

A copula-based simulation method for clustered multi-state survival data

F. Rotolo^{•*}, C. Legrand^{*}, I. Van Keilegom^{*}, M. Chiogna[•]

•Dipartimento di Scienze Statistiche



Università degli Studi di Padova



*Institut de Statistique, Biostatistique
et Sciences Actuarielles



Université Catholique de Louvain

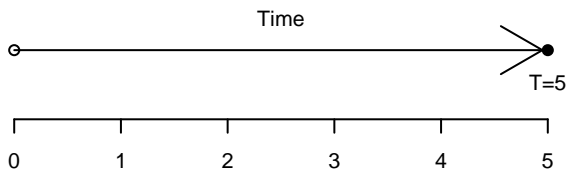


September 23, 2011

Survival Data

Time since an origin event until an event of interest.

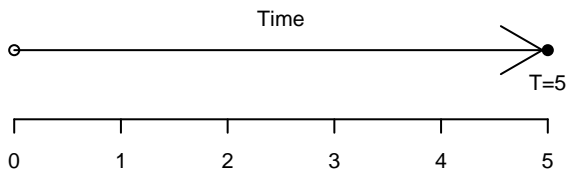
Example: from birth to death, since beginning of therapy until remission, etc.



Survival Data

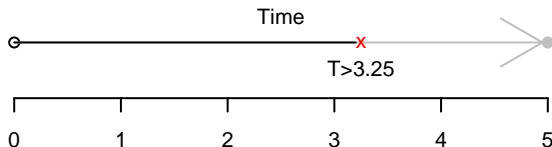
Time since an origin event until an event of interest.

Example: from birth to death, since beginning of therapy until remission, etc.



Censoring: some observations cannot be observed, the only available information being a lower bound.

Example: migration, change of therapy, loss to follow-up, etc.



Modeling Survival Data

Because of this peculiarity, instead of modeling the density $f(t)$ of T , the **hazard** is considered

$$h(t) = \lim_{\Delta t \searrow 0} \frac{\mathbb{P}[t \leq T < t + \Delta t | T \geq t]}{\Delta t} = \frac{f(t)}{S(t)} = -\frac{d}{dt} \log S(t),$$

with $S(t) = \int_t^\infty f(u)du = \mathbb{P}[T > t]$.

Note: $S(t) = \exp\{-\int_0^t h(u)du\}$.

Modeling Survival Data

Because of this peculiarity, instead of modeling the density $f(t)$ of T , the **hazard** is considered

$$h(t) = \lim_{\Delta t \searrow 0} \frac{\mathbb{P}[t \leq T < t + \Delta t | T \geq t]}{\Delta t} = \frac{f(t)}{S(t)} = -\frac{d}{dt} \log S(t),$$

with $S(t) = \int_t^\infty f(u)du = \mathbb{P}[T > t]$.

Note: $S(t) = \exp\{-\int_0^t h(u)du\}$.

The basic regression model for the hazard is the **Proportional Hazards (PH) Model** (Cox, 1972)

$$h(t|X) = h_0(t) \exp\{\beta'X\}.$$

Survival Models

Complications of Cox models have been developed

Frailty Models (FMs)

account for overdispersion
or clustering by means
of random effects

$$h(t|X_{ij}) = h_0(t)Z_i e^{\beta' X_{ij}},$$

similar to GLMM

$$\log[h(t|X_{ij})] = \log[h_0(t)] + W_i + \beta' X_{ij},$$

$$\text{with } Z_i = e^{W_i}$$

(Duchateau & Janssen, 2008; Wienke, 2010)

Survival Models

Complications of Cox models have been developed

Frailty Models (FMs)

account for overdispersion
or clustering by means
of random effects

$$h(t|X_{ij}) = h_0(t)Z_i e^{\beta' X_{ij}},$$

similar to GLMM

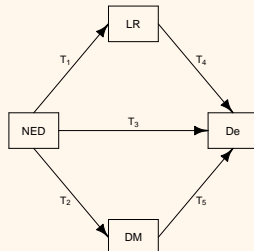
$$\log[h(t|X_{ij})] = \log[h_0(t)] + W_i + \beta' X_{ij},$$

$$\text{with } Z_i = e^{W_i}$$

(Duchateau & Janssen, 2008; Wienke, 2010)

Multi-State Models (MSMs)

consider several events
and their interactions



(Putter et al., 2007; de Wreede et al., 2010)

Survival Models

Complications of Cox models have been developed

Frailty Models (FMs)

account for overdispersion
or clustering by means
of random effects

$$h(t|X_{ij}) = h_0(t)Z_i e^{\beta' X_{ij}},$$

similar to GLMM

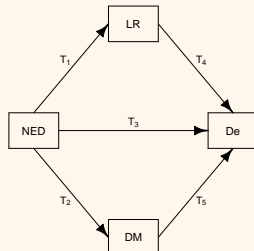
$$\log[h(t|X_{ij})] = \log[h_0(t)] + W_i + \beta' X_{ij},$$

$$\text{with } Z_i = e^{W_i}$$

(Duchateau & Janssen, 2008; Wienke, 2010)

Multi-State Models (MSMs)

consider several events
and their interactions



(Putter et al., 2007; de Wreede et al., 2010)

Possible integration?

Survival Models

Complications of Cox models have been developed

Frailty Models (FMs)

account for overdispersion
or clustering by means
of random effects

$$h(t|X_{ij}) = h_0(t)Z_i e^{\beta' X_{ij}},$$

similar to GLMM

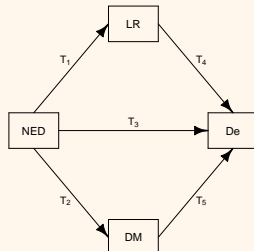
$$\log[h(t|X_{ij})] = \log[h_0(t)] + W_i + \beta' X_{ij},$$

$$\text{with } Z_i = e^{W_i}$$

(Duchateau & Janssen, 2008; Wienke, 2010)

Multi-State Models (MSMs)

consider several events
and their interactions



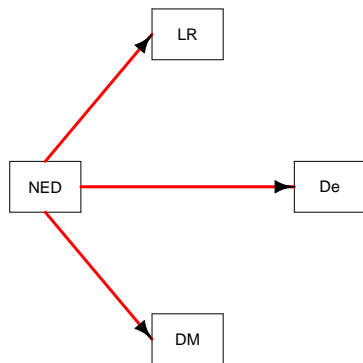
(Putter et al., 2007; de Wreede et al., 2010)

Possible integration? ► Simulation studies

Simulation of data

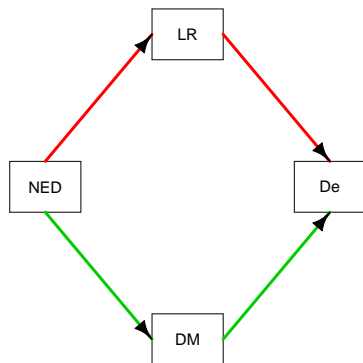
A simulation method should be able to generate

- ▶ the dependence of times of competing events



Simulation of data

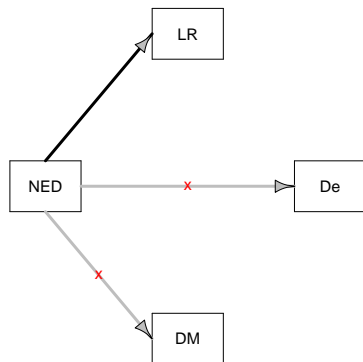
A simulation method should be able to generate



- ▶ the dependence of times of competing events
- ▶ the dependence of times of subsequent events

Simulation of data

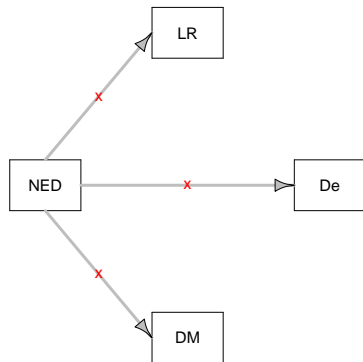
A simulation method should be able to generate



- ▶ the dependence of times of **competing events**
- ▶ the dependence of times of **subsequent events**
- ▶ the dependence between **clustered observations**
- ▶ the censoring due to competing events occurrence

Simulation of data

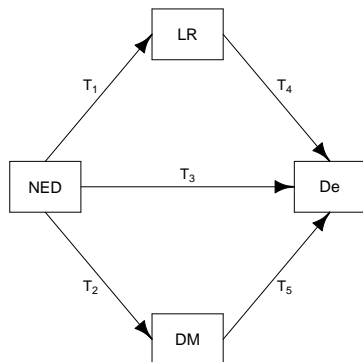
A simulation method should be able to generate



- ▶ the dependence of times of **competing events**
- ▶ the dependence of times of **subsequent events**
- ▶ the dependence between **clustered observations**
- ▶ the censoring due to competing events occurrence
- ▶ the censoring due to end of the study or loss to follow up

Simulation of data

A simulation method should be able to generate



- ▶ the dependence of times of **competing events**
- ▶ the dependence of times of **subsequent events**
- ▶ the dependence between **clustered observations**
- ▶ the censoring due to competing events occurrence
- ▶ the censoring due to end of the study or loss to follow up
- ▶ the event-specific covariates effect

Outline

Clustered Multi-State Survival Data

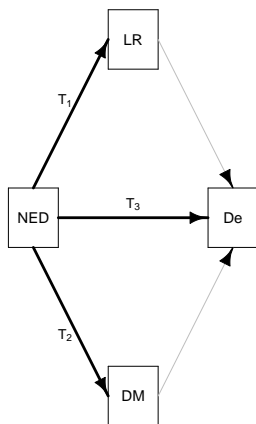
Simulation Algorithm

Clustering

Choice of Parameters

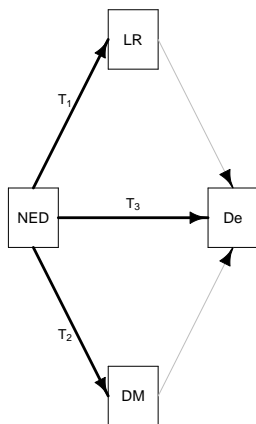
Example

Copula Model



- Marginal survival functions freely chosen $S_1(t)$, $S_2(t)$ and $S_3(t)$

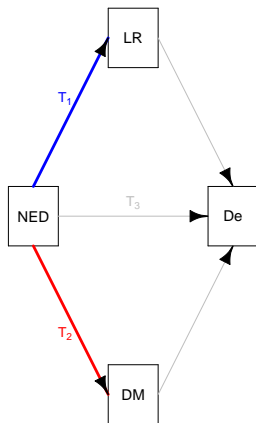
Copula Model



- ▶ Marginal survival functions freely chosen $S_1(t)$, $S_2(t)$ and $S_3(t)$
- ▶ Joint survival function by Clayton Copula

$$S_{123}(\mathbf{t}) = \left(\sum_{i=1}^3 S_i(t_i)^{-\theta} - 2 \right)^{-1/\theta}$$

Copula Model



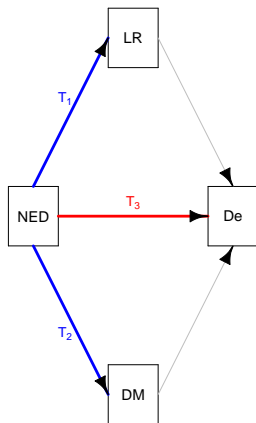
- Marginal survival functions freely chosen $S_1(t)$, $S_2(t)$ and $S_3(t)$

- Joint survival function by Clayton Copula $S_{123}(\mathbf{t}) = (\sum_{i=1}^3 S_i(t_i)^{-\theta} - 2)^{-1/\theta}$

- Conditional survivals from the joint

$$S_{2|1}(t_2|t_1) = \left[1 + \left(\frac{S_1(t_1)}{S_2(t_2)} \right)^\theta - S_1(t_1)^\theta \right]^{-1/\theta-1}$$

Copula Model



- Marginal survival functions freely chosen $S_1(t)$, $S_2(t)$ and $S_3(t)$

- Joint survival function by Clayton Copula $S_{123}(\mathbf{t}) = (\sum_{i=1}^3 S_i(t_i)^{-\theta} - 2)^{-1/\theta}$

- Conditional survivals from the joint

$$S_{2|1}(t_2|t_1) = \left[1 + \left(\frac{S_1(t_1)}{S_2(t_2)} \right)^\theta - S_1(t_1)^\theta \right]^{-1/\theta-1}$$

$$S_{3|12}(t_3|t_1, t_2) = \left(1 + \frac{S_3(t_3)^{-\theta} - 1}{S_1(t_1)^{-\theta} + S_2(t_2)^{-\theta} - 1} \right)^{-1/\theta-2}$$

Algorithm

Data from the copula model (Kpanzou, 2007) are simulated as follows

$$1 \blacktriangleright T_1 = S_1^{-1}(U_1)$$

with U_1, U_2, U_3, U_C i.i.d. $U(0, 1)$

Algorithm

Data from the copula model (Kpanzou, 2007) are simulated as follows

$$1 \blacktriangleright T_1 = S_1^{-1}(U_1)$$

$$2 \blacktriangleright T_2|t_1 = S_{2|1}^{-1}(U_2|t_1) =$$

$$S_2^{-1} \left(\left\{ \left[U_2^{-\frac{\theta}{1+\theta}} - 1 \right] S_1(t_1)^{-\theta} + 1 \right\}^{-1/\theta} \right)$$

with U_1, U_2, U_3, U_C i.i.d. $U(0, 1)$

Algorithm

Data from the copula model (Kpanzou, 2007) are simulated as follows

$$1 \blacktriangleright T_1 = S_1^{-1}(U_1)$$

$$2 \blacktriangleright T_2|t_1 = S_{2|1}^{-1}(U_2|t_1) =$$

$$S_2^{-1} \left(\left\{ \left[U_2^{-\frac{\theta}{1+\theta}} - 1 \right] S_1(t_1)^{-\theta} + 1 \right\}^{-1/\theta} \right)$$

$$3 \blacktriangleright T_3|t_1, t_2 = S_{3|12}^{-1}(U_3|t_1, t_2) =$$

$$S_3^{-1} \left(\left\{ \left[U_3^{-\frac{\theta}{1+2\theta}} - 1 \right] [S_1(t_1)^{-\theta} + S_2(t_2)^{-\theta} - 1] + 1 \right\}^{-1/\theta} \right)$$

with U_1, U_2, U_3, U_C i.i.d. $U(0, 1)$

Algorithm

Data from the copula model (Kpanzou, 2007) are simulated as follows

$$1 \blacktriangleright T_1 = S_1^{-1}(U_1)$$

$$2 \blacktriangleright T_2|t_1 = S_{2|1}^{-1}(U_2|t_1) =$$

$$S_2^{-1} \left(\left\{ \left[U_2^{-\frac{\theta}{1+\theta}} - 1 \right] S_1(t_1)^{-\theta} + 1 \right\}^{-1/\theta} \right)$$

$$3 \blacktriangleright T_3|t_1, t_2 = S_{3|12}^{-1}(U_3|t_1, t_2) =$$

$$S_3^{-1} \left(\left\{ \left[U_3^{-\frac{\theta}{1+2\theta}} - 1 \right] [S_1(t_1)^{-\theta} + S_2(t_2)^{-\theta} - 1] + 1 \right\}^{-1/\theta} \right)$$

$$C \blacktriangleright T_C = F_C^{-1}(U_C)$$

with U_1, U_2, U_3, U_C i.i.d. $U(0, 1)$

Algorithm

Data from the copula model (Kpanzou, 2007) are simulated as follows

$$1 \blacktriangleright T_1 = S_1^{-1}(U_1)$$

$$2 \blacktriangleright T_2|t_1 = S_{2|1}^{-1}(U_2|t_1) =$$

$$S_2^{-1} \left(\left\{ \left[U_2^{-\frac{\theta}{1+\theta}} - 1 \right] S_1(t_1)^{-\theta} + 1 \right\}^{-1/\theta} \right)$$

$$3 \blacktriangleright T_3|t_1, t_2 = S_{3|12}^{-1}(U_3|t_1, t_2) =$$

$$S_3^{-1} \left(\left\{ \left[U_3^{-\frac{\theta}{1+2\theta}} - 1 \right] [S_1(t_1)^{-\theta} + S_2(t_2)^{-\theta} - 1] + 1 \right\}^{-1/\theta} \right)$$

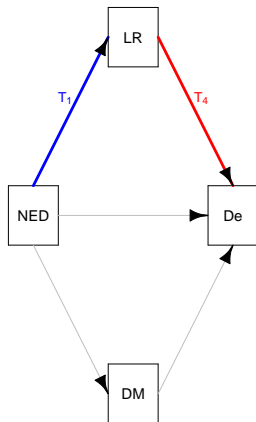
$$C \blacktriangleright T_C = F_C^{-1}(U_C)$$

$$T \blacktriangleright \min(T_C, T_1, T_2, T_3)$$

with U_1, U_2, U_3, U_C i.i.d. $U(0, 1)$

Second transitions

For patients with a transition into state **LR** or **DM**, an analogous copula model is used for second transition to state **De**



The following conditional survivals can be obtained

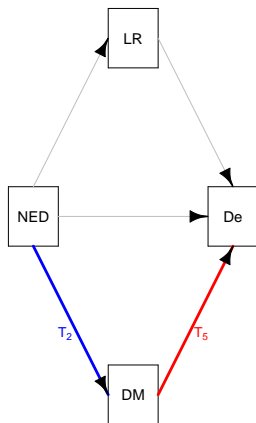
$$\blacktriangleright S_{4|1}(t_4|t_1) = \left[1 + \left(\frac{S_1(t_1)}{S_4(t_4)} \right)^\theta - S_1(t_1)^\theta \right]^{-1/\theta-1}$$

$$\blacktriangleright S_{5|2}(t_5|t_2) = \left[1 + \left(\frac{S_2(t_2)}{S_5(t_5)} \right)^\theta - S_2(t_2)^\theta \right]^{-1/\theta-1}$$

and the same algorithm is used to simulate second transition times, conditionally on first transition ones.

Second transitions

For patients with a transition into state **LR** or **DM**, an analogous copula model is used for second transition to state **De**



The following conditional survivals can be obtained

$$\blacktriangleright S_{4|1}(t_4|t_1) = \left[1 + \left(\frac{S_1(t_1)}{S_4(t_4)} \right)^\theta - S_1(t_1)^\theta \right]^{-1/\theta-1}$$

$$\blacktriangleright S_{5|2}(t_5|t_2) = \left[1 + \left(\frac{S_2(t_2)}{S_5(t_5)} \right)^\theta - S_2(t_2)^\theta \right]^{-1/\theta-1}$$

and the same algorithm is used to simulate second transition times, conditionally on first transition ones.

Clustering

The algorithm allows to freely specify the marginal survivals $S_i(t)$.
How can we insert clustering?

Clustering

The algorithm allows to freely specify the marginal survivals $S_i(t)$.
How can we insert clustering?

In a PH way

$$h_i(t|Z) = Z h_{0i}(t),$$

with $h_{0i}(t)$ the baseline hazard for transition i .

Clustering

The algorithm allows to freely specify the marginal survivals $S_i(t)$.
How can we insert clustering?

In a PH way

$$h_i(t|Z) = Z h_{0i}(t),$$

with $h_{0i}(t)$ the baseline hazard for transition i .

Since $S_{0i}(t) = \exp\{-\int_0^t h_{0i}(u)du\}$, then

$$S_i(t|Z) = \exp\left\{-Z \int_0^t h_{0i}(u)du\right\} = [S_{0i}(t)]^Z$$

Clustering

The algorithm allows to freely specify the marginal survivals $S_i(t)$.
How can we insert clustering?

In a PH way

$$h_i(t|Z) = Z h_{0i}(t),$$

with $h_{0i}(t)$ the baseline hazard for transition i .

Since $S_{0i}(t) = \exp\{-\int_0^t h_{0i}(u)du\}$, then

$$S_i(t|Z) = \exp\left\{-Z \int_0^t h_{0i}(u)du\right\} = [S_{0i}(t)]^Z$$

The copula model can be used for conditional survivals $\{S_i(t|Z)\}_{i \in \{1,2,3,4,5\}}$ and the same algorithm can be used, conditionally on Z .

Clustering and covariates

The effect of **covariates** X can be inserted in an analogous way.
The marginals are then

$$S_i(t|X, Z) = S_{0i}(t) Z e^{\beta_i' X}$$

and simulation via the copula model is done conditionally on (X, Z) .

The Clayton–Weibull model

Despite the model is quite general, we consider in the following a particular case:

- ▶ $T_i \sim \text{Wei}(\lambda_i, \rho_i)$, $i \in \{1, 2, 3, 4, 5\}$
- ▶ $T_C \sim \text{Wei}(\lambda_C, 1) \sim \text{Exp}(\lambda_C)$
- ▶ 72 months (6 years) of administrative censoring

The Clayton–Weibull model

Despite the model is quite general, we consider in the following a particular case:

- ▶ $T_i \sim \text{Wei}(\lambda_i, \rho_i)$, $i \in \{1, 2, 3, 4, 5\}$
- ▶ $T_C \sim \text{Wei}(\lambda_C, 1) \sim \text{Exp}(\lambda_C)$
- ▶ 72 months (6 years) of administrative censoring

This model

1. gives **simple** forms of **conditional distributions**
2. implies that $S_{i|X,Z}(t|x,z) = \exp\{-\lambda_i z e^{\beta_i^T x} t^{\rho_i}\}$,
i.e. $T_i|X, Z \sim \text{Wei}(\lambda_i z e^{\beta_i^T x}, \rho_i)$ is **still a Weibull r.v.**

Outline

Clustered Multi-State Survival Data

Simulation Algorithm

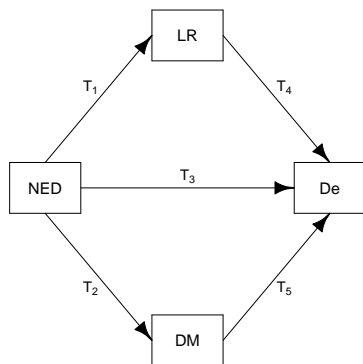
Clustering

Choice of Parameters

Example

Choice of parameters

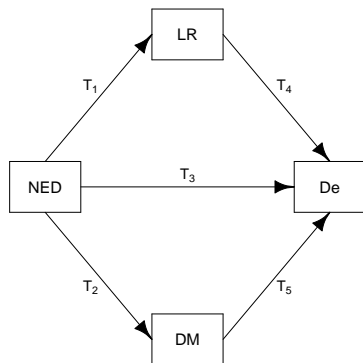
When simulating a dataset, one should be able to choose parameters in order to obtain particular **target values** for



p_i ► probabilities of **LR**, **DM**, **De** and censoring from **NED**

Choice of parameters

When simulating a dataset, one should be able to choose parameters in order to obtain particular **target values** for

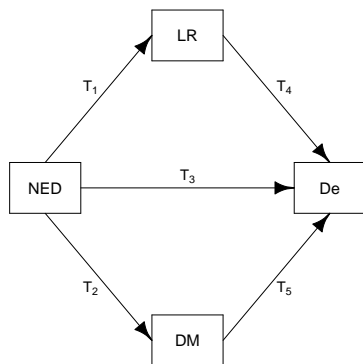


p_i ► probabilities of **LR**, **DM**, **De** and censoring from **NED**

m_i ► median of uncensored **LR**, **DM** and **De** times from **NED**

Choice of parameters

When simulating a dataset, one should be able to choose parameters in order to obtain particular **target values** for



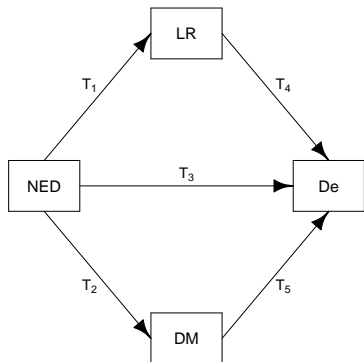
p_i ► probabilities of **LR**, **DM**, **De** and censoring from **NED**

m_i ► median of uncensored **LR**, **DM** and **De** times from **NED**

p_i ► probabilities of **De** and censoring from **LR** and from **DM**

Choice of parameters

When simulating a dataset, one should be able to choose parameters in order to obtain particular **target values** for



p_i ► probabilities of **LR**, **DM**, **De** and censoring from **NED**

m_i ► median of uncensored **LR**, **DM** and **De** times from **NED**

p_i ► probabilities of **De** and censoring from **LR** and from **DM**

m_i ► median of uncensored **De** times from **LR** and from **DM**

It is not possible to analytically express these quantities as functions of the parameters.

Criterion function

In order to find appropriate parameters for given **target values** $\{p_i, m_i\}$, we want to minimize the criterion function

$$\Upsilon(\Pi) = \sum_{i \in \{1,2,3,4,5\}} \left\{ \left[\log \frac{p_i}{\hat{p}_i(\Pi)} \right]^2 + \left[\log \frac{m_i}{\hat{m}_i(\Pi)} \right]^2 \right\} \\ \geq 0$$

with

$$\Pi = \{\lambda_i\}_{i \in \{1,2,3,C,4,C4,5,C5\}} \cup \{\rho_i\}_{i \in \{1,2,3,4,5\}} \in \mathbb{R}_+^{13}$$

and $\{\hat{p}_i(\Pi), \hat{m}_i(\Pi)\}$ the observed values in a simulated dataset with parameters Π

Criterion function

In order to find appropriate parameters for given **target values** $\{p_i, m_i\}$, we want to minimize the criterion function

$$\begin{aligned} \Upsilon(\Pi) &= \sum_{i \in \{1,2,3,4,5\}} \left\{ \left[\log \frac{p_i}{\hat{p}_i(\Pi)} \right]^2 + \left[\log \frac{m_i}{\hat{m}_i(\Pi)} \right]^2 \right\} \\ &= \Upsilon_{123}(\Pi_{123}) + \Upsilon_4(\Pi_4) + \Upsilon_5(\Pi_5) \geq 0 \end{aligned}$$

with

$$\begin{aligned} \Pi &= \{\lambda_i\}_{i \in \{1,2,3,C,4,5,C5\}} \cup \{\rho_i\}_{i \in \{1,2,3,4,5\}} \in \mathbb{R}_+^{13} \\ \Pi &= \Pi_{123} \cup \Pi_4 \cup \Pi_5 \in \mathbb{R}_+^7 \times \mathbb{R}_+^3 \times \mathbb{R}_+^3 \end{aligned}$$

and $\{\hat{p}_i(\Pi), \hat{m}_i(\Pi)\}$ the observed values in a simulated dataset with parameters Π

Criterion function

In order to find appropriate parameters for given target values $\{p_i, m_i\}$, we want to minimize the criterion function

$$\begin{aligned} \Upsilon(\Pi) &= \sum_{i \in \{1,2,3,4,5\}} \left\{ \left[\log \frac{p_i}{\hat{p}_i(\Pi)} \right]^2 + \left[\log \frac{m_i}{\hat{m}_i(\Pi)} \right]^2 \right\} \\ &= \Upsilon_{123}(\Pi_{123}) + \Upsilon_4(\Pi_4) + \Upsilon_5(\Pi_5) \geq 0 \end{aligned}$$

with

$$\begin{aligned} \Pi &= \{\lambda_i\}_{i \in \{1,2,3,C,4,C4,5,C5\}} \cup \{\rho_i\}_{i \in \{1,2,3,4,5\}} \in \mathbb{R}_+^{13} \\ \Pi &= \Pi_{123} \cup \Pi_4 \cup \Pi_5 \in \mathbb{R}_+^7 \times \mathbb{R}_+^3 \times \mathbb{R}_+^3 \end{aligned}$$

Further reduction of problem dimension...

$$\Pi = \Pi_{123} \cup \Pi_4 \cup \Pi_5 \in \mathbb{R}_+^{4+3} \times \mathbb{R}_+^{2+1} \times \mathbb{R}_+^{2+1}$$

and $\{\hat{p}_i(\Pi), \hat{m}_i(\Pi)\}$ the observed values in a simulated dataset with parameters Π

Minimization of criterion function

In order to further reduce the dimension of the problem, each of the parameter sets $\Pi_K, K \in \{\{123\}, \{4\}, \{5\}\}$ is split into the scale $\{\lambda_i\}$ and the shape parameters $\{\rho_i\}$. The optimization of the criterion function $\Upsilon_K(\Pi_K)$ is iterated on each subset

Example: algorithm for $K = \{123\}$

► Set $J = 1$

$$\lambda^{(0)} = \{\lambda_i^{(0)}\}_{i \in \{C,1,2,3\}} = \{1, 1, 1, 1\}$$

$$\rho^{(0)} = \{\rho_i^{(0)}\}_{i \in \{1,2,3\}} = \{1, 1, 1\}$$

Minimization of criterion function

In order to further reduce the dimension of the problem, each of the parameter sets $\Pi_K, K \in \{\{123\}, \{4\}, \{5\}\}$ is split into the scale $\{\lambda_i\}$ and the shape parameters $\{\rho_i\}$. The optimization of the criterion function $\Upsilon_K(\Pi_K)$ is iterated on each subset

Example: algorithm for $K = \{123\}$

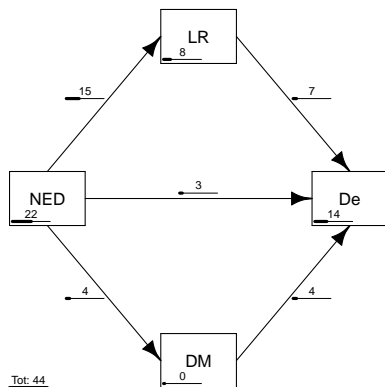
- ▶ Set $J = 1$
 - $\lambda^{(0)} = \{\lambda_i^{(0)}\}_{i \in \{C,1,2,3\}} = \{1, 1, 1, 1\}$
 - $\rho^{(0)} = \{\rho_i^{(0)}\}_{i \in \{1,2,3\}} = \{1, 1, 1\}$
- ▶ Repeat until $J = \text{maxit}$ or $\Upsilon_{123}(\lambda^{(J-1)}, \rho^{(J-1)}) < \text{th}$
 - ▶ Obtain $\lambda^{(J)}$ by minimizing $\Upsilon_{123}(\lambda, \rho^{(J-1)})$ over λ
 - ▶ Obtain $\rho^{(J)}$ by minimizing $\Upsilon_{123}(\lambda^{(J)}, \rho)$ over ρ
 - ▶ Set $J = J + 1$

where maxit and th are arbitrary termination parameters.

An example

A dataset of size 44 is available from a multi-center study on head and neck cancer.

Target values $\{p_i\}$ and $\{m_i\}$



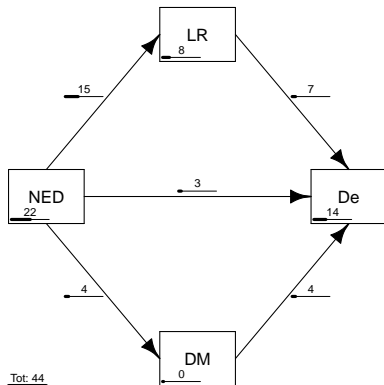
An example

A dataset of size 44 is available from a multi-center study on head and neck cancer.

Target values $\{p_i\}$ and $\{m_i\}$

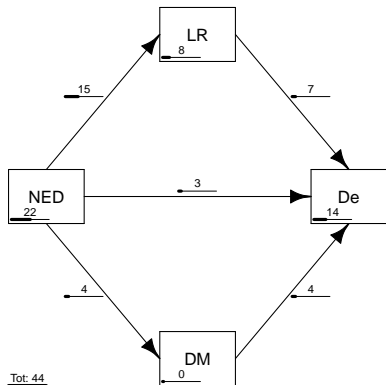
Frailty term

- ▶ 40 Hospitals
- ▶ random sizes
- ▶ $Z \sim \text{Gam}(1, 0.5)$



An example

A dataset of size 44 is available from a multi-center study on head and neck cancer.



Target values $\{p_i\}$ and $\{m_i\}$

Frailty term

- ▶ 40 Hospitals
- ▶ random sizes
- ▶ $Z \sim \text{Gam}(1, 0.5)$

Covariates

- ▶ **Age** $\sim \mathcal{N}(60, 7)$
with

$$\beta_{i,\text{Age}} = \begin{cases} \log(0.8)/10 & i = 1 \\ \log(0.9)/10 & i = 2 \\ \log(1.2)/10 & i = 3, 4, 5 \end{cases}$$

- ▶ **Treat** $\sim \text{Bin}(0.5)$
with

$$\beta_{i,\text{Treat}} = \begin{cases} \log(1/3) & i = 1 \\ 0 & i = 2 \\ \log(1.2) & i = 3, 4, 5 \end{cases}$$

Results

First transitions. The algorithm is run with datasets of size 10^4 , $\text{maxit} = 10$ and $\text{th} = 0.1$. The time of execution was 11:57' hours

NED \rightarrow {LR,DM,De}

λ_1	λ_2	λ_3	λ_C	ρ_1	ρ_2	ρ_3
0.276	0.019	0.013	0.031	0.851	1.076	0.569

Results

First transitions. The algorithm is run with datasets of size 10^4 , $\text{maxit} = 10$ and $\text{th} = 0.1$. The time of execution was 11:57' hours

NED \rightarrow {LR,DM,De}

λ_1	λ_2	λ_3	λ_C	ρ_1	ρ_2	ρ_3
0.276	0.019	0.013	0.031	0.851	1.076	0.569

NED \rightarrow {LR,DM,De}

	p_i				m_i		
	LR	DM	De	C	LR	DM	De
Target	0.34	0.09	0.07	0.50	6.00	10.00	3.00
Simulated	0.33	0.12	0.09	0.46	5.41	9.33	2.29

$$\Upsilon_{123}(\Pi_{123}) = 0.24$$

Results

Second transitions. Conditionally on first transitions data, the algorithm is run for second transitions from **LR** and **DM** with $\text{maxit} = 6$ and $\text{th} = 0.05$. The times of execution were 4:31' and 3:57' hours, respectively.

LR→De			DM→De		
λ_4	λ_{C4}	ρ_4	λ_5	λ_{C5}	ρ_5
0.029	0.099	1.078	0.192	0.039	1.000

Results

Second transitions. Conditionally on first transitions data, the algorithm is run for second transitions from **LR** and **DM** with $\text{maxit} = 6$ and $\text{th} = 0.05$. The times of execution were 4:31' and 3:57' hours, respectively.

LR→De			DM→De		
λ_4	λ_{C4}	ρ_4	λ_5	λ_{C5}	ρ_5
0.029	0.099	1.078	0.192	0.039	1.000

LR→De			Target	DM→De		
p_i	m_i	p_i		m_i		
De	C	De	De	C	De	
0.53	0.47	3.25	0.95	0.05	0.50	
0.50	0.50	3.32	Simulated	0.97	0.03	0.54

$$\Upsilon_4(\Pi_4) = 0.0043$$

$$\Upsilon_5(\Pi_5) = 0.0064$$

Conclusion

The proposed simulation procedure for **clustered MS** allows to

- MSMs◀ generate dependence between times of the **same subject**
(between both competing and subsequent event times)
- FMs◀ generate dependence between times of **clustered subjects**
(with arbitrary number and size of groups and free frailty distribution)
- PH◀ insert **covariates** via proportional hazards
- parMod◀ choose **marginal distributions** of time variables

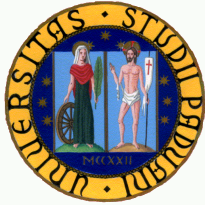
Conclusion

The proposed simulation procedure for **clustered MS** allows to

- MSMs◀ generate dependence between times of the **same subject**
(between both competing and subsequent event times)
 - FMs◀ generate dependence between times of **clustered subjects**
(with arbitrary number and size of groups and free frailty distribution)
 - PH◀ insert **covariates** via proportional hazards
 - parMod◀ choose **marginal distributions** of time variables
-
- ▶ **automatically find** appropriate **parameters**, given arbitrary **target values** for probabilities of censoring, of competing events and for medians of uncensored times
 - ▶ generate **censoring**, both random and administrative

References

- COX, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)* **34**, 187–220.
- DE WREEDE, L. C., FIOCCO, M. & PUTTER, H. (2010). The mstate package for estimation and prediction in non- and semi-parametric multi-state and competing risks models. *Comput Methods Programs Biomed* **99**, 261–74.
- DUCHATEAU, L. & JANSSEN, P. (2008). *The frailty model*. Springer.
- KPANZOU, T. A. (2007). Copulas in statistics. *African Institute for Mathematical Sciences (AIMS)* .
- PUTTER, H., FIOCCO, M. & GESKUS, R. B. (2007). Tutorial in biostatistics: competing risks and multi-state models. *Stat Med* **26**, 2389–430.
- WIENKE, A. (2010). *Frailty Models in Survival Analysis*. Chapman & Hall/CRC biostatistics series. Taylor and Francis.



F. Rotolo [federico.rotolo@stat.unipd.it – federico.rotolo@uclouvain.be]
PhD Student at University of Padova and Visiting PhD Student at UCL

under the supervision of

prof. **C. Legrand**, UCL

prof. **I. Van Keilegom**, UCL

prof. **M. Chiogna**, UniPd