Università degli studi di Padova
Dipartimento di Scienze Statistiche
Corso di laurea magistrale in
Scienze Statistiche

Alternative ways of estimating missing time to complete remission for AML patients

Relatore Prof. Guido Masarotto
Dipartimento di Scienze Statistiche
Correlatore Prof. Marta Fiocco
Department Medical Statistics and Bioinformatics (Leiden, NL)

Laureando: Alice Baccino
Matricola N 1013749

Anno Accademico 2012/2013

Contents

Introduction

Overview

The motivating example for this thesis comes from a retrospective study concerning children suffering from acute myeloid leukemia (AML).
AML is a cancer of the myeloid line of the blood cells, characterized by an abnormal increase of white blood cells called blast. The disease progresses rapidly and can be fatal within weeks.
Clinicians are interested in estimating survival indicators such as overall survival, event free survival and cumulative incidence which is the probability $F(t)$ of dying before time t.
The ultimate goal in any type of cancer is the achievement of complete remission. A patients is in complete remission if all signs of cancer have disappeared (according to some criteria developed by an International Working Group). Once a patient is in complete remission, it may happen that the disease comes back or in clinical terms that an individual experiences relapse. This implies that it is extremely important for clinicians to be able to estimate the survival indicators above introduced from time to achievement of complete remission.
The object under investigation is the time elapsed from the initiating event complete remission, to relapse or death. It is straightforward to estimate the quantity of interest by employing the classical survival analysis methodologies, but in the AML retrospective study for about 40% of the patients time to complete remission from diagnosis time is unknown. The only piece of information available is that the event complete remission has occurred in the past but it is not known at which time.
The challenge in this thesis is to estimate the statistics of interest by first reconstructing the missing time to the initiating event (complete remission) and then moving to the estimation process in the presence of censored observations. For the last aspect to be feasible, censoring must be independent, that is, an individual censored at time t should be representative for those still at risk at that time. In other words, those censored should not be indi-

viduals with systematically high or low risk of relapsing or dying.

Thesis contribution

In this thesis parametric and non parametric methods to estimate the unknown time to complete remission are provided.
The first methodology explored is multiple imputation which is a techniques born in the late 70's originally proposed by Rubin in 1977. The basic idea consists in replacing the missing value, which in the case under study is the unknown time to complete remission, with a set of plausible simulated values.
Several complete data sets are then reconstructed by employing an imputation model. Complete data sets are analyzed by standard methods and the results are combined together by inference techniques that take into account the uncertainty due to imputation.
The novelty of this thesis consists in proposing a non parametric multiple imputation algorithm for estimating missing values. The procedure is based on sampling values from the observed time to complete remission with probability computed according to the empirical cumulative distribution.
The advantage of the a non parametric multiple imputation algorithm is that no assumption about the underlying distribution of the missing time to complete remission are made and the data are not forced to follow a specific distribution.
In survival analysis several parametric models such us exponential, Weibull, gamma, log normal and Gompertz, are widely used to describe time to event data. For this reason, along with the non parametric approach, two parametric methodologies have been proposed for dealing with the unknown time to compete remission.
By following the parametric approach, theoretical parametric distributions are chosen to describe time to event. Several events are of interest in this contest: time to complete remission from diagnosis, time to relapse or death from the initiating event complete remission.
The challenging part is to find an appropriate method for reconstructing the missing time to complete remission from the initiating event diagnosis for about 40% of the individuals in the study case. The missing informations are incorporated in the likelihood function by integrating out the likelihood for all possible values that time to complete remission from diagnosis time may assume. Since the likelihood function has an intractable form numerical methods to estimate the parameters need to be used.
Within the parametric approach the expectation-maximization (EM) method-

ology is also proposed to deal with the problem under investigation. The EM algorithm is often applied when missing data are present. The algorithm consists in two steps; the expectation step where the complete likelihood is replaced by its conditional expectation given the observed data. The second step of the EM algorithm consists in maximizing the expectation computed in the first step. This procedure is iterated until a specific accuracy is reached. The EM algorithm, based on likelihood methods, is applied to the different parametric models proposed to estimate the overall survival. For the study case, different kind of missing informations (i.e. missing time to complete remission and censored observations) need to be taken into account. As a consequence the EM algorithm is computationally rather demanding since all possible scenarios must be included in the likelihood function.

Structure of the thesis

In Chapter 1 a short introduction on survival analysis is provided. Classical methodology, including parametric and non parametric techniques, used to analyze time to event data are described.
Motivating example and study case are illustrated in details in Chapter 2. A general overview concerning the missing data problem and the specific problem associated to the data under investigation are described. The three different methodologies proposed to estimate missing time to complete remission are outlined at the end of the chapter.
In Chapters 3-5 all details related to the three proposed methods are described. Applications to the study cases are also concisely given.
In Chapter 6 the methodology proposed in Chapters 3-4 is extended to the competing risks model where relapse and death are the two causes of failure.
Finally, in Chapter 7, the performances of the proposed methodologies are evaluated trough a simulation study. The simulations method is described in details and the results are then discussed.
Conclusions and Appendix end this thesis. In the Appendix a selection of R code written for implementing the methodology proposed in this thesis is presented. A complete R code overview can be found at "http://tesi.cab.unipd.it".

Chapter 1

Introduction to Survival Analysis

Survival analysis is a combination of statistical techniques for analyzing time to event data. It was originally developed for studying time of onset of treatment until death but survival data arises in several fields such as medicine, biology, epidemiology, economics, engineering and demography. Survival analysis focus on time interval between an initiating event (start of treatment, diagnosis of a disease) and the occurrence of an event of interest, called *event* or *failure*, even though it is not necessary a failure, it may be a success as the recovery from a disease.

1.1 General concepts

1.1.1 Definition of Survival function

Let X be the random variable representing the length of the interval from the reference point to the occurrence of the event. The *survival time distribution* at a generic time point x is defined as:

$$S(x) = P(X \geq x),$$

it represent the probability of a random individual of the population surviving at least until time x. The variable X may be either continuous or discrete. If X is continuous the survival function is the complement of the cumulative distribution $F(x) = 1 - S(x)$ and by simple mathematical steps the density is given as:

$$f(x) = -\frac{dS(x)}{dx}.$$

The quantity $f(x)dx$ may be viewed as the approximate probability that the event happens at time x. The survival function is monotone, non-increasing, equal to one at time zero and converging to zero as time approaches infinity.

1.1.2 Hazard function

The *hazard function* or *hazard rate* $h(x)$ is the instantaneous rate that a randomly-selected individual known to be event-free at time x will fail in the next instant of time

$$h(x) = \lim_{\Delta x \to 0} \frac{P(x \leq X < x + \Delta x | X \geq x)}{\Delta x}.$$

If X is a continuous random variable the hazard is defined as follows:

$$h(x) = \frac{f(x)}{S(x)}.$$

The cumulative hazard, also known as the integrated hazard, at time x is given by:

$$H(x) = \int_0^x h(u)du = -ln(S(x)).$$

The hazard function and the cumulative hazard are not a probability but they represent a measure of the risk.

1.2 Censoring and truncation

When studying lifetimes of a population, one often has incomplete data. The kind of incompleteness is divided into two wide schemes called *censoring* and *truncation*. In both of them it is still possible to estimate the survival function and others quantity of interest, by carefully incorporating the missing informations.
The survival time of an individual is said to be censored when the end-point of interest has not been observed. This may occur because some individuals are lost to follow-up during the study or they have not experienced the event of interest by the end of the study. The only information known is that an individual was still alive up to a certain moment.
Let denote by X the lifetime of an individual, X's are assumed to be iid with density function $f(x)$ and survival function $S(x)$.
Different censoring schemes are listed below:

- *Right censoring*:
  - *type I*: a subject is type I right censored if at the end of the study he has not experienced the event.
  - *type II*: this kind of censoring happens when the study continues until the failure of the first r individuals, where r is a specified integer.

For right-censored data the exact lifetime X will be known only if it is lower than the censoring time C_r. Data can be represented by a pair of random variable (T, δ) where $T = min(X, C_r)$ and δ represent whether the event has occurred (=1) or not(=0), in the latter case the individual is censored.

- *Left Censoring*: a subject is left censored if it is known that the event of interest has already occurred at some time before entering the study. For example, in a study where at high school boys were asked: "when did you first use marijuana?" the answer: "I have used it but cannot recall when the first time was" is left censored observation.
  Data from left censored scheme can be represented as before by a pair of random variable (T, δ) where $T = max(X, C_l)$ with C_l the left censored time and δ is the indicator of the event.

- *Interval censoring*: a subject is interval censored if it is known that the event of interest occurs between two times, but the exact time of failure is not known. The only information is that time to failure lies within a certain interval.

Another kind of missing information, sometimes confused with censoring, is *truncation*. Here the individual is observed if the event time lies in a specific interval (T_l, T_r). The people whose event time is outside this interval are not observed.
Also here it is possible to discern between left or right truncated:

- *Left Truncated*: T_r is infinitive and we observe only patients whose event time is larger then T_l. An example are patients into a retirement home, the only observed are those who are older enough to enter in the retirement home.

- *Right Truncated*: T_l is zero and we observe only the patients whose event time is lower then T_r. An example is the estimation of the distribution of the stars, the stars too far away are not visible and then right truncated.

1.2.1 Likelihood construction

For the construction of the likelihood all the informations, including censored or truncated patients, have to be taken into account. Every pattern of complete or reduced informations contribute to the likelihood in a different way. More specifically:

exact lifetimes	-	$f(t)$
right-censored observations	-	$S(C_r)$
left-censored observations	-	$1 - S(C_l)$
interval-censored observations	-	$[S(C_L) - S(C_R)]$
left-truncated observations	-	$f(t)/S(T_L)$
right-truncated observations	-	$f(t)/[1 - S(T_R)]$
interval-truncated observations	-	$f(t)/[S(T_L) - S(T_R)]$

The complete data likelihood is:

$$L \propto \prod_{i \in D} f(t_i) \prod_{R} S(C_r) \prod_{L} (1 - S(C_l)) \prod_{I} [S(C_L) - S(C_R)] \qquad (1.1)$$

where D is the set of event, R the set of right-censored observations, L the set of left-censored observations, and I the set of interval censored observations. In case of left-truncated data with interval (T_{Li}, T_{Ri}), the term $f(t)$ in equation (1.1) is replaced by $f(t_i)/[S(T_{Li}) - S(T_{Ri})]$ and the term $S(C_i)$ by $S(C_i)/[S(T_{Li}) - S(T_{Ri})]$. In presence of only censored observations the likelihood can be written as follow:

$$L = \prod_{i=1}^{n} [h(t_i)]^{\delta_i} S(t_i)$$

where δ_i is the indicator of the event.

1.3 Survival function estimate

Parametric and non-parametric methods are available to estimate the survival function in the presence of censored data. A crucial assumption when estimating the survival function concerns the independence between the censoring mechanism and the event process. In this case the censoring is non informative. In case this assumption is violated and therefore the censoring is informative, appropriate methodology to estimate the survival function must be used. In the sequel of this thesis the censoring mechanism will be non informative.

1.3.1 Non-parametric methods

In this section a short overview of the most important fully non-parametric tools for the analysis of survival data is given. The non-parametric techniques

are well-known in the field of survival analysis due to their simplicity. Non-parametric methods are used when no theoretical distribution adequately fits the data. Suppose that the events occurs at distinct time $t_1 < t_2 < ... < t_D$ and at each time there are d_i events. Let Y_i be the number of individuals at risk at time t_i. There are two non-parametric methods to estimate the survival:

- **Product-Limit estimator** proposed by Kaplan and Meier (1958):

$$\hat{S}(t) = \begin{cases} 1 & \text{if } t < t_1 \\ \prod_{t_i \leq t} \left(1 - \frac{d_i}{Y_i}\right), & \text{if } t_1 \leq t. \end{cases} \tag{1.2}$$

  The variance of the Product-Limit estimator is estimated by Greenwood's formula:

$$\hat{V}[\hat{S}(t)] = \hat{S}(t)^2 \sum_{t_i \leq t} \frac{t_i}{Y_i(Y_i - d_i)}.$$

- **Nelson-Aalen estimator** of the cumulative hazard:

$$\hat{H}(t) = \begin{cases} 0 & \text{if } t < t_1 \\ \sum_{t_i \leq t} \frac{d_i}{Y_i}, & \text{if } t_1 \leq t, \end{cases}$$

  with variance:

$$\hat{\sigma^2_H} = \sum_{t_i \leq t} \frac{d_i}{Y_i^2}.$$

  Hence, the survival function is given by the relation $\hat{S}(t) = exp(-\hat{H}(t))$.

1.3.2 Parametric methods

When the distribution of time to event follows a certain pattern it is possible to estimate the survival function by employing a parametric distribution. It is well known that several distributions can be used to model the time event distribution. The most commons are listed in Table 1.1.

distribution	$f(t)$	$h(t)$	$S(t)$
Exponential $\lambda > 0, t \geq 0$	$\lambda exp(-\lambda t)$	λ	$exp(-\lambda t)$
Weibull $\alpha, \lambda > 0, t \geq 0$	$\alpha\lambda t^{\alpha-1} exp(-\lambda t^{\alpha})$	$\alpha\lambda t^{\alpha-1}$	$exp(-\lambda t^{\alpha})$
Gompertz $\theta, \alpha > 0, t \geq 0$	$\theta e^{\alpha t} exp(\frac{\theta}{\alpha}(1 - e^{\alpha t}))$	$\theta e^{\alpha t}$	$exp(\frac{\theta}{\alpha}(1 - e^{\alpha t}))$

Table 1.1: Parametric survival distribution.

1.4 Competing Risks

Competing risks concern the situation where more than one cause of failure is possible. The occurrence of a type of failure may precludes the occurrence of the others. For example, if failures are different causes of death, only the first of these to occur is observed. In other situations, events after the first failure may be observable, but not of interest. In cancer, death due to cancer is the event of interest, and death due to other causes (surgical mortality, old age) are competing events. Alternatively, one could be interested in time to relapse, where death due to any cause is a competing event. We can represent a competing risks model graphically with an initial state (alive or more generally event-free) and a number of different endpoints, as shown in Fig. 1.1.
Let $X_i, i = 1, 2, ...K$ be several times of occurrence of one of K competing events and n is the number of patients. For each patient the observed time to event is given by $T = min(X_1, ..., X_k)$ and an indicator specifying the cause of failure, i.e. $\delta = k$. Define:

- d_{kj}: number of patients failing from cause k at t_j
- $d_j = \sum_{k=1}^{K} d_{kj}$: total number of failures (from any cause) at t_j
- n_j: number of patients at risk at t_j

The *cause specific hazard* is defined as follows:

$$\lambda_k(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t, \delta = k | T \leq t)}{\Delta t}$$

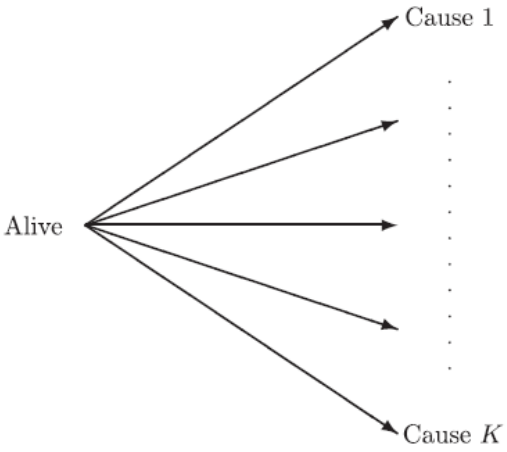

Figure 1.1: A competing risks situation with K causes of failure.

It represents the instantaneous rate of dying from cause k. The cumulative cause specific hazard is defined as: $\Lambda_k(t) = \int_0^t \lambda_k(s)ds$. In presence of competing events the overall survival represents the probability of being event free at least until time t, $S(t) = exp(-\sum_{i=1}^{K} \Lambda_i(t))$ or in other word is the probability of not having failed from any cause at time t. In competing risk analysis it may be of interest the probability of experiencing a specified event k, then the *cumulative incidence function* defined by

$$C_k(t) = P(T \leq t, \delta = k) = \int_0^t \lambda_k(s)S(s-)ds$$

must be considered. This is the probability of failing from cause k prior to time t. The cause specific hazard rate is estimated by:

$$\hat{\lambda}_k(t_j) = \frac{d_{kj}}{n_j}.$$

The probability of failing from cause k at t_j is estimated as:

$$\hat{p}_k(t_j) = \hat{\lambda}_k(t_j)\hat{S}(t_{j-1})$$

where $\hat{S}(t_{j-1})$ is the overall survival estimated by Kaplan-Meier estimator written in equation (1.2). Finally the cumulative incidence function is:

$$\hat{C}_k(t) = \sum_{j:t_j \leq t} \hat{p}_k(t_j).$$

Chapter 2

Motivating Example

When collecting data, there is often the possibility that they are incomplete or in other words that informations are missing. Of course many factors may be responsible for the presence of missing data, either given by chance either planned from the study design. This scenario may lead to statistical analysis with lack of power and to biased results. The aim of a statistician is to carry out efficient study adjusting for the lack of informations. In this chapter we describe the study case and the missingness problem associated. This thesis focus on the delicate aspect of incomplete data for a specific clinical data set and propose three methods for dealing with this problem.

2.1 Data description

The data set were collected by the Dutch Children Oncology Group (DCOG) and they come from a worldwide study [14] were children suffering from acute myeloid leukemia (AML) are followed since diagnosis of the disease. Leukemia is a type of cancer of the blood or bone marrow due to an abnormal increase of immature white blood cells called "blasts". AML is a type of leukemia that affects the myeloid line of blood cells and is characterized by a rapid progress of the disease. After treatment has been given patients may achieve a phase called *complete remission*. A patient is considered to be in complete remission if the disease has disappeared (using criteria developed by the International Working Group). However, it may happen that although a patient is in complete remission he might experience at later stage relapse, which means the return of the disease.
The data set used in this thesis, comes from a retrospective study from 19 childhood study groups worldwide. The aim of the study was to identify prognostic factor for clinical outcome as event free survival, overall survival

and cumulative incidence of relapse and employing them to stratify the patients in risk classes and eventually in risk-based therapy. In this study 837 children suffering from AML were included. A patient for which inexplicably time to complete remission (CR) and time to event are equal and a patient who died but for which the time of death was not recorded, were excluded. From the resulting 836 patients, 66 have not achieved complete remission. Table 2.1 shows the number of patients that experienced complete remission (CR=1) and those who did not (CR=0).

Event	$CR = 1$	$CR = 0$
	n	n
Relapse	94	3
Death	67	32
Relapse and death	124	5
Censored	485	26
Total	770	66

Table 2.1: Events distribution among patients.

The achievement of complete remission (CR), 5 years overall survival (OS), 5 years event-free survival (EFS) and cumulative incidence of relapse (CI) were analyzed. Univariate and multivariate analysis, such as Cox proportional hazard model, were performed to detect the prognostic factors.
Often in the field of survival analysis, the primary endpoint is to estimate the statistics of interest starting from a specified time origin, usually time from diagnosis or time from treatment. Since the disease presents several stages, the interest may move to estimate the survival function and related statistics, employing as starting point one of these intermediate phases.
Complete remission is an important stage that a patient may achieve as response to treatment. A non immunity to relapse, even if the symptoms are gone, makes very interesting the evaluation of the survival, or other relevant statistics, from the time of achievement of complete remission. Since in the study all the analysis were performed employing as origin time the time from diagnosis, it may be relevant to repeat the analysis rescaling the time in order to have as starting point the time to complete remission. Therefore, in this thesis, only patients that have achieved complete remission will be considered.
The baseline characteristics of patients from the reduced data set (i.e. including only patients with time to complete remission known) are shown in Table 2.2. Age and white blood cell count (WBC) are well known prognostic

factors. Later it has been found that a lower dose of anthacyclines in induction (the first two phases of chemoterapy) has a beneficial impact on disease.

Variable		n
Gender		
	Male	453
	Female	317
Age		
	Less than 6 years	253
	6 to less than 9 years	164
	9 to less than 13 years	210
	More than 13 years	143
WBC		
	Less than 10 x10^9/L	282
	10 to less than 20 x10^9/L	202
	20 to less than 50 x10^9/L	196
	50 or more x10^9/L	84
	Undefined	6
Anthracyclines dose in induction		
	Less than 300 mg/m^2	295
	300 to less than 360 mg/m^2	93
	360 to less than 420 mg/m^2	174
	420 or moremg/m^2	174
	Undefined	34

Table 2.2: Baseline characteristics of patients.

In the clinical data set a crucial problem to face is the incompleteness of the data. From a total of 770 patients who have reached complete remission for only 486 patients the exact time is known, for the remaining 284 patients the only information known is that complete remission has occurred sometime in the interval between time from diagnosis (time origin of the study) and time to event of interest either death or relapse.
In this situation the estimation of the survival distribution taking as starting point time to complete remission, can therefore be performed by using only individuals in the data set for whom time to complete remission is known.

However, due to the significant percentage of patients for whom time to complete remission is not known (around 40%), the exclusion from the analysis of these patients can not be considered as a wise solution, indeed the risk that it will leads to distorted estimates is high.
Missing values can cause serious problems if not properly handled. The statistical analysis with a large amount of missing data may results misleading and might introduce bias due to the lack of informations. Several techniques to avoid inconsistent analysis and unreliable results caused by missing data have been developed. The aim of this thesis is to investigate different statistical methods for our specific situation and produce reliable analysis. A simulation will be carried out to study the performances of the different proposed techniques.

2.2 Missing data

The term missing data means that there is an incomplete information on the phenomena in which we are interested. Missing data from surveys, experiments and observational studies are typically inevitable. This lack of information can be due to several causes including a non response of the subject, an impossibility to record some kind of variables or even a specifically choice of the study design.
Literature concerning the statistical approach for dealing with missing data goes back to the early 1970s [4, 12, 15]. Many procedures were not designed to handle incomplete data therefore a lot of research has been done in this field. The simplest technique replaces the missing values with sample mean, but this approach could lead to biased or unreliable answers. Denote by Z the variable with missing values. Several missing data mechanism can be summarized into the following categories:

- **MCAR**: Z is said to be *missing completely at random* if the probability of being missing does not depend on Z itself or on any other variable in the data set. Under this assumption we can carry on valid analysis excluding the subjects with missing values. The data set can be seen as a simple subsample from the original population.

- **MAR**: Z is said to be *missing at random* if the probability of being missing given the other variables in the data set does not depend on Z.

Inference can be done without any reference to the missing mechanism.

- **MNAR**: Z is said to be *missing not at random* if the probability of being missing, even accounting for all the other variables, still depends on Z. The analysis requires the explicit formulation of the missing data mechanism.

In our study we assume time to complete remission to be missing at random. This assumption has been made since the missing mechanism is not related to the outcomes of interest. This has been confirmed from the institute (DCOG) where the data have been collected. The MAR assumption is mathematically convenient because it allows to not formulate a model for the nonresponse mechanism. It is not possible to test whether MAR assumption holds or not.

2.3 Methodology overview

Several techniques have been employed to handle the problem of missing data. The most commons include removing individuals with missing informations and single imputation, but they generally lead to biased estimates. Deletion consists in removing all the subjects that present missing values with resulting reduced sample size together with the risk of not considering an important underlying pattern of missingness. For example in survey, there is frequently a difference between respondents and nonrespondents. If the analysis are conducted excluding the nonrespondents, the data may not be a representative sample of the larger population as a consequence the risk is to limit the external validity of the analysis.
Single imputation consists in replacing the missing data with a seemingly suitable value, often the mean of the observed subject, but relying to a single point is not a wise choice since the variability due to the unknown value is not taken into account. More sophisticated techniques have been developed in the last couple of decade especially based on the likelihood function. We now briefly describe the methodology used to solve the specific problem addressed in this thesis.

- Multiple Imputation: multiple imputation is a technique developed in 1970s [16] and consists in replacing each missing datum with several acceptable values, then the complete data set obtained from the imputed missing values can be analyzed.

- Parametric approach: the parametric approach is based on likelihood method [4]. A specific distribution, according to the observed values, is assumed for the data. The contribution to the likelihood from subject with missing value will be incorporated by integrating out all the possible values that the missing datum can assume. Then with maximum likelihood estimate of the complete likelihood, it is possible to compute the statistics of interest.

- Expectation-maximizazion (EM) algorithm: the EM-algorithm will be also based on likelihood methods (to distinguish from EM for non parametric models)[3, 10, 12]. The EM algorithm is an iterative procedure consisting in two step; the *expectation* step where the log likelihood for the complete data set is replaced by its conditional expectation given the observed values. The *maximization* step where the parameters are estimated according to maximum likelihood method. The procedure is carried on until convergence according to specific criteria. From the iterated maximum likelihood estimates the statistic of interest are estimated.

2.4 Notation

In this section we introduce the notation and the basic construction that will be used in the three methodologies described in Section 2.3.
Let T be the random variable representing the time between time origin (time from diagnosis) to time to complete remission, T is defined on $t \geq 0$. Let U be the random variable from time to complete remission to time to event, U is defined on $u \geq 0$. The time interval $[t_0, t_{event}]$ is divided into two intervals $[t_0, t_{cr}]$ and $[t_{cr}, t_{event}]$, where the splitting point is time of complete remission, see Figure 2.1.

Figure 2.1: Time interval.

According to this construction, in the first interval the event of interest is achievement of complete remission while in the second interval the event of interest is death or relapse. We will start by considering as unique event death and relapse, therefore the primary endpoint will be the overall survival.

In Figure 2.1, $\lambda_1(t)$ and $\lambda_2(u)$ denote respectively the hazard in the first and in the second interval. The two random variables are assumed to be independent. Further, we define V as the sum of T and U. This implies that V is defined on the interval $[t_0, t_{event}]$.
The mathematical structure written above and the notation introduced will be used in all methodologies discussed in the next chapters.

Chapter 3

Multiple Imputation

In this chapter the first methodology called *Multiple imputation* used to deal with the problem of missing time to complete remission will be described. First the theory underlying the algorithm will be introduced and then the application to the clinical data set described in details in Chapter 2 will be presented.

3.1 Introduction

Multiple imputation has become a very attractive approaches for handling the missing data. The technique was originally proposed by Rubin (1978) [16] and after analyzed in deep in Rubin (1987) [17]. The idea consists in creating plausible imputations for every missing value. While single imputation relies on a single value, which is usually the sample mean, multiple imputation try to reflect the uncertainty about the underlying value by replacing it with a set of seemingly closed values.
The procedure can be described as follow:

- Impute missing values using an appropriate model.
- Repeat this procedures m times and reconstruct m complete data set.
- Perform analysis on each data set using standard complete data methods.
- Average the values across the m samples to produce a single point estimate.

Rubin showed that even with a small number of imputed values, between 2 and 10, it is possible to achieve substantial improvements.

Let Y be the matrix of complete data and denote by Y_{obs} the observed part of Y and by Y_{mis} the missing part. Therefore the complete data set can be represented as $Y = (Y_{obs}, Y_{mis})$. The MAR (missing at random) assumption is taken in this study. This implies that the probability that an observation is missing may depend on Y_{obs}, but not on Y_{mis}. Under this assumption the missing mechanism is ignorable and the analysis can be performed without any further reference to it.
By following the algorithm described above and repeating the process m times, m complete data sets are produced. Therefore m sets of estimates and their associated variance will be produced. Rubin developed some rules in order to combine estimates and standard errors, based on each individual imputed data set, in an overall estimate with an associated variance. As a result an overall Multiple Imputation (MI) estimate and associated standard error will be computed. Let Q be the quantity of interest in the analysis. Assume that with complete data inference on Q would be based on normal approximation:

$$(Q - \hat{Q}) \sim N(0, U)$$

where $\hat{Q}$ is the statistic estimating Q and U is the associated variance.
By applying multiple imputation, m complete data set are constructed and for each data set, m statistic $Q_1^*, .., Q_m^*$ are computed. The multiple imputation overall point estimate is the average of the m estimates of Q^* from the m imputed data sets given by:

$$\bar{Q} = \tfrac{1}{m} \textstyle\sum_{i=1}^{m} Q_i^*.$$

The complete data variance $U_i^*, i = 1, .., m$ are combined in a similar way

$$\bar{U} = \frac{1}{m} \sum_{i=1}^{m} U_i^*. \tag{3.1}$$

Equation (3.1) represents the average within imputation variance,while the so called between imputation variance is given by:

$$B = \sum_{i=1}^{m} \frac{1}{m-1} (\hat{Q} - \bar{Q})^2. \tag{3.2}$$

Equation (3.1)-(3.2) are combined to obtain the total variance of the estimate of interest.

$$T = \bar{U} + \left(1 + \frac{1}{m}\right) B. \tag{3.3}$$

The factor $1/m$ in the total variance reflects the extra variability as a consequence of imputing the missing data using a finite number of imputations

instead of an infinite number. In situations where the between imputation variance B dominates $\bar{U}$ more accurate estimates can be obtained by increasing the number of imputations m. On the contrary when $\bar{U}$ dominates B, little can be gained by increasing m. The estimated confidence interval is

$$\bar{Q} \pm t_\nu(\alpha/2)\sqrt{T} \tag{3.4}$$

where t_v is the quantile of the student distribution with ν degree of freedom given by:

$$\nu = (m-1)\left(1+\frac{1}{r}\right)^2 \tag{3.5}$$

where r is the relative increase in variance due to missigness:

$$r = \left(1+\frac{1}{m}\right)B/\bar{U}. \tag{3.6}$$

For the proof of equation (3.3) see Rubin (1987, Section 3).
Multiple imputation is largely used due to the possibility to apply complete data method to the analysis and obtain the standard errors of the statistics. On the other hand, more time is required to run the analysis of the m repeated data sets. Finally multiple imputation does not produce the same results every time, since the simulated values are subject to random variability, this might be a problem when reproducibility results are necessary.

3.2 Application to the study case

In our study the primary goal is to estimate the survival function from time to complete remission. Therefore this will be the time origin for estimating the statistics of interest. As described in Chapter 2 in the data set used in this thesis there are about 40% individuals for whom time to complete remission is unknown (i.e. 40% missing time origin for estimating the survival function). We shall therefore applied multiple imputation method to reconstruct the missing values. The idea is to estimate time to complete remission for those patients for which this information is missing by using patients for which this information is known. Recall from Section 3.1 that data are missing at random i.e. the missigness mechanism is ignorable. Recall T the random variable representing the time to complete remission from diagnosis. Let $\tau = (\tau_1, \tau_2, ..., \tau_n)$ be the ordered vector $(\tau_1 < \tau_2 < .... < \tau_n)$ of known time to complete remission. Probability $P(T < \tau)$ will be estimated by using the empirical cumulative distribution. According to $P(T < \tau)$, we draw times from the set τ in order to impute the missing values. The imputation

procedure is carried out five times, according to Rubin's range of repeated imputation to obtain a relevant result. The algorithm is described in the following steps:

1. Compute the empirical cumulative distribution $P(T < t)$.
2. For each value of τ_i estimate $p_i = P(T < \tau_i) - P(T < \tau_{i-1})$.
3. Impute every missing value by choosing only $\tau < t_{event}$ and from the resulting set draw $m = 5$ times according to the probability (p) computed in step 2.
4. Reconstruct five complete data set and for each data sets estimate the statistic of interest. The primary endpoint is the overall survival (OS), i.e. the probability to be event (relapse,death) free from time to complete remission. The statistic is estimated by employing Kaplan-Meier's methodology. Let

$$OS_1^*(t), OS_2^*(t), ..., OS_5^*(t)$$

be the five estimates. Let

$$U_1^*(t), U_2^*(t), ..., U_5^*(t)$$

be the five estimated variance of the overall survival computed by using Greenwood variance estimator.

The final overall survival is the average over the five estimates at each time point :

$$\bar{OS}(t) = \frac{1}{5}\sum_{i=1}^{5} OS_i^*(t).$$

The total variance is given by:

$$T(t) = \bar{U}(t) + (1 + \frac{1}{5})B(t) \tag{3.7}$$

where

$$\bar{U}(t) = \textstyle\sum_{i=1}^{5} \frac{1}{5} U_i(t)$$

represents the within-imputation variance and

$$B(t) = \textstyle\sum_{i=1}^{5} \frac{1}{5-1}(OS^*(t) - \bar{OS}(t))^2$$

the between-imputation variance. Kaplan-Meier's methodology provides the survival (and variance) estimate in correspondence of the time points in which an event has occurred. Since five complete data set are reconstructed by imputing time to complete remission, five different set of time points are available.
In order to obtain the averaged estimate at each time points, for every data set the survival function is computed accounting for all set of values that time can assume. Figure 3.1 shows the estimated OS and relative confidence intervals when only patients with time to complete remission known are included in the analysis (set Y_{obs} in the terminology introduced before) and OS estimated on the complete data set (i.e. Y_{obs}, Y_{mis}). The 95% confidence interval for OS based on the multiple imputed data set are computed by employing equation (3.4). The confidence interval for OS based on the incomplete data set (i.e. Y_{obs} without missing values) is estimated by applying the asymptotic normality of the product limit estimator ($\hat{S}(t) \pm z_{1-\alpha/2}\hat{V}[\hat{S}(t)]$).
In the Appendix A Sections 1.1 and 1.2, R-code for the imputation technique introduced in this chapter and estimation of the OS and associated confidence interval for the complete (Y_{obs}, Y_{est}) and the observed data set (Y_{obs}) is given.

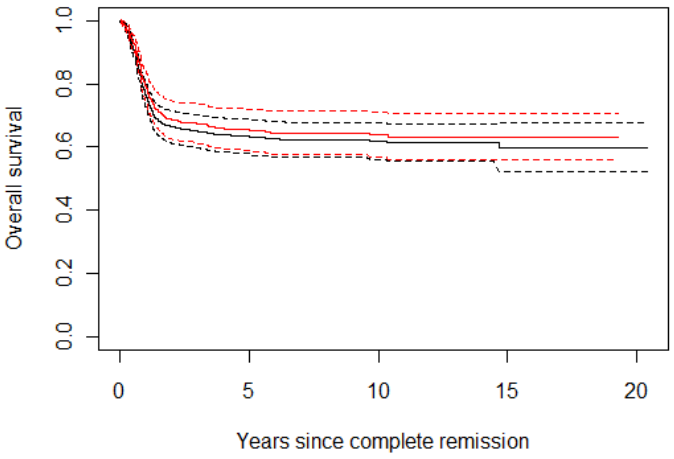

Figure 3.1: Overall survival estimate. Black line: imputed data set (Y_{obs}, Y_{est}), red line: reduced data set (Y_{obs})

Chapter 4

Parametric approach

In this chapter the missing time to complete remission from diagnosis will be estimated by employing a parametric approach. In this approach a parametric model will be used to describe the failure time of a patient. The methodology used will be described in Section 4.1. Application to the study case will be discussed in Sections 4.2-4.4.

4.1 Methodology

The starting point in this approach consists in choosing the more appropriate parametric model for the data under study. Therefore the cumulative hazard based on the complete data set is modeled by employing a known parametric distribution. To get the idea about the appropriate distribution to use, we shall look at the non-parametric estimated cumulative hazard based on patients with time to complete remission known.
Recall from Chapter 2 several times of interested are defined as follows: t_0: diagnosis time; t_{cr}: time to complete remission and t_{event}: time to event.
Figure 4.1 shows how the time interval is divided in two intervals.

Figure 4.1: Time interval.

The random variable T, defined in time interval $[t_0, t_{cr}]$, represents time to complete remission. The random variable U describes time to event of

interest from complete remission. The hazard rate associated to complete remission is indicated as $\lambda_1(t)$, while the hazard rate of event, either relapse or death, is indicated as $\lambda_2(u)$.
In Figure 4.2 a plot of the estimated cumulative hazards in both intervals is shown. By looking at the non-parametric estimation of the cumulative hazards as illustrated in Fig. 4.2 the exponential and the Weibull seem the more appropriate distributions for our data set. Due to their simplicity and general goodness of fit, these distributions are the most commonly used in survival analysis.
Once the distribution has been chosen, from the complete data likelihood the maximum likelihood estimate of the parameters are obtained allowing us to estimate the statistics of interest.
Three combinations for the variables T and U on the two intervals are illustrated in Figure 4.2. More specifically, the following combinations have been considered: exponential distribution on both intervals $[t_0, t_{cr}]$ and $[t_{cr}, t_{event}]$ (indicated here as exponential-exponential); exponential distribution on the interval $[t_0, t_{cr}]$ and Weibull on $[t_{cr}, t_{event}]$ (indicates as as exponential-Weibull). In the last combination the random variable Weibull will be considered on both intervals (indicated as Weibull-Weibull in Figure 4.2). The solid line in Figure 4.2, represents the non-parametric cumulative hazard estimated from the data set where only patients with time to complete remission known have been included. The dashed curve represents the parametric cumulative hazard. Origin time for the second interval is zero because time has been rescaled. Particular attention is given to the first interval where all the patients achieved the event of interest (complete remission). Values for the parameters have been chosen based on graphical inspection of the shape of the non-parametric cumulative hazards.
As it can be seen from Figure 4.2 fitting either the exponential or the Weibull on the first interval gives similar results (see Figure 4.2 panel on the left side). Neither the exponential nor the Weibull follows perfectly the behavior of the data, in particular the exponential distribution does not fit the slight curvature of data while the Weibull distribution shows a strong deviation at the extremity of the interval.
On the interval $[t_{cr}, t_{event}]$ the choice is easier than in the interval $[t_0, t_{cr}]$. The cumulative hazards based on the exponential distribution is completely different than the non-parametric one. As it can be seen from the right side of Figure 4.2 the cumulative hazards based on the Weibull distribution, even with randomly chosen parameters, seems to fit the data in a proper way. The combination of exponential random variables on both intervals (represented in Fig. 4.2 as exponential-exponential on the left upper corner) is a simplification of the real situation and does not provide a good fitting of the data

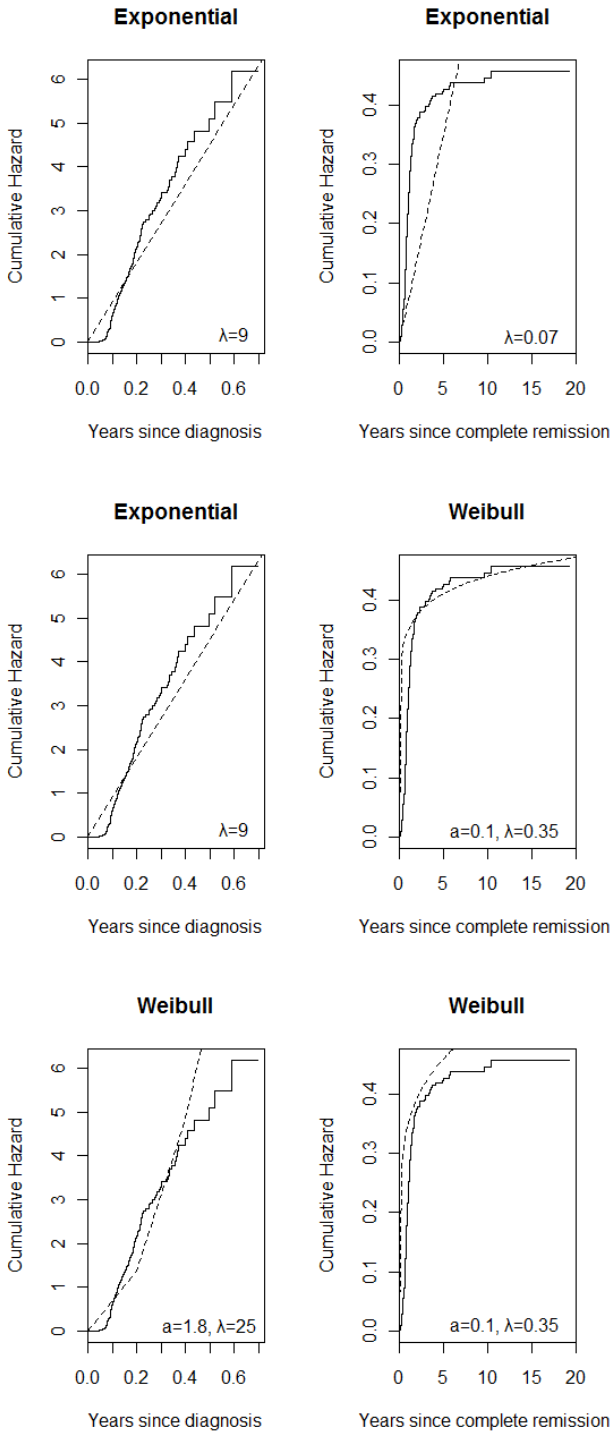

Figure 4.2: Cumulative hazard. Solid line: non-parametric, dashed line: parametric.

as the Weibull distribution does. However it is the only situation for which an explicit formula for the likelihood can be written.
The next step is the likelihood computation. Here for patients with time to complete remission missing, the only information available is that the unknown time falls in the interval $[t_0, t_{event}]$ where t_0 is diagnosis time. For patients with time to complete remission known the likelihood is given as:

$$\lambda_1(t)S_1(t)\lambda_2^{\delta}(u)S_2(u)$$

where $\lambda_1(t)$, $\lambda_2(u)$, $S_1(t)$, $S_2(u)$ are the hazard rate and survival respectively in $[t_0, t_{cr}]$ and $[t_{cr}, t_{event}]$ and δ is the event indicator, i.e. $\delta = 1$ if the event is observed $\delta = 0$ otherwise.
For patients with time to complete remission unknown we integrate out all possible range of values from zero to time to event in the interval $(0, v)$:

$$\int_0^v \lambda_1(t)S_1(t)\lambda_2^{\delta}(u)S_2(u)\,dt.$$

The likelihood for the complete data set is then given as:

$$\begin{aligned} L = &\prod_{i=1}^{n}(\lambda_1(t_i)S_1(t_i)\lambda_2^{\delta_i}(u_i)S_2(u_i))^{r_i}\cdot \\ &(\int_0^v \lambda_1(t_i)S_1(t_i)\lambda_2^{\delta_i}(u_i)S_2(u_i)\,dt)^{1-r_i} \end{aligned} \tag{4.1}$$

where r_i is an indicator for time to complete remission i.e $r_i = 1$ if for individual i time to complete remission is known and 0 otherwise.
Note the two different ways to indicate the missing information in (4.1). The indicator r_i refers to an event that did occur but it is unknown when, δ indicator is the well known indicator in survival analysis for censored observations (i.e. $\delta = 1$ if the event occurs $\delta = 0$ otherwise).
Once the maximum likelihood estimate is obtained, it is possible to estimate the overall survival function and relative confidence interval.
Let $\boldsymbol{\theta}$ be a vector of parameters and $\hat{\boldsymbol{\theta}}$ its associated maximum likelihood estimate. The pointwise 95% confidence interval at a generic time t, for the parametric overall survival estimate is:

$$\hat{S}(t,\hat{\theta}) \pm 1.96\sqrt{\hat{var}(\hat{S}(t,\hat{\theta}))} \tag{4.2}$$

where $\widehat{var}(\hat{S}(t,\hat{\theta}))$ is the estimated variance of the survival. To obtain the variance $\widehat{var}(\hat{S}(t,\hat{\theta}))$ the multivariate delta method is employed in the following way:

$$\widehat{var}(\hat{S}(t,\hat{\theta})) = \frac{\partial S(t,\theta)}{\partial\theta}\bigg|_{\theta=\hat{\theta}} \widehat{cov}(\hat{\theta}) \frac{\partial S(t,\theta)}{\partial\theta}^T\bigg|_{\theta=\hat{\theta}}$$

where $\widehat{cov}(\hat{\theta})$ is the covariance matrix of the maximum likelihood estimators, i.e. the inverse of the Fisher information matrix.
In Sections 4.2-4.4 the methodology described in this section will be applied to the three different combinations of parametric distributions used to model time to events in the two intervals illustrated in Fig. 4.1.

4.2 Exponential distribution on both intervals

Let T and U be random variables exponentially distributed with parameters λ_1 and λ_2 respectively. The likelihood will be computed by distinguishing two situations depending on the equality or inequality of the parameters λ_1, λ_2 in the two intervals.
If $\lambda = \lambda_1 = \lambda_2$ the likelihood either with time to complete remission missing or known, is:

$$\lambda^{\delta+1} e^{-\lambda v}.$$

Hence the likelihood for the complete data set is:

$$L = \prod_{i=1}^{n} \lambda^{\delta_i+1} e^{-\lambda v_i},$$

with log-likelihood:

$$\ell = (d+n)\log\lambda - \lambda \sum_{i=1}^{n} v_i$$

where d is the number of events.
If $\lambda_1 \neq \lambda_2$ the likelihood for a patient with time to complete remission known, has the following form:

$$\lambda_1 \lambda_2^{\delta} e^{-\lambda_1 t - \lambda_2 (v-t)}. \tag{4.3}$$

In case time to complete remission is missing the likelihood is:

$$\int_0^v \lambda_1 \lambda_2^{\delta} e^{(\lambda_2-\lambda_1)t-\lambda_2 v} dt = \frac{\lambda_1 \lambda_2^{\delta} e^{-\lambda_2 v}}{\lambda_2 - \lambda_1} (e^{(\lambda_2-\lambda_1)v} - 1). \tag{4.4}$$

Combining the equations (4.3)-(4.4), the likelihood for the complete data set is:

$$L = \prod_{i=1}^{n} \lambda_1 \lambda_2^{\delta_i} e^{-\lambda_2 v_i} (e^{(\lambda_2-\lambda_1)t_i})^{r_i} \left(\frac{e^{(\lambda_2-\lambda_1)v_i} - 1}{\lambda_2 - \lambda_1} \right)^{(1-r_i)}$$

and the log likelihood:

$$\ell = n \log \lambda_1 + d \log \lambda_2 - \lambda_2 \sum_{i=1}^{n} v_i + (\lambda_2 - \lambda_1) \sum_{i \in CR} t_i + \sum_{i \in CRM} \log \frac{e^{(\lambda_2 - \lambda_1) v_i} - 1}{(\lambda_2 - \lambda_1)}$$

where d is the number of events, CR is the set of patients with time to complete remission known, CRM is the set of patients with time to complete remission missing.
Although the equation of the log likelihood is rather simple, it is not possible to find a closed form for the maximum likelihood estimators, therefore maximization is done numerically by employing the R function `nlminb` from the library `optimx`. Maximum likelihood estimates and associated variance are reported in table 4.1.

	λ_1	λ_2
mle	7.73653341	0.06774362
std	0.347048098	0.003985097

Table 4.1: MLE and std for exponential-exponential combination.

From the maximum likelihood estimates it is possible to compute the statistic of interest. The overall survival from time to complete remission is equal to $e^{-\hat{\lambda}_2 u}$. The pointwise 95% confidence interval of the survival are computed according to equation (4.2), where the variance is given by:

$$\begin{aligned} \widehat{var}(\hat{S}(\hat{\lambda}_2, u)) &= \frac{\partial S(u, \lambda_2)}{\partial \lambda_2}^2 \Big|_{\lambda_2 = \hat{\lambda}_2} var(\hat{\lambda}_2) \\ &= u^2 e^{-2\hat{\lambda}_2 u} var(\hat{\lambda}_2) \end{aligned}$$

where $var(\hat{\lambda}_2)$ is the inverse of the Fisher information.
The estimated overall survival function and related pointwise confidence interval are plotted in Figure 4.3. Overall survival is shown for the first 5 or 6 years after complete remission since this is the time interval interesting for clinicians.

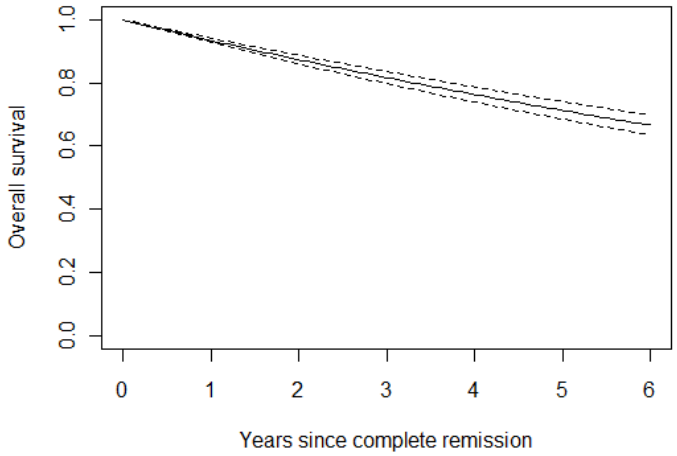

Figure 4.3: Overall survival function for the exponential-exponential combination.

4.3 Exponential distribution on $[t_0, t_{cr}]$ and Weibull distribution on $[t_{cr}, t_{event}]$

In this section the combination exponential-Weibull will be considered.
Let T be an exponential distribution with parameter λ_1 and U a Weibull distribution with parameters α_2 and λ_2 of shape and scale respectively.
Also here as in Section 4.2, the likelihood for the complete data set will be investigated.
The likelihood for a patient with time complete remission known and unknown is given by:

$$\lambda_1 e^{-\lambda_1 t}(\lambda_2 \alpha_2 u^{\alpha_2 - 1})^{\delta} e^{-\lambda_2 u^{\alpha_2}}$$

and

$$\int_0^v \lambda_1 e^{-\lambda_1 t}(\lambda_2 \alpha_2 u^{\alpha_2 - 1})^{\delta} e^{-\lambda_2 u^{\alpha_2}}\, dt$$

respectively. In this situation the integral can not be written down in closed form.

The likelihood for the complete data set is defined in the following way:

$$L = \prod_{i=1}^{n} (\lambda_1 e^{-\lambda_1 t_i} (\lambda_2 \alpha_2 u_i^{\alpha_2 - 1})^{\delta_i} e^{-\lambda_2 u_i^{\alpha_2}})^{r_i}.$$
$$(\int_0^{v_i} \lambda_1 e^{-\lambda_1 t_i} (\lambda_2 \alpha_2 u_i^{\alpha_2 - 1})^{\delta_i} e^{-\lambda_2 u_i^{\alpha_2}} \, dt)^{1-r_i}.$$

The maximum likelihood estimator cannot be found in closed form as in Setcion 4.2 . The only possible way to handle the problem of the intractable integral is to resort to a numerical method. In particular the R function `integrate` provides, with a specified accuracy, the value of the integral for a generic function, employing an adaptive quadrature method. Maximum likelihood estimates for the parameters λ_1 and λ_2, numerically computed with `nlminb`, are shown in Table 4.2.

	λ_1	α_2	λ_2
mle	7.7084675	0.5135147	0.1590852
std	0.34966499	0.03564428	0.01589787

Table 4.2: MLE and std for exponential-Weibull combination.

The estimated overall survival function from time to complete remission, based on the Weibull distribution, is given by:

$$\hat{S}(u, \hat{\alpha}_2, \hat{\lambda_2}) = e^{-\hat{\lambda}_2 u^{\hat{\alpha}_2}}.$$

As in Section 4.2, employing the multivariate delta method, the estimated variance is computed as follows:

$$\widehat{var}(\hat{S}(u, \hat{\alpha}_2, \hat{\lambda}_2)) = ds \; cov(\hat{\alpha}_2, \hat{\lambda}_2) ds^T$$

where ds is the vector of the first derivative of the survival function with respect to the two parameters defined as:

$$ds = \left(\frac{\partial S(u, \alpha_2, \lambda_2)}{\partial \alpha_2}, \frac{\partial S(u, \alpha_2, \lambda_2)}{\partial \lambda_2} \right) \Big|_{\alpha_2 = \hat{\alpha}_2, \lambda_2 = \hat{\lambda}_2}$$
$$= (-\hat{\lambda}_2 u^{\hat{\alpha}_2} \log(u) e^{-\hat{\lambda}_2 u^{\hat{\alpha}_2}}, -u^{\hat{\alpha}_2} e^{-\hat{\lambda}_2 u^{\hat{\alpha}_2}}),$$

and $cov(\hat{\alpha}_2, \hat{\lambda}_2)$ is the inverse of the Fisher information matrix. As in Section 4.2, the 95% pointwise interval confidence is plotted in Figure 4.4. The

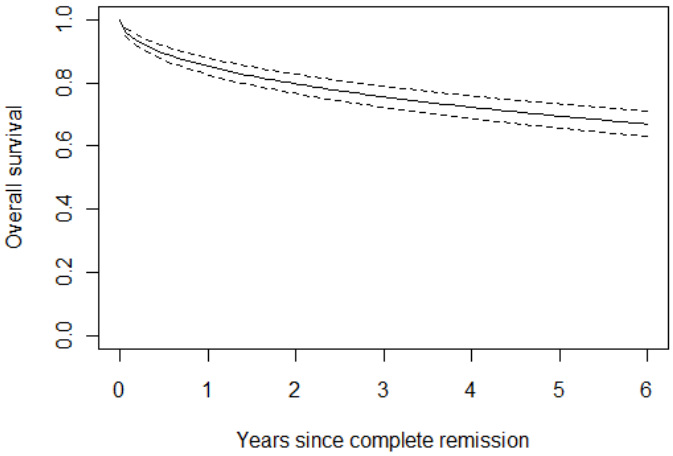

Figure 4.4: Overall survival function for the exponential-Weibull combination.

estimate of the confidence interval is based on (4.2). Also in Figure 4.4 the follow up is restricted to the first 6 years. Although the follow up is much longer the events of interest usually occurs in the first 5-6 years.

4.4 Weibull distribution on both intervals

In this section a Weibull model will be considered. This implies a Weibull distribution on $[t_0, t_{cr}]$ and $[t_{cr}, t_{event}]$ for the two random variables T and U. In particular, define $T \sim Weibull(\alpha_1, \lambda_1)$ and $U \sim Weibull(\alpha_2, \lambda_2)$, where α_1 and α_2 are shape parameters while the scale parameters are λ_1 and λ_2. This is indicated as the Weibull-Weibull model. Again as in the previous Sections 4.2 and 4.3 the likelihood for a patient with time complete remission known is:

$$\lambda_1 \alpha_1 t^{\alpha_1 - 1} e^{-\lambda_1 t^{\alpha_1}} (\lambda_2 \alpha_2 u^{\alpha_2 - 1})^{\delta} e^{-\lambda_2 u^{\alpha_2}}$$

and for time to complete remission missing:

$$\int_0^v \lambda_1 \alpha_1 t^{\alpha_1 - 1} e^{-\lambda_1 t^{\alpha_1}} (\lambda_2 \alpha_2 u^{\alpha_2 - 1})^{\delta} e^{-\lambda_2 u^{\alpha_2}} \, dt.$$

Also here the complete data likelihood has not a closed form, since it has the following form:

$$L = \prod_{i=1}^{n} (\lambda_1 \alpha_1 t_i^{\alpha_1 - 1} e^{-\lambda_1 t_i^{\alpha_1}} (\lambda_2 \alpha_2 u_i^{\alpha_2 - 1})^{\delta_i} e^{-\lambda_2 u_i^{\alpha_2}})^{r_i} \cdot$$
$$(\int_0^{v_i} \lambda_1 \alpha_1 t_i^{\alpha_1 - 1} e^{-\lambda_1 t_i^{\alpha_1}} (\lambda_2 \alpha_2 u_i^{\alpha_2 - 1})^{\delta_i} e^{-\lambda_2 u_i^{\alpha_2}} \, dt)^{1 - r_i}.$$

Again, the Weibull-Weibull model like the exponential-Weibull, does not provide an analytic form for the maximum likelihood estimators. The maximization of the likelihood is done numerically as before and again by using R functions `integrate` and `nlminb`. Maximum likelihood estimate for the parameters of the model are reported in Table 4.3.

	α_1	λ_1	α_2	λ_2
mle	1.8809534	36.0231927	0.5058668	0.1763163
std	0.05627384	3.68554941	0.02806724	0.01397453

Table 4.3: MLE and std for Weibull-Weibull combination.

The overall survival function estimate with its relative confidence interval are not reported since they are similar to the one computed with the exponential-Weibull model. Indeed the random variable U is Weibull distributed in both models. Computations of variance and confidence intervals are similar to the previous combination and therefore are not shown, further the maximum likelihood estimates are very close. The overall survival function plot is not shown since results very similar to Figure 4.4.
In Appendix A Section 2.1 R-code written to estimate the exponential-exponential model is provided. The code for the remaining models can be found online at "http://tesi.cab.unipd.it/".

Chapter 5

Expectation-maximization algorithm

In this chapter the last methodology proposed to deal with the missing data problem will be discussed. This chapter is organized as follows. In Section 5.1 a general introduction to the EM algorithm is given. In Section 5.2 details concerning the EM methodology applied to the study case analyzed in this thesis is illustrated.

5.1 Introduction

The expectation-maximization (EM) algorithm is a broadly applicable iterative process designed for the computation of maximum likelihood estimate rather useful when there are incomplete data. The name was given by Dempster, Laird and Rubin (1977)[3] even if the underlying idea was sketched by Orchard and Woodbury (1972)[12]. The algorithm consists in two steps, the Expectation and Maximization, carried out repeatedly until a convergence criterion is met. The EM algorithm is applied to a wide range of statistical fields, due to its formulation that reduces the complexity of the estimation problem. As mentioned before, the major application of the algorithm is when the maximum likelihood estimator has to be computed in the presence of incomplete data. The basic idea is to reduce the *incomplete* data problem to a *complete* data problem that is often more tractable, creating a link between the likelihood under the two conditions.
The iterative process consists of a series of steps in which the missing data is replaced by its conditional expectation given the observed data and the parameters are repeatedly update until convergence criteria are met.
The EM algorithm is not only useful for incomplete data problem but also

in situations where the incompleteness of information is not evident. There are situations where at first sight the problem under study may not appear as the classical incomplete data problem but by formulating it as such, it reduces the complexity of the problem.
The EM algorithm was firstly criticized because it does not produces an estimate of the covariance matrix of the maximum likelihood estimators, but later researches have been carried on to solve this problem [9, 11].
In the next section details about the algorithm are outlined. In Section 5.2 the EM algorithm will be applied to our data by using the models described in the previous chapters.

5.1.1 Algorithm formulation

Let Y be a random vector corresponding to the complete data with join density $f_c(y, \theta)$ and θ a p-dimensional parameters, $\theta \in \Theta \in R^p$. If the complete data vector y is observed, the maximum likelihood estimate of the parameters based on the data is found by maximizing the log likelihood function.

$$\log L_c(\theta, y) = \ell_c(\theta, y) = \log f_c(\theta, y).$$

In the presence of missing data y is not observed. The vector y can be written as (y_{obs}, y_{mis}). The observed data y_{obs}, with p.d.f. $f(y_{obs}, \theta)$, is seen as a function of the complete data $y_{obs} = y_{obs}(y)$. The EM algorithm is useful when the function $\ell(\theta, y_{obs})$ is difficult to be maximized compared to $\ell_c(\theta, y)$. In many situations the EM algorithm is particularly suitable, even if the problem does not at first appears as incomplete data problem.
The computation of the MLE is made by an iterative process in which the complete data likelihood is replaced by its conditional expectation given the observed data and the current estimate of the parameters.
More specifically, let $\theta^{(0)}$ be an initial value for θ. The first step, called expectation, involves the calculation of:

$$Q(\theta, \theta^{(0)}) = E_{\theta^{(0)}}(\log L_c(\theta)|y_{obs}).$$

The second step, the maximization, requires to maximize the function Q in θ and to find the value $\theta^{(1)}$ such that:

$$Q(\theta^{(1)}, \theta^{(0)}) \geq Q(\theta, \theta^{(0)}).$$

Then the E-step and M-step are carried out again in an iterative process where the estimate $\theta^{(0)}$ is replaced by the current fit $\theta^{(1)}$.
At a generic step k the procedure is described as follows:

E-step: Compute $Q(\theta, \theta^{(k)})$, where

$$Q(\theta, \theta^{(k)}) = E_{\theta^{(k)}}(\log L_c(\theta)|y_{obs})$$

M-step: Find $\theta^{(k+1)}$ that maximizes $Q(\theta, \theta^{(k)})$, that is:

$$Q(\theta^{(k+1)}, \theta^{(k)}) \geq Q(\theta, \theta^{(k)})$$

for all $\theta \in \Theta$
The procedure is carried out until the difference between the likelihood at step k and $k+1$:

$$L(\theta^{(k+1)}) - L(\theta^{(k)})$$

decreases by a very small quantity, $\epsilon > 0$.

The generalized EM algorithm (GEM)

The generalized EM algorithm is a simplification of the EM algorithm in which the maximization step requires only that $\theta^{(k+1)}$ is chosen such that the inequality

$$Q(\theta^{(k+1)}, \theta^{(k)}) \geq Q(\theta^{(k)}, \theta^{(k)})$$

is satisfied. This implies that it is not necessary the maximization of $Q(\theta, \theta^{(k)})$ for all $\theta \in \Theta$. This condition is sufficient to satisfy

$$L(\theta^{(k+1)}) \geq L(\theta^{(k)}).$$

The GEM algorithm produces a sequence of likelihood values that converge if bounded above.

5.1.2 Convergence of the algorithm

In this section the convergence of the likelihood values to a stationary value will be illustrated. First the monotonicity property of the likelihood will be evaluated and then the issue about convergence will be discussed. Dempster, Laird and Rubin (1977) shown that at each iteration the function $L(\theta)$ is not decreasing. To prove this property consider the complete data distribution as:

$$f_c(y, \theta) = f(y_{obs}, \theta) f_1(y_{mis}|y_{obs}, \theta)$$

where f_1 is the conditional distribution of the missing data given the observed. The log likelihood function is given by:

$$\ell(\theta, y_{obs}) = \ell_c(\theta, y) - \log f_1(y_{mis}|y_{obs}, \theta). \tag{5.1}$$

By taking the expectation of both side of equation (5.1) over the conditional distribution of y given y_{obs} at the current fit $\theta^{(k)}$ for θ, leads to:

$$\begin{aligned}\ell(\theta, y_{obs}) &= \int \ell_c(\theta, y) f(y|y_{obs}, \theta^{(k)}) dy - \int \log f_1(y_{mis}|y_{obs}, \theta) f(y|y_{obs}, \theta^{(k)}) dy \\ &= E_{\theta^{(k)}}(\ell_c(\theta, y)|y_{obs}) - E_{\theta^{(k)}}(\log f_1(y_{mis}|y_{obs}, \theta)|y_{obs}) \\ &= Q(\theta, \theta^{(k)}) - H(\theta, \theta^{(k)})\end{aligned} \tag{5.2}$$

where:

$$H(\theta, \theta^{(k)}) = E_{\theta^{(k)}}(\log f_1(y_{mis}|y_{obs}, \theta)|y_{obs}).$$

It follows from (5.2):

$$\begin{aligned}\ell(\theta^{(k+1)}) - \ell(\theta^{(k)}) =& Q(\theta^{(k+1)}, \theta^{(k)}) - Q(\theta^{(k)}, \theta^{(k)}) - \\ & (H(\theta^{(k+1)}, \theta^{(k)}) - H(\theta^{(k)}, \theta^{(k)})).\end{aligned} \tag{5.3}$$

The first difference, on the right side of the equation (5.3), is nonnegative since the parameter $\theta^{(k+1)}$ is chosen such that the following inequality holds for all $\theta \in \Theta$:

$$Q(\theta^{(k+1)}, \theta^{(k)}) \geq Q(\theta, \theta^{(k)}).$$

By employing Jensen's inequality for the second difference we have:

$$H(\theta^{(k+1)}, \theta^{(k)}) - H(\theta^{(k)}, \theta^{(k)}) \leq 0.$$

The above inequality yields to (5.3) greater than 0 for every $k \geq 0$. Therefore after an iteration of the EM algorithm the likelihood function is not decreasing. For a bounded sequence of likelihood values, $L(\theta^{(k)})$ converges in a monotone way to some value L^*. It is important to know under which conditions L^* is a stationary value and whether it is a local or global maximum. In almost all application L^* is a stationary value. Wu (1983) shows the convergence of any EM sequence to a stationary point (not necessarily a maximum) of the likelihood function when the complete data come from an exponential family with compact parameter space, and when the Q function satisfies a certain mild differentiability condition. If L has multiple stationary points, convergence of the EM sequence to either type (local or global maximizers, saddle points) depends upon the starting value $\theta^{(0)}$ for θ.
If $L(\theta)$ is unimodal, then any sequence $\theta^{(k)}$ will converge to the unique MLE of $L(\theta)$, irrespective of its starting value.

5.1.3 Covariance matrix estimation

As already mentioned in Section 5.1.1, the EM algorithm does not provide the covariance matrix for the MLE. Therefore alternative estimation techniques, within the EM framework, have been developed.
The estimated covariance matrix for MLE is given by the inverse of the observed information matrix $I(\theta_{mle}, y_{obs})$ computed after the evaluation of the MLE. The observed information matrix implies the computation of the second derivatives of $\ell(\theta, y_{obs})$. This is in most situations intractable and complicate to evaluate.
The solution consists on expressing the observed information matrix in terms of complete likelihood. Louis (1982) provides the relation between the complete and observed information matrix.

$$\begin{aligned} I(\theta, y_{obs}) &= \mathcal{I}_c(\theta, y_{obs}) - cov(S_c(y,\theta)|y_{obs}) \\ &= \mathcal{I}_c(\theta, y_{obs}) - E(S_c(y,\theta)S_c^T(y,\theta)|y_{obs}) + S(y_{obs},\theta)S^T(y_{obs},\theta) \end{aligned} \tag{5.4}$$

where $\mathcal{I}_c(\theta, y_{obs}) = E(I_c(\theta, y)|y_{obs})$ is the expected complete information matrix given the observed values, $S_c(y,\theta) = \partial \log L_c(\theta)/\partial\theta$ is the score function of the complete data and $S(y_{obs},\theta) = E_\theta(S_c(y,\theta)|y_{obs})$ is the expected score funtion given the observed data. By employing (5.4) the observed information matrix is estimated by:

$$I(\theta_{mle}, y_{obs}) = \mathcal{I}_c(\theta_{mle}, y_{obs}) - E(S_c(y,\theta)S_c^T(y,\theta)|y_{obs})_{\theta=\theta_{mle}}$$

since the last term of (5.4) is zero because it satisfies the maximum likelihood estimate property $S(y_{obs}, \theta_{mle}) = 0$.
In case of regular exponential family with θ as parameter, the information matrix is computed as follows:

$$I(\theta_{mle}, y_{obs}) = [cov_\theta(t(y)) - cov_\theta(t(y)/y_{obs})]_{\theta=\theta_{mle}} \tag{5.5}$$

where $t(y)$ is the complete sufficient statistic.

5.2 Application of the EM algorithm to the case study

In this section the EM algorithm is applied to the case study to deal with the problem of missing time to complete remission.
Recall T represents the random variable introduced in Section 2.3 to model

the time to complete remission from diagnosis, while U is defined as the random variable used to model the event of interest from complete remission. The two random variables are independent and $V = U + T$.
Since the EM algorithm is likelihood based method, it is necessary to specified a parametric distribution for the random variables T and U in order to evaluate the likelihood. The choice of the distributions T and U has been described in details in Chapter 4. We shall therefore use also here the three models proposed and discussed in Chapter 4.

5.2.1 Exponential distribution on both intervals

Let T and U be exponentially distributed with parameters λ_1 and λ_2 on the interval $[t_0, t_{cr}]$ and $[t_{cr}, t_{event}]$ respectively.
The likelihood for the complete data set assuming all time to complete remission known and no censored observation is given as follows:

$$L_c = \lambda_1^n \lambda_2^n e^{(-\lambda_1 \sum_{i=1}^n t_i - \lambda_2 \sum_{i=1}^n u_i)},$$

and the log likelihood:

$$\ell_c = n \log \lambda_1 + n \log \lambda_2 - \lambda_1 \sum_{i=1}^{n} t_i - \lambda_2 \sum_{i=1}^{n} u_i.$$

The data present two kind of missing informations: the first is related to the censoring mechanism and the second one concerns the absence for some patients of time to complete remission.
In the expectation step the conditional expectation of the log likelihood given the data is computed. Since the log likelihood is linear with respect to the complete data, the only computation needed is the expectation of the complete data given the observed. In order to simplify the computation, the conditional average is estimated by dividing the population in patients with time to complete remission known and missing.
Recall t_i,u_i and v_i are defined as: $t_i \geq 0$ time from t_0 (diagnosis) to complete remission, $u_i \geq 0$ time from complete remission to event and $v_i = t_i + u_i$ time from the origin t_0 to the event.

E-step
In the E-step we shall look at two situations depending on the information about time to complete remission.

1. Time to complete remission observed: here only the censored observations have to be taken into account. If a patient experienced the event,

there is no need to compute the conditional expectation, it is suffice to replace the observed values. On the contrary, if the event has not occurred, the only information known is $U \geq v - t$. The conditional expectation for a generic step k is given as follows:

$$E_{\lambda_2^{(k)}}(U_i|U_i > v_i - t_i) = v_i - t_i + \frac{1}{\lambda_2^{(k)}},$$

given that the conditional distribution is equal to:

$$\lambda_2 e^{\lambda_2(u_i-(v_i-t_i))} I_{(v_i-t_i,\text{inf})}(u_i).$$

2. Time to complete remission missing: the conditional expectations of the two random variable given the data are needed.

$$E_{\lambda_1^{(k)},\lambda_2^{(k)}}(T_i|V_i, \Delta_i = \delta_i),$$

$$E_{\lambda_1^{(k)},\lambda_2^{(k)}}(U_i|V_i, \Delta_i = \delta_i).$$

We have now to distinguish between the situation in which the event of interest has occurred ($\delta = 1$) or not ($\delta = 0$).
Let $\delta = 1$, the conditional distributions $f(t|v, \delta = 1)$ and $f(u|v, \delta = 1)$ are computed employing Bayes's theorem:

$$f(t|v,\delta=1) = \frac{f(v|t)f(t)}{\int_0^v f(v|t)f(t)dt}, \quad f(u|v,\delta=1) = \frac{f(v|u)f(u)}{\int_0^v f(v|u)f(u)du}. \tag{5.6}$$

The conditional distribution $f(v|t)$ is given as follows:

$$\begin{aligned} f(v|t) =& P(V = v|T = t) = P(U + T = v|T = t) \\ =& P(U = v - t|T = t) = P(U = v - t) \\ =& \lambda_2 e^{-\lambda_2(v-t)}, \end{aligned}$$

$$\begin{aligned} \int_0^v f(v|t)f(t)dt =& \int_0^v \lambda_2 e^{-\lambda_2(v-t)} \lambda_1 e^{-\lambda_1 t} dt \\ =& \int_0^v \lambda_1\lambda_2 e^{-\lambda_2 v} e^{(\lambda_2-\lambda_1)t} dt \\ =& \frac{\lambda_1\lambda_2}{\lambda_2-\lambda_1} e^{-\lambda_2 v}(e^{(\lambda_2-\lambda_1)v} - 1), \end{aligned}$$

by combining the above equations the conditional distribution is given as:

$$f(t|v,\delta=1)=\frac{(\lambda_2-\lambda_1)e^{(\lambda_2-\lambda_1)t}}{e^{(\lambda_2-\lambda_1)v}-1}. \tag{5.7}$$

Finally, the conditional expectation at step k is equal to:

$$\begin{aligned}E_{\lambda_1^{(k)},\lambda_2^{(k)}}(T_i|V_i=v_i,\Delta=1)=\int_0^{v_i}\frac{t(\lambda_2^{(k)}-\lambda_1^{(k)})e^{(\lambda_2^{(k)}-\lambda_1^{(k)})t}}{e^{(\lambda_2^{(k)}-\lambda_1^{(k)})v}-1}dt=\\ \frac{1}{e^{(\lambda_2^{(k)}-\lambda_1^{(k)})v_i}-1}\Bigg[v_ie^{(\lambda_2^{(k)}-\lambda_1^{(k)})v_i}+\\ (\lambda_2^{(k)}-\lambda_1^{(k)})^{-1}(1-e^{(\lambda_2^{(k)}-\lambda_1^{(k)})v_i})\Bigg].\end{aligned} \tag{5.8}$$

Similarly for the conditional distribution $f(u|v)$ we obtain:

$$f(u|v)=\frac{(\lambda_1-\lambda_2)e^{(\lambda_1-\lambda_2)u}}{e^{(\lambda_1-\lambda_2)v}-1}, \tag{5.9}$$

$$\begin{aligned}E_{\lambda_1^{(k)},\lambda_2^{(k)}}(U_i|V_i=v_i,\Delta=1)=\int_0^{v_i}\frac{u(\lambda_1^{(k)}-\lambda_2^{(k)})e^{(\lambda_1^{(k)}-\lambda_2^{(k)})u}}{e^{(\lambda_1^{(k)}-\lambda_2^{(k)})v}-1}du=\\ \frac{1}{e^{(\lambda_1^{(k)}-\lambda_2^{(k)})v_i}-1}\Bigg[v_ie^{(\lambda_1^{(k)}-\lambda_2^{(k)})v_i}+\\ (\lambda_1^{(k)}-\lambda_2^{(k)})^{-1}(1-e^{(\lambda_1^{(k)}-\lambda_2^{(k)})v_i})\Bigg].\end{aligned} \tag{5.10}$$

We now consider the situation where $\delta=0$, i.e. the event has not occurred. Here we need to compute:

$$f(t|v,\delta=0)=\frac{S(v|t)f(t)}{\int_0^v S(v|t)f(t)dt}\qquad f(u|v,\delta=0)=\frac{S(v|u)f(u)}{\int_0^v S(v|u)f(u)du}. \tag{5.11}$$

The survival $S(v|t)$ is given as:

$$\begin{aligned}S(v|t,\delta=0)=&P(V>v|T=t)=P(U+T>v|T=t)\\ =&P(U>v-t|T=t)=P(U>v-t)\\ =&e^{-\lambda_2(v-t)},\end{aligned}$$

$$\begin{aligned}\int_0^v S(v|t)f(t)dt &= \int_0^v e^{-\lambda_2(v-t)}\lambda_1 e^{-\lambda_1 t}dt \\ &= \int_0^v \lambda_1 e^{-\lambda_2 v}e^{(\lambda_2-\lambda_1)t}dt \\ &= \frac{\lambda_1 e^{-\lambda_2 v}}{\lambda_2-\lambda_1}(e^{(\lambda_2-\lambda_1)v}-1),\end{aligned}$$

the equations above lead to

$$f(t|v,\delta=0) = \frac{(\lambda_2-\lambda_1)e^{(\lambda_2-\lambda_1)t}}{e^{(\lambda_2-\lambda_1)v}-1}.$$

The conditional distribution $f(t|v,\delta=0)$ is equal to $f(t|v,\delta=1)$ see (5.7), and this implies that having experienced the event does not affect the conditional distribution $f(t|v)$. The conditional expectation $E(T|V>v_i)$ is given in equation (5.8).
In a similar way, the conditional distribution $f(u|v,\delta=0)$ is equal to $f(u|v,\delta=1)$ shown in equation (5.9). This leads to the conditional expectation shown in (5.10).
If $\lambda_1=\lambda_2$, the log likelihood for the complete uncensored data set is:

$$\ell_c = 2n\log\lambda - 2\lambda\sum_{i=1}^{n} v_i,$$

the values v_i related to patients who have not experienced the event, have to be replaced by the conditional expectation:

$$E_{\lambda^{(k)}}(V|V>v_i) = v_i + \frac{1}{2\lambda^{(k)}}$$

where $V=U+T$ is an exponential distribution with parameter 2λ.

M-step
The aim of the maximization step is to find the most expected value of the parameters of the function analyzed. The function to be considered is the log likelihood where the unknowns times to event or missing time to complete remission are replaced with the expected values computed before. The equation to be maximized for a generic step k is then:

$$\begin{aligned}Q((\lambda_1,\lambda_2),(\lambda_1,\lambda_2)^{(k)}) =& n\log\lambda_1\lambda_2 - \lambda_1\Big(\sum_{i\in CR} t_i + \sum_{i\in CRM} t^*\Big) - \\ & \lambda_2\left(\sum_{i\in CR} u_i + \frac{d_1}{\lambda_2^{(k)}} + \sum_{i\in CRM} u_i^*\right)\end{aligned}$$

where

$$t_i^* = \frac{v_i e^{(\lambda_2^{(k)}-\lambda_1^{(k)})v_i} + (\lambda_2^{(k)}-\lambda_1^{(k)})^{-1}\left[1-e^{(\lambda_2^{(k)}-\lambda_1^{(k)})v_i}\right]}{e^{(\lambda_2^{(k)}-\lambda_1^{(k)})v_i}-1},$$

$$u_i^* = \frac{v_i e^{(\lambda_1^{(k)}-\lambda_2^{(k)})v_i} + (\lambda_1^{(k)}-\lambda_2^{(k)})^{-1}\left[1-e^{(\lambda_1^{(k)}-\lambda_2^{(k)})v_i}\right]}{e^{(\lambda_1^{(k)}-\lambda_2^{(k)})v_i}-1}$$

and where d_1 is the number of patients who have achieved complete remission but have not experienced the event, CRM is the set of patients with time to complete remission missing and CR is the set of patients with time to complete remission known.
By employing the current estimate of the parameters $(\lambda_1^{(k)}, \lambda_2^{(k)})$, and maximizing Q with respect to λ_1 and λ_2 it can be easily found that the maximum likelihood estimators are:

$$(\hat{\lambda}_1, \hat{\lambda}_2) = \left(\frac{n}{\sum_{i\in CR} t_i + \sum_{i\in CRM} t^*}, \frac{n}{\sum_{i\in CR} u_i + \frac{d_1}{\lambda_2^{(k)}} + \sum_{i\in CRM} u_i^*}\right).$$

The iterative procedure is carried on till the inequality

$$Q((\lambda_1,\lambda_2)^{(k+1)}, (\lambda_1,\lambda_2)^{(k)}) - Q((\lambda_1,\lambda_2)^{(k)}, (\lambda_1,\lambda_2)^{(k)}) \leq \epsilon$$

is satisfied, where ϵ is a sufficiently small amount.
If $\lambda_1 = \lambda_2$ the quantity Q to be maximized becomes:

$$Q(\lambda, \lambda^{(k)}) = 2n\log\lambda - \lambda\left(\sum_{i=1}^{n} v_i + \frac{d_0}{2\lambda^{(k)}}\right)$$

where d_0 is the total number of censored observations. The maximum likelihood estimator is given by:

$$\hat{\lambda} = \frac{2n}{\sum_{i=1}^{n} v_i + \frac{d_0}{2\lambda^{(k)}}}.$$

Variance estimation

By employing equation (5.5) the Fisher information of the maximum likelihood estimator is the difference between the unconditional variance on the sufficient statistic and the conditional variance on the sufficient statistic given the observed data. Each of these variances are computed for the value of the parameter equals to the maximum likelihood estimate. Define $\hat{\lambda}_1, \hat{\lambda}_2$ the

maximum likelihood estimates for the parameters of random variables T and U respectively.
For λ_1, the sufficient statistic is $\sum_i^n t_i$ and therefore the unconditional variance is $\frac{n}{\hat{\lambda}_1^2}$. The conditional variance is given by:

$$\sum_{CRM} var(T|V) = \sum_{CRM} var(T|V = v_i)^{\delta_i} + var(T|V > v_i)^{1-\delta_i}.$$

Since the conditional distribution $f(T|V = v_i)$ is equal to $f(T|V > v_i)$, also the variance is equal and is computed as follows:

$$var(T|V) = E(T^2|V) - (E(T|V))^2. \tag{5.12}$$

The second term on the right side of equation (5.12) is the expectation computed before in (5.8) to the second power. The first term is given by:

$$\begin{aligned} E(T^2|V) &= \int_0^v t^2 f(t|v)dt \\ &= \frac{e^{(\lambda_2-\lambda_1)v}}{e^{(\lambda_2-\lambda_1)v} - 1}\left[v^2 - \frac{2v}{\lambda_2 - \lambda_1} + \frac{2}{(\lambda_2 - \lambda_1)^2}\right]. \end{aligned}$$

Similarly for λ_2 the unconditional variance is $\frac{n}{\hat{\lambda}_2^2}$. The conditional variance is given by:

$$\sum_{CRM} var(U|V) + \sum_{CR} var(U|U > v - t)^{1-\delta_i}.$$

The variance related to the maximum likelihood estimate is given by the inverse of the Fisher information, i.e. the difference between the unconditional and conditional variance.
The application of the EM algorithm to our situations gives the results reported in Table 5.1 where MLE and its relative standard errors have been estimated.

	λ_1	λ_2
mle	7.6871334	0.1644853
std	0.352178577	0.003839006

Table 5.1: MLE and std for exponential-exponential combination.

The overall survival from time to complete remission and its associated pointwise confidence interval are plotted in Figure 5.1, the follow up is restricted

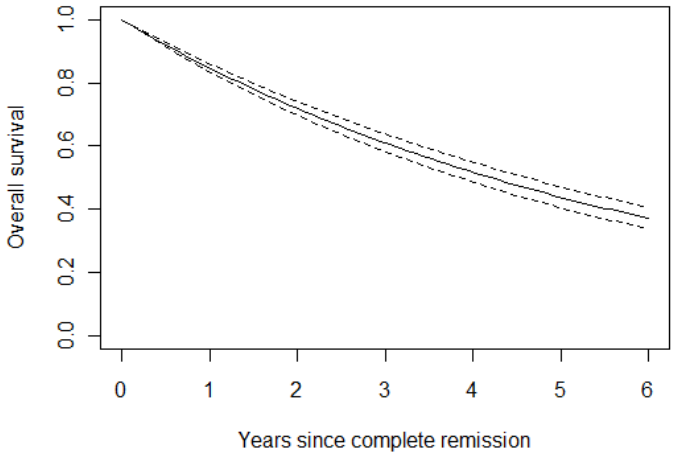

Figure 5.1: Overall survival function for the exponential-exponential combination.

as in Chapter 4 to 6 years. The variance is computed by employing the delta method described in Chapter 4.

5.2.2 Exponential distribution on $[t_0, t_{cr}]$ and Weibull distribution on $[t_{cr}, t_{event}]$

The second model is the same model described in Section 4.3, i.e. a mixture between exponential and Weibull distribution.
The random variable T related to the first interval (i.e. $[t_0, t_{cr}]$) is exponentially distributed with parameter λ_1. The random variable U defined on the intervals $[t_{cr}, t_{event}]$ follows a Weibull distribution with parameters α_2, λ_2 of shape and scale respectively.
The likelihood and the log likelihood for the complete uncensored data set are

$$L_c = (\lambda_1 \alpha_2 \lambda_2)^n e^{(-\lambda_1 \sum_{i=1}^n t_i - \lambda_2 \sum_{i=1}^n u_i^{\alpha_2})} \prod_{i=1}^n u_i^{\alpha_2 - 1}$$

and

$$\ell_c = n \log(\lambda_1 \alpha_2 \lambda_2) - \lambda_1 \sum_{i=1}^n t_i - \lambda_2 \sum_{i=1}^n u_i^{\alpha_2} + (\alpha_2 - 1) \sum_{i=1}^n \log u_i$$

respectively.
As illustred before, all the conditional expectations for the missing values are needed, but now it is not possible to find a closed form for the integrals. Further, the log likelihood is not a linear function of the complete data. To obtain the expected values, the conditional distribution of the complete data given the observed values are first computed and then the logarithmic and exponential expectation are estimated.
In the sequel we describe the E-step and the M-step in details. Here the computations are more demanding due to the different combinations of random variables considered.

E-step

Consider first the case where time to complete remission is known:

- if $\delta = 1$, the observed time to complete remission and time to event can be replaced.
- if $\delta = 0$, time to complete remission (t) is observed and the expected value of the missing u has to be computed. This yields to the following computations:

$$E_{\alpha_2^{(k)},\lambda_2^{(k)}}(\log U_i | U_i > v_i - t_i) = \int_{v_i - t_i}^{\infty} log(u)\alpha_2^{(k)}\lambda_2^{(k)} u^{\alpha_2^{(k)}-1} e^{-\lambda_2^{(k)}[u^{\alpha_2^{(k)}} - (v_i-t_i)^{\alpha_2^{(k)}}]} du,$$

$$E_{\alpha_2^{(k)},\lambda_2^{(k)}}(U_i^{\alpha_2} | U_i > v_i - t_i) = \int_{v_i - t_i}^{\infty} u^{\alpha_2}\alpha_2^{(k)}\lambda_2^{(k)} u^{\alpha_2^{(k)}-1} e^{-\lambda_2^{(k)}[u^{\alpha_2^{(k)}} - (v_i-t_i)^{\alpha_2^{(k)}}]} du.$$

Consider now the situation where time to complete remission is missing:

- If $\delta = 1$, the event has occurred but time to complete remission is missing, therefore we have to replace t_i and u_i respectively with their expected values.
  We now compute the conditional distributions by employing equation (5.6):

$$f(t|v, \delta = 1) = \frac{\lambda_1\alpha_2\lambda_2(v-t)^{\alpha_2-1}e^{-\lambda_1 t - \lambda_2(v-t)^{\alpha_2}}}{\int_0^v \lambda_1\alpha_2\lambda_2(v-t)^{\alpha_2-1}e^{-\lambda_1 t - \lambda_2(v-t)^{\alpha_2}} dt},$$

$$f(u|v,\delta=1)=\frac{\lambda_1\alpha_2\lambda_2u^{\alpha_2-1}e^{-\lambda_2u^{\alpha_2}-\lambda_1(v-u)}}{\int_0^v\lambda_1\alpha_2\lambda_2u^{\alpha_2-1}e^{-\lambda_2u^{\alpha_2}-\lambda_1(v-u)}du}.$$

The conditional expectations, using the current value of the parameters at step k, are computed as follows:

$$E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(T_i|V_i=v_i,\Delta_i=1)=\int_0^{v_i}tf(t|v,\delta=1)dt,$$

$$E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log U_i|V_i=v_i,\Delta_i=1)=\int_0^{v_i}\log(u)f(u|v,\delta=1)du,$$

$$E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(U_i^{\alpha_2}|V_i=v_i,\Delta_i=1)=\int_0^{v_i}u^{\alpha_2}f(u|v,\delta=1)du.$$

- If $\delta=0$, the event has not yet occurred, the conditional distributions computed by employing (5.11) are given as follows:

$$f(t|v,\delta=0)=\frac{\lambda_1e^{-\lambda_1t^-\lambda_2(v-t)^{\alpha_2}}}{\int_0^v\lambda_1e^{-\lambda_1t^-\lambda_2(v-t)^{\alpha_2}}dt},$$

$$f(u|v,\delta=0)=\frac{\alpha_2\lambda_2u^{\alpha_2-1}e^{-\lambda_2u^{\alpha_2}-\lambda_1(v-u)}}{\int_0^v\alpha_2\lambda_2u^{\alpha_2-1}e^{-\lambda_2u^{\alpha_2}-\lambda_1(v-u)}du}.$$

Thus, the conditional expectations are:

$$E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(T_i|V_i>v_i,\Delta_i=0)=\int_0^{v_i}tf(t|v,\delta=0)dt,$$

$$E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log U_i|V_i>v_i,\Delta_i=0)=\int_0^{v_i}\log(u)f(u|v,\delta=0)du,$$

$$E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(U_i^{\alpha_2}|V_i>v_i,\Delta_i=0)=\int_0^{v_i}u^{\alpha_2}f(u|v,\delta=0)du.$$

All the integrals are numerically computed by using the function `integrate` in R which employs an adaptive quadrature method.

M-step

In Section 5.1.1 the function Q has been introduced. This is the complete log likelihood in which the unknown values are replaced by their expected values. Define:

- $ul_1^* = E_{\alpha_2^{(k)},\lambda_2^{(k)}}(\log U_i | U_i > v_i - t_i)$,
- $ue_1^* = E_{\alpha_2^{(k)},\lambda_2^{(k)}}(U_i^{\alpha_2} | U_i > v_i - t_i)$,
- $t_1^* = E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(T_i | V_i = v_i, \Delta_i = 1)$,
- $ul_2^* = E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log U_i | V_i = v_i, \Delta_i = 1)$,
- $ue_2^* = E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(U_i^{\alpha_2} | V_i = v_i, \Delta_i = 1)$,
- $t_2^* = E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(T_i | V_i > v_i, \Delta_i = 0)$,
- $ul_3^* = E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log U_i | V_i > v_i, \Delta_i = 0)$,
- $ue_3^* = E_{(\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(U_i^{\alpha_2} | V_i > v_i, \Delta_i = 0)$.

The function Q to be maximized for the model considered in this section has a rather long form given by:

$$\begin{aligned} Q =& n\log(\lambda_1\alpha_2\lambda_2) - \lambda_1\Bigg[\sum_{i\in CR} t_i + \sum_{i\in CRME} (t_1^*)_i + \sum_{i\in CRMEM} (t_2^*)_i\Bigg] - \\ & \lambda_2\Bigg[\sum_{i\in CRE} u_i^{\alpha_2} + \sum_{i\in CREM} ue_{1i}^* + \sum_{i\in CRME} ue_{2i}^* + \sum_{i\in CRMEM} ue_{3i}^*\Bigg] + \\ & (\alpha_2 - 1)\Bigg[\sum_{i\in CRE} \log u_i + \sum_{i\in CREM} ul_{1i}^* + \sum_{i\in CRME} ul_{2i}^* + \sum_{i\in CRMEM} ul_{3i}^*\Bigg] \end{aligned}$$

where Q is the notation for $Q((\lambda_1, \alpha_2, \lambda_2), (\lambda_1, \alpha_2, \lambda_2)^{(k)})$ and the set in the sums are defined as:

- CR: set of patients with time to complete remission known;
- CRE: set of patients with known time to complete remission and event has occurred;
- CREM: set of patients with known time to complete remission and event has not occurred yet;
- CRME: set of patients with time to complete remission unknown and event has occurred;
- CRMEM: set of patients with time to complete remission unknown and event has not occurred yet.

Also in this case by employing the complete data log likelihood, does not exist a closed form for the maximum likelihood estimators. The maximization procedure is performed by using numerical techniques. Here as in Section 4.3 the R function `nlminb` is used.

Variance estimation

The variance associated to the maximum likelihood estimates in the exponential-Weibull model is computed according to equation (5.4).
Denote by y_{obs} the observed data and by $\hat{\lambda}_1, \hat{\alpha}_2, \hat{\lambda}_2$ the maximum likelihood estimates for the exponential and Weibull distribution respectively.
The observed Fisher information corresponding to the parameter λ_1 for the exponential distribution is given by:

$$\begin{aligned} I(\hat{\lambda}_1|y_{y_{obs}}) &= \left[E_{\lambda_1}\left(\frac{-\partial^2 \ell_c}{\partial \lambda_1^2}\middle| y_{obs} \right) - Var_{\lambda_1}\left(\frac{\partial \ell_c}{\partial \lambda_1}\middle| y_{obs} \right) \right]_{\lambda_1=\hat{\lambda}_1} \\ &= \frac{n}{\hat{\lambda}_1^2} - Var(T|y_{obs}). \end{aligned}$$

Similarly for the shape parameter α_2 for the Weibull distribution:

$$\begin{aligned} I(\hat{\alpha}_2|y_{obs}) &= \left[E_{\alpha_2}\left(\frac{-\partial^2 \ell_c}{\partial \alpha_2^2}\middle| y_{obs} \right) - Var_{\alpha_2}\left(\frac{\partial \ell_c}{\partial \alpha_2}\middle| y_{obs} \right) \right]_{\alpha_2=\hat{\alpha}_2} \\ &= \frac{n}{\hat{\alpha}_2^2} + \hat{\lambda}_2 E(U^{\hat{\alpha}_2}(\log U)^2|y_{obs}) - \\ &\quad \Big[\hat{\lambda}_2^2 Var(U^{\hat{\alpha}_2}\log U|y_{obs}) + Var(\log U|y_{obs}) + \\ &\quad \hat{\lambda}_2^2 cov(U^{\hat{\alpha}_1}\log U, \log U|y_{obs}) \Big]. \end{aligned}$$

The observed Fisher information for the scale parameter λ_2 is given as follows:

$$\begin{aligned} I(\hat{\lambda}_2|y_{obs}) &= \left[E_{\lambda_2}\left(\frac{-\partial^2 \ell_c}{\partial \lambda_2^2}\middle| y_{obs} \right) - Var_{\lambda_2}\left(\frac{\partial \ell_c}{\partial \lambda_2}\middle| y_{obs} \right) \right]_{\lambda_2=\hat{\lambda}_2} \\ &= \frac{n}{\hat{\lambda}_2^2} - Var(U^{\hat{\alpha}_2}|y_{obs}). \end{aligned}$$

In order to compute the variance of the statistic of interest, which is the overall survival, the delta method must be used. The covariance between the maximum likelihood estimators of parameters α_2 and λ_2 is required at this

stage. A bit of algebra yields to:

$$I(\hat{\alpha}_2, \hat{\lambda}_2|y_{obs}) = \left[E_{\alpha_2,\lambda_2}\left(\frac{-\partial^2 \ell_c}{\partial\alpha_2\partial\lambda_2}\middle|y_{obs}\right) - Var_{\alpha_2,\lambda_2}\left(\frac{\partial\ell_c}{\partial\alpha_2}\frac{\partial\ell_c}{\partial\lambda_2}\middle|y_{obs}\right)\right]_{\alpha_2=\hat{\alpha}_2,\lambda_2=\hat{\lambda}_2}$$
$$= E(U^{\hat{\alpha}_2}\log U|y_{obs}) -$$
$$[-cov(U^{\hat{\alpha}_2}, \log U|y_{obs}) + \hat{\lambda}_2 cov(U^{\hat{\alpha}_2}, U^{\hat{\alpha}_2}\log U|y_{obs})].$$

The variance of the maximum likelihood estimator is given by the inverse of the Fisher information. All the expected values are computed numerically. In Table 5.2 the MLE for the parameters of the exponential and the Weibull distribution and their corresponding standard errors are illustrated. Results for the MLE, as expected, are very similar to the MLE in Table 5.1 while standard error estimated in Table 5.2 are smaller for the Weibull distribution compared with values in Table 5.1.

	λ_1	α_2	λ_2
mle	7.6006340	0.7499809	0.1758731
std	0.346331694	0.032476097	0.008254766

Table 5.2: MLE and std for exponential-Weibull combination.

The overall survival and the corresponding pointwise confidence interval for the first 6 years after complete remission are plotted in Figure 5.2.

5.2.3 Weibull distribution on both intervals

In this Section the last combination of models is considered. On both intervals the random variables T and U follow a Weibull distribution with parameters α_1, λ_1 and α_2, λ_2 respectively.
The likelihood for the complete data set is given by

$$L_c = (\alpha_1\lambda_1\alpha_2\lambda_2)^n e^{(-\lambda_1\sum_{i=1}^n t_i^{\alpha_1} - \lambda_2\sum_{i=1}^n u_i^{\alpha_2})}\prod_{i=1}^n t_i^{\alpha_1-1}u_i^{\alpha_2-1}$$

and the log likelihood:

$$\ell_c = n\log(\alpha_1\lambda_1\alpha_2\lambda_2) - \lambda_1\sum_{i=1}^n t_i^{\alpha_1} - \lambda_2\sum_{i=1}^n u_i^{\alpha_2} +$$
$$(\alpha_1 - 1)\sum_{i=1}^n \log t_i + (\alpha_2 - 1)\sum_{i=1}^n \log u_i.$$

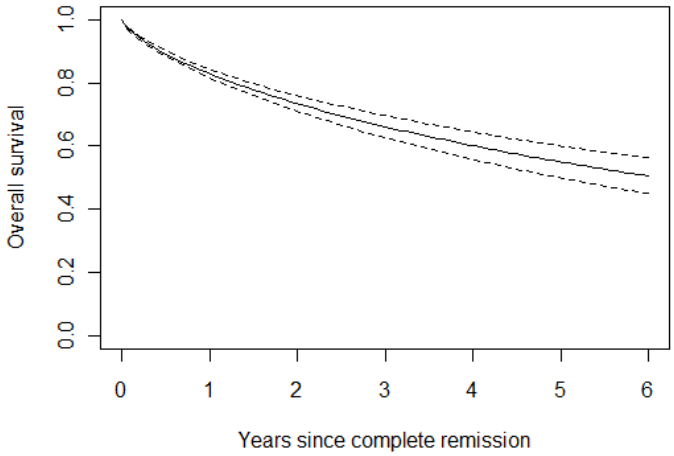

Figure 5.2: Overall survival function for the exponential-Weibull combination.

As seen before, in the exponential-Weibull model, the log likelihood is not a linear function of the complete data. Therefore, we shall firstly compute the conditional distribution of the complete data given the observed and later estimate the expected complete data logarithm and exponential.
In the sequel the E-step and the M-step for this model will be shortly described. This model is more demanding from the computational point of view, but the methodology is exactly as before has been described in Sections 5.2.1.-5.2.2. for the other two models.

E-step
Time to complete remission known:

- if $\delta = 1$ (i.e. event has occurred), t_i and u_i can be replaced with the observed values.

- if $\delta = 0$ (i.e. event has not yet occurred), time to complete remission can be replaced, instead the value of u_i has to be computed as follows:

$$E_{\alpha_2^{(k)},\lambda_2^{(k)}}(log(U_i)|U_i > v_i - t_i) =$$
$$\int_{v_i-t_i}^{\infty} log(u)\alpha_2^{(k)}\lambda_2^{(k)}u^{\alpha_2^{(k)}-1}e^{-\lambda_2^{(k)}[u^{\alpha_2^{(k)}}-(v_i-t_i)^{\alpha_2^{(k)}}]}du,$$

$$E_{\alpha_2^{(k)},\lambda_2^{(k)}}(U_i^{\alpha_2}|U_i > v_i - t_i) =$$
$$\int_{v_i-t_i}^{\infty} u^{\alpha_2}\alpha_2^{(k)}\lambda_2^{(k)}u^{\alpha_2^{(k)}-1}e^{-\lambda_2^{(k)}[u^{\alpha_2^{(k)}}-(v_i-t_i)^{\alpha_2^{(k)}}]}du.$$

Time to complete remission missing:

- If $\delta = 1$, first derive the conditional distributions and then the expected values. The general formula for the conditional distribution is written in equation (5.6) and the specific results for this model are:

$$f(t|v,\delta=1) = \frac{\alpha_1\lambda_1\alpha_2\lambda_2 t^{\alpha_1-1}(v-t)^{\alpha_2-1}e^{-\lambda_1 t^{\alpha_1}-\lambda_2(v-t)^{\alpha_2}}}{\int_0^v \alpha_1\lambda_1\alpha_2\lambda_2 t^{\alpha_1-1}(v-t)^{\alpha_2-1}e^{-\lambda_1 t^{\alpha_1}-\lambda_2(v-t)^{\alpha_2}}dt},$$

$$f(u|v,\delta=1) = \frac{\alpha_1\lambda_1\alpha_2\lambda_2 u^{\alpha_2-1}(v-u)^{\alpha_1-1}e^{-\lambda_2 u^{\alpha_2}-\lambda_1(v-u)^{\alpha_1}}}{\int_0^v \alpha_1\lambda_1\alpha_2\lambda_2 u^{\alpha_2-1}(v-u)^{\alpha_1-1}e^{-\lambda_2 u^{\alpha_2}-\lambda_1(v-u)^{\alpha_1}}du}.$$

  Then the conditional expectations, using the current value of the parameters, are computed as follows:

$$E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log(T_i)|V_i = v_i, \Delta_i = 1) = \int_0^{v_i} \log(t)f(t|v,\delta=1)dt,$$

$$E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(T_i^{\alpha_1}|V_i = v_i, \Delta_i = 1) = \int_0^{v_i} t^{\alpha_1}f(t|v,\delta=1)dt,$$

$$E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log(U_i)|V_i = v_i, \Delta_i = 1) = \int_0^{v_i} \log(u)f(u|v,\delta=1)du,$$

$$E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(U_i^{\alpha_2}|V_i = v_i, \Delta_i = 1) = \int_0^{v_i} u^{\alpha_2}f(u|v,\delta=1)du.$$

- If $\delta = 0$, employing the formulas in equation (5.11) yields to

$$f(t|v,\delta=0) = \frac{\alpha_1\lambda_1 t^{\alpha_1-1}e^{-\lambda_1 t^{\alpha_1}-\lambda_2(v-t)^{\alpha_2}}}{\int_0^v \alpha_1\lambda_1 t^{\alpha_1-1}e^{-\lambda_1 t^{\alpha}-\lambda_2(v-t)^{\alpha_2}}dt},$$

$$f(u|v,\delta=0) = \frac{\alpha_2\lambda_2 u^{\alpha_2-1}e^{-\lambda_2 u^{\alpha_2}-\lambda_1(v-u)^{\alpha_1}}}{\int_0^v \alpha_2\lambda_2 u^{\alpha_2-1}e^{-\lambda_2 u^{\alpha_2}-\lambda_1(v-u)^{\alpha_1}}du}.$$

  Thus, the conditional expectation is :

$$E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log(T_i)|V_i > v_i, \Delta_i = 0) = \int_0^{v_i} \log(t)f(t|v,\delta=0)dt,$$

$$E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(T_i^{\alpha_1}|V_i > v_i, \Delta_i = 0) = \int_0^{v_i} t^{\alpha_1} f(t|v, \delta = 0)dt,$$

$$E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log(U_i)|V_i > v_i, \Delta_i = 0) = \int_0^{v_i} \log(u) f(u|v, \delta = 0)du,$$

$$E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(U_i^{\alpha_1}|V_i > v_i.\Delta_i = 0) = \int_0^{v_i} u^{\alpha_1} f(u|v, \delta = 0)du.$$

As in the exponential-Weibull model described in Section 5.2.2 the integrals are computed numerically.

M-step
By using the following notation:

- $ul_1^{**} = E_{\alpha_2^{(k)},\lambda_2^{(k)}}(log(U_i)|U_i > v_i - t_i)$,
- $ue_1^{**} = E_{\alpha_2^{(k)},\lambda_2^{(k)}}(U_i^{\alpha_2}|U_i > v_i - t_i)$,
- $tl_1^{**} = E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log(T_i)|V_i = v_i, \Delta_i = 1)$,
- $te_1^{**} = E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(T_i^{\alpha_1}|V_i = v_i, \Delta_i = 1)$,
- $ul_2^{**} = E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log(U_i)|V_i = v_i, \Delta_i = 1)$,
- $ue_2^{**} = E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(U_i^{\alpha_2}|V_i = v_i, \Delta_i = 1)$,
- $tl_2^{**} = E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log(T_i)|V_i > v_i, \Delta_i = 0$,
- $te_2^{**} = E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(T_i^{\alpha_1}|V_i > v_i, \Delta_i = 0)$,
- $ul_3^{**} = E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(\log(U_i)|V_i > v_i, \Delta_i = 0)$,
- $ue_3^{**} = E_{(\alpha_1^{(k)},\lambda_1^{(k)},\alpha_2^{(k)},\lambda_2^{(k)})}(U_i^{\alpha_2}|V_i > v_i, \Delta_i = 0)$,

the function Q to be maximized is:

$$
\begin{aligned}
Q =& n\log(\alpha_1\lambda_1\alpha_2\lambda_2) - \lambda_1\Bigg[\sum_{i\in CR} t_i^{\alpha_1} + \sum_{i\in CRME}(te_1^{**})_i + \sum_{i\in CRMEM}(te_2^{**})_i\Bigg] - \\
& \lambda_2\Bigg[\sum_{i\in CRE} u_i^{\alpha_2} + \sum_{i\in CREM}(ue_1^{**})_i + \sum_{i\in CRME}(ue_2^{**})_i + \sum_{i\in CRMEM}(ue_3^{**})_i\Bigg] + \\
& (\alpha_1-1)\Bigg[\sum_{i\in CR}\log t_i + \sum_{i\in CRME}(tl_1^{**})_i + \sum_{i\in CRMEM}(tl_2^{**})_i\Bigg] + \\
& (\alpha_2-1)\Bigg[\sum_{i\in CRE}\log u_i + \sum_{i\in CREM}(ul_1^{**})_i + \sum_{i\in CRME}(ul_2^{**})_i + \sum_{i\in CRMEM}(ul_3^{**})_i\Bigg]
\end{aligned}
$$

where $Q = Q((\alpha_1,\lambda_1,\alpha_2,\lambda_2),(\alpha_1,\lambda_1,\alpha_2,\lambda_2)^{(k)})$ and the sets of the summary are defined as before (Section 5.2.2).
Since no closed form for the maximum likelihood estimators are available, they are compute by using numeric methods.

Variance estimation

The Weibull-Weibull model does not belong to the regular exponential family thus the variance is computed employing equation (5.4).
Denote by y_{obs} the observed data and by $\hat{\alpha}_1, \hat{\lambda}_1, \hat{\alpha}_2, \hat{\lambda}_2$ the maximum likelihood estimates.
The Fisher information for the shape and scale parameters α_1 and λ_1 are respectively

$$
\begin{aligned}
I(\hat{\alpha}_1|y_{obs}) &= \Big[E_{\alpha_1}(-\partial^2\ell_c/\partial\alpha_1^2|y_{obs}) - Var_{\alpha_1}(\partial\ell_c/\partial\alpha_1|y_{obs})\Big]_{\alpha_1=\hat{\alpha}_1} \\
&= \frac{n}{\hat{\alpha}_1^2} + \hat{\lambda}_1 E(T^{\hat{\alpha}_2}(\log T)^2|y_{obs}) - \\
&\quad \Big[\lambda_1^2 Var(T^{\hat{\alpha}_1}\log T|y_{obs}) + Var(\log T|y_{obs}) + \\
&\quad \hat{\lambda}_1^2 cov(T^{\hat{\alpha}_1}\log T, \log T|y_{obs})\Big]
\end{aligned}
\tag{5.13}
$$

and

$$
\begin{aligned}
I(\hat{\lambda}_1|y_{obs}) &= \Big[E_{\lambda_1}(-\partial^2\ell_c/\partial\lambda_1^2|y_{obs}) - Var_{\lambda_1}(\partial\ell_c/\partial\lambda_1|y_{obs})\Big]_{\lambda_1=\hat{\lambda}_1} \\
&= \frac{n}{\hat{\lambda}_1^2} - Var(T^{\hat{\alpha}_1}|y_{obs}).
\end{aligned}
\tag{5.14}
$$

The covariance between the maximum likelihood estimators of α_1 and λ_1 is computed by inverting the correspondent Fisher information:

$$\begin{aligned} I(\hat{\alpha}_1, \hat{\lambda}_1 | y_{obs}) &= \left[E_{\alpha_1,\lambda_1}\left(\frac{-\partial^1 \ell_c}{\partial\alpha_1 \partial\lambda_1} \middle| y_{obs} \right) - Var_{\alpha_1,\lambda_1}\left(\frac{\partial \ell_c}{\partial\alpha_1} \frac{\partial \ell_c}{\partial\lambda_1} \middle| y_{obs} \right) \right]_{\alpha_1=\hat{\alpha}_1, \lambda_1=\hat{\lambda}_1} \\ &= E(U^{\hat{\alpha}_1} \log U | y_{obs}) - \\ &\quad [cov(U^{\hat{\alpha}_1}, \log U | y_{obs}) - \hat{\lambda}_1 cov(U^{\hat{\alpha}_1}, U^{\hat{\alpha}_1} \log U | y_{obs})]. \end{aligned}$$

For α_2 and λ_2 the Fisher information is the same as (5.13)-(5.14) where the random variable T has been replaced by the random variable U, the parameters α_1 with α_2 and λ_1 with λ_2.
The variance of the maximum likelihood estimator is given by the inverse of the Fisher information.
All the expected values are computed numerically.
Results of the estimated model are reported in Table 5.3.

	α_1	λ_1	α_2	λ_2
mle	1.4833750	10.8748194	0.7219398	0.18683546
std	0.028802478	0.444622458	0.032958306	0.008769222

Table 5.3: MLE and std for Weibull-Weibull combination.

The estimated parameters for the second interval are very close to the one computed in the previous model, (see Table 5.2). The estimated OS and its relative pointwise confidence interval are very similar to Figure 5.2 and therefore are not shown here.
R code to implement the EM algorithm for the exponential-exponential model is given in the Appendix A.3. The code for the remaining models can be found online at "http://tesi.cab.unipd.it/".

Chapter 6

Competing risks analysis

Competing risks concern the situation where more than one cause of failure is possible. If failures are different causes of death, only the first of these to occur is observed. In other situations, other events after the first failure may be observed but the investigator is not interested in them. For the case study explored in this thesis clinicians are interested in the occurrence of relapse from complete remission. Therefore if death occurs before relapse it is a competing event.
In analogy with the analysis performed in Chapters 3-4, before estimating the cumulative incidence of relapse, all missing times to complete remission will be estimated by employing multiple imputation and the parametric approach described in previous chapters.
In Section 6.1 an introduction to the notation used for the competing risks is provided. In Sections 6.2-6.3 the two methodologies (multiple imputation and parametric approach) and the application to the case study are respectively described.

6.1 Notation

Figure 6.1 shows the division of the time interval $[t_0, t_{event}]$ under investigation into two intervals ($[t_0, t_{event}]$ and $[t_{cr}, t_{event}]$) as described in Section 2.4.

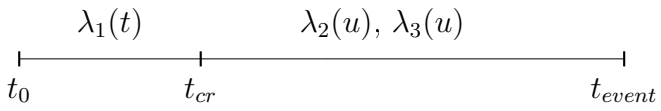

Figure 6.1: Time interval.

As before, in the interval $[t_0, t_{event}]$ the event of interest is the achievement of complete remission. While in the interval $[t_{cr}, t_{event}]$ there are two competing events: relapse and death. Figure 6.2 shows the competing risks models under study in this thesis. The competing risks model is represented graphically with the initial state complete remission (CR) and two different endpoints (relapse and death).

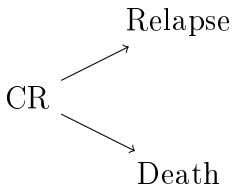

Figure 6.2: Competing risk models with two causes of failure.

Define T the random variable describing time from diagnosis to time to complete remission $t \geq 0$. The hazard function associated to the achievement of complete remissions is indicated by $\lambda_1(t)$. Let U be the random variable representing time to event from complete remission, $u \geq 0$. Let V be the sum of T and U.
In the interval $[t_{cr}, t_{event}]$ relapse and death are the two competing events, since a patient may die before experiencing a relapse.
Define U_2 the random variable representing survival time until relapse and U_3 the random variable describing the survival time until death. The random variables U_2 and U_3 are independent and U is defined as $min(U_2, U_3)$. The hazard function associated with relapse and death are indicated by $\lambda_2(u)$ and $\lambda_3(u)$ respectively (see Fig. 6.1).

6.2 Multiple imputation

Multiple imputation is a technique for estimating missing data which consists in replacing every missing value with a set of plausible values. The algorithm is described in details in Chapter 3.1.
Before estimating the cumulative incidence of relapse, the missing times of complete remission are imputed by employing the empirical cumulative distribution. The imputation algorithm is the same used for estimating the overall survival (see Chapter 3.2). Briefly, the missing time to complete remission are imputed by drawing times from the set of known time to complete remission, with probability estimated by the empirical distribution.
Five complete data sets are reconstructed and for each data set the cumulative incidence is estimated. Define $C_k(t), k = 1, 2$ the cumulative incidence

function of the event k at time t. Let

$$C^*_{k1}(t), C^*_{k2}(t), ..., C^*_{k5}(t), \qquad k = 1, 2$$

be the five estimates computed from the imputed data set.
Let also

$$V^*_{k1}(t), V^*_{k2}(t), ..., V^*_{k5}(t), \qquad k = 1, 2$$

be the estimated variances computed employing the Greenwood variance estimators.
The five estimated statistics are averaged to obtain an unique indicative statistic

$$\bar{C_k}(t) = \frac{1}{5}\sum_{i=1}^{5} C^*_{ki}(t), \qquad k = 1, 2.$$

The total variance estimated is given by:

$$T_k(t) = \bar{V_k}(t) + \left(1 + \frac{1}{5}\right) B_k(t), \qquad k = 1, 2$$

where

$$\bar{V_k}(t) = \sum_{i=1}^{5} \frac{1}{5} V_{ki}(t), \qquad k = 1, 2$$

represent the within-imputation variance and

$$B_k(t) = \sum_{i=1}^{5} \frac{1}{5-1}(C^*_k(t) - \bar{C_k}(t))^2, \qquad k = 1, 2$$

the between imputation variance.
The confidence interval for the estimated cumulative incidence of relapse ($\bar{C_1}(t)$) is computed in the following way:

$$\bar{C_1}(t) \pm t_\nu(\alpha/2)\sqrt{T_1(t)}$$

where t_v is the quantile of the student distribution with ν degree of freedom (see eq. (3.5)).
Figure 6.3 shows on the black line the estimated cumulative incidence of relapse $\bar{C_1}$ and relative confidence intervals. The red line represents the estimated cumulative incidence when only patients with time to complete remission known are included. The confidence interval for $C_1(t)$ computed from the incomplete date set (i.e including patients with known time to complete remission) are based on the asymptotic normality of the cumulative incidence (i.e.$\hat{C_1}(t)(t) \pm z_{1-\alpha/2}\hat{V}[\hat{C}_1(t)]$).

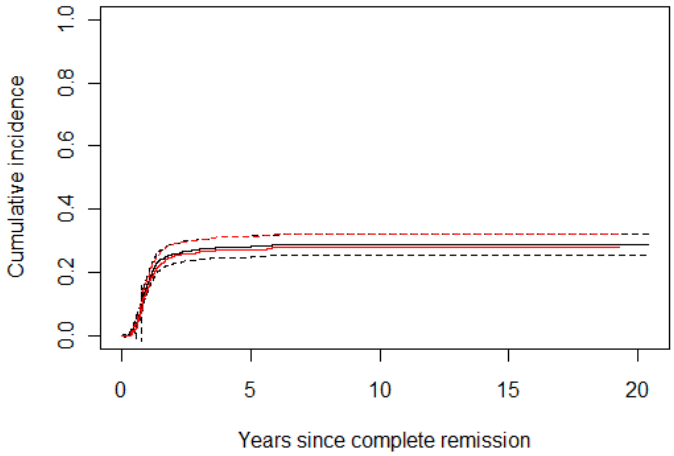

Figure 6.3: Cumulative incidence function for relapse.

6.3 Parametric approach

The parametric approach consists in employing a parametric distribution to describe failure time of a patient.
The random variables T, U_2 and U_3 follow a specific distribution chosen by looking at the non-parametric cumulative hazard computed from the observed data set in which patients with time to complete remission missing were excluded. Figure 6.4 shows the non parametric cumulative hazard (full line) and the exponential cumulative hazard (dashed line) in $[t_0, t_{CR}]$. To the random variable T is associated an exponential distribution.
Figure 6.5 shows on the black line the non parametric cumulative hazard for the two competing events relapse and death including only patients with time to complete remission known in $[t_{cr}, t_{event}]$ (time interval has been rescaled). The red line represents the exponential distribution while the blue line corresponds to the Weibull distribution.
In both intervals the Weibull distribution seems to adequately fits the data. The non parametric cumulative hazard curves plotted in Figure 6.5, even referring to a subset of patients, have a more structured behavior compared to the cumulative hazard of the exponential distribution. The propensity toward the choice of a simple distribution is most of all due to the reduced complexity of the computations allowing a clear understanding of the method applied.

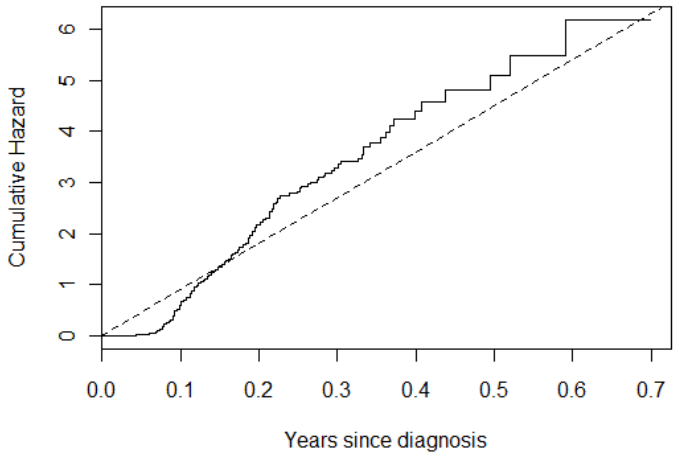

Figure 6.4: Cumulative hazard in the interval $[t_0, t_{CR}]$. Dashed line: exponential distribution, full line: non-parametric.

Either the model where the random variables $U1$ and $U2$ are both exponential or Weibull are applied to the data.
The likelihood function for the complete data is computed according to the parametric distribution and the parameters are estimated.
The likelihood for competing risks in the two intervals is defined as follows:

$$L = \lambda_1(t)S(t)\lambda_2(u)^{\delta_2}\lambda_3(u)^{\delta_3}S(u) \tag{6.1}$$

where λ_1 and $S(t)$ are the hazard and the survival in the first interval; λ_2 and λ_3 are the cause specific hazard rate respectively for relapse and death, $S(u)$ is the overall survival at u and

$$\delta_2 = \begin{cases} 1 & \text{if relapse has occurred} \\ 0, & \text{otherwise.} \end{cases} \qquad \delta_3 = \begin{cases} 1 & \text{if death has occurred} \\ 0, & \text{otherwise.} \end{cases}$$

From the maximum likelihood estimates it is possible to compute the cumulative incidence of relapse and relative confidence interval.
Consider two competing events. Define $\theta{=}(\theta_1, \theta_2)$ a vector of parameters where θ_1 is related to the competing event 1 and θ_2 to the event 2.
Define $C_k(t, \theta)$ the cumulative incidence function of the event $k = 1, 2$ at time

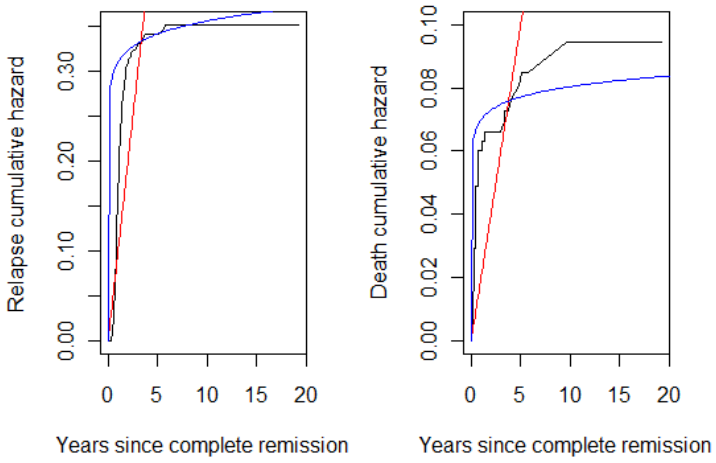

Figure 6.5: Cumulative hazard. Black line: non-parametric, blue line: Weibull distribution, red line: exponential distribution.

t. Let $\hat{\theta}$ be the maximum likelihood estimate of θ and $\hat{C}_k(t, \hat{\theta})$ the estimated cumulative incidence of event k at time t. The pointwise 95% confidence interval at a generic time t, for the estimated parametric cumulative incidence of event k is:

$$\hat{C}_k(t, \hat{\theta}) \pm 1.96\sqrt{\hat{var}(\hat{C}_k(t, \hat{\theta}))} \tag{6.2}$$

where $\widehat{var}(\hat{C}_k(t, \hat{\theta}))$ is the estimated variance of the cumulative incidence computed by the multivariate delta method in the following way:

$$\widehat{var}(\hat{C}_k(t, \hat{\theta})) = \left(\frac{\partial C_k(t, \theta)}{\partial \theta}\right)\Bigg|_{\theta=\hat{\theta}} \widehat{cov}(\hat{\theta}) \left(\frac{\partial C_k(t, \theta)}{\partial \theta}^T\right)\Bigg|_{\theta=\hat{\theta}} \tag{6.3}$$

where $\widehat{cov}(\hat{\theta})$ is the variance matrix of the maximum likelihood estimators, i.e. the inverse of the Fisher information matrix.
The equation (6.1) is suitable for the subject whose time to complete remission is known. The problem of missingness of time to complete remission is solved by integrating out the likelihood for all the values assumed by time to complete remission.
In analogy with the estimation of the overall survival, all two combinations of the random variables U_2 and U_3 to estimate the cumulative incidence of relapse are described.

6.3.1 Exponential distribution for the random variables T, U_2 and U_3

Let T, U_2 and U_3 be exponentially distributed with parameter λ_1, λ_2 and λ_3 respectively. The likelihood for a patient with time to complete remission known is given by:

$$L = \lambda_1 e^{-\lambda_1 t} \lambda_2^{\delta_2} \lambda_3^{\delta_3} e^{-(\lambda_2+\lambda_3)u}.$$

When time to complete remission is missing the likelihood for a patient assumes the following form:

$$\begin{aligned} L &= \int_0^v \lambda_1 e^{-\lambda_1 t} \lambda_2^{\delta_2} \lambda_3^{\delta_3} e^{-(\lambda_2+\lambda_3)u} dt \\ &= \lambda_1 \lambda_2^{\delta_2} \lambda_3^{\delta_3} e^{-(\lambda_2+\lambda_3)v} \left(\frac{e^{(\lambda_2+\lambda_3-\lambda_1)v} - 1}{\lambda_2+\lambda_3-\lambda_1} \right) \end{aligned}$$

where v is the time of event (death or relapse).
The likelihood for the complete data set is then given by:

$$L = \prod_{i=1}^{n} \lambda_1 \lambda_2^{\delta_{2i}} \lambda_3^{\delta_{3i}} e^{-(\lambda_2+\lambda_3)v} \left(e^{(\lambda_2+\lambda_3-\lambda_1)t_i} \right)^{r_i} \left(\frac{e^{(\lambda_2+\lambda_3-\lambda_1)v_i} - 1}{\lambda_2+\lambda_3-\lambda_1} \right)^{(1-r_i)}$$

where $r = 1$ if time to complete remission is known, $r = 0$ otherwise.
The function to maximize is the log likelihood given by:

$$\begin{aligned} \ell =& n \log(\lambda_1) + nr \log(\lambda_2) + nd \log(\lambda_3) - (\lambda_2+\lambda_3) \sum_{i \in 1}^{n} v + \\ & (\lambda_2+\lambda_3-\lambda_1) \sum_{i \in CR} t_i + \sum_{i \in CRM} \log \left(\frac{e^{(\lambda_2+\lambda_3-\lambda_1)v_i} - 1}{\lambda_2+\lambda_3-\lambda_1} \right) \end{aligned}$$

where nr is the number of relapse, nd is the number of death, CR is the set of patients with time to complete remission known and CRM is the set of patients with time to complete remission missing.
Since it is not possible to find a closed form for the maximum likelihood estimators, the log likelihood is numerically maximized by using the R function `nlminb`. The maximum likelihood estimates with associated standard errors are shown in Table 6.1.
Confidence interval is computed by employing equation (6.2).
Variance estimates associated to the estimated cumulative incidence is computed by employing the multivariate delta method (see eq. (6.3)).
Denote by $\theta = (\lambda_2, \lambda_3)$ the vector of the hazard rate respectively for relapse

	λ_1	λ_2	λ_3
mle	7.74146409	0.05127785	0.01575971
std	0.347049057	0.003473122	0.001925380

Table 6.1: MLE and std for exponential-exponential combination.

(λ_2) and for death (λ_3).
The cumulative incidence of relapse at a generic time t is given by:

$$\begin{aligned} C_1(t) &= \int_0^t \lambda_2 e^{-(\lambda_2+\lambda_3)x} dx \\ &= \frac{\lambda_2}{\lambda_2+\lambda_3}(1 - e^{-(\lambda_2+\lambda_3)t}). \end{aligned}$$

By employing equation (6.3) the vector of the first derivatives of the cumulative incidence of relapse with respect to θ is:

$$\begin{aligned} \frac{\partial C_1(t,\theta)}{\partial \theta} &= \left(\frac{\partial C_1(t,\theta)}{\partial \lambda_2}, \frac{\partial C_1(t,\theta)}{\partial \lambda_3}\right) \\ &= \left(\frac{\lambda_3}{(\lambda_2+\lambda_3)^2}(1 - e^{-(\lambda_2+\lambda_3)t}) + \frac{\lambda_2}{\lambda_2+\lambda_3}(te^{-(\lambda_2+\lambda_3)t}),\right. \\ &\quad \left. - \frac{\lambda_2}{(\lambda_2+\lambda_3)^2}(1 - e^{-(\lambda_2+\lambda_3)t})\right) \end{aligned}$$

The estimated cumulative incidence of relapse and relative confidence interval, computed by employing equation 6.2, are plotted in Figure 6.6.

6.3.2 Exponential distribution for the random variable T and Weibull distribution for the random variables U_2 and U_3

The Weibull distribution with two parameters, shape and scale, allows a more flexible fitting to the data.
Let $T \sim Exp(\lambda_1)$, $U_2 \sim Weibull(\alpha_2, \lambda_2)$ and $U_3 \sim Weibull(\alpha_3, \lambda_3)$. Shape parameters are indicated by α_2 and α_3, while scale parameters are indicated by λ_2 and λ_3.
The likelihood for a patient with time to complete remission known is given as follow:

$$L = \alpha_1(\alpha_2\lambda_2 u^{\alpha_2-1})^{\delta_2}(\alpha_3\lambda_3 u^{\alpha_3-1})^{\delta_3} e^{-\lambda_1 t - \lambda_2 u^{\alpha_2} - \lambda_3 u^{\alpha_3}}.$$

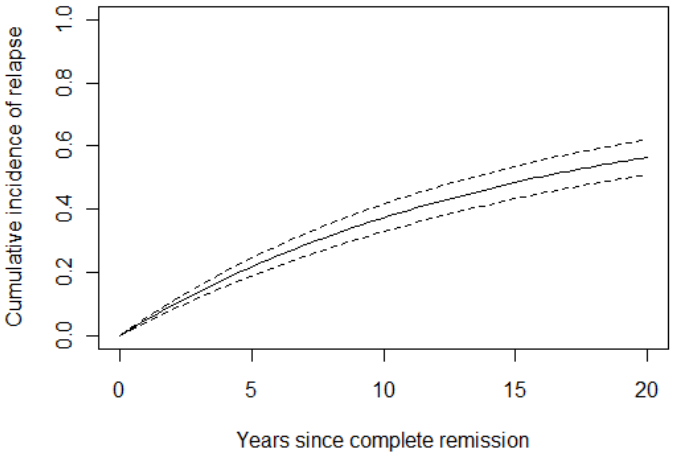

Figure 6.6: Cumulative incidence function based on the exponential model.

If time to complete remission is missing the contribution to the likelihood for a patient is given by the integral:

$$\int_0^v \alpha_1(\alpha_2\lambda_2(v-t)^{\alpha_2-1})^{\delta_2}(\alpha_3\lambda_3(v-t)^{\alpha_3-1})^{\delta_3}e^{-\lambda_1 t-\lambda_2(v-t)^{\alpha_2}-\lambda_3(v-t)^{\alpha_3}}dt$$

where v is time of event.
Since it is not possible to analytically compute the integral, the equation of the complete likelihood is given in the following form:

$$L=\prod_{i=1}^{n}\left[\alpha_1(\alpha_2\lambda_2 u^{\alpha_2-1})^{\delta_2}(\alpha_3\lambda_3 u^{\alpha_3-1})^{\delta_3}e^{-\lambda_1 t-\lambda_2 u^{\alpha_2}-\lambda_3 u^{\alpha_3}}\right]^{r_i}\cdot$$
$$\left[\int_0^v \alpha_1(\alpha_2\lambda_2(v-t)^{\alpha_2-1})^{\delta_2}(\alpha_3\lambda_3(v-t)^{\alpha_3-1})^{\delta_3}e^{-\lambda_1 t-\lambda_2(v-t)^{\alpha_2}-\lambda_3(v-t)^{\alpha_3}}dt\right]^{1-r_i}.$$

Again, the integrals and the maximum likelihood estimates are numerically obtained by using the R functions `integrate` and `nlminb`.
Table 6.2 shows the maximum likelihood estimates and associated standard errors.
The cumulative incidence of relapse at a generic time t is given by:

$$C_1(t,\alpha_2,\lambda_2)=\int_0^t \alpha_2\lambda_2 x^{\alpha_2-1}e^{-\lambda_2 x^{\alpha_2}-\lambda_3 x^{\alpha_3}}dx.$$

	λ_2	α_2	λ_2	α_3	λ_3
mle	7.432090	0.533934	0.127592	0.388165	0.050397
std	1.1977e-01	1.0747e-03	1.3044e-04	2.2019e-03	5.3206e-05

Table 6.2: MLE and std for Weibull-Weibull combination.

Denote by $\theta = (\theta_2, \theta_3)$ where $\theta_2 = (\alpha_2, \lambda_2)$ and $\theta_3 = (\alpha_3, \lambda_3)$ the vector of the parameters associated to the Weibull distributions. The first derivatives of the cumulative incidence function of relapse with respect to θ are computed in the following way:

$$\begin{aligned}\frac{\partial C_1(t,\theta)}{\partial \theta} &= \left(\frac{\partial C_1(t,\theta)}{\partial \alpha_2}, \frac{\partial C_1(t,\theta)}{\partial \lambda_2}, \frac{\partial C_1(t,\theta)}{\partial \alpha_3}, \frac{\partial C_1(t,\theta)}{\partial \lambda_3}\right) \\ &= \left(\int_0^t \frac{\partial}{\partial \alpha_2}\theta_2(x)S(x,\theta)dx, \int_0^t \frac{\partial}{\partial \lambda_2}\theta_2(x)S(x,\theta)dx, \right. \\ &\quad \left. \int_0^t \frac{\partial}{\partial \alpha_3}\theta_2(x)S(x,\theta)dx, \int_0^t \frac{\partial}{\partial \lambda_3}\theta_2(x)S(x,\theta)dx\right),\end{aligned}$$

where

$$\frac{\partial C_1(t,\theta)}{\partial \alpha_2} = \int_0^t \lambda_2 x^{\alpha_2-1} e^{-\lambda_2 u^{\alpha_2} - \lambda_3 u^{\alpha_3}} [1 + \alpha_2 \log(x) - \alpha_2 \lambda_2 x^{\alpha_2} \log(x)] dt,$$

$$\frac{\partial C_1(t,\theta)}{\partial \lambda_2} = \int_0^t \alpha_2 x^{\alpha_2-1} e^{-\lambda_2 u^{\alpha_2} - \lambda_3 u^{\alpha_3}} [1 - \alpha_2^2 x^{\alpha_2} \log(x)] dt,$$

$$\frac{\partial C_1(t,\theta)}{\partial \alpha_3} = \int_0^t -\alpha_2 \lambda_2 \lambda_3 x^{\alpha_2+\alpha_3-1} \log(x) e^{-\lambda_2 u^{\alpha_2} - \lambda_3 u^{\alpha_3}} dt,$$

$$\frac{\partial C_1(t,\theta)}{\partial \lambda_3} = \int_0^t -\alpha_2 \lambda_2 x^{\alpha_2+\alpha_3-1} e^{-\lambda_2 u^{\alpha_2} - \lambda_3 u^{\alpha_3}} dt.$$

Figure 6.7 shows the confidence interval computed employing equation 6.2. R code to estimate the parametric model described in Section 6.3.2 is given in the Appendix A Section 2.2. The R-code for multiple imputation and the parametric model described in Section 6.3.1 are provided online at "http://tesi.cab.unipd.it/".

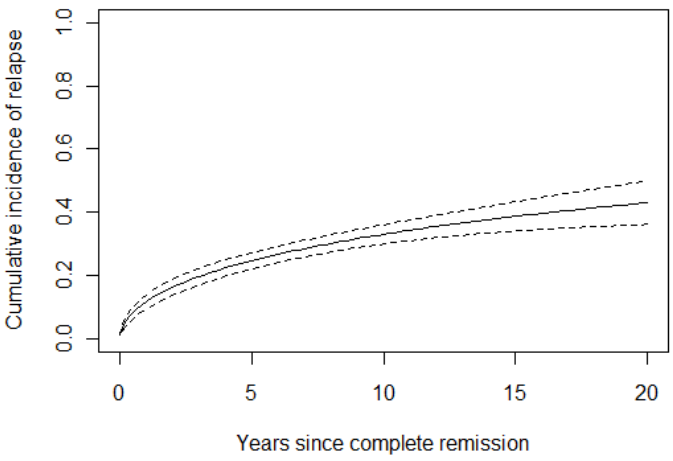

Figure 6.7: Cumulative incidence function based on the Weibull model.

Chapter 7

Simulation study

In this chapter a large simulation study was performed to compare the techniques proposed in this thesis to deal with the problem of missing data. Performance of parametric (parametric approach, EM-algorithm) and non parametric (multiple imputation) methodology are compared. The simulation procedure consists in constructing a number of *complete data sets*, with censored observation and missing values, by considering different scenarios. Several different patients sample size, percentage of missing time to complete remission and percentage of censored observations are considered.
Failure times (complete remission, relapse and death) are drawn from known parametric distributions. By applying on such generated data sets the techniques proposed in the previous chapters the statistics of interest are then estimated.
Bias and mean square error are then investigated in order to evaluate the performance of the methodology proposed to deal with the problem analyzed in this thesis.
The methodology used for simulations is described in details in Section 7.1. Simulation results are discussed in Section 7.2.

7.1 Method description

Let T_1, T_2 and T_3 be the random variables representing respectively time to complete remission, time from complete remission to the occurrence of relapse and time from complete remission to the event death. Let T_1, T_2 and T_3 be exponentially distributed with parameters λ_1, λ_2 and λ_3 respectively. Parameters values for the distributions T_i $i = 1, 2, 3$ were obtained by graphically investigating the failure times histogram in the real data set. In particular a constant hazard rate equals to 7.6 was associated to the achievement of com-

plete remission. Hazard of relapse (λ_2) and death (λ_3) were fixed to 0.27 and 0.09 respectively. Complete data sets were generated by simulating under different scenarios from the distributions described above. Simulations were varied in sample size (n), percentage of missing time to complete remission (mp) and percentage of censored observation (cp). In particulars sample size was set as small $n = 250$, moderate $n = 500$, and large $n = 1000$. The percentages of missing time to complete remission were chosen a bit lower and a bit higher than the observed percentage on the real data set, which is about 40%. Three percentages of censored observations were chosen, equal to the observed percentage in the real data (60%), lower (50%) and a higher (70%). The three simulations parameters, n, cp and mp are then combined together producing nine different simulation scenarios. In Table 7.1 the 9 scenarios indicated as n_i for $i = 1, \ldots, 9$ are illustrated.

	n	cp(%)	mp(%)
n_1	250	70	50
n_2	250	50	30
n_3	250	60	50
n_4	500	70	50
n_5	500	50	30
n_6	500	60	50
n_7	1000	70	50
n_8	1000	50	30
n_9	1000	60	50

Table 7.1: Simulations scenarios.

For each combinations of different settings of n, cp and mp, in Table 7.1, $M = 10000$ data sets were generated.
Steps in the simulation process

1. Simulate n time to complete remission (t_{cr}) from $T_1 \sim Exp(\lambda_1)$
2. Simulate n time to relapse (t_R) from $T_2 \sim Exp(\lambda_2)$
3. Simulate n time to death (t_D) from $T_3 \sim Exp(\lambda_3)$
4. Time to event is $t_{event} = min(t_R, t_D)$

5. Simulate $n * cp$ censored observations from the binomial distribution $\mathrm{B}(n, cp)$

6. Simulate $n * cm$ missing time to complete remission from the binomial distribution $\mathrm{B}(n, mp)$

Repeat steps 1-6 M times.
The statistics of interest, overall survival and cumulative incidence of relapse are estimated for each of the M simulated data sets by employing non-parametric and parametric techniques described in Chapters 3-6. Bias and mens square error are then computed.
Let

$$OS(t; \lambda_2, \lambda_3) = -e^{-(\lambda_2+\lambda_3)t}$$

and

$$C_2(t; \lambda_2, \lambda_3) = \frac{\lambda_2}{\lambda_2 + \lambda_3}\left(1 - e^{-(\lambda_2+\lambda_3)t}\right)$$

respectively the parametric overall survival and the cumulative incidence of relapse from time to complete remission based on the exponential model.
Let $t_k = k, k = 1, 2.., 6$ be the time points in years at which the statistics were computed. The follow up is restricted to the first 6 years since the events of interest usually occur in the first 5-6 years.
Let $OS_i^*(t_k), i = 1, 2, ..., M$ be the M estimates of the overall survival at time t_k. For the parametric method the estimated overall survival is given by:

$$OS_i^*(t_k) = OS_i^*(t_k, \lambda_{2i}^*, \lambda_{3i}^*)$$

where λ_{2i}^* and λ_{3i}^* are the estimates of the parameters in the M data sets.
Bias and mean square error of the overall survival at a specific point t_k are given by:

$$B_{OS} = \frac{1}{M}\sum_{i=1}^{M}\left(OS_i^*(t_k) - OS(t_k; \lambda_2, \lambda_3)\right),$$

$$MSE_{OS} = \frac{1}{M}\sum_{k=1}^{M}\left(OS_i^*(t_k) - OS(t_k; \lambda_2, \lambda_3)\right)^2$$

respectively.
Similarly, let $C_{2i}^*(t_k), i = 1, 2, .., M$ be the M estimates of the cumulative incidence of relapse at time t_k. Again, as the overall survival, the estimated cumulative incidence for the parametric methods is given by:

$$C_{2i}^*(t_k) = C_{2i}^*(t_k, \lambda_{2i}^*, \lambda_{3i}^*)$$

Bias and the mean square error are given as follow:

$$B_{C_2} = \frac{1}{M}\sum_{i=1}^{M}\left(C^*_{2i}(t_k) - C_2(t_k;\lambda_2,\lambda_3)\right)$$

and

$$MSE_{C_2} = \frac{1}{M}\sum_{i=1}^{M}\left(C^*_{2i}(t_k) - C_2(t_k;\lambda_2,\lambda_3)\right)^2.$$

7.2 Simulated results

Tables 7.2-7.4 show part of a large simulations study performed in order to compare the three methods investigated in this thesis. The simulation results correspond to the bias and mean square error computed at time $t_1 = 1$ under different scenarios as described in Table 7.1.
Multiple imputation associated to the non parametric techniques, parametric approach and EM algorithm for the parametric methods are compared in term of their bias and mean square error. In particular the exponential model on both interval was employed either for the parametric approach (here indicated as PA Exp-Exp) and for the EM algorithm (indicated as EM Exp-Exp). All details concerning these methods are given respectively in Section 4.2 and Section 5.2.1.
The exponential and Weibull model was only used for the parametric approach (here indicated as PA Exp-Weib), for details see Section 4.3.
As it can be seen from Tables 7.2-7.4, multiple imputation and parametric approach performances are quite similar. When the sample size is smaller (i.e. the simulations design indicated with n_1, n_2, n_3 in Table 7.1), bias and means square error associated to the parametric approach are a slight lower than the one computed with multiple imputation. This difference is equal to zero as the sample size increase.
When the percentage of missing time to complete remission and censored observation is very high (see simulations design n_1, n_4, n_7) either multiple imputation and the parametric approach lead to high values for bias and means square error.
A completely different situation is observed for the EM algorithm. In every scenario, even with high percentage of censored observations (70%) and missing times to complete remission (50%), the EM algorithm leads to small values for bias and means square error. This seems to suggest a better performance of the EM algorithm compared to the other techniques.
These results are also confirmed from Figure 7.1. Figure 7.1 represents the

bias for the overall survival computed at $t_k = k, k = 1, 2.., 6$ under the simulations scenario n_4, n_5, n_6. The blue circles correspond to the EM algorithm; the red circles represent the exponential-exponential model in the parametric approach and the green circles correspond to the multiple imputation method. As it can be seen from Figure 7.1, the parametric approach and multiple imputation method have the same behavior (green circles are almost covered by red circles) with high values for bias compared to EM algorithm which bias values are around zero.
It is rather difficult to give some guidelines about which method should be used to reconstruct the missing values. The non parametric approach has the advantage of not imposing any parametric model. From the computational point of view is also less demanding.
On the other hand although the EM methodology is rather demanding in terms of computations and implementation, it is well known that it is the more robust technique when missing data are present.
Future research should be done where imputed values are based on regression model and patients characteristics are considered.

Method	**Simulations design**		
	n_1	n_2	n_3
	Overall Survival		
Non-parametric			
MI	0.200 (0.040)	0.138(0.019)	0.168(0.029)
Parametric			
PA Exp-Exp	0.199(0.0399)	0.137 (0.018)	0.168 (0.028)
PA Exp-Weib	0.201(0.041)	0.138(0.020)	0.169 (0.029)
EM Exp-Exp	-0.0728(0.006)	0.017 (0.001)	-0.095(0.009)
	Cumulative Incidence		
Non-parametric			
MI	-0.150(0.022)	-0.104(0.011)	-0.126(0.016)
Parametric			
PA Exp-Exp	-0.150(0.022)	-0.103(0.011)	-0.126 (0.015)

Table 7.2: Simulation results. Bias(MSE) for different scenarios.

Method	Simulations design		
	n_4	n_5	n_6
Overall Survival			
Non-parametric			
MI	0.200 (0.040)	0.138(0.019)	0.168 (0.028)
Parametric			
PA Exp-Exp	0.200 (0.040)	0.137 (0.019)	0.168 (0.028)
PA Exp-Weib	0.200 (0.040)	0.169 (0.029)	0.17(0.029)
EM Exp-Exp	-0.072(0.005)	0.017 (0.000)	-0.094(0.009)
Cumulative Incidence			
Non-parametric			
MI	-0.150(0.023)	-0.103(0.010)	-0.126(0.016)
Parametric			
PA Exp-Exp	-0.149 (0.022)	-0.126(0.016)	-0.126(0.015)

Table 7.3: Continue: Simulation results.Bias(MSE) for different scenarios.

Method	Simulations design		
	n_7	n_8	n_9
Overall Survival			
Non-parametric			
MI	0.200 (0.040)	0.137(0.019)	0.169 (0.028)
Parametric			
PA Exp-Exp	0.200 (0.040)	0.137(0.019)	0.168 (0.028)
PA Exp-Weib	0.200 (0.040)	0.138 (0.019)	0.168(0.028)
EM Exp-Exp	-0.071(0.005)	0.018(0.000)	-0.093(0.009)
Cumulative Incidence			
Non-parametric			
MI	-0.150(0.023)	-0.103(0.010)	-0.126(0.016)
Parametric			
PA Exp-Exp	-0.150(0.022)	-0.103 (0.010)	-0.126(0.015)

Table 7.4: Continue: Simulation results. Bias(MSE) for different scenarios.

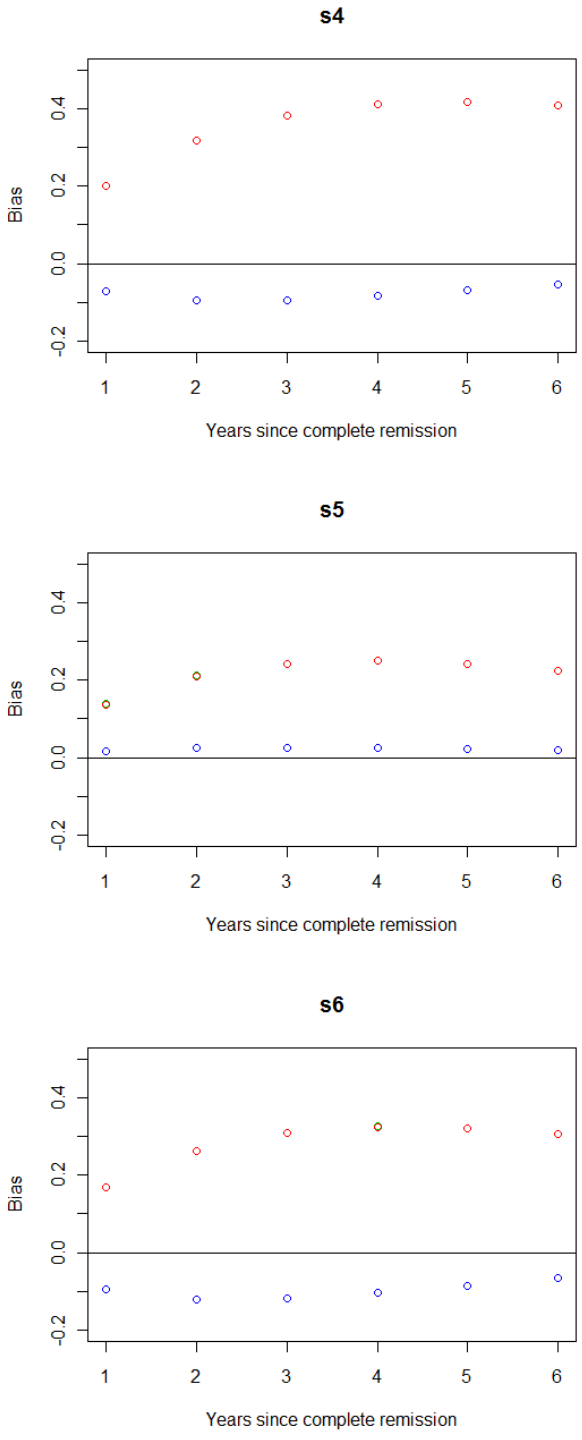

Figure 7.1: Simulations bias.○: MI;○: PA Exp-Exp and ○: EM Exp-Exp

Discussion

In this thesis we have proposed different techniques to deal with the problem of missing time to complete remission for a retrospective worldwide study involving children suffering from acute myeloid leukemia.
The techniques proposed were compared through a simulation study where different scenarios were considered. Sample size, percentage of censored observation and percentage of missing time to complete remission were considered in order to evaluate the performance of the methodology proposed under different scenarios.
First a non parametric method was applied. Multiple imputation by sampling from the observed time to complete remission has not forced the data to follow a fixed distribution. However, the estimated overall survival and cumulative incidence based only on the observed data and on the complete data set, where the unknown time to complete remission have been imputed, show very similar results.
Two parametric methods have also been presented. Parametric methodologies have the disadvantage to impose a specific parametric model on the data. However, if the parametric assumption, by an inspection of the data suggests that the model fit is adequate, theoretical results, such as estimator distribution, mean and variance of random variables, are already provided.
In the parametric approach several parametric models have been investigated. In order to estimate the overall survival the combination between exponential and Weibull distribution appears to be the best combination. More specific an exponential random variable has been chosen to describe the event complete remission from diagnosis while a Weibull distribution is more appropriate for describing time to death or relapse from complete remission.
Even by assuming that the parametric model fits data in a proper way, the performance of the technique is rather poor. Bias and mean square error are very similar to the one computed by multiple imputation. The parametric approach does not provide an analytical formula for the likelihood function and therefore the maximization of the log likelihood requires the use of nu-

merical methods.
The last methodology used to deal with missing data is the EM algorithm. EM techniques are used to deal with situations where the observed likelihood is intractable. This is carried out by creating a link between the complete log likelihood and the observed log likelihood. In this way the complexity of the problem is reduced. However, for the case under study, the application of the algorithm was computationally demanding. Two different aspects are responsible for the computations: different type of missing informations to include (i.e. censored observation and missing time to complete remission) and the distributions used to formulate the likelihood. Models, where the Weibull distribution is used to describe time to failure, do not allow to compute analytically the maximum likelihood estimators.
The EM algorithm shows the best performance throughout the simulations study. Bias and mean square errors are the smallest compared to multiple imputation and parametric approach. The results are rather good even in the presence of high percentages of missing time to complete remission and censored observation.
Based on the simulations study performed in this thesis, the EM algorithm shows a good performance when dealing with missing time to complete remission when compared to the other techniques.
It could be interesting to study the methodologies proposed in this thesis when regression models are involved in the imputing process. This might be investigated in future research.

Appendix A

R-code

A.1 Multiple Imputation

A.1.1 Imputation procedure

```
#Data set including patients with time_CR known
datacr<-data[!is.na(data$time_CR),]

# Estimate empirical cumulative distribution of time_CR
ed<-ecdf(datacr$time_CR)
taus<-sort(unique(datacr$time_CR)) #time_CR
crcd<-ed(taus) #empiric cumulative distribution time_CR
tci<-cbind(taus,crcd)
jumps<-diff(c(0,crcd))

#Set of times to event for missing values
mvdtevent<-data$time_event[is.na(data$time_CR)]

#Five samples time_CR
timej<-matrix(0,1,5)
for (i in 1:length(mvdtevent))
{
  txt<- which(taus< mvdtevent[i])
  tausj<-taus[txt]
  jumpsj<-jumps[txt]
  timej<-rbind(timej,sample(tausj,size=5,prob=jumpsj))

}
timej<-timej[-1,]
```

```
#Five imputed data sets
np=nrow(data)
data1<-data2<-data3<-data4<-data5<-data
j=1
for (i in 1:np)
{
  if(is.na(data$time_CR[i]))
  {
    data1$time_CR[i]<-timej[j,1]
    data2$time_CR[i]<-timej[j,2]
    data3$time_CR[i]<-timej[j,3]
    data4$time_CR[i]<-timej[j,4]
    data5$time_CR[i]<-timej[j,5]
    j<-j+1
  }
}

#Function joint; input: 5 objects of type
#(time,statisticOfInterest); output: a matrix
#with 6 columns where for each distinct
# timepoint is computed the statistics of interest
joint<-function(a,b,c,d,e)
{
  na<-nrow(a)
  nb<-nrow(b)
  nc<-nrow(c)
  nd<-nrow(d)
  ne<-nrow(e)
  #five matrix in the first column time,
  # one column statistic of interest,
  #and the other filled with 0
  a<-cbind(a,rep(0,na),rep(0,na),rep(0,na),rep(0,na))
  bi<-cbind(b[,1],rep(0,nb),b[,2],rep(0,nb),rep(0,nb),rep(0,nb))
  ci<-cbind(c[,1],rep(0,nc),rep(0,nc),c[,2],rep(0,nc),rep(0,nc))
  di<-cbind(d[,1],rep(0,nd),rep(0,nd),rep(0,nd),d[,2],rep(0,nd))
  ei<-cbind(e[,1],rep(0,ne),rep(0,ne),rep(0,ne),rep(0,ne),e[,2])
  mat<-rbind(a,bi,ci,di,ei)
  mat1<-mat[order(mat[,1]),] #matrix ordered by time
  time<-unique(mat1[,1])
  lt<-length(time)
```

```
mci<-matrix(0,1,6)
#Loop: the row with same time are summed up
for (i in 1:lt)
{
  temp<-c(0,0,0,0,0,0)
  for (j in 1:nrow(mat1))
  {
    if(time[i]==mat1[j,1])
    {
      temp<- temp+mat1[j,]
    }
  }
  temp[1]<-time[i]
  mci<-rbind(mci,temp)
}
row.names(mci)<-NULL
last<-c(max(which(mci[,2]>0)),max(which(mci[,3]>0)),
 max(which(mci[,4]>0)), max(which(mci[,5]>0)),max(which(mci[,6]>0)))
for (i in 1:5)
{
  ind<-last[i]
  for(k in ind:nrow(mci))
  {
    mci[k,i+1]<-mci[last[i],i+1]
  }
}
for(i in 1:nrow(mci))
{
  for(k in 2:6)
  {
    if(mci[i,k]==0)
    {
       for (j in 1:25 )
        {
           if(mci[i+j,k]>0)
            {
              mci[i,k]<-mci[i+j,k]
              break
            }
        }
     }
```

```
    }
  }
  return(mci)
}
```

A.1.2 Overall survival

```
#-------------------------------------------------------------
#Repeat for each data set the following steps:

#DATA PREPARATION:
# Timetoevent: variable representing time
# to event from time_CR
data1$Timetoevent<-data1$time_event-data1$time_CR

#Compute Overall Survival
os1 <-survfit(Surv(Timetoevent,event)~1, data=data1)

#Input data for function joint
f1<-os1$surv
f1t<-os1$time
f1sd<-os1$std

#(time,OS)
a1<-cbind(f1t,f1)

#(time,VarOS)
a2<-cbind(f1t,f1sd^2)
#-------------------------------------------------------------------
resOS<-joint(a1[-1,],b1[-1,],c1[-1,],d1[-1,],e1[-1,])
resOSVar<-joint(a2[-1,],b2[-1,],c2[-1,],d2[-1,],e2[-1,])

OSm<-apply(resOS[,-1],1,mean) #mean Overall Survival
OSwiv<-apply(resOSVar[,-1],1,mean) # within imputation variance
OSbiv<-apply(resOS[,-1],1, var) # between imputation variance

OSVar<-(1+1/5)*OSbiv+OSwiv #total variance overall survival
df<-4*(1*((1+1/5)*OSbiv/OSwiv)^-1)^2 #degree of freedom t-student
quant<-qt(0.975,df) # quantile t-student
```

A.1.3 Cumulative incidence of relapse

```
#-------------------------------------------------------------
#Repeat for each data set the following steps:

#DATA PREPARATION:
# Timetoevent: variable representing time
# to event from time_CR
data1$Timetoevent<-data1$time_event-data1$time_CR
# create colums with time to event and status for comp risks analysis
data1$Event <- 0
# code for event relapse: 1
index <- which(data$relapse==1)
data1$Event[index] <- 1
# code for event death: 2
index <- which(data$relapse==0 & data$death==1)
data1$Event[index] <- 2
#Cumulative incidence for the five complete dataset
ci1 <- Cuminc( "Timetoevent", "Event", data = data1)

#Input data for function joint
f1<-ci1$CI.1
f1t<-ci1$time
f1sd<-ci1$seCI.1

#(time,CI)
a1<-cbind(f1t,f1)

#(time,VarCI)
a2<-cbind(f1t,f1sd^2)
#-----------------------------------------------------------------
resCI<-joint(a1[-1,],b1[-1,],c1[-1,],d1[-1,],e1[-1,])
resCIVar<-joint(a2[-1,],b2[-1,],c2[-1,],d2[-1,],e2[-1,])

mCi<-apply(resCI[,-1],1,mean) #mean of the cumulative incidence
wivCI<-apply(resCIVar[,-1],1,mean) # within imputation variance
bivCI<-apply(resCI[,-1],1, var) # between imputation variance

CIVar<-(1+1/5)*bivCI+wivCI #total variance cumulative incidence
```

A.2 Parametric Approach

A.2.1 Overall Survival

Exponential distribution on both intervals

```
#Exponential-Exponetial negative overall survival log-likelihood
nlogLExp<-function(lambda,data)
{
  n<-nrow(data) #number of patients
  d<-sum(data$event) #number of events
  indcrm<-which(is.na(data$time_CR)) #indices patients time_CR missing
  indcr<-which(!is.na(data$time_CR)) #indices patients time_CR known
  ci<-sum(is.na(data$time_CR)) # number patients with time_CR missing
  #negative log-likelihood if the parameters are equal the log likelihood
  #is given by
  if (lambda[1]==lambda[2])
  {
    return( -((n+d)*log(lambda[1])-lambda[1]*sum(data$time_event)))
  }
  else
    return(
      -(n*log(lambda[1])+d*log(lambda[2])-lambda[2]*sum(data$time_event)+
      +(lambda[2]-lambda[1])* sum(data$time_CR[indcr])+ sum(log((exp((lambda[2]-
      +lambda[1])*data$time_event[indcrm])-1)/(lambda[2]-lambda[1])))))
}

ParEEmle<-nlminb(c(0.6,0.7),nlogLExp,lower=rep(10^-8,2),
 + upper=rep(10,2),data=data)$par #mle
#hessian
hes<-solve(hessian(nlogLExp, ParEEmle, data = data))
#Overall survival plot
plot(function(x) exp(-ParEEmle[2]*x), xlim=c(0,20),
 +ylab="Overall survival", xlab="Years since complete remission")
plot(function(x) exp(-ParEEmle[2]*x)+1.96*
 +sqrt(x^2*exp(-2*ParEEmle[2]*x)*hes[2,2]),0,20,add=T,lty=2)
plot(function(x) exp(-ParEEmle[2]*x)-1.96*
 +sqrt(x^2*exp(-2*ParEEmle[2]*x)*hes[2,2]),0,20,add=T,lty=2)
```

Exponential distribution on $[t_0, t_{CR}]$ and Weibull distribution on $[t_{cr}, t_{event}]$

```
#Exponential-Weibull negative overall survival log-likelihood
nlogLEW<-function(par,data)
{
  indcrm<-which(is.na(data$time_CR)) #indices patients with time_CR missing
  indcr<-which(!is.na(data$time_CR)) #indices patients with time_CR known
  #likelihood for a patient
  # x -> time_CR
  lik<-function(x,data)
  {
    par[1]*exp(-par[1]*x)*(par[2]*par[3]*
   +(data$time_event-x)^(par[2]-1))^data$event*
      +exp(-par[3]*(data$time_event-x)^par[2])
  }

 #computation of the integral for each patients with time_CR missing
  etcrm<-data$time_event[indcrm] #event time patients time_CR missing
  ncrm<-length(etcrm) #number patients time_CR miss
  datacrm<-data[indcrm,] #data set patients with time_CR missing
  int1=NULL
  for(i in 1:length(ncrm))
  {
       int1<-c(int1,integrate(lik,0,etcrm[i],data=datacrm[i,])$value)
  }
  #log-likelihood complete data(time_cr missing+ time_CR known)
  nlogL<-sum(log(int1))+sum(log(lik(data$time_CR[indcr],data[indcr,])))
  return(-nlogL)
}
ParEWmle<-nlminb(c(0.01,0.9,0.3),nlogLEW,lower=rep(10^-8,3),
+upper=rep(10,3),data=data)$par
hes<-solve(hessian(nlogLEW, ParEWmle, data = data))

#Function to compute overall survival variance
stdS<-function(x)
{
  varOS<-sqrt(c(-ParEWmle[3]*x^ParEWmle[2]*
   +log(x)*exp(-ParEWmle[3]*x^ParEWmle[2]),
   +-x^ParEWmle[2]*exp(-ParEWmle[3]*x^ParEWmle[2]))%*%
   +hes[-1,-1]%*%c(-ParEWmle[3]*x^ParEWmle[2]*log(x)*
```

```
    +exp(-ParEWmle[3]*x^ParEWmle[2]),-x^ParEWmle[2]*
    +exp(-ParEWmle[3]*x^ParEWmle[2])))
  return(varOS)
}
stdSv<-Vectorize(stdS,"x")

#Overall survival and confidence interval plot
plot(function(x) exp(-ParEWmle[3]*x^ParEWmle[2]),
+ xlim=c(0,20),ylab="Overall survival",
+ xlab="Years since complete remission")
plot(function(x) exp(-ParEWmle[3]*
  +x^ParEWmle[2])+1.96*stdSv(x),0,20,add=T,lty=2)
plot(function(x) exp(-ParEWmle[3]*
  +x^ParEWmle[2])-1.96*stdSv(x),0,20,add=T,lty=2)
```

Weibull distributions on both intervals

```
#Weibull-Weibull negative overall survival log-likelihood
nlogLWeib<-function(par,data)
{
   #indices patients with time_CR missing
  indcrm<-which(is.na(data$time_CR))
  #indices patients with time_CR known
  indcr<-which(!is.na(data$time_CR))
  #likelihood for a patient
  # x -> time_CR
  lik<-function(x,data)
  {
    par[1]*par[2]*x^(par[1]-1)*exp(-par[2]*x^par[1])*
      +(par[3]*par[4]*(data$time_event-x)^(par[3]-1))^data$event*
      +exp(-par[4]*(data$time_event-x)^par[3])
  }
  #computation of the integral for each patients with time_CR missing
  #time event patients complete remission missing
  etcrm<-data$time_event[indcrm]
  ncrm<-length(etcrm) #number patients time_CR miss
  datcrm<-data[indcrm,] #data time_CR missing
  int1=NULL
```

```
  for(i in 1:ncrm)
  {
     int1<-c(int1,integrate(lik,0,etcrm[i],data=datcrm[i,])$value)
  }
  #log-likelihood complete data(time_cr missing+ time_CR known)
  nlogL<-sum(log(int1))+sum(log(lik(data$time_CR[indcr],data[indcr,])))
  return(-nlogL)
}

ParWWmle<-nlminb(c(0.2,0.5,0.4,0.5),nlogLWeib,lower=rep(10^-8,4),
  + upper=rep(100,4),data=data)$par
hes<-solve(hessian(nlogLWeib, ParWWmle, data = data))

#Function to compute variance overall survival
stdS<-function(x)
{
  varOS<-sqrt(c(-ParWWmle[4]*x^ParWWmle[3]*log(x)*
    +exp(-ParWWmle[4]*x^ParWWmle[3]),-x^ParWWmle[3]*
    +exp(-ParWWmle[4]*x^ParWWmle[3]))%*%hes[-c(1,2),-c(1,2)]%*%
     +c(-ParWWmle[4]*x^ParWWmle[3]*log(x)*exp(-ParWWmle[4]*
     +x^ParWWmle[3]),-x^ParWWmle[3]*
     +exp(-ParWWmle[4]*x^ParWWmle[3])))
  return(varOS)
}
stdSv<-Vectorize(stdS,"x")
#Overall survival and confidence interval plot
plot(function(x) exp(-ParWWmle[4]*x^ParWWmle[3]),
 + xlim=c(0,20),ylab="Overall survival", xlab="Years since complete remission
plot(function(x) exp(-ParWWmle[4]*x^ParWWmle[3])+
 +1.96*stdSv(x),0,20,add=T,lty=2)
plot(function(x) exp(-ParWWmle[4]*x^ParWWmle[3])-
 +1.96*stdSv(x),0,20,add=T,lty=2)
```

A.2.2 Cumulative incidence of relapse

Exponential distribution for the random variables T, U_2, U_3

```
#Exponential Exponential Competing risk negative log-likelihood
compRiskLikEE<-function(par,data)
```

```
{
  n<-nrow(data)
  indr<-which(data$relapse==1) #ind. relapse
  nr<-length(indr) #number patients relapse
  indd<-which(data$relapse==0 & data$death==1) #ind. death
  nd<-length(indd) #number patients death before relapse
  indcr<-which(!is.na(data$time_CR)) #ind. patients time_CR known
  indcrm<-which(is.na(data$time_CR)) #ind. patients time_CR known
  sum(nd+nr)
  #negative log-likelihood
 -(n*log(par[1])+nr*log(par[2])+nd*log(par[3])-(par[2]+par[3])*
   +sum(data$time_event)+(par[3]+par[2]-par[1])*sum(data$time_CR[indcr])+
   +sum(log( (exp((par[3]+par[2]-par[1])*data$time_event[-indcr])-1) /
   +(par[3]+par[2]-par[1]))) )
  }

CompRiskmleEE<-nlminb(c(3,0.1,0.1),compRiskLikEE,lower=rep(10^-8,3),
  + upper=rep(15,3),data=data)$par
hes<-solve(hessian(compRiskLikEE, CompRiskmleEE, data = data))
#Variance Cumulative incidence of relapse
stdS<-function(x)
{
 der<-c(CompRiskmleEE[3]/(CompRiskmleEE[2]+CompRiskmleEE[3])^2*
      +(1-exp(-(CompRiskmleEE[2]+CompRiskmleEE[3])*x))
      +CompRiskmleEE[2]/(CompRiskmleEE[2]+CompRiskmleEE[3])*
      +(x*exp(-(CompRiskmleEE[2]+CompRiskmleEE[3])*x)),
      +-CompRiskmleEE[2]*(1-exp(-(CompRiskmleEE[2]+CompRiskmleEE[3])*
      +x))/(CompRiskmleEE[2]+CompRiskmleEE[3])^2)

  varCI<-sqrt(der%*%hes[-1,-1]%*% der)
  return(varCI)
}
#Cumulative incidence of relapse function
cir<-function(x)
{
  CompRiskmleEE[2]/(CompRiskmleEE[2]+CompRiskmleEE[3])*
  +(1-exp(-(CompRiskmleEE[2]+CompRiskmleEE[3])*x))
}
stdSv<-Vectorize(stdS,"x")
cirv<-Vectorize(cir,"x")
#Plot cumulative incidence of relapse
```

```
plot(function(x) cirv(x), xlim=c(0.1,6),
  +ylim=c(0,0.35),ylab="Cumulative incidence of relapse", xlab="Time")
plot(function(x) cirv(x)+1.96*stdSv(x),0,6,add=T,lty=2)
plot(function(x) cirv(x)-1.96*stdSv(x),0,6,add=T,lty=2)
```

Exponential distribution for the random variable$T$ and Weibull distribution for the random variables U_2, U_3

```
#Weibull-Weibull Competing risk negative log likelihood
compRiskLikWW<-function(par,data)
{
  indr<-which(data$relapse==1) #ind. relapse
  nr<-length(indr) #number patients relapse
  indd<-which(data$relapse==0 & data$death==1) #ind. death
  nd<-length(indd) #number patients death before relapse
  indcr<-which(!is.na(data$time_CR)) #ind. patients time_CR known
  #likelihood for a patient
  lik<-function(x,data)
  {
    par[1]*(par[2]*par[3]*(data$time_event-x)^(par[2]-1))^data$relapse*
    + (par[4]*par[5]*(data$time_event-x)^(par[4]-1))^
    +(data$relapse==0 & data$death==1)*exp(-par[1]*x-par[3]*
    +(data$time_event-x)^par[2]-par[5]*(data$time_event-x)^par[4])
  }
  #computation of the integral for each patients with time_CR missing
  #time event patients complete remission missing
  etcrm<-data$time_event[-indcr]
  ncrm<-length(etcrm) #number patients time_CR miss
  datcrm<-data[-indcr,] #data time_CR missing
  int1=NULL
  for(i in 1:ncrm)
  {
     int1<-c(int1,integrate(lik,0,etcrm[i],data=datcrm[i,])$value)
  }
  #log-likelihood complete data(time_cr missing+ time_CR known)
  nlogL<-sum(log(int1))+sum(log(lik(data$time_CR[indcr],data[indcr,])))
  return(-nlogL)
}
CompRiskmleWW<-nlminb(c(3,0.5,0.6,0.2,0.5),compRiskLikWW,lower=rep(10^-8,5),
  + upper=rep(10,5),data=data)$par
```

```
hes<-solve(hessian(compRiskLikWW,CompRiskmleWW,data=data))
CompRiskStdWW<-c(hes[1,1],hes[2,2],hes[3,3],hes[4,4],hes[5,5])
CIRWW<-CompRiskmleWW[-1]

#Variance cumulative incidence of relpase
stdW<-function(x)
{
#First derivative of cumulative incidence with
#respect to alpha2
fa2<-function(t)
{
  CIRWW[2]*t^(CIRWW[1]-1)*exp(-CIRWW[2]*
   +t^(CIRWW[1]) -CIRWW[4]*t^(CIRWW[3]))*
   +(1+CIRWW[1]*log(t)-CIRWW[1]*CIRWW[2]*
   +t^(CIRWW[1])*log(t))
}
#First derivative of cumulative incidence with
#respect to lambda2
fl2<-function(t)
{
  CIRWW[1]*t^(CIRWW[1]-1)*exp(-CIRWW[2]*
     +t^(CIRWW[1]) -CIRWW[4]*t^(CIRWW[3]))*
    +(1-CIRWW[2]^2*t^(CIRWW[1])*log(t))
}
#First derivative of cumulative incidence with
#respect to alpha3
fa3<-function(t)
{
  -CIRWW[2]*CIRWW[1]*CIRWW[4]*t^(CIRWW[1]+
  +CIRWW[3]-1)* log(t)*+exp(-CIRWW[2]*t^(CIRWW[1])-
   +CIRWW[4]*t^(CIRWW[3]))
}
#First derivative of cumulative incidence with
#respect to lambda3
fl3<-function(t)
{
  -CIRWW[2]*CIRWW[1]*t^(CIRWW[1]+CIRWW[3]-1)*
    +exp(-CIRWW[2]*t^(CIRWW[1]) -CIRWW[4]*t^(CIRWW[3]))
}
der<-c(integrate(fa2,0,x,subdivisions=1e7)$value,
 +integrate(fl2,0,x)$value,integrate(fa3,0,x)$value,
```

```
  +integrate(fl3,0,x)$value)
varCI<-sqrt(der%*%hes[-1,-1]%*%der)
return(varCI)
}

#Cumulative incidence function
cirW<-function(x)
{
  ff<-function(t)
  {
    CIRWW[2]*CIRWW[1]*t^(CIRWW[1]-1)*
      +exp(-CIRWW[2]*t^(CIRWW[1]) -CIRWW[4]*t^(CIRWW[3]))
  }
  return(integrate(ff,0,x)$value)
}
stdWv<-Vectorize(stdW,"x")
cirWv<-Vectorize(cirW,"x")
#Plot cumulative incidence function and conf interval
plot(function(x) cirWv(x), 10^-2 ,6,ylim=c(0,0.35),
 +ylab="Cumulative incidence of relapse", xlab="Time")
plot(function(x) cirWv(x)+1.96*stdWv(x),10^-2,6,add=T,lty=2)
plot(function(x) cirWv(x)-1.96*stdWv(x),10^-2,6,add=T,lty=2)
```

A.3 EM-algorithm

A.3.1 Overall Survival

Exponential distribution on both intervals

```
#Exponential-Exponential negative Q-function
QfunEE<-function(lambda,lambdak,ustar,tstar,data)
{
  indcrm<-which(is.na(data$time_CR))#indices patients with time_CR missing
  indcr<-which(!(is.na(data$time_CR))) #indices patients with time_CR known
  n=nrow(data) #number of patients
  d=sum(which(data$event==1)) # number of events
  d0<-sum(which(data$event==0)) # number of censored
  #control: if the parameters are equals
  #the likelihood is given by:
  if(lambdak[1]==lambdak[2] || lambda[1]==lambda[2])
  {
```

```
    return(-(2*n*log(lambda[1])-lambda[1]*
    +sum(data$time_event)-lambda[1]*d0/(2*lambdak[1])))
  }
  else{
   # number of censored observation with time_CR known
   d1<- sum((data$event==0 & !(is.na(data$time_CR)) ))
  #Q-function: complete likelihood with missing values
  # replaced by their conditional expectation
  return(-(n*log(lambda[1]*lambda[2])-lambda[2]*
  +(sum(data$time_event[indcr]-data$time_CR[indcr])+d1/lambdak[2]
  +sum(ustar))-lambda[1]*(sum(data$time_CR[indcr])+sum(tstar))))}
}

#Computation oftstar= E(T/V>v)=E(T/V=v)
#and ustar= E(U/V>v)=E(U/V=v)
elementEE<-function(data,lambdak)
{
  indcrm<-which(is.na(data$time_CR))#indices patients with time_CR missing
  n=nrow(data) #number of patients
  #Conditional expectation of u_i given the data ustar=E(U/V>v)=E(U/V=v)
  ustar<-(data$time_event[indcrm]*exp((lambdak[1]-lambdak[2])*
   +data$time_event[indcrm])+(lambdak[1]-lambdak[2])^(-1)*
    +(1-exp((lambdak[1]-lambdak[2])*data$time_event[indcrm])))/
    +( exp((lambdak[1]-lambdak[2])*data$time_event[indcrm]) -1)
  #Conditional expectation of t_i given the data tstar=E(T/V>v)=E(T/V=v)
  tstar<- (data$time_event[indcrm]*exp((lambdak[2]-lambdak[1])*
    +data$time_event[indcrm])+(lambdak[2]-lambdak[1])^(-1)*
    +(1-exp((lambdak[2]-lambdak[1])*data$time_event[indcrm])))/
    +( exp((lambdak[2]-lambdak[1])*data$time_event[indcrm]) -1)
  return(list(ustar=ustar,tstar=tstar))

}
#EM algorithm: function that takes in input the initial
# values for the parameters, the precision at which the
#estimation should be done, and the data.
#Return MLE and number of iterations
EMExpExp<-function(lambdak,precision,data)
{
  n<-nrow(data)
  difference<-1
  iter<-0
```

```
  # number of censored observation with time_CR known
  d1<- sum((data$event==0 & !(is.na(data$time_CR)) ))
  #control: the function exit from while if the precision is obtained
  #or the maximum number of iteration are reached
  while(difference>precision & iter<10000)
  {
    iter<-iter+1
    el<-elementEE(data,lambdak)
    tstar<-el$tstar
    ustar<-el$utstar
    indcr<-which(!(is.na(data$time_CR))) #indices patients with time_CR known
    #MLE at generic step k+1
    mle<-c(n/sum(sum(tstar)+sum(data$time_CR[indcr])),
             +n/(sum(data$time_event[indcr]-data$time_CR[indcr])+
             +d1/lambdak[2]+sum(ustar)))
    #compute  Q(lamda^k+1)-Q(lambda^k)
    difference<-abs(abs(QfunEE(lambdak,lambdak,ustar,tstar,data))-
                      +abs(QfunEE(mle,lambdak,ustar,tstar,data)))
    #at each step is chosen the value for which the Q function is greater
    if(QfunEE(mle,lambdak,ustar,tstar,data)<
       +QfunEE(lambdak,lambdak,ustar,tstar,data))
    {

      #if the difference between mle(lambda(k+1))
      # and lambdak is lower than 10^-5 the function ends
      if ((abs(mle[1]-lambdak[1]))<10^-5 & (abs(mle[2]-lambda[2]))<10^-5)
      {return(list(lambdak=lambdak,iter=iter,
           +difference=difference,ustar=ustar,tstar=tstar)))}
      lambdak<-mle
     }
  }
  return(list(mle=lambdak,iter=iter,difference=difference,ustar=ustar,tstar=t
}
EMEEres<-EMExpExp(c(0.3,0.5),10^-8,data)
l1<-EMEEres$mle[1] #lambda1
l2<-EMEEres$mle[2] #lambda2

#E(T^2|V=v)=E(T^2|V>v)
et2<-(data$time_event[indcrm]^2*exp((l2-l1)*data$time_event[indcrm])-2*
 +data$time_event[indcrm]*exp((l2-l1)*data$time_event[indcrm])/(l2-l1)^2-
 +2/(l2-l1)^2)/(exp((l2-l1)*data$time_event[indcrm])-1)
```

```
#E(U^2|V=v)=E(U^2|V>v)
eu2<-(data$time_event[indcrm]^2*exp((l1-l2)*data$time_event[indcrm])-2*
 +data$time_event[indcrm]*exp((l1-l2)*data$time_event[indcrm])/(l1-l2)^2-
 +2/(l1-l2)^2)/(exp((l1-l2)*data$time_event[indcrm])-1)

#Variance lambda1, lambda2
EMEEvar<-c(1/(nrow(data)/(l1)^2-sum(et2-(EMEEres$tstar)^2)),
 +1/(nrow(data)/l2^2- length(indcrem)/l2^2-sum(eu2-(EMEEres$ustar)^2)))
EMEEStd<-sqrt(EMEEvar)
#Overall survival plot
plot(function(x) exp(-EMEEres$mle[2]*x), xlim=c(0,20),
 +ylab="Overall survival", xlab="Years since complete remission")
plot(function(x) exp(-EMEEres$mle[2]*x)+1.96*sqrt(x^2*exp(-2*
 +EMEEres$mle[2]*x)*EMEEvar[2]),0,20,add=T,lty=2)
plot(function(x) exp(-EMEEres$mle[2]*x)-1.96*sqrt(x^2*exp(-2*
 +EMEEres$mle[2]*x)*EMEEvar[2]),0,20,add=T,lty=2)
```

Exponential distribution on $[t_0, t_{CR}]$ and Weibull distribution on $[t_{cr}, t_{event}]$

```
#Exponential-Weibull negative Q-function
# in input in addition to the current estimate of the
#parameter(thetak) are given the expected values
#computed separately in order to accelerate the algorithm
QfunEW<-function(theta,thetak,data,
 +u1log=u1log,u2log=u2log,u3log=u3log,t1star,t2star)
{
  n=nrow(data) #number of patients
  #indices patients with time_CR known
  indcr<-which(!(is.na(data$time_CR)))
  #indices obs with time_CR known and event
  indcre<-which(!(is.na(data$time_CR)) & data$event==1)
  #indices censored obs with time_CR known
  indcrem<-which(!(is.na(data$time_CR)) & data$event==0)
  # indices patients with time_CR missing and event
  indcrme<-which(is.na(data$time_CR) & data$event==1)
  #indices censored obs with time_CR missing
  indcrmem<-which(is.na(data$time_CR) & data$event==0)
```

```
#function to compute conditional expectation ue*1=E(U^alpha2/U>v-t)
fu1exp<-function(x,data)
{
  thetak[2]*thetak[3]*x^(theta[2]+thetak[2]-1)*exp(-thetak[3]*
    +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
}

#Loop for all censored observation with time_CR known
ncrem<-length(indcrem) #number of censored obs. with time_CR known
u1exp<-NULL
i<-1
for(i in 1:ncrem)
{
  u1exp<-c(u1exp,integrate(fu1exp,data$time_event[indcrem[i]]-
  +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)

}
#-------------------------------------------------------------
#function to compute conditional expectation ue*2=E(U^alpha2/V=v)
funtuexp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  thetak[1]*thetak[2]*thetak[3]*x^(theta[2]*thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}

#Loop for patients with time_CR missing and event=1
ncrme<-length(indcrme) #number patients time_CR miss and event=1
u2exp<-NULL

for(i in 1:ncrme)
{
  u2exp<-c(u2exp,integrate(funtuexp,0,data$time_event[indcrme[i]],
   +data=data[indcrme[i],])$value)
 }
```

```
  #-------------------------------------------------------------

  #function to compute conditional expectation ue*3=E(U^alpha2/V>v)
  survtuexp<-function(x,data)
  {
    f1<-function(z,data=data)
    {
      thetak[3]*thetak[2]*z^(thetak[2]-1)*
        +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
    }

    thetak[3]*thetak[2]*x^(theta[2]+thetak[2]-1)*
      +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
      +integrate(f1,0,data$time_event,data=data)$value
  }
  #Loop for censored obs. with time_CR missing
  ncrmem<-length(indcrmem)
  u3exp<-NULL
  for(i in 1:ncrmem)
  {
    u3exp<-c(u3exp,integrate(survtuexp,0,data$time_event[indcrmem[i]],
   +data=data[indcrmem[i],])$value)
   }

  #Q-function: negative complete likelihood with missing values
  # replaced by their conditional expectation
  return(-( n*log(theta[1]*theta[2]*theta[3])-
    +theta[1]*sum((sum(data$time_CR[indcr])+sum(t1star)+
    +sum(t2star)))+(theta[2]-1)*sum(sum(log
    +(data$time_event[indcre]-data$time_CR[indcre]))
    +sum(u1log)+sum(u2log)+sum(u3log))-theta[3]*
     +sum(sum((data$time_event[indcre]-
    +data$time_CR[indcre])^theta[2]) +sum(u1exp)+
    +sum(u2exp)+sum(u3exp)) ))
}


#Function to compute for a given thetak the estimates of the expected value
#for every missing values
EWelement<-function(data,thetak)
{
```

```
#indices patients with time_CR known
indcr<-which(!(is.na(data$time_CR)))
#indices obs with time_CR known and event
indcre<-which(!(is.na(data$time_CR)) & data$event==1)
#indices censored obs with time_CR known
indcrem<-which(!(is.na(data$time_CR)) & data$event==0)
#indices patients with time_CR missing and event
indcrme<-which(is.na(data$time_CR) & data$event==1)
#indices censored obs with time_CR missing
indcrmem<-which(is.na(data$time_CR) & data$event==0)

#function to compute conditional expectation ul*1=E(log(U)/U>v-t)
fu1log<-function(x,data)
{
  log(x)*thetak[2]*thetak[3]*x^(thetak[2]-1)*exp(-thetak[3]*
    +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
}
 #Loop for all censored observation with time_CR known
ncrem<-length(indcrem) #number of censored obs. with time_CR known
u1log<-NULL
i<-1
for(i in 1:ncrem)
{
  u1log<-c(u1log,integrate(fu1log,data$time_event[indcrem[i]]-
 +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
 }
#-----------------------------------------------------------------------
#function to compute conditional expectation t*1=E(T/V=v)
funtv<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*(data$time_event-z)^(thetak[2]-1)*
      +exp(-thetak[1]*z-thetak[3]*(data$time_event-z)^thetak[2])
  }
  thetak[1]*thetak[2]*thetak[3]*x*(data$time_event-x)^(thetak[2]-1)*
    +exp(-thetak[1]*x-thetak[3]*(data$time_event-x)^thetak[2])/
  +integrate(f1,0,data$time_event,data=data)$value
}

#function to compute conditional expectation ul*1=E(log(U)/V=v)
```

```
funtulog<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  log(x)*thetak[1]*thetak[2]*thetak[3]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
  +integrate(f1,0,data$time_event,data=data)$value
}

#Loop for patients with time_CR missing and event=1
ncrme<-length(indcrme) #number patients time_CR miss and event=1
t1star<-NULL
u2log<-NULL

for(i in 1:ncrme)
{
  t1star<-c(t1star,integrate(funtv,0,data$time_event[indcrme[i]],
  +data=data[indcrme[i],])$value)
  u2log<-c(u2log,integrate(funtulog,0,data$time_event[indcrme[i]],
  +data=data[indcrme[i],])$value)
 }
#-------------------------------------------------------------------
#function to compute conditional expectation t*2=E(T/V>v)
survtv<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*exp(-thetak[1]*z-thetak[3]*(data$time_event-z)^thetak[2])
  }
  thetak[1]*x*exp(-thetak[1]*x-thetak[3]*(data$time_event-x)^thetak[2])/
  +integrate(f1,0,data$time_event,data=data)$value
}

#function to compute conditional expectation ul*3=E(log(U)/V>v)
survtulog<-function(x,data)
{
  f1<-function(z,data=data)
  {
```

```
      thetak[3]*thetak[2]*z^(thetak[2]-1)*
        +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
    }
    log(x)*thetak[3]*thetak[2]*x^(thetak[2]-1)*
      +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
      +integrate(f1,0,data$time_event,data=data)$value
  }

  #Loop for all censored observation with time_CR missing
  ncrmem<-length(indcrmem)
  t2star<-NULL
  u3log<-NULL
  for(i in 1:ncrmem)
  {
    t2star<-c(t2star,integrate(survtv,0,data$time_event[indcrmem[i]],
     +data=data[indcrmem[i],])$value)
    u3log<-c(u3log,integrate(survtulog,0,data$time_event[indcrmem[i]],
     +data=data[indcrmem[i],])$value)

    }

  return(list(u1log=u1log,u2log=u2log,u3log=u3log,
   +t1star=t1star,t2star=t2star))
}

#EM algorithm: function that takes in input the initial values
# for the parameters, the precision at which the estimation
# should be done, and the data.employing the EM algorithm
#return the MLE and the number of iterations
EMExpWeib<-function(thetak,precision,data)
{
  difference<-1
  iter<-0
  n<-nrow(dat)
  indcr<-which(!(is.na(data$time_CR))) #indices patients with time_CR known
  #control: the function exit from while if the precision is obtained or
  #the maximum number of iteration are reached or the difference
  #between the estimated mle and the current thetak is lower then 10^-4
  while(difference>precision & iter<1000)
  {
    iter<-iter+1
```

```
    el<-EWelement(data,thetak) #computation of the expected values
    u1log<-el$u1log
    u2log<-el$u2log
    u3log<-el$u3log
    t1star<-el$t1star
    t2star<-el$t2star
    #MLE at generic step k+1
    mle<-nlminb(c(lambda1,0.6,0.1),QfunEW,lower=rep(10^-15,3),
    upper=rep(50,3),data=data,thetak=thetak,u1log=u1log,u2log=u2log
     +,u3log=u3log,t1star=t1star,t2star=t2star)$par

    #compute Q(theta^k+1)-Q(theta^k)
    difference<-abs(abs(  QfunEW(theta=thetak,data=data,
     +thetak=thetak,u1log=u1log,u2log=u2log,u3log=u3log,
     +t1star=t1star,t2star=t2star))-abs(QfunEW(theta=mle,
     +data=data,thetak=thetak,u1log=u1log,u2log=
     +u2log,u3log=u3log,t1star=t1star,t2star=t2star)))

    #at each step is chosen the value for which the Q function is greater
    if(QfunEW(theta=mle,data=data,thetak=thetak,u1log=u1log,u2log=u2log,
     +u3log=u3log,t1star=t1star,t2star=t2star)<QfunEW(theta=thetak,data=data,
     +thetak=thetak,u1log=u1log,u2log=u2log,u3log=u3log,
     +t1star=t1star,t2star=t2star))
    {
      #if the difference between mle(theta(k+1))
      # and thetak is lower than 10^-4 the function ends
      if ((abs(mle[1]-thetak[1]))<10^-4 & (abs(mle[2]-thetak[2]))<10^-4
         +& (abs(mle[3]-thetak[3]))<10^-4 )
      {return(list(mle=thetak,iter=iter,diff=difference))}
      thetak<-mle

     }
  }
  return(list(mle=thetak,iter=iter,diff=difference))
}

EMEWres<-EMExpWeib(c(7,0.8,0.1),10^-5,data)


#Function to compute variance MLE
EMEWVarfun<-function(thetak,data)
```

```
{
 #indices patients with time_CR known
  indcr<-which(!(is.na(data$time_CR)))
  #indices obs with time_CR known and event
  indcre<-which(!(is.na(data$time_CR)) & data$event==1)
  #indices censored obs with time_CR known
  indcrem<-which(!(is.na(data$time_CR)) & data$event==0)
  #indices patients with time_CR missing and event
  indcrme<-which(is.na(data$time_CR) & data$event==1)
  #indices censored obs with time_CR missing
  indcrmem<-which(is.na(data$time_CR) & data$event==0)
    ##############################
  #VARIANCE LAMBDA2

 #-------------------------------------------------------------------
 # E(U^alpha2/U>v-u)
  fu1exp<-function(x,data)
  {
    thetak[2]*thetak[3]*x^(2*thetak[2]-1)*exp(-thetak[3]*
      +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
  }
  # E((U^alpha2)^2/U>v-u)
  fu12exp<-function(x,data)
  {
    thetak[2]*thetak[3]*x^(3*thetak[2]-1)*exp(-thetak[3]*
      +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
  }
  #cicle for all censored observation with time_CR known
  ncrem<-length(indcrem) #number of censored obs. with time_CR known
  u1exp<-NULL
  u12exp<-NULL
  i<-1
  for(i in 1:ncrem)
  {
    u1exp<-c(u1exp,integrate(fu1exp,data$time_event[indcrem[i]]-
    +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
    u12exp<-c(u12exp,integrate(fu12exp,data$time_event[indcrem[i]]-
     +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)

  }
  #Var(U^alpha2/U>v-u)
```

```
var1uexp<-sum(u12exp-(u1exp)^2)

#-----------------------------------------------------------------------
#E(U^alpha2/V=v)
funtuexp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  thetak[1]*thetak[2]*thetak[3]*x^(2*thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((U^alpha2)^2/V=v)
funtu2exp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  thetak[1]*thetak[2]*thetak[3]*x^(3*thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}

ncrme<-length(indcrme) #number patients time_CR miss and event
u2exp<-NULL
u22exp<-NULL
for(i in 1:ncrme)
{
  u2exp<-c(u2exp,integrate(funtuexp,0,data$time_event[indcrme[i]],
  +data=data[indcrme[i],])$value)
  u22exp<-c(u22exp,integrate(funtu2exp,0,data$time_event[indcrme[i]],
  +data=data[indcrme[i],])$value)

}
#Var(U^alpha2/V=v)=E((U^alpha2)^2/V=v)-(#E(U^alpha2/V=v))^2
var2uexp<-sum(u22exp-(u2exp)^2)
```

```
#-------------------------------------------------------------
#E(U^alpha2/V>v)
survtuexp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  thetak[3]*thetak[2]*x^(2*thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E(U^alpha2^2/V>v)
survtu2exp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  thetak[3]*thetak[2]*x^(3*thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
ncrmem<-length(indcrmem)
u3exp<-NULL
u32exp<-NULL
for(i in 1:ncrmem)
{
  u3exp<-c(u3exp,integrate(survtuexp,0,data$time_event[indcrmem[i]],
   +data=data[indcrmem[i],])$value)
  u32exp<-c(u32exp,integrate(survtu2exp,0,data$time_event[indcrmem[i]],
   +data=data[indcrmem[i],])$value)

}
#Var(U^alpha2/V>v)=E((U^alpha2)^2/V>v)-(#E(U^alpha2/V>v))^2
var3uexp<-sum(u32exp-(u3exp)^2)
```

```
  InfLambda1 lambda2
#I(lambda2)= n/lambda2^2-Var(U^alpha2/V>v)
 InfLambda2<-n/thetak[3]^2-(var2uexp+var3uexp+var1uexp)

 ####################################
 #VARIANCE ALPHA2

 #-------------------------------------------------------------------------
 # E(U^alpha2*log(U)^2/U>v-u)
 fu1exp1<-function(x,data)
 {
   log(x)^2*thetak[2]*thetak[3]*x^(2*thetak[2]-1)*exp(-thetak[3]*
     +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
 }

 # E(U^alpha2*log(U)/U>v-u)
 fu1exp2<-function(x,data)
 {
   log(x)*thetak[2]*thetak[3]*x^(2*thetak[2]-1)*exp(-thetak[3]*
     +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
 }
 # E((U^alpha2*log(U))^2/U>v-u)
 fu12exp2<-function(x,data)
 {
   (log(x)*x^thetak[2])^2*thetak[2]*thetak[3]*x^(thetak[2]-1)*
   +exp(-thetak[3]*(x^thetak[2]-(data$time_event-data$time_CR)^
   +thetak[2]))
 }
 # E(log(U)/U>v-u)
 fu1exp3<-function(x,data)
 {
   log(x)*thetak[2]*thetak[3]*x^(thetak[2]-1)*exp(-thetak[3]*
     +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
 }
 # E((log(U))^2/U>v-u)
 fu12exp3<-function(x,data)
 {
   log(x)^2*thetak[2]*thetak[3]*x^(thetak[2]-1)*exp(-thetak[3]*
     +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
 }
```

```
#cicle for all censored observation with time_CR known
ncrem<-length(indcrem) #number of censored obs. with time_CR known
u1exp1<-NULL
u1exp2<-NULL
u12exp2<-NULL
u1exp3<-NULL
u12exp3<-NULL
i<-1
for(i in 1:ncrem)
{
  u1exp1<-c(u1exp1,integrate(fu1exp1,data$time_event[indcrem[i]]-
    +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
  u1exp2<-c(u1exp2,integrate(fu1exp2,data$time_event[indcrem[i]]-
     +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
  u12exp2<-c(u12exp2,integrate(fu12exp2,data$time_event[indcrem[i]]
   +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
  u1exp3<-c(u1exp3,integrate(fu1exp3,data$time_event[indcrem[i]]
    +-data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
  u12exp3<-c(u12exp3,integrate(fu12exp3,data$time_event[indcrem[i]]-
 +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
}
#lambda2*E(U^alpha2*(logU)^2/U>v-u)-lambda2^2*
#  Var(U^alpha2*log(U)/U>v-u)-Var(log(U)/U>v-u)
var1uexp<-thetak[3]*sum(u1exp1)-thetak[3]^2*
    +sum(u12exp2-(u1exp2)^2)-sum(u12exp3-(u1exp3)^2)
#-----------------------------------------------------------------------
#E(U^alpha2*log(u)^2/V=v)
funtuexp1<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  log(x)^2*thetak[1]*thetak[2]*thetak[3]*x^(2*thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E(U^alpha2*log(u)/V=v)
funtuexp2<-function(x,data)
{
```

```
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  log(x)*thetak[1]*thetak[2]*thetak[3]*x^(2*thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((U^alpha2*log(U))^2/V=v)
funtu2exp2<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  (log(x)*x^thetak[2])^2*thetak[1]*thetak[2]*thetak[3]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}

#E(log(u)/V=v)
funtuexp3<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  log(x)*thetak[1]*thetak[2]*thetak[3]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((log(U))^2/V=v)
funtu2exp3<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
```

```
  }
  log(x)^2*thetak[1]*thetak[2]*thetak[3]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}

ncrme<-length(indcrme) #number patients time_CR miss and event
u2exp1<-NULL
u2exp2<-NULL
u22exp2<-NULL
u2exp3<-NULL
u22exp3<-NULL
for(i in 1:ncrme)
{
  u2exp1<-c(u2exp1,integrate(funtuexp1,0,data$time_event[indcrme[i]],
    +data=data[indcrme[i],])$value)
  u2exp2<-c(u2exp2,integrate(funtuexp2,0,data$time_event[indcrme[i]],
     +data=data[indcrme[i],])$value)
  u22exp2<-c(u22exp2,integrate(funtu2exp2,0,data$time_event[indcrme[i]],
     +data=data[indcrme[i],])$value)
  u2exp3<-c(u2exp3,integrate(funtuexp3,0,data$time_event[indcrme[i]],
     +data=data[indcrme[i],])$value)
  u22exp3<-c(u22exp3,integrate(funtu2exp3,0,data$time_event[indcrme[i]],
     +data=data[indcrme[i],])$value)

}
#n/alpha2^2+lambda2*E(U^alpha2*(logU)^2/U>v-u)-lambda2^2*
#Var(U^alpha2*log(U)/U>v-u)-Var(log(U)/U>v-u)
var2uexp<-thetak[3]*sum(u2exp1)-thetak[3]^2*sum(u22exp2-(
   +u2exp2)^2)-sum(u22exp3-(u2exp3)^2)
#-----------------------------------------------------------------
#E(U^alpha2*logU^2/V>v)
survtuexp1<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  log(x)^2*thetak[3]*thetak[2]*x^(2*thetak[2]-1)*
```

```
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}

#E(U^alpha2*logU/V>v)
survtuexp2<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  log(x)*thetak[3]*thetak[2]*x^(2*thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((U^alpha2*logU)^2/V>v)
survtu2exp2<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  (x^thetak[2]*log(x))^2*thetak[3]*thetak[2]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E(logU/V>v)
survtuexp3<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  log(x)*thetak[3]*thetak[2]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
```

```
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((logU)^2/V>v)
survtu2exp3<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  log(x)^2*thetak[3]*thetak[2]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
ncrmem<-length(indcrmem)
u3exp1<-NULL
u3exp2<-NULL
u32exp2<-NULL
u3exp3<-NULL
u32exp3<-NULL
for(i in 1:ncrmem)
{
  u3exp1<-c(u3exp1,integrate(survtuexp1,0,
   +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
  u3exp2<-c(u3exp2,integrate(survtuexp2,0,
   +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
  u32exp2<-c(u32exp2,integrate(survtu2exp2,0,
   +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
  u3exp3<-c(u3exp3,integrate(survtuexp3,0,
   +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
  u32exp3<-c(u32exp3,integrate(survtu2exp3,0,
   +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)

  }
#lambda2*E(U^alpha2*(logU)^2/U>v-u)-lambda2^2*
#Var(U^alpha2*log(U)/U>v-u)-Var(log(U)/U>v-u)
var3uexp<-thetak[3]*sum(u3exp1)-thetak[3]^2*
 +sum(u32exp2-(u3exp2)^2)-sum(u32exp3-(u3exp3)^2)
```

```
 #Fisher information alpha2
#I(alpha2)=n/alpha2^2+lambda2*(E(U/obs))-
#[lambda2^2*Var(U^alpha2/obs)+Var(log(U)/obs)]
InfAlpha2<-var3uexp+var1uexp+var2uexp+
 + n/thetak[2]^2+thetak[2]*sum((data$time_event[indcre]-
  +data$time_CR[indcre])^thetak[3]*log(data$time_event[indcre]-
  +data$time_CR[indcre]))
####################################
#VARIANCE LAMBDA1

#E(T/V=v)
funtv<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*(data$time_event-z)^(thetak[2]-1)*
      +exp(-thetak[1]*z-thetak[3]*(data$time_event-z)^thetak[2])
  }
  thetak[1]*thetak[2]*thetak[3]*x*(data$time_event-x)^(thetak[2]-1)*
    +exp(-thetak[1]*x-thetak[3]*(data$time_event-x)^thetak[2])/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E(T^2/V=v)
funtv1<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*(data$time_event-z)^(thetak[2]-1)*
      +exp(-thetak[1]*z-thetak[3]*(data$time_event-z)^thetak[2])
  }
  thetak[1]*thetak[2]*thetak[3]*x^2*(data$time_event-x)^(thetak[2]-1)*
  +exp(-thetak[1]*x-thetak[3]*(data$time_event-x)^thetak[2])/
  +integrate(f1,0,data$time_event,data=data)$value
}

ncrme<-length(indcrme) #number patients time_CR miss and event
t1star<-NULL
t1star1<-NULL

for(i in 1:ncrme)
{
```

```
  t1star<-c(t1star,integrate(funtv,0,data$time_event[indcrme[i]]
  +,data=data[indcrme[i],])$value)
  t1star1<-c(t1star1,integrate(funtv1,0,data$time_event[indcrme[i]],
   +data=data[indcrme[i],])$value)
}
#Var(T/V=v)
vart1<-sum((t1star1-(t1star)^2))
#-------------------------------------------------------------------
#E(T/V>v)
survtv<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*exp(-thetak[1]*z-thetak[3]*(data$time_event-z)^thetak[2])
  }
  thetak[1]*x*exp(-thetak[1]*x-thetak[3]*(data$time_event-x)^thetak[2])/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E(T^2/V>v)
survtv1<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*exp(-thetak[1]*z-thetak[3]*(data$time_event-z)^thetak[2])
  }
  thetak[1]*x^2*exp(-thetak[1]*x-thetak[3]*(data$time_event-x)^thetak[2])/
    +integrate(f1,0,data$time_event,data=data)$value
}
ncrmem<-length(indcrmem)
t2star<-NULL
t2star1<-NULL
for(i in 1:ncrmem)
{
  t2star<-c(t2star,integrate(survtv,0,data$time_event[indcrmem[i]],
   +data=data[indcrmem[i],])$value)
  t2star1<-c(t2star1,integrate(survtv1,0,data$time_event[indcrmem[i]],
   +data=data[indcrmem[i],])$value)
}
#Var(T/V>v)
vart2<-sum((t2star1-(t2star)^2))
```

```
  #Fisher information lambda1
  #I(lambda1)=n/lambda1^2-Var(T/obs)
  InfLambda1<-n/thetak[1]^2- vart1-vart2
##############################################
#COVARIANCE(alpha2,lambda2)

# E((U^alpha2+logU)/U>v-u)
fu1expU<-function(x,data)
{
  (log(x)+x^thetak[2])*thetak[2]*thetak[3]*x^(thetak[2]-1)*exp(-thetak[3]*
    +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
}
# E((U^alpha2+log(U))^2/U>v-u)
fu12expU<-function(x,data)
{
  (log(x)+x^thetak[2])^2*thetak[2]*thetak[3]*x^(thetak[2]-1)*exp(-thetak[3]*
    +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
}
# E((U^alpha2+U^alpha2*logU)/U>v-u)
fu1expU1<-function(x,data)
{
  (log(x)+1)*x^thetak[2]*thetak[2]*thetak[3]*
     +x^(thetak[2]-1)*exp(-thetak[3]*
    +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
}
# E((U^alpha2+U^alpha2*log(U))^2/U>v-u)
fu12expU1<-function(x,data)
{
  (x^thetak[2]*log(x)+x^thetak[2])^2*thetak[2]*thetak[3]*
  +x^(thetak[2]-1)*exp(-thetak[3]*
    +(x^thetak[2]-(data$time_event-data$time_CR)^thetak[2]))
}
#cicle for all censored observation with time_CR known
ncrem<-length(indcrem) #number of censored obs. with time_CR known
u1expU<-NULL
u12expU<-NULL
u1expU1<-NULL
u12expU1<-NULL
i<-1
for(i in 1:ncrem)
{
```

```
  u1expU<-c(u1expU,integrate(fu1expU,data$time_event[indcrem[i]]-
   +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
  u12expU<c(u12expU,integrate(fu12expU,data$time_event[indcrem[i]]-
  +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
  u1expU1<-c(u1expU1,integrate(fu1expU1,data$time_event[indcrem[i]]-
   +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)
  u12expU1<-c(u12expU1,integrate(fu12expU1,data$time_event[indcrem[i]]-
  +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)

}
#E(log(u)+U^alpha2/V=v)
funtuexpU<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  (log(x)+x^thetak[2])*thetak[1]*thetak[2]*thetak[3]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((log(U)+U^alpha2)^2/V=v)
funtu2expU<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  (log(x)+x^thetak[2])^2*thetak[1]*
   +thetak[2]*thetak[3]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((1+log(u))*U^alpha2/V=v)
funtuexpU1<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
```

```
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  (x^thetak[2]*log(x)+x^thetak[2])*thetak[1]*
    +thetak[2]*thetak[3]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((log(U)*U^alpha+U^alpha2)^2/V=v)
funtu2expU1<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }
  (log(x)*x^thetak[2]+x^thetak[2])^2*thetak[1]*
    +thetak[2]*thetak[3]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}

ncrme<-length(indcrme) #number patients time_CR miss and event
u2exp1U<-NULL
u2exp2U<-NULL
u2exp1U1<-NULL
u2exp2U1<-NULL
for(i in 1:ncrme)
{
  u2exp1U<-c(u2exp1U,integrate(funtuexpU,0,
 +data$time_event[indcrme[i]],data=data[indcrme[i],])$value)
  u2exp2U<-c(u2exp2U,integrate(funtu2expU,0,
  +data$time_event[indcrme[i]],data=data[indcrme[i],])$value)
  u2exp1U1<-c(u2exp1U1,integrate(funtuexpU1,0,
  +data$time_event[indcrme[i]],data=data[indcrme[i],])$value)
  u2exp2U1<-c(u2exp2U1,integrate(funtu2expU1,0,
  +data$time_event[indcrme[i]],data=data[indcrme[i],])$value)

}

#E(U^alpha+logU/V>v)
survtuexpU<-function(x,data)
```

```
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  (log(x)+x^thetak[2])*thetak[3]*thetak[2]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((U^alpha+logU)^2/V>v)
survtu2expU<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  (log(x)+x^thetak[2])^2*thetak[3]*thetak[2]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}

#E(U^alpha+U^alpha*logU/V>v)
survtuexpU1<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  (x^thetak[2]*log(x)+x^thetak[2])*thetak[3]*thetak[2]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}
#E((U^alpha+U^alpha*logU)^2/V>v)
survtu2expU1<-function(x,data)
{
```

```
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[2]*z^(thetak[2]-1)*
      +exp(-thetak[3]*z^thetak[2]-thetak[1]*(data$time_event-z))
  }

  (x^thetak[2]*log(x)+x^thetak[2])^2*thetak[3]*thetak[2]*x^(thetak[2]-1)*
    +exp(-thetak[3]*x^thetak[2]-thetak[1]*(data$time_event-x))/
    +integrate(f1,0,data$time_event,data=data)$value
}

ncrmem<-length(indcrmem)
u3exp2U<-NULL
u32exp2U<-NULL
u3exp3U<-NULL
u32exp3U<-NULL
for(i in 1:ncrmem)
{
  u3exp2U<-c(u3exp2U,integrate(survtuexpU,0,
   +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
  u32exp2U<-c(u32exp2U,integrate(survtu2expU,0,
    +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
  u3exp3U<-c(u3exp3U,integrate(survtuexpU1,0,
    +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
  u32exp3U<-c(u32exp3U,integrate(survtu2expU1,0,
   +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)

}
#var(U^alpha2)
a1<-var2uexp+var3uexp+var1uexp
#var(U^alpha2log(U))
a2<-sum(u12exp2-(u1exp2)^2)+sum(u22exp2-(u2exp2)^2)+
  +sum(u32exp2-(u3exp2)^2)
#var(log(U))
a3<-sum(u12exp3-(u1exp3)^2)+sum(u22exp3-(u2exp3)^2)+
  +sum(u32exp3-(u3exp3)^2)
#E(U^alpha2log(U))
a4<-sum(u1exp2)+sum(u2exp2)+sum(u3exp2)
#var(U^alpha2+log(U))
a5<-sum(u2exp2U-u2exp1U^2)+sum(u12expU-u1expU^2)+
  +sum(u32exp2U-u3exp2U^2)
```

```
#var(U^alpha2*(1+log(U)))
a6<-sum(u2exp2U1-u2exp1U1^2)+sum(u12expU1-u1expU1^2)+
 +sum(u32exp3U-u3exp3U^2)

#Fisher Information alpha2,lambda2
#1/Cov(alpha2,lambda2)
InfAL2<-a4+a1/2+a3/2-a5/2-thetak[3]/2*(+a1+a2-a6)

return(c(1/ InfLambda1,1/InfAlpha2,1/InfLambda2,1/InfAL2))
}
```

Weibull distribution on both intervals

```
#Weibull-Weibull negative Q-function
# in input in addition to the current estimate of the
#parameter(thetak) are given the expected values
#computed separately in order to accelerate the algorithm
QfunWW<-function(theta,thetak,data,u1log,u2log=u2log,u3log=u3log,
  +t1starlog=t1starlog,t2starlog=t2starlog)
{
  n=nrow(data) #number of patients
  #indices patients with time_CR known
  indcr<-which(!(is.na(data$time_CR)))
  #indices obs with time_CR known and event
  indcre<-which(!(is.na(data$time_CR)) & data$event==1)
  #indices censored obs with time_CR known
  indcrem<-which(!(is.na(data$time_CR)) & data$event==0)
  #indices patients with time_CR missing and event
  indcrme<-which(is.na(data$time_CR) & data$event==1)
  #indices censored obs with time_CR missing
  indcrmem<-which(is.na(data$time_CR) & data$event==0)

  #function to compute conditional expectation ue*1=E(U^alpha2/U>v-t)
  fu1exp<-function(x,data)
  {
    thetak[3]*thetak[4]*x^(theta[3]+thetak[3]-1)*exp(-thetak[4]*
    +(x^thetak[3]-(data$time_event-data$time_CR)^thetak[3]))
  }

  #Loop for all censored observation with time_CR  known
```

```
ncrem<-length(indcrem) #number of censored obs. with time_CR known
u1exp<-NULL
i<-1
for(i in 1:ncrem)
{
  u1exp<-c(u1exp,integrate(fu1exp,data$time_event[indcrem[i]]-
  +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)

}
 #function to compute conditional expectation te*1=E(T^alpha1/V=v)
funtvexp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*thetak[4]*z^(thetak[1]-1)*
    +(data$time_event-z)^(thetak[3]-1)*exp(-thetak[2]*
    +z^thetak[1]-thetak[4]*(data$time_event-z)^thetak[3])
  }
  thetak[1]*thetak[2]*thetak[3]*thetak[4]*x^(theta[1]+thetak[1]-1)
  +*(data$time_event-x)^(thetak[3]-1)*exp(-thetak[2]*x^thetak[1]-
   +thetak[4]*(data$time_event-x)^thetak[3])/
    +integrate(f1,0,data$time_event,data=data)$value
}

#function to compute conditional expectation ue*2=E(U^alpha2/V=v)
funtuexp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*thetak[4]*z^(thetak[3]-1)*
    +(data$time_event-z)^(thetak[1]-1)*exp(-thetak[4]*
    +z^thetak[3]-thetak[2]*(data$time_event-z)^thetak[1])
  }
  thetak[1]*thetak[2]*thetak[3]*thetak[4]*x^(theta[3]+thetak[3]-1)*
  + (data$time_event-x)^(thetak[1]-1)*exp(-thetak[4]*
   +x^thetak[3]-thetak[2]*(data$time_event-x)^thetak[1])/
    +integrate(f1,0,data$time_event,data=data)$value
}

#Loop for patients with time_CR missing and event=1
ncrme<-length(indcrme) #number patients time_CR miss and event=1
```

```
t1starexp<-NULL
u2exp<-NULL
for(i in 1:ncrme)
{
   t1starexp<-c(t1starexp,integrate(funtvexp,0,
   +data$time_event[indcrme[i]],data=data[indcrme[i],])$value)
  u2exp<-c(u2exp,integrate(funtuexp,0,
  +data$time_event[indcrme[i]],data=data[indcrme[i],])$value)

}
#-------------------------------------------------------------
#function to compute conditional expectation te*2=E(T^alpha_1/V>v)
  survtvexp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*z^(thetak[1]-1)*
      +exp(-thetak[2]*z^thetak[1]-thetak[4]*
      +(data$time_event-z)^thetak[3])
  }
  thetak[1]*thetak[2]*x^(theta[1]+thetak[1]-1)*
    +exp(-thetak[2]*x^(2*thetak[1]-1)-thetak[4]*
     +(data$time_event-x)^thetak[3])/
    +integrate(f1,0,data$time_event,data=data)$value
}
#function to compute conditional expectation ue*3=E(U^alpha_2/V>v)
survtuexp<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[3]*thetak[4]*z^(thetak[3]-1)*
      +exp(-thetak[4]*z^thetak[3]-thetak[2]*(data$time_event-z)^thetak[1])
  }
  thetak[3]*thetak[4]*x^(theta[3]+thetak[3]-1)*
    +exp(-thetak[4]*x^thetak[3]-thetak[2]*(data$time_event-x)^thetak[1])/
    +integrate(f1,0,data$time_event,data=data)$value
}

#Loop for all censored observation with time_CR missing
t2starexp<-NULL
u3exp<-NULL
```

```
  for(i in 1:ncrmem)
  {
    t2starexp<-c(t2starexp,integrate(survtvexp,0,
     +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
    u3exp<-c(u3exp,integrate(survtuexp,0,
      +data$time_event[indcrmem[i]],data=data[indcrmem[i],])$value)
    }

  #Q-function: complete negative log-likelihood with missing values
  # replaced by their conditional expectations
  return(-( n*log(theta[1]*theta[2]*theta[3]*theta[4])+
    + (theta[1]-1)*sum( sum(log(data$time_CR[indcr]))+ sum(t1starlog)+
    + sum(t2starlog) )+ (theta[3]-1)*sum(sum(log(data$time_event[indcre]-
    + data$time_CR[indcre]))+sum(u1log)+sum(u2log)+sum(u3log)) -
    +theta[2]*sum(sum((data$time_CR[indcr])^theta[1])+sum(t1starexp)
    +sum(t2starexp)) - theta[4]*sum(sum((data$time_event[indcre]-
    +data$time_CR[indcre])^theta[3]) +sum(u1exp)+sum(u2exp)+sum(u3exp)) ))
}

#Function to  compute for a given thetak the estimate
# of the expected value for every missing values
WWelement<-function(data,thetak)
{
 #indices patients with time_CR known
  indcr<-which(!(is.na(data$time_CR)))
  #indices obs with time_CR known and event
  indcre<-which(!(is.na(data$time_CR)) & data$event==1)
  #indices censored obs with time_CR known
  indcrem<-which(!(is.na(data$time_CR)) & data$event==0)
  #indices patients with time_CR missing and event
  indcrme<-which(is.na(data$time_CR) & data$event==1)
  #indices censored obs with time_CR missing
  indcrmem<-which(is.na(data$time_CR) & data$event==0)

  #function to compute conditional expectation ul*1=E(log(U)/U>v-t)
  fu1log<-function(x,data)
  {
    log(x)*thetak[3]*thetak[4]*x^(thetak[3]-1)*exp(-thetak[4]*
    +(x^thetak[3]-(data$time_event-data$time_CR)^thetak[3]))
  }
  #Loop all censored observation with time_CR known
```

```
ncrem<-length(indcrem) #number of censored obs. with time_CR known
u1log<-NULL
i<-1
for(i in 1:ncrem)
{
  u1log<-c(u1log,integrate(fu1log,data$time_event[indcrem[i]]-
  +data$time_CR[indcrem[i]],Inf,data=data[indcrem[i],])$value)

}
#-----------------------------------------------------------------------
#function to compute conditional expectation tl*1=E(log(T)/V=v)
funtvlog<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*thetak[4]*z^(thetak[1]-1)*
    +(data$time_event-z)^(thetak[3]-1)*exp(-thetak[2]*
    +z^thetak[1]-thetak[4]*(data$time_event-z)^thetak[3])
  }
  log(x)*thetak[1]*thetak[2]*thetak[3]*thetak[4]*x^(thetak[1]-1)*
  +(data$time_event-x)^(thetak[3]-1)*exp(-thetak[2]*x^thetak[1]-
  +thetak[4]*(data$time_event-x)^thetak[3])/
    +integrate(f1,0,data$time_event,data=data)$value
}

#function to compute conditional expectation ul*2= E(log(U)/V=v)
funtulog<-function(x,data)
{
  f1<-function(z,data=data)
  {
    thetak[1]*thetak[2]*thetak[3]*thetak[4]*z^(thetak[3]-1)*
    +(data$time_event-z)^(thetak[1]-1)* exp(-thetak[4]*
    +z^thetak[3]-thetak[2]*(data$time_event-z)^thetak[1])
  }
  log(x)*thetak[1]*thetak[2]*thetak[3]*thetak[4]*x^(thetak[3]-1)*
  +(data$time_event-x)^(thetak[1]-1)* exp(-thetak[4]*
  +x^thetak[3]-thetak[2]*(data$time_event-x)^thetak[1])/
    +integrate(f1,0,data$time_event,data=data)$value
}

#Loop for patients with time_CR missing and event=1
```

```
 ncrme<-length(indcrme) #number patients time_CR miss and event=1
 t1starlog<-NULL
 u2log<-NULL
  for(i in 1:ncrme)
 {
   t1starlog<-c(t1starlog,integrate(funtvlog,0
    +,data$time_event[indcrme[i]], data=data[indcrme[i],])$value)
   u2log<-c(u2log,integrate(funtulog,0,data$time_event[indcrme[i]],
    +data=data[indcrme[i],])$value)

   }
 #-----------------------------------------------------------------
#function to compute conditional expectation tl*2=E(log(T)/V>v)
 survtvlog<-function(x,data)
 {
   f1<-function(z,data=data)
   {
     thetak[1]*thetak[2]*z^(thetak[1]-1)*
       +exp(-thetak[2]*z^thetak[1]-thetak[4]*
       +(data$time_event-z)^thetak[3])
   }
   log(x)*thetak[1]*thetak[2]*x^(thetak[1]-1)*
     +exp(-thetak[2]*x^thetak[1]-thetak[4]*
      +(data$time_event-x)^thetak[3])/
     +integrate(f1,0,data$time_event,data=data)$value
 }

 #function to compute conditional expectation ul*3=E(log(U)/V>v)
 survtulog<-function(x,data)
 {
   f1<-function(z,data=data)
   {
     thetak[3]*thetak[4]*z^(thetak[3]-1)*
       +exp(-thetak[4]*z^thetak[3]-thetak[2]*
       +(data$time_event-z)^thetak[1])
   }
   log(x)*thetak[3]*thetak[4]*x^(thetak[3]-1)*
     +exp(-thetak[4]*x^thetak[3]-thetak[2]*
      +(data$time_event-x)^thetak[1])/
     +integrate(f1,0,data$time_event,data=data)$value
 }
```

```
    #Loop for all censored observation with time_CR missing
   ncrmem<-length(indcrmem)
  t2starlog<-NULL
  u3log<-NULL
   for(i in 1:ncrmem)
  {
    t2starlog<-c(t2starlog,integrate(survtvlog,0,
     +data$time_event[indcrmem[i]],
    +data=data[indcrmem[i],])$value)
    u3log<-c(u3log,integrate(survtulog,0,data$time_event[indcrmem[i]],
     +data=data[indcrmem[i],])$value)

    }
  #-----------------------------------------------------------------
  return(list(u1log=u1log,u2log=u2log,u3log=u3log,
    +t1starlog=t1starlog,t2starlog=t2starlog))
}

#EM algorithm: function that takes in input the initial
#values for the parameter, the precision at which the
# estimation should be done, and the data.
#employing the EM algorithm return the MLE and the number of iterations
EMWeibWeib<-function(thetak,precision,data)
{
  difference<-1
  iter<-0
  #control: the function exit from while if the precision is obtained or
  #the maximum number of iteration are reached or
  #the difference between the estimated
  #mle and the current thetak is lower then 10^-4
  while(difference>precision & iter<1000)
  {
    iter<-iter+1
    el<-WWelement(data,thetak) #computation of the expected values
    u1log<-el$u1log
    u2log<-el$u2log
    u3log<-el$u3log
    t1starlog<-el$t1starlog
    t2starlog<-el$t2starlog
    #MLE at generic step k+1
```

```
    mle<nlminb(c(1,7,0.8,0.1),QfunWW,lower=rep(10^-7,4),
    +upper=rep(100,4),data=data,thetak=thetak,u1log=u1log,u2log=u2log,
   +u3log=u3log,t1starlog=t1starlog,t2starlog=t2starlog)$par

     #compute  Q(theta^k+1)-Q(theta^k)
     difference<-abs(abs(  QfunWW(theta=thetak,data=data
     +,thetak=thetak,u1log=u1log, u2log=u2log,u3log=u3log,
      +t1starlog=t1starlog,t2starlog=t2starlog))-abs( QfunWW(
     +theta=mle,data=data,thetak=thetak,u1log=u1log,u2log=u2log,
     +u3log=u3log,t1starlog=t1starlog,t2starlog=t2starlog)))

     #at each step is chosen the value for which the Q function is greater
     if(QfunWW(theta=mle,data=data,thetak=thetak,u1log=u1log,u2log=u2log,
    +u3log=u3log,t1starlog=t1starlog,t2starlog=t2starlog)<
    +QfunWW(theta=thetak,data=data,thetak=thetak,u1log=u1log,
     +u2log=u2log,u3log=u3log,
     +t1starlog=t1starlog,t2starlog=t2starlog))
    {
      #if the difference between mle and thetak
      # is lower than 10^-4 the function ends
      if ((abs(mle[1]-thetak[1]))<10^-4 & (abs(mle[2]-thetak[2]))<10^-4 &
        +(abs(mle[3]-thetak[3]))<10^-4 & (abs(mle[4]-thetak[4]))<10^-4 )
      {return(list(mle=thetak,iter=iter,diff=difference))}
    }
  }
   return(list(mle=thetak,iter=iter,diff=difference))
}

EMWWres<-EMWeibWeib(c(1,7,0.6,0.1),10^-4,data)
```

A.4 Simulation

```
  #Code to generate a data set
  lambda1<-7.6 #hazard CR
  lambda2<-0.27 #hazard relapse
  lambda3<-0.09 #hazard death
  #percentage censored
  perc
```

```
#percentage missing
perm

#repeat M times
time_CR<-rexp(n,lambda1) #time_CR
time_relapse<-rexp(n,lambda2) # time relapse
time_death<-rexp(n,lambda3) #time death
#Indicator variable 1=relapse, 2=death
CRAevent<-rep(1,n)
CRAevent[which(time_relapse>time_death)]<-CRAevent[which(time_relapse>time_
time_temp<-apply(cbind(time_relapse,time_death),1,min) #time to event from
time_event<-time_CR+time_temp #time to event from origin time
indCens<-sample(1:n,size=perc*n) # indices censored observation
#Indicator variable 0=censored, 1=relapse, 2=death
CRAevent[indCens]<-0
#Indicator variable 0=censored, 1=event
event<-rep(1,n)
event[indCens]<-0
indMiss<-sample(1:n,percm*n) #indices missing
time_CR[indMiss]<-NA #missing vlues
data<-cbind(time_CR,time_event,event,CRAevent)
data<-data.frame(data)
```

Bibliography

[1] L. C. de Wreede, M. Fiocco, and H. Putter. The mstate package for estimation and prediction in non- and semi-parametric multi-state and competing risks models. *Computer Methods and Programs in Biomedicine*, 99:261–274, 2010.

[2] L. C. de Wreede, M. Fiocco, and H. Putter. mstate: An r package for the analysis of competing risks and multi-state models. *Journal of Statistical Software*, 38, 2011.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B*, 39(1):1–38, 1977.

[4] H. Hartley and R. Hocking. The analysis of incomplete data. *Biometrics*, 27:7783–808, 1971.

[5] J. H. Jeong and J. Fine. Direct parametric inference for the cumulative incidence function. *Journal of the Royal Statistical Society Series C*, 55:187–200, 2006.

[6] V. Kapetanakis, F. E. Matthews, and A. van den Hout. A semi-markov model for stroke with piecewise-constant hazards. *Statistics in Medicine*, 2012.

[7] M. G. Kenward and J. Carpenter. Multiple imputation: current perspectives. *Statistical Methods in Medical Research*, 16(3):199–218, 2007.

[8] J. P. Klein and M. L. Moeschberger. *Survival Analysis: Techniques for Censored and Truncated Data*. Springer, second edition, 2003.

[9] T. A. Louis. Finding the observed information matrix when using the em algorithm. *Journal of the Royal Statistical Society. Series B*, 44(2):226–233, 1982.

[10] G. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley Series in Probability and Statistics - Applied Probability and Statistics Section Series. John Wiley & Sons, 1996.

[11] X. L. Meng and D. B. Rubin. Using EM to Obtain Asymptotic Variance-Covariance Matrices: The SEM Algorithm. *Journal of the American Statistical Association*, 86(416):899–909, 1991.

[12] T. Orchard and M. A. Woodbury. A missing information principle: Theory and applications. In E. L. Lucien Marie Le Cam, editor, *Proceedings of the 6 th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 697–715. University of California Press, 1972.

[13] H. Putter, M. Fiocco, and R. B. Geskus. Tutorial in biostatistics: Competing risks and multi-state models. *Statistics in Medicine*, 26:2389–2430, 2006.

[14] A. M. Reedijk, G. Kaspers, M. Fiocco, A. Pession, D. Reinhardt, M. Zimmerman, M. Dworzak, T. A. Alonzo, D. Johnston, M. Zapotocky, B. D. Moerloose, F. Finita, V. Lee, T. Taga, A. Tawa, A. Auvrignon, B. Zeller, C. Salgado, W. Balwierz, A. Popa, J. Rubnitz, H. B. Beverloo, G. C. J. Harrison, and B. Gibson. Clinical impact of additional cytogenetic aberrations and treatment in pediatric t(8;21)-positive aml: Results from an international retrospective i-bfm-sg study. Manuscript in preparation.

[15] D. B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

[16] D. B. Rubin. Multiple imputations in sample surveys: a phenomenological Bayesian approach to nonresponse. In *Proceedings of the Survey Research Methods Section*, pages 20–28, 1978.

[17] D. B. Rubin. *Multiple Imputation for Nonresponse in Surveys*. Wiley, 1987.

[18] J. L. Schafer. Multiple imputation: a primer. *Statistical Methods in Medical Research*, 8:3–15, 1999.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Professor Marta Fiocco for her support and guidance throughout this dissertation. Despite the distance, she has unfailingly e-mailed the information I needed (every time even in the night). This work would not have been possible without her.
I am very thankful to my Professor Guido Masarotto for his kindness and his continuous availability. His advices and opinions have made possible the conclusion of this thesis.
Finally, I am grateful for the endless encouragement which my family and my friends has given to me in every situation.
The Dutch Children Oncology Group (DCOG) is gratefully acknowledged for providing the data.