Abstract

Frailty models are very useful for analysing correlated survival data, when observations are clustered into groups or for recurrent events. The aim of this article is to present the new version of an R package called frailtypack. This package allows to fit Cox models and four types of frailty models (shared, nested, joint, additive) that could be useful for several issues within biomedical research. It is well adapted to the analysis of recurrent events such as cancer relapses and/or terminal events (death or lost to follow-up). The approach uses a maximum penalized likelihood estimation. Right-censored or left-truncated data are considered. It also allows stratification and time-dependent covariates during analysis.

Keywords: Frailty models, R, Penalized likelihood, Marquardt, Cross-validation, Correlated survival data, Splines, Hazard functions.

1. Introduction

Frailty models (Duchateau and Janssen 2008; Hougaard 2000; Wienke 2010; Hanagal 2011) are extensions of the Cox proportional hazards model (Cox 1972) which is the most popular model in survival analysis. In many clinical applications, the study population needs to be considered as a heterogeneous sample or as a cluster of homogeneous groups of individuals such as families or geographical areas. Sometimes, due to the lack of knowledge or for economical reasons, some covariates related to the event of interest are not measured. The frailty approach is a statistical modelling method which aims to account for the heterogeneity caused by
unmeasured covariates. It does so by adding random effects which act multiplicatively on the
hazard function. frailtypack is an R package which allows to fit four types of frailty models, for
left-truncated and right-censored data, adapted to most survival analysis issues. The aim of
this paper is to present the new version of the R package frailtypack, which is available from
the Comprehensive R Archive Network at http://CRAN.R-project.org/, and the various
new models proposed. The initial version of this package (Rondeau and Gonzalez 2005) was
proposed for a simple shared frailty model. The shared frailty model (Rondeau, Commenges,
and Joly 2003) can be used, when observations are supposed to be clustered into groups. The
nested frailty model (Rondeau, Filleul, and Joly 2006) is most appropriate, when there are
two levels of hierarchical clustering. However, several relapses (recurrent events) are likely to
increase the risk of death, thus the terminal event is considered as an informative censoring.
Using a joint frailty model, it is possible to fit jointly the two hazard functions associated with
recurrent and terminal events (Rondeau, Mathoulin-Pelissier, Jacqmin-Gadda, Brouste, and
Soubeyran 2007), when these events are supposed to be correlated. The additive frailty model
(Rondeau, Michiels, Liquet, and Pignon 2008) is more adapted to study both heterogeneity
across trial and treatment-by-trial heterogeneity (for instance meta-analysis or multicentric
datasets study). Depending on the models, stratification and time-dependent covariates are
allowed or not. The frailty models discussed in recent literature present several drawbacks.
Their convergence is too slow, they do not provide standard errors for the variance estimate of
the random effects and they can not estimate smooth hazard function. frailtypack use a non-
parametric penalized likelihood estimation and the smooth estimation of the baseline hazard
functions is provided by using an approximation by splines. frailtypack was first written in
Fortran 77 and was implemented for the statistical software R. Section 2 presents the models
that frailtypack can fit and the estimation method. Section 3 describes all the functions
and the arguments of frailtypack. Section 4 provides some examples illustrating frailtypack
functions.

2. Models and estimation methods

2.1. Models

In the following models, the covariates could be time-dependent but to simplify the expres-
sions we replace for instance $X_{ij}(t)$ by $X_{ij}$.

Cox model

The proportional hazards Cox model is a frailty model without random effect. With
frailtypack, it is possible to fit such a model with parameters estimated by the penalized
likelihood maximization.

Shared frailty model

When observations are clustered into groups such as hospitals or cities, or when observations
are recurrent events times (cancer relapses), the shared gamma frailty model is the most	en often adapted model (Rondeau et al. 2003). For the $j$th ($j = 1, ..., n_i$) individual of the $i$th
group \((i = 1, \ldots, G)\), let \(T_{ij}\) denote the recurrent event times under study, let \(C_{ij}\) be the right-censoring times and let \(L_{ij}\) be the left truncation times. The observations \(Y_{ij}\) equal to \(\min(T_{ij}, C_{ij})\) and the censoring indicators are \(\delta_{ij} = I_{(Y_{ij} = T_{ij})}\). The recurrent event times may be left-truncated and right-censored. Stratified analysis by a binary variable is allowed. The hazard function for a shared frailty model is

\[
\lambda_{ij}(t|v_i) = v_i \lambda_0(t) \exp(\beta' X_{ij}) = v_i \lambda_{ij}(t),
\]

where \(\lambda_0(t)\) is the baseline hazard function, \(X_{ij}\) the covariate vector associated with \(\beta\) the vector of regression parameters, and \(v_i\) is the random effect associated with the \(i^{th}\) group. We assume that the \(v_i\) are independently and identically distributed (i.i.d.) from a gamma distribution with \(E(v_i) = 1\) and \(\text{Var}(v_i) = \theta\), i.e. \(v_i \sim \Gamma\left(\frac{1}{\theta}, \frac{1}{\theta}\right)\).

The full marginal loglikelihood for this model has an analytical formulation (Klein 1992)

\[
l(\Phi) = \sum_{i=1}^{G} \left\{ \sum_{j=1}^{n_i} \delta_{ij} \ln \lambda_{ij}(Y_{ij}) - \left( \frac{1}{\theta} + m_i \right) \ln \left[ 1 + \theta \sum_{j=1}^{n_i} \Lambda_{ij}(Y_{ij}) \right] \right. \\
+ \frac{1}{\theta} \ln \left[ 1 + \theta \sum_{j=1}^{n_i} \Lambda_{ij}(L_{ij}) \right] + I_{\{m_i \neq 0\}} \sum_{k=1}^{m_i} \ln \left( 1 + \theta (m_i - k) \right) \right\},
\]

where \(\Phi = (\lambda_0(\cdot), \beta, \theta)\), the cumulative baseline hazard function is \(\Lambda_0(\cdot)\) and the number of recurrent events is \(m_i = \sum_{j=1}^{n_i} I_{\{\delta_{ij} = 1\}}\).

**Nested frailty model**

The nested frailty models account for the hierarchical clustering of the data by including two nested random effects that act multiplicatively on the hazard function. Such models are appropriate when observations are clustered at several hierarchical levels such as in geographical areas (Rondeau et al. 2006). For instance, the American Cancer Society Study on air pollution and mortality (Pope, Thun, Namboodiri, Dockery, Evans, Speizer, and Heath 1995) included 552,138 subjects from 151 metropolitan areas throughout the United States, themselves nested within 44 states. As the hypothesis of independent observations is not a priori obvious in this cohort, a flexible survival model with two nested random effects at state and city levels is needed to obtain valid estimates. We consider \(G\) independent clusters and within the \(i^{th}\) there are \(n_i\) subclusters. Left-truncated and right-censored data are allowed. Stratified analysis by a binary variable is allowed, too. Let \(T_{ijk}\) denote the survival times under study for the \(k^{th}\) subject \((k = 1, \ldots, K_{ij})\) from the \(j^{th}\) subgroup \((j = 1, \ldots, n_i)\) of the \(i^{th}\) group \((i = 1, \ldots, G)\), \(C_{ijk}\) the right-censoring times and \(L_{ijk}\) the left-truncating times. The observations are \(Y_{ijk} = \min(T_{ijk}, C_{ijk})\) and the censoring indicators \(\delta_{ijk}\). We assumed that \(T_{ijk}, L_{ijk}\) and \(C_{ijk}\) are independent.

The hazard function for a nested frailty model is

\[
\lambda_{ijk}(t|v_i, z_{ij}) = v_i z_{ij} \lambda_0(t) \exp(\beta' X_{ijk}) = v_i z_{ij} \lambda_{ijk}(t),
\]

where \(\lambda_0(t)\) is the baseline hazard function and \(X_{ijk}\) the covariate vector associated with \(\beta\) the corresponding vector of regression parameters. The cluster random effect \(v_i\) and the subcluster random effect \(z_{ij}\) are both independently and identically gamma-distributed:
The full marginal loglikelihood takes the following form (Rondeau et al. 2006)

\[
l(\Phi) = \sum_{i=1}^{G} \left\{ \frac{1}{\eta} \prod_{j=1}^{n_i} \left[ 1 + \eta v_i \sum_{k=1}^{K_{ij}} \Lambda_{ijk}(Y_{ijk}) \right]^{1/\eta + m_{ij}} \right\} - \ln \int_{0}^{\infty} \frac{v_i^{1/\theta - 1} \exp(-v_i/\theta)}{\prod_{j=1}^{n_i} \left[ 1 + \eta v_i \sum_{k=1}^{K_{ij}} \Lambda_{ijk}(Y_{ijk}) \right]^{1/\eta + m_{ij}}} \, dv_i - \ln \int_{0}^{\infty} \frac{v_i^{1/\theta - 1} \exp(-v_i/\theta)}{\prod_{j=1}^{n_i} \left[ \eta v_i \sum_{k=1}^{K_{ij}} \Lambda_{ijk}(L_{ijk}) + 1 \right]^{1/\eta}} \, dv_i,
\]

(4)

where \( \Phi = (\lambda_0(.), \beta, \eta, \theta) \), the cumulative baseline hazard function is \( \Lambda_{ijk}(.) \), the number of recurrent events in the \( i^{th} \) group is \( m_i = \sum_{j=1}^{n_i} \sum_{k=1}^{K_{ij}} I_{\{\delta_{ijk}=1\}} \) and the number of observed events in the \( j^{th} \) subgroup is \( m_{ij} \).

### Joint frailty model

The observation of successive events across time (recurrent events) for subjects in cohort studies may be terminated by loss to follow-up, end-of-study, or a major failure event such as death. In this context, the major failure event could be correlated with recurrent events and the usual assumption of noninformative censoring of the recurrent event process by death is not valid. Joint frailty models allow to study the joint evolution over time of two survival processes by considering the terminal event as informative censoring (Rondeau et al. 2007). A common frailty term \( v_i \) for the two rates takes into account the heterogeneity in the data, associated with unobserved covariates. We assume that the \( v_i \) are independently and identically distributed (i.i.d.) from a gamma distribution with \( \mathbb{E}(v_i) = 1 \) and \( \text{Var}(v_i) = \theta \), i.e. \( v_i \sim \Gamma \left( \frac{1}{\theta}, \frac{1}{\theta} \right) \). The frailty term acts differently on the two rates (\( v_i \) for the recurrent rate and \( v_i^a \) for death rate). The parameters \( \theta \) and \( \alpha \) characterizes the dependence between recurrent event process \( T_{ij} \) and terminal event time \( T_i^* \) attributed to the unobserved random effects. Zero value of \( \alpha \) implies that the dependence between \( T_{ij} \) and \( T_i^* \) can be fully explained by the (observed) covariates. If \( \theta = 0 \), \( T_{ij} \) and \( T_i^* \) are considered independent and the parameter \( \alpha \) is non interpretable. On the other hand, if \( \theta \) is nonzero and \( \alpha \) is also nonzero, then \( \theta \) not only accounts for the intra-subjects correlations, but also represents the dependence between the recurrent event rate and the terminal event rate. The covariates could be different for the recurrent rate and the death rate. This model can be fitted to right-censored and/or left-truncated data, but stratification by a boolean variable is not allowed.

We denote for subject \( i \) (\( i = 1, ..., G \)) \( T_{ij} \) the \( j^{th} \) recurrent time (\( j = 1, ..., n_i \)) considered as time-to-event, \( C_i \) the censoring time, \( L_i \) the left-truncating time, \( D_i \) the death time. \( Y_{ij} = \text{min}(T_{ij}, C_i, D_i) \) corresponds to each observation time, and \( \delta_{ij} = I_{\{Y_{ij}=T_{ij}\}} \) is the recurrent event indicator. \( T_i^* = \text{min}(C_i, D_i) \) is the last follow-up time and \( \delta_i^* = I_{\{T_i^*=D_i\}} \) the
death indicator. And we observe \( Y_{ij}, T_i^*, \delta_{ij}, \delta_i^* \).

The hazard functions system for joint frailty models of recurrent events and death is

\[
\begin{aligned}
\{ \ & r_{ij}(t|v_i) = v_i r_0(t) \exp(\beta'_i X_{ij}) = v_i r_{ij}(t) \\
\ & \lambda_i(t|v_i) = v_i^\alpha \lambda_0(t) \exp(\beta'_2 X_i) = v_i^\alpha \lambda_i(t) \end{aligned}
\]

(recurrent events) (death),

(5)

where \( r_0(t) \) (resp. \( \lambda_0(t) \)) is the recurrent (resp. terminal) event baseline hazard function, \( \beta_1 \) (resp. \( \beta_2 \)) the regression coefficients vector associated with \( X_{ij} \) (resp. \( X_i \)) the covariate vector and \( v_i \sim \Gamma(\frac{1}{\theta}, \frac{1}{\theta}) \) are i.i.d..

Contrary to the shared frailty model, the full log-likelihood does not take a simple form because the integrals do not have a closed form (Rondeau et al. 2007). The full marginal log-likelihood expression is

\[
I(\Phi) = \sum_{i=1}^G \left( \sum_{j=1}^{n_i} \delta_{ij} \ln \left( r_{ij}(Y_{ij}) \right) + \delta_i^* \ln \left( \lambda_i(T_i^*) \right) - \ln \left( \Gamma(\frac{1}{\theta}) \right) - \frac{1}{\theta} \ln \theta \\
+ \ln \int_0^\infty v_i^{m_i+\alpha \delta_i^*+1/\theta-1} \exp \left[ -v_i \int_0^{T_i^*} dR_{ij}(t) dt - v_i^\alpha \int_0^{T_i^*} d\Lambda_i(t) dt - \frac{v_i}{\theta} \right] dv_i \right),
\]

(6)

where \( \Phi = (r_0(.), \lambda_0(.), \beta, \alpha, \theta) \), with \( \beta = (\beta_1, \beta_2) \) the covariates vector for recurrent events or death, with \( Y_{i0} = 0 \) and \( Y_{im_i} = T_i^* \). \( \Lambda_i(t) = \int_0^t \lambda_i(u) du \) (resp. \( R_{ij}(t) = \int_0^t r_{ij}(u) du \)) is the cumulative hazard function for death (resp. for recurrent events) and \( m_i \) is the total number of recurrent events for subject \( i \).

Additive frailty model

In a meta-analysis of clinical trials, the main objective consists in both looking at the heterogeneity between trials of underlying risk and treatment effects. An additive frailty model is a proportional hazards model with two correlated random effects at the trial level that act multiplicatively on the hazard function and on the interaction with treatment. This approach allows not only variations of the baseline hazard function across trials but also variations of the treatment effect across trials in a meta-analysis concerning individual survival data (Vaida and Xu 2000; Legrand, Ducrocq, Janssen, Sylvester, and Duchateau 2005; Rondeau et al. 2008; Ha, Sylvester, Legrand, and MacKenzie 2011). If we are only interested in the variation of the baseline hazard function across trials, a simple shared frailty model can be used (See 2.1.).

In the case of an additive frailty model, only right-censored data are allowed, but not left-truncated data. We suppose that the G trials are independent. For the \( j^{th} \) (\( j = 1, ..., n_i \)) individual of the \( i^{th} \) trial (\( i = 1, ..., G \)), let \( T_{ij} \) denote the survival times under study and let \( C_{ij} \) be the corresponding right-censoring times. The observations \( Y_{ij} \) equal to \( \min(T_{ij}, C_{ij}) \) and the censoring indicators \( \delta_{ij} = I_{\{Y_{ij}=T_{ij}\}} \). For each subject, we observe a binary variable \( X_{ij1} \) which equals 1 when the patient is in the experimental arm and 0 when in the standard arm, as a treatment arm indicator. Stratified analysis by a binary variable is allowed.

The hazard function for the \( j^{th} \) individual of the \( i^{th} \) trial with random trial effect \( u_i \) and random treatment-by-trial interaction \( w_i \) is

\[
\lambda_{ij}(t|u_i, w_i) = \lambda_0(t) \exp(u_i + w_i X_{ij1} + \sum_{k=1}^p \beta_k X_{ijk}),
\]

(7)
where \( \lambda_0(t) \) is the baseline hazard function, \( \beta_k \) the fixed effect associated with the covariate \( X_{ijk} \) \((k=1,...,p)\), \( \beta_1 \) is the treatment effect and \( X_{ij1} \) the treatment variable. \( \rho \) is the corresponding correlation coefficient for the two frailty terms.

The variance \( \sigma^2 \) of \( u_i \) represents the heterogeneity between trials of the overall underlying baseline risk and the variance \( \tau^2 \) of \( w_i \) represents the heterogeneity between trials of the overall treatment effect \( \beta_1 \). If \( \sigma^2 = 0 \), then observations from the same trial are independent. A larger variance implies greater heterogeneity across trials and a greater correlation of the survival times for individuals belonging to the same trial. A null variance \( \tau^2 \) indicates no heterogeneity of the treatment effect over trials. The full marginal log-likelihood expression for an additive frailty model is

\[
\ln G \prod_{i=1}^G \int_{\mathbb{R}} \int_{\mathbb{R}} \left\{ \prod_{j=1}^{n_i} \lambda(Y_{ij}|u_i, w_i) S(Y_{ij}|u_i, w_i) \right\} f(u_i, w_i) du_i dw_i,
\]

where \( \Phi = (\lambda_0(\cdot), \beta, \sigma^2, \tau^2, \rho) \). The calculus of this log-likelihood is detailed elsewhere (Rondeau et al. 2008).

This marginal log-likelihood depends on integrals that have no analytical solution. We approximate it by using the Laplace integration technique, which seems to provide good estimates.

2.2. Computational methods and estimation

The estimation method used in \texttt{frailtypack} is the maximization of the penalized loglikelihood (Joly, Commenges, and Letenneur 1998; Rondeau et al. 2003).

**Penalized likelihood**

The penalized loglikelihood has different expressions according to the models.

For Cox, shared, nested, additive frailty models, it is

\[
pl(\Phi) = l(\Phi) - \kappa \int_0^\infty \lambda''_0(t)^2 dt,
\]

where \( \kappa \) is a positive smoothing parameter which controls the trade-off between the data fit and the smoothness of the functions.

If it is a stratified analysis with a maximum of two strata (for instance, a stratification on gender), the penalized loglikelihood is

\[
pl_{str}(\Phi) = l_{str}(\Phi) - \kappa_1 \int_0^\infty \lambda''_{0m}(t)^2 dt - \kappa_2 \int_0^\infty \lambda''_{0f}(t)^2 dt,
\]

where \( \lambda_{0m}(\cdot) \) and \( \lambda_{0f}(\cdot) \) are the baseline hazard functions for men and women respectively. We suppose that these 2 baseline hazard functions are different. That is why it requires two positive smoothing parameters: \( \kappa_1 \) for the stratum 1 (men) and \( \kappa_2 \) for the stratum 2 (women).

The penalized loglikelihood for joint frailty models is thus defined as

\[
pl_{joint}(\Phi) = l_{joint}(\Phi) - \kappa_1 \int_0^\infty \lambda''_0(t)^2 dt - \kappa_2 \int_0^\infty \lambda''_0(t)^2 dt,
\]
Two positive smoothing parameters (κ₁ and κ₂) are necessary because a joint frailty model requires two hazard functions (r₀(.) for recurrent events and λ₀(.) for death). Stratified analyses with this model are not allowed.

For a fixed value of the smoothing parameter, the maximization of the penalized likelihood provides estimators for Φ, the parameters of the model.

Maximization of the Penalized likelihood estimation

The estimated parameters are obtained by the robust Marquardt algorithm (Marquardt 1963), which is a combination between a Newton Raphson algorithm and a steepest descent algorithm. This algorithm has the advantage of being more stable than the Newton Raphson algorithm while preserving its fast convergence property. To be sure of having positive hazard functions at all stages of the algorithm, the variance of the frailties and the spline coefficients need to be positive. We ensure this positiveness by using a square transformation.

The vector Φ of the parameters is updated until the convergence using the following iterative expression

\[ \Phi^{(r+1)} = \Phi^{(r)} - \delta \left( \tilde{H}^{(r)} \right)^{-1} \Delta \left( L(\Phi^{(r)}) \right), \]  

(12)

The step δ is equal to 1 by default but can be modified to ensure that the likelihood is improved at each iteration. The matrix \( \tilde{H} \) is a diagonal-inflated Hessian matrix to ensure positive definiteness. The term \( \Delta( L(\Phi^{(r)}) ) \) corresponds to the penalized log-likelihood gradient at the \( r^{th} \) iteration. The iterations stop when the difference between two consecutive log-likelihoods is small (< 10^{-4}), the coefficients are stable (< 10^{-4}) and the gradient is small enough (< 10^{-6}). The first and second derivatives are calculated using finite differences method. After the convergence, the standard errors of the estimates are directly obtained from the inverse of the Hessian matrix.

The integrals in the full log-likelihood expression do not always have an analytical solution. They are evaluated using a Gaussian quadrature with Laguerre polynomials with 20 points for the nested and the joint frailty models. They are evaluated using the Laplace integration method for the additive frailty models.

Approximation with splines

The estimator of the baseline hazard function λ₀(.) has no analytical solution, but can be approximated on the basis of splines with Q knots: \( \tilde{\lambda}_0(.) = \sum_{i=1}^{m} \eta_i M_i(\cdot) \), with m=Q+2.

Cubic M-splines (polynomial functions of 3rd order that are combined linearly to approximate a function on an interval) which are a variant of cubic B-splines are used. The second derivative of the baseline hazard function \( \lambda''_0(.) \) is approximated by a sum of polynomial functions of 1st order. This approximation allows flexible shapes of the hazard function while reducing the number of parameters. The more knots we use, the closer was the approximation to the true hazard function. An approximation for the confidence bands at 95% of λ₀(.) is provided:

\[ \tilde{\lambda}_0(t) \pm 1.96 \sqrt{M(t)^T \hat{I}_\eta^{-1} M(t)}, \]  

(13)

where \( M(t) = \left( M_1(t), ..., M_m(t) \right) \) is the M-splines vector and \( \hat{I}_\eta = \frac{\partial^2 \rho(\eta)}{\partial \eta^2} \).
Parameter initialisation

The frailtypack programs include several algorithms which need initial values. It is very important to choose good initial values for the maximization of the penalized likelihood. The closer the initial value is to the true value, the faster the convergence. According to the model used, we applied different methods to provide initial values.

• For a shared frailty model, the splines and the regression coefficients are initialized to 0.1. The program fits, firstly, an adjusted Cox model to give new initial values for the splines and the regression coefficients. The variance of the frailty term $\theta$ is initialized to 0.1. Then, a shared frailty model is fitted.

• For a joint frailty model, the splines and the regression coefficients are initialized to 0.5. The program fits an adjusted Cox model to provide new initial values for the regression and the splines coefficients. The variance of the frailty term $\theta$ and the coefficient $\alpha$ are initialized to 1. Then, it fits a joint frailty model.

• For a nested frailty model, the splines and the regression coefficients are initialized to 0.1. The program fits an adjusted Cox model to provide new initial values for the regression and the splines coefficients. The variances of the frailties are initialized to 0.1. Then, a shared frailty model with covariates considering only subgroup frailties is fitted to give a new initial value for the variance of the subgroup frailty term. Then, a shared frailty model with covariates and considering only group frailties is fitted to give a new initial value for the variance of the group frailties. In a last step, a nested frailty model is fitted.

• For an additive frailty model, the splines and the regression coefficients are initialized to 0.1. An adjusted Cox model is fitted, to provide new initial values for the splines coefficients and the regression coefficients. The variances of the frailties are initialized to 0.1. Then an additive frailty model with independent frailties is fitted. At last, an additive frailty model with correlated frailties is fitted.

Another important point is the choice of the smoothing parameters.

How to estimate the smoothing parameter - Kappa

The smoothing parameter can be fixed by the user or evaluated by an automatic method: the maximization of a likelihood cross-validation criterion, (Joly et al. 1998) for Cox model. This method provides an estimated value of the smoothing parameter by minimizing the following function $\bar{V}(\kappa)$

$$\bar{V}(\kappa) = \frac{1}{n} \left\{ tr(\hat{H}_{pl}^{-1} \hat{H}_l - l(\hat{\Phi}_\kappa)) \right\},$$

(14)

where $\hat{\Phi}_\kappa$ is the maximum penalized likelihood estimators, $\hat{H}_l$ is minus the converged Hessian matrix of the log-likelihood, $\hat{H}_{pl}$ is minus the converged Hessian matrix of the penalized log-likelihood and $l(.)$ is the full log-likelihood.

To minimize $\bar{V}(\kappa)$, we calculate it for several and very remote values of $\kappa$ (for avoiding the local minimum).
The cross-validation method is not implemented for more than one smoothing parameter. Thus, it can not be used for a **stratified analysis** and for a **joint frailty model**.

Concerning the choice of the kappas for a joint frailty model, first we have to fit two shared frailty models with cross-validation method: one with recurrences as event of interest and the other one with death as event of interest. Then, the two $\kappa$ obtained are used in the joint frailty model. If the terminal event survival curve is too smoothed, we can set $\kappa_2=\kappa_1$ (See modJoint.gap in section 4.2). Choosing the kappas for a stratified model is based on the same idea. First, we fit a shared frailty model without stratification (See. mod.sha.gap in section 4.2) using a cross-validation method for $\kappa$. Then this $\kappa$ is used for the two strata in the stratified shared frailty model (See mod.sha.str.gap in section 4.2)

### 3. Frailtypack arguments

The two main functions which can be used under the library **frailtypack** are **frailtyPenal** for shared, joint and nested frailty models and **additivePenal** for additive frailty models. Different arguments for each type of model are proposed. We describe here each of the arguments needed for these functions and explain how to parametrize them to fit each type of model.

#### 3.1. Arguments of the main functions

The standard code for fitting a Cox model or a shared, joint and nested frailty model is:

```r
frailtyPenal(formula, formula.terminalEvent, data, Frailty = FALSE, joint = FALSE, recurrentAG = FALSE, cross.validation = FALSE, n.knots, kappa1, kappa2, maxit = 350)
```

The standard code for fitting an additive frailty model is:

```r
additivePenal(formula, data, correlation=FALSE, n.knots, cross.validation = FALSE, kappa1, kappa2, maxit = 350).
```

*Common arguments for fitting a model*

- **formula**: indicates the model which is fitted. It is different according to the model. (See the following subsections)
- **data**: indicates the name of the data file. The database structure is different according to the model. (See section 4.1)
- **recurrentAG**: Logical value (TRUE or FALSE). If recurrentAG=TRUE, it indicates that the counting process approach of Andersen and Gill (Andersen and Gill 1982) with a calendar timescale for recurrent event times is used. Within a calendar timescale, the time corresponds to the time since entry/inclusion in the study. Within a gap timescale, the time corresponds to the time between two recurrent events. Default is FALSE, in particular for recurrent events or clustered data with gap-time as the timescale. Always FALSE when an additive frailty model is fitted.
• **n.knots**: is the number of knots to use. It corresponds to the n.knots+2 splines functions for the approximation of the baseline hazard function or the survival functions. The number of knots must be between 4 and 20. It is recommended to start with a small number of knots (for instance: n.knots=7) and to increase the number of knots until the graph of the baseline hazard function remains unchanged.

• **kappa1**: is the smoothing parameter of the penalized likelihood.

• **kappa2**: is the second smoothing parameter, it is required if the analysis is stratified (only for Cox, shared, nested and additive frailty models). This parameter will corresponds to the smoothing parameter for the second baseline hazard function. If a joint frailty model is fitted, this parameter will correspond to the smoothing parameter for the death baseline hazard function (**kappa1** being the smoothing parameter for the recurrent events baseline hazard function).

• **cross.validation**: if=TRUE, indicates that a cross validation procedure is used for estimating the best smoothing parameter. **kappa1** is used as the seed for estimating the smoothing parameter (See section 2.2). It is important to underline that if a joint frailty model is fitted or a stratified analysis for Cox, shared, additive, nested frailty models is used, the cross-validation method is not allowed, i.e. cross.validation=FALSE. Default is FALSE.

• **maxit**: is the maximum number of iterations for the Marquardt algorithm. Default is 350.

**Specified arguments for fitting a Cox or a Shared frailty model**

• **formula** (example):

  frailtyPenal(Surv(time,event) ~ var1 + var2 + cluster(id), n.knots=12, kappa1=10000, data=database, Frailty=FALSE)

  **cluster** is a survival function which identifies groups of correlated observations, **time** is the follow-up time, **event** is a event indicator (0=censored, 1=event), **var1** and **var2** are some explanatory variables.

• **Frailty**: Logical value (TRUE or FALSE). If Frailty=TRUE, the model includes a frailty term (a shared frailty model is fitted) and the variance of the frailty parameter is estimated. If Frailty=FALSE, a Cox proportional hazards model is estimated using penalized likelihood on the hazard function. Default is FALSE.

**Specified arguments for fitting a Joint frailty model**

• **formula** (example):

  frailtyPenal(Surv(t1,t2,event) ~ cluster(id) + var1 + var2 + terminal(status.terminal), formula.terminalEvent=var1, data=database,
terminal is a special function used in the context of recurrent event models with terminal event (e.g., censoring variable related to recurrent events). It contains the recurrent event indicator, 0=no relapse, 1=relapse. status.terminal is a death indicator. \texttt{t1} is the entry time, \texttt{t2} is the last follow-up time.

- \textbf{joint}: Logical value (TRUE or FALSE). The default is FALSE. If TRUE a joint frailty model is fitted and if so, the ’formula.terminalEvent’ argument is required.

- \textbf{formula.terminalEvent}: indicates which covariates the terminal event rate adjusted for.

**Specified arguments for fitting a Nested frailty model**

- \textbf{formula} (example type):
  \texttt{frailtyPenal(Surv(t1,t2,event) ~ cluster(group)+subcluster(subgroup) + cov1 + cov2, data=database, n.knots=8, kappa1=50000, recurrentAG=TRUE)}
  The \texttt{subcluster} function identifies subgroups levels, the \texttt{cluster} function from the survival package identifies groups levels.

**Specified arguments for fitting an Additive frailty model**

- \textbf{formula} (example type):
  \texttt{additivePenal(Surv(t2,event) ~ cluster(group)+slope(var1)+var1, data=database, correlation=TRUE, n.knots=8, kappa1=10000)}
  The \texttt{slope} function identifies the variable in interaction with the random slope ($w_i$).

- \textbf{correlation}: Logical value (TRUE or FALSE). Are the two random effects ($u_i$ and $w_i$) correlated? If so, the correlation coefficient is estimated. The default is TRUE.

### 3.2. Objects returned by frailtyPenal and additivePenal

The objects returned by frailtyPenal and additivePenal are detailed in the reference manual (see \textbf{Value} part), which is available from the Comprehensive \texttt{R} Archive Network at \url{http://CRAN.R-project.org/}.

### 3.3. Other available functions: \texttt{plot}, \texttt{summary} and \texttt{print}

\texttt{frailtypack} includes \texttt{R} methods for summarizing, printing, and plotting objects of different classes depending on the model fitted. \texttt{summary} gives estimations and confidence intervals for the hazard ratios of each covariate. \texttt{print} provides a short summary of the parameter estimates. \texttt{plot} is useful to draw baseline survival or hazard functions for each type of model. For a joint frailty model, it is possible to plot the terminal event and/or the recurrent hazard or survival functions. (See section 4.4.)
4. Illustrating examples

The Cox proportional hazards model and the shared frailty model have already been introduced in the first version of frailtypack (Rondeau and Gonzalez 2005).

To illustrate the models provided by frailtypack, three datasets are proposed: readmission, dataNested, dataAdditive. The first subsection explains how to build a dataset adapted to each model. Then, in the second subsection we describe the R code to fit the different models and the parameter estimates in the output obtained with the R function print. We also present the R function summary adapted to these models. Finally, we present the different options of the R function plot according to the model.

4.1. Build an adapted dataset

Several variables are necessary to fit the models provided by frailtypack.

Cox, shared frailty and joint frailty models

The following variables are common for all these models (see Table 1). For instance, when studying recurrent events with a Cox model or a shared frailty model, it is necessary to specify the identification number of the subject (id), an event indicator (event) and some covariates (chemo, sex, dukes, charlson) for each observation. If the interest is on the duration between two events, the gap-time timescale (e.g. recurrentAG=FALSE) is appropriate and then a variable (time) that indicates the duration between two events is needed. If the focus is on the occurrence of events during the follow-up, the calendar-time timescale (e.g. recurrentAG=TRUE) must be used and two variables such as (time.start) and (time.stop) are needed. time.start indicates the time at which the subject entered the study or the time at which the event last occurred if it is not its first occurrence. time.stop corresponds to the time when the event occurs or to the end of the follow-up when there is no more occurrence. For instance, patient 2 entered the study at time time.start=0, then developed an event at time time.stop=489 and is then censored at time time.stop=1182. In a joint frailty model, we fit jointly the terminal event rate and the recurrent event rate.

The dataset readmission is used for the Cox, shared and joint frailty model. It contains rehospitalization data of patients diagnosed with colorectal cancer (Gonzalez, Fernandez, Moreno, Ribes, Peris, Navarro, Cambray, and Borras 2005). The data describe the calendar time (in days) of the successive hospitalizations after the date of surgery. The first readmission time was considered as the time between the date of the surgical procedure and the first rehospitalization after discharge related to colorectal cancer. Each subsequent readmission time was defined as the difference between the current hospitalization date and the previous discharge date. A total of 861 rehospitalization events were recorded for the 403 patients included in the analysis. Several readmissions can occur for the same patient, and an individual frailty may influence the occurrence of subsequent rehospitalizations.

Nested frailty model

In the nested frailty model, we use two levels of clustering. We need to specify the group and the subgroup of each subject such as the variables group and subgroup in dataNested which
Table 1: Extract from the dataset readmission

<table>
<thead>
<tr>
<th>id</th>
<th>t.start</th>
<th>t.stop</th>
<th>time</th>
<th>event</th>
<th>chemo</th>
<th>sex</th>
<th>dukes</th>
<th>charlson</th>
<th>death</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>24</td>
<td>24</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>24</td>
<td>457</td>
<td>433</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>457</td>
<td>1037</td>
<td>580</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>489</td>
<td>489</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>489</td>
<td>1182</td>
<td>693</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>15</td>
<td>15</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>783</td>
<td>768</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>163</td>
<td>163</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>163</td>
<td>288</td>
<td>125</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>288</td>
<td>638</td>
<td>350</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>638</td>
<td>686</td>
<td>48</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>686</td>
<td>2048</td>
<td>1362</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

...   ...   ...   ...   ...   ...   ...   ...   ...   ...   ...   ...

indicates the identification number of the subgroup (see Table 2). t1 is the time of entrance in the study and t2 corresponds to the time of event if the event indicator event equals 1 otherwise t2 corresponds the end of follow-up. var1 and var2 are generated covariates. This dataset, included in the package, contains a simulated sample of 400 observations which may be devided 20 clusters with 4 subgroups and 5 subjects in each subgroup, in order to obtain two levels of grouping. Two independent gamma frailty parameters with a variance fixed at 0.1 for the cluster effect and at 0.5 for the subcluster effect were generated. Independent survival times were generated from a Weibull baseline risk function. The percentage of censoring data was around 30 per cent. The right-censoring variables were generated from a uniform distribution on [1,36] and a left-truncating variable was generated with a uniform distribution on [0,10]. Observations were included only if the survival time was greater than the truncation time.

Additive frailty model

In an additive frailty model, we are interested in the heterogeneity across trials. We need to specify the treatment variable. The treatment variable could be var1 (see Table 3). This model doesn’t allow left-truncated data, thus the time of entrance t1 for each subject has to be equal to zero and only gap-time timescale can be used (e.g recurrentAG=FALSE). To illustrate this model we used a generated dataset named dataAdditive included in the package.

This dataset contains simulated samples of 100 clusters with 100 subjects in each cluster, such as a compilation of clinical trials databases. Two correlated centered gaussian random effects are generated with the same variance fixed at 0.3 and the covariance at -0.2. The regression coefficient β is fixed at -0.11. The percentage of right censoring data is around 30 percent which are generated from a uniform distribution on [1,150]. Independent survival times were generated from a Weibull baseline risk function.

Then we can fit the adapted model.
4.2. Fit the different models and print the parameter estimates

The objective of the study described in (Gonzalez et al. 2005) is to analyse the hospital readmission times related to colorectal cancer after surgical procedure. We first estimate the hazard rate ratios of readmission time for covariates analysed using a Cox proportional model, then using a shared frailty model and finally using a joint frailty model.

Before fitting the model, we need to load the package frailtypack and the dataset readmission. This step is necessary for every model. An extract from this dataset is provided in Table 1 of the previous section (4.1).

> library(frailtypack)
> data(readmission)

Cox model

The following lines correspond to the R code for fitting a Cox model. We use cross-validation (cross.validation=TRUE) for the smoothing parameter because it provides smoother functions (hazard or survival).
> mod.cox.gap <- frailtyPenal(Surv(time, event) ~ cluster(id) +
  as.factor(dukes) + as.factor(charlson) + sex + chemo, Frailty=FALSE,
  n.knots=10, kappa1=1, data=readmission, cross.validation=TRUE)

With the print function, the parameter estimates of the models can be presented.

> print(mod.cox.gap)

Call:
frailtyPenal(formula = Surv(time, event) ~ cluster(id) + as.factor(dukes) +
  as.factor(charlson) + sex + chemo, data = readmission, Frailty = FALSE,
  cross.validation = TRUE, n.knots = 10, kappa1 = 1)

Cox proportional hazards model parameter estimates
using a Penalized Likelihood on the hazard function

<table>
<thead>
<tr>
<th></th>
<th>coef</th>
<th>exp(coef)</th>
<th>SE coef (H)</th>
<th>SE coef (HIH)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>as.factor(dukes)2</td>
<td>0.301</td>
<td>1.351</td>
<td>0.121</td>
<td>0.121</td>
<td>2.48</td>
<td>1.3e-02</td>
</tr>
<tr>
<td>as.factor(dukes)3</td>
<td>1.032</td>
<td>2.806</td>
<td>0.139</td>
<td>0.139</td>
<td>7.43</td>
<td>1.1e-13</td>
</tr>
<tr>
<td>as.factor(charlson)1</td>
<td>0.494</td>
<td>1.639</td>
<td>0.203</td>
<td>0.203</td>
<td>2.44</td>
<td>1.5e-02</td>
</tr>
<tr>
<td>as.factor(charlson)3</td>
<td>0.382</td>
<td>1.465</td>
<td>0.114</td>
<td>0.114</td>
<td>3.35</td>
<td>8.0e-04</td>
</tr>
<tr>
<td>sex</td>
<td>-0.474</td>
<td>0.622</td>
<td>0.101</td>
<td>0.101</td>
<td>-4.69</td>
<td>2.7e-06</td>
</tr>
<tr>
<td>chemo</td>
<td>-0.238</td>
<td>0.788</td>
<td>0.104</td>
<td>0.104</td>
<td>-2.29</td>
<td>2.2e-02</td>
</tr>
</tbody>
</table>

penalized marginal log-likelihood = -3275.92
n= 861
n events= 458 n groups= 403
number of iterations: 4
Exact number of knots used: 10
Smoothing parameter estimated by Cross validation: 2.11e+08, DoF: 11.00

In this output, we can read:

- **coef**: Regression coefficients
- **exp(coef)**: Hazard ratios
- **SE coef (H)**: Standard error estimated by inverting the Hessian matrix
- **SE coef (HIH)**: Standard error estimated using the matrix product $H^{-1}IH^{-1}$ where $H^{-1}$ is the inverse of the Hessian matrix and $I$ the Fisher Information matrix
- **z**: Wald statistics.
- **p**: p-value which is the probability $P(|z|>1.96)$. If $p<0.05$, the covariate is significantly different from 0.
**Shared frailty model**

We notice that for some subjects several relapses occur. Logically, we can imagine that a correlation within subject for the relapse times could exist, in which cases a shared frailty model would be more accurate. The change with regard to the code for a Cox model is that: Frailty=TRUE.

```r
> mod.sha.gap <- frailtyPenal(Surv(time,event) ~ cluster(id) +
  as.factor(dukes) + as.factor(charlson) + sex + chemo, Frailty=TRUE,
  n.knots=10, kappa1=1, data=readmission,cross.validation=TRUE)
> print(mod.sha.gap)
```

```
Call:
frailtyPenal(formula = Surv(time, event) ~ cluster(id) +
  as.factor(dukes) + as.factor(charlson) + sex + chemo, data = readmission,
  Frailty = TRUE, cross.validation = TRUE, n.knots = 10, kappa1 = 1)

Shared Gamma Frailty model parameter estimates
using a Penalized Likelihood on the hazard function

<table>
<thead>
<tr>
<th></th>
<th>coef</th>
<th>exp(coef)</th>
<th>SE coef (H)</th>
<th>SE coef (HIH)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>as.factor(dukes)2</td>
<td>0.298</td>
<td>1.347</td>
<td>0.161</td>
<td>0.161</td>
<td>1.85</td>
<td>6.4e-02</td>
</tr>
<tr>
<td>as.factor(dukes)3</td>
<td>1.056</td>
<td>2.875</td>
<td>0.195</td>
<td>0.195</td>
<td>5.43</td>
<td>5.7e-08</td>
</tr>
<tr>
<td>as.factor(charlson)1</td>
<td>0.452</td>
<td>1.571</td>
<td>0.259</td>
<td>0.259</td>
<td>1.74</td>
<td>8.1e-02</td>
</tr>
<tr>
<td>as.factor(charlson)3</td>
<td>0.410</td>
<td>1.507</td>
<td>0.137</td>
<td>0.137</td>
<td>3.00</td>
<td>2.7e-03</td>
</tr>
<tr>
<td>sex</td>
<td>-0.538</td>
<td>0.584</td>
<td>0.139</td>
<td>0.139</td>
<td>-3.88</td>
<td>1.1e-04</td>
</tr>
<tr>
<td>chemo</td>
<td>-0.206</td>
<td>0.813</td>
<td>0.143</td>
<td>0.143</td>
<td>-1.44</td>
<td>1.5e-01</td>
</tr>
</tbody>
</table>

Frailty parameter, Theta: 0.672 (SE (H): 0.142 ) (SE (HIH): 0.142 )

penalized marginal log-likelihood = -3236.42
n= 861
n events= 458  n groups= 403
number of iterations:  9
Exact number of knots used:  10
Best smoothing parameter estimated by
  an approximated Cross validation:  2.11e+08, DoF:  11.00
```

The variance of the frailty term \textit{theta} is significantly different from 0 \((W_m = 0.672/0.142 = 4.73 > 1.64)\), meaning that there is heterogeneity between subjects. We can deduce this by using a modified Wald test: \(W_m(\theta) = 0.672/0.142 = 4.73 > 1.64\). The modified Wald test \((W_m)\) is a significance test for the variance of the random effects distribution occurring on the boundary of the parameter space. The usual squared Wald statistic is simplified to a mixture of two distributions and hence the critical values must be derived from this mixture \((Molenberghs and Verbeke 2007)\). We have a p-value <0.05 for the covariates \textit{dukes}=3,
charlson=3 and sex. This suggests the existence of a higher risk to be rehospitalized for men with a Dukes’s stage at 3 and a Charlson index at 3.

**Shared frailty model with stratification**

Stratified analysis can be conducted using `frailtypack` if the stratified variable had 2 modalities (maximum number of strata=2). We need to set the value of the two smoothing parameters (kappa1 and kappa2) and introduce the function `strata` in the formula of the model. As the cross-validation method is not possible in this case, we use the estimation of kappa in the previous model "mod.sha" for the initial of the kappas in the stratified model. This stratification procedure is the same for all models. We will see below that stratification is allowed for nested and additive frailty models but not for joint frailty models.

```r
> mod.sha.str.gap <- frailtyPenal( Surv(time,event) ~ cluster(id) + as.factor(charlson) + as.factor(dukes) + chemo + strata(sex), Frailty=TRUE, n.knots=10, kappa1=2.11e+08, kappa2 = 2.11e+08, data=readmission)
> print(mod.sha.str.gap)
Call:
frailtyPenal(formula = Surv(time, event) ~ cluster(id) + as.factor(dukes) + as.factor(charlson) + strata(sex) + chemo, data = readmission, Frailty = TRUE, n.knots = 10, kappa1 = 2.11e+08, kappa2 = 2.11e+08)

Shared Gamma Frailty model parameter estimates using a Penalized Likelihood on the hazard function (Stratification structure used)

<table>
<thead>
<tr>
<th>Effect</th>
<th>coef</th>
<th>exp(coef)</th>
<th>SE coef (H)</th>
<th>SE coef (HIH)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>as.factor(dukes)=2</td>
<td>0.300</td>
<td>1.350</td>
<td>0.160</td>
<td>0.160</td>
<td>1.87</td>
<td>6.1e-02</td>
</tr>
<tr>
<td>as.factor(dukes)=3</td>
<td>1.071</td>
<td>2.918</td>
<td>0.195</td>
<td>0.195</td>
<td>5.50</td>
<td>3.8e-08</td>
</tr>
<tr>
<td>as.factor(charlson)=1</td>
<td>0.439</td>
<td>1.552</td>
<td>0.258</td>
<td>0.258</td>
<td>1.70</td>
<td>8.9e-02</td>
</tr>
<tr>
<td>as.factor(charlson)=3</td>
<td>0.404</td>
<td>1.497</td>
<td>0.137</td>
<td>0.137</td>
<td>2.95</td>
<td>3.2e-03</td>
</tr>
<tr>
<td>chemo</td>
<td>-0.214</td>
<td>0.807</td>
<td>0.143</td>
<td>0.143</td>
<td>-1.50</td>
<td>1.3e-01</td>
</tr>
</tbody>
</table>

Frailty parameter, Theta: 0.666 (SE (H): 0.141 ) (SE (HIH): 0.141 )

penalized marginal log-likelihood = -3230.02
n= 861
events= 458  n groups= 403
number of iterations: 16
Exact number of knots used: 10
Value of the smoothing parameter: 2.11e+08  2.11e+08, DoF: 11.00

In the dataset `readmission`, there is a death indicator named `death`. Thus, we can take into account the subject’s death by using a joint frailty model (death could be an informative
Joint frailty model

The following code is used for fitting a joint frailty model:

```r
> modJoint.gap <- frailtyPenal(Surv(time, event) ~ cluster(id) + as.factor(dukes) + as.factor(charlson) + sex + chemo + terminal(death), formula.terminalEvent = ~ as.factor(dukes) + as.factor(charlson) + sex + chemo, data=readmission, n.knots=8, kappa1=2.11e+08, kappa2=9.53e+11, Frailty=TRUE, joint=TRUE)
```

The smoothing parameter \( \kappa_1 \) is obtained from a shared frailty model with recurrent event as the outcome using the cross-validation method (See section 4.2 - output of the model mod.sha.gap). Similarly, the smoothing parameter \( \kappa_2 \) is obtained from a shared frailty model with death as the outcome using the cross-validation method.

```r
> print(modJoint.gap)
Call:
frailtyPenal(formula = Surv(time, event) ~ cluster(id) + as.factor(dukes) +
  as.factor(charlson) + sex + chemo + terminal(death), formula.terminalEvent = ~as.factor(dukes) +
  as.factor(charlson) + sex + chemo, data = readmission, joint = TRUE,
  n.knots = 10, kappa1 = 2.11e+08, kappa2 = 9.53e+11)
```

Joint gamma frailty model for recurrent and a terminal event processes using a Penalized Likelihood on the hazard function

Recurrences:

<table>
<thead>
<tr>
<th></th>
<th>coef</th>
<th>exp(coef)</th>
<th>SE coef (H)</th>
<th>SE coef (HIH)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>as.factor(dukes)2</td>
<td>0.356</td>
<td>1.427</td>
<td>0.164</td>
<td>0.164</td>
<td>2.16</td>
<td>3.0e-02</td>
</tr>
<tr>
<td>as.factor(dukes)3</td>
<td>1.288</td>
<td>3.624</td>
<td>0.207</td>
<td>0.207</td>
<td>6.21</td>
<td>5.2e-10</td>
</tr>
<tr>
<td>as.factor(charlson)1</td>
<td>0.406</td>
<td>1.500</td>
<td>0.254</td>
<td>0.254</td>
<td>1.59</td>
<td>1.1e-01</td>
</tr>
<tr>
<td>as.factor(charlson)3</td>
<td>0.406</td>
<td>1.500</td>
<td>0.137</td>
<td>0.137</td>
<td>2.96</td>
<td>3.1e-03</td>
</tr>
<tr>
<td>sex</td>
<td>-0.537</td>
<td>0.584</td>
<td>0.141</td>
<td>0.141</td>
<td>-3.82</td>
<td>1.4e-04</td>
</tr>
<tr>
<td>chemo</td>
<td>-0.153</td>
<td>0.771</td>
<td>0.145</td>
<td>0.145</td>
<td>-1.06</td>
<td>2.9e-01</td>
</tr>
</tbody>
</table>

Terminal event:

<table>
<thead>
<tr>
<th></th>
<th>coef</th>
<th>exp(coef)</th>
<th>SE coef (H)</th>
<th>SE coef (HIH)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>as.factor(dukes)2</td>
<td>1.384</td>
<td>3.990</td>
<td>0.358</td>
<td>0.354</td>
<td>3.870</td>
<td>1.1e-04</td>
</tr>
<tr>
<td>as.factor(dukes)3</td>
<td>3.376</td>
<td>29.253</td>
<td>0.464</td>
<td>0.461</td>
<td>7.269</td>
<td>3.6e-13</td>
</tr>
<tr>
<td>as.factor(charlson)1</td>
<td>0.532</td>
<td>1.702</td>
<td>0.643</td>
<td>0.643</td>
<td>0.826</td>
<td>4.1e-01</td>
</tr>
<tr>
<td>as.factor(charlson)3</td>
<td>1.299</td>
<td>3.667</td>
<td>0.264</td>
<td>0.263</td>
<td>4.929</td>
<td>8.3e-07</td>
</tr>
<tr>
<td>sex</td>
<td>-0.260</td>
<td>0.771</td>
<td>0.234</td>
<td>0.233</td>
<td>-1.113</td>
<td>2.7e-01</td>
</tr>
</tbody>
</table>
The interest of this model is that we can see the covariates’ effect for the two events (recurrent and terminal).

In this model, for instance, a chemotherapy does not seem to be a useful treatment to decrease relapses (p=0.29). However, chemotherapy was positively associated with death, i.e., chemotherapy treatment increased significantly the risk of death (p=1.2e-05 < 0.05 and the hazard ratio is 3.2). Similar conclusions about the influence of chemotherapy or gender are not possible with the previous models (Cox, shared or shared + stratification). Indeed, only one event of interest is analyzed using these models whereas joint frailty models allow to link two events of interest. The variance of the frailty (\( \theta \)): \( \theta = 0.733 \) (SE (H): 0.106) means that there is heterogeneity between subjects explained by non-observed covariates. The positive value of the coefficient \( \alpha \): \( \alpha = 0.981 \) (SE (H): 0.303) in the joint model indicates that the incidence of recurrences is positively associated with death. We can deduce this by using the modified Wald test: \( Wm = 0.733/0.106 = 6.92 > 1.64 \) for the frailty variance. For the coefficient \( \alpha \) we do a classical Wald test: \( W = 0.981/0.303 = 3.24 > 1.96 \).

### Nested frailty model

The following code is an example for fitting a nested frailty model. An extract from the dataset dataNested is provided in Table 2 of the previous section.

```r
> library(frailtypack)
> data(dataNested)
> modNested <- frailtyPenal( Surv(t1, t2, event) ~ cluster(group) + subcluster(subgroup) + cov1 + cov2, data=dataNested, n.knots=8, kappa1=10000, cross.validation=TRUE)

> print(modNested)
Call:
frailtyPenal(formula = Surv(t1, t2, event) ~ cluster(group) + subcluster(subgroup) + cov1 + cov2, data = dataNested,
  n.knots = 8, kappa1 = 10000, cross.validation = TRUE)

Frailty parameters:
theta (variance of Frailties, Z): 0.733 (SE (H): 0.106) (SE (HIH): 0.106)
a alpha (Z^alpha for terminal event): 0.981 (SE (H): 0.303) (SE (HIH): 0.303)
penalized marginal log-likelihood = -3970.98
n = 861
n recurrent events= 458 n groups= 403
n terminal events= 109
number of iterations: 14
Exact number of knots used: 10
Value of the smoothing parameters: kappa1=2.11e+08 and kappa2=9.53e+11
left truncated structure used

Nested Frailty model parameter estimates using
a Penalized Likelihood on the hazard functions

<table>
<thead>
<tr>
<th></th>
<th>coef</th>
<th>exp(coef)</th>
<th>SE coef</th>
<th>(H) SE coef</th>
<th>(HIH)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>cov1</td>
<td>-0.552</td>
<td>0.576</td>
<td>0.132</td>
<td>0.131</td>
<td>-4.19</td>
<td>2.8e-05</td>
<td></td>
</tr>
<tr>
<td>cov2</td>
<td>1.276</td>
<td>3.581</td>
<td>0.147</td>
<td>0.146</td>
<td>8.66</td>
<td>0.0e+00</td>
<td></td>
</tr>
</tbody>
</table>

Frailty parameters:
alpha (group effect): 0.374 (SE(H):0.108) (SE(HIH):0.107)
eta (subgroup effect): 0.106 (SE(H):0.0708) (SE(HIH):0.07)

penalized marginal log-likelihood = -1021.41
n= 400
n events= 287 n groups= 20
number of iterations: 18
Exact number of knots used: 8
Value of the smoothing parameter: 60439

Using the modified Wald test, we can deduce that the group effect \((Wm = 0.374/0.108 = 3.46 > 1.64)\) is significant whereas the subgroup effect is not \((Wm = 0.106/0.0708 = 1.50 < 1.64)\). So in this case a simple shared frailty model would be sufficient.

Additive frailty models with no correlation between the random effects

In the model \(\text{modAdd2cov.withoutCorr}\), we suppose that the random effects are not correlated \((\text{correlation}=\text{FALSE})\). An extract from the dataset \(\text{dataAdditive}\) is provided in Table 3 of the previous section.

> library(frailtypack)
> data(dataAdditive)
> dataAdditive$var2<-rbinom(nrow(dataAdditive),1,0.5)

> modAdd2cov.withoutCorr <- additivePenal( Surv(t1,t2,event) ~ cluster(group) + var1 + var2 + slope(var1), cross.validation=TRUE, correlation=FALSE, data=dataAdditive, n.knots=10, kappa1=1)

> print(modAdd2cov.withoutCorr)
Call:
additivePenal(formula = Surv(t1, t2, event) ~ cluster(group) + var1 + var2 + slope(var1), data = dataAdditive, correlation = FALSE, cross.validation = TRUE, n.knots = 10, kappa1 = 1)
Additive gaussian frailty model parameter estimates using a Penalized Likelihood on the hazard function

<table>
<thead>
<tr>
<th>coef</th>
<th>exp(coef)</th>
<th>SE coef (H)</th>
<th>SE coef (HIH)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>var1</td>
<td>-0.1949</td>
<td>0.823</td>
<td>0.0586</td>
<td>-3.328</td>
<td>0.00087</td>
</tr>
<tr>
<td>var2</td>
<td>-0.0243</td>
<td>0.976</td>
<td>0.0247</td>
<td>-0.987</td>
<td>0.3200</td>
</tr>
</tbody>
</table>

Variance for random intercept: 0.327 (SE (H): 0.0527) (SE (HIH): 0.0527)
Variance for random slope: 0.285 (SE (H): 0.0509) (SE (HIH): 0.051)

penalized marginal log-likelihood = -30154.38
n= 10000
n events= 6887 n groups= 100
number of iterations: 9
Exact number of knots used: 10
Smoothing parameter estimated by Cross validation: 16735

We can see that the two variances are significantly different from 0, which means that there is heterogeneity between trials and heterogeneity of the treatment effect across trials:
Wm=0.327/0.0527 = 6.20 > 1.64 for the random intercept variance and Wm=0.285/0.0509 = 5.58 > 1.64 for the random slope variance.
If the variance for the random slope was not significantly different from 0 then a simple shared frailty model would be sufficient.

Additive frailty models with a correlation between the random effects

If we suppose that the two randoms effects are correlated, we have to change the option correlation by: correlation=TRUE.

```r
> modAdd2cov.withCorr <- additivePenal( Surv(t1,t2,event) ~ cluster(group) + var1 + var2 + slope(var1), cross.validation=TRUE, data=dataAdditive, correlation=TRUE, n.knots=10, kappa1=1)
> print(modAdd2cov.withCorr)
```

Call:
additivePenal(formula = Surv(t1, t2, event) ~ cluster(group) + var1 + var2 + slope(var1), data = dataAdditive, correlation = TRUE, cross.validation = TRUE, n.knots = 10, kappa1 = 1)

Additive gaussian frailty model parameter estimates using a Penalized Likelihood on the hazard function
FRAILTYPACK: An R package for frailty models

<table>
<thead>
<tr>
<th></th>
<th>coef</th>
<th>exp(coef)</th>
<th>SE</th>
<th>coef (H)</th>
<th>SE</th>
<th>coef (HIH)</th>
<th>z</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>var1</td>
<td>-0.1901</td>
<td>0.827</td>
<td>0.0618</td>
<td>0.0618</td>
<td>0.0618</td>
<td>-3.076</td>
<td>0.0021</td>
<td></td>
</tr>
<tr>
<td>var2</td>
<td>-0.0212</td>
<td>0.979</td>
<td>0.0246</td>
<td>0.0246</td>
<td>0.0246</td>
<td>-0.862</td>
<td>0.3900</td>
<td></td>
</tr>
</tbody>
</table>

Covariance (between the two frailty terms, the intercept and the slope): -0.227 (SE: 0.047)
Corresponding correlation between the two frailty terms: -0.677
Variance for random intercept: 0.358 (SE (H): 0.0567) (SE (HIH): 0.0567)
Variance for random slope: 0.314 (SE (H): 0.0546) (SE (HIH): 0.0546)

penalized marginal log-likelihood = -30125.31
n= 10000
n events= 6887 n groups= 100
number of iterations: 7
Exact number of knots used: 10
Smoothing parameter estimated by Cross validation: 16735

The covariance between the two frailty terms (= -0.227 (SE: 0.047)) is significantly different from 0 (Wm = -0.227/0.047 = 4.83 > 1.64). So these two random effects are not independent. The parameter estimates for an additive frailty model with correlated random effects have slightly changed.

4.3. Print the hazard ratios

The package allows to estimate the hazard ratios using the R function named summary.

> summary(model,level=0.95)

The option level=0.95 indicates the level of confidence (here it is 95 percent). For instance, the summary function for the joint frailty models gives the following output:

> summary(modJoint.gap,level=0.95)

Recurrences:

<table>
<thead>
<tr>
<th></th>
<th>hr</th>
<th>95% C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td>as.factor(dukes)2</td>
<td>1.43</td>
<td>(1.03 - 1.97)</td>
</tr>
<tr>
<td>as.factor(dukes)3</td>
<td>3.62</td>
<td>(2.41 - 5.44)</td>
</tr>
<tr>
<td>as.factor(charlson)1</td>
<td>1.50</td>
<td>(0.91 - 2.47)</td>
</tr>
<tr>
<td>as.factor(charlson)3</td>
<td>1.50</td>
<td>(1.15 - 1.96)</td>
</tr>
<tr>
<td>sex</td>
<td>0.58</td>
<td>(0.44 - 0.77)</td>
</tr>
<tr>
<td>chemo</td>
<td>0.86</td>
<td>(0.65 - 1.14)</td>
</tr>
</tbody>
</table>
Terminal event:
---------------

<table>
<thead>
<tr>
<th></th>
<th>hr</th>
<th>95% C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td>as.factor(dukes)2</td>
<td>3.99</td>
<td>(1.98 - 8.04)</td>
</tr>
<tr>
<td>as.factor(dukes)3</td>
<td>29.25</td>
<td>(11.77 - 72.69)</td>
</tr>
<tr>
<td>as.factor(charlson)1</td>
<td>1.70</td>
<td>(0.48 - 6.00)</td>
</tr>
<tr>
<td>as.factor(charlson)3</td>
<td>3.67</td>
<td>(2.19 - 6.15)</td>
</tr>
<tr>
<td>sex</td>
<td>0.77</td>
<td>(0.49 - 1.22)</td>
</tr>
<tr>
<td>chemo</td>
<td>3.26</td>
<td>(1.92 - 5.52)</td>
</tr>
</tbody>
</table>

4.4. Draw the survival or hazard baseline functions

With frailtypack survival or hazard baseline functions may be drawn using an adapted R function named plot. The same code is used to draw hazard baseline functions for a Cox model, a shared, a nested or an additive frailty model:

```r
> plot(model, type.plot="hazard", level=0.95, conf.bands=TRUE,
       pos.legend="topright", cex.legend=0.7)
```

conf.bands=TRUE means that the confidence bands are drawn too. To draw survival baseline functions, the option type.plot has to be replaced by: type.plot="survival". The location of the legend in the graph can be specified by setting the argument pos.legend to a single keyword from the list "bottomright", "bottom", "bottomleft", "left", "topleft", "top", "topright", "right" and "center". The default is "topright". The argument cex.legend allows to change the size of the legend which could otherwise hide the curves, the default value is 0.7.

Figure 1 represents the baseline survival functions for the Cox model or the shared frailty model with and without stratification.

```r
> plot(mod.cox_gap,type.plot="survival",main="Cox model",conf.bands=TRUE)
> plot(mod.sha_gap,type.plot="survival",main="Shared ",conf.bands=TRUE)
> plot(mod.sha.str_gap,type.plot="survival",main="Shared + Stratification",
      conf.bands=TRUE,pos.legend="bottomleft",cex.legend=1)
```

Figure 2 represents the baseline hazard functions for the additive frailty models as previously fitted.

```r
plot(modAdd2cov.withoutCorr,type.plot="hazard",main="Correlation=False",
     conf.bands=TRUE)
plot(modAdd2cov.withCorr,type.plot="hazard",main="Correlation=True",
     conf.bands=TRUE)
```

For nested frailty models, the same as the cox, shared or additive frailty models may be done.
Figure 1: Baseline survival functions (Cox model, Shared frailty model)

Figure 2: Baseline hazard functions (additive frailty model)
Figure 3: Survival baseline functions for death and recurrent events (joint frailty model)

> plot(modNested, type.plot="hazard", conf.bands=TRUE)

For a joint frailty model, a supplementary option exists called `event` allows to draw the baseline functions for the recurrent event only (`event="recurrent"`), the terminal event only (`event="terminal"`) or both (`event="both"`). The code for drawing survival baseline functions for both these rates is:

> plot(modJoint_gap,type.plot="survival",event="recurrent",main="Recurrent", conf.bands=TRUE,pos.legend="topleft",cex.legend=1,ylim=c(0,2))
> plot(modJoint_gap,type.plot="survival",event="terminal",main="Terminal", conf.bands=TRUE,pos.legend="topleft",cex.legend=1,ylim=c(0,2))
> plot(modJoint_gap,type.plot="survival",event="both",main="Both", conf.bands=TRUE,pos.legend="topleft",cex.legend=1,ylim=c(0,2))

Figure 3 represents baseline survival functions for death and recurrent events (joint frailty model).

5. Conclusion

The new version of the package `frailtypack` allows to deal with correlated survival data using shared, nested, joint, additive frailty models or the Cox model. These models can be used when survival data are clustered into several levels, when the terminal event is considered as an informative censoring data, for meta-analysis studies or multicentric datasets. This article shows how to build the database according to the model, how to code for getting parameter estimates, hazard ratios and hazard or survival function curves and how to interpret the results.

A set of models can be fitted with `frailtypack`. In a family of frailty models, for instance in the joint frailty models family we might include different types of random effects, we then obtain several candidate models from which a best-fitting model need to be selected. The Likelihood Cross-Validation criterion (LCV) is adopted here to guide the choice of the model used in this analysis. As LCV is particularly computationally demanding when \( n \) is large, an approximate
version $LCV_a$ has been proposed by O’Sullivan (O Sullivan 1988) for the estimation of the hazard function in a survival case and adapted recently by Commenges et al. (Commenges, Joly, Gégout-Petit, and Liquet 2007). Lower values of $LCV_a$ indicate a better fitting model. The $LCV_a$ is then defined as:

$$LCV_a = \frac{1}{n}(trace(H^{-1}_a H_l) - l(.)$$

with $H_{pl}$ minus the converged hessian of the penalized log-likelihood, $H_l$ minus the converged hessian of the log-likelihood and $l(.)$ is the full log-likelihood. In the case of a parametric approach $trace(H^{-1}_{pl} H_l) - l(.)$ will represent the number of parameters and $LCV_a$ will be approximately equivalent to the AIC criterion. The $LCV_a$ for parametric approaches is defined as:

$$LCV = \frac{1}{n}(np - l(.))$$

with $np$ the total number of parameters. The inclusion of the $LCV_a$ criteria is currently in progress and will be include in the package in the near future frailtypack.

By developing the frailtypack package in R we hope to have provided a useful, easy and pertinent tool which address many biomedical issues. We plan to include in the near future parametric hazard functions and prediction estimations in the package and to extend it regularly by adding more functions and other frailty models.

**References**


Affiliation:

Virginie RONDEAU
INSERM CR897 (Biostatistics Team) Université Victor Segalen Bordeaux 2
146, rue Léo Saignat, 33076 Bordeaux Cedex, France
Telephone: +33/5/57/57/45/31
Fax: +33/5/56/24/00/81
E-mail: Virginie.Rondeau@isped.u-bordeaux2.fr

Yassin MAZROUI
INSERM CR897 (Biostatistics Team) Université Victor Segalen Bordeaux 2
146, rue Léo Saignat, 33076 Bordeaux Cedex, France
Telephone: +33/5/57/57/11/36
Fax: +33/5/56/24/00/81
E-mail: Yassin.Mazroui@isped.u-bordeaux2.fr

Juan R. GONZALEZ
Centre for Research in Environmental Epidemiology (CREAL)
Biomedical Park Research of Barcelona (PRBB)
Avda. Dr Aiguader, 88 Barcelona 08003, Spain
Telephone: +34/93/214/73/27
Fax: +34/93/214/73/02
E-mail: jrgonzalez@creal.cat
URL: http://www.creal.cat/jrgonzalez/software.htm

Journal of Statistical Software
published by the American Statistical Association
Volume VV, Issue II
Submitted: yyyy-mm-dd
Accepted: yyyy-mm-dd