Probation review report

Alessandro Gasparini

2017-08-04

Contents

Introduction

This report presents the work I have done during my first year as a PhD student at the Department of Health Sciences, University of Leicester, under the supervision of Dr. Michael Crowther and Prof. Keith Abrams.

I will begin by briefly introducing the topic of survival analysis in Chapter 1. Second, I will introduce survival models with random effects (e.g. frailties, in the simplest form) and joint models for longitudinal and time-to-event data in Chapters 2 and 3, respectively. Computational challenges that survival models with random effects and joint models pose are presented in Chapter 4. Third, I will present the results of two simulation studies in Chapters 5 and 6; the first simulation study investigates the accuracy of quadrature methods when approximating analytically intractable terms, while the second simulation study investigates the impact of model misspecification in survival models with shared frailty terms. Fourth, I will introduce an interactive tool I have been developing to aid the dissemination of results from simulation studies in Chapter 7. Then, I will introduce the problem of informative visiting process in clinical research using healthcare consumption data in Chapter 8, and how we aim to evaluate and compare the different approaches that have been proposed and utilised in literature to tackle such problem in Chapter 9. Finally, I will briefly summarise the training and personal development activities I have participated to during the first year of my PhD in Chapter 10.

The report is written using bookdown (Xie, 2016), and can be accessed online at https://ellessenne.github.io/prr/.

Chapter 1

Introduction to survival analysis

Survival analysis is a branch of statistics in which the main outcome consists in the time until the occurrence of a given event. Time could be years, months, weeks, or any amount of calendar time or even age time; event could be death, disease occurrence or relapse, or any other experience of interest. Survival analysis is also known as reliability theory in engineering, duration analysis in economics, and event history analysis in sociology. A broad overview of survival analysis is given in Kalbfleisch and Prentice (2011) and in Kleinbaum and Klein (2012).

Some examples of time to event data are:

- disease remission in leukemia patients. In this study, leukemia patients are followed over several weeks to study how long they stay in remission status;
- heart disease occurrence. In this study, healthy subjects are followed over several years until occurrence of heart disease, or end of the study;
- renal failure. In this study, individuals with kidney disease are followed until renal failure, or end of the study;
- reliability of complex technical installations. For instance, studies assessing failure rates of components such as bulbs and valves.

In this Chapter I will define survival data and its peculiarities in Section 1.1 and 1.2. Terminology and notation used throughout this report will be introduced in Section 1.3. I will introduce common non-parametric and parametric methods in survival analysis in Sections 1.4 and 1.5. I will introduce the widely used semi-parametric Cox model in Section 1.6. Finally, I will provide a brief overview of modern, advanced statistical methods in survival analysis in Section 1.7.

1.1 Survival data

Survival data generally consists - as previously mentioned - in an event of interest and time until its occurrence. In the leukemia remission example, time to event would be how many weeks it takes before a given patient experiences disease relapse and the event would be whether the individual relapsed or not before the end of the study. Nevertheless, in certain situations we may have some information about the survival time but the actual survival time may be unknown. This problem is know as censoring and it is presented in Section 1.2.

1.2 Censoring

Censoring is a mechanisms that causes survival times to be unobserved. There are many reasons why censoring may occur; among others:

1. a person does not experience the event before the end of the study;
2. a person drops out of the study before the occurrence of the event of interest;
3. a person experience a competing event that impedes the occurrence of the event of interest (e.g.: death, when death is not the study outcome).

I simulated survival data for illustration purposes: I assumed a clinical trial with 10 individuals enrolled during a recruitment window of 1 year, and followed for up to 5 years. Not all individuals experience the event of interest during the study period, and are therefore censored after five years from the start of the study. The observation time for each individual is depicted in Figure 1.1 with a solid dark grey line, a cross represents the occurrence of the study event, and a circle represents censoring. Individuals `A`, `E`, and `J` all have censored survival time: I know that they were still event-free at the end of follow-up, i.e. their real survival time is greater than the observed one, but the former is unknown. The simulated data is presented in Figure 1.1: in panel A, survival data is plotted against the calendar time; conversely, in panel B, survival data is plotted against the study time, e.g. each individual is assigned a *time zero* corresponding to their enrollment in the study, and survival time is counted from there.

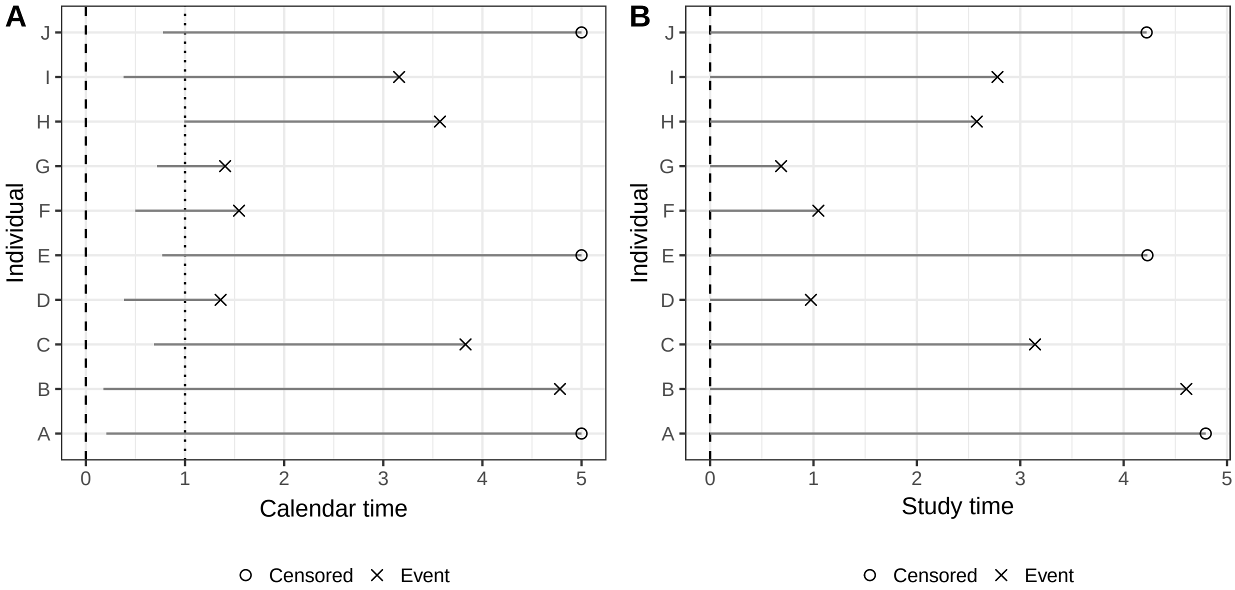

Figure 1.1: Simulated right censored survival data, plotted by their calendar time in panel A and by their study time in panel B.

This example represents a particular form of censoring: *right censoring*. The defining characteristic of right-censored data is that it is censored (or incomplete) at the right side of the follow-up time, hence the true survival time is greater than the observed time. This example represents *administrative censoring* as well, as individuals are censored at the end of the study to artificially restrict follow-up time (e.g. for financial reasons).

It is also possible to encounter data that is *left censored* or *interval censored*. In the former case, the true survival time is shorter that the observed one, e.g. I know that the event occurred before the observation time, but I do not know when - imagine onset of a viral infection, which can be detected only at a visit time. In the latter, I know that the event occurred within a certain interval of time but I do not know when; using the same example of infection onset, if infection was detected at a visit date but the individual was known to be infection-free at the previous visit, the true infection onset time is unknown and the event time is said to be interval censored.

Finally, another important concept related to right censoring is that of *left truncation* (or *delayed entry*). Left truncation occurs when an individual enrolls in the study some time after the inclusion criteria are satisfied; individuals that die (or emigrate, ...) before the start of observation time will never enter the study, and inclusion time may differ between individuals. Data arising from such phenomenon is therefore said to be left truncated.

1.3 Terminology and notation

I denote the random variable for an individual's survival time with S; since it denotes time, S can assume any non-negative value. The lower-case s represent a specific value of interest drawn from S for a given individual. In the case of right censoring, I denote with C the random variable representing censoring time, and c its realisation. The observed time is denoted with $T = \min(S, C)$, and its realisation is t. Finally, I denote with $D = I(S \leq C)$ the random variable indicating either occurrence of the event of interest or censorship; analogously as before, its realisation is lower-case d.

Next, I defined two quantities of interest in survival analysis, the *survival function* and the *hazard function*. They are both functions of the observed time t and are denoted by $S(t)$ and $h(t)$, respectively.

The survival function is the complement of the cumulative distribution function of the observed time T and represent the probability that a given individual survives[1] longer than a specified time t:

$$S(t) = 1 - F_T(t) = 1 - P(T \leq t) = P(T > t)$$

t ranges (theoretically) between 0 and infinity, hence the survival function can be plotted as a smooth, continuous function that tends to 0 as t goes to infinity. In practice, though, the survival function appears as a step function as (1) individuals can be observed at discrete times only and (2) not all individuals may experience the event before the end of the study. Figure 1.2 depicts this difference: in panel A I plotted a theoretical survival function, restricted to 15 years of follow-up for comparison purposes, while in panel B I plotted the survival function relative to the survival data simulated in Section 1.2. The former is a smooth function of time, and should we extend the x-axis to infinity the function would eventually reach zero. Conversely, the latter is a step function with steps at each event time, and should we extend the x-axis to infinity the function would remain flat after the last observed event.

The hazard function $h(t)$ is the limit of the probability of the survival time T laying within an interval $[t, t + \Delta(t))$ given that an individual survived up to time t divided by the length of the interval $\Delta(t)$, for $\Delta(t)$ approaching zero:

$$h(t) = \lim_{\Delta(t) \to 0} \frac{P(t \leq T < t + \Delta(t) | T \geq t)}{\Delta(t)}$$

[1] I use the term *survives* loosely speaking, for conciseness - formally, I refer to *not experiencing the event of interest*.

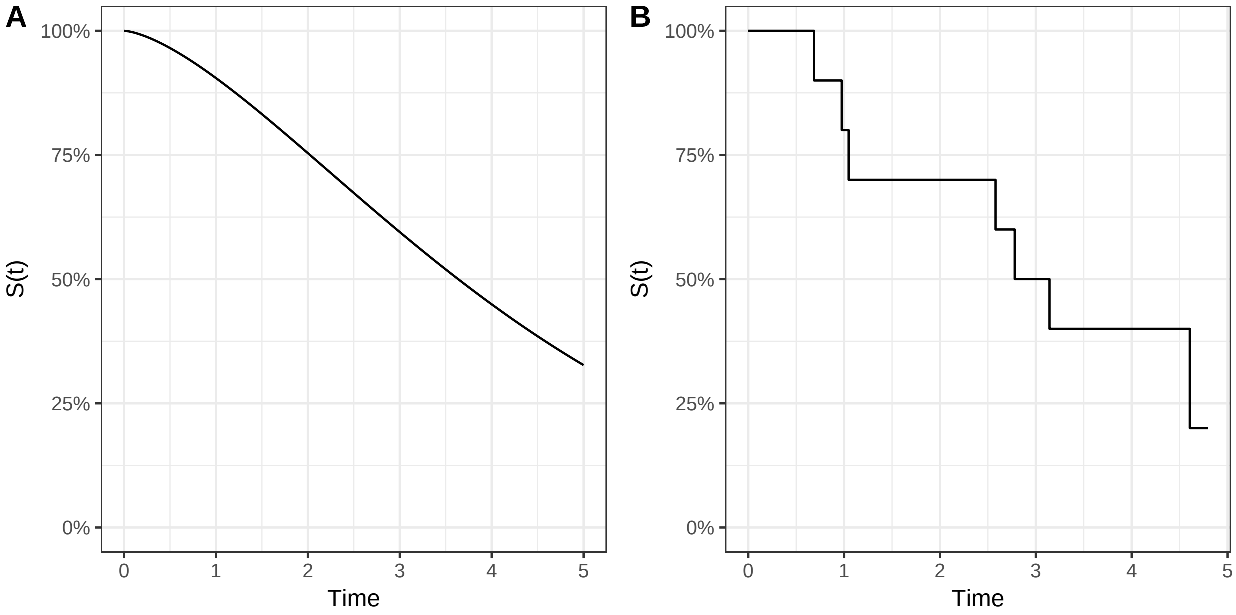

Figure 1.2: Theoretical survival function (A) and observed survival function for simulated data (B).

It represent the instantaneous potential (e.g. risk) for the event to occur within the interval $[t, t + \Delta(t))$ (with $\Delta(t) \rightarrow 0$), given that the individual survived up to time t. The hazard function is always non-negative, it can assume different shapes over time, and it has no upper bound. In Figure 1.3 I present a simple hazard function; it increases over time, which means that the instantaneous risk of event increases over time.

The survival function from Figure 1.2, panel A, and the hazard function from Figure 1.3 are strictly related. In fact, there is a clearly defined mathematical relationship between the survival and the hazard function: it is possible to derive the form of $S(t)$ when knowing the form of $h(t)$, and vice versa. Formally:

$$S(t) = \exp\left[-\int_0^t h(u)\ du\right]$$

$$h(t) = -\left[\frac{dS(t)/dt}{S(t)}\right]$$

Finally, a third quantity of interest in survival analysis that is strictly related to the survival and hazard functions is the cumulative hazard function $H(t)$. The cumulative hazard function represents the accumulation of hazard (e.g. $h(t)$) over time, and can be defined as

$$H(t) = \int_0^t h(u)\ du;$$

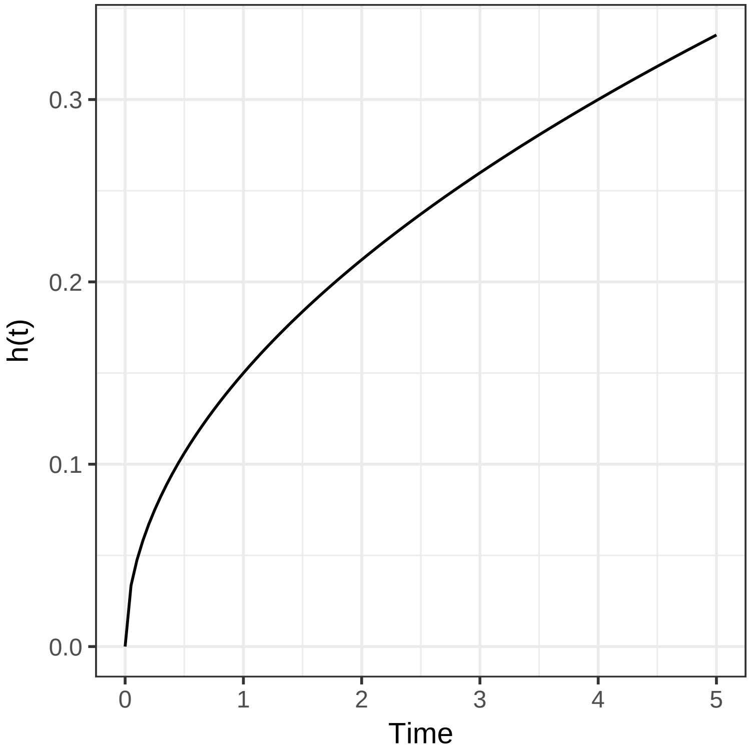

Figure 1.3: Example of hazard function.

it can conveniently be expressed in terms of survival function via the relationship $H(t) = -\log S(t)$, or alternatively with $S(t) = \exp(-H(t))$.

1.4 Estimation of the survival function

The survival function presented in Figure 1.2, panel B, is a non-parametric estimate of the true survival function based on the data only. The estimator employed is this case is the Kaplan-Meier estimator of the survival function (Kaplan and Meier, 1958), with which the estimated survival probabilities are obtained using a product limit formula. The general form for the Kaplan-Meier estimator at time $t_{(i)}$ is

$$\hat{S}(t_{(i)}) = \hat{S}(t_{(i-1)}) \times \hat{P}(T > t_{(i)} | T \geq t_{(i)}),$$

with $t_{(i)}$ being the i^{th} ordered failure time. The interpretation is straightforward: it is the product of the probability of surviving past the previous event-time ($\hat{S}(t_{(i-1)})$) times the conditional probability of surviving past the current time $t_{(i)}$ given survival to at least the current time ($\hat{P}(T > t_{(i)} | T \geq t_{(i)})$). The product limit formula is:

$$\hat{S}(t_{(i)}) = \prod_{j=1}^{i} \hat{P}(T > t_{(j)} | T \geq t_{(j)})$$

The conditional probability in the product limit formula can be estimated from the observed data as:

$$\hat{P}(T > t_{(i)} | T \geq t_{(i)}) = \frac{r_{(i)} - e_{(i)}}{r_{(i)}},$$

where $r_{(i)}$ and $e_{(i)}$ are the number of individuals at risk and the number of events at time $t_{(i)}$, respectively.

The Kaplan-Meier estimator can be computed using R and the function `survfit` from the `survival` package. An example using the simulated data from Section 1.2 (stored in a data frame named `data`):

```
# library(survival)
fit = survfit(Surv(time = t0, event = d) ~ 1, data = data)
summary(fit)
```

```
## Call: survfit(formula = Surv(time = t0, event = d) ~ 1, data = data)
##
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI
##  0.687     10       1      0.9  0.0949       0.7320        1.000
##  0.975      9       1      0.8  0.1265       0.5868        1.000
##  1.047      8       1      0.7  0.1449       0.4665        1.000
##  2.580      7       1      0.6  0.1549       0.3617        0.995
##  2.780      6       1      0.5  0.1581       0.2690        0.929
##  3.143      5       1      0.4  0.1549       0.1872        0.855
##  4.606      2       1      0.2  0.1612       0.0412        0.971
```

By doing so, I obtain an estimate of the survival function (column `survival`) at each distinct failure time (column `time`). For instance, the survival probability at $t = 1.047$ is 0.700, with 95% confidence interval (0.467 - 1.000).

Finally, plotting the estimated survival curve I obtain Figure 1.4, which is exactly the same survival curve presented in panel B of Figure 1.2.

```
# library(ggfortify)
autoplot(fit, conf.int = FALSE, censor = FALSE) +
  theme_bw() +
```

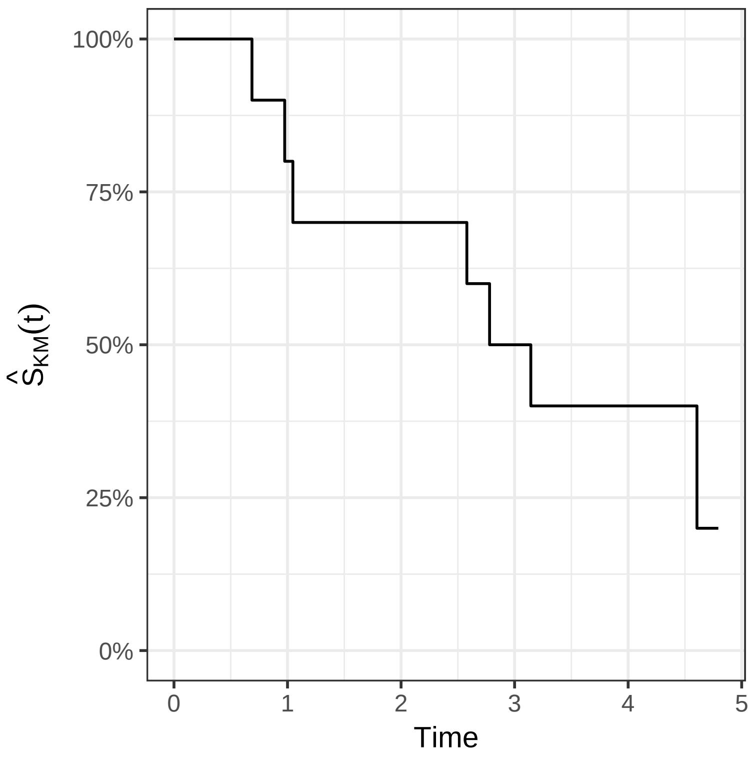

Figure 1.4: Estimated survival function using the Kaplan-Meier estimator on the simulated data.

```
  coord_cartesian(ylim = c(0, 1)) +
  labs(x = "Time", y = expression(hat(S)[KM](t)))
```

An alternative way of estimating the survival function is to use the *Nelson-Aalen* estimator for the cumulative hazard

$$\hat{H}(t) = \sum_{t_i<t} \frac{e_i}{r_i} = \sum_{t_i<t} \hat{h}_i,$$

and then use the relationship presented in Section 1.3 to obtain the survival function.

1.5 Parametric survival analysis

In applied settings it is often of interest to assess the association between observed covariates and the survival time of interest. For instance, it may be of interest to study whether a treatment is effective in slowing disease relapse (e.g. relapse of leukemia), whether there are difference between genders or age categories. A common way of assessing the effect of covariates on a time to event outcome, while adjusting for potentially confounding factors at the same time, consists in using a regression model.

In the context of survival data, two models are commonly used: the *accelerated failure time* model (AFT), and the *proportional hazards* (PH) model. In the former, the natural logarithm of the observed survival time $\log t$ is expressed as a linear function of the covariates X:

$$\log t = X\beta + \epsilon,$$

with β a vector of regression coefficients and ϵ a vector of residual error terms. Assuming a parametric distribution for ϵ determines the regression model: log-normal, log-logistic, Weibull, etc. In the AFT model, a positive association of the covariates with survival time implies an increased expected time to event. In the PH model, the covariates have a multiplicative effect on the hazard function:

$$h(t; X) = h_0(t)g(X),$$

for some $h_0(t)$ and $g(X)$, with $g(\cdot))$ a non-negative function of the covariates. A popular choice for the latter is $g(X) = \exp(X\beta)$; conversely, is is possible to either left the former unspecified, or assume a parametric distribution. The focus of this Section is on specifying a parametric distribution for $h_0(t)$, yielding the so-called *parametric survival regression models*; I will present commonly assumed parametric distributions in Section 1.5.1, the estimation procedure in Section 1.5.2, and an example using data from the International Stroke Trial (IST) (International Stroke Trial Collaborative Group, 1997, Sandercock et al. (2011)) in Section 1.5.3. Leaving $h_0(t)$ unspecified yields the semi-parametric Cox model, that I will present in Section 1.6.

From now on I will focus on the proportional hazards formulation of the survival model.

1.5.1 Failure time distribution

I mentioned in Section 1 that the random variable representing the survival time is non-negative; hence, we can choose any non-negative distribution to assign to $h_0(t)$. Commonly used distribution are the Exponential, Weibull, log-Normal, and Gompertz distributions; other possible distributions are the inverse Weibull, the log-skew-Normal (Azzalini, 1985), the log-logistic and complex mixture distributions (such as the two components mixture Weibull distribution, McLachlan and McGiffin (1994)). Each distribution yields a different Survival and hazard function:

- Exponential distribution:
  - $h_0(t) = \lambda$

  - $S(t) = \exp(-\lambda t)$
  - $\lambda > 0$
- Weibull distribution:
  - $h_0(t) = \lambda p t^{p-1}$
  - $S(t) = \exp(-\lambda t^p)$
  - $\lambda, p > 0$
- log-Normal distribution:
  - $h_0(t) = \frac{\phi(\frac{\log t - \mu}{\sigma})}{\sigma t \left(1 - \Phi\left[\frac{\log t - mu}{\sigma}\right]\right)}$
  - $S(t) = 1 - \Phi\left(\frac{\log t - \mu}{\sigma}\right)$
  - $\mu \in R; \sigma > 0$
- Gompertz distribution:
  - $h_0(t) = \lambda \exp(\gamma t)$
  - $S(t) = \exp\left[-\frac{\lambda}{\gamma}(\exp(\gamma t) - 1)\right]$
  - $\lambda, \gamma > 0$
- inverse Weibull distribution:
  - $h_0(t) = \frac{\lambda p t^{-(p+1)}}{\exp(\lambda t^{-p}) - 1}$
  - $S(t) = 1 - \exp(-\lambda t^{-p})$
  - $\lambda, p > 0$
- log-logistic distribution:
  - $h_0(t) = \frac{\exp(\alpha) \kappa t^{\kappa - 1}}{1 + \exp(\alpha) t^{\kappa}}$
  - $S(t) = \frac{1}{1 + \exp(\alpha) t^{\kappa}}$
  - $\alpha \in R; \kappa > 0$
- log-skew-Normal distribution[2]:
  - $h_0(t) = \frac{1 \phi\left(\frac{\log t - \xi}{\omega}\right) \Phi\left(\alpha \frac{\log t - \xi}{\omega}\right)}{t \omega [1 - SN(\log t; \xi, \omega, \alpha)]}$
  - $S(t) = 1 - SN\left(\log t; \xi, \omega, \alpha\right)$
  - $\xi, \alpha \in R; \omega > 0$
- Two components mixture Weibull distribution:
  - $h_0(t) = \frac{\lambda_1 p_1 t^{p1-1} \pi \exp(-\lambda_1 t^{p1}) + \lambda_2 p_2 t^{p2-1} (1-\pi) \exp(-\lambda_2 t^{p2})}{\pi \exp(-\lambda_1 t^{p1}) + (1-\pi) \exp(-\lambda_2 t^{p2})}$
  - $S(t) = \pi \exp(-\lambda_1 t^{p_1}) + (1 - \pi) \exp(-\lambda_2 t^{p_2})$
  - $\lambda_1, \lambda_2, p_1, p_2 > 0; \pi \in [0, 1]$

[2] $SN(\cdot)$ is the cumulative distribution function of a skew-Normal random variable with parameters ξ, ω, and α (Azzalini, 1985)

1.5.2 Estimation procedure

Assume n observations with the bivariate response (t_i, d_i), with $i = 1, \ldots, n$. For a given survival function $S(t)$ the density function is given by

$$f(t) = -\frac{dS(t)}{dt},$$

and the hazard function by

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

The parameters of the parametric proportional hazards survival model presented in Section 1.5 can be estimated via the maximum likelihood method. A subject that experiences the event of interest at time t_i contributes to the likelihood the density at time t_i, i.e. $f(t_i)$; conversely, a censored observation know to survive until tile t_i contributes $S(t_i)$ to the likelihood. The individual contribution to the likelihood L_i can therefore be written as

$$L_i = h(t_i)_i^d S(t_i),$$

where d_i is the event indicator variable. The overall likelihood is the product of the individual contributions:

$$L = \prod_{i=1}^{n} L_i.$$

Taking the natural logarithm of the likelihood for ease of computation:

$$\begin{aligned} \log L &= \sum_{i=1}^{n} \left[d_i \log f_i(t_i) + (1 - d_i) \log S_i(t_i) \right] = \\ &= \sum_{i=1}^{n} \left[d_i \log h_i(t_i) + \log S_i(t_i) \right] \end{aligned}$$

Implicit in the above log-likelihood are the regression parameters β and the parameters of the parametric distribution of choice for $h_0(t)$.

The log-likelihood function $\log L$ has a closed-form; maximum likelihood estimates for β and the distribution parameters can hence be obtained by maximising $\log L$, e.g. using one of the many general purpose optimisers available in R (`optim`, `nlm`, ...).

1.5.3 Data analysis example

The International Stroke Trial (IST) was a large, prospective, randomised controlled trial conducted between 1991 and 1996. The aim of the trial was to assess whether early administration of aspirin, heparin, both or neither influenced clinical outcomes in patients with acute ischaemic stroke (International Stroke Trial Collaborative Group, 1997, Sandercock et al. (2011)).

As illustration, I will evaluate the association between tretment with aspirin and/or heparin and survival after acute ischaemic stroke. I will start by reading the data, store in the `ist.csv` file. This file is a subset of the full IST dataset containing information on age, gender, treatment, and survival; further, individuals with missing values and individuals with a survival time of zero were dropped.

```
# library(readr)
ist = read_csv("data/ist.csv",
  col_names = c("gender", "age", "rxasp", "rxhep", "d", "t"),
  col_types = "ciccii", skip = 1)
attr(ist, "spec") = NULL # removing "spec" attribute

# turn treatments into factors
ist$rxasp = factor(ist$rxasp, levels = c("N", "Y"))
ist$rxhep = factor(ist$rxhep, levels = c("N", "L", "H"))
```

I fit first a parametric survival model assuming a Weibull distribution for $h_0(t)$. The hazard function, including covariates and the imposing proportional hazards, has the form

$$h(t; X) = \lambda p t^{p-1} \exp(X\beta),$$

while the survival function has the form

$$S(t; X) = \exp(-\lambda t^p \exp(X\beta)).$$

X is the model design matrix, and β is the vector of regression coefficients. The log-likelihood has the form

$$\log L = \sum_{i=1}^{n} [d_i \log h_i(t_i) + \log S_i(t_i)]$$

First, I code a function with the model log-likelihood. The function depends on (1) the model parameters β, λ, and p (pars argument), (2) the model design matrix X (X argument), and (3) survival time t and event indicator d (t and d arguments):

```
ll = function(pars, X, t, d) {
  lambda = exp(pars[1])
  p = exp(pars[2])
  beta = pars[-(1:2)]
  log_hi = log(lambda) + log(p) + (p - 1) * log(t) + c(X %*% beta)
  log_Si = -lambda * t ^ p * exp(c(X %*% beta))
  ll = sum(d * log_hi + log_Si + log(t))
  # + sum(log(t)) is the same adjustment that Stata
  # does to remove the time units from log L
  return(-ll)
}
```

The function ll() returns the negative log-likelihood as most optimisers minimise a target function (and so does optim); however, minimising the negative log-likelihood function is equivalent to maximising the log-likelihood.

I define the model matrix X for a model with aspirin treatment, heparin treatment, and their interactions. The first column is removed to avoid collinearity:

```
X = with(ist, model.matrix(t ~ rxasp * rxhep - 1))[,-1]
```

Next, I define the starting values for the optmisation routine. I choose the value 1 for the parameters of the Weibull distribution and the value 0 for the regression coefficients:

```
start = c(1, 1, rep(0, ncol(X)))
names(start) = c("lambda", "p", colnames(X))
```

The value of the log-likelihood function at the starting values is -92861140101.392. Finally, I use the modification robust-variance modification of the Marquard algorithm, which is more efficient than the Gauss-Newton-like algorithm when starting from points very far from the optimum (Marquardt, 1963, Commenges et al. (2006)):

```
# library(marqLevAlg)
fit = marqLevAlg(b = start,
  fn = function(x) ll(x, X = X, t = ist$t, d = ist$d))
```

```
##
## Be patient. The program is computing ...
## The program took 8.67 seconds
```

Assess convergency:

```
fit$istop
```

```
## [1] 1
```

The convergence status indicator is equal to 1, hence the convergence criteria were satisfied. The log-likelihood at the maximum likelihood estimates is 61566.567. The optimising routine returns the upper triangle matrix of variance-covariance estimates at the stopping point, which can be used to obtain standard errors of the estimated coefficients:

```
fit$vcov = matrix(0,
  nrow = length(fit$b),
  ncol = length(fit$b))
fit$vcov[upper.tri(fit$vcov, diag = TRUE)] = fit$v
fit$vcov[lower.tri(fit$vcov)] = t(fit$vcov)[lower.tri(fit$vcov)]
```

Finally, I build a table of results:

```
res = data.frame(
  coef = fit$b,
  hr = exp(fit$b),
  se = sqrt(diag(fit$vcov)))
res$z = res$coef / res$se
res$p = 2 * pmin(pnorm(-abs(res$z)), 1 - pnorm(abs(res$z)))
kable(res,
  digits = 3,
  align = "rrrrr",
  booktabs = TRUE,
  col.names = c("Beta", "Hazard ratio", "SE (Beta)", "Z", "P > |Z|"),
  linesep = "",
  caption = "Results from a parametric Weibull model.")
```

I test the interaction term using the Wald test to assess whether combining aspirin and heparin alter their association with time to event. I use the Wald χ^2 test statistic as the sample size is big enough for it to be equivalent to its F counterpart:

```
# identify the interaction terms
idx = grepl(":", names(fit$b))
```

Table 1.1: Results from a parametric Weibull model.

	Beta	Hazard ratio	SE (Beta)	Z	P > |Z|
lambda	-3.852	0.021	0.047	-82.665	0.000
p	-0.759	0.468	0.015	-52.050	0.000
rxaspY	-0.037	0.964	0.044	-0.850	0.395
rxhepL	0.085	1.088	0.052	1.640	0.101
rxhepH	0.044	1.045	0.052	0.843	0.399
rxaspY:rxhepL	-0.095	0.910	0.075	-1.269	0.204
rxaspY:rxhepH	0.037	1.038	0.074	0.503	0.615

```
# compute the W statistic
W = t(fit$b[idx]) %*% solve(fit$vcov[idx, idx]) %*% fit$b[idx]

# produce the test
c(W = W, df = sum(idx), `p-value` = 1 - pchisq(W, sum(idx)))
```

```
##         W        df   p-value
## 2.6104650 2.0000000 0.2711095
```

The interaction terms seem to be not statistically significant. We can conclude the association of aspirin and heparin treatments with survival are not dependent on one another.

I can then re-fit the model excluding the interaction terms:

```
X = with(ist, model.matrix(t ~ rxasp + rxhep - 1))[,-1]
start = c(1, 1, rep(0, ncol(X)))
names(start) = c("lambda", "p", colnames(X))
re_fit = marqLevAlg(b = start,
  fn = function(x) ll(x, X = X, t = ist$t, d = ist$d))
```

```
##
## Be patient. The program is computing ...
## The program took 4.51 seconds
```

```
re_fit$istop
```

```
## [1] 1
```

The routine converged. I produce the variance-covariance matrix:

Table 1.2: Results from a parametric Weibull model with no interactions.

	Beta	Hazard ratio	SE (Beta)	Z	P > |Z|
lambda	-3.845	0.021	0.044	-87.441	0.000
p	-0.759	0.468	0.015	-52.056	0.000
rxaspY	-0.051	0.950	0.030	-1.689	0.091
rxhepL	0.039	1.040	0.037	1.043	0.297
rxhepH	0.062	1.064	0.037	1.684	0.092

```r
re_fit$vcov = matrix(0,
  nrow = length(re_fit$b),
  ncol = length(re_fit$b))
re_fit$vcov[upper.tri(re_fit$vcov, diag = TRUE)] = re_fit$v
re_fit$vcov[lower.tri(re_fit$vcov)] = t(re_fit$vcov)[lower.tri(re_fit$vcov)]
```

Finally, I build a new table of results:

```r
re_res = data.frame(
  coef = re_fit$b,
  hr = exp(re_fit$b),
  se = sqrt(diag(re_fit$vcov)))
re_res$z = re_res$coef / re_res$se
re_res$p = 2 * pmin(pnorm(-abs(re_res$z)), 1 - pnorm(abs(re_res$z)))
kable(re_res,
  digits = 3,
  align = "rrrrr",
  col.names = c("Beta", "Hazard ratio", "SE (Beta)", "Z", "P > |Z|"),
  booktabs = TRUE,
  linesep = "",
  caption = "Results from a parametric Weibull model with no interactions.")
```

I now test the significance of the two coefficients related to heparin treatment jointly:

```r
idx = grepl("^rxhep", names(re_fit$b))
W = t(re_fit$b[idx]) %*% solve(re_fit$vcov[idx, idx]) %*% re_fit$b[idx]
c(W = W, df = sum(idx), `p-value` = 1 - pchisq(W, sum(idx)))
```

```
##       W      df p-value
## 3.08001 2.00000 0.21438
```

Heparin treatment seems to be not statistically significantly associated with time to death in acute ischaemic stroke patients; the effect size is small, with a 4% and 7% increased risk for the `L` and `H` heparin treatment modalities versus no heparin treatment, respectively.

Finally, the treatment with aspirin is also not statistically significantly associated with the outcome; effect size is small as well, approximately a 5% risk reduction for aspirin treatment compared to no treatment with aspirin (Table 1.2).

This is a simple application of parametric survival models; a fully developed analysis should take further aspects into account, such as considering different hazard distributions. It is possible to estimate various models and compare their fit to a specific distribution using information criteria such as the Akaike information criterion (AIC) and the Bayesian information criterion (BIC).

1.6 The Cox proportional hazards model

The parametric survival models of Section 1.5 could have both the accelerated failure time form and the proportional hazards form. Recall that the latter is formulated in terms of the hazard function:

$$h(t; X) = h_0(t) \exp(X\beta)$$

As I mentioned before, this model requires specifying the baseline hazard function $h_0(t)$ (e.g. using one of the parametric distributions of Section 1.5.1) and by leaving it unspecified I obtain the Cox proportional hazards model. Such model is also called *semi-parametric* as it is formed by a non-parametric component (the baseline hazard left unspecified) and a parametric component (the modelling assumption for the functional form of $g(\cdot)$, the usual $\exp(X\beta)$ in this case). The survival function for a Cox model can be written as:

$$S(t; X) = \exp\left[-\int_0^t h_0(u) \exp(X\beta)\ du\right]$$

The main problems when fitting a Cox model are related to estimation of the regression coefficients β and of the survival function $S(t)$.

The main method for estimating the regression coefficients is the method of partial likelihood, proposed and discussed in detail in Cox (1972) and Cox (1975). In brief, the observed data are assumed to have density function $f(t; \theta, \beta)$ in which β is the vector of regression coefficients of interest and θ can be considered a vector of nuisance parameters. In particular, θ represents the unspecified function $h_0(t)$. It can be showed that is is possible to factorise the density into two terms, one of which only depends on β: this term is called *partial likelihood.* Ignoring the term that depends on θ, and even if the partial likelihood is not directly interpretable as a likelihood in the ordinary sense, it can be used like an ordinary likelihood for estimation purposes as the usual asymptotic properties formulas and properties associated with the likelihood function and likelihood estimation apply. The partial likelihood applies directly to the relative risk model $h(t; X)$, assuming independent right censoring. The individual contribution to the likelihood has the form

$$L_i(\beta) = \frac{h(t_i; x_i)\Delta t_i}{\sum_{l \in R(t_i)} h(t_i; x_l)\Delta t_i},$$

and provides information on failures occurrence in the interval $[t_i, t_i + \Delta t_i)$; $R(t_i)$ is the risk set of individuals at risk of failing at time t_i^-, right before t_i. Under the relative risk model, the baseline hazard in $h(t; X)$ cancels out in the numerator and denominator; the product over i gives the partial likelihood for β:

$$L(\beta) = \prod_{i=1}^{n} \frac{\exp(x_i\beta)}{\sum_{l \in R(t_i)} \exp(x_l\beta)}.$$

The values of β that maximise the partial likelihood $\hat{\beta}$ can be obtained by using a Newton-Raphson-like algorithm; asymptotics are fully analogous to a parametric likelihood. A caveat of the partial likelihood method is that it assumes continuous failure times: in practice, that is unrealistic and there will be tied failure times (e.g. due to rounding). In that case, several methods have been proposed to adjust the partial likelihood in order to handle ties; see for instance Peto (1972), Breslow (1974), and Efron (1977)

Next, consider deriving an estimator for the survival function from a Cox model. The form of $h_0(t)$ is unspecified, hence it is not possible to directly estimate the parameters of the distribution as in fully parametric survival models. Under a Cox model, the survival function has the form

$$S(t; X) = S_0(t)^{\exp(X\beta)}$$

The coefficients β are estimated using the penalised likelihood procedure, and the baseline survival function $S_0(t)$ is estimated by assuming that the baseline hazard function is constant between each pair of consecutive observed failure times. The resulting estimator, known as the Breslow estimator, estimates the cumulative baseline hazard function as

$$\hat{H}_0(t) = \sum_{t(i) \leq t} \frac{e_{(i)}}{\sum_{l \in R(t_{(i)})} \exp(x_l\hat{\beta})},$$

with $e_{(i)}$ the number of events at time $t_{(i)}$. The baseline survival function follows as

$$\hat{S}_0(t) = \exp\left[-\hat{H}_0(t)\right],$$

and the survival function as

$$\hat{S}(t; X) = \hat{S}_0(t)^{\exp(X\hat{\beta})}.$$

An alternative estimator based on approximating the baseline survival function as a step function and consequently solving k simultaneous equations has been proposed by Kalbfleisch and Prentice (2011), and is omitted here.

Finally, the Cox model relies on two main assumptions. First, the assumption of non-informative censoring, e.g. the censoring process must be independent of any covariate, observed and not. Second, the proportional hazards assumption requires hazards to be proportional across time, e.g. the hazard must be constant. There are several ways of testing the proportional hazards assumption, both analytical and graphical; see Chapter 4 of Kleinbaum and Klein (2012) for further details.

1.6.1 Data analysis example

In this section I re-analyse the IST data of Section 1.5.3 using a semi-parametric Cox model. I first read the dataset:

```r
# library(readr)
ist = read_csv("data/ist.csv",
  col_names = c("gender", "age", "rxasp", "rxhep", "d", "t"),
  col_types = "ciccii", skip = 1)
attr(ist, "spec") = NULL # removing "spec" attribute

# turn treatments into factors
ist$rxasp = factor(ist$rxasp, levels = c("N", "Y"))
ist$rxhep = factor(ist$rxhep, levels = c("N", "L", "H"))
```

I fit the Cox model using the `coxph()` function from the `survival` package:

```r
# library(survival)
fit = coxph(Surv(t, d) ~ rxasp * rxhep, data = ist)
summary(fit)
```

```
## Call:
```

```
## coxph(formula = Surv(t, d) ~ rxasp * rxhep, data = ist)
##
##   n= 19378, number of events= 4315
##
##                    coef exp(coef) se(coef)      z Pr(>|z|)
## rxaspY         -0.03710   0.96358  0.04356 -0.852    0.394
## rxhepL          0.08017   1.08347  0.05161  1.553    0.120
## rxhepH          0.04215   1.04305  0.05221  0.807    0.419
## rxaspY:rxhepL -0.09049   0.91348  0.07458 -1.213    0.225
## rxaspY:rxhepH  0.03595   1.03661  0.07415  0.485    0.628
##
##                exp(coef) exp(-coef) lower .95 upper .95
## rxaspY            0.9636     1.0378    0.8847     1.049
## rxhepL            1.0835     0.9230    0.9792     1.199
## rxhepH            1.0431     0.9587    0.9416     1.155
## rxaspY:rxhepL     0.9135     1.0947    0.7893     1.057
## rxaspY:rxhepH     1.0366     0.9647    0.8964     1.199
##
## Concordance= 0.513  (se = 0.004 )
## Rsquare= 0   (max possible= 0.987 )
## Likelihood ratio test= 7.98  on 5 df,   p=0.1574
## Wald test            = 8.02  on 5 df,   p=0.1554
## Score (logrank) test = 8.02  on 5 df,   p=0.1551
```

I test again the joint significancy of the interaction terms using the Wald test:

```
idx = grepl(":", names(coef(fit)))
W = t(coef(fit)[idx]) %*% solve(vcov(fit)[idx, idx]) %*% coef(fit)[idx]
c(W = W, df = sum(idx), `p-value` = 1 - pchisq(W, sum(idx)))
```

```
##         W        df   p-value
## 2.3929136 2.0000000 0.3022633
```

Analogously as before, the interaction is not significantly different than zero. I re-fit the model without the interaction term:

```
# library(survival)
re_fit = coxph(Surv(t, d) ~ rxasp + rxhep, data = ist)
summary(re_fit)
```

```
## Call:
## coxph(formula = Surv(t, d) ~ rxasp + rxhep, data = ist)
##
##   n= 19378, number of events= 4315
##
##            coef exp(coef) se(coef)      z Pr(>|z|)
## rxaspY -0.05079   0.95048  0.03046 -1.668   0.0954 .
## rxhepL  0.03641   1.03708  0.03725  0.978   0.3283
## rxhepH  0.05991   1.06174  0.03707  1.616   0.1061
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##        exp(coef) exp(-coef) lower .95 upper .95
## rxaspY    0.9505     1.0521    0.8954     1.009
## rxhepL    1.0371     0.9642    0.9641     1.116
## rxhepH    1.0617     0.9419    0.9873     1.142
##
## Concordance= 0.511  (se = 0.004 )
## Rsquare= 0   (max possible= 0.987 )
## Likelihood ratio test= 5.58  on 3 df,   p=0.1337
## Wald test            = 5.59  on 3 df,   p=0.1333
## Score (logrank) test = 5.59  on 3 df,   p=0.1332
```

Testing the significancy of the heparin treatment using the Wald test:

```
idx = grepl("^rxhep", names(coef(fit)))
W = t(coef(fit)[idx]) %*% solve(vcov(fit)[idx, idx]) %*% coef(fit)[idx]
c(W = W, df = sum(idx), `p-value` = 1 - pchisq(W, sum(idx)))
```

```
##         W        df   p-value
## 2.4963250 2.0000000 0.2870317
```

Treatment with heparin is not statistically significantly associated with risk of death; besides that, the effect size of heparin treatment is small: approximately 4% and 6% risk increase for heparin treatment modalities L and H compared to no heparin treatment, respectively.

The treatment with aspirin is also not barely significantly different than zero, with a p-value of 0.0954; the effect size is comparable to the effect size estimated with the Weibull model, approximately 5% risk reduction for treatment with aspirin compared to no aspirin treatment.

1.7 Advanced survival analysis

Chapter 2

Survival models with random effects

Chapter 3

Joint models for longitudinal and survival data

Chapter 4

Computational challenges in survival models with random effects

Chapter 5

Simulation study: accuracy of Gaussian quadrature

5.1 Aim

5.2 Data-generating mechanisms

5.3 Methods

5.4 Estimands

5.5 Performance measures

5.6 Results

Chapter 6

Simulation study: impact of misspecification in survival models with shared frailty terms

6.1 Aim

6.2 Data-generating mechanisms

6.3 Methods

6.4 Estimands

6.5 Performance measures

6.6 Results

Chapter 7

Exploring results from simulation studies interactively

Chapter 8

Informative visiting process

Chapter 9

Future research developments

Chapter 10

Personal development

In this chapter I will introduce and briefly discuss the personal development activities I carried out during the first year of my PhD. In particular, I will present the supervisory meetings, training courses, and conferences I attended.

10.1 Supervisory meetings

I have been having frequent meetings with my supervisors, formally and informally. Formal supervisory meetings, recorded on PROSE (https://prose.le.ac.uk), have been held on average every other week, with summaries produced and shared between us. A comprehensive list is available on PROSE. Additionally, we held informal meetings to discuss developments and more urgent matters more often, whenever it was needed and every week on average.

10.2 Training and courses

I have attended a wide variety of courses during my first year, both externally and internally to the University of Leicester. The external courses I attended are:

- *Efficient R Programming*, on November 8th 2016, organised by the Royal Statistical Society in London. The instructor was Dr. Colin Gillespie, from the University of Newcastle, United Kingdom, and Jumping Rivers. The course covered how to program efficiently with R; in particular, it covered common pitfalls when writing R code, code profiling, RCpp, and parallel programming. General hints and tips were provided.

- *Introduction to causal inference*, on April 25th and 26th 2017, organised by the Biostatistics Research Group at the University of Leicester and delivered by Dr. Arvid Sjölander from Karolinska Institutet, Stockholm, Sweden. The course provided foundational concepts of causal inference such as the difference between association and causation, the counterfactual framework, exchangeability, directed acyclic graphs, methods for estimating a causal effect, etc. Additionally, it provided an introduction to more advanced methods such as intrumental variables and Mendelian randomisation.

- *Using simulation studies to evaluate statistical methods*, on May 22nd 2017, organised by University College London. The course was delivered by Dr. Tim Morris, Prof. Ian White and Dr. Michael Crowther, and it covered the rationale for using simulation studies, important concepts to keep in mind when planning a simulation study, computational tools, estimates of uncertainty, and tools for improving reporting and dissemination.

- Workshop on *Joint modelling of longitudinal and time-to-event data with R*, on July 5th, 2017, organised by the Department of Biostatistics of the University of Liverpool. The course was delivered by Dr. Graeme Hickey, and provided an introduction to joint models of longitudinal and survival data, including extensions to incorporate competing risks and multiple longitudinal processes and a practical session using R.

I have attended a few courses within the University and not offered on PROSE; specifically, I attended a course on *Time series analysis with R* (November 10th, 2016), a course on *Data visualisation* (November 15th, 2016), and a course on *High performance computing at Leicester* (February 8th, 2017). The latter was particularly important, as it allowed me to make better use of the high-performance computing facilities offered by the University. I also attended the *Preparing to teach in higher education* workshop, strand A (July 24th and 27th 2017).

Additionally, I have attended the following PROSE training sessions to develop personal and communication skills in research settings. These are listed below:

- *Planning your literature search*, October 21st 2016;

- *Conducting your literature search*, October 25th 2016 ;

- *Assertiveness*, November 14th 2016;

- *Introduction to critical thinking*, December 15th 2016;

- *Presentations A: Fundamentals of an effective presentation*, January 30th 2017;

- *Communication in research and other work settings*, January 31st 2017;
- *Enhancing your digital profile*, February 2nd 2017;
- *Saying it with your abstract*, February 10th 2017;
- *Designing a poster*, February 27th 2017;
- *Leadership in research and other work environments*, February 28th 2017;
- *Preparing for the probation review (Physical natural and medical sciences)*, May 30th 2017.

10.3 Conferences

I have attended a number of conferences during this year, in which I delivered the following oral presentations:

- Survival Analysis for Junior Researchers conference, held in Leicester, UK, on April 5th and 6th 2017. I delivered a talk titled *Direct likelihood maximisation using numerical quadrature to approximate intractable terms*;
- Statistical Analysis of Multi-Outcome Data (SAM) conference, held in Liverpool, UK, on July 3rd and 4th 2017. I delivered a talk titled *Impact of model misspecification in survival models with frailties*;
- Annual Conference of the International Society for Clinical Biostatistics conference, held in Vigo, Spain, on July 9th to July 13th 2017. I delivered two talks: a titled *Impact of model misspecification in survival models with frailties* during the main conference, and a talk titled *Exploring results from simulation studies interactively* during the Students' Day organised on July 13th.

Additionally, I delivered an oral presentation on previous work external to my PhD project during the 54th ERA-EDTA Congress held in Madrid, Spain, between June 3rd and June 6th. The ERA-EDTA Congress is the main conference in the field of Nephrology in Europe, with approximately 10,000 participants in 2017. I delivered my presentation, titled *Inappropriate prescription of nephrotoxic drugs to individuals with chronic kidney disease*, to an audience of clinicians, epidemiologists, clinical researchers, and other stakeholders.

Appendix A

Slides

Appendix B

Manuscript

Bibliography

Azzalini, A. (1985). A class of distributions which include the normal ones. *Scandinavian Journal of Statistics*, 12(2):171–178.

Breslow, N. (1974). Covariance analysis of censored survival data. *Biometrics*, 30(1):89–99.

Commenges, D., Jacqmin-Gadda, H., Proust, C., and Guedj, J. (2006). A newton-like algorithm for likelihood maximization: the robust-variance scoring algorithm. *ArXiv Mathematics e-prints*.

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 34(2):187–220.

Cox, D. R. (1975). Partial likelihood. *Biometrika*, 62(2):269–276.

Efron, B. (1977). The efficiency of Cox's likelihood function for censored data. *Journal of the American Statistical Association*, 72(359):557–565.

International Stroke Trial Collaborative Group (1997). The international stroke trial (ist): a randomised trial of aspirin, subcutaneous heparin, both, or neither among 19 435 patients with acute ischaemic stroke. *The Lancet*, 349(9065):1569–1581.

Kalbfleisch, J. D. and Prentice, R. L. (2011). *The statistical analysis of failure time data*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc, 2 edition.

Kaplan, E. L. and Meier, P. (1958). Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, 53(282):457–481.

Kleinbaum, D. G. and Klein, M. (2012). *Survival analysis: A self-learning text*. Springer-Verlag New York, 3 edition.

Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441.

McLachlan, G. and McGiffin, D. (1994). On the role of finite mixture models in survival analysis. *Statistical Methods in Medical Research*, 3(3):221–226.

Peto, R. (1972). Contribution to discussion of paper by D.R. Cox. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 34:205–207.

Sandercock, P., Niewada, M., and Czlonkowska, A. (2011). International stroke trial database (version 2). Technical report, University of Edinburgh, Department of Clinical Neuroscience.

Xie, Y. (2016). *bookdown: Authoring books and technical documents with R Markdown*. The R Series. Chapman & Hall / CRC.