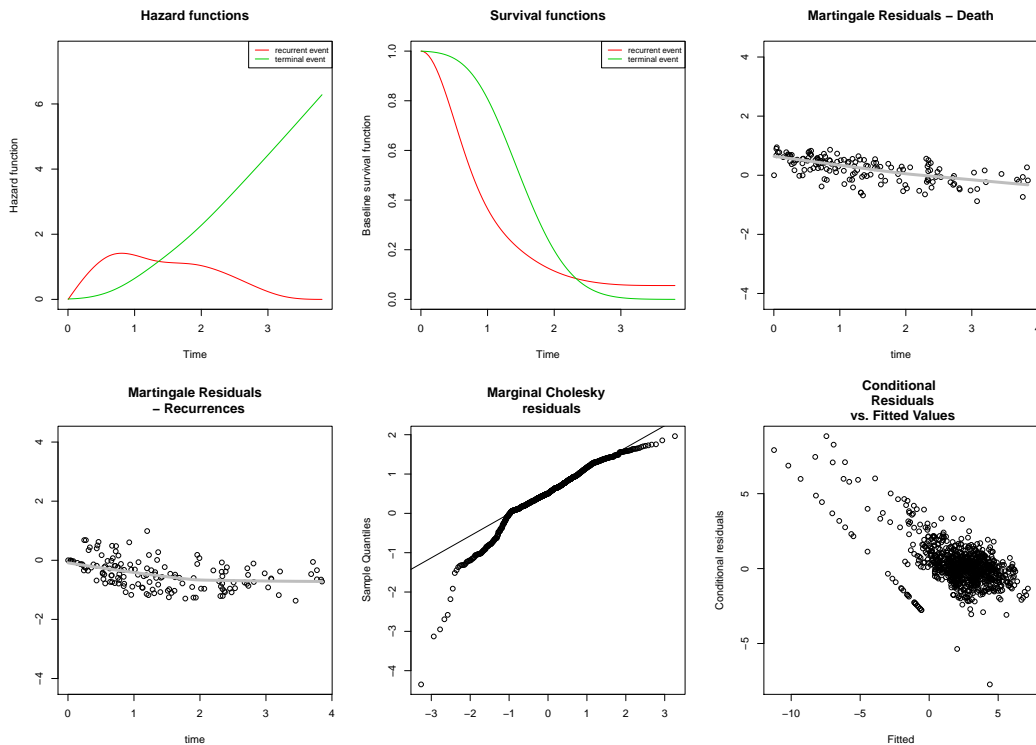
- Rondeau V, Mathoulin-Pélissier S, Jacqmin-Gadda H, Brouste V, Soubeyran P (2007). “Joint Frailty Models for Recurring Events and Death Using Maximum Penalized Likelihood Estimation: Application on Cancer Events.” *Biostatistics*, **8**(4), 708–721.
- Rondeau V, Mazroui Y, Gonzalez JR (2012). “**frailtypack**: An R Package for the Analysis of Correlated Survival Data with Frailty Models Using Penalized Likelihood Estimation or Parametrical Estimation.” *Journal of Statistical Software*, **47**(4), 1–28. doi:10.18637/jss.v047.i04.
- Rondeau V, Pignon JP, Michiels S (2015). “A Joint Model for the Dependence between Clustered Times to Tumour Progression and Deaths: A Meta-Analysis of Chemotherapy in Head and Neck Cancer.” *Statistical Methods in Medical Research*, **24**(6), 711–729. doi:10.1177/0962280211425578.
- SAS Institute Inc (2013). *The SAS System, Version 9.4*. SAS Institute Inc., Cary, NC. URL <http://www.sas.com/>.
- Sinha D, Maiti T, Ibrahim JG, Ouyang B (2008). “Current Methods for Recurrent Events Data with Dependent Termination: A Bayesian Perspective.” *Journal of American Statistical Association*, **103**(482), 866–878. doi:10.1198/016214508000000201.
- StataCorp (2015). *Stata Data Analysis Statistical Software: Release 14*. StataCorp LP, College Station, TX. URL <http://www.stata.com/>.
- Therneau TM (2017). *survival: Survival Analysis*. R package version 2.41-3, URL <https://CRAN.R-project.org/package=survival>.
- Wulfsohn MS, Tsiatis AA (1997). “A Joint Model for Survival and Longitudinal Data Measured with Error.” *Biometrics*, **53**(1), 330–339. doi:10.2307/2533118.
- Yu Z, Liu L (2011). “A Joint Model of Recurrent Events and a Terminal Event with a Nonparametric Covariate Function.” *Statistics in Medicine*, **30**(22), 2683–2695. doi:10.1002/sim.4297.
- Yu Z, Liu L, Bravata DM, Williams LS (2014). “Joint Model of Recurrent Events and a Terminal Event With Time-Varying Coefficients.” *Biometrical Journal*, **56**(2), 183–197. doi:10.1002/bimj.201200160.
- Zhang D, Chen MH, Ibrahim JG, Boye ME, Shen W (2016). “**JMFit**: A SAS Macro for Joint Models of Longitudinal and Survival Data.” *Journal of Statistical Software*, **71**(3), 1–24. doi:10.18637/jss.v071.i03.
- Zhao X, Liu L, Liu Y, Xu W (2012). “Analysis of Multivariate Recurrent Event Data with Time-Dependent Covariates and Informative Censoring.” *Biometrical Journal*, **54**(5), 585–599. doi:10.1002/bimj.201100194.

A. Summary of the package frailtypack

	Cox	Shared	Nested	Additive	Joint standard (Bivariate: 1 RE + 1 TE)	Joint cluster (Bivariate: 1 RE + 1 TE)	Joint general (Bivariate: 1 RE + 1 TE)	Joint nested (Bivariate: 1 RE + 1 TE)	Joint longitudinal (Bivariate: 1 LO + 1 TE)	Joint trivariate (Trivariate: 1 LO + 1 RE + 1 TE)	Joint Multivariate (Trivariate: 2 RE + 1 TE)
Available options											
Gamma distribution		×	×		×	×	×	×			
Log-Normal distribution		×		×	×				×	×	×
Left-truncation	×	×	×								
Interval Censoring	×	×			×	×					
Two strata	×	×	×	×							
More strata (max=6)	×	×			×						
Time-dependant covariates	×	×			×	×					
Calendar timescale	×	×	×		×			×		×	×
Weibull	×	×	×	×	×	×		×	×	×	×
Piecewise	×	×	×	×	×	×					×
Available output											
Predicted frailties		×	×	×	×			×	×	×	
Variances of the frailties		×									
Martingale residuals	×	×	×	×	×			×	×	×	
Prediction methods											
Marginal prediction of a terminal event	×	×			×			×	×	×	
Conditional prediction of a terminal event	×	×									
Marginal prediction of a new recurrent event		×			×						
Conditional prediction of a new recurrent event		×									
Model evaluation											
Cmeasures	×										
Epoce					×				×	×	

Figure 9: Package characteristics (version 2.12.3). Blue cross is for option available for a given type of model in the package on CRAN, orange cross is for option included in the package but not on CRAN yet. Empty cells mean that an option is not available for a given type of model. RE = Recurrent event. TE = Terminal event. LO = Longitudinal outcome.

B. Additional graphics

Figure 10: Log-hazard ratios for sex for `modJoint.gap.timedep` model.Figure 11: Goodness-of-fit of trivariate model `modTrivariate`. From the top-left plot: baseline hazard and survival functions, martingale residuals (the grey line corresponds to a smooth curve obtained with `lowess`), Q-Q normal plot of the marginal Cholesky residuals of the biomarker and the conditional residuals against the fitted values of the biomarker.

Affiliation:

Agnieszka Król, Audrey Mauguen, Virginie Rondeau
INSERM U1219 (Biostatistics Team), Université de Bordeaux
146, rue Léo Saignat, 33076 Bordeaux Cedex, France
Telephone: +33/5/57/57/45/31
E-mail: Agnieszka.Listwon@u-bordeaux.fr, Audrey.Mauguen@u-bordeaux.fr,
Virginie.Rondeau@u-bordeaux.fr
URL: <http://www.bordeaux-population-health.center>

Yassin Mazroui
Sorbonne Universités, Université Pierre et Marie Curie,
Laboratoire de Statistique Théorique et Appliquée
F-75005, Paris, France
and
INSERM, UMR S 1136,
Institut Pierre Louis d'Épidémiologie et de Santé Publique
F-75013, Paris, France
E-mail: Yassin.Mazroui@upmc.fr

Stefan Michiels
Service de Biostatistique et d'Epidemiologie
Gustave Roussy
114 Rue Edouard Vaillant
94800 Villejuif, France
and
INSERM U1018, CESP
Université Paris-Sud, Université Paris-Saclay
Villejuif, France
E-mail: Stefan.Michiels@gustaveroussy.fr