Bayesian Models for Joint Longitudinal and Multi-State Survival Data

by

Allison Kate Cullen Furgal

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Biostatistics)
in The University of Michigan
2021

Doctoral Committee:

Research Professor Ananda Sen, Co-Chair
Professor Jeremy M. G. Taylor, Co-Chair
Assistant Professor Yang Chen
Professor Susan Murray

Allison Kate Cullen Furgal

acullen@umich.edu
ORCID: 0000-0001-8081-2444

To Joe and Aubrey

Acknowledgments

I am grateful to many people who have helped in my long journey to completing this dissertation. I would first like to thank my advisor Ananda Sen for your encouragement and mentorship these several years. I could not have done this without your help. To my co-chair Jeremy M. G. Taylor, thank you for offering me a spot on the Cancer Training grant when I first came to the department. That set me on the path to this work and I am thankful for the opportunity to learn from you. I would also like to thank Susan Murray and Yang Chen for being on my committee and for your helpful comments after my proposal.

I am grateful to the many people I have worked with and learned from in my time as a GSRA including Yun Li, Stephanie Daignault-Newton, and the members of the Family Medicine department including Missy Plegue, Mike Fetters, Justine Wu, Marie Claire O'Dwyer, Kim McKee, Todd Arnedt, and Chi Tajima. I have gained a tremendous amount of confidence in myself as a biostatistician while working with you all. I would also like to thank Stephanie Daignault-Newton and Dr. Barbra S. Miller for permission to use the adrenocortical carcinoma data in this work.

Thank you to all of my friends who mean so much to me. To my Biostat friends, Nick Seewald, Jeremy Quick, Emma Beyers-Carlson, Jed Carlson, Evan and Riley Reynolds, Lauren Beesley, Sarah Abraham, Marco Benedetti, Julie Strominger, and Krithika Suresh, thank you for all of the good times and for reminding me that I too could finish this. To Patty Walworth, thank you for your daily texts, your support, and for making it easy to stay in touch.

I would like to thank my family for all their support and love. Thank you to my sister Lindsey Kryger for being there when I needed to talk. To my mom, Shelly Gore, for everything you've done for my my entire life. To my dad, Tim J. Cullen, I know you would have been so proud of me. And thank you to my brother Tim R. Cullen, my step-dad Marty Gore, and my in-alws Judy, Denis and Katie Furgal for being there for me.

To baby Furgal, thank you for that extra bit of motivation over the last few months to get this done.

Finally, thank you to my husband, Joe Furgal, for being by my side for the last thirteen years and beyond. Thank you for putting up with me on my bad days, celebrating the good days, and for never letting me quit.

Table of Contents

List of Tables

List of Figures

List of Appendices

Abstract

Biomedical data commonly include repeated measures of biomarkers and disease states over time. When the processes determining the biomarker levels and disease states are related, a joint longitudinal and survival model is needed to properly handle the data. In a recent study of adrenal cancer patients at the University of Michigan, their tumors were monitored with repeated radiography scans. Other body measurements, called morphomics, were also measured from these scans. At each scan, it was noted whether the patient's disease was stable, progressing or regressing. In addition, the data include time to death or end of follow-up. Motivated by this data we explore joint models for longitudinal and survival data of several types.

In Chapter 2 we compare computational approaches to joint longitudinal and survival models with a single type of event. We examine different joint model formulations especially those most often implemented in software available to statisticians and clinicians. We apply and compare several models to the adrenal data and perform a simulation study to further evaluate each model and software.

In Chapter 3 we examine the relationship between a morphomic variable and time to first disease state change which can be either cancer progression or regression, in the adrenal cancer data. We develop Bayesian joint models for longitudinal and competing risks survival data. A seldom considered aspect of competing risk joint models is the relationship between the two competing outcomes. This cannot be examined when using the most common technique, cause-specific hazards models. With that motivation for our future projects, we work under the assumption that each risk has a latent failure time for each individual. We begin with the simple case of conditionally in-

dependent risks and model the survival times using parametric distributions. We apply our models to the adrenal data and examine the performance via simulations.

In Chapter 4 we extend our joint longitudinal and competing risks models for dependent competing risks. We begin with a discussion of survival copulas and the general joint survival function we will use which is based on an Archimedean copula model. We prove that dependent variables with this joint survival function can be written in terms of independent variables which is useful for simulating data. We develop the model with Weibull marginals. We fit this model to the adrenal data and examine the models using a simulation study. We discuss interpretations of the model and how it can be used to learn about the dependence between risks.

Finally, in Chapter 5 we will develop a joint model that incorporates multiple longitudinal outcomes and multistate survival data. We will develop an appropriate model and apply it to the adrenal cancer data.

Chapter 1

Introduction

Observational studies with a time-to-event outcome will often include longitudinal measurements associated with the event time outcome. An early and still studied example is CD4 counts over time and time until progression from HIV to AIDS (Wulfsohn and Tsiatis, 1997; Wang and Taylor, 2001; Huang et al., 2011). In cancer, increasing PSA levels are known to be associated with prostate cancer recurrence (Proust-Lima and Taylor, 2009). Other examples include the study of biomarkers in cancer vaccine trials, quality of life measurements in cancer, and time-to-failure and degradation in engineering applications (Brown and Ibrahim, 2003; Ibrahim et al., 2010; Lehmann, 2009). A new example motivating this work is a study of adrenocortical carcinoma patients at the University of Michigan that recorded body measurements over time as well as times for events including cancer progression and death.

In such contexts, separate models for the longitudinal and survival components that do not take into account the dependence between the longitudinal and survival processes produce inefficient results and are prone to bias (Ibrahim et al., 2010). Ideally, a joint model would be used. Less complicated alternatives to joint models include using time-dependent covariates or two-stage models but these each have drawbacks. Including the longitudinal process as a time-dependent covariate in the survival model requires the unrealistic assumptions that the longitudinal measurement has negligible measurement error and that there are observations at every event time (Mccrink et al., 2013). Traditional Two-Stage models will fit the longitudinal model first, disregarding the survival outcome. Then the fitted longitudinal trajectory is included as a time-dependent variable in the survival model (Tsiatis et al., 1995). This will account for the measurement error within the longitudinal process in the survival model. However, this strategy fails to account for the possible

1

informative censoring in the longitudinal process created by the survival event.

A fully specified joint model incorporates the dependence between the longitudinal and survival components properly and provides efficient inference that is less prone to bias. Joint models lend themselves to answering several possible research questions such as the effect of covariates on one outcome, the association between the outcomes, hypothesis testing, or outcome prediction (Rizopoulos, 2012, p. 9-10).

There are differing approaches to formulating joint models. We shall focus on random effects models as these are the most commonly used. Random effects models assume the likelihoods for the longitudinal and time-to-event models each include correlated random effects, consequently linking the two processes. Some other likelihood approaches include Selection Models and Pattern-Mixture Models (Mccrink et al., 2013). Another formulation is latent class models, in which subjects are assumed to belong to one of a number of latent classes and association is induced by class membership (Proust-Lima et al., 2014; Proust-Lima and Taylor, 2009; Lin et al., 2002; Proust-lima et al., 2015; Rouanet et al., 2016; Andrinopoulou et al., 2018; Sun et al., 2019).

Early work on joint models dates back to the mid-nineties (De Gruttola and Tu, 1994). Since then a body of work has steadily emerged in this area of research (Henderson et al., 2000; Wang and Taylor, 2001; Wulfsohn and Tsiatis, 1997). Some early Bayesian work on this topic includes those by Faucett and Thomas (1996), and Brown and Ibrahim (2003). More recently, there have been several extensions in the directions of dynamic predictions and prognostic tools (Proust-Lima and Taylor, 2009; Taylor et al., 2013; Rizopoulos et al., 2014), competing risks (Elashoff et al., 2007; Hu et al., 2009), recurrent events (Liu and Huang, 2009), multiple longitudinal variables (Li et al., 2007), cure rates (Yu et al., 2004), and diagnostics (Huang et al., 2009). Several comprehensive reviews of joint modeling of longitudinal and survival processes have been published (Tsiatis and Davidian, 2004; Mccrink et al., 2013; Gould et al., 2014; Papageorgiou et al., 2019).

Joint models are useful in a few different scenarios. The main interest can be in the longitudinal process with an event causing informative dropout. Interest could be equally on a longitudinal and a survival process that are associated. A third common scenario occurs when a time-to-event

process is modeled with a longitudinal covariate measured intermittently and with error. Such data require joint modeling to fully capture the association and reduce bias.

We will begin by describing the basic framework of a joint model in Section 1.2 and usual estimation techniques in Section 1.3. Section 1.5 describes our motivating dataset from adrenal cancer patients at the University of Michigan.

1.1 Notation

Throughout this work we will use the following notation unless otherwise specified. I_m denotes an $m \times m$ identity matrix and 0_m denotes a vector of zeros of length m. $Ind(\cdot)$ is the indicator function. A superscript T denotes the transpose of a vector or matrix. The function $\log(x)$ is the natural logarithm of x and may be used interchangeably with $\ln(x)$. The symbol $\propto$ stands for "is proportional to". We will use the phrases survival time, event time, and failure time interchangeably. Also we will use cause, death, event, and failure interchangeably in regards to the time-to-event endpoint. $Pr(A)$ will denote the probability of event A. We use $f(\cdot)$ for the probability density function of a random variable, $F(\cdot)$ the cumulative distribution function, $S(\cdot)$ the survival function where $S(\cdot) = 1 - F(\cdot)$, $h(\cdot)$ the hazard function, and $H(\cdot)$ the cumulative hazard function.

1.2 Modeling Framework

We will begin by describing common joint models with longitudinal measurements and a single survival outcome. Consider $i = 1, ..., N$ subjects with repeated longitudinal measurements and a terminal event measured. The repeated measurements and event time are assumed to be associated. Each subject i has J_i measurements taken at different times with the intervals between measurements possibly differing. The total number of measurements is $n = \sum_{i=1}^{N} J_i$.

1.2.1 Longitudinal Submodel

Let $Y_i(t)$ denote the longitudinal process for subject i at time t. We assume there is a true underlying process, or trajectory, $m_i(t)$ from which $Y_i(t)$ is measured with error $e_i(t)$, $Y_i(t) = m_i(t) + e_i(t)$. In practice, we do not observe the longitudinal process at all times. Instead for subject i we observe $Y_i(t)$ at n_i times $(t_{i,1}, ..., t_{i,J_i})$. Let $Y_i = (Y_{i,1}, ..., Y_{i,J_i}) = (Y_i(t_{i,1}), ..., Y_i(t_{i,J_i}))$ be the vector of observations. The trajectory $m_i(t)$ is modeled with fixed (possibly time-dependent) covariates, $X_{i,j}$ and parameter coefficients β, and random effects vector U_i with possibly time-dependent design matrix $Z_{i,j}$. The random effects are traditionally assumed to follow a Normal distribution. A typical longitudinal submodel is shown in (1.1).

$$Y_{i,j} = m_i(t_{i,j}) + e_i(t_{i,j}) = X_{i,j}\beta + Z_{i,j}U_i + e_{i,j},$$
$$\text{with } e_i = (e_{i,1}, e_{i,2}, \ldots, e_{i,J_i})^T \sim N(0_{J_i}, \sigma^2 I_{J_i}), \tag{1.1}$$
$$U_i \sim N(0, \Sigma_U), \quad i = 1, ..., N, j = 1, ..., J_i;$$

where σ^2 and Σ_U are the dispersion parameters for the error and the random effects, respectively.

1.2.2 Time-to-Event Submodel

Let T_i^* denote the event time for subject i, C_i the censoring time, $T_i = \min\{T_i^*, C_i\}$ the observed time, and $D_i = Ind(T_i^* \leq C_i)$ the event indicator. A common time-to-event submodel is a relative risk model of the form (1.2).

$$h_i(t|M_i(t)) = h_0(t)\exp\{W_i\gamma + \alpha m_i(t)\} \tag{1.2}$$

where $M_i(t)$ is the history of the longitudinal process up to t, $M_i(t) = \{m_i(s), 0 \leq s < t\}$. The matrix of fixed covariates, W_i, can include the same variables as X_{ij} in the longitudinal submodel but they do not necessarily overlap. The trajectory $m_i(t)$ from the longitudinal model is included to link the two processes and α measures the strength of this association. While proportional hazards

models are common, accelerated failure time and other survival models have been implemented (Mccrink et al., 2013; Tseng et al., 2005)(Rizopoulos, 2012, p.137).

The model form in (1.2) will be referred to as the current value form since the association is through the current value of the longitudinal trajectory, $m_i(t)$, on the right-hand side of the equation. This form is most often used when the survival time is of interest and the longitudinal process is thought of as a time-dependent covariate measured with error. In (1.2), $h_i(t)$ is the event hazard and $h_0(t)$ is the baseline hazard. In ordinary Cox regression $h_0(t)$ is left unspecified, avoiding restrictions that come from specifying a parametric form for the baseline hazard (Yuen and Mackinnon, 2016). However if an unspecified hazard is applied in a joint model, it has been shown that standard errors of the parameter estimates can be underestimated (Hseih et al., 2006). This can be remedied by estimating the standard errors with an additional method such as bootstrapping (Rizopoulos, 2010; Yuen and Mackinnon, 2016). To avoid this issue, parametric but flexible functions are often used for $h_0(t)$, such as piecewise constant or spline models (Rizopoulos, 2012, p.53). These flexible functions can sufficiently approximate the baseline hazard and has been noted as the preferred choice for $h_0(t)$ by some (Rizopoulos, 2010; Yuen and Mackinnon, 2016)(Rizopoulos, 2012, p.53).

Alternatively, the submodels can be linked through a shared parameters model. This is often used when the longitudinal process is the main interest with informative censoring or when the focus is on both processes equally. Assuming a longitudinal submodel as in (1.1), a survival submodel as shown in (1.3) is common

$$h_i(t) = h_0(t)\text{exp}\{W_i\gamma + \alpha Z_{2i}(t)U_i\} \qquad (1.3)$$

where U_i is the same vector of random effects as in (1.1) and α again measures the association. A typical example is a random coefficients model where $Z_{2i}(t)U_i = U_{0i} + U_{1i}t$ with U_{0i} and U_{1i} correlated, often multivariate normal (Wulfsohn and Tsiatis, 1997). The association can be generalized so that $\alpha Z_{2i}(t)U_i = \alpha_0 U_{0i} + \alpha_1 U_{1i}t$ and α_0 need not equal α_1 (Mccrink et al., 2013).

5

1.3   Estimation

1.3.1   Frequentist Estimation

It is commonly assumed that conditionally given the random effects, the longitudinal and time-to-event outcomes are independent and so are the longitudinal measurements taken on a single subject. Denoting the set of all parameters by $\Omega = (\beta, \gamma, \alpha)$, the log-likelihood contribution from subject i is

$$l_i(\Omega) = log \int \left(\prod_{j=1}^{J_i} f(y_{i,j}|U_i; \Omega) \right) h_i(T_i|U_i; \Omega)^{D_i} S(T_i|U_i; \Omega) f(U_i; \Omega) dU_i,$$

where $f(\cdot)$ denotes the density function and $S(\cdot)$ denotes the survival function. There is generally no closed form solution to the likelihood equations. Numerical integration and optimization techniques such as an EM algorithm treating the random effects as missing data, are used in practice. Due to high-dimensional integration and potential correlation induced by the random effects, the process of convergence can be slow. Other methods include Newton-type or hybrid-EM and quasi-Newton algorithms (Rizopoulos, 2012, p.64)(Henderson et al., 2000; Hseih et al., 2006; Tsiatis and Davidian, 2004; Wulfsohn and Tsiatis, 1997; Yu et al., 2004).

1.3.2   Bayesian Estimation

Joint models from a Bayesian perspective have also been implemented. The full posterior distribution conforms to the structure

$$f(\Omega, U|T, D, y) \propto \prod_{i=1}^{n} \left(f(y_i|U_i; \Omega) f(T_i, D_i|U_i; \Omega) f(U_i; \Omega) \right) \pi(\Omega),$$

where $\pi(\Omega)$ denotes the joint prior for the model parameters. MCMC techniques such as Gibbs Sampling or Metropolis-Hastings algorithm can be used for inference (Bekele and Shen, 2005; Faucett and Thomas, 1996; Gould et al., 2014; Henderson et al., 2000; Rizopoulos and Ghosh,

2011; Tsiatis and Davidian, 2004; Yu et al., 2004).

Bayesian estimation can have faster convergence rates than maximum likelihood methods (Mc-crink et al., 2013). Bayesian models also make prediction fairly straightforward using the posterior predictive distribution (Sweeting and Thompson, 2011). A drawback is needing to specify priors which may influence the estimation. Because of this sensitivity analyses are recommended. On the other hand, Bayesian models avoid asymptotic approximations and can incorporate historical data (Gould et al., 2014).

1.4   Some Extensions in Joint Modeling

Chapter 2 will focus on joint models with a single longitudinal and a single survival event but joint models have been extended in many ways. In this work we will first consider extensions in the survival submodel. Chapters 3 and 4 will examine joint models with competing risks outcomes. In this situation the event can have one of multiple possible causes. A traditional way to formulate such data is through latent risks. Assuming there are $K \geq 1$ possible causes of failure (or risks), we can assume that each subject $i = 1, .., N$ theoretically has an event time for each risk $T_{i1}^*, ..., T_{iK}^*$. Subjects can still be independently censored at time T_{i0}. Dependent censoring can be included as one of the K risks. The observed data in this case is the minimum event time $T_i = \min(T_{i0}, T_{i1}^*, ..., T_{iK}^*)$ and the cause of the event $D_i = k$ if $T_{ik} \leq T_{il}$ for all $l = 0, 1, ..., K$.

Chapter 5 will further extend the survival submodel to consider multi-state data of which competing risks is one subset. In multi-state data we assume that there are $R \geq 1$ possible states and a subject can transition between these states over time. In this case we have something like a competing risks outcome at each transition time and subjects can have multiple transition times. Additionally in Chapter 5 we will also extend the longitudinal submodel to include more than one longitudinal outcome.

1.5   Motivating Study on Adrenocortical Carcinoma

We are motivated by a study of patients with adrenocortical cancer at the University of Michigan Rogel Cancer Center. Between 1983 and 2011 (inclusive) there were 176 people diagnosed with adrenocortical carcinoma (ACC). We will also use the phrase adrenal cancer to refer to ACC in this work. Patients were repeatedly subject to CT scans for monitoring their disease (McDuffie and Aufforth, 2016). The scans were used to determine the state of disease at each scan. To do this images of the patient's tumor were compared to those in the last scan. An example of the timeline with possible states is shown in Figure 1.1. Each patient has an initial scan, labeled Scan 0. The state of their disease at that scan is considered the reference state. Between that scan and the next scan the disease could change. At the next scan it is determined whether the tumor did not significantly change in size (stable), became bigger (progressed), or became smaller (regressed). At that point, the latest disease state is now considered the reference state for the next scan. This continues for all the patient's scans. The diagram in Figure 1.1 shows two scans before the end of follow-up, but this could be any number of scans. We also have information on a terminal event, namely death. At the end of follow-up the patient may have died; otherwise the patient would be censored.
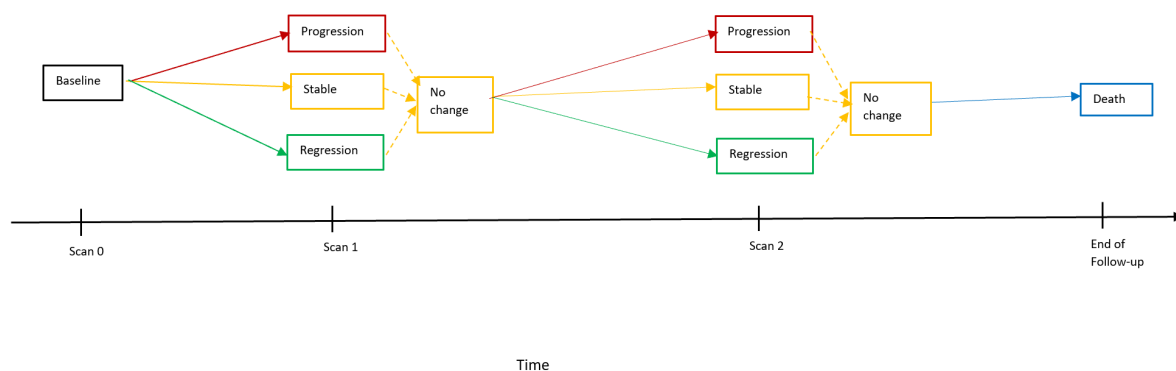
Figure 1.1: Diagram of possible states over time in the adrenal cancer study.

Baseline covariates available are age at diagnosis, sex, cancer stage, tumor grade, operative

Variable		Mean	Std. dev.	N (%) Missing
Age at diagnosis		45.8	13.2	0
Scan Time (years)		3.3	3.7	0
		Count	**%**	**N (%) Missing**
Sex	Male	72	44	11 (6)
	Female	93	56	
Stage	1	4	2	5 (3)
	2	74	43	
	3	53	31	
	4	40	23	
Tumor Grade	Low	84	48	0
	High	63	36	
	Unknown	29	16	
Operative resection	Open	108	74	31 (18)
	Laparoscopic	37	26	
Margin positive	Yes	26	62	30 (17)
	No	91	18	
	Unknown	29	20	
Vascular invasion	Yes	87	49	0
	No	35	20	
	No comment	54	31	
Chemo at scan	Yes	42	10	716 (63)
	No	217	51	
	Mitotane	132	31	
	Mitotane & chemo	29	7	
	unknown	4	1	

Table 1.1: Description of demographic and clinical information in adrenal cancer study data.

resection, margin positive, and vascular invasion. Information on if the patient was receiving chemo at the time of scan is available for 37% of the scans. Demographic and clinical information included in the data are summarized in Table 1.1.

From the scans markers of body composition, called morphomics, were also measured (Bayar et al., 2017). Morphomics data reflect a patient's body composition in terms of adipose tissue as well as bones and organs which can be closely linked to nutritional status. Morphomic variables are more refined and informative than the commonly used BMI. These measurements could have clinical significance such as in improving the dosing of cytotoxic drugs (Bayar et al., 2017). Ad-

Variable	N Missing	% Missing
visceral fat area	240	21
body depth	240	21
fascia depth	243	21
fascia area	243	21
central visceral depth	243	21
vb2 front skin	240	21
central back fat depth	718	63
central sub-cutaneous depth	238	21
total psoas perimeter area	244	21
average psoas perimeter mean	244	21
lean psoas muscle density	243	21
lean psoas muscle area	243	21

Table 1.2: List of morphomic measures in adrenal cancer study.

ditional studies have found morphomic data to be useful in researching other malignancies such as pancreatic cancer (Balentine et al., 2010), oropharyngeal cancer (Wang et al., 2016), melanoma (Sabel et al., 2015), hepatocellular cancer (Singal et al., 2016), colon cancer (Sabel et al., 2013), renal cell cancer (Xiao et al., 2018) and other conditions including kidney disease (Locke et al., 2017) and Crohn's disease (Stidham et al., 2015).

Twelve time-dependent morphomic variables were recorded at each scan. See Table 1.2 for the list of longitudinal measurement variables available and see Appendix A for a description of these morphomic variables (Holcombe et al., 2016). There are between 1 and 47 scans for each patient with a mean of 6.5 and median of 4 scans. The data includes 1140 longitudinal measurements in total. For 240 of the scans, the morphomic data is not available as the scans could not be processed.

Time was measured from the date of adrenal cancer diagnosis. Of the patients, 98 (56%) died and the other 78 (44%) were censored. Time until death or censoring fell between 0.1 and 24.5 years with a median of 2.5 years. There are 117 (66%) with at least one progression. Time to first progression among those patients was between 0.05 and 14.5 and years with a median of 0.8 years. For regression, 55 (31%) had at least one regression and time to regression among those patients

Variable		Mean	Std. dev.
Number of Progressions		2.5	4.1
Number of Regressions		0.5	1.0
Number of Scan Events (Pro- or Re-gressions)		3.0	4.6
		Count	**%**
Died	Yes	98	56
	No (Censored)	78	44
First Event	Progression	96	55
	Regression	37	21
	Death	3	2
	Censored	40	23
Died without Progression		5	3
Censored without Progression		54	31
Died without Progression or Re-gressions		3	2
Censored without Progression or Regression		40	23
		Percent	
Had $\geq$ 1 Progression		66	
% scans found Progression		37	
Had $\geq$ 1 Regression		31	
% scans found Regression		6	
Had $\geq$ 1 Event		76	
% scan have an Event		42	
		Median	**Range**
Time to first Progression (if had progression)		0.8	(0.05, 14.5)
Time to first Regression (if had re-gression)		0.5	(0.03, 8.7)
Time to First Event (any)		0.8	(0.03, 24.5)
Time to Death without Progression		1.5	(0.2, 3.5)
Time to Death without Progression or Regression		1.5	(0.2, 2.2)

Table 1.3: Description of disease state and survival information in adrenal cancer study data.

was between 0.03 and 8.7 years with a median of 0.5.

We also calculated time to first event of any type. There were 4 possible first event types: (1) progression, (2) regression, (3) death, or (4) censoring. If the first event was of type 3 or 4 the patient had no progression or regression during the entire follow up period. Time to first event was between 0.03 and 24.5 years with a median of 0.8 years. Survival and disease state information is summarized in Table 1.3. A few examples of patient data are shown in Figure 1.2.

The primary research question is whether the morphomic variables are associated with prognosis. A previous study separately investigated the relationship between several morphomic variables and recurrence-free and overall survival in ACC (Miller et al., 2012). Specifically, they looked at
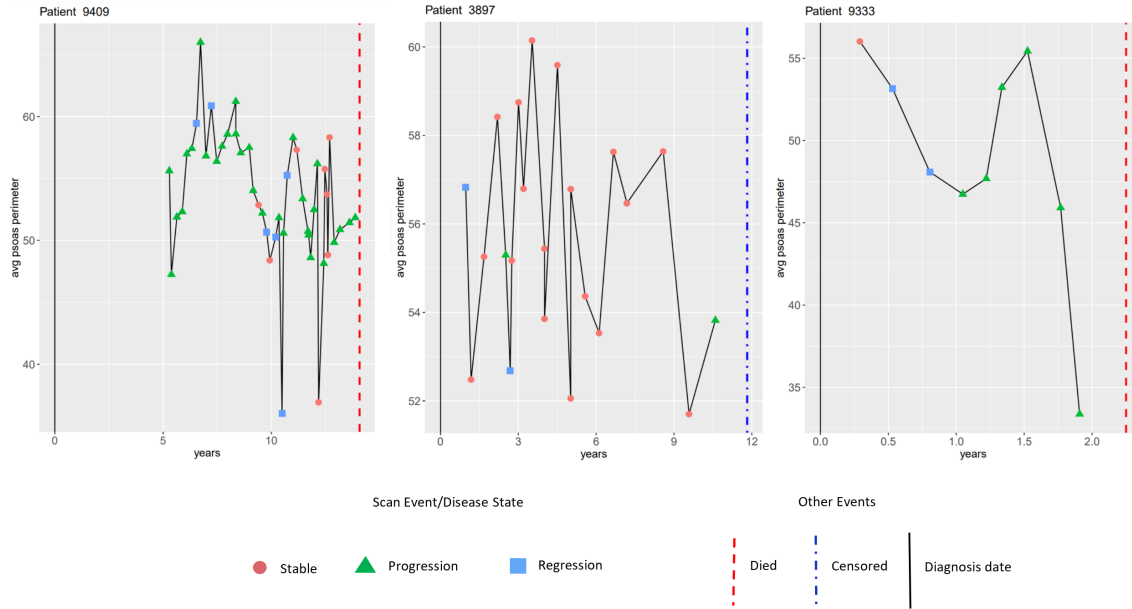
Figure 1.2: Example plots of ACC data.

the variables psoas muscle density (PMD), lean psoas muscle area (LPMA), lumbar skeletal muscle index (LSMI), intra-abdominal (IA) fat, and subcutaneous (SC) fat. The study found that PMD, LPMA and IA fat had significant associations with the survival outcomes.

Since the morphomic variables are measured from the scans, they will be subject to measurement error. Therefore, a joint model for the longitudinal morphomic data and the survival data is needed. We begin in the simplest case in Chapter 2 and investigate the relationship between a single morphomic variable, psoas density, and survival time using joint models within various computational settings. In Chapters 3 and 4 we consider how the longitudinal and survival processes are related when we incorporate the information on cancer progression and regression. Specifically, we treat the survival process as a competing risks problem. We investigate the relationship between a single morphomic variable lean psoas muscle area (LPMA) and time to either first progression or first regression. Finally in Chapter 5 we will utilize more of the available of information via a multistate model. The data include recurrent events, from the declaration of disease state at each scan, and these events can be one of multiple types (progression, regression, or no event). We also have the terminal event of death. Further, we have multiple morphomic variables and we would

like to include more than one in our joint multistate model to examine the relationship between the morphomics and disease state, while accounting for all aspects of the data. We will do this with a joint model with two longitudinal outcomes and multistate survival data.

Chapter 2

Comparison of Computational Approaches for Joint Longitudinal and Time-to-Event Models

2.1 Introduction

With multiple submodels and shared random effects, joint models can become complicated and difficult to utilize. This motivates the need for creating an efficient computational platform to fit these models. Due to the complexity of joint models, implementation can be slow, but can still be useful for those wanting to fit relatively simple joint models for data analysis. Major statistical software such as R, SAS, and Stata include joint modeling functions. The purpose of this chapter is to provide a comprehensive review of associated implementation issues and explore new applications in related software platforms. There has been limited review of software implementation in the joint modeling literature. Some early reviews listed software available at the time (Gould et al., 2014; Mccrink et al., 2013), utilized a single software for data analysis (Rizopoulos, 2012) or compared a couple in the context of analyzing a dataset (Mccrink et al., 2013). Other reviews have compared selected softwares, such as WinBUGS and SAS PROC NLMIXED (Guo and Carlin, 2004). Documentation of the %JM macro in SAS compares the available features of the macro to the JM package in R (Garcia-Hernandez and Rizopoulos, 2015). A review of joint modeling literature by (Sudell et al., 2016) discussed the frequency with which each joint modeling software

was used. More recently, Yuen and Mackinnon (2016) compare some of the available software with an application to a dataset documenting time to psychosis transition.

There are several ways in which this work differs from those in the existing literature. Most of the reviews focus primarily on the survival component of the joint analysis. By contrast, we provide findings from simulation that assess simultaneous performance of the time-to-event and the longitudinal model. Our documentation is also more comprehensive than most reviews covering specialized topics such as latent classes, competing risks, multiple longitudinal outcomes and more. Through extensive simulation, we present a comprehensive appraisal of the different implementations. In addition, the different software are contrasted by means of their performance when used to fit to data from an ongoing trial on adrenal cancer patients enrolled in the University of Michigan Rogel Cancer Center.

This work has been published in (Furgal et al., 2019). In Section 1.2, we will demonstrate the software capabilities with an example data analysis. We will discuss software implementation in Section 2.2. This includes the `JM` and `joineR` packages in `R`, the `%JM` macro in SAS, and the `stjm` command in Stata. In Section 2.4 we describe our simulations. Section 2.5 outlines implementation of a Bayesian joint model by means of the `JMbayes` package in `R`. Other models such as competing risks joint models and some specialized software functions are introduced in Section 2.6 and further described in Appendix C. In Section 2.7 we conclude with a discussion of our findings.

In this chapter we will use the following symbolic notation for distributions. Unless otherwise noted $N(\mu, \sigma^2)$ denotes a (univariate) Normal distribution with mean μ and variance σ^2 while a multivariate Normal distribution with mean vector μ and covariance matrix Σ will be denoted $MVN(\mu, \Sigma)$. $\log N(\mu, \sigma^2)$ is a log-Normal model with mean μ and variance σ^2. $Weibull(\gamma, \mu)$ is a Weibull distribution with shape γ and scale μ. We will use $\sim \Gamma(a, b)$ to denote a parameter follows a Gamma distribution with parameters a and b. $Wishart(B, a)$ is a Wishart distribution with a degrees of freedom and variance matrix B. $\phi(x)$ and $\Phi(x)$ are the probability density function and the cumulative distribution function of the standard Normal distribution, respectively.

2.2   Assessment of Implementation Platform

Fitting a joint model can be computationally intensive. As indicated in the data analysis section, there are a few software packages that are designed to fit such models in an efficient manner. However, the class of such models is quite large and the software use different iterative algorithms for estimation and integration. It is thus important to compare and contrast the implementation platforms to assess and appraise the performance of the competing tools. Such assessment is difficult to make on the basis of fits to few specific datasets. We undertake this evaluation with simulations using various statistical software modules available for fitting joint models. In order to lay the groundwork for the simulation, we first need to briefly describe the available software packages. All packages described in this section implement (1.1) and either (1.2) or (1.3) under a frequentist framework.

2.2.1   JM Package in R

The `JM` package in the `R` language was designed for fitting joint models with the `jointModel()` function. A full description of this package and the `jointModel()` function are available in the CRAN documentation (Rizopoulos, 2016a).

The main arguments for the `jointModel()` function include the output from a linear mixed model (from the `R` function `lme()` (Pinheiro et al., 2016)) and the output from a Cox proportional hazard model (usually from the `R` function `coxph()`) (Therneau and Lumley, 2016). Using these separate model fits, `jointModel()` fits a corresponding current-value joint model with submodels having the same covariates and forms as in the separate models and with the additional association added to the survival submodel.

The `jointModel()` function fits three different model forms using the `parameterization` argument. The default is `parameterization = "value"` and fits the current value model in (1.2). For `parameterization = "slope"`, the survival submodel is linked to the longitudi-

Functionality	JM R package	joineR R package	%JM SAS Macro	%JMfit SAS Macro	stjm Stata Command	JMbayes R package
Longitudinal Submodel						
Gaussian linear model	x	x	x	x	x	x
Generalized linear models	x		x			x
Covariance matrix options			x		x	
Survival Submodel						
Relative Risk model	x	x	x	x	x	x
AFT Weibull model	x					
Unspecified baseline hazard	x	x				
Piecewise baseline hazard	x		x	x		
Spline baseline hazard	x		x	x	x	x
Association						
Current value assoc.	x		x		x	x
Current slope assoc.	x		x		x	x
Random intercept assoc.		x	x	x	x	x
Random slope association		x	x	x	x	x
Separate associations		x		x		x
Interactions in associations	x		x	x	x	x
Model fit options						
Lagged effects	x		x			x
Competing risks models	x		x			
Initial value options	x		x	x		x
Stratification in survival submodel	x		x			
Piecewise/Spline hazard customization	x		x	x		x
Estimation options						
EM only	x	x				
Quasi-Newton	x		x	x		
Bayesian MCMC						x
Number of iteration control	x	x	x	x	x	x
Convergence tolerance control	x	x	x	x	x	
Piecewise/Spline knots control	x		x	x		x
Quadrature points control	x	x	x	x	x	
Adaptive Gauss-Hermite quadrature			x	x	x	
Pseudo-adaptive Gauss-Hermite quadrature	x					
Laplace approximation	x		x			
Gauss-Kronrod Rule	x		x		x	
Other options						
AIC or BIC			x	x		x
Plotting	x	x			x	x
Predictions	x					x
Approx SE default	x		x	x	x	
Bootstrap SE option		x	x		x	

Table 2.1: Overview of available functionalities for joint modeling in each software; 'x' means the software in that column has the feature in the corresponding row.

nal process through the slope of the trajectory as in (2.1).

$$h_i(t|M_i(t)) = h_0(t)\exp\{X_{2i}\gamma + \alpha m_i'(t)\} \tag{2.1}$$

The final option is `parameterization = "both"` which fits a model with both the current value and the slope of the trajectory in the survival submodel, namely $h_i(t|M_i(t)) = h_0(t)\exp\{X_{2i}\gamma + \alpha_1 m_i(t) + \alpha_2 m_i'(t)\}$

The `method` argument specifies the form of the baseline hazard, the form of the model, and the method of numerical integration. The available options are: `weibull-PH-aGH`, `piecewise-PH-aGH`, `spline-PH-aGH`, `weibull-AFT-aGH`, `Cox-PH-aGH`, and `ch-Laplace`. The method `ch-Laplace` uses the fully exponential Laplace approximation described by Rizopoulos et al. (2009). All other options follow a similar format. The first word describes the baseline hazard: `weibull` uses a Weibull baseline hazard, `piecewise` a piece-wise constant baseline hazard, `spline` a B-spline approximation, and `Cox` an unspecified baseline risk. Options with `PH` fit a proportional hazards survival submodel. There is one option for an accelerated failure time model using a Weibull baseline (`weibull-AFT-GH`). Methods ending in `aGH` use pseudo-adaptive Gauss-Hermite quadrature for integral approxima- tion, where the quadrature knots are reassigned once after the first iteration (Rizopoulos, 2010). Each method can instead end in `GH` which uses standard Gauss-Hermite quadrature. Adaptive quadrature is generally preferred due to a reduced computational load using fewer quadrature points while still achieving error on the same order of magnitude as with the standard quadra- ture (Yuen and Mackinnon, 2016). Quasi-Newton techniques are used if EM iterations do not achieve convergence quickly. Only EM is used with the unspecified baseline hazard. Stratification is allowed only with `method="spline-PH-aGH"` or `method="spline-PH-GH"`.

The `jointModel()` function allows for some extensions such as a competing risks model with the `CompRisk` argument. Other extensions can be formulated with the `interFact` and `derivForm` options as described in Rizopoulos (2010) and Rizopoulos (2016a).

2.2.2 joineR Package in R

The `joineR` package was created to analyze longitudinal studies, possibly with an event time causing informative censoring. Full description of this package is in the CRAN documentation (Philipson et al., 2012). The `joint()` function fits a joint model and requires data in a specific format that is the output from the function `jointdata()` in the same package. The user supplies the `joint()` function with the data and a formula object specifying the form for each of the longitudinal and survival submodels. This function fits a shared parameter joint model as in (1.1) and (1.3) with an unspecified baseline hazard. This function is not capable of specifying a parametric form for the baseline hazard. The `model` argument determines the shared random effects. A random slope and intercept model is the default, whereas `model="int"` specifies a random intercept only, and `model="quad"` adds a quadratic time effect to the intercept and slope. The default settings fit a common association when there is more than one random effect terms ($\alpha Z_{2i}(t)U_i = \alpha U_{0i} + \alpha U_{1i}t$). A separate association, as discussed in Section 1.2.2, can be implemented using the `sepassoc=TRUE` option ($\alpha Z_{2i}(t)U_i = \alpha_0 U_{0i} + \alpha_1 U_{1i}t$, $\alpha_0 \neq \alpha_1$). This function also uses an EM algorithm for estimation with some options available for control of this approximation. Since not specifying a baseline hazard may lead to underestimated standard errors, the `joineR` package includes a separate function to calculate bootstrap standard errors for the joint model (`jointSE()`).

2.2.3 %JM Macro in SAS

The %JM macro in the SAS language was written to fit several joint models in several possible forms. A full description of the macro was written by Garcia-Hernandez and Rizopoulos (2015). The %JM macro allows the longitudinal data to be fit to varying outcome types. The longitudinal data can conform to a Normal, binomial, or Poisson distribution, corresponding to continuous, categorical, or count outcomes, respectively. These can be specified with the `LongiType` option. The longitudinal model can also fit different random effects with the

`LongiTimeModel` option, including a linear random intercept and slope model and multiple random splines. There are many options for the baseline hazard of the survival submodel using the `EventModel` option, such as exponential, Weibull, piecewise, and several spline options. The survival submodel can include stratification factors using the `EventStrata` argument. The association can be set with the `SharedParam` argument with the options including current value as in (1.2), slope (equivalent to our model (1.3) with $Z_{2i}(t)U_i = U_{0i} + U_{1i}t$), cumulative ($h_i(t|M_i(t)) = h_0(t)\exp\{X_{2i}\gamma + \alpha \int_0^t m_i(s)ds\}$), and "coefficients" in which the user can specify which random effects from the longitudinal submodel should be included in the survival submodel. Multiple associations can be used at the same time. The `SharedCoefficients` and `SharedLongitTerm` arguments can be used to create even more joint model parameterizations. Estimation is carried out by the PROC NLMIXED procedure which is described in Appendix C.1.

2.2.4 stjm Command in Stata

In Stata, joint models can be fit with the `stjm` command. A detailed description is given by (Crowther et al., 2013). A linear mixed model and a proportional hazards model can be fit as the submodels with several association structures available. The association between submodels can be of the current value or the current slope form, similar to the `JM` package in `R`. The association can also be through shared parameters with or without covariates. Estimation is carried out by Newton-Ralphson method and numerical integration is implemented with standard or adaptive Gauss-Hermite quadrature. We found Stata to have more difficulties in fitting models with various parameter values in our simulations.

2.2.5 Comparison of Software Functionality

In summary, all three major statistical softwares `R`, SAS and Stata fit joint models with comparable functionality. A detailed list of available options is provided in Table 2.1. All software will fit a Gaussian longitudinal submodel as in (1.1). Some software will fit a generalized linear model where the longitudinal outcome Y has a non-Gaussian distribution such as binomial or Poisson.

Each software will also fit a standard proportional hazards survival submodel modeling the hazard function as in (1.2) or (1.3). Only `JM` in `R` will fit a survival model in an accelerated failure time (AFT) framework. Available association options are listed including current value (1.2), random intercept or slope (1.3) and separate associations like we saw in section 2.2.2. Model fit options describe different forms of the survival submodels that may be available such as a competing risks model, lagged effects, or stratification. Estimation options include choices for the model fitting algorithm and numerical integration. Finally, some other possibly useful options in the software are listed including built-in AIC or BIC calculations, plotting, predictions, and standard error calculations.

2.3   Application to the Adrenal Cancer Data

We explore the available software platforms through analysis of the adrenocortical carcinoma (ACC) dataset described in Section 1.5. We selected a single scan measurement, psoas density, as the longitudinal response variable. Psoas density is measured in Hounseld Units (HU) using the density of pixels in the scan (Holcombe et al., 2016). A previous study with similar data describes how psoas muscle density and size can be measures of patient frailty, and scan measurements can be associated with survival (Miller et al., 2012). Baseline covariates used were age, cancer stage, and tumor grade. For simplicity we performed a complete case analysis. There were 160 patients with psoas density measurements and all baseline covariates. There are between 1 and 45 scans for each patient with a mean of 5.5 and median of 3 scans. In this group 100 patients died. Time until death or censoring fell between 0.1 and 17.9 years with a median of 2.4 years. Table 2.2 summarizes the relevant variables. Joint models are fit to the data using three major software, namely `R`, SAS, and Stata. Both the current value (1.2) and the shared parameter (1.3) forms are implemented.

First, a current-value joint model was fit to this data using `JM` in `R`, SAS, and Stata. The

Variable		Mean or Count	SD or %
Psoas Density		54.4	8.5
Scan Time (years)		3.1	3.2
Age		46.1	13.6
Stage	1 or 2	71	44.4%
	3 or 4	89	55.6%
Tumor Grade	Low	79	49.4%
	High	58	36.3%
	Unknown	23	14.4%

Table 2.2: Description of adrenal cancer data used in Ch 2.

longitudinal submodel included a random intercept and slope in each software.

$$
\begin{aligned}
\text{PsoasDensity}_{ij} = m_i(t_{ij}) + e_{ij} &= \beta_0 + \beta_1 t_{ij} + \beta_2 \text{Age}_i + \beta_3 \text{Stage}_i \\
&+ \beta_4 \text{TumorGradeHigh}_i + \beta_5 \text{TumorGradeUnknown}_i + U_{0i} + U_{1i} t_{ij} + e_{ij}
\end{aligned}
\tag{2.2}
$$

Survival submodels with flexible baseline hazards were chosen since the true model is unknown. In Stata models with a flexible baseline hazard did not converge, so a specific parametric (Weibull) hazard was used. Specifically, the software fit survival submodels with the form

$$
h_i(t) = h_0(t) \exp(\gamma_1 \text{Age}_i + \gamma_2 \text{Stage}_i + \gamma_3 \text{TumorGradeHigh}_i + \gamma_4 \text{TumorGradeUnknown}_i + \alpha m_i(t))
\tag{2.3}
$$

with baseline hazards

$$
\text{R JM PWC: } h_0(t) \sim \text{Piecewise constant function}
$$

$$
\text{SAS \%JM PWC: } h_0(t) \sim \text{Piecewise constant function}
$$

$$
\text{Stata stjm Weib: } h_0(t) \sim \text{Weibull}
$$

For SAS and `R JM`, the program defaults were used to create the flexible baseline hazards. Both procedures by default use six equally spaced internal knots to partition the observed event

times. For the SAS model, the random effects are assumed uncorrelated because we found fitting issues with an unstructured matrix (Garcia-Hernandez and Rizopoulos, 2015).

The Two-Stage model in (2.4) was fit for comparison.

$$
\left\{
\begin{aligned}
&\text{First: PsoasDensity}_{ij} = \beta_0 + \beta_1 t_{ij} + \beta_2\text{Age}_i + \beta_3\text{Stage}_i + \beta_4\text{TumorGradeHigh}_i \\
&\qquad + \beta_5\text{TumorGradeUnknown}_i + U_{0i} + U_{1i}t_{ij} + e_{ij}, \\
&\text{Then: } h_i(t) = h_0(t)\exp(\gamma_1\text{Age}_i + \gamma_2\text{Stage}_i \\
&\qquad + \gamma_3\text{TumorGradeHigh}_i + \gamma_4\text{TumorGradeUnknown}_i + \alpha\hat{m}_i(t)), \\
&h_0(t) \text{ unspecified}
\end{aligned}
\right. \tag{2.4}
$$

Bootstrapping was used to estimate the coefficients and bias-corrected bootstrapped confidence intervals (Efron and Tibshirani, 1993), as shown in Table 2.3.

We also fit a shared parameter model as in (1.3) with `joineR`, SAS, and Stata. The longitudinal submodel included a random intercept in `joineR` and Stata, while the default in SAS included a random slope. The model fit is

$$
\begin{aligned}
\text{PsoasDensity}_{ij} &= \beta_0 + \beta_1 t_{ij} + \beta_2\text{Age}_i + \beta_3\text{Stage}_i + \beta_4\text{TumorGradeHigh}_i \\
&\quad + \beta_5\text{TumorGradeUnknown}_i + U_i + e_{ij} \\
h_i(t) &= h_0(t)\exp(\gamma_1\text{Age}_i + \gamma_2\text{Stage}_i + \gamma_3\text{TumorGradeHigh}_i + \\
&\quad \gamma_4\text{TumorGradeUnknown}_i + \alpha U_i)
\end{aligned}
$$

with baseline hazards

$$
\begin{aligned}
&\text{R joineR: } h_0(t) \text{ Unspecified} \\
&\text{SAS \%JM PWC: } h_0(t) \sim \text{Piecewise constant function} \\
&\text{Stata stjm Weib: } h_0(t) \sim \text{Weibull}
\end{aligned}
$$

The findings are contrasted with a shared parameter two-stage model similar to that in (2.4) but with $\alpha \hat{m}_i(t)$ replaced with $\alpha Z_{2i}(t) U_i = \alpha_0 U_i$ with scalar U_i.

Comparing Tables 2.3 and 2.4, the results are similar for the current-value and shared parameters models. Nearly all joint models estimate the coefficients of the longitudinal submodel very similarly. Time has a significant negative value showing that psoas density tends to decrease over time, which fits with previous knowledge (Miller et al., 2012). Older patients tend to have lower psoas density. Stage 3 or 4 and High Tumor Grade tend to decrease the average psoas density compared to lower stages and grade, but these are not significant in any software or models. There are more differences in the results of the survival submodel since each software fits the survival model differently. The effects are estimated to be larger in the Two-Stage current-value models than in the true joint models. Age does not have much of an effect on death hazard. Most software and model forms find higher stage to slightly increase the death hazard, but significance depends on the software. There is a small and significant positive coefficient for High Tumor Stage. Finally, each software except the current-value Two-Stage model found a small but significant negative association with psoas density. Log-likelihoods are shown in Table 2.5 and are similar for all software and both model forms.

2.4 Simulations

The joint modeling software were compared through simulation. Data were generated under three scenarios, namely a random intercept only and a random intercept and slope model in the current-value form, and a random intercept shared-parameters model. Each scenario includes $N = 500$ subjects and 100 simulated datasets. Data were generated in R version 3.2.2 and simulations were run in Windows 7 on a 3.2 GHz Intel Core i5 processor.

Variable	Two-Stage*	JM PWC	SAS %JM PWC	Stata stjm Weib
N	89	87	100	100
Intercept	64.4 (59.5,68.2)	64.6 (59.2,68.3)	65.4 (59.8,69.7)	64.6 (60.6,68.6)
Time	-0.5 (-1.4,-0.1)	-0.6 (-1.7,-0.2)	-0.9 (-2.7,-0.4)	-0.6 (-1.9,-0.3)
Age	-0.2 (-0.3,-0.1)	-0.2 (-0.3,-0.1)	-0.2 (-0.3,-0.1)	-0.2 (-0.4,-0.2)
Stage 3 or 4	-0.6 (-1.7,0.2)	-0.6 (-1.7,0.3)	-0.6 (-1.8,0.05)	-0.6 (-2.2,-0.1)
Tumor Grade: High	-0.6 (-3.8,2.2)	-0.7 (-4.1,2.1)	-0.7 (-4.1,2.1)	-0.7 (-3.6,2.1)
Tumor Grade: Unknown	-2.9 (-7.5,0.3)	-3.2 (-8.4,-0.6)	-3.1 (-8.0,0.2)	-3.1 (-8.5,-0.4)
Surv-Age	0.1 (-0.009,0.6)	0.002 (-0.01,0.06)	-0.009 (-0.03,0.008)	-0.01 (-0.03,0.001)
Surv-Stage 3 or 4	0.7 (-0.06,3.4)	0.2 (0.02,0.3)	0.2 (-0.01,0.3)	0.2 (-0.06,0.4)
Surv-Tumor Grade: High	1.3 (-3.6,4.2)	0.6 (0.004,1.0)	0.5 (0.06,1.1)	0.4 (-0.04,1.1)
Surv-Tumor Grade: Unknown	3.4 (0.4,12.0)	1.0 (0.3,1.5)	0.9 (0.2,1.7)	1.0 (0.2,1.9)
Association	0.8 (0.1,3.0)	-0.07 (-0.1,-0.04)	-0.1 (-0.2,-0.06)	-0.09 (-0.1,-0.04)

Table 2.3: Current-value joint model parameter estimates and bias-corrected bootstrapped confidence intervals for the adrenal data. N is the number of models in 100 bootstraps that successfully converged.
*Two-Stage intervals are the 2.5 and 97.5 quantiles of the bootstrapped values, not bias-corrected.

Variable	Two-Stage*	joineR	SAS %JM PWC	Stata stjm Weib
N	100	100	100	100
Intercept	63.7 (59.5,67.7)	63.1 (58.3,67.1)	64.5 (60.0,68.5)	64.4 (59.5,68.1)
Time	-0.3 (-0.6,-0.08)	-0.4 (-0.7,-0.2)	-0.6 (-1.4,-0.3)	-0.6 (-1.2,-0.3)
Age	-0.2 (-0.3,-0.1)	-0.2 (-0.3,-0.1)	-0.2 (-0.3,-0.1)	-0.2 (-0.3,-0.09)
Stage 3 or 4	-0.6 (-1.7,0.1)	-0.7 (-1.6,0.1)	-0.6 (-1.7,0.09)	-0.6 (-2.3,0.3)
Tumor Grade: High	-0.6 (-3.6,2.0)	0.6 (-2.0,3.7)	-0.7 (-3.7,2.1)	-0.6 (-4.0,2.2)
Tumor Grade: Unknown	-3.2 (-9.1,1.6)	-2.9 (-7.7,3.3)	-3.1 (-8.1,-0.09)	-3.2 (-7.5,-0.3)
Surv-Age	0.01 (-0.003,0.03)	0.01 (-0.004,0.02)	0.01 (-0.003,0.03)	0.006 (-0.02,0.02)
Surv-Stage 3 or 4	0.2 (0.06,0.4)	0.2 (0.01,0.4)	0.2 (0.04,0.4)	0.2 (-0.02,0.5)
Surv-Tumor Grade: High	0.5 (-0.01,1.0)	-0.5 (-1.0,0.02)	0.6 (0.04,1.2)	0.4 (-0.04,1.1)
Surv-Tumor Grade: Unknown	1.1 (0.6,1.8)	0.7 (-0.2,1.3)	1.2 (0.4,1.8)	1.3 (0.4,2.1)
Association	-0.09 (-0.1,-0.06)	-0.09 (-0.1,-0.05)	-0.1 (-0.1,-0.06)	-0.09 (-0.1,-0.05)

Table 2.4: Shared parameter joint model parameter estimates and bootstrapped confidence intervals for the adrenal data. N is the number of models in 100 bootstraps that successfully converged.
*Intervals are the 2.5 and 97.5 quantiles of the bootstrapped values, not bias-corrected.

	Current-value	Shared Parameter
JM PWC	-3130.6	NA
joineR	NA	-3392.1
SAS %JM PWC	-3147.5	-3147.0
Stata stjm Weib	-3100.9	-3083.3

Table 2.5: Log-likelihood values for the adrenal data models.

2.4.1 Scenario 1

Data for the current-value association random intercept only joint model, which we will call Scenario 1, were generated as follows. The longitudinal data were generated from the model

$$Y_{ij} = m_i(t_{ij}) + e_{ij} = \beta_0 + \beta_1 t_{ij} + \beta_2 X_{1i} + \beta_3 X_{2i} + U_{0i} + e_{ij} \tag{2.5}$$

We have two covariates in our joint model, a binary group indicator X_1 and a continuous X_2. The random effect U_{0i} is a scalar. Measurement times are between 0 and 3 years. The random effect and measurement error are normally distributed and independent. Survival times T_i^*, were generated from a relative risk model.

$$h_i(t) = \exp(\gamma_0 + \gamma_1 X_{1i} + \gamma_2 X_{2i} + \alpha m_i(t)) \tag{2.6}$$

(See Appendix B.1 for details on generating the survival times). The parameter values used are shown below.

$$
\begin{aligned}
&X_{1i} \sim Bernoulli(0.5), \\
&X_{2i} \sim N(0, 1^2), \\
&(t_{i1}, t_{i2}, ..., t_{i7}) = (0, 0.5, 1, 1.5, 2, 2.5, 3), \\
&e_{ij} \sim N(0, 4^2), \\
&U_{0i} \sim N(0, 2.25^2)
\end{aligned}
\qquad
\begin{aligned}
&\beta_0 = 0.6 \qquad \gamma_0 = -4.4 \\
&\beta_1 = 0.2 \qquad \gamma_1 = 0.07 \\
&\beta_2 = -0.1 \qquad \gamma_2 = 0.2 \\
&\beta_3 = 0.4 \qquad \alpha = 0.5
\end{aligned}
$$

Independent censoring times C_i were drawn from a uniform distribution, $C_i \sim \text{Unif}(0, 3.25)$, and the observed time T_i is the minimum, , $T_i = \min(T_i^*, C_i)$. Any longitudinal measurements Y_{ij} at times after the observed time T_i (where $t_{ij} > T_i$ for subject i) were dropped.

The R JM, joineR, Stata stjm, and Two-Stage software fit (2.5) to the longitudinal part. The SAS macro by default fits a model with a random intercept and slope. The %JM macro has no built-in option for an intercept only association. Therefore we fit a misspecified longitudinal model in SAS, replacing U_{0i} with $U_{0i} + U_{1i}t_{ij}$ in (2.5). An unstructured covariance matrix was used for the SAS PWC model. For the SAS Weibull model, the default diagonal covariance matrix was used to achieve convergence.

For the survival part, a current-value association was used if available. The R JM package, Stata command, and the %JM macro in SAS fit a current-value survival submodel as specified in (2.6). The joineR package only fits joint models with the shared parameters association conforming to the structure in (2.7).

$$h_i(t) = h_0(t) \exp(X_{1i}\gamma_1^* + X_{2i}\gamma_2^* + \alpha U_{0i}) \tag{2.7}$$

where U_{0i} is a scalar. This amounts to a re-parameterization of the current-value model where the coefficients γ_1^* and γ_2^* in the joineR survival submodel converge to combinations of the true coefficients from both submodels, specifically $\gamma_1^* = \gamma_1 + \alpha\beta_2$ and $\gamma_2^* = \gamma_2 + \alpha\beta_3$. For all results, the joineR coefficients will be compared to these combinations. See Appendix B.2 for more on this reparameterization.

The baseline hazards used for these software are in (2.8). Pseudo-adaptive Gauss-Hermite (GH) quadrature is implemented in R JM Weib, PWC and Spl. For the model with an unspecified baseline hazard (R JM Unspec NA) we used the standard (nonadaptive) GH quadrature, since we found using the adaptive algorithm led to poor convergence. In order to investigate whether there is a difference in the nonadaptive (NA) versus the pseudo-adaptive GH quadrature, we also ran models with each of Weibull, piecewise constant, and spline based baseline hazards, using standard GH quadrature, labeled R JM Weib NA, R JM PWC NA, and R JM Spl NA.

$$
\begin{aligned}
&\text{R JM Weib: } h_0(t) \sim \text{Weibull} \\[4pt]
&\text{R JM PWC: } h_0(t) \sim \text{Piecewise constant function} \\[4pt]
&\text{R JM Spl: } \log(h_0(t)) \sim \text{B-spline approximation} \\[4pt]
&\text{R JM Unspec NA: } h_0(t) \text{ unspecified} \\[4pt]
&\text{SAS \%JM Weib: } h_0(t) \sim \text{Weibull} \\[4pt]
&\text{SAS \%JM PWC: } h_0(t) \sim \text{Piecewise constant function} \\[4pt]
&\text{Stata stjm Weib: } h_0(t) \sim \text{Weibull} \\[4pt]
&\text{R joineR: } h_0(t) \text{ unspecified}
\end{aligned}
\tag{2.8}
$$

A Two-Stage model is also fit to compare this more simple technique to the true joint models. The Two-Stage model is fit as follows.

$$
\begin{cases}
\text{First: } Y_{ij} = \beta_0 + \beta_1 t_{ij} + \beta_2 X_{1i} + \beta_3 X_{2i} + U_{0i} + e_{ij}, \\[8pt]
\text{Then: } h_i(t) = h_0(t)\exp(\gamma_1 X_{1i} + \gamma_2 X_{2i} + \alpha \hat{m}_i(t))
\end{cases}
$$

We used the default options for numerical integration and for defining the flexible baseline hazard functions. Specifically, the R JM PWC and SAS %JM PWC models both use a baseline hazard constructed from six equally spaced internal knots that partition the observed event times with the function being constant in between knots. For the R JM Spl model, the B-spline approximation is constructed with five internal knots.

Results are shown in Figure 2.1, plotting the bias of the estimated coefficients compared to the truth. Coverage probabilities are shown in Table 2.6. Table 2.7 shows the bias and MSE for each parameter. Empirical standard deviations are in Table 2.8 and the width of confidence intervals are in Table 2.9. Widths were calculated as the average of confidence interval lengths using a Normal approximation, specifically the average of $2*1.96*$(standard error). Table 2.10 shows the run-times for each software.

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Assoc.
Two-Stage	100	88	94	93	94	**97**	91	93
R JM Weib	93	86	91	92	**96**	**97**	90	**98**
R JM Weib NA	92	86	91	91	**97**	**97**	91	**98**
R JM PWC	100	87	92	93	**96**	**97**	91	**97**
R JM PWC NA	100	86	92	92	**97**	**97**	91	**97**
R JM Spl	93	86	91	92	**96**	**97**	90	**97**
R JM Spl NA	93	85	91	91	**97**	**97**	90	**97**
R JM Unspec NA	100	86	92	92	**96**	92	91	79
SAS % JM Weib*	100	87	94	93	**96**	**97**	91	**97**
SAS %JM PWC*	100	83	91	93	**96**	94	89	92
Stata Weib	33	58	85	82	88	**97**	91	**97**
R joineR**	100	85	94	92	92	**99**	**96**	89

Table 2.6: Coverage probabilities for Scenario 1 (in %).
* The SAS longitudinal model is misspecified, including a random slope not in the data generation model.
** The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.

Software	n models fit	Intercept		Time		X1		X2		Surv X1		Surv X2		Association	
		Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE
Two-Stage	100	-0.24	2.68	-1.14	0.42	0.92	2.98	-0.44	0.84	2.31	21.69	4.11	6.96	-10.64	4.33
R JM Weib	93	-0.06	2.79	0.18	0.43	1.01	3.16	0.02	0.84	0.89	23.04	1.02	8.15	0.46	6.66
R JM Weib NA	92	-0.01	2.82	0.27	0.43	0.86	3.18	-0.15	0.82	0.55	23.17	2.08	7.23	-0.97	4.88
R JM PWC	100	-0.09	2.67	0.02	0.42	0.93	3.01	-0.04	0.83	2.72	22.81	0.22	7.49	0.88	5.76
R JM PWC NA	100	-0.08	2.67	0.02	0.42	0.91	3.01	-0.04	0.83	2.73	22.84	0.18	7.56	0.86	5.75
R JM Spl	93	-0.06	2.80	0.22	0.43	1.01	3.16	0.04	0.84	0.33	23.25	1.09	8.07	0.23	7.12
R JM Spl NA	93	-0.05	2.80	0.21	0.43	0.99	3.16	0.04	0.84	0.36	23.28	1.08	8.06	-0.05	6.48
R JM Unspec NA	100	-0.08	2.67	-0.01	0.42	0.92	3.01	-0.05	0.83	2.58	22.59	0.34	7.55	-0.01	5.65
SAS %JM Weib*	100	0.004	2.66	0.10	0.42	0.91	3.02	0.04	0.83	3.66	24.25	-1.61	11.47	6.73	11.76
SAS %JM PWC*	100	-1.80	2.69	-0.68	0.55	-0.40	2.98	-0.01	0.74	-8.02	23.98	-5.34	12.22	6.10	8.74
Stata Weib	33	-34.74	1.51	-10.99	0.42	5.67	0.89	-25.34	0.33	-3.01	3.07	-14.74	2.86	-32.59	1.89
R joineR**	100	0.59	1.09	-1.30	0.36	-0.15	0.81	-0.98	0.67	-17.46	7.23	-49.41	27.20	-12.00	7.38

Table 2.7: Bias*100 and MSE*100 (Mean Squared Error) of the estimates from Scenario 1.
* The SAS longitudinal model is misspecified, including a random slope not in the data generation model.
** The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.
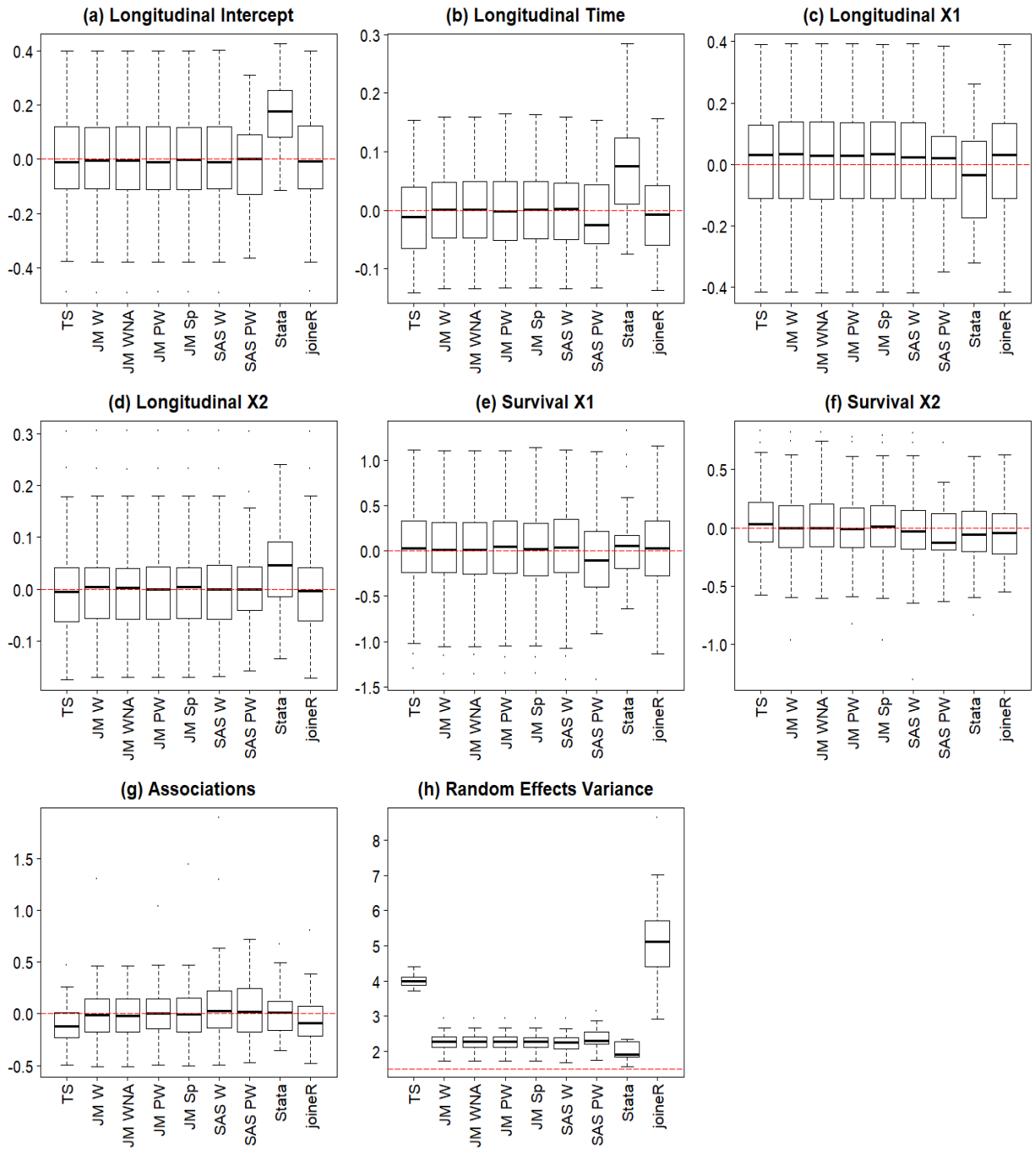
Figure 2.1: Boxplots showing the bias of the Scenario 1 estimates. Model names abbreviated as: TS=Two-Stage, JM W=R JM Weibull, JM PW=R JM Piece-Wise, JM Sp=R JM Spline, JM Un=R JM Unspecified NA, SAS W=SAS Weibull, SAS PW=SAS Piece-Wise. Figure 1(h) plots the actual values estimated for the random effect variance with the dotted line at the true value.

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Association
Two-Stage	100	0.16	0.06	0.17	0.09	0.47	0.26	0.18
R JM Weib	93	0.17	0.07	0.18	0.09	0.48	0.29	0.26
R JM Weib NA	92	0.17	0.07	0.18	0.09	0.48	0.27	0.22
R JM PWC	100	0.16	0.07	0.17	0.09	0.48	0.28	0.24
R JM PWC NA	100	0.16	0.07	0.17	0.09	0.48	0.28	0.24
R JM Spl	93	0.17	0.07	0.18	0.09	0.48	0.29	0.27
R JM Spl NA	93	0.17	0.07	0.18	0.09	0.49	0.29	0.26
R JM Unspec NA	100	0.16	0.07	0.17	0.09	0.48	0.28	0.24
SAS %JM Weib*	100	0.16	0.07	0.17	0.09	0.49	0.30	0.34
SAS %JM PWC*	100	0.16	0.07	0.17	0.09	0.49	0.25	0.29
Stata Weib	33	0.31	0.12	0.11	0.18	0.25	0.18	0.25
R joineR**	100	0.10	0.06	0.09	0.08	0.21	0.17	0.24

Table 2.8: Empirical Standard Deviations of the estimates from Scenario 1.
* The SAS longitudinal model is misspecified, including a random slope not in the data generation model.
** The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Association
Two-Stage	100	0.51	0.25	0.66	0.33	1.84	0.96	0.73
R JM Weib	93	0.51	0.25	0.67	0.33	1.87	0.99	0.87
R JM Weib NA	92	0.51	0.25	0.67	0.33	1.87	0.98	0.85
R JM PWC	100	0.51	0.25	0.66	0.33	1.87	0.99	0.86
R JM PWC NA	100	0.51	0.25	0.66	0.33	1.87	0.99	0.86
R JM Spl	93	0.51	0.25	0.67	0.33	1.87	0.98	0.87
R JM Spl NA	93	0.51	0.25	0.67	0.33	1.87	0.99	0.86
R JM Unspec NA	100	0.51	0.24	0.66	0.33	1.61	0.97	0.56
SAS %JM Weib*	100	0.51	0.25	0.67	0.33	1.90	1.01	0.99
SAS %JM PWC*	100	0.51	0.25	0.67	0.33	1.92	1.03	0.99
Stata Weib	33	0.13	0.11	0.19	0.10	0.61	0.33	0.29
R joineR**	100	0.51	0.24	0.65	0.33	2.01	0.94	0.75

Table 2.9: Average width of confidence intervals from Scenario 1.
* The SAS longitudinal model is misspecified, including a random slope not in the data generation model.
** The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.

	Scenario 1	Scenario 2	Scenario 3
Two-Stage	0.5	1.5	0.1
R JM Weib	3.1	10.9	5.6
R JM Weib NA	5.3	44.5	8.2
R JM PWC	7.8	20.6	11.7
R JM PWC NA	10.5	85.2	15.8
R JM Spl	21.1	28.5	15.4
R JM Spl NA	23.7	104.9	20.6
R JM Unspec NA	6.6	223.8	93.7
SAS %JM Weib	35.6	49.7	10.6
SAS %JM PWC	94.5	392.5	49.3
Stata Weib	1201.3	609.6	661.9
R joineR*	96.9	121.1	48.5

Table 2.10: Average runtime for Scenarios 1, 2, and 3.
* joineR run-times include estimating bootstrapped standard errors with 50 bootstrapped samples.

All software except Stata were able to fit models to a majority of the 100 datasets. Stata was only able to fit around one third of the datasets. Each software had relatively good coverage although Stata did not cover the intercept term well. The intervals around the association parameter from R JM Unspec NA are narrower than the other R JM models. Stata tends to have a larger bias but smaller MSE than the other software leading to narrow confidence intervals. Every software overestimated the random effect variance, with the Two Stage and joineR models being most biased. Despite SAS including an extra random slope term, the random intercept variance estimates were similar to R JM and Stata. Empirical standard deviations were stable across software.

Models with a flexible baseline hazard took longer to run, as would be expected. Stata was exceptionally slow in this scenario. Comparing the models using NA (nonadaptive) quadrature (ex. R JM PWC NA) to the corresponding models using pseudo-adaptive quadrature (ex. R JM PWC) shows that the nonadaptive versions slightly increased the runtime but otherwise choice of numerical integration algorithm made very little difference in the results.

2.4.2 Scenario 2

Our Scenario 2 includes a random intercept and slope joint model, sometimes called the random coefficients model, generated from the following equations.

$$Y_{ij} = m_i(t_{ij}) + e_{ij} = \beta_0 + \beta_1 t_{ij} + \beta_2 X_{1i} + \beta_3 X_{2i} + U_{0i} + U_{1i} t_{ij} + e_{ij} \tag{2.9}$$

$$h_i(t) = \exp(\gamma_0 + \gamma_1 X_{1i} + \gamma_2 X_{2i} + \alpha m_i(t)) \tag{2.10}$$

As in Scenario 1, there are two covariates, one binary and one continuous. The random effects are bivariate Normal. The parameter values used are shown below.

$$X_{1i} \sim Bernoulli(0.5),$$

$$X_{2,i} \sim N(0, 1^2), \qquad \qquad \beta_0 = 1 \qquad \gamma_0 = -4.4$$

$$(t_{i1}, t_{i2}, ..., t_{i7}) = (0, 0.5, 1, 1.5, 2, 2.5, 3), \qquad \beta_1 = 0.2 \qquad \gamma_1 = 0.1$$

$$e_{ij} \sim N(0, 4^2), \qquad \qquad \beta_2 = -0.1 \qquad \gamma_2 = 0.25$$

$$\beta_3 = 0.4 \qquad \alpha = 0.5$$

$$U_i = (U_{0i}, U_{1i})^T \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix} \right)$$

We draw independent censoring times $C_i \sim \text{Unif}(1.25, 3.25)$ and the observed times are $T_i = \min(T_i^*, C_i)$. There was approximately 10% censoring.

JM, joineR in R, %JM in SAS, and stjm in Stata, fits (2.9) to the longitudinal part of the data. The default covariance structure was used in all models. As in Scenario 1, several survival submodels were evaluated. The R JM package, SAS, and Stata fit a current-value survival submodel as in (2.10). The shared-parameter model below was fit with joineR which is a reparameterization of the current-value model similar to Scenario 1.

$$h_i(t) = h_0(t) \exp\left(\gamma_1^* X_{1i} + \gamma_1^* X_{2i} + \alpha(U_{0i} + U_{1i} t) \right)$$

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Assoc.
Two-Stage	100	**98**	87	**97**	89	**95**	94	**97**
R JM Weib	88	**95**	93	**97**	91	93	90	90
R JM Weib NA	95	**98**	**96**	**98**	91	94	93	92
R JM PWC	87	**97**	94	**98**	90	**95**	92	**98**
R JM PWC NA	94	**97**	**96**	**98**	91	**96**	91	**95**
R JM Spl	83	**95**	93	**95**	87	93	92	**95**
R JM Spl NA	94	**97**	**96**	**98**	91	**95**	93	**96**
R JM Unspec NA	85	**96**	**95**	**98**	92	91	92	64
SAS %JM Weib	100	92	91	91	86	88	89	86
SAS %JM PWC	100	**98**	**97**	**97**	91	**95**	92	89
Stata Weib	99	**97**	**97**	**97**	90	94	93	90
R joineR	25	**100**	88	**100**	96	**96**	80	92

Table 2.11: Coverage probabilities for Scenario 2 (in %).
* The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.

The baseline hazards are the same as in (2.8) including models using both adaptive and nonadaptive (NA) in R JM Weib, PWC and Spl. Default settings were used for the piecewise constant and spline functions. Finally, a Two-Stage model was fit for comparison.

Results for Scenario 2 are in Figure 2.2, and Tables 2.11 - 2.14. The Table 2.10 shows the run-times for this scenario. More results including bias, MSE and confidence interval widths are in the Supplemental Material.

The software were able to fit models to most of the 100 simulated datasets except for `joineR`. The `joineR` models could fit only around a quarter of the datasets. Coverage probabilities were similar for all software with one exception. Coverage for the association parameter was noticeably lower for R JM Unspec NA. The confidence interval widths were generally very similar in each software. As in Scenario 1, on average the confidence intervals around the association estimate in R JM Unspec NA were considerably tighter.

In the boxplots, we see that all R JM models and Stata estimated the random effect covariance parameters well. In SAS, the covariance of the random effects was assumed zero according to the

Software	n models worked	Intercept		Time		X1		X2		Surv X1		Surv X2		Association	
		Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE
Two-Stage	100	2.12	0.90	-5.80	0.87	-1.31	1.90	0.18	0.71	-1.17	13.55	-1.77	3.27	1.93	0.91
R JM Weib	88	1.35	0.94	-2.63	0.60	-0.32	1.87	0.34	0.69	1.66	15.11	-2.33	3.77	8.11	2.29
R JM Weib NA	95	0.92	0.88	-1.47	0.56	-0.84	1.83	-0.11	0.70	0.86	15.94	-2.33	3.65	7.56	1.89
R JM PWC	87	1.04	0.84	-1.98	0.53	-1.12	1.78	0.45	0.71	0.30	15.60	-2.86	3.78	9.18	2.10
R JM PWC NA	94	0.80	0.87	-1.15	0.57	-0.77	1.84	-0.31	0.68	1.08	16.40	-2.61	3.70	9.57	2.42
R JM Spl	83	0.52	0.88	-2.42	0.61	-0.72	1.93	0.61	0.76	-0.43	15.41	-1.91	3.88	7.30	1.94
R JM Spl NA	94	0.90	0.87	-1.32	0.57	-0.76	1.85	-0.29	0.67	0.38	15.93	-2.38	3.63	8.31	1.98
R JM Unspec NA	85	0.88	0.87	-1.62	0.59	-0.24	1.95	0.34	0.66	-0.38	15.18	-3.29	3.83	9.74	2.21
SAS %JM Weib	100	-4.88	0.85	-2.85	0.54	-0.67	1.77	-2.45	0.64	-1.28	15.47	-3.12	14.18	5.92	2.23
SAS %JM PWC	100	0.99	0.86	-1.44	0.57	-1.53	1.82	0.27	0.71	1.72	17.54	-3.72	15.14	15.92	6.35
Stata Weib	99	1.23	0.85	-1.78	0.57	-1.21	1.86	0.41	0.70	0.10	4.93	-2.46	3.42	8.06	1.93
R joineR	25	5.70	1.08	-2.39	0.78	-3.54	1.49	1.25	0.60	3.48	10.92	60.70	4.60	-4.67	1.12

Table 2.12: Bias*100 and MSE*100 of estimates from Scenario 2.
*The Surv X1 and Surv X2 estimates are compared to the value to which they converge, as described in Section B.2.

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Association
Two-Stage	100	0.09	0.07	0.14	0.08	0.37	0.18	0.09
R JM Weib	88	0.10	0.07	0.14	0.08	0.39	0.19	0.13
R JM Weib NA	95	0.09	0.07	0.14	0.08	0.40	0.19	0.12
R JM PWC	87	0.09	0.07	0.13	0.08	0.40	0.19	0.11
R JM PWC NA	94	0.09	0.08	0.14	0.08	0.41	0.19	0.12
R JM Spl	83	0.09	0.08	0.14	0.09	0.39	0.20	0.12
R JM Spl NA	94	0.09	0.07	0.14	0.08	0.40	0.19	0.11
R JM Unspec NA	85	0.09	0.08	0.14	0.08	0.39	0.19	0.11
SAS %JM Weib	100	0.11	0.07	0.13	0.08	0.40	0.18	0.12
SAS %JM PWC	100	0.09	0.07	0.13	0.08	0.42	0.19	0.20
Stata Weib	99	0.09	0.07	0.14	0.08	0.40	0.18	0.11
R joineR	25	0.09	0.09	0.12	0.08	0.34	0.22	0.10

Table 2.13: Empirical Standard Deviations of the estimates from Scenario 2.
*The Surv X1 and Surv X2 estimates are compared to the value to which they converge, as described in Section B.2.
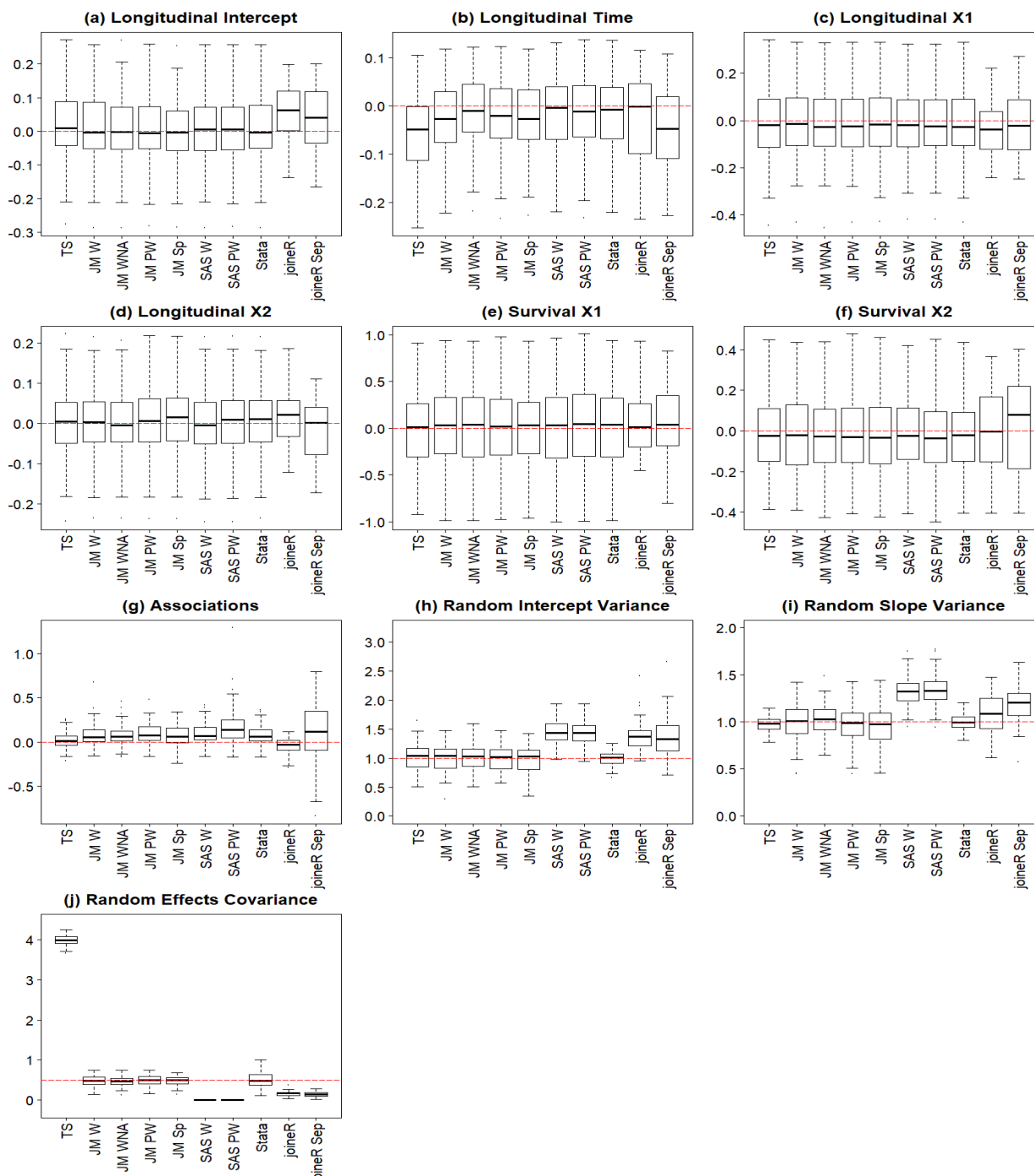
Figure 2.2: Boxplots showing the bias of the Scenario 2 estimates. Model names abbreviated as: TS=Two-Stage, JM W=R JM Weibull, JM PW=R JM Piece-Wise, JM Sp=R JM Spline, JM Un=R JM Unspecified NA, SAS W=SAS Weibull, SAS PW=SAS Piece-Wise. Figures 2(h,i,j) plot the actual values estimated for the random effect variances and covariance with the dotted line at the true value.

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Association
Two-Stage	100	0.44	0.29	0.60	0.30	1.33	0.67	0.40
R JM Weib	88	0.43	0.29	0.59	0.29	1.35	0.68	0.42
R JM Weib NA	95	0.44	0.29	0.60	0.30	1.39	0.70	0.43
R JM PWC	87	0.90	0.28	0.59	0.29	1.39	0.70	0.46
R JM PWC	94	0.44	0.29	0.60	0.30	1.40	0.70	0.46
R JM Spl	83	0.42	0.28	0.56	0.28	1.38	0.70	0.46
R JM Spl NA	94	0.44	0.29	0.60	0.30	1.39	0.70	0.47
R JM Unspec NA	85	0.44	0.29	0.60	0.30	1.23	0.69	0.22
SAS %JM Weib	100	0.45	0.31	0.60	0.30	1.40	0.70	0.46
SAS %JM PWC	100	0.45	0.31	0.60	0.30	1.43	0.72	0.55
Stata Weib	99	0.44	0.29	0.60	0.30	1.39	0.70	0.44
R joineR*	25	0.44	0.29	0.62	0.31	1.66	0.74	0.43

Table 2.14: Average width of Confidence Intervals for Scenario 2.
*The Surv X1 and Surv X2 estimates are compared to the value to which they converge, as described in Section B.2.

macro defaults and we see that the variance of the intercept and the slope were overestimated by SAS. The Two-Stage model estimated the random effect variances well but severely overestimated the covariance, possibly including additional variance in the longitudinal submodel random effects estimates that would be explained by the association with the survival submodel in a joint model. Empirical standard deviations were consistent in all software. We see that the random coefficients models in this scenario took longer to run on average than the models in Scenario 1 for all software except Stata. Using nonadaptive quadrature increased runtime more compared to Scenario 1.

2.4.3 Scenario 3

The final simulation scenario utilized a shared coefficients form as in (1.3), with a random intercept only, i.e. $U_i = U_{0i}$ is a scalar. The data were generated from

$$Y_{ij} = \beta_0 + \beta_1 t_{ij} + \beta_2 X_{1i} + \beta_3 X_{2i} + U_{0i} + e_{ij} \tag{2.11}$$

$$h_i(t) = \exp(\gamma_0 + \gamma_1 X_{1i} + \gamma_2 X_{2i} + \alpha U_{0i}) \tag{2.12}$$

Parameter values are below.

$$X_{1i} \sim Bernoulli(0.5),$$

$$X_{2,i} \sim N(0, 0.5^2),$$

$$(t_{i1}, t_{i2}, ..., t_{i7}) = (0, 0.5, 1, 1.5, 2, 2.5, 3),$$

$$e_{ij} \sim N(0, 4^2),$$

$$U_{0i} \sim N(0, 0.7^2)$$

$\beta_0 = 6$	$\gamma_0 = -4.4$
$\beta_1 = 0.2$	$\gamma_1 = 0.9$
$\beta_2 = -0.1$	$\gamma_2 = 1.2$
$\beta_3 = 0.4$	$\alpha = 0.5$

For this scenario, independent censoring times were drawn $C_i \sim$ Unif$(1.25, 3.25)$. There was approximately 20% censoring.

A random intercept only shared parameter survival submodel was used since the most software could fit the correct model. The R function joineR and Stata fit the same longitudinal submodel in (2.11). Again, the default in SAS is to fit a longitudinal model with both a random intercept and slope as was the case in Scenario 1 which in this case is an misspecified model. The joineR, SAS, and Stata software fit (2.12) to the survival part. The JM package in R only fits joint models with a current-value association, which amounts to a reparameterization of the shared parameter models. In this scenario R JM fits the survival submodel in (2.13). The survival coefficients will converge to a combination of true parameters, namely $\gamma_1^* = \gamma_1 - \alpha\beta_2$ and $\gamma_2^* = \gamma_2 - \alpha\beta_3$. See Appendix B.2 for details.

$$h_i(t) = h_0(t) \exp(\gamma_1^* X_{1i} + \gamma_2^* X_{2i} + \alpha m_i(t)) \tag{2.13}$$

Baseline hazards are listed in (2.8). Both adaptive and nonadaptive (NA) numerical integration was again utilized in R JM with Weibull, piecewise constant, and spline baseline hazard functions. We used the default settings to create piecewise constant and spline baseline hazard functions. A Two-Stage model was also fit with the survival model being a standard semi-parametric Cox model.

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Assoc.
Two-Stage	100	91	94	**98**	93	**97**	92	94
R JM Weib	82	**95**	93	**99**	91	88	**95**	**95**
R JM Weib NA	80	91	90	**99**	91	**96**	94	**96**
R JM PWC	88	**95**	93	**99**	93	95	**98**	82
R JM PWC	89	94	93	**98**	92	95	**97**	80
R JM Spl	97	94	**95**	**98**	92	**97**	**95**	87
R JM Spl NA	100	94	94	**98**	92	**97**	**95**	88
R JM Unspec NA	45	93	89	**100**	89	93	88	11
SAS %JM Weib	100	87	**96**	**96**	94	**100**	**97**	84
SAS %JM PWC	100	88	**96**	**96**	94	**98**	**100**	80
Stata Weib	89	51	85	**98**	91	**99**	**100**	80
R joineR*	100	93	94	**95**	89	80	51	91

Table 2.15: Coverage probabilities for Scenario 3 (in %).
* The SAS longitudinal model is misspecified, including a random slope not in the data generation model.
** The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.

Figure 2.3 and Tables 2.15 - 2.18 show the results for Scenario 3. Table 2.10 shows the runtimes for all scenarios.

Models could be fit to all or almost all of the 100 simulated datasets by each software in this scenario. In this scenario R joineR had a considerably lower coverage probability for the X_2 coefficient in the survival submodel. Nearly all software except R JM Weib (and R JM Weib NA) had relatively low coverage for the association parameter, with the coverage from R JM Unspec NA being especially poor. Again, the average width of confidence intervals around the association estimate from R JM Unspec NA was much smaller than all other software. The average confidence interval widths for X_1 and X_2 from R JM Spl are much larger than the other software, possibly due to poor model fitting on the simulated datasets corresponding to the outliers.

`R JM` and Stata estimated the random intercept variance best although all (except Two Stage) underestimated this value. The Two-Stage model severely overestimated this value as in the first two scenarios. Interestingly fitting models to this data with R JM Spl produced estimates with

		Intercept		Time		X1		X2		Surv X1		Surv X2		Association	
Software	n models worked	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE	Bias	MSE
Two-Stage	100	0.62	1.09	-1.68	0.36	-0.34	0.82	-1.23	0.67	0.17	3.57	0.61	2.85	-6.47	5.18
R JM Weib**	82	0.60	0.91	-0.01	0.34	-0.47	0.73	0.28	0.56	-1.88	4.41	-1.41	3.28	-1.71	5.75
R JM Weib** NA	80	-1.20	1.09	0.95	0.41	0.45	0.72	0.75	0.68	-1.44	4.37	-2.54	3.62	1.59	5.97
R JM PWC**	88	0.43	1.00	-0.73	0.35	0.24	0.80	-0.40	0.64	-5.59	4.21	9.67	4.89	-26.48	13.16
R JM PWC NA**	89	0.68	0.99	-0.93	0.35	0.06	0.74	-0.50	0.66	-5.06	4.18	8.30	4.49	-24.36	12.36
R JM Spl**	97	0.24	1.08	-1.16	0.34	0.34	0.80	-0.40	0.65	-4.36	4.20	11.62	4.93	-28.04	12.67
R JM Spl NA**	100	0.24	1.07	-0.97	0.34	0.15	0.80	-0.66	0.67	-4.39	3.85	11.65	4.82	-28.53	11.96
R JM Unspec NA**	45	0.42	1.13	-1.62	0.41	0.81	0.66	0.29	0.70	-0.60	4.21	9.61	3.20	-21.76	8.00
SAS %JM Weib*	100	-1.43	1.24	-1.22	0.31	0.80	1.07	-2.22	0.82	0.86	3.24	1.85	167.99	3.40	47.32
SAS %JM PWC*	100	-0.61	1.17	-1.37	0.30	0.57	1.05	-2.25	0.83	2.08	3.65	4.48	168.58	5.65	23.61
Stata Weib	89	14.31	2.68	4.65	0.84	-0.18	0.87	-0.26	0.65	-0.74	11.99	-0.24	3.13	0.20	9.05
R joineR	100	0.59	1.09	-1.30	0.36	-0.15	0.81	-0.98	0.67	-22.46	9.22	-29.41	11.43	-12.00	7.38

Table 2.16: Bias*100 and Mean Squared Error (MSE)*100 of the estimates from Scenario 3.
* The SAS longitudinal model is misspecified, including a random slope not in the data generation model.
** The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Association
Two-Stage	100	0.10	0.06	0.09	0.08	0.19	0.17	0.22
R JM Weib**	82	0.10	0.06	0.09	0.08	0.21	0.18	0.24
R JM Weib NA**	80	0.10	0.06	0.09	0.08	0.21	0.19	0.25
R JM PWC**	88	0.10	0.06	0.09	0.08	0.20	0.20	0.25
R JM PWC NA**	89	0.10	0.06	0.09	0.08	0.20	0.20	0.25
R JM Spl**	97	0.10	0.06	0.09	0.08	0.20	0.19	0.22
R JM Spl NA**	100	0.10	0.06	0.09	0.08	0.19	0.19	0.20
R JM Unspec NA**	45	0.11	0.06	0.08	0.08	0.21	0.15	0.18
SAS %JM Weib*	100	0.11	0.05	0.10	0.09	0.18	0.18	0.69
SAS %JM PWC*	100	0.11	0.05	0.10	0.09	0.19	0.15	0.49
Stata Weib	89	0.08	0.08	0.09	0.08	0.19	0.18	0.30
R joineR	100	0.10	0.06	0.09	0.08	0.21	0.17	0.24

Table 2.17: Empirical Standard Deviations of the estimates from Scenario 3.
* The SAS longitudinal model is misspecified, including a random slope not in the data generation model.
** The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.
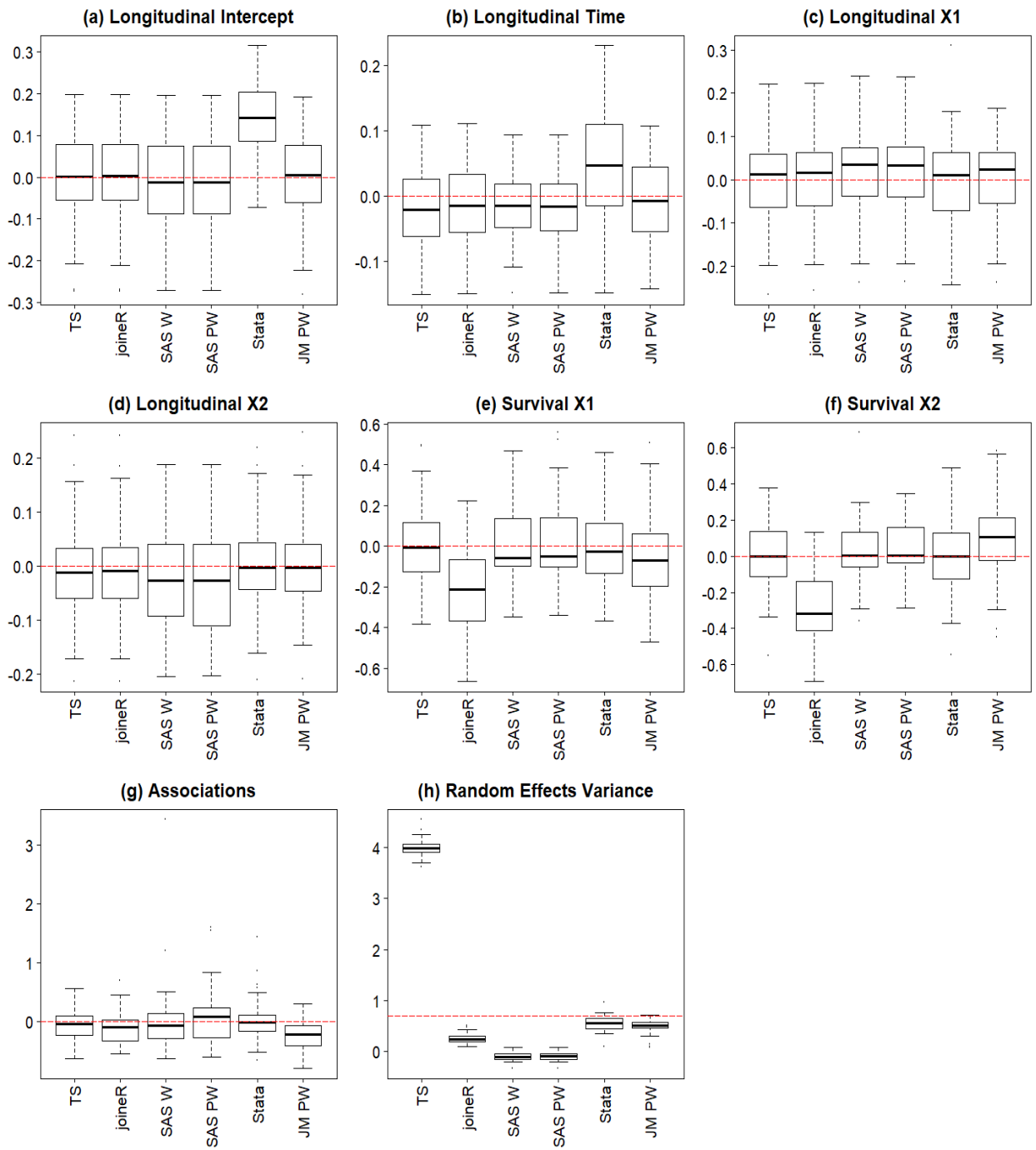
Figure 2.3: Boxplots showing the bias of the Scenario 3 estimates. Model names abbreviated as: TS=Two-Stage, JM W=R JM Weibull, JM PW=R JM Piece-Wise, JM Sp=R JM Spline, JM Un=R JM Unspecified NA, SAS W=SAS Weibull, SAS PW=SAS Piece-Wise. Figure 3(h) plots the actual values estimated for the random effect variance with the dotted line at the true value. Note: Outliers in Intercept and Association from JM Sp have been omitted to aid in interpreting the boxplots.

41

Software	n models	Intercept	Time	X1	X2	Surv X1	Surv X2	Association
Two-Stage	100	0.36	0.21	0.42	0.30	0.81	0.59	0.89
R JM Weib**	82	0.36	0.22	0.41	0.30	0.85	0.69	1.02
R JM Weib NA**	80	0.36	0.22	0.42	0.30	0.86	0.69	1.00
R JM PWC**	88	0.36	0.22	0.42	0.30	0.81	0.69	0.97
R JM PWC NA**	89	0.36	0.22	0.42	0.30	0.83	0.69	0.97
R JM Spl**	97	0.36	0.21	0.41	0.30	4.04	3.56	1.01
R JM Spl NA**	100	0.36	0.21	0.42	0.30	0.83	0.69	0.99
R JM Unspec NA**	45	0.36	0.21	0.41	0.30	0.80	0.60	0.11
SAS %JM Weib*	100	0.35	0.21	0.40	0.28	0.84	0.64	1.32
SAS %JM PWC*	100	0.35	0.21	0.40	0.28	0.82	0.62	1.33
Stata Weib	89	0.29	0.28	0.40	0.29	0.82	0.61	0.91
R joineR	100	0.36	0.20	0.41	0.29	0.84	0.59	1.06

Table 2.18: Average width of Confidence Intervals from Scenario 3.
* The SAS longitudinal model is misspecified, including a random slope not in the data generation model.
** The Surv X1 and Surv X2 estimates are compared to the value to which they converge and standard errors are estimated through bootstrapping. Details in Section B.2.

larger bias for most covariates and considerably smaller MSE for some covariates compared to R JM with the other baseline options. We noted that SAS produced survival submodel X_2 coefficient estimates with an unusually large MSE. Also, joineR generated more biased estimates for the survival submodel despite fitting a model in the same form as the data generating model. As in the first two scenarios, empirical standard deviations were very similar across all software. Lastly, SAS and R joineR ran for notably shorter times than in the first two Scenarios.

2.5   Bayesian Models and Associated Software

The Bayesian implementation was not compared to the simulations for the maximum likelihood methods in the last section. Instead, in this section we will discuss the single option for Bayesian joint modeling in the software and also use this to analyze the adrenal cancer data.

2.5.1 JMbayes Package in R

The `JMbayes` package in `R` was written to fit joint models in a Bayesian framework (Rizopoulos, 2016c). The function for fitting joint models is named `jointModelBayes()`. The arguments for this function are very similar to the `jointModel()` function in the `JM` package in `R`. A linear mixed model is fit to the longitudinal data unless the user specifies a different distribution with the `desLong` argument. A relative risk model is fit for the time-to-event data. The baseline hazard is estimated using splines, either penalized P-Splines (the default) or regression-splines (Rizopoulos, 2016b). The `param` argument determines the form of the association between the submodels. The default association is current value as in (1.2). Other options are association based on the current slope of the longitudinal trajectory analogous to the `R` `JM` package in (2.1), both the current value and slope, or shared parameters like in (1.3). A final association option is a combination of shared random effects and fixed effects, β^*, such as $h_i(t|M_i(t)) = h_0(t)\exp\{X_{2i}\gamma + \alpha^T(\beta^* + U_i)\}$. Available functionalities in the `JMbayes` package are listed in Table 2.1. Extra flexibility is available for the association structure since the user can define any transformation function using the `extraForm` and `transFun` arguments. Using this, the association can be defined to be any function of the current value or any function of the shared random effects.

Estimation is done using Markov Chain Monte Carlo (MCMC) sampling from the posterior conditional distributions of the random effects and the parameters. Usually a random walk Metropolis can be used, but in some cases Metropolis-Hastings or slice sampling are needed (Rizopoulos, 2016c). Initial values can be set using the `init` argument but if left unspecified initial values are taken from the outputs from the separate models that are included as arguments to the `jointModelBayes()` function. Priors can also be specified by the user with the `priors` argument. If not specified, standard prior distributions are used: all the fixed parameters from both submodels as well as the association parameter are given independent diffuse Normal priors, a inverse Wishart prior is assumed for the covariance matrix of the random effects when fitting a Normally distributed longitudinal outcome, and an inverse Gamma prior for the error variance

Variable	Posterior Mean	95% Credible Interval
Intercept	62.0	(57.4,66.5)
Time	-0.5	(-0.9,-0.2)
Age	-0.2	(-0.24,-0.07)
Stage 3 or 4	-0.4	(-1.4,0.5)
Tumor Grade: High	-0.5	(-3.2,2.2)
Tumor Grade: Unknown	-3.2	(-6.7,0.4)
Surv-Age	-0.007	(-0.02,0.01)
Surv-Stage 3 or 4	0.2	(0.01,0.3)
Surv-Tumor Grade: High	0.40	(-0.03,0.9)
Surv-Tumor Grade: Unknown	0.8	(0.2,1.4)
Association	-0.09	(-0.1,-0.06)

Table 2.19: Parameter estimates and credible intervals for the joint models fit to the adrenal data with JMbayes.

(Rizopoulos, 2016c). The `JMbayes` package also includes functions for plotting and running dynamic predictions (Rizopoulos, 2016b).

2.5.2 Application of Bayesian Model to Adrenocortical Carcinoma Data

A joint model with the current value association as in (2.2) and (2.3) with a P-spline baseline hazard is fit to the adrenal cancer data from Section 2.3 using `JMbayes`. MCMC is run for 52,000 iterations with a burn-in of 2000. The results are shown in Table 2.19. The `JMbayes` estimates are generally similar to the frequentist current-value results in Table 2.3 and interpretations of the covariate effects on psoas density and survival are the same.

2.6 Extensions and Specialized Joint Models and their Implementation

The growing interest in joint modeling has led to many extensions, such as joint modeling with competing risks, recurrent events, or multiple longitudinal processes. The addition of many software to fit standard joint models has also encouraged the development of more specialized

software. Here we briefly describe several model extensions and software available for implementation. See Appendix C for more implementations via software. The extent of software implementation for specialized joint modeling is likely to increase in the future.

Up until this point we have focused on longitudinal measurements with a Gaussian distribution. Yet situations often arise that require a non-Gaussian outcome in the longitudinal submodel, such as a logistic or Poisson model. The `JM` and `JMbayes` software in `R` as well as the `%JM` macro in SAS, described in section 2.2, can accommodate non-Gaussian longitudinal processes. The `Jointlcmm()` function in the `lcmm` package in `R` can implement different distributions in the longitudinal submodel of a latent class mixed model (Proust-Lima et al., 2016).

Joint latent class mixed modeling is an extension that can be used to investigate class-specific differences. These models typically include three submodels, a multinomial logistic model to determine the latent class, a class-specific linear (or latent process) mixed model, and a class-specific survival model. The number of latent classes must be set a priori based on knowledge of the situation from which the data were collected. The `lcmm` package in `R` was written for latent class mixed modeling and includes a function, `Jointlcmm()` which fits a joint latent class mixed model for longitudinal and time-to-event data (Proust-Lima et al., 2016). The baseline risk can be common or class-specific. Choice of initial values is important and it is preferred that the user specify initial values with the `B` argument over using the defaults.

In medical studies it is likely more than one biomarker is measured for each patient. This necessitates joint models with multiple longitudinal variables. Hickey et al. (2016) review developments in multivariate joint models including software implementations. Hickey et. al. mentioned a new package for multivariate joint models, `sjmsoft` for `R`, available from the author's website (Brown, 2005). Another option is the `R` package `joineRML` (Hickey et al., 2018).

An interesting but currently less studied extension is joint modeling with competing risks. The longitudinal submodel for a single longitudinal outcome is the same as a single risk joint model. Literature has almost exclusively focused on a survival submodel with proportional cause-specific hazards (Armero et al., 2016; Blanche et al., 2015; Elashoff et al., 2007; Hu et al., 2009; Huang

et al., 2010; Williamson et al., 2008). Usually the survival submodel has the following form, defining the cause specific hazard $h^{(k)}$ for cause $k = 1, ..., K$.

$$
\begin{aligned}
h^{(k)}(t; X_{2i}, U_i, \gamma^{(k)}, \alpha^{(k)}) &= \lim_{h \to 0} h^{-1} P(t \leq T_i < t + h, D_i = k | T_i \geq t, X_{2i}(t), U_i) \\
&= h_0^{(k)}(t) \exp\{X_{2i}(t)\gamma^{(k)} + \alpha^{(k)T} U_i\}
\end{aligned}
$$

The parameters are defined similar to those in Section 1.2.2, X_{2i} are fixed effects which have possibly cause-specific coefficients $\gamma^{(k)}$, U_i are the random effects for subject i with possibly cause-specific association $\alpha^{(k)}$, and cause-specific baseline hazard $h_0^{(k)}(t)$.

Some of the joint modeling software will fit a competing risks model. The `JM` package in R has an option `CompRisk` in the `jointModel()` function. The `joint()` function in the `joineR` package will fit a cause-specific hazard joint model if the event indicator has multiple levels but the documentation states this only works for two causes in addition to censoring, and no more (Philipson et al., 2012). The SAS macro `%JM` can also fit a competing risk model with the option `COMPETING` which is listed under `AdditionalOptions`. Documentation for the `stjm` Stata command states that extension to the competing risks setting is planned but there is no indication that this has been completed (Crowther et al., 2013). There is also no built-in option for fitting a competing risks model in `JMbayes`. Recently a SAS macro called `%SPM` was proposed specifically for fitting joint competing risks models (Wang et al., 2017). Finally, the `Jointlcmm()` function in `lcmm` in R can also handle competing risks (Proust-Lima et al., 2016).

Another common situation is data including recurrent events such as repeated hospitalizations or time between system breakdown in industry. The R package `frailtypack` will fit a standard joint model with function `longiPenal` and also joint models with longitudinal measurements, a terminal event plus recurrent events using the function `trivPenal` (Rondeau et al., 2017, 2012).

With the increasing number of studies utilizing joint modeling, it may be useful to analyze the results from multiple studies in a meta-analytic context. A package for joint modeling in a this context, `joineRmeta` in R, has been developed which can pool model parameters from multiple joint models using standard meta-analysis techniques or analyze the data from all studies

simultaneously (Sudell et al., 2018; Sudell, 2018).

2.7   Discussion

Joint modeling is a growing field of statistical research and the available software encourages the use of these complicated models in applications. We have given an overview of joint modeling methodology and then compiled a comprehensive list of available software. We compared through simulation and data analysis the most common and user-friendly software: `JM`, `joineR`, and `JMbayes` in `R`, `%JM` in SAS, and `stjm` in Stata. We also included a short description of extensions to joint modeling such as those that accommodate competing risks, multiple longitudinal markers or recurrent events. The `%JMfit` macro developed for SAS, described in Appendix C, includes some goodness-of-fit calculations such as decomposition of AIC, BIC, ΔAIC, and ΔBIC (Zhang et al., 2016). Assessing model fit in a joint modeling framework has had limited study and further development of this area is likely in the future (Zhang et al., 2014).

Our simulations show joint modeling software is preferable to Two-Stage models when the longitudinal and survival processes are correlated, as the theoretical work finds. Ours is the first investigation that explored the performance of regression coefficients on both longitudinal and survival components. The packages in `R` fit only one type of association while SAS and Stata include many more options. All software we compared are similar in their performance when fitting the longitudinal submodel in simulations. There are more differences in the survival submodel options and performance. All except `joineR` included a flexible parametric baseline hazard which is most applicable to real data when the true model is unknown. A Cox-type survival submodel using an unspecified baseline hazard is currently available in the `R` packages `joineR` and `JM`. In each of our simulation scenarios we found that the average width of the confidence intervals around the association estimate were smaller than all other software. This could be evidence of underestimation of standard error. This is similar to what was found in simulations in Yuen and Mackinnon (2016). Otherwise we did not see underestimation for the other covariates. Still, documentation of the `R` `JM` package does not indicate that any techniques are used to correct for the theoretical

underestimation. Hence this should be considered when using the `R JM` package with a Cox type survival submodel. We found convergence issues when using the adaptive Gauss-Hermite quadrature option only when using an unspecified baseline hazard in the `R JM` function. Additionally, we saw in our simulations that using a restrictive parametric baseline risk function that matches the truth does not aid in estimation but may shorten runtime. Unless there is a compelling reason to use a restrictive parametric function, such as Weibull, for the baseline hazard, we would recommend using a flexible hazard such as piecewise constant or spline based.

In our investigation through simulation, we found that the Stata `stjm` command can be very sensitive to the parameter values used in data generation. SAS can also have difficulties depending on the parameters chosen when using an unstructured covariance matrix, but the default model assuming uncorrelated random effects was always able to fit the models. The `R` functions were always able to fit models to at least some of the datasets, no matter the parameter values used.

Basic joint modeling capabilities are available in the main statistical programming languages, `R`, SAS, and Stata, giving the user an option to use the implementation in the language of their choice. Each software has good features as well as limitations. Overall, we would recommend `R JM` or SAS. The SAS macro `%JM` offers the most functionality, including submodel, association, and estimation options. `R JM` was consistently the fastest and includes almost as many options as SAS, so this may be a better option if runtime is a concern. The `joineR` function is limited in the type of models it can fit. But if a joint model with a Gaussian longitudinal response and a shared parameter survival submodel is appropriate for the data, `joineR` may be a good option because of its simplicity.

Our investigation was focused on giving an overview of the available software and their features. Our simulations were limited to three scenarios that could be modeled in each common software. Other simulation scenarios were explored but not every software was able to fit models to the data. Running more simulations with different data forms and more focus on time slopes in longitudinal models, different forms of the submodels, and various associations would add to the knowledge of these software. Development of this group of software is expected based on the con-

48

tinued research in this field and the applicability of these models. With increased generalization in the form of each part (longitudinal and survival submodels, and the association) the software could accommodate many more types of data. Some software are currently being developed and released to accommodate useful extensions such as competing risks and multiple longitudinal outcomes as we discussed. Further work on more robust estimation as well as faster algorithms would be useful. Fitting a simple joint model with only a few covariates is not prohibitively restrictive on time. The time required will increase with an increase in the number of covariates or a non-normal longitudinal models, especially if one has to rely on re-sampling techniques for inference purposes. Such extensions may require creative enhancements and approximations that would be computationally efficient. An important area in joint modeling that is largely unexplored is model diagnostics. While some of the software offer some basic diagnostics (Rizopoulos, 2010, 2016c) and dynamic predictions (Crowther et al., 2013; Garcia-Hernandez and Rizopoulos, 2015; Rizopoulos, 2010, 2016c), there is room for further expansion.

Bayesian models can be very powerful and are gaining traction in joint modeling literature but the implementation is relatively limited. Analyzing joint models in a general multi-purpose Bayesian statistical programming language such as OpenBUGS, JAGS or Stan is possible and attractive due to its flexibility. Defining a joint model can become complicated in OpenBUGS due to the lack of closed-form for the integrals encountered in estimation. A common simplifying assumption is that the survival process follows a parametric distribution, often Weibull (Guo, 2003; Guo and Carlin, 2004). Full proportional hazards survival submodels have been implemented in the literature and code is available in the supplemental materials of Rizopoulos and Ghosh (2011) for the case of multiple longitudinal outcomes and Andrinopoulou et al. (2013) for the case of two longitudinal variables and competing risks data. A competing risks joint model has been implemented in WinBUGS (Deslandes and Chevret, 2010). In the next chapter we will shift our

focus onto the competing risks scenario and develop our own models with Bayesian estimation.

Chapter 3

Bayesian Inference and Dynamic Prediction from Joint Models Under Competing Risks

3.1 Introduction

Joint models were developed to study the relationship between a longitudinal measurement and a time-to-event outcome, such as CD4 counts and time to AIDS diagnosis (De Gruttola and Tu (1994); Tsiatis et al. (1995)). The relationship between the longitudinal and time-to-event processes is further complicated when the event can have one of multiple causes, called competing risks. Competing risks data can arise in clinical trials and observational studies when multiple reasons for death are recorded or when death is a competing risk of the main event of interest such as disease progression. This is the case in our motivating dataset from a study of adrenocortical carcinoma (ACC) patients. The morphomics, which were collected over time, are believed to be related to time to disease change.

Typically, joint longitudinal and competing risks models use a mixed effects models framework for the longitudinal part and cause-specific proportional hazards (PH) models for the survival component with some shared variables inducing an association between the two parts (Williamson et al. (2008); Elashoff et al. (2007); Li et al. (2010); Hu et al. (2009); Rue et al. (2017); Hickey et al. (2018)). While cause-specific PH models are common, such modeling cannot capture any dependence between the competing risks themselves (Lakhal et al. (2008)). Further, not all survival models conform to a PH specification. In this chapter, we propose using parametric survival submodels from a log-location scale family in the survival part of our joint model. Utilizing para-

metric survival models assumes that the resulting hazard functions have a specific form usually described by a relatively small number of parameters. This reduces the number of parameters necessary to estimate in our models compared to proportional hazards models, whose baseline, in its completely unspecified form, is infinite-dimensional and typically gives rise to estimation issues. In the context of joint models, it is well-known that unspecified baseline hazard estimation leads to underestimated standard errors if not properly corrected (Hseih et al. (2006)). Therefore most joint models using proportional hazards will define the baseline hazard to be a piece-wise spline function (Furgal et al. (2019)). However, when one allows these baseline hazards to differ for each event cause in a competing risks setting, the number of parameters to either be estimated or set apriori increases. A fully parametric model offers parsimony and allows flexibility to go beyond a specific structural formulation.

Specifically, we focus on Weibull and log-Normal models for our competing risks component. Weibull and log-Normal are two widely implemented formulations adopted in the analysis of parametric time-to-event data. Both of the distributions fall under the umbrella of the log-location-scale family of models. While Weibull conforms to a PH structure, assuming the failure times to follow either a log-Normal or a Weibull distribution results in an accelerated failure time class of models. The use of parametric survival models has recently been used in joint models with a single cause of failure and to date we have not seen any application of parametric survival distributions with joint longitudinal and competing risks models (Dil and Karasoy (2020)).

We develop and estimate our models under a Bayesian framework. The Bayesian technique can help leverage data with a relatively small sample size. Further, joint models are known to produce convergence problems and computational challenges and Bayesian estimation may be a more efficient approach. The posterior predictive distribution can also be used for relatively easy prediction. Joint models have often been used for dynamic prediction in biomedical settings, predictions that can be quite useful in clinical settings (Proust-Lima and Taylor (2009); Taylor et al. (2013); Rizopoulos et al. (2014); Blanche et al. (2015); Andrinopoulou et al. (2017, 2018); Li and Luo (2019); Wu et al. (2019)). The event prediction can be updated with each new longitudinal

biomarker measurement in a seamless way under the Bayesian setup.

The rest of the chapter is organized as follows. In Section 3.2 we develop our models and we describe the Bayesian estimation in Section 3.3. We apply our models to the adrenocortical carcinoma data in Section 3.4 and then explore model performance with simulations in Section 3.5. Section 3.6 explains the premise of dynamic predictions in our context and supplies an example. We conclude with a discussion in Section 3.7.

3.2 Framework

Let N be the number of subjects, indexed by $i = 1, ..., N$. Each subject has $J_i \geq 1$ measurements of the longitudinal process $Y(\tau)$ with τ measuring time since the start of the study. Subject i has observations at times $\tau_i = (\tau_{i1}, \tau_{i2}, ..., \tau_{ij}, ..., \tau_{iJ_i})^T$. Let Y_{ij} denote subject i's measurement at time τ_{ij}, i.e. $Y_{ij} = Y(\tau_{ij})$. Let $Y_i = (Y_{i1}, ..., Y_{iJ_i})^T$ be the vector of longitudinal measurements for subject i.

Each subject can experience an event with one of $K \geq 2$ causes indexed by k. Let T_i be the event time for subject i and $D_i = (D_{i1}, ..., D_{ik}, ..., D_{iK})^T$ be the vector of event indicators where $D_{ik} = 1$ if subject i has event with cause k and $D_{il} = 0$ for all $l \neq k$. A subject's event time may be right-censored in which case $D_{ik} = 0$ for all $k = 1, ..., K$.

Baseline covariates are denoted X_i in the longitudinal model and W_i in the survival submodel. X_i may depend on the observation time τ_{ij}. In the survival submodels, for simplicity, we assume W_i is time independent. The ensemble X_i and W_i may share covariates, but need not be the same. Let the number of covariates in the longitudinal submodel be p and the survival submodel be q.

3.2.1 Longitudinal submodel

We assume the linear mixed effects model in (3.1) for the longitudinal outcome Y_i. Here we assume that Y is continuous and normally distributed. Extending this to a generalized linear model for non-Gaussian outcomes Y is straightforward. We include subject-specific random effects U_i with design matrix Z_i. In (3.1), β is a p-vector of regression parameters and the ϵ_{ij} are random

measurement errors with $\epsilon_{ij} \sim \mathrm{N}(0, \sigma_\epsilon^2)$. Let $\epsilon_i = (\epsilon_{i1}, ..., \epsilon_{iJ_i})^T$.

$$Y_i = X_i\beta + Z_iU_i + \epsilon_i \tag{3.1}$$

We assume that given U_i the Y_{ij} are independent, i.e. $Y_{ij} \perp Y_{ij'} \mid U_i$ if $j \neq j'$. For simplicity we will work with a subject-specific random intercept only, i.e. $Z_i = 1$, and $U_i = U_{0i} \sim \mathrm{N}(0, \sigma_U^2)$.

3.2.2 Competing Risks Survival Submodels

For the competing risks data, we use parametric models from the log-location-scale family (Basu et al. (2003); Mukhopadhyay and Roy (2016)). We model the distribution of the failure times directly within a latent failure time framework. A common critique of the latent failure time approach to competing risks analysis is that this framework can suffer from identifiability issues (Tsiatis (1975)). This problem can be avoided by assuming the latent failure times are conditionally independent given random effects or by including a regression model with covariates (Heckman and Honorè (1989)). Our models both assume conditional independence and include covariates so the parameters in our model are identifiable.

We assume that if there are K possible causes of failure, each subject, in theory, has a time to event for each cause. Denote the time to event of cause k for subject i as T_{ik}^*, $i = 1, ..., N$, $k = 1, ..., K$. Subjects may also be independently censored and we will call T_{i0}^* the independent censoring time. Dependent censoring can be included as one of the K possible causes of failure. We observe only the minimum failure time called T_i for subject i, $T_i = \min\left(T_{i0}^*, T_{i1}^*, ..., T_{iK}^*\right)$. We assume that given some random effects V_i, sometimes called a frailty, the T_{ik} are independent, i.e. $T_{ik} \perp T_{ik'} \mid V_i, W_i$ for $k \neq k'$. We also assume that the longitudinal and event time outcomes for a single subject are conditionally independent given the random effects $Y_{ij} \perp T_i \mid U_i, V_i, X_i, W_i$ for all i and j. We focus on two specific distributions in the log-location-scale family: Weibull and log-Normal, which are by far the most commonly used distributions for parametric survival analysis.

Model W: Weibull

In the Weibull model we assume that given V_i the latent event times follow a Weibull distribution with scale parameter γ_k and shape parameter μ_{ik}. The shape and scale depend on the cause of failure k and the shape μ_{ik} will also depend on the subject i. Then

$$T_{ik}^* | V_i \sim \text{Weibull} \left(\mu_{ik}, \gamma_k \right), \quad k = 1, ..., K \tag{3.2}$$

To incorporate the covariates and random effects, we put a regression model on the log of the Weibull scale, assuming

$$\log \left(\mu_{ik} \right) = W_i \alpha_k + \theta_k^T V_i \tag{3.3}$$

where W_i is the vector of covariates with regression coefficient q-vector α_k. Note that we are assuming the covariates W_i are the same for all k for simplicity. Theoretically these W_i could depend on k.

An association between the longitudinal and survival submodels is induced by assuming the random effects in (3.1) and (3.3), U_i and V_i, are correlated. For simplicity, we will assume they are the same, i.e. $V_i = U_i$. The regression model can be rewritten as

$$\log \left(\mu_{ik} \right) = W_i \alpha_k + \theta_k^T U_i \tag{3.4}$$

The parameter θ_k measures the strength of the association between the longitudinal and survival processes. Here $\theta_k = 0$ would mean there is no association.

When interpreting the results from fitting this model, an increase in the scale parameter μ_{ik} with constant γ_k moves the center of the distribution to the right on the x-axis, thus moving the density away from 0. Hence an increase in $\mu_{ik} = \exp(W_i \alpha_k + \theta_k^T U_i)$, meaning a larger value for either α_k or θ_k, holding other parameters constant, corresponds to generally longer survival times. A positive association parameter θ_k means that a larger difference in Y from its expected value is associated with longer survival and a negative θ_k value means a larger baseline Y is associated

with shorter survival. The shape parameters γ_k determine the shape of the Weibull distribution with $\gamma_k = 1$ reducing to an exponential distribution.

Model L: Log-Normal

We also consider a joint model with the latent failure times following a log-Normal distribution with mean μ_{ik} and standard deviation γ_k, given random intercept $V_i = U_i$, i.e.

$$T_{ik}^* | U_i \sim \text{logNormal}\left(\mu_{ik}, \gamma_k^2\right), \quad k = 1, ..., K \tag{3.5}$$

Here we put a regression model on the mean

$$\mu_{ik} = W_i \alpha_k + \theta_k^T U_i \tag{3.6}$$

with α_k, W_i, and θ_k as described in Section 3.2.2. Interpretation is similar to the Weibull model. Here a larger positive value of μ_{ik} with a constant standard deviation γ_k leads to more density away from zero meaning longer survival times.

3.3 Bayesian Model and Estimation

We use Bayesian techniques to fit these models. In our Bayesian model, we assume that the random effect is a scalar U_{0i} and our longitudinal outcome Y_{ij} follows a multivariate Normal distribution with mean $X_{ij}\beta + U_{0i}$ and standard deviation $\sigma_\epsilon I_{J_i}$. Here I_J is the $J \times J$ identity matrix.

$$Y_i | X_i, \beta, U_{0i}, \sigma_U, \sigma_\epsilon^2 \sim \text{MVN}\left(X_i\beta + U_{0i}, \sigma_\epsilon I_{J_i}\right)$$

Our survival submodels will assume the latent failure event times for cause k ($k = 1, ..., K$) follow either a Weibull or log-Normal distribution as in (3.2) or (3.5) called Model W and Model

L, respectively.

Model W: $T_i|D_{i,k} = 1, W_i, U_{0i}, \gamma_k, \alpha_k, \theta_k \sim \text{Weibull}\left(\mu_{ik}, \gamma_k\right),$ with $\log\left(\mu_{ik}\right) = W_i\alpha_k + \theta_k U_{0i}$

Model L: $T_i|D_{i,k} = 1, W_i, U_{0i}, \gamma_k, \alpha_k, \theta_k \sim \text{logNormal}\left(\mu_{ik}, \gamma_k^2\right),$ with $\mu_{ik} = W_i\alpha_k + \theta_k U_{0i}$

3.3.1 Priors

We will call the set of parameters $\Omega = \{\beta, \sigma_U, \sigma_\epsilon, \gamma_1, ..., \gamma_K, \alpha_1, ..., \alpha_K, \theta_1, ..., \theta_K\}$ on which we put the following priors.

$$\beta \sim \text{MVN}\left(m_\beta, s_\beta^2 I_p\right)$$
$$\gamma_k \sim \Gamma\left(a, b\right)$$
$$U_{0i} \sim \text{N}\left(0, \sigma_U^2\right)$$
$$\alpha_k \sim \text{MVN}\left(m_\alpha, s_\alpha^2 I_q\right)$$
$$\sigma_U \sim \Gamma\left(a, b\right)$$
$$\theta_k \sim \text{N}\left(m_\theta, s_\theta^2\right)$$
$$\sigma_\epsilon \sim \Gamma\left(a, b\right)$$

We have hyperparameters m_β, s_β, m_α, s_α, m_θ, s_θ, a, and b. The Gamma distribution hyperparameters a and b could differ for each parameter but for simplicity we will use the same values.

3.3.2 Complete Data Likelihood and Posterior

The likelihood contribution from the longitudinal submodel is below where $p(Y_i|X_i, \beta, U_{0i}, \sigma_U, \sigma_\epsilon)$ is the density function of the multivariate normal distribution.

$$L_Y = \prod_{i=1}^{N} p(Y_i|X_i, \beta, U_{0i}, \sigma_U, \sigma_\epsilon)$$
$$= \prod_{i=1}^{N} (2\pi\sigma_\epsilon)^{-\frac{J_i}{2}} \exp\left\{-\frac{1}{2\sigma_\epsilon^{J_i}}\left(Y_i - \beta^T X_i - U_{0i}\right)^T \left(Y_i - \beta^T X_i - U_{0i}\right)\right\}$$

The likelihood contribution from the competing risks survival submodels takes the form

$$L_T = \prod_{i=1}^{N} \prod_{k=1}^{K} h_k(T_i|D_i, W_i, U_{0i}, \gamma_k, \alpha_k, \theta_k)^{D_{ik}} S_k(T_i|D_i, W_i, U_{0i}, \gamma_k, \alpha_k, \theta_k)$$

where h_k and S_k are the hazard function and survival function, respectively, for cause k. The hazard and survival functions for Model W are

$$\text{Model W: } h_k(T_i|D_i, W_i, U_i, \gamma_k, \alpha_k, \theta_k) = \frac{\gamma_k}{\mu_{ik}} T_i^{\gamma_k - 1}$$

$$S_k(T_i|D_i, W_i, U_i, \gamma_k, \alpha_k, \theta_k) = \exp\left\{ -\frac{T_i^{\gamma_k}}{\mu_{ik}} \right\}$$

The functions for Model L are below where $\phi(x)$ and $\Phi(x)$ are the density function and cumulative distribution function (CDF) of a standard normal distribution, respectively.

$$\text{Model L: } h_k(T_i|D_i, W_i, U_i, \gamma_k, \alpha_k, \theta_k) = \frac{\phi\left(\gamma_k^{-1}(\log T_i - \mu_{ik})\right)}{T_i \gamma_k \left(1 - \Phi\left(\gamma_k^{-1}(\log T_i - \mu_{ik})\right)\right)}$$

$$S_k(T_i|D_i, W_i, U_i, \gamma_k, \alpha_k, \theta_k) = 1 - \Phi\left(\gamma_k^{-1}(\log T_i - \mu_{ik})\right)$$

Let $p(\Omega)$ denote the product of the density functions for the priors of all parameters in Ω and $p(U) = \prod_{i=1}^{N} p(U_{0i})$ the product of the densities of U_i for $i = 1, ..., N$. The posterior is proportional to the product of the likelihood and priors.

$$p(\Omega|Y, T, D, X, WU) \propto L_Y L_T p(U) p(\Omega).$$

The posterior is high-dimensional and intractable. The standard path is to take a computational approach using a Markov Chain Monte Carlo (MCMC) technique. The full conditionals for both models are included in Appendix G.1. Drawing from the posterior is accomplished through a Hamiltonian Monte Carlo which is available in Stan. We fit our models in Stan through the R package rstan version 2.19.3 and using R version 3.6.1 (Guo et al. (2020)).
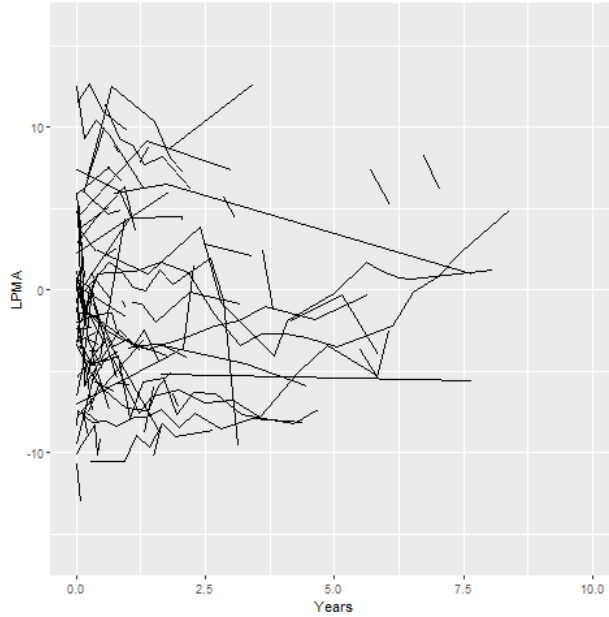
Figure 3.1: Centered LPMA longitudinal measurements from the ACC data.

3.4 Application to ACC Data

We apply these models to the adrenocortical carcinoma (ACC) data from Section 1.5. Patients received repeated CT scans in order to monitor their disease (McDuffie and Aufforth (2016)). Images of the patient's tumor were compared to those in the previous scan to determine the state of disease. The disease could be categorized into one of three states: progression (increased tumor size), regression (decreased tumor size), or stable (no significant tumor size change). Here we are interested in time to first disease status change which can be either a progression or a regression. Time is measured from date of diagnosis.

A previous study found that some morphomics, including lean psoas muscle area (LPMA), had significant associations with the survival outcomes separately (Miller et al. (2012)). We are interested in the association between LPMA and time to disease status change. Figure 3.1 shows the longitudinal LPMA measurements for all patients in the study.

The ACC data used in this chapter is summarized in Table 3.1. The study included 159 patients. The patients had between 1 and 19 scans with an average of 3.1 scans and a median of 2 scans per patient. The majority of patients (54%) had a progression and 23% had a regression. Information

Baseline Covariates		Mean	Std. dev.
Age* (years), at diagnosis		46.3	13.3
		Count	**%**
Sex	Male	68	43
	Female	91	57
Longitudinal Data		**Mean**	**Std. dev.**
LPMA* (mm^2)		17.6	6.1
Scan time (years)		2.1	3.3
Survival Data		**Count**	**%**
Event: Change in disease state	Progression	86	54
	Regression	37	23
	Censored	36	23
		Median	**Range**
Time to Event (years)		0.8	(0.03, 24.5)

Table 3.1: Description of adrenal cancer data used in Chapter 3. * Variables were centered in models, i.e. X-mean(X).

on age, sex were used as baseline covariates.

We fit our models to the ACC data. The longitudinal submodel includes a time slope and two covariates, binary sex (X_{1i}) and continuous baseline age (X_{2i}). A random subject-specific intercept U_i is included. There are two competing risks; $k = 1$ corresponds to first progression and $k = 2$ to first regression. Age is the only covariate included in the survival submodels. Adding additional covariates to the survival submodels led to fitting problems due to the small sample. We fit both Model W and Model L to the data. The form of the distributions for our joint models are shown in (3.7).

$$Y_i \sim \text{N} \left(\beta_0 + \beta_1 \tau_i + \beta_2 X_{1i} + \beta_3 X_{2i} + U_i, \ \sigma_\epsilon^2 I_{J_i} \right)$$

$$\text{Model W:} \quad T_i \sim \text{Weibull} \left(\mu_{ik}, \gamma_k \right) \quad \text{with} \ \log(\mu_{ik}) = \alpha_{0k} + \alpha_{1k} X_{2i} + \theta_k U_{0i} \tag{3.7}$$

$$\text{Model L:} \quad T_i \sim \text{logN} \left(\mu_{ik}, \gamma_k^2 \right) \quad \text{with} \ \mu_{ik} = \alpha_{0k} + \alpha_{1k} X_{2i} + \theta_k U_{0i}$$

Let $\beta = (\beta_0, \beta_1, \beta_2, \beta_3)^T$, and $\alpha_k = (\alpha_{0k}, \alpha_{1k})^T$, $k = 1, 2$. We use the priors in Section 3.3.1

with hyperparameters $m_\beta = (0, -1, -1, -1)^T$, $m_\alpha = (1, 1)^T$, $m_\theta = -1$, $s_\beta = s_\alpha = s_\theta = 2$, $a = 3$, and $b = 0.5$. These hyperparameters were chosen to be weakly informative to aid in fitting the model. We fit the models with Stan using four chains with separate initial values and 27500 warm-up iterations and 30000 iterations total.

We also fit a cause-specific proportional hazards (PH) model as in 3.8 for comparison. For simplicity we use a parametric baseline hazard with a Weibull form for both outcomes.

$$Y_i \sim N\left(\beta_0 + \beta_1 \tau_i + \beta_2 X_{1i} + \beta_3 X_{2i} + U_i, \ \sigma_\epsilon^2 I_{J_i}\right)$$

$$\text{Model W PH: } h_k(T_i) = \frac{\gamma_k}{\alpha_{0k}} T_i^{\gamma_k - 1} \exp\left(\alpha_{1k} X_{2i} + \theta_k U_{0i}\right)$$

(3.8)

This PH model was fit in Stan with 5000 iterations, 3000 warm-up iterations and the same priors as in Section 3.3.1. The likelihood contributions have the same form as in Section 3.3.2 with the hazard and survival functions below (Derivations for the survival functions are in Appendix B.3.)

$$\text{Model W PH: } h_k(T_i|D_i, W_i, U_i, \gamma_k, \alpha_k, \theta_k) = \frac{\gamma_k}{\alpha_{0k}} T_i^{\gamma_k - 1} \exp\left(\alpha_{1k} X_{2i} + \theta_k U_{0i}\right)$$

$$S_k(T_i|D_i, W_i, U_i, \gamma_k, \alpha_k, \theta_k) = \exp\left(-\int_0^{T_i} \frac{\gamma_k}{\alpha_{0k}} s^{\gamma_k - 1} \exp\left(\alpha_{1k} X_{2i} + \theta_k U_{0i}\right) ds\right)$$

$$= \exp\left(-\frac{1}{\alpha_{0k}} T_i^{\gamma_k} \exp\left(\alpha_{1k} X_{2i} + \theta_k U_{0i}\right)\right)$$

3.4.1 Results

Table 3.2 shows the results and trace plots can be found in Figures F.1 and F.2 in the Appendix. We found that in all models, LPMA tended to decrease over time ($\beta_1 < 0$), older patients had lower LPMA ($\beta_3 < 0$) and females had much lower LPMA than males ($\beta_2 < 0$). We see that the association parameters (θ_1 and θ_2) in Models W and L have similar estimates. There is a small positive association between U_{0i} and progression and a small negative association with progression. We found that 92% of the estimates of θ_1 from the post-warm-up iterations in Model W and 95% in Model L were above 0 and 68% of estimates from Model W and 67% of estimates from Model

L of θ_2 were below zero. In Model W PH the association parameters have opposite signs and 92% of θ_1 estimates were below 0 while 64% of θ_2 estimates were above zero. In our models a higher θ_k value means a larger μ_k and longer survival times. On the other hand, in the W PH model a larger θ_k corresponds to a higher hazard and therefore smaller survival times. So the difference in signs between our models and the W PH model is reasonable. In each model time to regression tended to be longer than time to progression ($\alpha_{0,1} < \alpha_{0,2}$). Age had a small negative effect in the survival submodels for progression in each model ($\alpha_{1,1} < 0$) and regression for Models W andL ($\alpha_{1,2} < 0$). In these cases older patients tended to have a shorter time to event. The bottom row of Table 3.2 shows the mean Deviance Information Criteria (DIC) values from the post-warm-up iterations and the 2.5% and 97.5% quantiles. According to DIC Model W fit the data slightly better than Model L and Model W PH had the worst fit.

3.5   Simulations

For our simulations, we generated data to be similar to the ACC study. We ran simulations for a sample size of $N = 160$ (similar to the ACC data) and for $N = 1000$ subjects to investigate the effect of a larger sample size. For each subject, we have two covariates, one binary and one continuous, with distributions similar to sex and centered age in decades, respectively. For $i = 1, ..., N$, $X_{1i} \sim \text{Bin}(0.57)$ and $X_{2i} \sim \text{N}(0, 1.5^2)$. A random intercept is defined as $U_{0i} \sim \text{N}(0, 4^2)$.

For each subject we generate longitudinal measurements at starting at time 0 and every 0.125 years until 3 years, then every 0.5 years until 5 years and every 1 year until 8 years, i.e. $\tau = (0, 0.125, 0.25, ..., 2.875, 3, 3.5, 4, 4.5, 5, 6, 7, 8)^T$. Thus subjects have up to 32 longitudinal measurements. These times were chosen to represent patients getting regular scans early after diagnosis and having less frequent scans as time since diagnosis increases. At each measurement time a longitudinal outcome Y_{ij} ($i = 1, ..., N$, $j = 1, ..., 32$) is calculated as

$$Y_{ij} = \beta_0 + \beta_1 \tau_j + \beta_2 X_{1i} + \beta_3 X_{2i} + U_{0i} + \epsilon_{ij}$$

The measurement errors are $\epsilon_{ij} \sim \text{N}(0, 2.2^2)$ and the true values for β differ for Models W and L.

| | Model W | | | Model L | | |
Parameter	Mean	CI	Pr > 0	Mean	CI	Pr > 0
β_0	3.38	(2.31, 4.42)	1.00	3.37	(2.29, 4.42)	1.00
β_1	-0.14	(-0.30, 0.01)	0.06	-0.13	(-0.27, 0.02)	0.07
β_2	-5.81	(-7.13, -4.39)	0.00	-5.82	(-7.12, -4.45)	0.00
β_3	-1.42	(-1.96, -0.87)	0.00	-1.43	(-1.97, -0.87)	0.00
$\alpha_{0,1}$	1.34	(1.05, 1.66)	1.00	0.65	(0.36, 0.96)	1.00
$\alpha_{1,1}$	-0.05	(-0.29, 0.19)	0.36	-0.04	(-0.26, 0.17)	0.33
$\alpha_{0,2}$	3.18	(2.42, 4.16)	1.00	2.77	(1.87, 3.87)	1.00
$\alpha_{1,2}$	-0.20	(-0.73, 0.31)	0.21	-0.24	(-0.77, 0.30)	0.19
θ_1	0.08	(-0.02, 0.17)	0.92	0.07	(0.00, 0.15)	0.95
θ_2	-0.06	(-0.23, 0.10)	0.32	-0.05	(-0.23, 0.12)	0.33
γ_1	0.75	(0.64, 0.88)	1.00	1.59	(1.36, 1.87)	1.00
γ_2	0.53	(0.40, 0.67)	1.00	3.18	(2.46, 4.16)	1.00
σ_U	4.31	(3.78, 4.91)	1.00	4.30	(3.77, 4.89)	1.00
σ_ϵ	2.18	(2.02, 2.36)	1.00	2.18	(2.02, 2.36)	1.00
DIC	-2931.6	-3255.5; -2616.9		-2665.4	-2975.3; -2400.6	

| | Model W PH | | |
Parameter	Mean	CI	Pr > 0
β_0	3.51	(2.40, 4.60)	1.00
β_1	-0.15	(-0.34, 0.04)	0.06
β_2	-6.10	(-7.51, -4.69)	0.00
β_3	-1.45	(-2.01, -0.87)	0.00
$\alpha_{0,1}$	2.70	(2.03, 3.56)	1.00
$\alpha_{1,1}$	-0.11	(-0.58, 0.34)	0.33
$\alpha_{0,2}$	5.04	(3.55, 6.90)	1.00
$\alpha_{1,2}$	0.16	(-0.48, 0.75)	0.69
θ_1	-0.05	(-0.12, 0.02)	0.08
θ_2	0.01	(-0.07, 0.10)	0.64
γ_1	0.76	(0.64, 0.88)	1.00
γ_2	0.51	(0.38, 0.64)	1.00
σ_U	4.43	(3.88, 5.06)	1.00
σ_ϵ	2.05	(1.87, 2.25)	1.00
DIC	-1074.5	-1098.2; -1053.8	

Table 3.2: Posterior Mean (Mean), 95% Credible Interval (CI), and probability the estimate was greater than 0 (Pr > 0) for the ACC data using Model W. The bottom row contains the average Deviance Information Criteria (DIC) under Mean and the 2.5% and 97.5% quantiles under CI.

Model	$\alpha_{0,1}$	$\alpha_{1,1}$	$\alpha_{0,2}$	$\alpha_{1,2}$	θ_1	θ_2	γ_1	γ_2
W	1.5	0.1	3.1	-0.1	0.05	-0.2	0.8	0.5
L	0.7	-0.02	1.9	0.2	0.1	-0.12	1.5	2.7

Table 3.3: True parameter values used to generate competing risks survival data for simulation study.

For Model W we have $\beta_0 = 2.4$, $\beta_1 = -0.1$, $\beta_2 = -5.1$, $\beta_3 = -1$. For Model L we have $\beta_0 = 2$, $\beta_1 = -0.02$, $\beta_2 = -4.2$, $\beta_3 = -1.7$.

We generate competing risks survival data with $K = 2$ risks. We do this once assuming Model W is the truth and then assuming Model L is the truth. For each subject μ_{ik} is calculated as follows for each Model.

$$\text{Model W: } \mu_{ik} = \exp(\alpha_{0k} + \alpha_{1k}X_{2i} + \theta_k U_{0i})$$

$$\text{Model L: } \mu_{ik} = \alpha_{0k} + \alpha_{1k}X_{2i} + \theta_k U_{0i}$$

A survival time for each risk ($k = 1, 2$) is drawn from a Weibull or log-Normal distribution.

$$\text{Model W: } T_{ik}^* \sim \text{Weibull}(\mu_{ik}, \gamma_k)$$

$$\text{Model L: } T_{ik}^* \sim \text{logN}(\mu_{ik}, \gamma_k^2)$$

True parameter values for the survival data generation are shown in Table 3.3. An independent censoring time is drawn from a uniform distribution $T_{i,0}^* \sim \text{Unif}(0, 7)$. The observed time is the minimum $T_i = \min(T_{i,0}^*, T_{i,1}^*, T_{i,2}^*)$. We keep the longitudinal outcome Y_{ij} for a subject if $\tau_j < T_i$ and we drop all other Y_{ij}. Models were fit using Stan via R with four chains with 57500 warm-up iterations and 60000 iterations total. We generated 200 datasets for each combination of Model (W or L) and N (160 or 1000) and fit the model on each dataset.

3.5.1 Results

Simulation results are shown in Tables 3.4 and 3.5. The table includes the average of the posterior means (Mean) for the estimate from the 200 generated datasets. We also show the bias and Mean

squared error (MSE) of these estimates multiplied by 1000. Finally we list the coverage probability which is the percent of credible intervals from the 200 dataset fits that include the true value.

We see similarities between each model (Models W and L) and for each N. Bias and MSE are generally small in each model and bias and MSE are reduced with the larger sample size, N=1000. The parameters with the largest biases and MSEs are the longitudinal intercept (β_0) and the X_1 coefficient (β_2). In the survival models, parameters related to the second event type, specifically $\alpha_{0,2}$, $\alpha_{1,2}$ and γ_2, tend to have larger bias and MSE than corresponding parameters for event type 1. This is likely from a smaller number of events of type 2 observed in the data. Coverage probabilities are generally close to 95. The few parameters that have a lower coverage probability in the N=160 case, β_0, β_2, and γ_2 in Model L, have better coverage for $N = 1000$. Data was generated for two other simulation scenarios with different true parameter values. Results for those scenarios are similar to the results discussed here and are shown Tables D.1, D.2, D.3 and D.4 in the Appendix.

True surv.	N	Param	True Value	Mean	Bias*1000	MSE*1000	CP
Weibull	160	β_0	2.4	2.09	-313	300	89
(Model W)		β_1	-0.1	-0.11	-6	3	98
		β_2	-5.1	-4.53	575	633	87
		β_3	-1.0	-0.96	38	53	94
		σ_U	4.0	3.95	-53	59	95
		σ_ϵ	2.2	2.20	-1	2	96
		$\alpha_{0,1}$	1.5	1.52	18	31	98
		$\alpha_{1,1}$	0.1	0.11	7	11	96
		$\alpha_{0,2}$	3.1	3.00	-99	165	96
		$\alpha_{1,2}$	-0.1	-0.06	35	43	96
		θ_1	0.05	0.06	5	2	94
		θ_2	-0.2	-0.19	14	7	96
		γ_1	0.8	0.81	7	6	96
		γ_2	0.5	0.53	32	5	95
Weibull	1000	β_0	2.4	2.34	-56	43	93
(Model W)		β_1	-0.1	-0.10	-2	1	95
		β_2	-5.1	-4.99	107	77	92
		β_3	-1.0	-1.00	3	7	96
		σ_U	4.0	4.00	-4	9	95
		σ_ϵ	2.2	2.20	-1	0	95
		$\alpha_{0,1}$	1.5	1.51	6	5	94
		$\alpha_{1,1}$	0.1	0.10	2	2	95
		$\alpha_{0,2}$	3.1	3.09	-7	44	92
		$\alpha_{1,2}$	-0.1	-0.10	-1	8	95
		θ_1	0.05	0.05	1	0	95
		θ_2	-0.2	-0.19	6	1	95
		γ_1	0.8	0.80	0	1	92
		γ_2	0.5	0.50	5	1	94

Table 3.4: Simulation results for data generated with Model W as the truth. Data was generated for either N=160 (top) or N=1000 (bottom). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP).

True surv.	N	Param	True Value	Mean	Bias*1000	MSE*1000	CP
log-Normal	160	β_0	2.0	1.69	-313	266	92
(Model L)		β_1	-0.02	-0.01	9	5	97
		β_2	-4.2	-3.64	555	595	87
		β_3	-1.7	-1.69	14	47	94
		σ_U	4.0	3.99	-9	56	96
		σ_ϵ	2.2	2.20	2	2	98
		$\alpha_{0,1}$	0.7	0.72	24	23	97
		$\alpha_{1,1}$	-0.02	-0.02	-1	12	95
		$\alpha_{0,2}$	1.9	1.81	-88	114	96
		$\alpha_{1,2}$	0.2	0.18	-20	41	93
		θ_1	0.1	0.10	-2	2	95
		θ_2	-0.12	-0.12	-3	7	93
		γ_1	1.5	1.50	-1	13	96
		γ_2	2.7	2.56	-135	95	88
log-Normal	1000	β_0	2.0	1.95	-55	38	95
(Model L)		β_1	-0.02	-0.02	-2	1	93
		β_2	-4.2	-4.06	137	83	92
		β_3	-1.7	-1.68	21	8	93
		σ_U	4.0	4.00	-2	10	94
		σ_ϵ	2.2	2.20	-1	0	97
		$\alpha_{0,1}$	0.7	0.71	5	4	98
		$\alpha_{1,1}$	-0.02	-0.02	3	2	96
		$\alpha_{0,2}$	1.9	1.87	-35	28	93
		$\alpha_{1,2}$	0.2	0.20	-5	5	97
		θ_1	0.1	0.10	0	0	94
		θ_2	-0.12	-0.12	4	1	94
		γ_1	1.5	1.50	1	2	97
		γ_2	2.7	2.67	-27	16	93

Table 3.5: Simulation results for data generated with Model L as the truth. Data was generated for either N=160 (top) or N=1000 (bottom). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP).

3.6 Dynamic Prediction

We model our approach to dynamic prediction based on that used by Andrinopoulou and colleagues (Andrinopoulou et al. (2017)). For a new subject l, denote the set of subject l's longitudinal

measurements up until time s as $\mathcal{Y}_l(s)$. The subject has baseline covariate data X_l and W_l. Let $\mathcal{O}$ be the data on which the model was originally fit, $\mathcal{O} = \{Y_i, T_i, D_i, X_i, W_i \mid i = 1, ..., N\}$. We are interested in the probability of subject l experiencing an event of cause k before time t after surviving event-free up until time s ($s < t$). We denote this probability as $\pi_{l,k}(s, t)$.

$$\pi_{l,k}(s, t) = Pr(T_{lk} < t \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \mathcal{O}) \tag{3.9}$$

Based on the posterior predictive distribution of the parameters Ω, we can write $\pi_{l,k}(t, s)$ as an integral.

$$\pi_{l,k}(s, t) = \int Pr(T_{lk} < t \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega) p(\Omega \mid \mathcal{O}) \, d\Omega \tag{3.10}$$

The first part of the integrand in (3.10) can be written in terms of the overall survival function $S(t)$ and the cumulative incidence function $CIF(s, t, k) = \int_s^t h_k(v) S(v) dv$.

$$
\begin{aligned}
&Pr(T_{lk} < t \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega) \\
&= \int Pr(T_{lk} < t \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega, U_l) p(U_l \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega) dU_l \\
&= \int Pr(T_{lk} < t \mid T_l > s, W_l, \Omega, U_l) p(U_l \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega) dU_l \\
&= \int \frac{Pr(T_{lk} < t, T_l > s \mid W_l, \Omega, U_l)}{Pr(T_l > s \mid W_l, \Omega, U_l)} p(U_l \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega) dU_l \\
&= \int \frac{CIF(s, t, k \mid W_l, \Omega, U_l)}{S(s \mid W_l, \Omega, U_l)} p(U_l \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega) dU_l
\end{aligned}
$$

Since the longitudinal and survival outcomes are assumed independent given U_l, we have

$$
\begin{aligned}
p(U_l \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega) &\propto p(\mathcal{Y}_l(s), T_l > s \mid U_l, X_l, W_l, \Omega) p(U_l \mid X_l, W_l, \Omega) \\
&= p(\mathcal{Y}_l(s) \mid U_l, X_l, W_l, \Omega) p(T_l > s \mid U_l, X_l, W_l, \Omega) p(U_l \mid X_l, W_l, \Omega) \\
&= p(\mathcal{Y}_l(s) \mid U_l, X_l, \Omega) S(s \mid U_l, W_l, \Omega) p(U_l \mid \Omega)
\end{aligned}
$$

Here $p(\mathcal{Y}_l(s) \mid U_l, X_l, \Omega)$ is the likelihood for $\mathcal{Y}_l$, $S(s \mid U_l, W_l, \Omega)$ is the overall survival function, and $p(U_l \mid \Omega)$ is the prior for U_l. We assumed that given U_l the latent failure times are independent

so the overall survival function is simply the product of the cause-specific survival functions which are defined in Section 3.3.2, i.e. $S(s \mid U_l, W_l, \Omega) = \prod_{k=1}^{K} S_k(s \mid U_l, W_l, \Omega)$. The longitudinal measurements for a single subject at different times are also independent given the random effect so $p(\mathcal{Y}_l(s) \mid U_l, X_l, \Omega) = \prod_{\{j:\tau_{lj} \leq s\}} p(Y_{lj} \mid U_l, X_l, \Omega)$. Therefore,

$$
\begin{aligned}
\pi_{l,k}(s,t) &= \int Pr(T_{lk} < t \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega)p(\Omega \mid \mathcal{O}) \, d\Omega \\
&= \int \int \frac{CIF(t,s,k \mid W_l, \Omega, U_l)}{S(s \mid W_l, \Omega, U_l)} p(U_l \mid T_l > s, \mathcal{Y}_l(s), X_l, W_l, \Omega)p(\Omega \mid \mathcal{O}) \, dU_l d\Omega \\
&\propto \int \int \frac{CIF(t,s,k \mid W_l, \Omega, U_l)}{S(s \mid W_l, \Omega, U_l)} \prod_{\{j:\tau_{lj} \leq s\}} p(Y_{lm} \mid U_l, X_l, \Omega) \prod_{k=1}^{K} S_k(s \mid U_l, W_l, \Omega)p(U_l \mid \Omega) \, dU_l d\Omega
\end{aligned}
$$

The probability of interest $\pi_{l,k}(s,t)$ can be estimated using a Monte Carlo simulation scheme with the following three steps iterated B times. At iteration b ($b = 1, ..., B$),

1. Draw $\Omega^{(b)}$ from the MCMC sample of the posterior $p(\Omega|\mathcal{O})$.

   We have $\Omega^{(b)} = \left\{ \beta^{(b)}, \sigma_U^{(b)}, \sigma_\epsilon^{(b)}, \gamma_1^{(b)}, ..., \gamma_K^{(b)}, \alpha_1^{(b)}, ..., \alpha_K^{(b)}, \theta_1^{(b)}, ..., \theta_K^{(b)} \right\}$

2. Draw $U_l^{(b)}$ from $p(U_l|\Omega^{(b)})$ using a Metropolis-Hastings algorithm as follows. Let

$$
g(U \mid X_l, W_l, \Omega) = \prod_{\{j:\tau_{lj} \leq s\}} p(Y_{lm} \mid U, X_l, \Omega) \prod_{k=1}^{K} S_k(s \mid U, W_l, \Omega)p(U \mid \Omega)
$$

   - Draw a new U^* from $N(U_l^{(b-1)}, \sigma_\epsilon^{(b)^2})$
   - Calculate $a = \min\left(1, \frac{g(U^* \mid X_l, W_l, \Omega^{(b)})}{g(U_l^{(b-1)} \mid X_l, W_l, \Omega^{(b)})}\right)$
   - Draw A from a Uniform$(0,1)$ distribution
   - If $A < a$, set $U_l^{(b)} = U^*$. Otherwise set $U_l^{(b)} = U_l^{(b-1)}$

3. Compute $\pi_{l,k}^{(b)}(s,t \mid W_l, U_l^{(b)}, \Omega^{(b)}) = \frac{\widehat{CIF}(s,t,k \mid W_l, U_l^{(b)}, \Omega^{(b)})}{S(s \mid W_l, U_l^{(b)}, \Omega^{(b)})}$.

   The CIF is estimated empirically with this empirical estimate denoted $\widehat{CIF}$. Note that $CIF(s,t,k)$ is the probability of having an event of type k before time t given no event up to time s ($s < t$). We describe this for $K = 2$ risks, but the method can be easily extended

for any K. At iteration b, we have parameter estimates $\Omega^{(b)}$ including $\gamma_1^{(b)}$ and $\gamma_2^{(b)}$ and we can calculate $\mu_{l1}^{(b)}$ and $\mu_{l2}^{(b)}$ as in (3.4) or (3.6) depending on the model chosen.

$$\text{Model W} : \mu_{lk}^{(b)} = \exp\left(W_l \alpha_k^{(b)} + \theta_k^{(b)^T} U_l^{(b)} \right)$$

$$\text{Model L} : \mu_{lk}^{(b)} = W_l \alpha_k^{(b)} + \theta_k^{(b)^T} U_l^{(b)}$$

We draw random variables $R_1^{[m]}$ and $R_2^{[m]}$ for $m = 1, ..., M$ from the following survival distributions. In Model W, $R_1^{[m]} \sim \text{Weibull}(\mu_{l1}^{(b)}, \gamma_1^{(b)})$, $R_2^{[m]} \sim \text{Weibull}(\mu_{l2}^{(b)}, \gamma_2^{(b)})$. In Model L, $R_1^{[m]} \sim \text{logN}(\mu_{l1}^{(b)}, \gamma_1^{(b)^2})$, $R_2^{[m]} \sim \text{logN}(\mu_{l2}^{(b)}, \gamma_2^{(b)^2})$. Let $R_{min}^{[m]} = \min(R_1^{[m]}, R_2^{[m]})$. Then

$$\widehat{CIF}(s, t, 1 \mid W_l, U_l^{(b)}, \Omega^{(b)}) = \frac{1}{M} \sum_{m=1}^{M} \left(Ind\left(s < R_{min}^{[m]} \leq t \right) \cdot Ind\left(R_1^{[m]} < R_2^{[m]} \right) \right)$$
(3.11)

$$\widehat{CIF}(s, t, 2 \mid W_l, U_l^{(b)}, \Omega^{(b)}) = \frac{1}{M} \sum_{m=1}^{M} \left(Ind\left(s < R_{min}^{[m]} \leq t \right) \cdot Ind\left(R_2^{[m]} < R_1^{[m]} \right) \right)$$
(3.12)

Here $Ind(\cdot)$ is the indicator function. The sum in Equation (3.11) counts how many of the M simulated subjects survive until time s and have an event of type 1 before time t and before an event of type 2. Equation (3.12) is similar except for an event of type 2 occurring first.

Steps 1-3 are repeated B times and the overall estimate of $\pi_{l,k}(s, t)$ defined in (3.9) is the mean of the $\pi_{l,k}^{(b)}$ from iterations $b = 1, ..., B$.

$$\hat{\pi}_{l,k}(s, t) = \frac{1}{B} \sum_{b=1}^{B} \pi_{l,k}^{(b)}(s, t \mid W_l, U_l^{(b)}, \Omega^{(b)})$$
(3.13)

3.6.1 Application

In order to demonstrate our dynamic predictions, we simulated data with covariate effects stronger than in the ACC data. We simulated data as in Section 3.5 but with the following true values: $\beta = (3, -1, -5, -2)^T$, $\sigma_U = 4$, $\sigma_\epsilon = 2$, $\alpha_1 = c(1, -2)^T$, $\alpha_2 = (3, -3)^T$, $\gamma = (0.75, 0.5)^T$, $\theta = (1, -1)^T$. We simulated data for 1000 subjects with Weibull survival times and fit Model W.

Data for three new subjects was then generated. Subject 1 had $X_1 = 0$ and $X_2 = -0.08$, Subject 2 had $X_1 = 1$ and $X_2 = -1.31$, and Subject 3 had $X_1 = 0$ and $X_2 = -0.36$. Subjects 1 and 2 had a progression at 0.38 and 1.28 years, respectively, while Subject 3 had a regression at 0.93 years. The longitudinal Y values generated for each subject are shown in Figure 3.2. Each subject has an increase in Y prior to the event. Subject 2 has the largest range of Y values and Subject 3 has overall much higher values of Y than the other two new subjects.
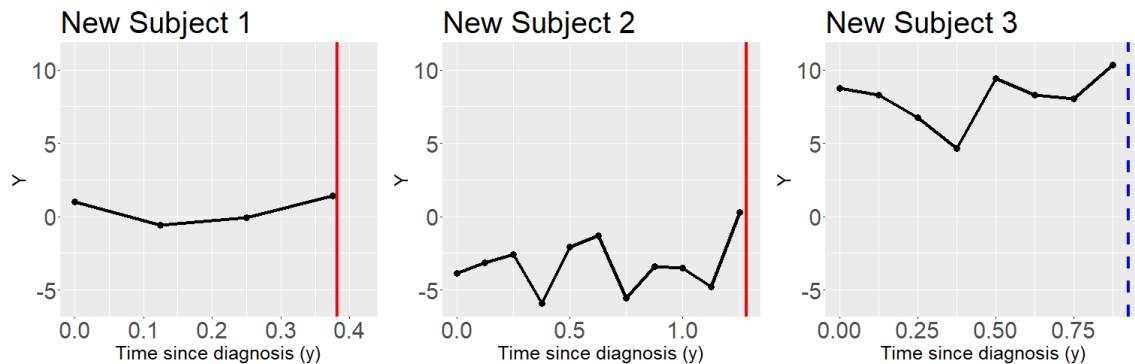

Figure 3.2: Longitudinal measurements over time for new subjects. Vertical line indicates the simulated event time with a solid red line corresponding to a progression and a dashed blue line corresponding to a regression.

At the time of each of a patient's longitudinal measurements τ_{ij} we calculated the estimated probability in (3.13) of experiencing either a progression or death given event-free survival up until time τ_{ij}, i.e. $\hat{\pi}_{l,k}(\tau_{ij}, t)$, $k = 1, 2$. We calculated this for every 0.05 years up to 3 years, meaning for $t = 0, 0.05, 0.1, 0.15, ..., 3$ if $t > \tau_{ij}$. We used $B = 200$ iterations to estimate the probabilities and $M = 200$ random variables for the empirical estimation of the CIF. Since we generated the data according to Model W, we show predictions only from the Weibull model. Figure 3.3 shows the estimated probabilities $\hat{\pi}_{l,k}(\tau_{ij}, t)$ over time t. We plotted only up to a short time after the last longitudinal measurement, specifically 0.5 years for Subject 1, 1.75 years for Subject 2 and 1.5 years for Subject 3. Note that while theoretically a CIF curve will be monotonically non-decreasing, our estimated CIF curves may have some small decreases due to the empirical nature of our estimate.
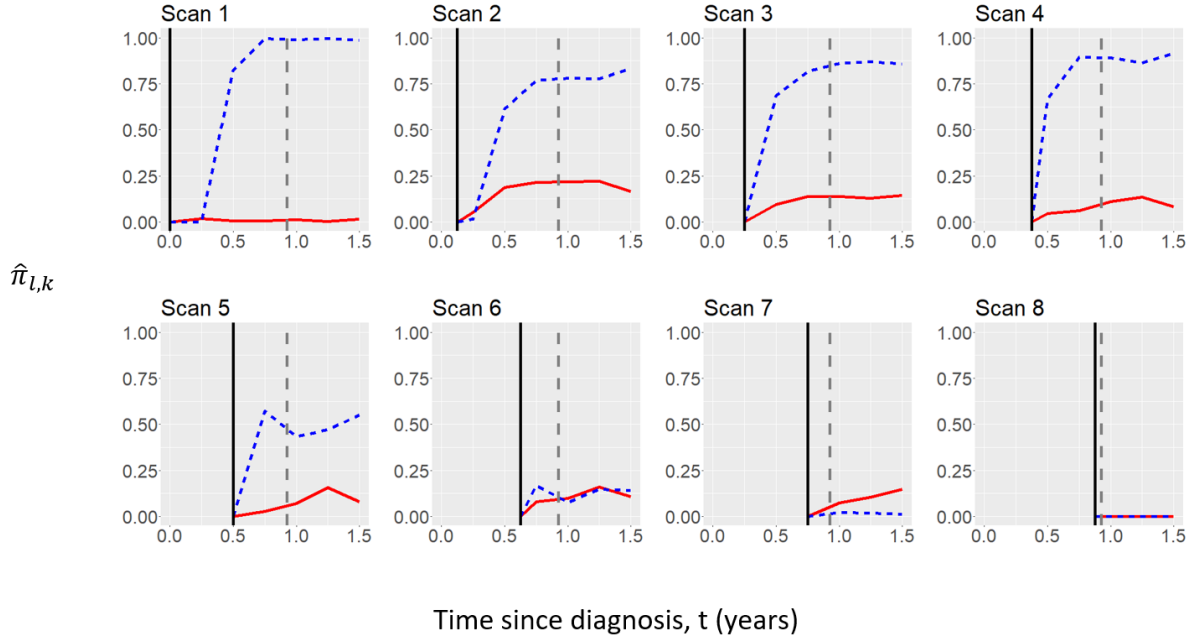
Predictions for Subject 1 interestingly start with almost no probability of either event but then
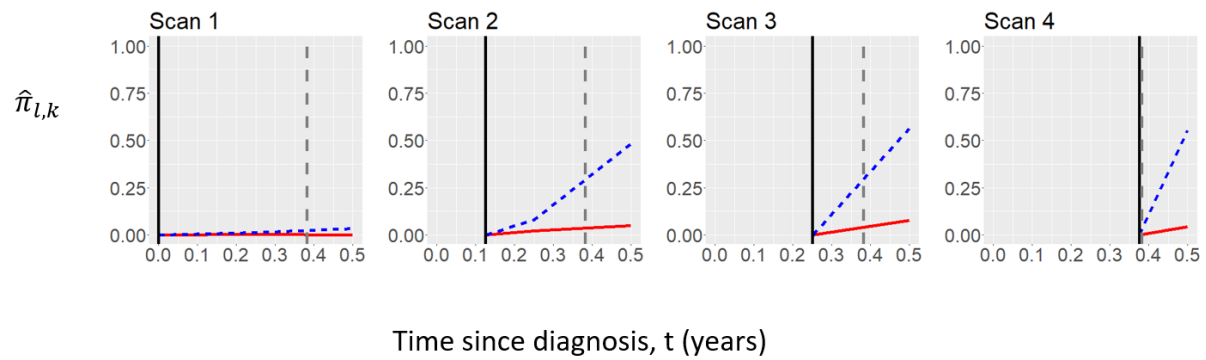
(a) New Subject 1
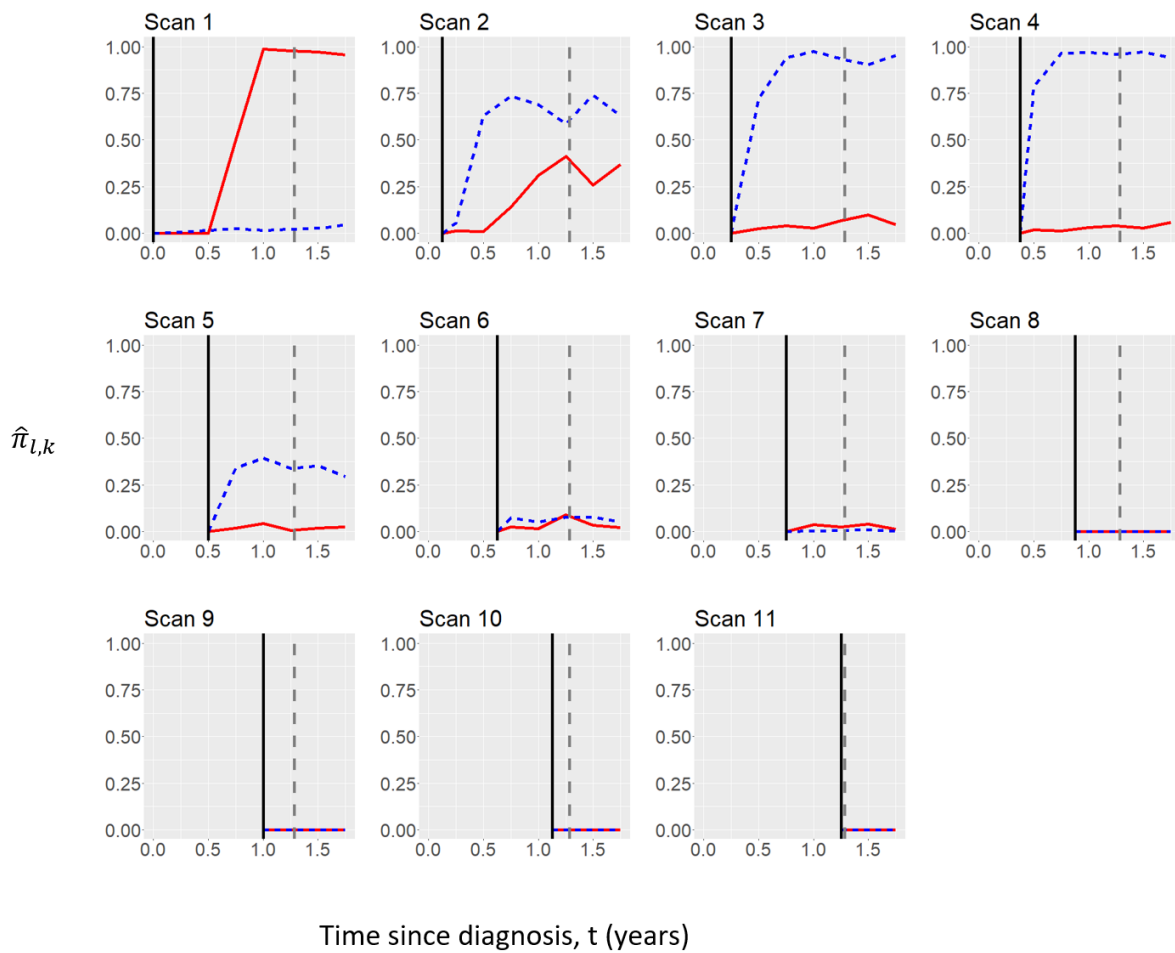
(b) New Subject 2

(c) New Subject 3

Figure 3.3: Dynamic predictions for new patients from Model W. The solid red curve is the probability of progression ($\hat{\pi}_{l,1}(\tau_{ij}, t)$) and the dashed blue line is the probability of regression ($\hat{\pi}_{l,2}(\tau_{ij}, t)$). The solid vertical line indicates the time of the last longitudinal measurement used to predict (τ_{ij}) and the dashed vertical line is at the observed event time in our data.

predicts a higher chance of regression after a decrease in Y. At scans 2 and 3 new subject 1's Y increases and so we see a small increase in the probability of progression. Subject 2 initially has a much higher chance of progression but after scan 2 regression has a higher probability until scan 6. Subject 3 starts with a high Y value and has a higher predicted probability of regression from scans 1 to 6. Model W predicts very little probability of either event for the last four scans for Subject 2 and the last scan for Subject 3 but we do not see the same flattening of both curves for Subject 1 who has an event earlier than the other two new subjects.

3.7 Discussion

This chapter proposes the joint modeling of longitudinal and competing risks data using parametric distributions for the survival submodel. The Bayesian approach is natural in this context as it exploits the hierarchical structure of the modeling framework and offers an efficient solution in

a situation that is often numerically challenging. We developed survival submodels using Weibull and log-Normal distributions and described how these joint models can be utilized for dynamic prediction. Our models were used to model data from the study of adrenocortical carcinoma and found results with similar interpretations for both the Weibull and log-Normal models. Comparing our models to a cause-specific proportional hazards model we found similar results for the longitudinal submodel parameters and most of the covariate effects in the survival submodels. But the PH model estimated the association parameters to have the opposite signs as our models. In all cases the association was small (θ_k was close to zero). While in our models these association parameters measure the effect of deviations of the LPMA from the mean on the survival time distribution, in the PH model these parameters change the hazard function and the opposite signs can be explained by this difference in parameterization. Our simulation results show small bias and MSE with good coverage even for the relatively small sample size of 160.

Parametric survival submodels can be construed as parsimonious alternatives to the oft-used cause-specific PH formulation. Indeed the Weibull model considered in this article conforms to the cause-specific semi-parametric PH structure. Parametric modeling allows further flexibility by allowing non-PH formulation (e.g. log-Normal) or the possibility of formulating regression of the scale parameters.

In Section 3.2 we assumed that the random effects in the longitudinal and competing risks submodels were the same ($V_i = U_i$). This was done for the sake of simplicity but it could be considered a strong assumption. We could instead assume that these are only correlated, for example with a multivariate normal distribution, i.e. $(V_i, U_i)^T \sim MVN(0, \Sigma)$. Additionally we used only a random subject-specific intercept but instead could have included additional random effects like a random slope. These options can be more flexible but also require estimation of additional parameters.

Although the current article focuses on a linear model for the longitudinal component, aligning it closer to the motivating data, it is apparent that the methodology will work in principle for scenarios where the longitudinal observations stem from generalized linear models (e.g. binary

or count data). Perhaps a more significant extension to the current discourse lies in the treatment of recurrent events, research on which is still developing in the joint modeling context (Han et al. (2007); Kim et al. (2012); Król et al. (2016); Cai et al. (2017); Ren et al. (2019)).

In analyzing competing causes of mortality, sometimes one may encounter missingness in the documentation of the exact cause. Such a phenomenon, commonly referred to as masking, has applications in the analysis of registry data or investigation of the failure pattern of complex multi-component industrial systems (Basu et al. (1999, 2003); Mukhopadhyay and Basu (2007); Sen et al. (2010); Bakoyannis et al. (2010); shou Ko (2019)). Bayesian framework is particularly useful in this case as it avoids imposing non-testable assumptions on the probability of missingness. The modeling framework we have proposed here can easily adapt to the masked data case.

A natural extension of the work proposed here is the incorporation of dependence among the risk components. Such dependence can be induced through a frailty or Copula structure. Of course, one has to be cognizant of the potential non-identifiability that can generate from the dependent models. Inclusion of covariates comes to the rescue as subject level heterogeneity enables separation of information across different risk components. Study of such dependence modeling is considered in the next chapter.

Chapter 4

Bayesian Inference for Joint Models Under Dependent Competing Risks

4.1 Introduction

In classical competing risks models the causes of failure are assumed stochastically independent (Carrière, 1995) but in some cases it can be unreasonable to assume such independence. This is especially true in medical studies where the risks may be causes of death or diagnosis of disease in a patient with complicated health. An example of possibly dependent risks are death without tumor, death from other causes with tumor, and death from tumor in the data described by Craiu and Reiser (2006). The cause-specific hazards are identifiable but such models cannot estimate dependence between the competing risks. Quantifying the dependence between these risks may be of interest. This dependence can be modeled in different ways such as through frailties (Hougaard, 1986) or copula models (Genest and Nešlehová, 2006). Copula models define the joint distribution using a function of the marginal distributions of multiple random variables. A copula can also be defined in terms of the survival functions. If $T_1, ..., T_K$ are continuous random variables with marginal survival functions $S_1, .., S_K$ and joint survival function $\mathcal{S}$, a survival copula $\mathcal{C}$ is a function such that

$$S(t_1, ..., t_K) = \mathcal{C}\left(S_1(t_1), ..., S_K(t_K)\right)$$

Copula models allow for flexibility in the choice of margins and provide a way to qualify the dependence nature of a continuous random vector in terms of the underlying copula (Genest and

Nešlehová, 2006). Copulas have been utilized in the competing risks setting without joint longitudinal data. Lo and Wilke (2010) examines Archimedean copula model for three competing risks. Shih and Emura (2018) used the FGM (Farlie-Gumbel-Morgenstern) copula with Burr III marginals for competing risks data. Wu et al. (2017) used a Gumbel copula model in a competing risks setting which has a similar form to our copula model in Section 4.2 and with marginals similar to our Weibull model described in Section 4.3.

The desire to measure the dependence between risks motivates the use of a copula model for the competing risks part of our joint model. In Section 4.2 we describe the multivariate survival function based on an copula model that we will focus on in this work. We consider the competing risks data in a latent failure time framework. Competing risks models in this framework can suffer from identifiability issues. We discuss why the parameters of our models are identifiable in Section 4.2.2.

An additional reason to stray from the common cause-specific Cox-type competing risks model is that the data may not conform to the proportional hazards assumption. Accelerated failure time models have been used (Tseng et al., 2005; Hanson et al., 2011). Another option is to use parametric survival models. As discussed in the last chapter, parametric models can be more parsimonious than a proportional hazards model which often uses a piece-wise baseline hazard function (Furgal et al., 2019). In Section 4.3 we lay out the framework for our joint model. We use Weibull cumulative hazard functions as a specific case of the general copula model from Section 4.2. We use Bayesian Hamiltonian Monte Carlo to estimate the parameters in our model and describe the Bayesian estimation in Section 4.4.

Our competing risks submodel includes a parameter δ that measures the strength of the dependence between the causes of failure. In theory, this parameter can take a value in $(0, 1]$ where $\delta = 1$ reduces to a model with independent risks. Since our Bayesian model described in Section 4.4 puts a Beta prior on δ we also test for the case when $\delta = 1$ using Bayes factors. This is described in Section 4.5.

We apply our models to the ACC data in Section 4.6. We then evaluate our models via simula-

tion in Section 4.7. Section 4.8 concludes with a discussion.

4.2 Formulation of Dependence

We will develop our models for the competing risks data under a latent failure time framework similar to the previous chapter. We assume that each subject i ($i = 1, ..., N$) theoretically has an event time for each of the $K \geq 2$ risks. Let T_{ik}^* be the event time for subject i caused by risk k ($i = 1, ..., N$, $k = 1, ..., K$). Subjects may also be independently censored and we will call the censoring time T_{i0}^*. We only observe the minimum of these event times. Denote the observed time as $T_i = \min(T_{i0}^*, T_{i1}^*, ..., T_{iK}^*)$.

We will assume that the latent failure times have a joint survival function $S(t_1, ..., t_K)$ with the form in (4.1). This is a valid survival function for any arbitrary cumulative hazard functions $H_k(t)$ ($k = 1, ..., K$) and with $0 < \delta \leq 1$. (See Appendix H.2 for proof.)

$$S(t_1, ..., t_K) = \exp\left(-\left[H_1(t_1)^{\frac{1}{\delta}} + ... + H_K(t_K)^{\frac{1}{\delta}}\right]^\delta\right) \tag{4.1}$$

The multivariate survival function in (4.1) is similar to or an extension of models studied over the years (Hougaard, 1986; Lu and Bhattacharyya, 1990, 1991; Wang and Ghosh, 2000; Wang, 2012; Schwarz et al., 2013). Lu and Bhattacharyya (1990) motivate the bivariate version from the association between two component liftetimes with some common environmental stress. Another reason for using this survival function is the relative ease with which we can simulate data following that distribution. This is because random variables with the joint survival function in (4.1) can be written in terms of independent variables. This is discussed in Section 4.2.1.

The parameter δ measures the strength of the dependence between the latent failure times. Wang (2012) shows that for a version of the joint survival model above, Kendall's tau is equal to $1 - \delta$. Therefore $\delta = 1$ the model reduces to an independent failure time case.

If we use the relationship between the cumulative hazard and survival functions, $H_k(t_k) =$

$-\log(S_k(t_k))$, we can rewrite (4.1) as

$$S(t_1, ..., t_K) = \exp\left(-\left[(-\log(S_1(t_1)))^{\frac{1}{\delta}} + ... + (-\log(S_K(t_K)))^{\frac{1}{\delta}}\right]^{\delta}\right)$$

We see that this is an Archimedean survival copula with generator $\psi(t) = \exp(-t^{\delta})$. An Archimedean survival copula is of the form $S(t_1, ..., t_K) = \psi\left(\psi^{-1}(S_1(t_1)) + ... + \psi^{-1}(S_K(t_K))\right)$ for some generator function $\psi : [0, \infty) \to [0, 1]$ where $\psi(0) = 1$, $\psi(t) \to 0$ as $t \to \infty$, and ψ is K-monotone (Genest and Nešlehová, 2006; Jia, 2018). Archimedean copulas have become popular in applications in biostatistics (Tao et al., 2013; Suresh et al., 2019), insurance and finance (Savu and Trede, 2010; Li and Lu, 2019), engineering (Singh and Zhang, 2007; Fenech et al., 2015), and other fields (Zhang et al., 2012; Ayantobo et al., 2019) because of their simple form and connections to frailty models. Specifically, if the generator ψ is completely monotone, then the copula can be interpreted as the survival copula with lifetimes following a multiplicative hazard model with a frailty Z and ψ is the Laplace transform of Z (Genest and Nešlehová, 2006). Any Archimedean copula $\mathcal{C}^*(u_1, ..., u_K)$ is symmetric (i.e. $\mathcal{C}^*(u_1, u_2) = \mathcal{C}^*(u_2, u_1)$ for $K = 2$) and associative (i.e. $\mathcal{C}^*(u_1, ..., u_K) = \mathcal{C}^*(\mathcal{C}^*(u_1, .., u_{K-1}), u_K) = ... = \mathcal{C}^*(\mathcal{C}^*(...\mathcal{C}^*(\mathcal{C}^*(u_1, u_2), u_3), ..., u_{k-1}), u_K))$ meaning the dependence structure between all the random variables is the same (Lo and Wilke, 2010).

4.2.1 Representation Result

One reason for using the copula model in (4.1) is because of the relative ease with which we can simulate data having this joint survival function. Lee (1979) proved that in the bivariate case (i.e. $K = 2$) with Weibull marginal H_k functions, the dependent random variables T_1 and T_2 can be represented in terms of independent variables V_1 and V_2, where $V_2 \sim Beta(1, 1)$ and V_1 is a mixture of $\Gamma(2, 1)$ and $\Gamma(1, 1)$ variables. Here we extend this result for an arbitrary finite number of dependent random variables K and for arbitrary cumulative hazard functions H_k for $k = 1, ..., K$. Specifically we have the following result.

Theorem 1 *Let K random variables $T_1, ..., T_K$ have joint survival function*

$$S_T(t_1, t_2, ..., t_K) = \exp\left(-\left[H_1(t_1)^{\frac{1}{\delta}} + ... + H_n(t_K)^{\frac{1}{\delta}}\right]^{\delta}\right)$$

with $0 < \delta \leq 1$, and $H_1, ..., H_K$ arbitrary cumulative hazard functions.

Let $Z_k = H_k(T_k)^{\frac{1}{\delta}}$, $k = 1, ..., K$. Define the random variables $V_1 = (Z_1 + ... + Z_K)^{\delta}$, $V_2 = \frac{Z_1}{Z_1 + Z_2}$, $V_3 = \frac{Z_1 + Z_2}{Z_1 + Z_2 + Z_3}$, ..., $V_K = \frac{Z_1 + ... + Z_{K-1}}{Z_1 + ... + Z_{K-1} + Z_K}$.

Then $V_1, ..., V_K$ are independent with the following distributions: $V_2 \sim Beta(1, 1), ..., V_K \sim Beta(K-1, 1)$ and V_1 is a mixture of the Gamma distributions $\Gamma(K, 1)$, $\Gamma(K-1, 1)$, ..., $\Gamma(2, 1)$, $\Gamma(1, 1)$ with the following distribution

$$\begin{aligned}
f_{V_1}(v) = \frac{1}{\delta\Gamma(K)} \exp(-v)\Big(&a_{K,K}v^{K-1} + (-1)^{K-(K-1)}a_{K,K-1}v^{K-2} + ... \\
&+ (-1)^{K-k}a_{K,k}v^{k-1} + ... \\
&+ (-1)^{K-2}a_{K,2}v + (-1)^{K-1}a_{K,1}\Big)
\end{aligned}$$

where $\Gamma(k)$ is the Gamma function, i.e. $\Gamma(k) = \int_0^{\infty} x^{k-1}e^{-x}dx$, and the $a_{K,k}$ are defined recursively as follows

$$a_{1,1} = \delta;$$

$$a_{K,K} = \delta a_{K-1,K-1};$$

$$a_{K,1} = (\delta - (K-1))a_{K-1,1};$$

$$a_{K,k} = (k\delta - (K-1))a_{K-1,k} + \delta a_{K-1,k-1}, \textit{for } 2 \leq k \leq K-1.$$

The proof of this theorem is in Appendix H.

This can be used to simulate data by first drawing the independent $V_1, ..., V_K$ which is very simple for $V_2, ..., V_K$ and relatively easy for V_1. We can then calculate the Z_k given the following

80

relationships,

$$Z_1 = V_2 \cdots V_K V_1^{1/\delta}$$

$$Z_2 = (1 - V_2)V_3 \cdots V_K V_1^{1/\delta}$$

$$Z_3 = (1 - V_3)V_4 \cdots V_K V_1^{1/\delta}$$

$$...$$

$$Z_K = (1 - V_K)V_1^{1/\delta}$$

Then our dependent event times are $T_k^* = H_k^{-1}(Z_k^\delta)$. Since the H_k are cumulative hazard functions, the inverse function H_k^{-1} exists.

4.2.2 Competing Risks and Identifiability

While there are several nice properties of the survival copula in (4.1) and the representation from Section 4.2.1 allows for easy simulation, there may be concerns about identifiability. It is well known that Tsiatis showed competing risks models in a latent failure time framework are not identifiable with the identified minimum alone (Tsiatis, 1975). Specifically given any joint distribution for latent failure times there exists a distribution with independent failure times that gives the same identified minimum distribution (Heckman and Honorè, 1989). Heckman and Honorè demonstrate conditions under which including covariates as regressors circumvents non-identifiability for proportional hazards and accelerated failure time models. Zheng and Klein (1995) established that if the form of the copula is known, the marginal survival functions are identifiable from the identified minimum in the bivariate case and Carrière (1995) extended this to the case with more than two competing risks.

We chose to estimate using a Bayesian approach. Bayesian methods using both informative and non-informative priors are discussed by Wang and Ghosh (2000). While Heckman and Honorè prove identifiability for models of a certain form with a continuous covariate, Wang et al. (2015) provide conditions for identifiability in bivariate frailty models related to Achrimedean copulas

given a possibly discrete covariate. Escarela and Carrière (2003) also discuss bivariate competing risks models using an Archimedean copula and they propose using a fully parametric model with an assumed copula. Wang (2014) shows that assuming an Archimedean copula model, the marginal survival functions of dependent survival times are functionals of the Archimedean copula and therefore can be estimated. Also if one of the marginal functions is known, the joint survival function can be determined for many common families of Acrhimedean copulas.

In Section 4.3 we lay out the framework for our joint model. The survival submodels are based on the Archimedean copula model in (4.1) and we include a regression model with a continuous covariate. Given the results above, our the parameters in our model will be identifiable.

4.3   Framework

Assume we have N subjects and subject i has $J_i \geq 1$ longitudinal measurements for $i = 1, ..., N$. The observation times for subject i are τ_{ij} for $j = 1, ..., J_i$ with the longitudinal measurement at τ_{ij} denoted Y_{ij}. We will write the vector of longitudinal measurements as $Y_i = (Y_{i1}, ..., Y_{iJ_i})^T$ for subject i. For the competing risks data, each subject can have an event with one of $K \geq 2$ causes or may be right-censored. Denote the observed event time for subject i as T_i. We also have an event indicator matrix $D_i = (D_{i1}, ..., D_{ik}, ..., D_{iK})^T$ where $D_{ik} = 1$ if subject i had an event with cause k $(k = 1, ..., K)$ and $D_{ik} = 0$ otherwise. At most one of the $D_{i1}, ..., D_{iK}$ is equal to one; all D_{ik} will be 0 if the subject is censored. In addition we have covariate information. In the longitudinal model denote the covariate matrix by X_{ij} for subject i and it may depend on time j. We assume if these covariates depend on time they are exogenous (Rizopoulos, 2012, p.44). In the survival submodel let W_i be the covariate matrix which we assume is time-independent for simplicity. Theoretically W_i could depend on the event type k but again for simplicity we assume it is independent of k. The X_{ij} and W_i may but do not have to share covariates. Assume we have p covariates in the longitudinal model and q in the survival model. Therefore X_{ij} is a $J_i \times p$ matrix and W_i is a q-vector.

4.3.1 Longitudinal Submodel

We assume our longitudinal outcome Y_i is continuous and normally distributed. Hence we use a linear mixed effects model. Extension to a generalized linear mixed effects model for non-Gaussian outcomes is straightforward. We have subject-specific random effects denoted U_i with design matrix R_i, a p-vector of regression parameters β and random measurement errors ϵ_{ij}. Let $\epsilon_{ij} \sim \mathrm{N}(0, \sigma_\epsilon^2)$ and $\epsilon_i = (\epsilon_{i1}, ..., \epsilon_{iJ_i})^T$.

$$Y_i = X_i\beta + R_iU_i + \epsilon_i \tag{4.2}$$

Assume that the longitudinal measurements are independent given the random effects, i.e. $Y_i \perp Y_l | U_i, U_l, i \neq l$.

4.3.2 Competing Risks Survival Submodels

We will consider a specific case of (4.1) with cumulative hazard functions based on the common parametric survival function Weibull. The Weibull distribution is part of the log-location-scale family of distributions. This family has often been used to describe lifetimes (Hong et al., 2015). The cumulative distribution function (CDF) for distributions in this family can be written in the form $F(t; \mu, \gamma) = \Phi^* \left(\frac{log(t) - \mu}{\gamma} \right)$. Here μ is called the location parameter and γ the scale parameter. Φ^* is the standard CDF for the distribution (location = 0, scale = 1). We also chose to examine this distribution since Weibull is both a proportional hazards as well as an accelerated failure time model.

Define $H_k(t) = \mu_k t^{\gamma_k}$ giving us the following joint survival function.

$$S(t_1, ..., t_K) = \exp \left(- \left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + ... + (\mu_K t_K^{\gamma_K})^{\frac{1}{\delta}} \right]^\delta \right) \tag{4.3}$$

We put a regression model on the μ_k parameters.

$$\log(\mu_k) = W_i \alpha_k + \theta_k^T U_i$$

Here W_i is a vector of covariates, U_i is the same random effect as in the longitudinal model in (4.2), and θ_k measures the association between the longitudinal and survival data. Although we are considering a regression model on the location parameter μ_k, it possible to instead or additionally put a regression model on scale parameter γ_k or dependence parameter δ.

4.4   Bayesian Model and Estimation

We chose to estimate the parameters of our model with Bayesian techniques since this can be less computationally intensive than frequentist techniques which often require an EM algorithm. Additionally Bayesian models can account for additional variability. Fitting can be aided by incorporating any prior knowledge into the priors on the parameters.

We will use a random intercept only so $R_i = 1, i = 1, ..., N$ and we will write U_i as U_{0i} to make clear the random effect is a scalar and θ_k is also a scalar. We assume $U_{0i} \sim \mathrm{N}(0, \sigma_U^2)$. For our Bayesian model we put priors on our parameters.

$$\beta \sim \text{MVN}(m_\beta, s_\beta^2 I_p)$$
$$\sigma_U \sim \Gamma(a_U, b_U)$$
$$\sigma_\epsilon \sim \Gamma(a_\epsilon, b_\epsilon)$$
$$\gamma_k \sim \Gamma(a_\gamma, b_\gamma)$$
$$\alpha_k \sim \text{MVN}(m_\alpha, s_\alpha^2 I_q)$$
$$\theta_k \sim \text{N}(m_\theta, s_\theta^2)$$
$$\delta \sim \text{Beta}(a_\delta, b_\delta)$$

Since δ can be equal to 1, we will test for this case which is discussed in Section 4.5.

4.4.1 Likelihood and Posterior

Let $\Omega = (\beta, \sigma_U, \sigma_\epsilon, \alpha_1, ..., \alpha_K, \theta_1, ..., \theta_K, \gamma_1, ..., \gamma_K, \delta)^T$ be the vector of parameters. The posterior, $p(\Omega|Y, T, D, X, W)$, is proportional to the product of the likelihood and priors. We write the likelihood contribution from the longitudinal model as L_Y and from the survival model as L_T. Let $p(\Omega)$ denote the product of the prior density functions for the parameters in Ω.

$$p(\Omega|Y, T, D, X, W, U) \propto L_Y L_T p(U|\Omega) p(\Omega)$$

L_Y has the following form

$$
\begin{aligned}
L_Y &= \prod_{i=1}^{N} p(Y_i|X_i, \beta, U_{0i}, \sigma_U, \sigma_\epsilon) \\
&= \prod_{i=1}^{N} (2\pi\sigma_\epsilon)^{-\frac{J_i}{2}} \exp\left\{-\frac{1}{2\sigma_\epsilon^{J_i}} (Y_i - X_i\beta - U_{0i})^T (Y_i - X_i\beta - U_{0i})\right\}
\end{aligned}
$$

The likelihood contribution from the competing risks survival data has the form

$$
\begin{aligned}
L_T = \prod_{i=1}^{N} &\left(\prod_{k=1}^{K} h_k(T_i|D_{i,k} = 1, W_i, U_{0i}, \gamma_k, \alpha_k, \theta_k)^{D_{ik}}\right) \cdot \\
&S(T_i|D_i, W_i, U_{0i}, \gamma_1, ..., \gamma_K, \alpha_1, ..., \alpha_K, \theta_1, ..., \theta_K)
\end{aligned}
$$

where h_k and S are the cause-specific hazard function and joint survival function, respectively. We will write the functions for $K = 2$ risks. With the Weibull marginals, these functions are

$$h_k(t|D_{i,k} = 1, W_i, U_i, \gamma_1, \gamma_2, \alpha_1, \alpha_2, \theta_1, \theta_2) = \gamma_k \mu_k^{\frac{1}{\delta}} t^{\frac{\gamma_k}{\delta}-1} \left((\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t^{\gamma_2})^{\frac{1}{\delta}}\right)^{\delta-1}$$

$$S(t|D_i, W_i, U_i, \gamma_1, \gamma_2, \alpha_1, \alpha_2, \theta_1, \theta_2) = \exp\left(-\left((\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t^{\gamma_2})^{\frac{1}{\delta}}\right)^{\delta}\right)$$

Derivation of the hazard function h_k is in Appendix B.4. Full conditionals are listed in Appendix G.2. We fit our models using Hamiltonian Monte Carlo via Stan through the `R` package rstan version 2.21.2 and using `R` version 4.0.4 (Guo et al., 2020).

4.5 Testing for Independence

When introducing our general joint survival function in (4.1), we stated that the dependence parameter δ could take any value in $(0, 1]$. Our Bayesian model in Section 4.4 puts a Beta prior on δ which excludes the case when δ exactly equals 1, or the independent failure times case. We therefore need to test whether the data supplies strong evidence for the $\delta = 1$ case. To do this we use a Bayes factor (Kass and Raftery, 1995).

Kass and Raftery state that Bayes factors can be used to evaluate the evidence in favor of a null hypothesis. They explain Bayes factors as follows. We assume our data $\mathcal{D}$ arose under one of our two hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$. The data have probability density $Pr(\mathcal{D}|\mathcal{H}_0)$ or $Pr(\mathcal{D}|\mathcal{H}_1)$ and the hypotheses have a priori probabilities $Pr(\mathcal{H}_0)$ and $Pr(\mathcal{H}_1) = 1 - Pr(\mathcal{H}_0)$. Posterior probabilities induced by the data are $Pr(\mathcal{H}_0|\mathcal{D})$ and $Pr(\mathcal{H}_1|\mathcal{D})$. Using Bayes theorem we can see that the posterior odds is

$$\frac{Pr(\mathcal{H}_0|\mathcal{D})}{Pr(\mathcal{H}_1|\mathcal{D})} = \frac{Pr(\mathcal{D}|\mathcal{H}_0)}{Pr(\mathcal{D}|\mathcal{H}_1)} \frac{Pr(\mathcal{H}_0)}{Pr(\mathcal{H}_1)}$$

Transformation from the prior odds to the posterior odds is equivalent to multiplication by the term

$$B_{01} = \frac{Pr(\mathcal{D}|\mathcal{H}_0)}{Pr(\mathcal{D}|\mathcal{H}_1)}$$

which is called the Bayes factor.

In our situation the hypotheses are $\mathcal{H}_0 : \delta = 1$ and $\mathcal{H}_1 : 0 < \delta < 1$. We calculate the Bayes factor with these two hypotheses using the bridgesampling R package (Gronau et al., 2020). To do this we fit two Stan models to the data: (1) Model 1 as described in Section 4.4 assuming δ is random and in $(0, 1)$ and (2) Model 0 where δ is assumed fixed at 1. Documentation for the bridgesampling package recommends running models that will be used for testing for more iterations than are needed for only estimating the parameters. Therefore we run these models for 50000 iterations after 10000 warm-up iterations. The log marginal likelihoods are calculated using the `bridge_sampler` function from the bridgesampling package which follows

the bridge sampling described by Meng and Wong (Meng and Wong, 1996). Output from the `bridge_sampler` is supplied to the `bf` function to calculate the Bayes factor. Kass and Raftery say that the Bayes factor can be interpreted based on the following ranges. A Bayes factor of 1 to 3.2 is "not worth more than a bare mention", 3.2 to 10 is substantial, 10 to 100 is strong, and greater than 100 is decisive evidence in favor of $\mathcal{H}_0$.

4.6 Application to Adrenal Cancer Data

We apply our models to the 159 adrenocortical carcinoma (ACC) patients as in the previous chapter. Again we define our competing risks outcome to be time until first disease status change which can be either a progression or a regression. Death and loss to follow-up censor our outcome. While in theory death may not be independent of disease state change we group deaths with censoring events because there are very few patients that died before a progression or regression. Data used are the same as described in Table 3.1. We fit our models to this data with a longitudinal model for lean psoas muscle area (LPMA) (Y) including a fixed time slope and covariates for binary sex (X_1, reference male) and (centered) age in decades (X_2). In our survival submodel for the observed event time T_i we included age as the only covariate.

$$Y_i \sim \text{MVN} \left(\beta_0 + \beta_1 \tau_i + \beta_2 X_{1i} + \beta_3 X_{2i} + U_{0i}, \ \sigma_\epsilon^2 I_{J_i} \right)$$

$$T_i = t | D_{i,k} = 1 \text{ with survival function } S(t, ..., t) = \exp \left(- \left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + ... + (\mu_K t^{\gamma_K})^{\frac{1}{\delta}} \right]^\delta \right)$$

$$\text{with } \log(\mu_{ik}) = \alpha_{0k} + \alpha_{1k} X_{2i} + \theta_k U_{0i}$$

$$(4.4)$$

Define $\beta = (\beta_0, \beta_1, \beta_2, \beta_3)^T$ and $\alpha_k = (\alpha_{0k}, \alpha_{1k})^T$ for $k = 1, 2$. We set priors on the parameters as in Section 4.4 with the following hyperparameters chosen so that our priors are weakly informative: $m_\beta = (1, 1, 1, 1)^T$, $s_\beta = 2$, $a_U = a_\epsilon = a_\gamma = 3$, $b_U = b_\epsilon = b_\gamma = 2$, $m_\alpha = (-1, 1)^T$, $s_\alpha = 2$, $m_\theta = (1, -1)^T$, $s_\theta = 2$, $a_\delta = 1$, and $b_\delta = 1$.

To test the sensitivity of the estimates for the δ parameter we also fit the model with different

hyperparameters for the Beta prior. Priors for all other parameters were the same as above. We fit four additional models. The Beta(0.03,0.07) and Beta(0.05,0.05) priors have density concentrated near 0 and 1. The other two priors used are Beta(3,7) and Beta(5,5) and have densities centered near 0.3 and 0.5. Figures 4.1 - 4.5 show histograms of the Beta priors.

As in the previous chapter, we fit the cause-specific proportional hazards (PH) model in 4.5 for comparison. A Weibull baseline hazard was used for simplicity. In this case the risks are independent and so there is no δ parameter.

$$
\begin{aligned}
Y_i &\sim \mathrm{N}\left(\beta_0 + \beta_1 \tau_i + \beta_2 X_{1i} + \beta_3 X_{2i} + U_i, \ \sigma_\epsilon^2 I_{J_i}\right) \\
h_k(T_i) &= \frac{\gamma_k}{\alpha_{0k}} T_i^{\gamma_k - 1} \exp\left(\alpha_{1k} X_{2i} + \theta_k U_{0i}\right)
\end{aligned}
\tag{4.5}
$$

This model is fit in Stan with the same priors as in Section 4.4. The likelihood contributions have the same form as in Section 4.4.1 with the survival functions below.

$$
h_k(T_i | D_{i,k} = 1, W_i, U_i, \gamma_1, \gamma_2, \alpha_1, \alpha_2, \theta_1, \theta_2) = \frac{\gamma_k}{\alpha_{0k}} T_i^{\gamma_k - 1} \exp\left(\alpha_{1k} X_{2i} + \theta_k U_{0i}\right)
$$

$$
\begin{aligned}
S(T_i | D_i, W_i, U_i, \gamma_1, \gamma_2, \alpha_1, \alpha_2, \theta_1, \theta_2) &= \\
&\prod_{k=1}^{2} \exp\left(-\int_0^{T_i} \frac{\gamma_k}{\alpha_{0k}} s^{\gamma_k - 1} \exp\left(\alpha_{1k} X_{2i} + \theta_k U_{0i}\right) ds\right) \\
&= \exp\left(-\frac{1}{\alpha_{0,1}} T_i^{\gamma_1} \exp\left(\alpha_{1,1} X_{2i} + \theta_1 U_{0i}\right) - \right. \\
&\qquad \left. \frac{1}{\alpha_{0,2}} T_i^{\gamma_2} \exp\left(\alpha_{1,2} X_{2i} + \theta_2 U_{0i}\right)\right)
\end{aligned}
$$

4.6.1 Results

Our models were fit using Stan with four chains with different initial values (see Appendix Section F.2). The chains were run with 10000 warm-up iterations and an additional 1000 iterations after warm-up. Trace plots are in Appendix F.3. Table 4.1 shows the results. We tested for $\delta = 1$ as described in Section 4.5 and found a Bayes factor of 1.62 in favor of $\mathcal{H}_0 : \delta = 1$ over $\mathcal{H}_1 : 0 <$

Parameter	Mean	CI	Pr > 0
β_0	3.45	(2.32, 4.50)	1.00
β_1	-0.14	(-0.33, 0.05)	0.07
β_2	-5.95	(-7.33, -4.52)	0.00
β_3	-1.41	(-1.97, -0.85)	0.00
σ_U	4.31	(3.79, 4.89)	1.00
σ_ϵ	2.04	(1.86, 2.24)	1.00
$\alpha_{0,1}$	-1.51	(-1.85, -1.20)	0.00
$\alpha_{1,1}$	0.05	(-0.12, 0.22)	0.74
$\alpha_{0,2}$	-1.98	(-2.53, -1.42)	0.00
$\alpha_{1,2}$	0.10	(-0.12, 0.32)	0.83
θ_1	-0.04	(-0.11, 0.02)	0.12
θ_2	0.00	(-0.07, 0.09)	0.53
γ_1	0.74	(0.64, 0.85)	1.00
γ_2	0.53	(0.41, 0.66)	1.00
δ	0.61	(0.14, 0.99)	1.00
DIC	-1168.6	-1191.3; -1147.1	

Table 4.1: Posterior Mean (Mean), 95% Credible Interval (CI) and probability the parameter is greater than 0 (Pr > 0) for the model fit to the ACC data. The bottom row contains the mean and 2.5% and 97.5% quantiles for the DIC. Here $a_\delta = 1$ and $b_\delta = 1$.

$\delta < 1$. This is not strong evidence in favor of $\delta = 1$.

The results in Table 4.1 and show a slight negative time slope in the longitudinal model (β_1). The covariates were found to have significant negative coefficients with females and older patients having lower LPMA measurements (β_1, and β_2, resp.). Age in the survival submodels ($\alpha_{1,1}$, $\alpha_{1,2}$) had a small positive effect and the majority of estimates were greater than zero. We find a small negative association between U_{0i} and progression (θ_1). There does not seem to be any association between regression and U_{0i} ($\theta_2 \approx 0$). The model estimates δ to be about 0.6, although the credible interval includes a large range of the possible values. The Bayes factor of 1.62 suggests that there is no evidence for independence in this case.

Comparing to the model assuming independence in Chapter 3 (Table 3.2) we see that the results for the longitudinal model (β, σ_U, σ_ϵ) and the Weibull shapes (γ_1, γ_2) are very similar. Interestingly,

when accounting for dependence between the risks in this chapter, we find different results for the competing risks model than in Table 3.2. The estimates of $\alpha_{0,1}$ and $\alpha_{0,2}$ are much closer in value than in the previous chapter. Here we find a small positive association for age for both progression and regression ($\alpha_{1,1}$ and $\alpha_{1,2}$, resp.) Finally now the estimate for the association between U_{0i} and progression θ_1 is negative. Since we found evidence in favor of dependence with the small Bayes factor and δ estimate different from 1, we would argue that the dependence should be included in the model.

Results for the models fit with different hyperparameters for the prior on δ are shown in Table 4.2, histograms of the estimates are shown in Figures 4.1 - 4.5 and trace plots are in Appendix Figures F.7 - F.11. We see that the estimates for the parameters other than δ are consistent but the estimates for δ differ depending on the prior. Figures 4.1 - 4.5 show that the posterior distributions of θ_k are similar but not exactly the same as the priors. This suggests that the data does not supply much information about δ. Based on DIC, the model with a Beta(5,5) has the best fit to the data. Bayes factors for all models are in Table 4.3 and we see that there is not strong evidence for $\delta = 1$ with any priors except Beta(0.03,0.07). The model with a Beta (0.03,0.07) has a Bayes factor of 5.23 which Kass and Raftery classify as substantial evidence in favor of $\delta = 1$. Comparing DIC, the model with a Beta(5,5) prior has the best fit to the data.

Table 4.4 shows the results for the cause-specific PH model. Estimates for the longitudinal submodel parameters are similar to our model. The α_{0k} parameters in the survival submodels have different estimates but these are also included in each model differently and therefore are not directly comparable. The α_{1k} and θ_k parameters effect the μ_k in our models and effect the hazard in the cause-specific PH model. Still we see similar estimates in our models and the cause-specific PH model. Based on DIC the cause-specific PH model fits the data slightly better than any of our models.

90

	Beta(0.03,0.07)			Beta(0.05,0.05)		
Parameter	Mean	CI	Pr > 0	Mean	CI	Pr > 0
β_0	3.44	(2.35, 4.55)	1.00	3.47	(2.44, 4.57)	1.00
β_1	-0.15	(-0.34, 0.04)	0.06	-0.15	(-0.32, 0.04)	0.06
β_2	-5.94	(-7.33, -4.55)	0.00	-5.92	(-7.31, -4.61)	0.00
β_3	-1.42	(-1.95, -0.86)	0.00	-1.42	(-1.96, -0.81)	0.00
σ_U	4.32	(3.80, 4.89)	1.00	4.31	(3.78, 4.91)	1.00
σ_ϵ	2.04	(1.87, 2.23)	1.00	2.03	(1.85, 2.24)	1.00
$\alpha_{0,1}$	-1.68	(-1.95, -1.40)	0.00	-1.66	(-1.95, -1.35)	0.00
$\alpha_{1,1}$	0.04	(-0.14, 0.22)	0.68	0.05	(-0.13, 0.23)	0.69
$\alpha_{0,2}$	-2.32	(-2.72, -1.86)	0.00	-2.29	(-2.70, -1.68)	0.00
$\alpha_{1,2}$	0.12	(-0.14, 0.39)	0.82	0.12	(-0.14, 0.38)	0.83
θ_1	-0.05	(-0.12, 0.02)	0.08	-0.05	(-0.12, 0.02)	0.08
θ_2	0.02	(-0.07, 0.10)	0.66	0.02	(-0.07, 0.11)	0.62
γ_1	0.75	(0.64, 0.88)	1.00	0.75	(0.64, 0.87)	1.00
γ_2	0.52	(0.39, 0.66)	1.00	0.52	(0.39, 0.66)	1.00
δ	0.95	(0.51, 1.00)	1.00	0.92	(0.30, 1.00)	1.00
DIC	-1169.1	-1192.4; -1148.1		-1168.0	-1190.6; -1146.7	

|	Beta(3,7)			Beta(5,5)		
Parameter	Mean	CI	Pr > 0	Mean	CI	Pr > 0
β_0	3.43	(2.35, 4.49)	1.00	3.41	(2.37, 4.46)	1.00
β_1	-0.14	(-0.33, 0.05)	0.07	-0.14	(-0.32, 0.05)	0.07
β_2	-5.94	(-7.29, -4.55)	0.00	-5.93	(-7.26, -4.62)	0.00
β_3	-1.40	(-1.98, -0.82)	0.00	-1.41	(-1.96, -0.83)	0.00
σ_U	4.31	(3.81, 4.91)	1.00	4.31	(3.80, 4.88)	1.00
σ_ϵ	2.04	(1.87, 2.25)	1.00	2.04	(1.86, 2.24)	1.00
$\alpha_{0,1}$	-1.39	(-1.64, -1.16)	0.00	-1.46	(-1.72, -1.23)	0.00
$\alpha_{1,1}$	0.06	(-0.10, 0.22)	0.80	0.06	(-0.10, 0.21)	0.77
$\alpha_{0,2}$	-1.72	(-2.13, -1.39)	0.00	-1.90	(-2.31, -1.53)	0.00
$\alpha_{1,2}$	0.08	(-0.09, 0.25)	0.81	0.09	(-0.10, 0.29)	0.82
θ_1	-0.04	(-0.10, 0.03)	0.13	-0.04	(-0.10, 0.03)	0.12
θ_2	-0.01	(-0.07, 0.06)	0.43	0.00	(-0.07, 0.07)	0.52
γ_1	0.72	(0.63, 0.82)	1.00	0.73	(0.63, 0.84)	1.00
γ_2	0.55	(0.44, 0.67)	1.00	0.54	(0.42, 0.66)	1.00
δ	0.35	(0.13, 0.64)	1.00	0.51	(0.24, 0.80)	1.00
DIC	-1171.4	-1193.6; -1150.3		-1173.5	-1197.1; -1152.0	

Table 4.2: Results with different hyperparameters for delta. Posterior Mean (Mean), 95% Credible Interval (CI) and probability the parameter is greater than 0 (Pr > 0) for the model fit to the ACC data. The bottom row contains the mean and 2.5% and 97.5% quantiles for the DIC.

Prior	Bayes Factor
Beta(1,1)	1.62
Beta(0.03,0.07)	5.23
Beta(0.05,0.05)	2.30
Beta(3,7)	2.44
Beta(5,5)	2.14

Table 4.3: Bayes factors for models fit with differing priors on delta.

Parameter	Mean	CI	Pr > 0
β_0	3.42	(2.33, 4.45)	1.00
β_1	-0.15	(-0.34, 0.03)	0.04
β_2	-5.96	(-7.28, -4.56)	0.00
β_3	-1.42	(-1.96, -0.85)	0.00
σ_U	4.32	(3.79, 4.93)	1.00
σ_ϵ	2.04	(1.86, 2.24)	1.00
$\alpha_{0,1}$	4.97	(3.93, 6.22)	1.00
$\alpha_{1,1}$	0.05	(-0.14, 0.23)	0.68
$\alpha_{0,2}$	7.31	(5.75, 9.13)	1.00
$\alpha_{1,2}$	0.11	(-0.11, 0.32)	0.81
θ_1	-0.05	(-0.12, 0.02)	0.07
θ_2	0.01	(-0.07, 0.08)	0.57
γ_1	0.73	(0.62, 0.85)	1.00
γ_2	0.48	(0.36, 0.60)	1.00
δ	NA	NA	NA
DIC	-1183.8	-1207.4; -1163.3	

Table 4.4: Posterior Mean (Mean), 95% Credible Interval (CI) and probability the parameter is greater than 0 (Pr > 0) for the cause-specific proportional hazards model fit to the ACC data. The bottom row contains the mean and 2.5% and 97.5% quantiles for the DIC. Here $a_\delta = 1$ and $b_\delta = 1$.
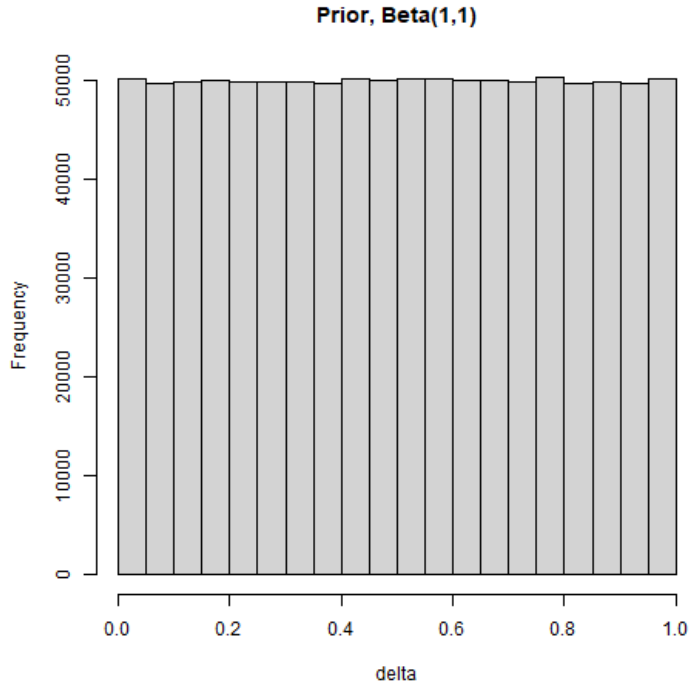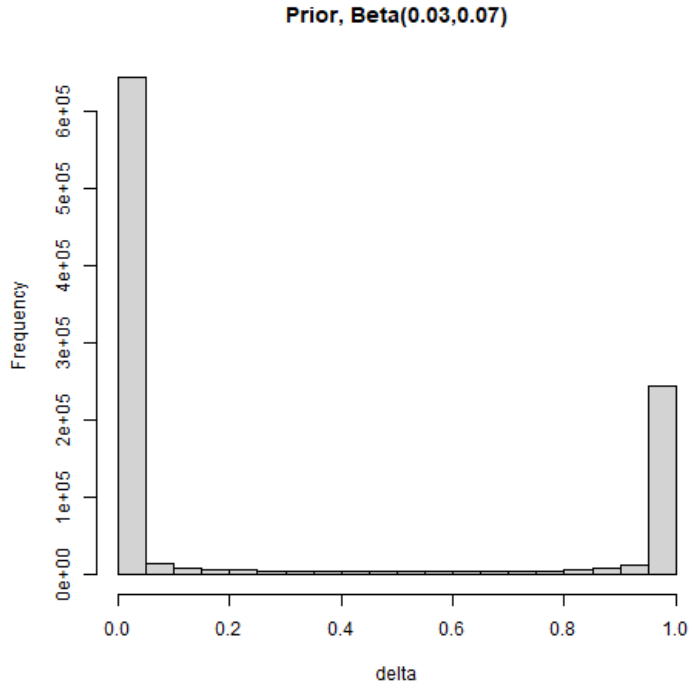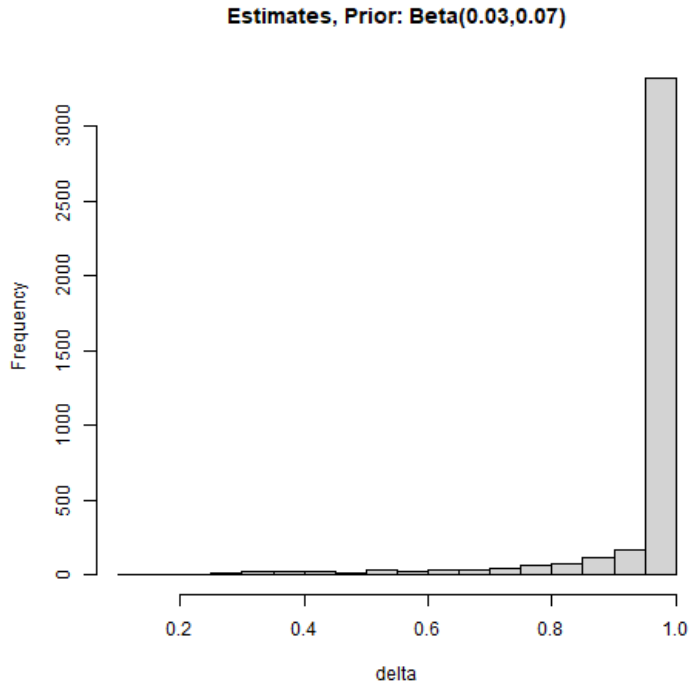
(a)
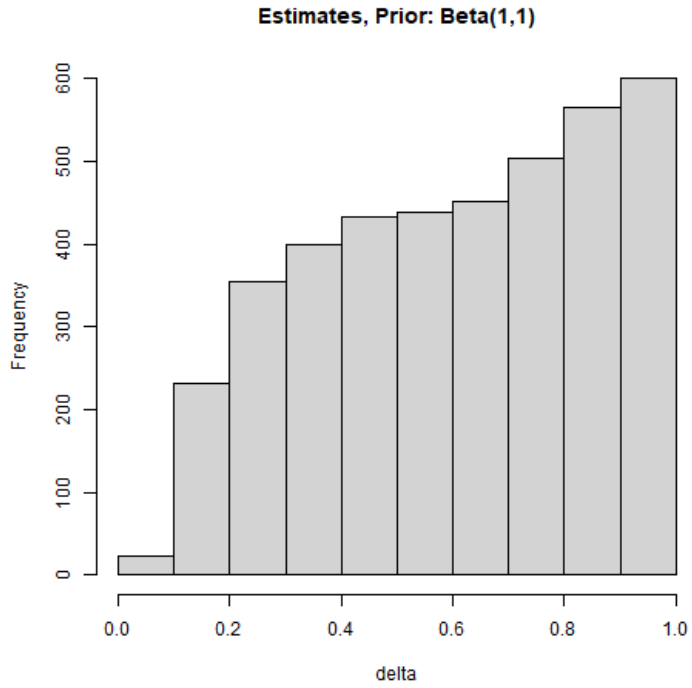
(b)

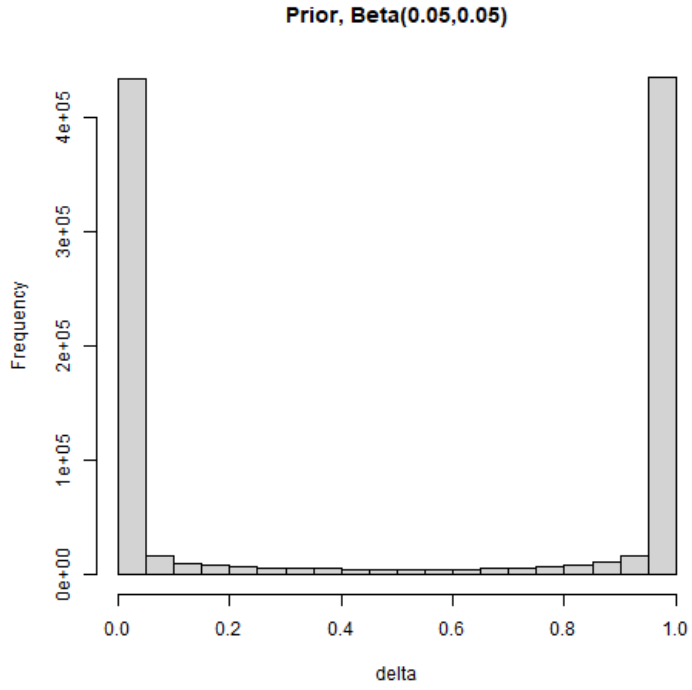Figure 4.1: Prior (a) and estimates (b) from model fit with a Beta(1,1) prior.

Figure 4.2: Prior (a) and estimates (b) from model fit with a Beta(0.03,0.07) prior.

Prior, Beta(0.05,0.05)

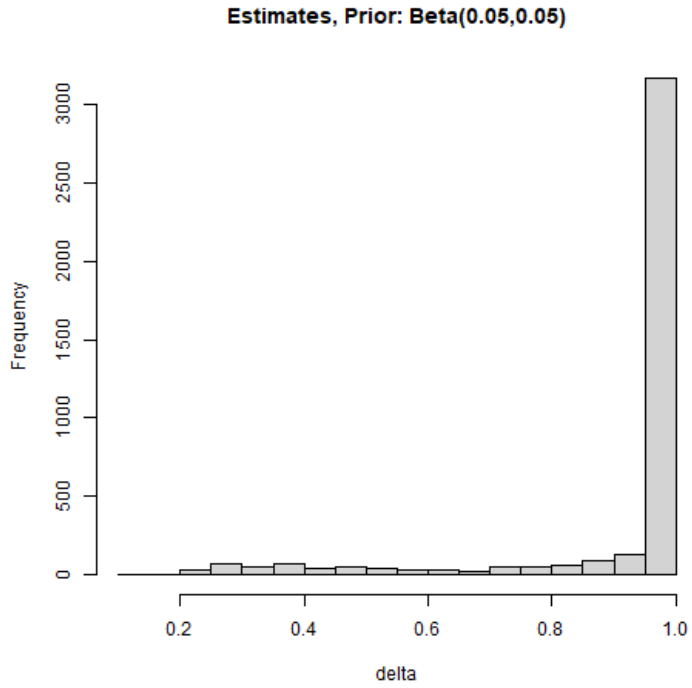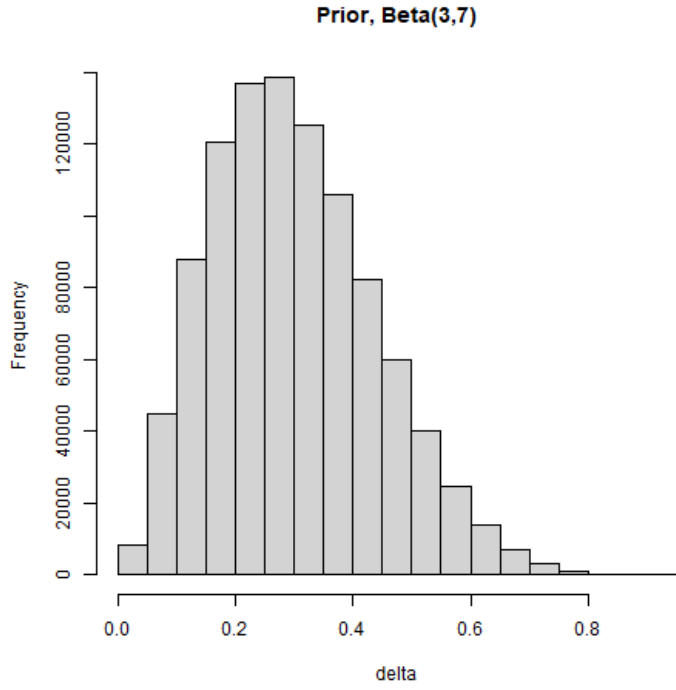(a)

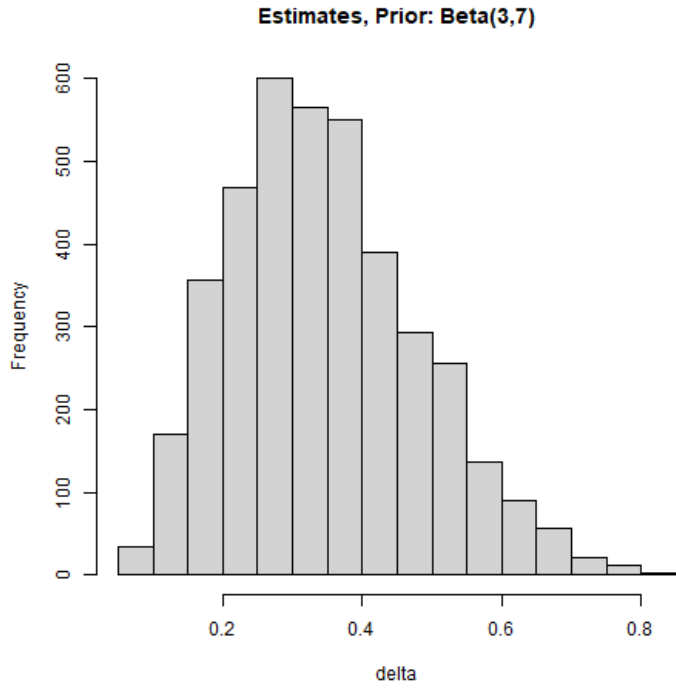Estimates, Prior: Beta(0.05,0.05)

(b)

Figure 4.3: Prior (a) and estimates (b) from model fit with a Beta(0.05,0.05) prior.
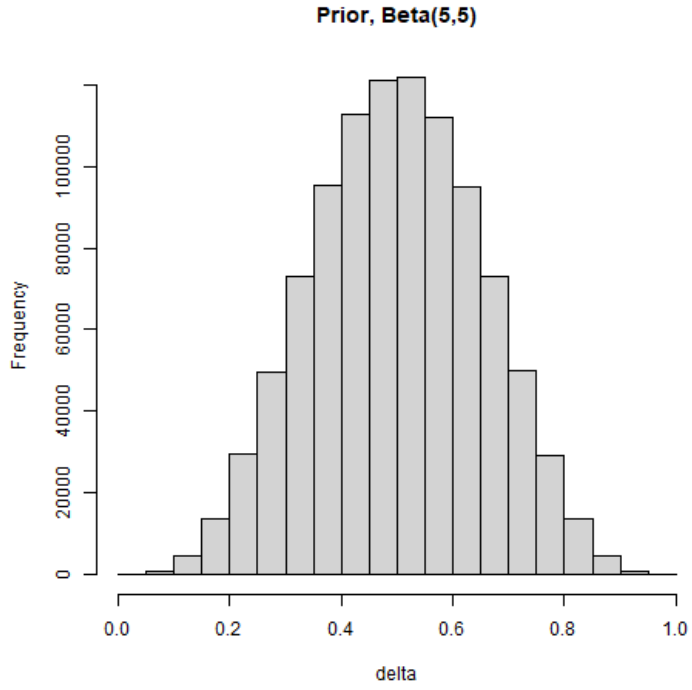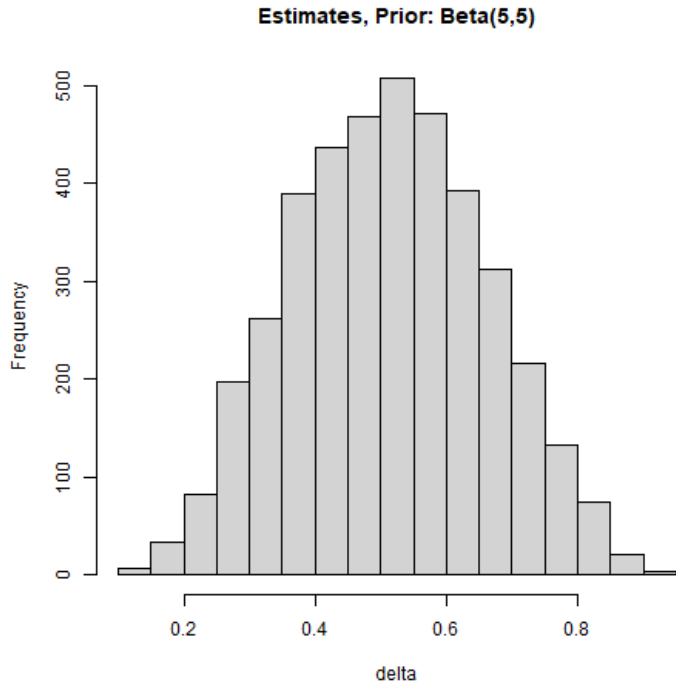
(a)

(b)

Figure 4.4: Prior (a) and estimates (b) from model fit with a Beta(3,7) prior.

(a)

(b)

Figure 4.5: Prior (a) and estimates (b) from model fit with a Beta(5,5) prior.

4.7 Simulations

We test our model with a simulation study by generating data to be similar to the ACC data. We ran simulations for two sample sizes $N = 160$ as in the ACC data and $N = 1000$. Each subject has two covariates a binary $X_{1i} \sim \text{Bin}(0.57)$ similar to sex, and continuous $X_{2i} \sim \text{N}(0, 1.5^2)$ similar to centered age in decades. We draw a random intercept $U_{0i} \sim \text{N}(0, 4.3^2)$. We calculate longitudinal measurements starting at time 0 and then every 0.125 years until 3 years, then every 0.5 years until 5 years and every 1 year until 8 years, so $\tau = (0, 0.125, 0.25, ..., 2.875, 3, 3.5, 4, 4.5, 5, 6, 7, 8)^T$. These measurements times were chosen to increase the likelihood that subjects with a short event time still have more than one observation and the time between observations increases to represent that patients have less frequent scans as time since diagnosis increases. Random measurement errors are drawn as $\epsilon_{ij} \sim \text{N}(0, 2^2)$ and the longitudinal outcome for subject i at time point j is

$$Y_{ij} = \beta_0 + \beta_1 \tau_j + \beta_2 X_{1i} + \beta_3 X_{2i} + U_{0i} + \epsilon_{ij}$$

True values for β are set to be $\beta_0 = 3.4$, $\beta_1 = -0.15$, $\beta_2 = -6$, $\beta_3 = -1.5$.

For the competing risks data we generate times for $K = 2$ risks with Weibull marginals. We calculate μ_{ik} as

$$\mu_{ik} = \exp(\alpha_{0k} + \alpha_{1k} X_{2i} + \theta_k U_{0i})$$

The true values used in the regression model are $\alpha_1 = (-1.5, 0.1)^T$, $\alpha_2 = (-2, 0.2)^T$, $\theta_1 = -0.4$, $\theta_2 = 0.5$, $\gamma_1 = 0.75$, and $\gamma_2 = 0.5$. We also generate data for different levels of dependence with $\delta = 0.5, 0.8,$ or 1.

In order to calculate the event times, we utilize the representation from the Theorem in Section 4.2.1. We can draw $V_2 \sim Beta(1, 1)$ and independently draw V_1 from a mixture of $\Gamma(1, 1)$ and $\Gamma(2, 1)$ distributions with mixture proportion δ. We can calculate $Z_1 = V_1^{\frac{1}{\delta}} V_2$ and $Z_2 = V_1^{\frac{1}{\delta}} (1 - V_2)$. Then our dependent event times are $T_k = H_k^{-1}(Z_k^\delta)$, $k = 1, 2$ where the H_k is defined in Section 4.3 and derivations for the H_k^{-1} function is in Appendix B.5.

We fit models as described in Section 4.4. Hyperparameters were the same as in Section 4.6. We also tested for the $\delta = 1$ case as detailed in Section 4.5 for 10 datasets. The small number of datasets tested was due to the additional computational resources required.

4.7.1 Results

True Value	N	Param	Mean	Bias*1000	MSE*1000	CP	N	Param	Mean	Bias*1000	MSE*1000	CP
3.4	160	β_0	3.19	-214	218	88	1000	β_0	3.35	-53	31	92
-0.15		β_1	-0.15	0	3	96		β_1	-0.15	0	1	98
-6		β_2	-5.76	237	193	86		β_2	-5.96	43	22	95
-1.5		β_3	-1.44	61	62	94		β_3	-1.48	22	9	93
4.3		σ_U	4.22	-84	76	94		σ_U	4.28	-22	13	94
2		σ_ϵ	2.00	1	1	94		σ_ϵ	2.00	-1	0	93
0.1		$\alpha_{1,1}$	0.08	-20	21	92		$\alpha_{1,1}$	0.09	-11	3	94
0.2		$\alpha_{1,2}$	0.27	73	40	88		$\alpha_{1,2}$	0.21	11	4	92
-0.4		θ_1	-0.45	-46	7	92		θ_1	-0.41	-10	1	93
0.5		θ_2	0.58	82	18	87		θ_2	0.51	9	1	95
0.75		γ_1	0.81	62	12	91		γ_1	0.76	12	1	92
0.5		γ_2	0.55	52	8	89		γ_2	0.51	6	1	95
0.5		δ	0.55	54	26	98		δ	0.52	17	9	95

Table 4.5: Simulation results for data generated with $\delta = 0.5$. Data was generated for either N=160 (left side) or N=1000 (right side). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP).

We used Stan to fit the models with four chains each with 10000 burn-in iterations and 1000 additional iterations after burn-in. Tables 4.5, 4.6 and 4.7 show the results for $\delta = 0.5$, $\delta = 0.8$ and $\delta = 1$, respectively. Results from the boundary testing are shown in Table 4.8. Our results show small bias and MSE for all parameters as well as good coverage for $N = 160$ and the results are even better for $N = 1000$. We see similar results for the $\delta = 0.8$ and $\delta = 1$ cases. When testing for independence, we find essentially no evidence for independence in the $\delta = 0.5$ simulations since the maximum Bayes factor value found was 2.37 with $N = 160$, below the threshold of 3.2 set by Kass and Raftery (1995). In the $\delta = 0.8$ case, some of the simulations found "substanial" evidence of independence with a maximum Bayes factor of 5.45 with $N = 160$ and 9.17 with $N = 1000$ but the average Bayes factor value with $N = 1000$ was below the 3.2 threshold. In the $\delta = 1$ case

True Value	N	Param	Mean	Bias*1000	MSE*1000	CP	N	Param	Mean	Bias*1000	MSE*1000	CP
3.4	160	β_0	3.16	-236	222	89	1000	β_0	3.36	-44	30	94
-0.15		β_1	-0.15	3	4	94		β_1	-0.15	2	1	98
-6		β_2	-5.73	266	224	86		β_2	-5.97	30	21	98
-1.5		β_3	-1.44	59	61	94		β_3	-1.48	24	9	92
4.3		σ_U	4.21	-89	71	95		σ_U	4.28	-23	13	94
2		σ_ϵ	2.00	1	2	94		σ_ϵ	2.00	1	0	94
0.1		$\alpha_{1,1}$	0.08	-24	19	94		$\alpha_{1,1}$	0.10	-5	2	96
0.2		$\alpha_{1,2}$	0.25	45	30	94		$\alpha_{1,2}$	0.22	16	4	94
-0.4		θ_1	-0.41	-13	5	96		θ_1	-0.41	-7	1	95
0.5		θ_2	0.54	39	11	90		θ_2	0.51	9	1	96
0.75		γ_1	0.78	34	7	96		γ_1	0.76	10	1	96
0.5		γ_2	0.54	40	7	92		γ_2	0.51	8	1	96
0.8		δ	0.73	-72	26	98		δ	0.80	301	96	99

Table 4.6: Simulation results for data generated with $\delta = 0.8$. Data was generated for either N=160 (left side) or N=1000 (right side). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP).

we should find evidence for independence. We see that while some of the simulations did not find evidence for independence with $N = 160$ (given the minimum Bayes factor of 1.05) the average Bayes factor with $N = 160$ shows "substantial" evidence but the average with $N = 1000$ shows "strong evidence" (Bayes factor of 10 to 100).

4.8   Discussion

In this chapter we propose jointly modeling longitudinal and competing risks data with survival submodels from a multivariate copula. We explored a joint survival copula model with a general structure and then in the specific cases with cumulative hazard functions from a Weibull distribution. This model is an alternative to the often used cause-specific proportional hazards models in competing risks settings. We estimate via a Bayesian approach that makes use of the hierarchical structure of the model. Bayesian techniques also allow us to incorporate prior information, if available, which can help alleviate some identifiability issues and are useful in joint modeling which can be numerically challenging in a frequentist setting.

We examined performance of the models via a simulation study. Simulations showed generally good performance even with a relatively small sample size of 160. Testing for independence with

True Value	N	Param	Mean	Bias*1000	MSE*1000	CP	N	Param	Mean	Bias*1000	MSE*1000	CP
3.4	160	β_0	3.14	-251	231	90	1000	β_0	3.34	-60	35	92
-0.15		β_1	-0.15	-1	4	93		β_1	-0.15	0	1	94
-6		β_2	-5.70	298	250	84		β_2	-5.95	50	29	93
-1.5		β_3	-1.44	64	59	94		β_3	-1.47	23	10	92
4.3		σ_U	4.20	-101	76	94		σ_U	4.27	-28	13	90
2		σ_ϵ	2.00	3	2	96		σ_ϵ	2.00	-1	0	94
0.1		$\alpha_{1,1}$	0.09	-14	16	94		$\alpha_{1,1}$	0.09	-6	3	94
0.2		$\alpha_{1,2}$	0.24	44	28	94		$\alpha_{1,2}$	0.21	7	4	96
-0.4		θ_1	-0.39	12	4	96		θ_1	-0.39	8	1	94
0.5		θ_2	0.51	15	9	92		θ_2	0.49	-11	1	95
0.75		γ_1	0.76	8	7	92		γ_1	0.75	-2	1	96
0.5		γ_2	0.54	35	7	92		γ_2	0.50	2	1	96
1		δ	0.80	-196	48	NA		δ	0.91	-90	10	NA

Table 4.7: Simulation results for data generated with $\delta = 1$. Data was generated for either N=160 (left side) or N=1000 (right side). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP). Note that coverage probabilities were not included for δ since the model uses a Beta prior and so will not include the true value of 1.

	N	Mean	Min	Max	N	Mean	Min	Max
$\delta = 0.5$	160	0.82	0.02	2.37	1000	0.02	0.00	0.09
$\delta = 0.8$	160	3.30	0.05	5.45	1000	2.17	0.05	9.17
$\delta = 1$	160	5.04	1.05	10.75	1000	11.45	3.32	22.12

Table 4.8: Simulation results Bayes factors testing $\delta = 1$.

Bayes factors did show stronger evidence in favor of independence when $\delta = 1$ as expected. We also illustrated the technique by applying the joint model to the ACC data. We did not find a significant association between the longitudinal LPMA and time to either disease state change but there was evidence of dependence between the risks.

We explored the sensitivity of δ to the priors using the ACC data and found that the data may not supply much information about the dependence parameter. In the simulations using the Beta(1,1) prior we were able to accurately estimate the δ parameter when δ was 0.5 and 0.8 with small bias and MSE. A larger bias was found when $\delta = 1$. Additional sensitivity analyses to examine the prior influence on δ in the simulations would be useful. The δ parameter may only be "weakly" identifiable (Cao et al., 2013; Ho and Nguyen, 2016) and further study is needed.

We used Bayes Factors to test for independence between the competing risks, meaning $\delta = 1$.

The model we fit assumes that δ is actually between 0 and 1, not including 1. Another option would be to allow δ to be exactly equal to 1 in the fitted model. This can be done by including $\delta = 1$ in the prior such as by using a spike-and-slab prior.

We applied our model with Weibull marginals. The Weibull distribution is a popular and simple distribution which can be interpreted in both proportional hazards and accelerated failure time form. Another option could be the log-Normal distribution which is popular in accelerated failure time models. Both of these examples have ties to accelerated failure time models which are the main alternative to the common proportional hazards models. Our model is developed such that any cumulative hazard function could be substituted depending on the application. Also, while we assumed that the cumulative hazard functions for each latent failure time were of the same form with differing parameters, in theory we could use a different parametric form for each time. This would complicate the likelihood derivation but adds flexibility.

In this work we considered the time to first disease progression or regression as a competing risks endpoint. In fact, our motivating data from the ACC study includes more information on disease states. Specifically, at each timepoint we know if the patient's disease is stable, progressing or regressing, as well as time of death or censoring. This is multistate data (Hougaard, 1999) and incorporation of this type of data into joint models is beginning to be studied (Dantan et al., 2011; Hu et al., 2012; Cai et al., 2017; Król et al., 2016; Ferrer et al., 2016; Mwanyekange et al., 2019; Dessie et al., 2020). In addition, there are multiple longitudinal morphomics variables that were measured at each scan. It is possible to incorporate more than one longitudinal submodel into a joint model with a single survival endpoint (Hatfield et al., 2011; Tang and Tang, 2015; Hickey et al., 2016; Mauff et al., 2020), competing risks (Andrinopoulou et al., 2013), or recurrent events (Musoro et al., 2015). To the best of our knowledge combining both multiple longitudinal and multi-state models has been explored only with a two-stage estimation technique (Alafchi et al., 2021). The next chapter extends this model to a joint longitudinal and multistate model to fully

understand the relationship between ACC disease and body morphomics.

Chapter 5

Bayesian Joint Models for Multiple Longitudinal Outcomes and Multi-State Survival Data

5.1 Introduction

Our motivating data from adrenocortical carcinoma (ACC) patients at the University of Michigan includes more information than has been utilized in our earlier chapters. To start, information is available on disease states over time, more than just the time to first of progression or regression. We have information on whether the disease was stable, progressive or regressive at each longitudinal timepoint as well as if the individual died or was censored. This falls into the category of multistate data (Hougaard, 1999; Andersen and Keiding, 2002). Multi-state survival modeling has recently been incorporated in joint models. Alafchi et al. (2021) developed a joint model for multiple longitudinal processes and multi-state survival data which was estimated in two-stages. Implementations which estimate using the full joint likelihood exclusively consider a single longitudinal biomarker and generally use a proportional hazards format for the multistate hazards. This includes Dantan et al. (2011) who modeled a longitudinal biomarker and an illness-death multi-state model with latent state transitions. Ferrer et al. (2016) studied prostate cancer using a joint linear mixed model and Markov multistate model. Dessie et al. (2020) studied viral load in HIV patients with a similar model. Musoro et al. (2015) developed a joint model for two longitudinal

biomarkers and two types of repeated events. Mwanyekange et al. (2019) describe a Bayesian joint model for a single longitudinal process and a recurrent event with up to 3 recurrences plus a terminal event using a Semi-Markov model. Other studies of joint models for a longitudinal process, recurrent events and possibly a terminal event include Cai et al. (2017) using an accelerated failure time model, Han et al. (2007) and Król et al. (2016).

In the ACC data multiple morphomics variables were repeatedly measured. These could be related to the survival process in different ways and so it may be useful to include more than one longitudinal outcome in the model. This has received some limited attention when joint modeling with multistate data as described above. More work has been done on including multiple longitudinal outcomes with a single survival time (Hatfield et al., 2011; Baghfalaki et al., 2014; Tang and Tang, 2015; Musoro et al., 2015; Yang et al., 2016; Hickey et al., 2016; Long and Mills, 2018; Mauff et al., 2020) and with a competing risks survival setting (Andrinopoulou et al., 2013, 2017).

We propose jointly modeling two longitudinal variables and the repeatedly measured multistate outcomes with a possible terminal event. We plan to use a joint survival function based on the copula explored in 4 with parametric Weibull marginal survival functions to define the multistate survival information. Copulas have been explored in the multistate context by Rotolo et al. (2013)and Diao and Cook (2014), for recurrent events by Huang et al. (2020) and Malehi et al. (2015), and for semi-competing risks by Wu et al. (2020). To our knowledge there has been no work including both multiple longitudinal outcomes and a multistate model derived from a copula. In this chapter we will develop a joint model based on the motivating study on adrenocortical carcinoma first described in Section 1.5.

5.1.1 Motivating Study on Adrenocortical Carcinoma

The example timeline from Chapter 1 is replicated in Figure 5.1. Recall that each patient has an initial scan and the state of disease at that scan is considered the reference state. It is determined at the next scan whether the tumor did not change size significantly (stable), became larger (progressed) or became smaller (regressed). The latest disease state is then considered the reference

state for the next scan and this continues for all the patient's scans. The diagram in Figure 5.1 shows two scans before the end of follow-up, but this could be any number of scans. We also have information on a terminal event, namely death. At the end of follow-up the patient may have died; otherwise the patient would be censored.
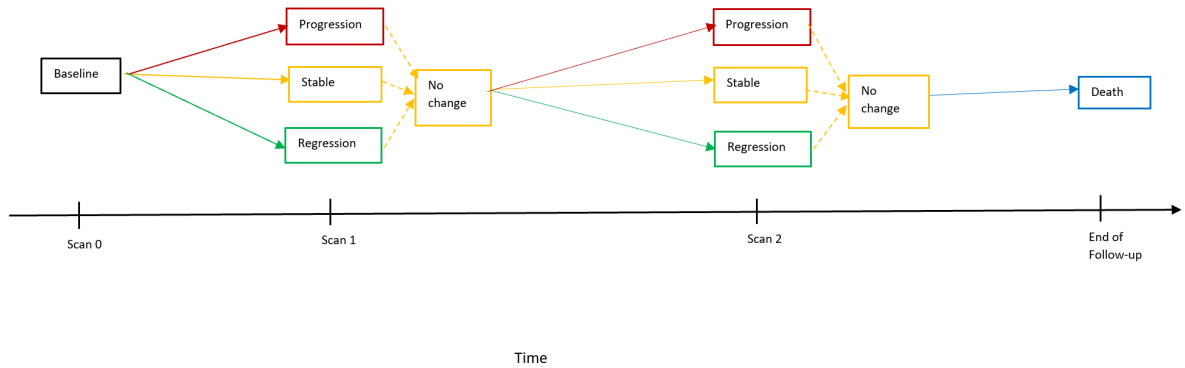

Figure 5.1: Diagram of possible states over time in the adrenal cancer study.

Recall that twelve time-dependent morphomic variables were recorded at each scan listed in Table 1.2 and described in Appendix A. As stated in Chapter 1, a previous study separately investigated the relationship between several morphomic variables and recurrence-free and overall survival in ACC (Miller et al., 2012). The study found that psoas muscle density (PMD), lean psoas muscle area (LPMA), and intra-abdominal (IA) fat had significant associations with the survival outcomes.

Since the morphomic variables are measured from the scans, they will be subject to measurement error. Therefore, a joint model for the longitudinal morphomic data and the survival data is needed. We began with the simplest case in Chapter 2 and investigated the relationship between a single morphomic variable, psoas density, and survival time using joint models within various computational settings. In Chapters 3 and 4 we considered how the longitudinal and survival processes are related when we incorporated the information on cancer progression. Specifically, we treated the survival process as a competing risks problem. We investigated the relationship between a single morphomic variable lean psoas muscle area (LPMA) and time to either first progression or

death without progression. Those who have died without progression could have died from causes other than their cancer. Analyzing the data as competing risks can help us learn about the relationship between LPMA and worsening disease (progression) while accounting for the competing risk of death not attributable to the disease. Finally here in Chapter 5 we will utilize more of the information available via a multistate model. The data include recurrent events, from the declaration of disease state at each scan, and these events can be one of multiple types (progression, regression, or no event). We also have the terminal event of death. Further, we have multiple morphomic variables and we would like to include more than one in our joint multistate model to examine the relationship between the morphomics and disease state, while accounting for all aspects of the data. We will do this with a joint model with two longitudinal outcomes and multistate survival data.

5.2   Multi-state Survival Data

Assume we have N subjects indexed by $i = 1, ..., N$ and R possible states. Subject i has K_i state transitions. Let D_t denote the state at time $t \geq 0$. The hazard function for going to state l from state k at time t is defined in (Hougaard, 1999) (with slightly different notation here) as

$$\lambda_{l|k}(t \mid D_u, u \in [0, t)) = \lim_{\Delta t \searrow 0} \frac{Pr\left(D_{t+\Delta t} = l \mid D_{t-} = k, \mathcal{F}_{t-}\right)}{\Delta t} \tag{5.1}$$

where $\mathcal{F}_{t-}$ is the history of the state process over time $[0, t)$. The total hazard from state k is

$$\lambda_k(t) = \sum_{l \neq k} \lambda_{l|k}(t) \tag{5.2}$$

In the ACC data we have four possible states: (1) stable, (2) progressing, (3) regressing, and (4) dead. State 4 (dead) is an absorbing state, so there can be no transitions out of this state. The hazard functions for going between each state are labeled on the state diagram in Figure 5.2.

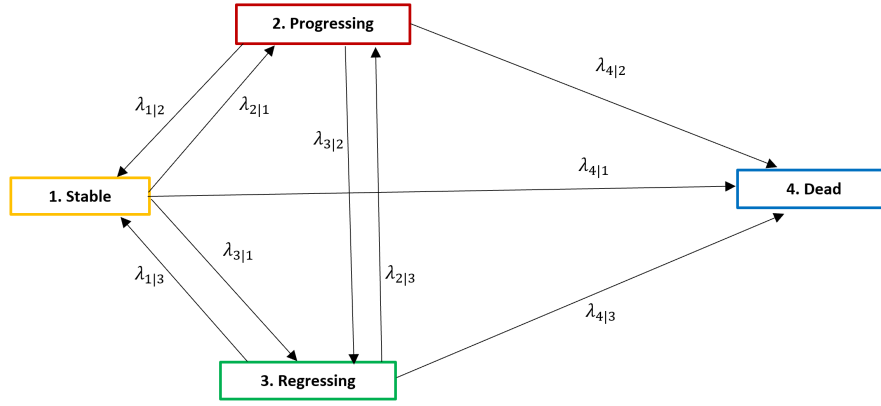We will use the Markov assumption which states that instead of the full state process history

Figure 5.2: Diagram of possible states in the ACC study.

we only need to know the last state before time t, i.e.

$$Pr(D_t = l | D_u, u \in [0, v]) = Pr(D_t = l \mid D_v) \tag{5.3}$$

Define the Markov transition probabilities as

$$P_{l|k}(t, v) = Pr(D_t = l \mid D_v = k) \tag{5.4}$$

As in previous chapters we will use the joint survival function with Weibull marginal hazards. Denote the time of the kth state transition for subject i by $t_{i,k}$ and the state at time $t_{i,k}$ by $D_{i,k}$. State transition times are measured as time from the start of the study (time since diagnosis in the ACC data) and so we assume the times are increasing, $t_{i,1} < t_{i,2} < ... < t_{i,K_i}$ for each i.

$$S(t_{i,1}, ..., t_{i,K_i} | D_{i,1}, ..., D_{i,K_i}) = \exp\left(-\left[(\mu_{i,D_{i,1}} t_{i,1}^{\gamma_{D_{i,1}}})^{\frac{1}{\delta}} + ... + (\mu_{i,D_{i,K_i}} t_{i,K_i}^{\gamma_{D_{i,K_i}}})^{\frac{1}{\delta}}\right]^{\delta}\right) \tag{5.5}$$

Using the Markov assumption, for transition at time $t_{i,k}$ we only need to know time $t_{i,k-1}$ from Markov assumption. So the relevant joint survival function for each transition is

$$S(t_{i,k-1}, t_{i,k} | D_{i,k-1}, D_{i,k}) = \exp\left(-\left[(\mu_{i,D_{i,k-1}} t_{i,k-1}^{\gamma_{D_{i,k-1}}})^{\frac{1}{\delta}} + (\mu_{i,D_{i,k}} t_{i,k}^{\gamma_{D_{i,k}}})^{\frac{1}{\delta}}\right]^{\delta}\right) \tag{5.6}$$

108

Hougaard (1999) gives the likelihood contribution for subject i as

$$\left[\prod_{k=1}^{K_i} \lambda_{D_{i,k}|D_{i,k-1}}(t_{i,k}|t_{i,k-1}) \exp\left(-\int_{t_{i,k-1}}^{t_{i,k}} \lambda_{D_{i,k-1}}(v|t_{i,k-1})dv\right)\right] \exp\left(-\int_{t_{i,K_i}}^{C_i} \lambda_{D_{i,K_i}}(v|t_{i,K_i})dv\right)$$

(5.7)

where C_i is the censoring time if the subject is censored after transition K_i (i.e. K_i is not an absorbing state). We need the conditional hazards $\lambda_{D_k|D_{k-1}}$ for each possible D_{k-1} and D_k. Shaked and Shanthikumar (1987) defines this conditional hazard using the joint survival and joint probability density functions. We write this out for $\lambda_{1|2}$ below and generalizing to any two states other than states 1 and 2 is straightforward. See Appendix B.6 for details.

$$\begin{aligned}
\lambda_{1|2}(t \mid t_2) &= \lim_{\Delta t \searrow 0} \frac{Pr\left(t < T_1 \le t + \Delta t \mid T_1 > t, T_2 = t_2\right)}{\Delta t} \\
&= -\frac{f(t, t_2)}{\left(\frac{\partial}{\partial t_2} S(t, t_2)\right)}
\end{aligned}$$

(5.8)

The joint probability density function is $f(t_1, t_2) = \frac{(-1)^2 \partial^2 S(t_1, t_2)}{\partial t_1 \partial t_2}$ (Crowder, 2012, p.105). Hence

$$\begin{aligned}
f(t, t_2) &= \frac{1}{\delta} \mu_1^{\frac{1}{\delta}} \gamma_1 t^{\frac{\gamma_1}{\delta}-1} \cdot \frac{1}{\delta} \mu_2^{\frac{1}{\delta}} \gamma_2 t_2^{\frac{\gamma_2}{\delta}-1} \cdot \exp\left(-\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta}\right) \cdot \\
&\quad \left[\delta(1-\delta)\left(\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-2}\right) + \delta^2 \left(\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{2\delta-2}\right)\right]
\end{aligned}$$

The denominator from (5.8) is

$$\frac{\partial}{\partial t_2} S(t, t_2) = -\exp\left(-\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta}\right) \left(\delta\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right) \frac{1}{\delta} \mu_2^{\frac{1}{\delta}} \gamma_2 t_2^{\frac{\gamma_2}{\delta}-1}$$

Giving

$$\lambda_{1|2}(t \mid t_2) = \frac{1}{\delta} \mu_1^{\frac{1}{\delta}} \gamma_1 t^{\frac{\gamma_1}{\delta}-1} \left[(1-\delta)\left(\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{-1}\right) + \left(\delta\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\right]$$

In general for transition to state l from state k, the conditional hazard is

$$\lambda_{l|k}(t|t_k) = \frac{1}{\delta}\mu_l^{\frac{1}{\delta}}\gamma_{l}t^{\frac{\gamma_l}{\delta}-1}\left[(1-\delta)\left(\left[(\mu_l t^{\gamma_l})^{\frac{1}{\delta}} + (\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}}\right]^{-1}\right) + \left(\delta\left[(\mu_l t^{\gamma_l})^{\frac{1}{\delta}} + (\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\right]$$

(5.9)

and the total hazard from state k is

$$\lambda_{k}(t|t_k) = \sum_{l\neq k}\left(\frac{1}{\delta}\mu_l^{\frac{1}{\delta}}\gamma_{l}t^{\frac{\gamma_l}{\delta}-1}\left[(1-\delta)\left(\left[(\mu_l t^{\gamma_l})^{\frac{1}{\delta}} + (\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}}\right]^{-1}\right) + \left(\delta\left[(\mu_l t^{\gamma_l})^{\frac{1}{\delta}} + (\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\right]\right)$$

(5.10)

Using these hazard functions we can write out the likelihood contribution for each subject as in (5.7).

5.3 Multiple Longitudinal Outcomes

Suppose for each subject i we have $M \geq 1$ different longitudinal processes measured over time which we will call a biomarker. Subject i has J_{im} total measurements of biomarker $m = 1, ..., M$ at times $\tau_i^{(m)} = (\tau_{i1}^{(m)}, ..., \tau_{ij}^{(m)}, ...\tau_{iJ_{im}}^{(m)})^T$. Let $Y_{i,j}^{(m)}$ be subject i's biomarker m measurement at time $\tau_{ij}^{(m)}$ and let $Y_i^{(m)}$ be the vector of biomarker measurements $Y_i^{(m)} = (Y_{i,1}^{(m)}, ..., Y_{i,J_m}^{(m)})^T$. We will assume a linear mixed effects model for $Y_i^{(m)}$,

$$Y_i^{(m)} = X_i^{(m)}\beta^{(m)} + Z_i^{(m)}U_i^{(m)} + \epsilon_i^{(m)}, m = 1, ..., M \qquad (5.11)$$

Here $X_i^{(m)}$ is a covariate matrix with coefficients $\beta^{(m)}$, $U_i^{(m)}$ is a vector of random effects with design matrix $Z_i^{(m)}$, and $\epsilon_i^{(m)}$ is a vector of random measurement errors $\epsilon_i^{(m)} = (\epsilon_{i1}^{(m)}, ..., \epsilon_{iJ_{im}}^{(m)})^T$. Assume $\epsilon_i^{(m)} \sim \text{N}(0, \sigma_\epsilon^{(m)2}I_{J_{im}})$.

For simplicity we will use only a random intercept, i.e. $Z_i^{(m)}U_i^{(m)} = U_{0i}^{(m)}$. The M biomarkers may be correlated with each other so we assume a multivariate normal distribution for the random intercepts with mean 0 and covariance matrix Σ_U, $(U_{0i}^{(1)}, ..., U_{0i}^{(M)})^T \sim \text{MVN}(0, \Sigma_U)$.

5.4 Bayesian Model and Estimation

5.4.1 Longitudinal Submodels

We will describe the Bayesian model that will be fit to our ACC data specifically. For the ACC data we will consider $M = 2$ biomarkers: LMPA ($Y^{(1)}$) and body depth (BD) ($Y^{(2)}$). Examples of the LPMA and BD measurements over time for selected patients are shown in Figures 5.3 and 5.4.

As in previous chapters we will include an intercept, time slope, binary sex X_{1i} and centered continuous age in decades X_{2i} in the longitudinal models. Here we include the same baseline covariates in each longitudinal model though these could differ between models. We have the following submodels.

$$Y_i^{(1)} \sim \mathrm{N}(\beta_0^{(1)} + \beta_1^{(1)} \tau_i^{(1)} + \beta_2^{(1)} X_{1i} + \beta_3^{(1)} X_{2i} + U_{0i}^{(1)}, {\sigma_\epsilon^{(2)}}^2 I_{J_{i1}})$$

$$Y_i^{(2)} \sim \mathrm{N}(\beta_0^{(2)} + \beta_1^{(2)} \tau_i^{(2)} + \beta_2^{(2)} X_{1i} + \beta_3^{(2)} X_{2i} + U_{0i}^{(2)}, {\sigma_\epsilon^{(2)}}^2 I_{J_{i2}})$$

$$(U_{0i}^{(1)}, U_{0i}^{(2)})^T \sim \mathrm{MVN}((0,0)^T, \Sigma_U)$$

Let $\Sigma_U = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$.

5.4.2 Multi-state Submodels

For our multistate model the possible transitions shown in Figure 5.2 are $2|1$, $3|1$, $4|1$, $1|2$, $3|2$, $4|2$, $1|3$, $2|3$, and $4|3$. No transitions are allowed out of state 4 (dead) since this is an absorbing state. The conditional and total hazards for these transitions are as described in Section 5.2. We incorporate covariates by putting a regression model on the μ_k parameters.

$$\mu_{i,k} = \exp\left(W_i\alpha_k + \theta_k^{(1)} U_{0i}^{(1)} + \theta_k^{(2)} U_{0i}^{(2)}\right) \tag{5.12}$$
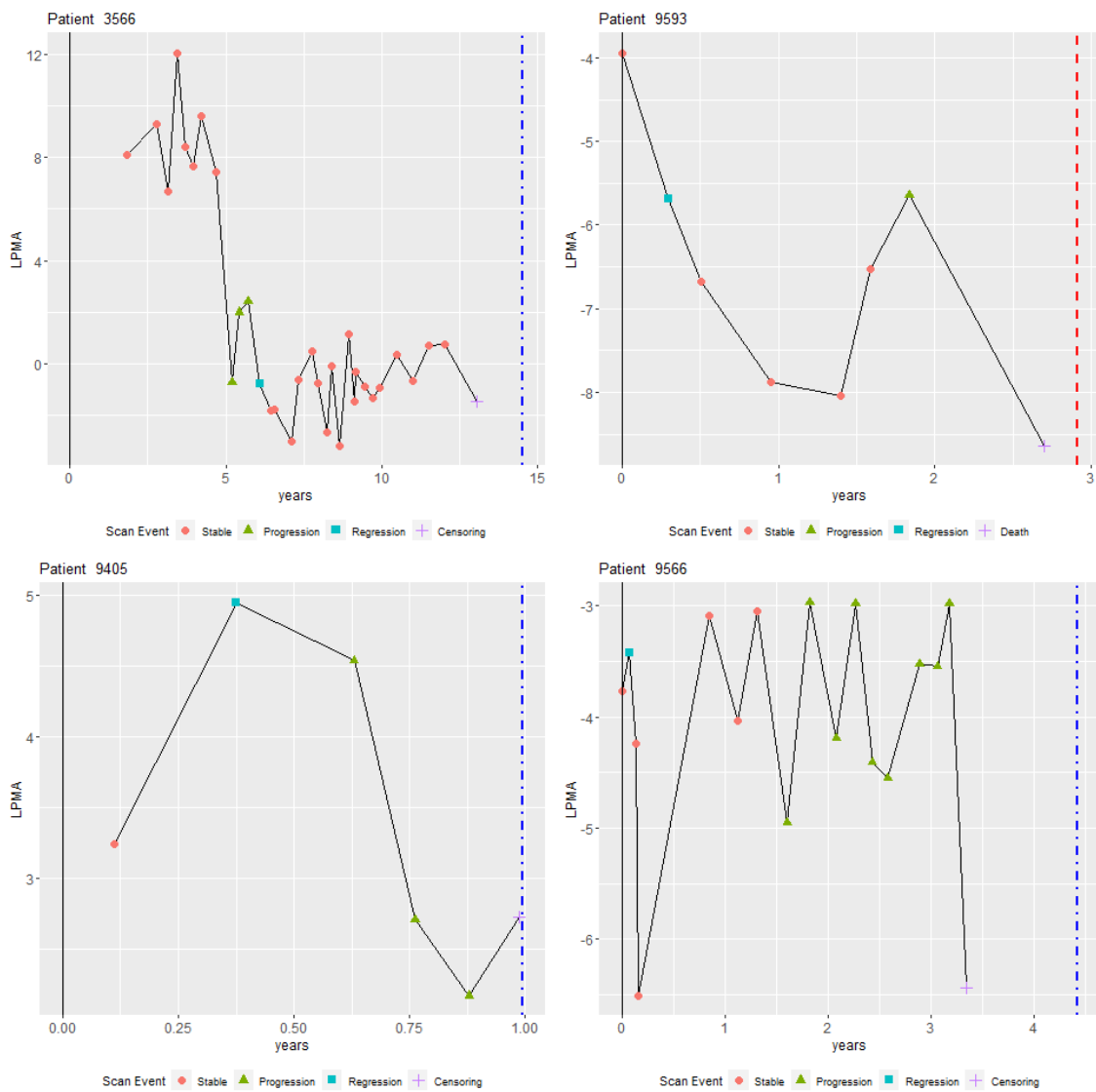
Figure 5.3: Example plots of LPMA from the ACC data. A vertical red dashed line indicates time of death. A vertical blue dot-dashed line indicates time of censoring. LPMA values in plots can be negative because the values have been centered.
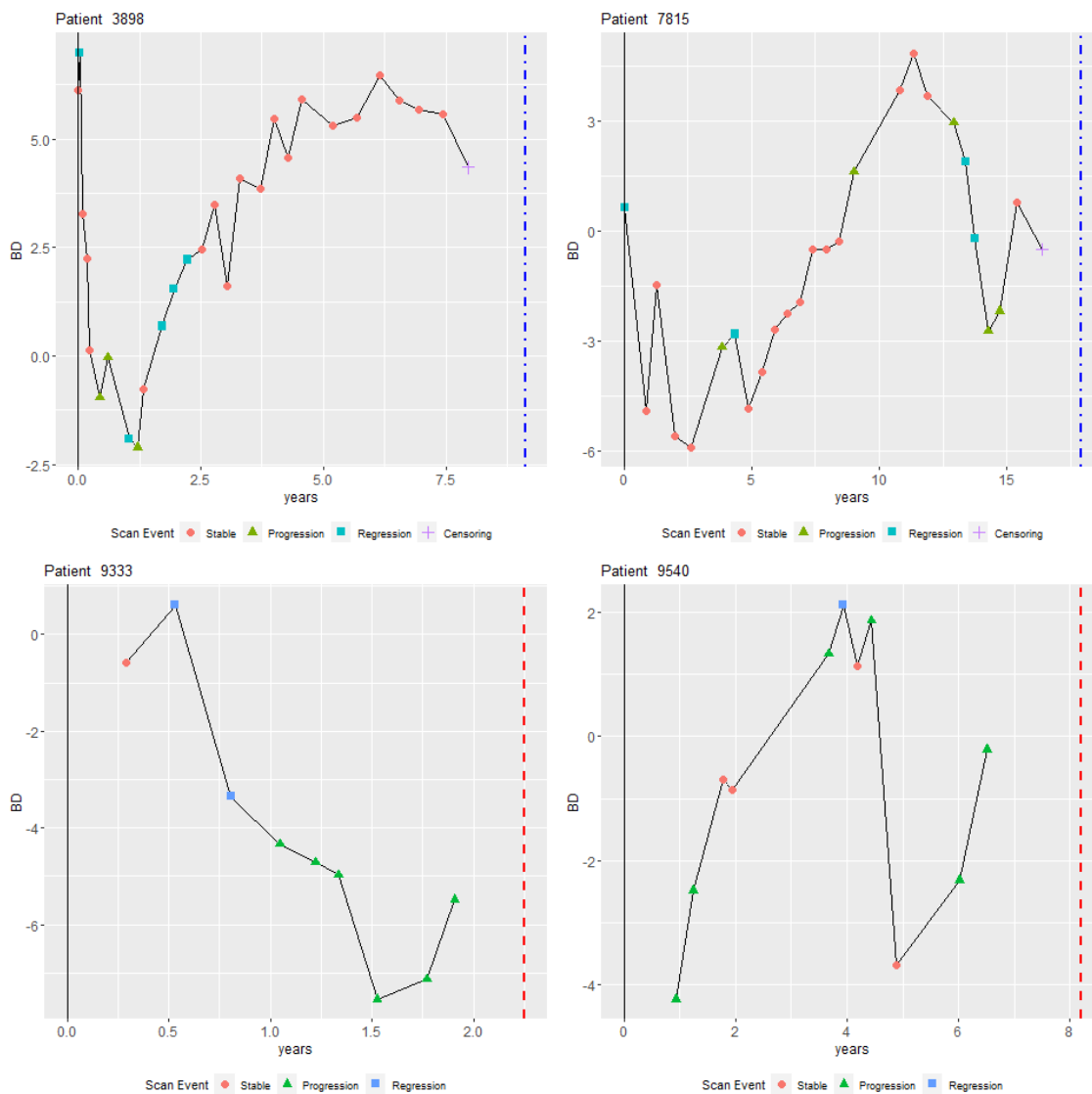
Figure 5.4: Example plots of BD from the ACC data. A vertical red dashed line indicates time of death. A vertical blue dot-dashed line indicates time of censoring. BD values in plots can be negative because the values have been centered.

Here W_i are time-independent covariates, which may coincide with the $X_i^{(m)}$ from the longitudinal models, α_k are regression coefficients, the $U_i^{(m)}$ are the same random effects as in the longitudinal submodels and the $\theta_k^{(m)}$ measure the strength of the association between the state k and the biomarker m. In our example we will include the centered age in decades as a covariate in each of these submodels, so

$$\mu_{i,k} = \exp\left(\alpha_{0k} + \alpha_{1k}X_{2i} + \theta_k^{(1)}U_{0i}^{(1)} + \theta_k^{(2)}U_{0i}^{(2)}\right).$$

5.4.3 Priors

For our Bayesian model we put priors on our parameters as follows with $m = 1, 2$.

$$\beta^{(m)} \sim \text{MVN}(m_\beta, s_\beta^2 I_4)$$

$$\Sigma_U \sim \text{Wishart}(a_U, B_U)$$

$$\sigma_\epsilon^{(m)} \sim \Gamma(a_e, b_e)$$

$$\alpha_k \sim \text{MVN}(m_\alpha, s_\alpha^2 I_2)$$

$$\theta_k^{(m)} \sim \text{N}(m_\theta, s_\theta^2)$$

$$\gamma_k \sim \Gamma(a_\gamma, b_\gamma)$$

$$\delta \sim \text{Beta}(a_\delta, b_\delta)$$

Hyperparameters used are $m_\beta = (2, 2, 2, 2)^T$, $s_\beta = 4$, $a_U = 4$, $B_U = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $a_\epsilon = 5$, $b_\epsilon = 5$, $m_\alpha = (2, 2)^T$, $s_\alpha = 4$, $m_\theta = 2$, $s_\theta = 4$, $a_\gamma = 5$, $b_\gamma = 5$, $a_\delta = 2$, $b_\delta = 2$.

5.4.4 Likelihood

Denote the vector of parameters by
$\Omega = (\beta^{(1)}, \beta^{(2)}, \Sigma_U, \sigma_\epsilon^{(1)}, \sigma_\epsilon^{(2)}, \alpha_1, ..., \alpha_4, \theta_1^{(1)}, ..., \theta_4^{(1)}, \theta_1^{(2)}, ..., \theta_4^{(2)}, \gamma_1, ..., \gamma_4, \delta)^T$. The posterior will

be proportional to the product of the likelihood and the prior densities. Let

$p(\Omega|Y_i^{(1)}, Y_i^{(2)}, T_{i,1}, ..., T_{i,K_i}, D_{i,1}, ..., D_{i,K_i}, X_i, W_i, U_{0i}^{(1)}, U_{0i}^{(2)})$ denote the posterior, L_Y the likelihood contribution from the longitudinal submodels, L_T the likelihood contribution from the multistate submodels, and $p(\Omega)$ be the product of the prior densities for all components of Ω. Then

$$p(\Omega|Y^{(1)}, Y^{(2)}, T, D, X, W, U^{(1)}, U^{(2)}) \propto L_Y L_T p(U|\Omega)p(\Omega)$$

where

$$L_Y = \prod_{i=1}^{N} \prod_{m=1}^{2} p(Y_i^{(m)}|X_i^{(m)}, \beta^{(m)}, U_{0i}^{(m)}, \Sigma_U, \sigma_\epsilon^{(m)})$$

$$= \prod_{i=1}^{N} \prod_{m=1}^{2} (2\pi\sigma_\epsilon^{(m)})^{-\frac{J_m}{2}} \exp\left\{-\frac{1}{2\sigma_\epsilon^{(m)J_m}} \left(Y_i^{(m)} - \beta^{(m)T}X_i^{(m)} - U_{0i}^{(m)}\right)^T \left(Y_i^{(m)} - \beta^{(m)T}X_i^{(m)} - U_{0i}^{(m)}\right)\right\}$$

and

$$L_T = \prod_{i=1}^{N} \prod_{k=1}^{K_i} \lambda_{D_{i,k}|D_{i,k-1}}(T_{i,k}|T_{i,k-1}) \exp\left(-\int_{T_{i,k-1}}^{T_{i,k}} \lambda_{D_{i,k-1}}(v|T_{i,k-1})dv\right) \cdot$$

$$\exp\left(-\int_{T_{i,K_i}}^{C_i} \lambda_{D_{i,K_i}}(v|T_{i,k-1})dv\right) \tag{5.13}$$

We see that the likelihood contribution in L_T contains integrals which can not be evaluated analytically. Due to difficulties in using the Stan numerical integration function (`integrate_1d`) we estimated these integrals using the midpoint method. Specifically,

$$\int_{T_{k-1}}^{T_k} \lambda_{D_{k-1}}(v|T_{k-1})dv \approx (T_k - T_{k-1}) * \lambda_{D_{k-1}}\left(\frac{T_{k-1} + T_k}{2}|T_{k-1}\right)$$

This is a crude approximation but since generally the time between state transitions, (T_{k-1}, T_k), is relatively small using this approximation is reasonable. Time between transitions varied between 0.003 years and 7.97 years with a median of 0.31 years and third quantile of 0.78 years.

5.5 Results

We fit our models using Hamiltonian Monte Carlo with Stan through the `R` package rstan version 2.21.2 and `R` version 4.0.4 (Guo et al., 2020). Stan code is provided in Appendix E.4. Initial values were generated automatically. Due to slow computation time the models were run for 1000 iterations total with 500 iterations of warm-up for four chains. Despite the small number of iterations, Rhat values for the parameters were between 1.00 and 1.01 indicating that the chains have mixed. See Appendix Table F.2. Trace plots in Figures 5.5 - 5.10 also show mixing of the two chains.

Results are shown in Table 5.1. As in previous chapters we see that LPMA tends to decrease over time ($\beta_1^{(1)} < 0$), females have significantly lower LPMA ($\beta_2^{(1)} < 0$) and LPMA decreases with increasing age ($\beta_3^{(1)} < 0$). We do not see strong effects of time, sex, or age on body depth ($\beta_1^{(2)}, \beta_2^{(2)}, \beta_1^{(2)} \approx 0$). In the multistate model, we see that age does not have a significant associ-ation with stable ($\alpha_{1,1}$), regression ($\alpha_{1,3}$), or death states ($\alpha_{1,4}$). Older patients do tend to have a shorter time to progression ($\alpha_{1,2} < 0$). We find no significant association between LPMA and tran-sition times for stable ($\theta_1^{(1)}$) and progression ($\theta_2^{(1)}$) states. Higher LPMA is associated with longer transition time to regression ($\theta_3^{(1)}$) and shorter time to death ($\theta_4^{(1)}$). Unsurprisingly given the lack of association found in the longitudinal model for body depth, we do not find significant associa-tion between body depth and the transition times ($\theta_1^{(2)}, \theta_2^{(2)}, \theta_3^{(2)}, \theta_4^{(2)}$). The dependence parameter δ is found to be 1 indicating independence between the transition times. This is an interesting result that warrants further study. The dependence between specific states (say progression and death) may be different than dependence between other states and this model cannot capture such heterogeneity in the single dependence parameter δ.

Parameter	Mean	CI	Pr > 0	Parameter	Mean	CI	Pr > 0
$\beta_0^{(1)}$	9.37	(7.19, 11.51)	1.00	$\alpha_{0,1}$	1.03	(0.36, 1.71)	1.00
$\beta_1^{(1)}$	-0.22	(-0.31, -0.12)	0.00	$\alpha_{1,1}$	-0.04	(-0.17, 0.08)	0.26
$\beta_2^{(1)}$	-6.73	(-7.8, -5.57)	0.00	$\alpha_{0,2}$	0.68	(-0.01, 1.39)	0.97
$\beta_3^{(1)}$	-1.34	(-1.77, -0.89)	0.00	$\alpha_{1,2}$	-0.21	(-0.35, -0.06)	0.00
$\beta_0^{(2)}$	-0.01	(-0.04, 0.02)	0.19	$\alpha_{0,3}$	-1.06	(-2.22, 0.07)	0.03
$\beta_1^{(2)}$	0.00	(0.00, 0.00)	0.40	$\alpha_{1,3}$	0.12	(-0.07, 0.33)	0.88
$\beta_2^{(2)}$	-0.02	(-0.03, 0.00)	0.01	$\alpha_{0,4}$	-0.81	(-1.70, 0.03)	0.03
$\beta_3^{(2)}$	0.00	(0.00, 0.01)	0.96	$\alpha_{1,4}$	-0.10	(-0.27, 0.07)	0.14
σ_1	3.77	(3.32, 4.26)	1.00	$\theta_1^{(1)}$	0.00	(-0.06, 0.05)	0.46
σ_2	0.04	(0.04, 0.05)	1.00	$\theta_2^{(1)}$	0.00	(-0.06, 0.05)	0.55
ρ	0.15	(-0.02, 0.31)	0.96	$\theta_3^{(1)}$	0.09	(0.02, 0.16)	0.99
$\sigma_\epsilon^{(1)}$	3.02	(2.87, 3.17)	1.00	$\theta_4^{(1)}$	-0.17	(-0.25, -0.10)	0.00
$\sigma_\epsilon^{(2)}$	0.02	(0.02, 0.02)	1.00	$\theta_1^{(2)}$	2.57	(-1.54, 6.61)	0.89
δ	1.00	(0.99, 1.00)	1.00	$\theta_2^{(2)}$	3.83	(-0.32, 7.91)	0.97
γ_1	0.70	(0.55, 0.86)	1.00	$\theta_3^{(2)}$	-0.07	(-5.34, 4.98)	0.49
γ_2	0.50	(0.37, 0.63)	1.00	$\theta_4^{(2)}$	4.26	(-0.47, 8.96)	0.96
γ_3	0.30	(0.16, 0.45)	1.00				
γ_4	0.74	(0.56, 0.93)	1.00				

Table 5.1: Posterior Mean (Mean), 95% Credible Interval (CI) and probability the parameter is greater than 0 (Pr > 0) for the joint model fit to the ACC data.
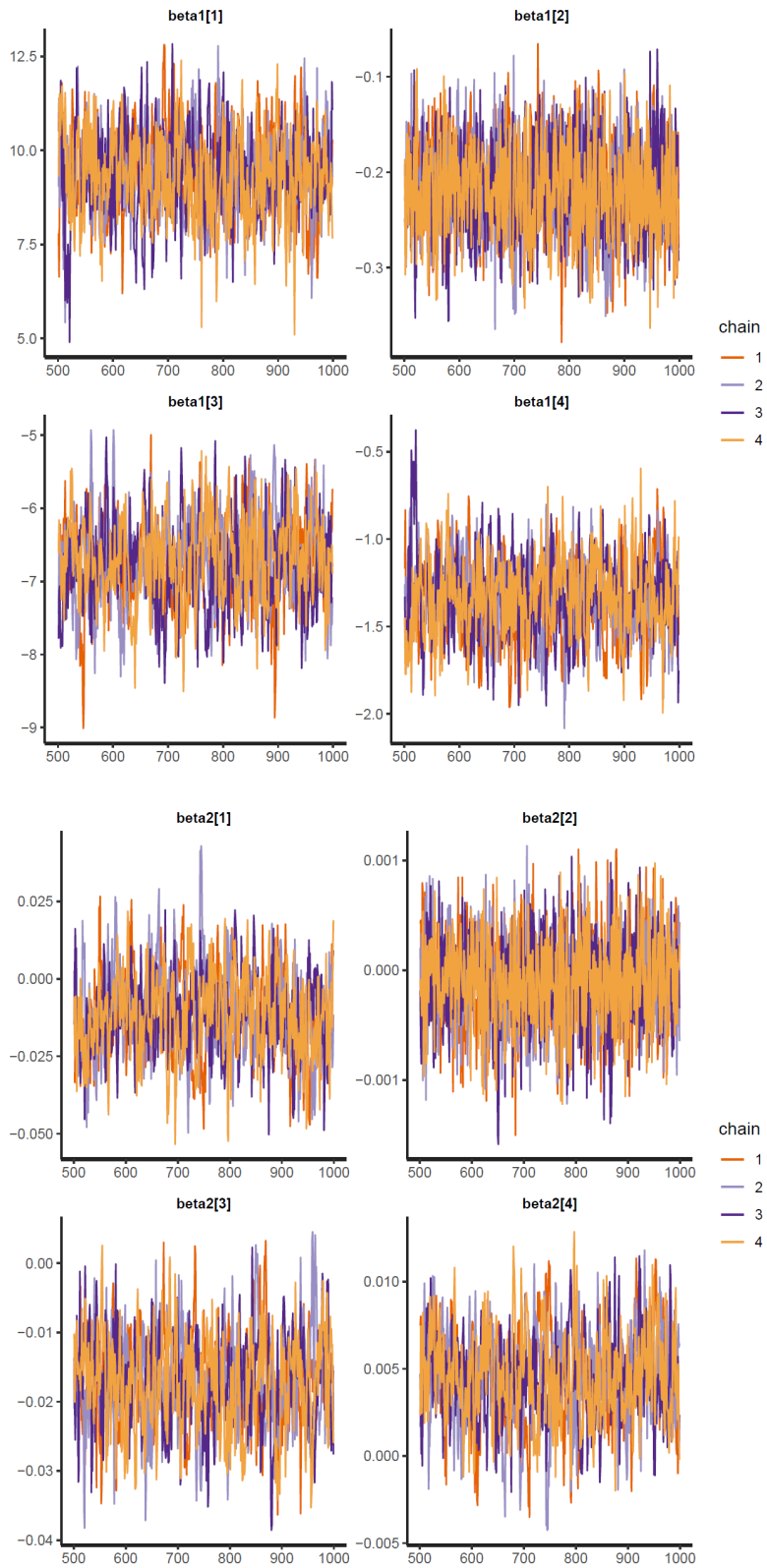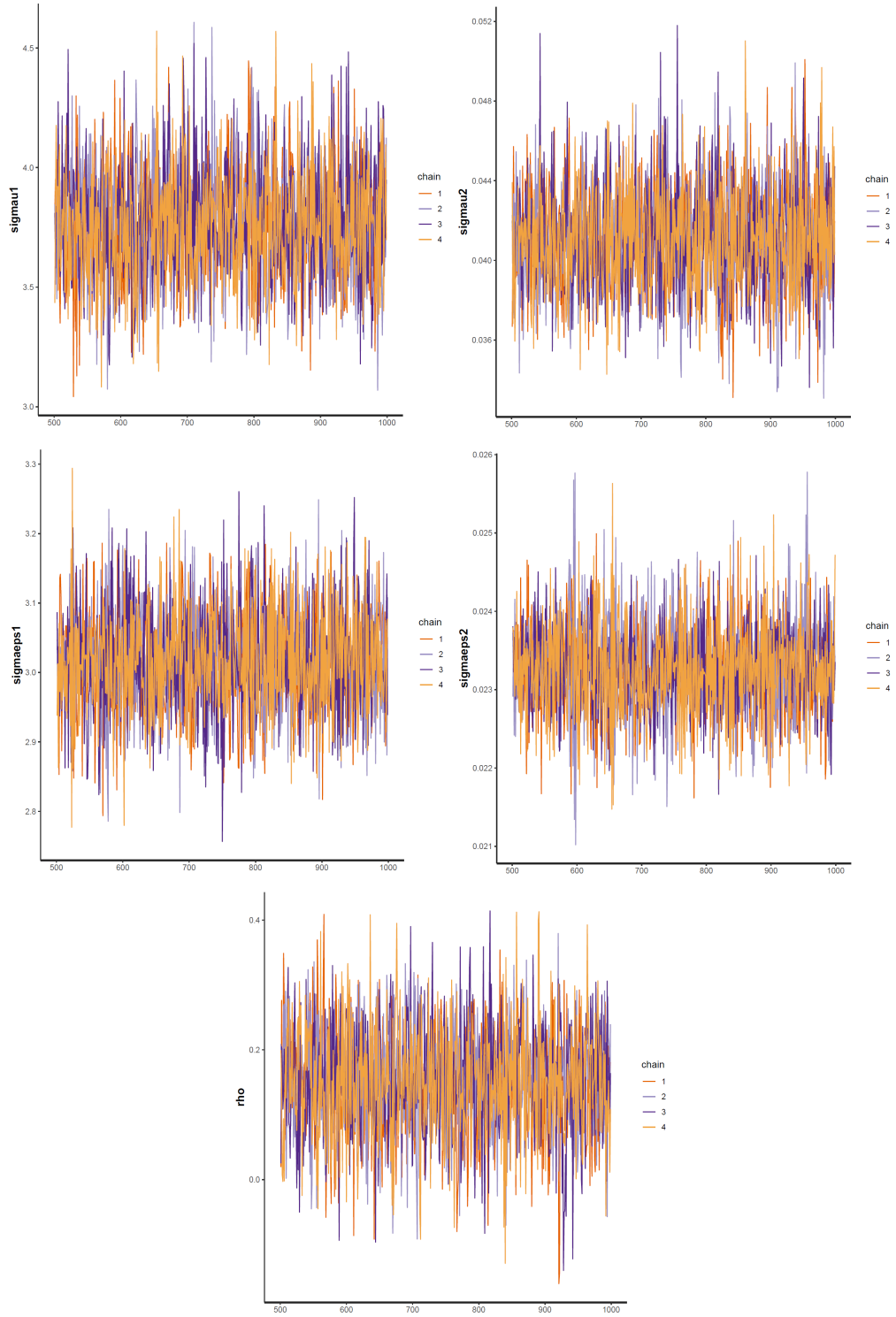
Figure 5.5: Trace plots for beta parameters.

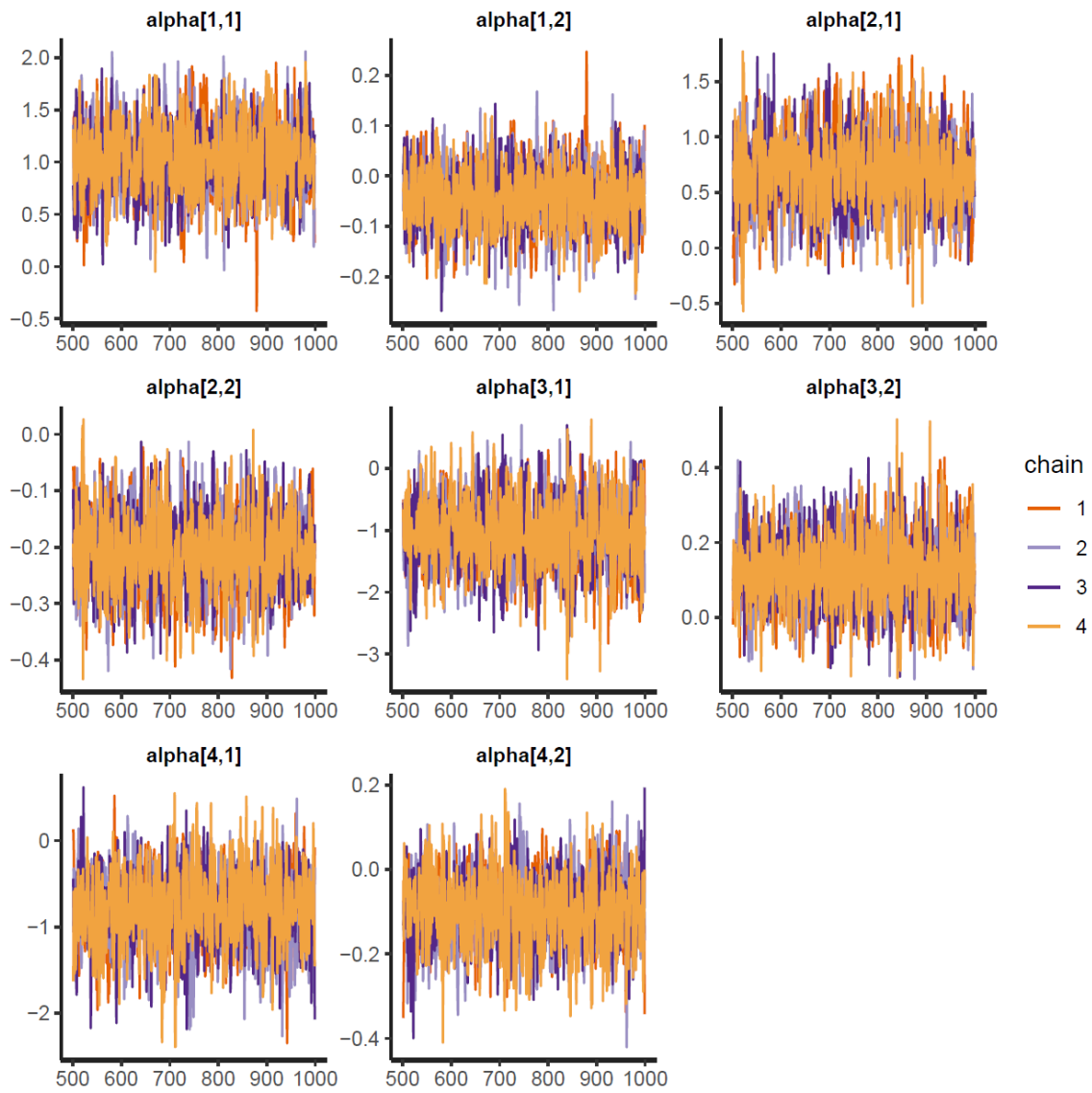Figure 5.6: Trace plots for standard deviation and correlation parameters.

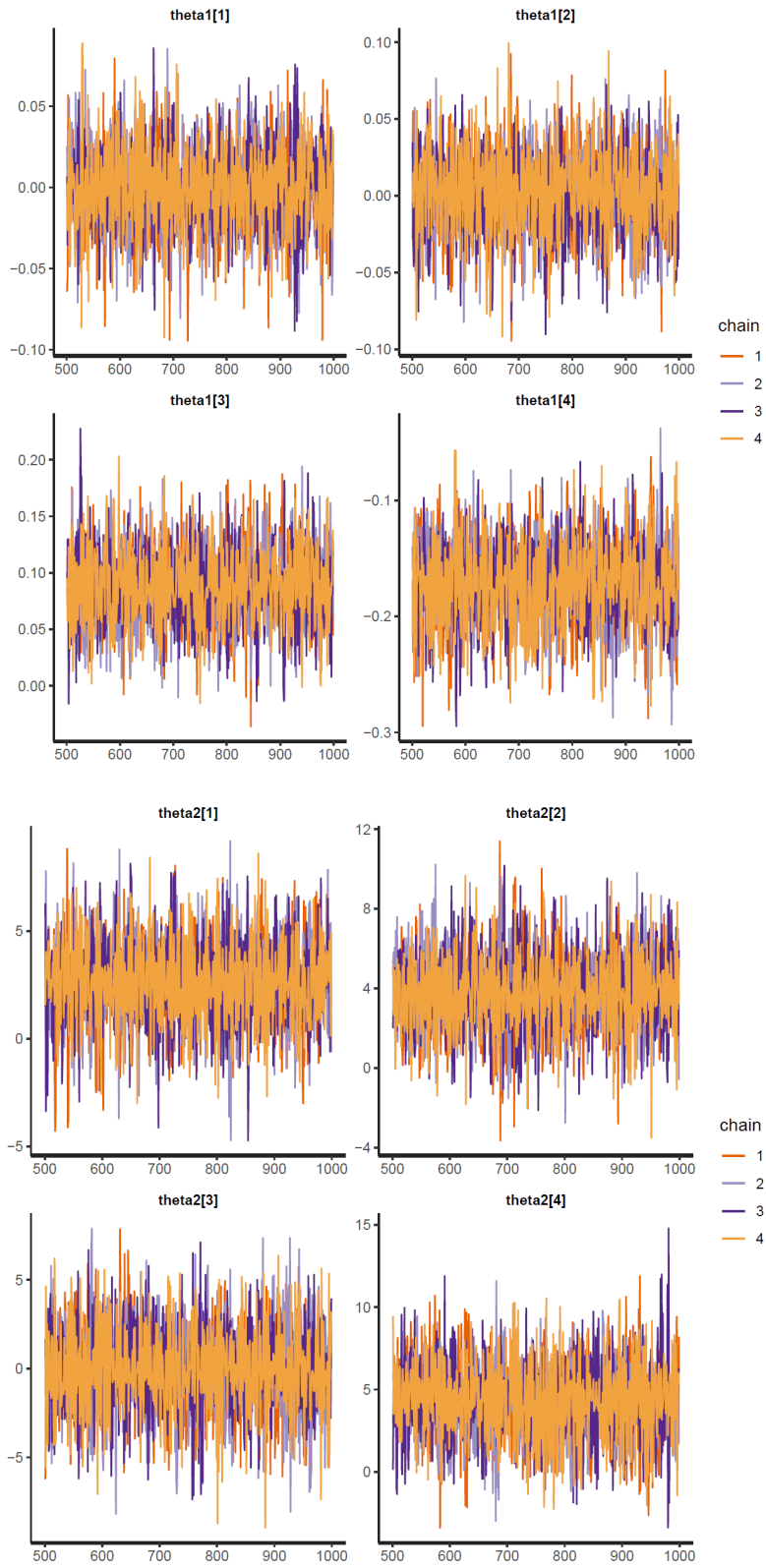Figure 5.7: Trace plots for alpha parameters.
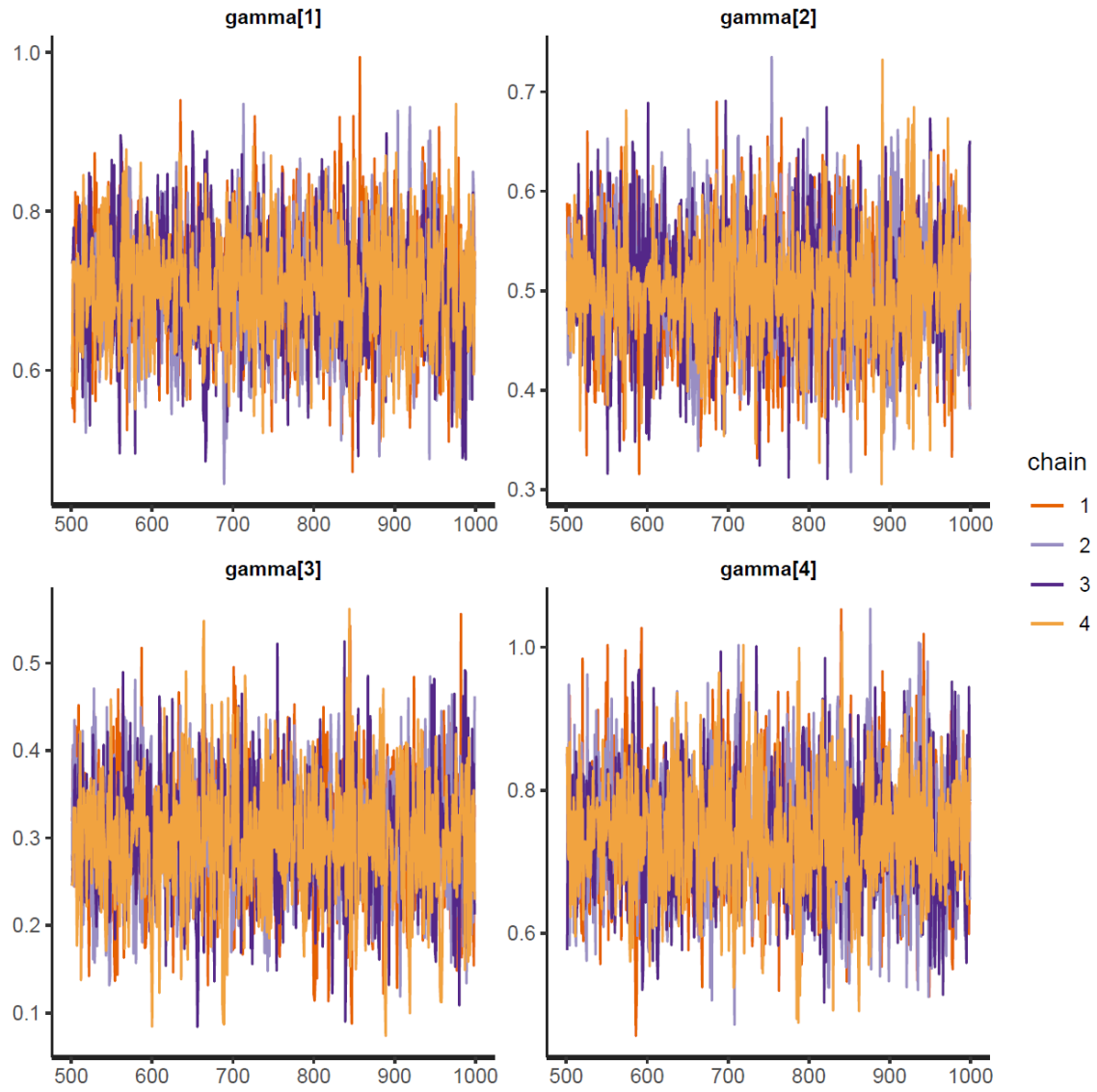
Figure 5.8: Trace plots for theta parameters.

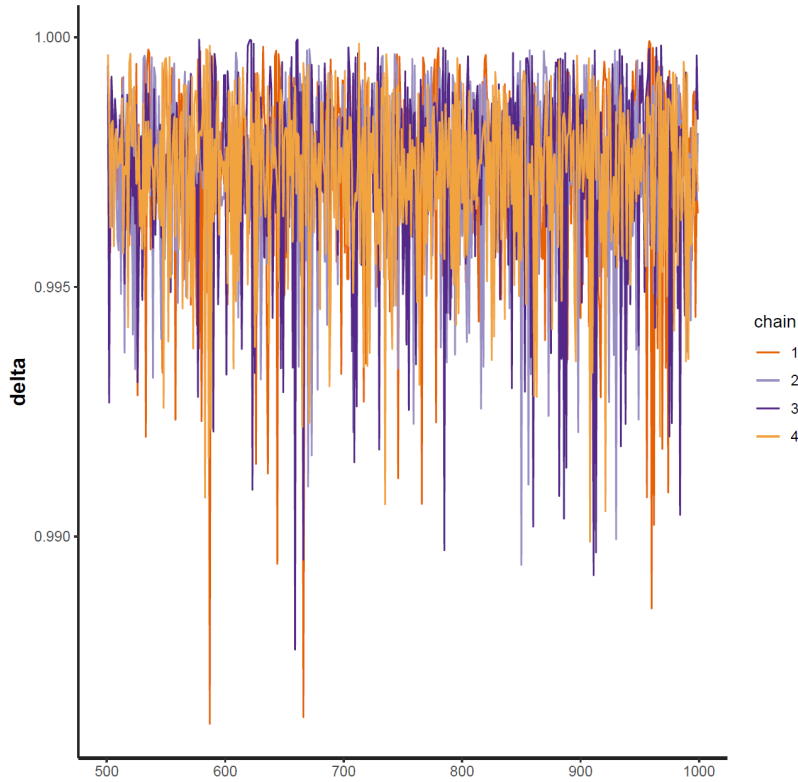Figure 5.9: Trace plots for gamma parameters.

Figure 5.10: Trace plot for delta parameter.

5.6 Discussion

In this chapter we developed a joint model based on the motivating ACC data. This included more than one longitudinal outcome and a multistate model for the survival part. We worked under the Markov assumption, a common simplifying assumption in multistate modeling. One extension would be to weaken this assumption such as assuming a semi-Markov situation. While the possible transitions described in this work were based on the ACC data, the hazards for transition from state k to l are general and could be applied to a problem with any possible transitions.

As in other chapters we use a joint survival function from an Archimedean copula with Weibull marginals. Archimedean copulas have some nice properties as discussed in Chapter 4 and Weibull is a common distribution in event time analysis. This formulation is an alternative to the often used proportional hazards models for transition hazards. Utilizing a copula also allows us a way to study the dependence between the times between state transitions. One limitation to the model studied

in this chapter is that the dependence is quantified in a single parameter. One may be interested in dependence between specific transitions more than others or differences in dependence between different transitions. Future work could consider how to extend this model to measure dependence with multiple parameters. Additional flexibility could be added to the multistate model by allowing the marginal distributions for the transition times to be from different distributions (instead of all Weibull). This would require careful considerations of what distributions are best suited to each transition time and coding and model interpretation would be more complicated.

Our longitudinal outcomes are both continuous but using a generalized linear mixed model for binary, count or other non-continuous outcomes is straightforward. We included only a single random intercept in each longitudinal submodel. Additional or different random effects could be included such as a random slope. This would obviously require estimating additional variance and correlation parameters for these added random effects. We also included the random intercepts from both longitudinal submodels in the regression models for the multistate data. Another option could be to include the trajectory of the longitudinal variables in the multistate regression as in $\mu_k = \alpha_k^T W_i + \theta_k^{(1)} \left(X_i^{(1)} \beta^{(1)} + U_{0i}^{(1)} \right) + \theta_k^{(2)} \left(X_i^{(2)} \beta^{(2)} + U_{0i}^{(2)} \right)$ for a random intercept model and $\mu_k = \alpha_k^T W_i + \theta_k^{(1)} \left(X_i^{(1)} \beta^{(1)} + \eta_{0,1} U_{0i}^{(1)} + \eta_{1,1} U_{1i}^{(1)} \right) + \theta_k^{(2)} \left(X_i^{(2)} \beta^{(2)} + \eta_{0,2} U_{0i}^{(2)} + \eta_{0,2} U_{1i}^{(2)} \right)$ for a random intercept and slope model. We chose to put the regression model on the shape parameter of the Weibull distribution but instead a regression model could be put on the scale parameter γ_k or the dependence parameter δ.

Our code to fit the model relied on a crude approximation to integrals in the likelihood. Future work ideally would use an approximation method expected to have less bias. We also were only able to run the model for a relatively small number of iterations and in the future could run the model for longer to hopefully verify the results are similar. We showed that this model can be applied to the ACC data. A simulation study would be useful to verify that the inferences from this

model are accurate and precise.

Chapter 6

Conclusion

In this dissertation we have explored joint models for longitudinal and survival data of multiple types. In particular we considered a standard joint model with a single survival event in Chapter 2, with competing risks in Chapters 3 and 4. In Chapter 5 we generalized the model to explore survival data with multiple states over time as well as multiple longitudinal biomarkers. This work was motivated by a study of adrenocortical carcinoma in the University of Michigan's Rogel Cancer Center, but these models can be applied to other problems including other cancers such as progression and regression in prostate cancer and association longitudinal PSA levels (Ferrer et al., 2016).

Medical studies often collect both longitudinal and time-to-event data which necessitate a joint modeling approach to fully account for all dependencies in the data. But these joint models can be difficult to implement for clinicians and statisticians who do not work with these models frequently. This has motivated the creation of various software packages for implementing the most common joint models seen in practice. Comparing these software implementations was the focus of Chapter 2. In later chapters of this dissertation we explored extensions in the survival data including competing risks and multistate data. Software to implement these types of models would help facilitate increased use in practical settings. There has been limited development of joint longitudinal and competing risks software but developing user-friendly software for implementing these models is a constant source of future work.

In Chapter 3 we discussed how the model we developed could be used for dynamic prediction. Dynamic prediction with joint models quickly became an area of interest once joint modeling became more accessible and common. These predictions can be directly utilized in a clinical

setting if implemented in a way clinicians can access such as a website (Taylor et al., 2013). While we detailed the algorithm for calculating dynamic predictions using our models in Chapter 3, there is still work that could be done. The model and predictions have not been validated. Sensitivity analyses investigating the influence of our priors and other model assumptions could be performed. Additional parametric distributions could also be used. Finally the dynamic predictions could be extended to use the models with dependence in Chapter 4 and the models with multistate data in Chapter 5.

This work has focused on developing joint models that fit the ACC study and are also novel in the way the survival submodels are defined. We have described possible settings where these models may be more appropriate than standard joint models which often use proportional hazards formulations. These settings include when a proportional hazards assumption does not fit the data. In this work we concentrated on describing the models and demonstrating through simulation that the model estimates are close to the true values. A useful area of future work would be to use simulation studies to discover how these models compare to the more common, usually proportional hazards, models. It would be useful to advocating for the use of these models to know when our new models have better fit to the data.

We chose to focus on parametric and copula-based survival submodels but there are other options that could be explored. First, in Chapters 4 and 5 we used Weibull cumulative hazards and marginals, respectively, but we could have considered using the log-Normal distribution as in Chapter 3 or another distribution. Both Weibull and log-Normal are from the log-location-scale family and so other members of that family are possible candidates for study. We could also study joint models in a competing risks setting using a Fine-and-Gray model defining the survival submodel through the subdistribution functions (Fine and Gray, 1999). This option has received little consideration in the literature (Deslandes and Chevret, 2010; Musoro et al., 2018)

A related area that could use future study is goodness of fit and model diagnostics in joint modeling. We used DIC to compare the Weibull and log-Normal model fits to the ACC data in Chapter 3 but this is a generic measure output by the Bayesian software Stan. Diagnostics have

127

received some study in joint modeling with a single longitudinal outcome and single survival event (Dobson and Henderson, 2003; Rizopoulos et al., 2010; Park and Qiu, 2014) (Rizopoulos, 2012, Chapter 6)(Elashoff et al., 2017, p.198). Variable selection is another important consideration that has received little attention in the joint modeling context (Chen and Wang, 2017)(Elashoff et al., 2017, p.202). Extension of model diagnostics and variable selection methods to joint models with competing risks and with multistate data has, to the best of our knowledge, yet to be explored.

We hope that the work in this dissertation encourages future exploration into different aspects of joint modeling. We have considered unique parametric and copula-based approaches that we believe deserve further study.

Appendix A

Description of Morphomic Variables in

Adrenocortical Carcinoma Data

Below is a short description of each of the morphomics variables in the adrenal cancer study. This information is from the online Morphomics Data Dictionary (Holcombe et al., 2016). These measurements are taken from CT scans of the chest, abdomen, or pelvis. Most measurements are taken at each vertebral level and are reported as a function of body region. Some measures of muscle density are reported in Hounsfield Units (HU) (Hounsfield, 1980).

Average psoas perimeter mean

Mean pixel intensity of all pixels in psoas region of scan, a measure of psoas muscle, in HU.

Body depth

Distance between front and back of body (aligned to body habitus), a body measure, in mm.

Central back fat depth

Distance from posterior tip of spinous process to back skin, a body measure, in mm.

Central sub-cutaneous depth

Linear distance from anterior fascia to anterior skin, a body measure, in mm.

Central visceral depth

Central visceral depth, distance from anterior of vertebral body to anterior fascia, a body measure, in mm.

Fascia area

Cross-sectional area of the fascia region, a body measure, in mm^2.

Fascia depth

Central sub-cutaneous depth, linear distance from anterior fascia to anterior skin, in mm.

Lean psoas

From Holcombe et al. (2016): " "lean psoas" is a mathematical combination of measures of psoas cross sectional area (mm^2) and psoas density (in HU) inside the muscle boundary... This can be thought of as "normalizing a muscle's density between -85 HU (very fatty and low density) and +85 HU (very dense)". The values of ± 85 were chosen by inspection of some of our sickest and healthiest individuals' data points. " The formula for lean psoas is given as:

$$\text{(lean psoas)} = \frac{\text{(mean psoas density)} + 85}{170} \cdot \text{(psoas cross sectional area)}$$

LMPA lean psoas muscle area

Area covered by psoas muscles in cross-sectional image, usually sampled at the L4 vertebral level.

LPMD lean psoas muscle density

Measure of fatty infiltration in psoas muscles found in cross-sectional image.

Psoas muscles

Core muscles running alongside the lumbar spine.

Total psoas perimeter area

Total area within the perimeter of the left and right psoas, a measure of psoas muscle, in mm^2.

VB2 front skin

Central visceral depth, distance - anterior of vertebral body to anterior fascia, in mm.

Visceral fat area

Area inside fascia meeting fat density thresholds (-205 to -51 HU), a fat measure, in mm^2.

Variable	Description	N Missing	% Missing
average psoas perimeter mean	Mean pixel intensity of all pixels in psoas region of scan, a measure of psoas muscle, in HU	244	21
body depth	Distance between front and back of body (aligned to body habitus), a body measure, in mm	240	21
central back fat depth	Distance from posterior tip of spinous process to back skin, a body measure, in mm	718	63
central sub-cutaneous depth	Linear distance from anterior fascia to anterior skin, a body measure, in mm	238	21
central visceral depth	Distance from anterior of vertebral body to anterior fascia, a body measure, in mm	243	21
fascia area	Cross-sectional area of the fascia region, a body measure, in mm^2	243	21
fascia depth	Central sub-cutaneous depth, linear distance from anterior fascia to anterior skin, in mm	243	21
lean psoas muscle area	Area covered by psoas muscles in cross-sectional image, usually sampled at the L4 vertebral level, in mm^2	243	21
lean psoas muscle density	Measure of fatty infiltration in psoas muscles found in cross-sectional image, in HU	243	21
total psoas perimeter,	Total area within the perimeter of the left and right psoas, a measure of psoas muscle, in mm^2	244	21
vb2 front skin	Central visceral depth, distance - anterior of vertebral body to anterior fascia, in mm	240	21
visceral fat area	Area inside fascia meeting fat density thresholds (-205 to -51 HU), a fat measure, in mm^2	240	21

Table A.1: Morphomic measures in adrenal cancer study.

Appendix B

Derivations

B.1 Generation of Survival Times in Chapter 2 Simulations

The generalized hazard function for the survival submodel of Scenario 2 is

$$h_i(t) = h_0(t) \exp(\gamma_0 + \gamma_1 X_{1i} + \gamma_2 X_{2i} + \alpha m_i(t)) \tag{B.1}$$

where, in general terms,

$$Y_i(t) = m_i(t) + e_i(t) = \beta_0 + \beta_1 t + \beta_2 X_{1i} + \beta_3 X_{2i} + U_{0i} + U_{1i}t + e_i(t)$$

We set $h_0(t) = 1$ and have a constant γ_0 absorbed into the exponential. After dropping the subscript i,

$$h(t) = \exp(\gamma_0 + \gamma_1 X_1 + \gamma_2 X_2 + \alpha(\beta_0 + \beta_1 t + \beta_2 X_1 + \beta_3 X_2 + U_0 + U_1 t))$$

Then the cumulative hazard function is

$$
\begin{aligned}
H(t) &= \int_0^t h(y)dy \\
&= \int_o^t \exp(\gamma_0 + \gamma_1 X_1 + \gamma_2 X_2 + \alpha(\beta_0 + \beta_1 y + \beta_2 X_1 + \beta_3 X_2 + U_0 + U_1 y))dy \\
&= \frac{\exp(\gamma_0 + \alpha\beta_0 + \alpha U_0 + (\gamma_1 + \alpha\beta_2)X_1 + (\gamma_2 + \alpha\beta_3)X_2)}{\alpha\beta_1 + \alpha U_1}(\exp(\alpha\beta_1 t + \alpha U_1 t) - 1)
\end{aligned}
$$

Let $V \sim \text{Unif}(0, 1)$ be a random survival probability and set $H(t) = -\log(V)$. Then

$$-\log(V) = \frac{\exp(\gamma_0 + \alpha\beta_0 + \alpha U_0 + (\gamma_1 + \alpha\beta_2)X_1 + (\gamma_2 + \alpha\beta_3)X_2)}{\alpha\beta_1 + \alpha U_1} (\exp(\alpha\beta_1 t + \alpha U_1 t) - 1)$$

$$\exp(\alpha\beta_1 t + \alpha U_1 t) = \frac{-\log(V)(\alpha\beta_1 + \alpha U_1)}{\exp(\gamma_0 + \alpha\beta_0 + \alpha U_0 + (\gamma_1 + \alpha\beta_2)X_1 + (\gamma_2 + \alpha\beta_3)X_2)} + 1$$

$$\alpha(\beta_1 + U_1)t = \log\left(\frac{-\log(V)(\alpha\beta_1 + \alpha U_1)}{\omega_0 \exp(\gamma_0 + \alpha\beta_0 + \alpha U_0 + (\gamma_1 + \alpha\beta_2)X_1 + (\gamma_2 + \alpha\beta_3)X_2)} + 1\right)$$

and so

$$t = \frac{1}{\alpha(\beta_1 + U_1)} \log\left(\frac{-\log(V)(\alpha(\beta_1 + U_1))}{\exp(\gamma_0 + \alpha\beta_0 + \alpha U_0 + (\gamma_1 + \alpha\beta_2)X_1 + (\gamma_2 + \alpha\beta_3)X_2)} + 1\right)$$

The calculation for a random intercept only model (Scenario 1) is the same but with $U_1 = 0$. Calculation for Scenario 3 (a shared parameter model) is simpler with $m_i(t)$ replaced by just U_i in (B.1).

B.2 Chapter 2 Simulation Model Reparameterization

In Scenario 1, we fit a shared parameter model as in (2.7) with `joineR` while the data was generated from the current-value model described in (2.5) and (2.6). So the data are generated from

$$Y = \beta_0 + \beta_1 t + \beta_2 X_1 + \beta_3 X_2 + U + e$$

and

$$h(t) = \exp(\gamma_0) \exp(\gamma_1 X_1 + \gamma_2 X_2 + \alpha(\beta_0 + \beta_1 t + \beta_2 X_1 + \beta_3 X_2 + U)) \tag{B.2}$$

The longitudinal submodel is the same for `joineR` but the survival submodel is of the form

$$h(t) = h_0(t) \exp(\gamma_1^* X1_1 + \gamma_2^* X_2 + \alpha^* U) \tag{B.3}$$

Rearranging terms in (B.2),

$$h(t) = \exp(\gamma_0) \exp(\gamma_1 X_1 + \gamma_2 X_2 + \alpha(\beta_0 + \beta_1 t + \beta_2 X_1 + \beta_3 X_2 + U))$$

$$= \exp(\gamma_0 + \alpha\beta_0 + \alpha\beta_1 t) \exp((\gamma_1 + \alpha\beta_2)X_1 + (\gamma_2 + \alpha\beta_3)X_2 + \alpha U)$$

Comparing this to (B.3) we see that

$$\gamma_1^* = \gamma_1 + \alpha\beta_2$$
$$\gamma_2^* = \gamma_2 + \alpha\beta_3$$

(B.4)

Since `joineR` assumes an unspecified baseline hazard, the form of the baseline hazard does not matter when estimating. When calculating the bias and MSE for the `joineR` model, we compare the estimates output to the combination of true parameters on the left hand side of the equations in (B.4). In order to calculate the coverage probabilities for γ_1^* and γ_2^*, we estimate standard errors by bootstrapping. Specifically, we select, with replacement, data for 500 subjects and run a model on that bootstrap sample. We do this 100 times for each of the 100 simulated datasets and calculate $\gamma_1^* - \alpha\beta_2$ and $\gamma_2^* - \alpha\beta_3$. We then use the standard deviation of these values as an estimate of the standard error.

The coefficients for the covariates in the survival submodel of Scenario 2 work out to be exactly the same as above and standard errors were again found by bootstrapping.

In Scenario 3 the `JM` model is in a different form than the true model. The data are generated from

$$Y = \beta_0 + \beta_1 t + \beta_2 X_1 + \beta_3 X_2 + U + e$$

$$h(t) = \exp(\gamma_0) \exp(\gamma_1 X_1 + \gamma_2 X_2 + \alpha U)$$

Whereas `JM` fits a model of the form

$$h(t) = h_0(t) \exp(\gamma_1^* X_1 + \gamma_2^* X_2 + \alpha(\beta_0 + \beta_1 t + \beta_2 X_1 + \beta_3 X_2 + U))$$

$$= h_0(t) \exp(\alpha(\beta_0 + \beta_1 t)) \exp((\gamma_1^* + \alpha\beta_2)X_1 + (\gamma_2^* + \alpha\beta_3)X_2 + \alpha U)$$

135

So we see that

$$\gamma_1^* + \alpha\beta_2 = \gamma_1$$

$$\gamma_2^* + \alpha\beta_3 = \gamma_2$$

Bias and MSE for γ_1^* and γ_2^* are calculated compared to the combinations $\gamma_1 - \alpha\beta_2$ and $\gamma_2 - \alpha\beta_3$, respectively. Standard errors are estimated with bootstrapping similar to the method described above for Scenario 1.

B.3 Chapter 3 Cause-Specific Proportional Hazards Model Survival Functions Derivations

For Model W PH

$$
\begin{aligned}
S_k(T_i|D_i, W_i, U_i, \gamma_k, \alpha_k, \theta_k) &= \exp\left(-\int_0^{T_i} \frac{\gamma_k}{\alpha_{0k}} s^{\gamma_k - 1} \exp\left(\alpha_{1k}X_{2i} + \theta_k U_{0i}\right) ds\right) \\
&= \exp\left(-\exp\left(\alpha_{1k}X_{2i} + \theta_k U_{0i}\right) \frac{\gamma_k}{\alpha_{0k}} \frac{s^{\gamma_k}}{\gamma_k}\Big|_0^{T_i}\right) \\
&= \exp\left(-\exp\left(\alpha_{1k}X_{2i} + \theta_k U_{0i}\right) \frac{1}{\alpha_{0k}}(T_i^{\gamma_k} - 0)\right) \\
&= \exp\left(-\frac{1}{\alpha_{0k}}T_i^{\gamma_k} \exp\left(\alpha_{1k}X_{2i} + \theta_k U_{0i}\right)\right)
\end{aligned}
$$

B.4 Chapter 4 Cause-specific Hazard Function Derivations

We have

$$S(t_1, t_2) = \exp\left(-\left((\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right)^{\delta}\right)$$

and

$$\begin{aligned}
h_k(t) &= -\frac{\partial}{\partial t_k} \log\left(S(t_1, t_2)\right)|_{(t_1, t_2) = (t,t)} \\
&= -\frac{\partial}{\partial t_k} \log\left(\exp\left(-\left((\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right)^{\delta}\right)\right)|_{(t_1, t_2) = (t,t)} \\
&= -\frac{\partial}{\partial t_k} \left(-\left((\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right)^{\delta}\right)|_{(t_1, t_2) = (t,t)} \\
&= \delta\left((\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right)^{\delta-1} \frac{1}{\delta}(\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}-1} \mu_k \gamma_k t_k^{\gamma_k - 1}|_{(t_1, t_2) = (t,t)} \\
&= \gamma_k \mu_k^{\frac{1}{\delta}} t^{\frac{\gamma_k}{\delta} - 1} \left((\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t^{\gamma_2})^{\frac{1}{\delta}}\right)^{\delta-1}
\end{aligned}$$

B.5 Inverse H Function for Dependent Competing Risks in Chapter 4

$$H_k(t) = \mu_k t^{\gamma_k}$$

$$y = \mu_k t^{\gamma_k}$$

$$\frac{y}{\mu_k} = t^{\gamma_k}$$

$$\left(\frac{y}{\mu_k}\right)^{1/\gamma_k} = t$$

$$\Rightarrow H_k^{-1}(y) = \left(\frac{y}{\mu_k}\right)^{1/\gamma_k}$$

So our event times are $T_k = H_k^{-1}(Z_k^\delta)$, meaning

$$T_k = \left(\frac{Z_k^\delta}{\mu_k}\right)^{1/\gamma_k}$$

B.6   Transition Hazards in Multistate Submodel from Chapter 5

We write this out for $\lambda_{1|2}$ below and generalizing to any two states other than states 1 and 2 is straightforward.

$$\lambda_{1|2}(t \mid t_2) = \lim_{\Delta t \searrow 0} \frac{Pr\left(t < T_1 \le t + \Delta t \mid T_1 > t, T_2 = t_2\right)}{\Delta t}$$

$$= -\frac{f(t, t_2)}{\left(\frac{\partial}{\partial t_2} S(t, t_2)\right)} \tag{B.5}$$

The joint probability density function is $f(t_1, t_2) = \frac{(-1)^2 \partial^2 S(t_1, t_2)}{\partial t_1 \partial t_2}$ (Crowder, 2012, p.105). Hence

$$
\begin{aligned}
f(t_1, t_2) &= \frac{\partial^2}{\partial t_1 \partial t_2} \exp\left(-\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^\delta\right) \\
&= \frac{\partial}{\partial t_2}\left[\exp\left(-\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^\delta\right)\left(-\delta\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\frac{1}{\delta}(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}-1}\mu_1\gamma_1 t_1^{\gamma_1-1}\right] \\
&= \frac{1}{\delta}\mu_1^{\frac{1}{\delta}}\gamma_1 t_1^{\frac{\gamma_1}{\delta}-1}\left[\exp\left(-\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^\delta\right)\left(-\delta(\delta-1)\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-2}\right)\cdot\right. \\
&\quad \left(\frac{1}{\delta}(\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}-1}\mu_2\gamma_2 t_2^{\gamma_2-1}\right) \\
&\quad + \left(-\delta\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\exp\left(-\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^\delta\right)\left(-\delta\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\cdot \\
&\quad \left.\left(\frac{1}{\delta}(\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}-1}\mu_2\gamma_2 t_2^{\gamma_2-1}\right)\right] \\
&= \frac{1}{\delta}\mu_1^{\frac{1}{\delta}}\gamma_1 t_1^{\frac{\gamma_1}{\delta}-1}\cdot\frac{1}{\delta}\mu_2^{\frac{1}{\delta}}\gamma_2 t_2^{\frac{\gamma_2}{\delta}-1}\cdot\exp\left(-\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^\delta\right)\cdot \\
&\quad \left[\delta(1-\delta)\left(\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-2}\right) + \delta^2\left(\left[(\mu_1 t_1^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{2\delta-2}\right)\right]
\end{aligned}
$$

The denominator from (B.5) is

$$\frac{\partial}{\partial t_2} S(t, t_2) = \frac{\partial}{\partial t_2} \exp\left(-\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta}\right)$$

$$= \exp\left(-\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta}\right)\left(-\delta\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\frac{1}{\delta}(\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}-1}\mu_2\gamma_2 t_2^{\gamma_2-1}$$

$$= -\exp\left(-\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta}\right)\left(\delta\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\frac{1}{\delta}\mu_2^{\frac{1}{\delta}}\gamma_2 t_2^{\frac{\gamma_2}{\delta}-1}$$

Giving

$$\Rightarrow \lambda_{1|2}(t \mid t_2) = -\frac{f(t, t_2)}{\left(\frac{\partial}{\partial t_2} S(t, t_2)\right)}$$

$$= -\frac{1}{\delta^2}\mu_1^{\frac{1}{\delta}}\gamma_1 t^{\frac{\gamma_1}{\delta}-1} \cdot \mu_2^{\frac{1}{\delta}}\gamma_2 t_2^{\frac{\gamma_2}{\delta}-1} \cdot \exp\left(-\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta}\right) \cdot$$

$$\frac{\left[\delta(1-\delta)\left(\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-2}\right) + \delta^2\left(\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{2\delta-2}\right)\right]}{-\exp\left(-\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta}\right)\left(\delta\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\frac{1}{\delta}\mu_2^{\frac{1}{\delta}}\gamma_2 t_2^{\frac{\gamma_2}{\delta}-1}}$$

$$= \frac{1}{\delta}\mu_1^{\frac{1}{\delta}}\gamma_1 t^{\frac{\gamma_1}{\delta}-1}\left[(1-\delta)\left(\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-2-\delta+1}\right) + \left(\delta\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{2\delta-2-\delta+1}\right)\right]$$

$$= \frac{1}{\delta}\mu_1^{\frac{1}{\delta}}\gamma_1 t^{\frac{\gamma_1}{\delta}-1}\left[(1-\delta)\left(\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{-1}\right) + \left(\delta\left[(\mu_1 t^{\gamma_1})^{\frac{1}{\delta}} + (\mu_2 t_2^{\gamma_2})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\right]$$

In general for transition to state l from state k, the conditional hazard is

$$\lambda_{l|k}(t|t_k) = \frac{1}{\delta}\mu_l^{\frac{1}{\delta}}\gamma_l t^{\frac{\gamma_l}{\delta}-1}\left[(1-\delta)\left(\left[(\mu_l t^{\gamma_l})^{\frac{1}{\delta}} + (\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}}\right]^{-1}\right) + \left(\delta\left[(\mu_l t^{\gamma_l})^{\frac{1}{\delta}} + (\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\right]$$

$$(B.6)$$

and the total hazard from state k is

$$\lambda_k(t) = \sum_{l \neq k}\left(\frac{1}{\delta}\mu_l^{\frac{1}{\delta}}\gamma_l t^{\frac{\gamma_l}{\delta}-1}\left[(1-\delta)\left(\left[(\mu_l t^{\gamma_l})^{\frac{1}{\delta}} + (\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}}\right]^{-1}\right) + \left(\delta\left[(\mu_l t^{\gamma_l})^{\frac{1}{\delta}} + (\mu_k t_k^{\gamma_k})^{\frac{1}{\delta}}\right]^{\delta-1}\right)\right]\right)$$

$$(B.7)$$

Appendix C

Other Joint Modeling Software Implementations

C.1 PROC NLMIXED in SAS

Both the %JM and %JMfit SAS macros call `PROC NLMIXED` to fit the joint model. `PROC NLMIXED` can also be used separately to fit a joint model in SAS that can not be fit with the macros described above. The `NLMIXED` procedure was designed for fitting nonlinear mixed models but has been used for fitting joint models (Gould et al., 2014; SAS Institute, 2015b,a; Schabenberger, 2004)(Littell et al., 2006, p.595). The user must specify the joint likelihood of the joint model but this procedure allows for much more flexibility. The association can take any form that the user writes into the joint likelihood. `PROC NLMIXED` fits the models by approximating the likelihood integrated over the random effects and then maximizing (SAS Institute, 2015b). Different options are available for integral approximation including adaptive Gaussian quadrature and first-order Taylor series expansion.

C.2 %JMfit Macro in SAS

The %JMfit macro for the SAS software was created to fit joint models and while simultaneously assessing the fit of the models. A full description of the macro is given by (Zhang et al., 2016). The %JMfit macro fits only shared parameter models as in (1.3). This macro can fit a joint

model in one of four forms: SPM1L, SPM1Q, SPM2L, and SPM2Q described in (C.1).

$$\text{SPM1L: } h_i(t) = h_0(t)\exp\{X_{i2}\gamma + \alpha(U_{0i} + U_{1i}t)\}$$

$$\text{SPM1Q: } h_i(t) = h_0(t)\exp\{X_{i2}\gamma + \alpha(U_{0i} + U_{1i}t + U_{2i}t^2)\}$$

$$\text{SPM2L: } h_i(t) = h_0(t)\exp\{X_{i2}\gamma + \alpha(U_{0i} + U_{1i})\}$$ (C.1)

$$\text{SPM2Q: } h_i(t) = h_0(t)\exp\{X_{i2}\gamma + \alpha(U_{0i} + U_{1i} + U_{2i})\}$$

One addition to %JMfit not readily available in other software is built-in decomposition of AIC, BIC, ΔAIC, and ΔBIC for examining goodness-of-fit, and comparing between the joint model forms (SPM1L, SPM1Q, SPM2L, and SPM2Q) (Zhang et al., 2016). Estimation is done with the PROC NLMIXED procedure. This macro was not able to fit joint models to our simulated data or the adrenal cancer data. This macro generally had issues with the optimization convergence or quadrature accuracy.

C.3   gsem Command in Stata

The `gsem` command in the `Stata` software was created to fit generalized structural equation models and multilevel data. It can be used for joint modeling with flexible latent processes but the survival outcome must be modeled parametrically (Marchenko, 2016)(Stata Corp LP, 2015, p.449-473). Common distributions for the survival submodel can be specified such as Exponential, Weibull, Gamma, or log-Normal (Stata Corp LP, 2015, p.95,467). The association is through shared parameters as in (1.3) but it is flexible since any random terms can be included.

C.4   jmxtstset Command in Stata

The unofficial Stata command `jmxtstcox` fits joint models (Marchenko, 2016) and is currently available from the authors. The joint model in this command has a random-intercept Cox model and uses nonparametric maximum likelihood for estimation of model parameters. This command can also accommodate stratification and standard post-estimation commands in Stata

such as `text`, `predict`, and `lincom` (Marchenko, 2016). The `jmxtstcox` command does not yet support random coefficients models; only a random intercept shared parameter model can be fit, but extensions are planned.

C.5 MATLAB

`MATLAB`, a mathematical programming language, has been used for joint modeling. Estimation of parameters from the joint likelihood can be achieved with the `fmisearch` and `fmincon` functions, which are unconstrained and constrained nonlinear optimization function, respectively. This has been used by (Thomasson, 2012) and (Hwang, 2013) who also provide sample code for MCMC estimation.

Appendix D

Additional Simulations

D.1   Additional Simulations with Conditionally Independent Competing Risks in Chapter 3

True surv.	True Value	N	Param	Mean	Bias*1000	MSE*1000	CP
Weibull	3.4	160	β_0	2.90	-495	478	82
(Model W)	-0.15		β_1	-0.14	5	4	95
	-5.8		β_2	-4.99	809	1038	75
	-1.4		β_3	-1.37	32	55	96
	4.3		σ_U	4.25	-50	66	94
	2.2		σ_ϵ	2.20	3	1	95
	1.3		$\alpha_{0,1}$	1.34	36	33	95
	-0.1		$\alpha_{1,1}$	-0.09	6	14	94
	3.2		$\alpha_{0,2}$	3.11	-93	187	95
	-0.2		$\alpha_{1,2}$	-0.20	-1	43	96
	0.1		θ_1	0.10	4	2	95
	-0.1		θ_2	-0.09	14	7	93
	0.75		γ_1	0.75	4	5	95
	0.5		γ_2	0.54	42	7	93
Weibull	3.4	1000	β_0	3.30	-97	55	93
(Model W)	-0.15		β_1	-0.15	1	1	97
	-5.8		β_2	-5.66	137	97	92
	-1.4		β_3	-1.39	11	8	94
	4.3		σ_U	4.30	-4	11	95
	2.2		σ_ϵ	2.20	1	0	96
	1.3		$\alpha_{0,1}$	1.31	8	5	95
	-0.1		$\alpha_{1,1}$	-0.10	2	2	96
	3.2		$\alpha_{0,2}$	3.18	-23	45	92
	-0.2		$\alpha_{1,2}$	-0.20	-3	8	97
	0.1		θ_1	0.10	1	0	96
	-0.1		θ_2	-0.10	3	1	96
	0.75		γ_1	0.75	2	1	97
	0.5		γ_2	0.51	8	1	93

Table D.1: Scenario 2 simulation results for data generated with Model W as the truth. Data was generated for either N=160 (top) or N=1000 (bottom). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP).

True surv.	True Value	N	Param	Mean	Bias*1000	MSE*1000	CP
log-Normal	3.4	160	β_0	2.88	-515	518	85
(Model L)	-0.15		β_1	-0.15	4	4	96
	-5.8		β_2	-4.98	817	1061	77
	-1.4		β_3	-1.35	54	56	94
	4.3		σ_U	4.24	-57	54	97
	2.2		σ_ϵ	2.20	-3	2	94
	0.7		$\alpha_{0,1}$	0.69	-10	29	96
	-0.05		$\alpha_{1,1}$	-0.04	8	11	96
	2.8		$\alpha_{0,2}$	2.50	-302	259	88
	-0.25		$\alpha_{1,2}$	-0.24	10	47	97
	0.1		θ_1	0.10	3	1	98
	-0.1		θ_2	-0.09	7	6	97
	1.6		γ_1	1.61	10	17	96
	3.2		γ_2	2.89	-306	185	86
log-Normal	3.4	1000	β_0	3.32	-81	48	93
(Model L)	-0.15		β_1	-0.15	1	1	94
	-5.8		β_2	-5.67	129	96	93
	-1.4		β_3	-1.39	6	9	96
	4.3		σ_U	4.29	-7	11	95
	2.2		σ_ϵ	2.20	0	0	95
	0.7		$\alpha_{0,1}$	0.69	-11	5	94
	-0.05		$\alpha_{1,1}$	-0.05	1	1	96
	2.8		$\alpha_{0,2}$	2.71	-86	49	93
	-0.25		$\alpha_{1,2}$	-0.24	8	9	94
	0.1		θ_1	0.10	0	0	96
	-0.1		θ_2	-0.10	3	1	96
	1.6		γ_1	1.59	-12	3	96
	3.2		γ_2	3.13	-68	32	92

Table D.2: Scenario 2 simulation results for data generated with Model L as the truth. Data was generated for either N=160 (top) or N=1000 (bottom). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP).

True surv.	True Value	N	Param	Mean	Bias*1000	MSE*1000	CP
Weibull	3.0	160	β_0	2.92	-81	144	94
(Model W)	-1.0		β_1	-0.99	11	6	96
	-2.0		β_2	-1.91	85	113	94
	-1.0		β_3	-0.94	57	58	91
	4.0		σ_U	3.92	-76	73	93
	2.0		σ_ϵ	2.00	0	3	96
	1.0		$\alpha_{0,1}$	1.02	20	164	94
	-1.5		$\alpha_{1,1}$	-1.44	62	71	93
	3.0		$\alpha_{0,2}$	2.83	-170	295	95
	-1.0		$\alpha_{1,2}$	-1.04	-36	85	94
	1.0		θ_1	1.04	36	11	94
	-1.0		θ_2	-0.98	23	18	95
	0.75		γ_1	0.82	70	20	96
	0.5		γ_2	0.55	47	7	90
Weibull	3.0	1000	β_0	3.00	4	22	96
(Model W)	-1.0		β_1	-1.00	3	1	95
	-2.0		β_2	-1.98	17	22	93
	-1.0		β_3	-1.00	-3	7	96
	4.0		σ_U	3.98	-22	10	95
	2.0		σ_ϵ	2.00	0	0	95
	1.0		$\alpha_{0,1}$	1.02	21	26	93
	-1.5		$\alpha_{1,1}$	-1.51	-9	8	97
	3.0		$\alpha_{0,2}$	2.97	-31	72	94
	-1.0		$\alpha_{1,2}$	-0.99	14	15	94
	1.0		θ_1	1.00	4	1	97
	-1.0		θ_2	-1.00	3	3	93
	0.75		γ_1	0.76	8	2	98
	0.5		γ_2	0.50	2	1	97

Table D.3: Scenario 3 simulation results for data generated with Model W as the truth. Data was generated for either N=160 (top) or N=1000 (bottom). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP).

True surv.	True Value	N	Param	Mean	Bias*1000	MSE*1000	CP
log-Normal	3.0	160	β_0	2.94	-64	177	90
(Model L)	-1.0		β_1	-1.00	4	4	95
	-2.0		β_2	-1.85	147	166	92
	-1.0		β_3	-0.98	22	37	98
	4.0		σ_U	3.94	-60	73	94
	2.0		σ_ϵ	2.00	3	2	96
	1.0		$\alpha_{0,1}$	0.99	-8	185	96
	-1.5		$\alpha_{1,1}$	-1.47	28	61	97
	3.0		$\alpha_{0,2}$	2.62	-378	485	90
	-1.0		$\alpha_{1,2}$	-0.98	20	106	93
	1.0		θ_1	1.01	8	9	96
	-1.0		θ_2	-0.95	52	21	95
	2.0		γ_1	1.89	-108	61	91
	3.0		γ_2	2.76	-244	153	85
log-Normal	3.0	1000	β_0	2.99	-6	26	93
(Model L)	-1.0		β_1	-1.00	-1	1	93
	-2.0		β_2	-1.97	27	24	94
	-1.0		β_3	-1.00	2	9	93
	4.0		σ_U	4.00	-4	10	95
	2.0		σ_ϵ	2.00	1	0	94
	1.0		$\alpha_{0,1}$	1.01	7	28	98
	-1.5		$\alpha_{1,1}$	-1.50	5	13	93
	3.0		$\alpha_{0,2}$	2.94	-63	84	92
	-1.0		$\alpha_{1,2}$	-1.01	-6	18	96
	1.0		θ_1	1.00	5	1	96
	-1.0		θ_2	-0.99	7	3	95
	2.0		γ_1	1.99	-15	9	95
	3.0		γ_2	2.93	-72	25	92

Table D.4: Scenario 3 simulation results for data generated with Model L as the truth. Data was generated for either N=160 (top) or N=1000 (bottom). Data was generated 200 times. For each parameter the results include the mean of the 200 posterior means, the bias*1000, the mean squared error (MSE)*1000, and the coverage probability (CP).

D.2 Chapter 4 Chain Initial Values for Simulation

Parameter	Chain 1	Chain 2	Chain 3	Chain 4
β_0	3.5	2.5	1.5	2
β_1	0.15	0.55	-0.15	0.05
β_2	2	1	3	0.5
β_3	3	2	4	1.5
σ_U	0.5	1	2	1.5
σ_ϵ	2	1	3	0.5
$\alpha_{0,1}$	1.2	2.2	0.2	0.6
$\alpha_{1,1}$	-0.4	0.4	-1.4	-0.2
$\alpha_{0,2}$	1.1	2.1	0.1	0.5
$\alpha_{1,2}$	-0.5	0.5	-1.5	-0.1
θ_1	-1.1	1.1	-2.1	-0.1
θ_2	-0.9	0.9	-1.9	-0.5
γ_1	5	6	4	5.5
γ_2	8	7	6	9
δ	0.4	0.3	0.7	0.6
U_{0i}	0	1	2	0.5

Table D.5: Initial values for four chains used to fit Model W to the simulated data.

Appendix E

Code

E.1 Code for Joint Longitudinal and Single Survival Event from Chapter 2

E.1.1   R Code

```
library(JM)
library(nlme)
library(survival)
library(joineR)
library(JMbayes)
library(R2WinBUGS)
library(foreign)
library(sas7bdat)


set.seed(5202017)


##################################################
#Random Intercept Model
##################################################
```

```r
#Software Comparison Simulation Data


#total run time for R
t.tot.start<- proc.time()[1]


#number of subjects
n=500


##################################################
# Simulation loops
#################################################


softwareNames=c("JM_Weib","JM_Weib_No_Adpt","JM_PW",
"JM_Spl","joineR","Two-Stage")


#number of simulations
nSim=100


#load("data_01202018_int.RData")


t.tot.start<- proc.time()[1]
###################################
# Model Fitting
###################################
####################
# two-stage model
```

```
####################

for(k in 1:nSim){

  t<- proc.time()[1]


  tryCatch({

    fittry=rep(NA,length(simdataLongit[[k]][,1]))

    #fit separate models

    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],

    random=~1|subjID,control=lmeControl(opt='optim'))


    for(i in 1:length(simdataLongit[[k]][,1])){

      fittry[i]=sum(c(lmemodel$coefficients$fixed,1)*

      c(1,simdataLongit[[k]]$timevar[i],

      simdataLongit[[k]]$Z1[i],

      simdataLongit[[k]]$Z2[i],

      as.vector(lmemodel$coefficients$random$subjID[

      simdataLongit[[k]]$subjID[i]])))

    }

    subjID=simdataLongit[[k]]$subjID

    timevar=simdataLongit[[k]]$timevar

    lmefits=cbind(subjID,fittry,timevar)


    subjID=unique(subjID)


    survdata=cbind(simdataSurv[[k]],subjID)


    tostata=merge(lmefits,survdata,by=c("subjID"))
```

```r
tostatasave=tostata

tostata=tostata[order(tostata$subjID),]

tostata$last=rep(0, length(tostata$X))

tostata$time1=rep(-1, length(tostata$X))

tostata$time2=rep(-1, length(tostata$X))

tostata$event=rep(0, length(tostata$X))


for(i in 1:length(tostata$X)){

  if(i==length(tostata$X)){

    tostata$last[i]=1

  }else if(tostata$subjID[i] != tostata$subjID[i+1]){

    tostata$last[i]=1

  }

}


for(i in 1:length(tostata$X)){

  if(tostata$last[i]==1){

    tostata$time1[i]=tostata$timevar[i]

    tostata$time2[i]=tostata$X[i]

    if( tostata$D[i]==1){

      tostata$event[i]=1

    }

  }else{

    tostata$time1[i]=tostata$timevar[i]

    tostata$time2[i]=tostata$timevar[i+1]

  }
```

```
    }


    coxmodel=coxph(Surv(time1,time2,event)~Z1+Z2+fittry,

    data=tostata,x=TRUE)


    twostagefits[[k]]=list(lmemodel, coxmodel)
  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  #save runtime
  runtimes[k,nSoftware]=proc.time()[1]-t
  }


############################################
#    JM
############################################
for(k in 1:nSim){
  t<- proc.time()[1]


  ################# Weibull Baseline Haz
  tryCatch({
    #fit separate models
    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],

    random=~1|subjID,control=lmeControl(opt='optim'))

    coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)


    #separate models are input to joint model
    #fit joint model
```

```
    JM.model=jointModel(lmemodel,coxmodel,"timevar",

    method="weibull-PH-aGH", parameterization = "value")



    JMfits.weib[[k]]=JM.model

}, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})

#save runtime

runtimes[k,1]=proc.time()[1]-t



}

for(k in 1:nSim){

################# Weibull Baseline Haz  NOT ADAPTED

t<- proc.time()[1]

#fit separate models

tryCatch({

    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],

    random=~1|subjID,control=lmeControl(opt='optim'))

    coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)



    #separate models are input to joint model

    #fit joint model

    JM.model=jointModel(lmemodel,coxmodel,"timevar",

    method="weibull-PH-GH")



    JMfits.weibNoAdapt[[k]]=JM.model

}, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})

#save runtime

runtimes[k,2]=proc.time()[1]-t
```

```
}


for(k in 1:nSim){

  ################## Piecewise Baseline Haz

  t<- proc.time()[1]

  tryCatch({

    #fit separate models

    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],

    random=~1|subjID,control=lmeControl(opt='optim'))

    coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)


    #separate models are input to joint model

    #fit joint model

    JM.model=jointModel(lmemodel,coxmodel,"timevar",

    method="piecewise-PH-aGH", parameterization = "value")


    JMfits.pw[[k]]=JM.model

  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})

  #save runtime

  runtimes[k,3]=proc.time()[1]-t


}

for(k in 1:nSim){

  ################ Splines Baseline Haz

  t<- proc.time()[1]

  tryCatch({
```

155

```r
    #fit separate models

    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],

    random=~1|subjID,control=lmeControl(opt='optim'))

    coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)


    #separate models are input to joint model

    #fit joint model

    JM.model=jointModel(lmemodel,coxmodel,"timevar",

    method="spline-PH-aGH", parameterization = "value")


    JMfits.spl[[k]]=JM.model
  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  #save runtime
  runtimes[k,4]=proc.time()[1]-t


}


####################################################
#    joineR
####################################################


#Note joineR only fits Cox PH models, no other options
###################
# Same association
###################
for(k in 1:nSim){
  t<- proc.time()[1]
```

```
#need specific data
#for longitudinal model need: longitudinal measurements,
#time of measurements, ID
lmedata.joineR=simdataLongit[[k]][,c("Y","subjID","timevar")]
#for survival data need: survival time, indicator, ID
survdata.joineR=simdataSurv[[k]][,c("X","D")]
survdata.joineR=cbind(survdata.joineR,1:n)
colnames(survdata.joineR)=c("x","D","subjID")
#baseline covariate data: covariates, ID
covariates.joineR=simdataSurv[[k]][,c("Z1","Z2")]
covariates.joineR=cbind(covariates.joineR,1:n)
colnames(covariates.joineR)=c("Z1","Z2","subjID")


tryCatch({
  #create the data
  joineRdata=jointdata(lmedata.joineR,survdata.joineR,
  covariates.joineR, id.col="subjID",time.col="timevar")


  #fit the joint model
  joineR.model=joint(joineRdata,long.formula=Y~1+timevar+Z1+Z2,
     surv.formula=Surv(X,D)~Z1+Z2,model="int")
  joineRfits[[k]]=joineR.model#joineR.model$coefficients
}, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
runtimes[k,5]=proc.time()[1]-t


}
```

```
###############################################

#put names on runtimes table for easier interpretation
runtimes=as.data.frame(runtimes)
colnames(runtimes)=softwareNames

### Calc total run time for R
totalruntime=proc.time()[1]-t.tot.start
totalruntime=totalruntime/60 #time in minutes
totalruntime.h=totalruntime/60 #time in hours
totalruntime.d=totalruntime.h/24 #time in days
totalruntime
totalruntime.h
totalruntime.d


#printing mean runtimes for software
colMeans(runtimes)



###################################################
#Random Coefficients Model
###################################################


t.tot.start<- proc.time()[1]
###############################################
```

```
# Model Fitting

####################################################

#load("data_01202018_slope.RData")


####################

# two-stage model

####################

for(k in 1:nSim){

  t<- proc.time()[1]


  tryCatch({

    fittry=rep(NA,length(simdataLongit[[k]][,1]))

    #fit separate models

    lmemodel=lme(Y~timevar+Z1+Z2+timevar*Z1,

    data=simdataLongit[[k]],

    random=~1+timevar|subjID,control=lmeControl(opt='optim'))


    randeffs=as.data.frame(lmemodel$coefficients$random)


    for(i in 1:length(simdataLongit[[k]][,1])){

      fittry[i]=sum(c(lmemodel$coefficients$fixed,1,

      simdataLongit[[k]]$timevar[i])*

                    c(1,simdataLongit[[k]]$timevar[i],

                    simdataLongit[[k]]$Z1[i],

                    simdataLongit[[k]]$Z2[i],

                      simdataLongit[[k]]$timevar[i]*
```

```r
                              simdataLongit[[k]]$Z1[i],
                               lmemodel$coefficients$random$subjID[
                               simdataLongit[[k]]$subjID[i],][1],
                               lmemodel$coefficients$random$subjID[
                               simdataLongit[[k]]$subjID[i],][2]))
}
subjID=simdataLongit[[k]]$subjID
timevar=simdataLongit[[k]]$timevar
lmefits=cbind(subjID,fittry,timevar)


subjID=unique(subjID)


survdata=cbind(simdataSurv[[k]],subjID)


tostata=merge(lmefits,survdata,by=c("subjID"))
tostatasave=tostata
tostata=tostata[order(tostata$subjID),]
tostata$last=rep(0, length(tostata$X))
tostata$time1=rep(-1, length(tostata$X))
tostata$time2=rep(-1, length(tostata$X))
tostata$event=rep(0, length(tostata$X))


for(i in 1:length(tostata$X)){
  if(i==length(tostata$X)){
    tostata$last[i]=1
  }else if(tostata$subjID[i] != tostata$subjID[i+1]){
```

160

```r
        tostata$last[i]=1

      }

    }



  for(i in 1:length(tostata$X)){

     if(tostata$last[i]==1){

        tostata$time1[i]=tostata$timevar[i]

        tostata$time2[i]=tostata$X[i]

        if( tostata$D[i]==1){

           tostata$event[i]=1

        }

     }else{

        tostata$time1[i]=tostata$timevar[i]

        tostata$time2[i]=tostata$timevar[i+1]

     }



   }



   coxmodel=coxph(Surv(time1,time2,event)~Z1+Z2+fittry,

   data=tostata,x=TRUE)



   twostagefits[[k]]=list(lmemodel, coxmodel)

 }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})

 #save runtime

 runtimes[k,nSoftware]=proc.time()[1]-t



}
```

```
k
#################################################
#       JM
#################################################
for(k in 1:nSim){
  t<- proc.time()[1]


  ################# Weibull Baseline Haz
  tryCatch({
    #fit separate models
    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],
    random=~1+timevar|subjID,control=lmeControl(opt='optim'))
    coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)


    #separate models are input to joint model
    #fit joint model


    JM.model=jointModel(lmemodel,coxmodel,"timevar",
    method="weibull-PH-aGH")


    JMfits.weib[[k]]=JM.model
  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  #save runtime
  runtimes[k,1]=proc.time()[1]-t


}
```

```
k

for(k in 1:nSim){

  ################ Weibull Baseline Haz  NOT ADAPTED

  t<- proc.time()[1]

  #fit separate models

  tryCatch({

    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],

    random=~1+timevar|subjID,control=lmeControl(opt='optim'))

    coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)


    #separate models are input to joint model

    #fit joint model

    JM.model=jointModel(lmemodel,coxmodel,"timevar",

    method="weibull-PH-GH")


    JMfits.weibNoAdapt[[k]]=JM.model

  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})

  #save runtime

  runtimes[k,2]=proc.time()[1]-t


}

k

for(k in 1:nSim){

  ################ Piecewise Baseline Haz

  t<- proc.time()[1]

  tryCatch({

    #fit separate models
```

```r
      lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],

      random=~1+timevar|subjID,control=lmeControl(opt='optim'))

      coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)


      #separate models are input to joint model

      #fit joint model

      JM.model=jointModel(lmemodel,coxmodel,"timevar",

      method="piecewise-PH-aGH")


      JMfits.pw[[k]]=JM.model

  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})

  #save runtime

  runtimes[k,3]=proc.time()[1]-t


}

k

for(k in 1:nSim){

  ################# Splines Baseline Haz

  t<- proc.time()[1]

  tryCatch({

    #fit separate models

    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],

    random=~1+timevar|subjID,control=lmeControl(opt='optim'))

    coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)


    #separate models are input to joint model

    #fit joint model
```

```r
    JM.model=jointModel(lmemodel,coxmodel,"timevar",
      method="spline-PH-aGH")


    JMfits.spl[[k]]=JM.model
  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  #save runtime
  runtimes[k,4]=proc.time()[1]-t


}
k


##################################################
#     joineR
#################################################


#Note joineR only fits Cox PH models, no other options
###################
# Same association
##################
for(k in 1:nSim){
  t<- proc.time()[1]


  #need specific data
  #for longitudinal model need: longitudinal measurements,
  #time of measurements, ID
  lmedata.joineR=simdataLongit[[k]][,c("Y","subjID","timevar")]
  #for survival data need: survival time, indicator, ID
```

```
survdata.joineR=simdataSurv[[k]][,c("X","D")]

survdata.joineR=cbind(survdata.joineR,1:n)

colnames(survdata.joineR)=c("x","D","subjID")

#baseline covariate data: covariates, ID

covariates.joineR=simdataSurv[[k]][,c("Z1","Z2")]

covariates.joineR=cbind(covariates.joineR,1:n)

colnames(covariates.joineR)=c("Z1","Z2","subjID")


tryCatch({

  #create the data

  joineRdata=jointdata(lmedata.joineR,survdata.joineR,

  covariates.joineR, id.col="subjID",time.col="timevar")


  #fit the joint model

  joineR.model=joint(joineRdata,long.formula=Y~1+timevar+Z1+Z2,

                     surv.formula=Surv(X,D)~Z1+Z2,

                     model="intslope",sepassoc=FALSE)

  joineRfits[[k]]=joineR.model#joineR.model$coefficients

}, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})

runtimes[k,5]=proc.time()[1]-t


}

k

#########################

# Separate association

#########################

for(k in 1:nSim){
```

```
t<- proc.time()[1]


#need specific data
#for longitudinal model need: longitudinal measurements,
#time of measurements, ID
lmedata.joineR=simdataLongit[[k]][,c("Y","subjID","timevar")]
#for survival data need: survival time, indicator, ID
survdata.joineR=simdataSurv[[k]][,c("X","D")]
survdata.joineR=cbind(survdata.joineR,1:n)
colnames(survdata.joineR)=c("x","D","subjID")
#baseline covariate data: covariates, ID
covariates.joineR=simdataSurv[[k]][,c("Z1","Z2")]
covariates.joineR=cbind(covariates.joineR,1:n)
colnames(covariates.joineR)=c("Z1","Z2","subjID")


tryCatch({
  #create the data
  joineRdata=jointdata(lmedata.joineR,survdata.joineR,
  covariates.joineR, id.col="subjID",time.col="timevar")


  #fit the joint model
  joineR.model=joint(joineRdata,
  long.formula=Y~1+timevar+Z1+Z2,
  surv.formula=Surv(X,D)~Z1+Z2,model="intslope",sepassoc=TRUE)
  joineRSepfits[[k]]=joineR.model#joineR.model$coefficients
}, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
runtimes[k,6]=proc.time()[1]-t
```

```
}

k


####################################################

#put names on runtimes table for easier interpretation

runtimes=as.data.frame(runtimes)

colnames(runtimes)=softwareNames


### Calc total run time for R

totalruntime=proc.time()[1]-t.tot.start

totalruntime=totalruntime/60 #time in minutes

totalruntime.h=totalruntime/60 #time in hours

totalruntime.d=totalruntime.h/24 #time in days

totalruntime

totalruntime.h

totalruntime.d



#printing mean runtimes for software

colMeans(runtimes)



######################################################

#Random Intercept Model: SHARED PARAM

######################################################
```

```
#load("data_SP_01202018.RData")


t.tot.start<- proc.time()[1]
###################################################
# Model Fitting
###################################################


###################
# two-stage model
###################
for(k in 1:nSim){
  t<- proc.time()[1]


  tryCatch({


    #fit separate models
    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],
    random=~1|subjID,control=lmeControl(opt='optim'))
    randint=as.vector(lmemodel$coefficients$random$subjID)
    tostata=cbind(simdataSurv[[k]],randint)
    coxmodel=coxph(Surv(X,D)~Z1+Z2+randint,data=simdataSurv[[k]],
    x=TRUE)


    twostagefits[[k]]=list(lmemodel, coxmodel)
  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  #save runtime
```

```
runtimes[k,1]=proc.time()[1]-t


}


###################################################
#     JM
###################################################


for(k in 1:nSim){
  ################# Piecewise Baseline Haz
  t<- proc.time()[1]
  tryCatch({
    #fit separate models
    lmemodel=lme(Y~timevar+Z1+Z2,data=simdataLongit[[k]],
    random=~1|subjID,control=lmeControl(opt='optim'))
    coxmodel=coxph(Surv(X,D)~Z1+Z2,data=simdataSurv[[k]],x=TRUE)


    #separate models are input to joint model
    #fit joint model
    JM.model=jointModel(lmemodel,coxmodel,"timevar",
    method="piecewise-PH-aGH")



    JMfits.pw[[k]]=JM.model
  }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  #save runtime
  runtimes[k,2]=proc.time()[1]-t
```

```
}


##################################################
#     joineR
##################################################


#Note joineR only fits Cox PH models, no other options
###################
# Same association
###################
for(k in 1:nSim){
  t<- proc.time()[1]


  #need specific data
  #for longitudinal model need: longitudinal measurements,
  #time of measurements, ID
  lmedata.joineR=simdataLongit[[k]][,c("Y","subjID","timevar")]
  #for survival data need: survival time, indicator, ID
  survdata.joineR=simdataSurv[[k]][,c("X","D")]
  survdata.joineR=cbind(survdata.joineR,1:n)
  colnames(survdata.joineR)=c("x","D","subjID")
  #baseline covariate data: covariates, ID
  covariates.joineR=simdataSurv[[k]][,c("Z1","Z2")]
  covariates.joineR=cbind(covariates.joineR,1:n)
  colnames(covariates.joineR)=c("Z1","Z2","subjID")
```

```r
tryCatch({
  #create the data
  joineRdata=jointdata(lmedata.joineR,survdata.joineR,
  covariates.joineR, id.col="subjID",time.col="timevar")


  #fit the joint model
  joineR.model=joint(joineRdata,long.formula=Y~1+timevar+Z1+Z2,
                     surv.formula=Surv(X,D)~Z1+Z2,model="int")
  joineRfits[[k]]=joineR.model#joineR.model$coefficients
}, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
runtimes[k,3]=proc.time()[1]-t


}


##################################################


#put names on runtimes table for easier interpretation
#runtimes=as.data.frame(runtimes)
#colnames(runtimes)=softwareNames



### Calc total run time for R
totalruntime=proc.time()[1]-t.tot.start
totalruntime=totalruntime/60 #time in minutes
totalruntime.h=totalruntime/60 #time in hours
totalruntime.d=totalruntime.h/24 #time in days
totalruntime
```

172

```
totalruntime.h

totalruntime.d



#printing mean runtimes for software

colMeans(runtimes)
```

E.1.2   SAS Code

```
libname softw "C:\Users\AKC\Box Sync\SoftwareComparisonPaper";

*libname softw "C:\Users\acullen\Box

Sync\SoftwareComparisonPaper";


*OPTIONS NOERRORABEND;

*OPTIONS NOSYNTAXCHECK ;


/* Load for the %JM macro (must change working folder to folder

with the JM macro files)*/

/*

%include 'C:\Users\acullen\Box

Sync\SoftwareComparisonPaper\calculateknotspartition.sas';

%include 'C:\Users\acullen\Box

Sync\SoftwareComparisonPaper\spline.sas';

%include 'C:\Users\acullen\Box

Sync\SoftwareComparisonPaper\ncspline.sas';

%include 'C:\Users\acullen\Box
```

```
Sync\SoftwareComparisonPaper\bspline.sas';

%include 'C:\Users\acullen\Box

Sync\SoftwareComparisonPaper\kronrodrule15p.sas';

%include 'C:\Users\acullen\Box

Sync\SoftwareComparisonPaper\jm.sas';

*/


/**/
%include 'C:\Users\AKC\Box

Sync\SoftwareComparisonPaper\calculateknotspartition.sas';

%include 'C:\Users\AKC\Box

Sync\SoftwareComparisonPaper\spline.sas';

%include 'C:\Users\AKC\Box

Sync\SoftwareComparisonPaper\ncspline.sas';

%include 'C:\Users\AKC\Box

Sync\SoftwareComparisonPaper\bspline.sas';

%include 'C:\Users\AKC\Box

Sync\SoftwareComparisonPaper\kronrodrule15p.sas';

%include 'C:\Users\AKC\Box Sync\SoftwareComparisonPaper\jm.sas';

/**/

proc printto log="C:\Users\AKC\Box Sync\SoftwareComparisonPaper\

saslog_intweibpwc_12052017.log";

run;


proc sort data=softw.final_int_11272017;

by dataset subjID;
```

```
run;

/* Run JM macro for each dataset 1,...,100*/
/* Random Int Model */
%JM(
Data=softw.final_int_11272017,
Where = (dataset=1),
SubjectVar = subjID,
LongiTimeModel= LINEAR,
LongiVar = Y,
LongiTimevar = timevar,
LongiCovariates= Z1 Z2,
LongiTimeInter = Z1,
LongiGMatrix = UN,
LongiModelOptions = METHOD=ML,
EventTimeVar  = X,
EventVar = D,
EventVal = 1,
EventModel = PIECEWISE,
EventCovariates = Z1 Z2,
NLMIXEDOptions = GCONV=0 QTOL=0.0005 QPOINTS=1 METHOD=GAUSS,
SharedParam = CURRENT_VALUE,
AdditionalOptions = CALCULATEEXECTIME SKIPMACROHEADER,
OutputParameters = softw.jmmacrooutput);

data softw.jmmacrooutput;
set softw.jmmacrooutput;
```

```sas
dataset = 1;

run;


/* Random Coefficients Model */


/* Run JM macro for each dataset 1,...,100*/


%JM(

Data=softw.final_slope_11272017,

Where = (dataset=1),

SubjectVar = subjID,

LongiTimeModel= LINEAR,

LongiVar = Y,

LongiTimevar = timevar,

LongiCovariates= Z1 Z2,

LongiTimeInter = Z1,

LongiGMatrix = UN,

LongiModelOptions = METHOD=ML,

EventTimeVar  = X,

EventVar = D,

EventVal = 1,

EventModel = PIECEWISE,

EventCovariates = Z1 Z2,

NLMIXEDOptions = GCONV=0 QTOL=0.0005 QPOINTS=1 METHOD=GAUSS,

SharedParam = CURRENT_VALUE,

AdditionalOptions = CALCULATEEXECTIME SKIPMACROHEADER,

OutputParameters = softw.jmmacrooutput);
```

176

```
data softw.jmmacrooutput;

set softw.jmmacrooutput;

dataset = 1;

run;



/* Shared Parameter Model */


%JM(

Data=softw.final_SP_11272017,

Where = (dataset=1),

SubjectVar = subjID,

LongiTimeModel= LINEAR,

LongiVar = Y,

LongiTimevar = timevar,

LongiCovariates= Z1 Z2,

LongiTimeInter = Z1,

LongiGMatrix = UN,

LongiModelOptions = METHOD=ML,

EventTimeVar  = X,

EventVar = D,

EventVal = 1,

EventModel = PIECEWISE,

EventCovariates = Z1 Z2,

NLMIXEDOptions = GCONV=0 QTOL=0.0005 QPOINTS=1 METHOD=GAUSS,

SharedParam = COEFFICIENTS,
```

```
SharedCoefficients = bi0,

AdditionalOptions = CALCULATEEXECTIME SKIPMACROHEADER,

OutputParameters = softw.jmmacrooutput);


data softw.jmmacrooutput;

set softw.jmmacrooutput;

dataset = 1;

run;
```

E.1.3 Stata Code

```
version 14

clear


use "C:\Users\acullen\Desktop\data_stjm_finalINT.dta", clear

cd "C:\Users\acullen\Desktop"


ssc install stjm

ssc install rcsgen

ssc install estout


capture log close

log using softwSimstjm, text replace



set rmsg on, perm

set more off, perm
```

```
stset time2 if dataset==1, id(subjID) enter(time1)
failure(event==1)
stjm Y Z1_x Z2_x if dataset==1, panel(subjID)  survmodel(weibull)
rfp(0)  gh(25)  survcov(Z1_x Z2_x)  difficult
estimates store model1


stset time2 if dataset==1, id(subjID) enter(time1)
failure(event==1)
stjm Y Z1_x Z2_x if dataset==1, panel(subjID)  survmodel(weibull)
rfp(1)  gh(25)  survcov(Z1_x Z2_x)  difficult
estimates store model1



stset time2 if dataset==1, id(subjID) enter(time1)
failure(event==1)
stjm Y Z1_x Z2_x  if dataset==1, panel(subjID)  survmodel(weibull)
rfp(0)   gh(25)  survcov(Z1_x Z2_x )
difficult nocurrent intassociation
estimates store model1
```

E.2 Stan Code for Joint Model with Conditionally Independent Competing Risks in Chapter 3

E.2.1 For Model W

```
functions{
real mysurv_lpdf(vector t, matrix d, matrix mu, vector gam){
```

```
vector[num_elements(t)] probs;

real lprob;

for(i in 1:num_elements(t)){

probs[i] = 0;

}

for(i in 1:num_elements(t)){          ////// num elements t = N

for(j in 1:num_elements(gam)){   ////// num elements gam = K

probs[i] = probs[i] + d[i,j]*(log(gam[j]) -

gam[j]*log(mu[i,j]) + (gam[j]-1)*log(t[i])) -

(t[i]/mu[i,j])^(gam[j]);

}

}

lprob = sum(probs);

return lprob;

}

}

data{

int K; // Number of competing risks

//(not including independent censoring)

int Ntot; // length of vector of all

//longitudinal outcome observations

int N;   // number of subjects

int P;   // number of covars in longitudinal model

int<lower=1,upper=N> subj[Ntot]; // subject ID

  vector[Ntot] Y;     // longitudinal outcome

int Q; // num covars in survival model

vector[N] surt; // survival times
```

```
matrix[N,K] D; // failure indicators

matrix[Ntot,P] XL; // covariates

matrix[N,Q] XS;

vector[P] mbeta;  // Hyper priors

real<lower=0> sbeta;

real<lower=0> au;

real<lower=0> bu;

real<lower=0> aeps;

real<lower=0> beps;

real<lower=0> su;

matrix[K,Q] malpha;

real<lower=0> salpha;

real<lower=0> agamma;

real<lower=0> bgamma;

vector[K] mtheta;

real<lower=0> stheta;

}

parameters{

vector[P] beta;  // longitudinal regression coeffs

real<lower=0> sigmau;  // std dev of random intercept

real<lower=0> sigmaeps;  // std dev of longitudinal error

vector[N] u0;  // subject-specific random intercept

matrix[K,Q] alpha; // survival regression coeffs

vector<lower=0>[K] gamma; // Weibull shape param

vector[K] theta; // association params

}

model{
```

```
real mul;

matrix[N,K] mus;



// Priors

for(i in 1:P){

beta[i] ~ normal(mbeta[i],sbeta);

}

sigmau ~ gamma(au,bu);

sigmaeps ~ gamma(aeps,beps);

u0 ~ normal(0,sigmau);

for(i in 1:K){

for(j in 1:Q){

alpha[i,j] ~ normal(malpha[i,j],salpha);

}

theta[i] ~ normal(mtheta[i],stheta);

gamma[i] ~ gamma(agamma,bgamma);

}



// Longutudinal Model

for(i in 1:Ntot){

mul = XL[i]*beta + u0[subj[i]];

Y[i] ~ normal(mul,sigmaeps);

}



// Survival Model

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

```
for(j in 1:K){

mus[i,j] = exp(XS[i]*(alpha[j]') + theta[j]*u0[i]);

}

}

surt ~ mysurv(D, mus, gamma);


}
```

E.2.2   For Model L

```
functions{

real mysurv_log(vector t, matrix d, matrix mu, vector gam){

vector[num_elements(t)] probs;

real lprob;

matrix[num_elements(t),num_elements(gam)] thexs;


for(i in 1:num_elements(t)){

probs[i] = 0;

}

for(i in 1:num_elements(t)){

for(j in 1:num_elements(gam)){

thexs[i,j] = (log(t[i])-mu[i,j])/gam[j];

probs[i] =  probs[i] +

d[i,j]*( -(1.0/2.0)*log(2.0) -

(1.0/2.0)*log(pi())-

log(gam[j])-log(t[i])-

(1.0/(2.0*(gam[j]^2)))*((log(t[i]) - mu[i,j])^2)) +

(1.0-d[i,j])*log(1-Phi(thexs[i,j]));
```

```
}

}

lprob = sum(probs);

return lprob;

}

}

data{

int K; // Number of competing risks

//(not including independent censoring)

int Ntot; // length of vector of all

//longitudinal outcome observations

int N;   // number of subjects

int P;   // number of covars in longitudinal model

int<lower=1,upper=N> subj[Ntot]; // subject ID

  vector[Ntot] Y;     // longitudinal outcome

int Q; // num covars in survival model

vector[N] surt; // survival times

matrix[N,K] D; // failure indicators

matrix[Ntot,P] XL; // covariates

matrix[N,Q] XS;

vector[P] mbeta;  // Hyper priors

real<lower=0> sbeta;

real<lower=0> au;

real<lower=0> bu;

real<lower=0> aeps;

real<lower=0> beps;

real<lower=0> su;
```

```stan
matrix[K,Q] malpha;

real<lower=0> salpha;

real<lower=0> agamma;

real<lower=0> bgamma;

vector[K] mtheta;

real<lower=0> stheta;

}

parameters{

vector[P] beta;  // longitudinal regression coeffs

real<lower=0> sigmau;  // std dev of random intercept

real<lower=0> sigmaeps;  // std dev of longitudinal error

vector[N] u0;  // subject-specific random intercept

matrix[K,Q] alpha; // survival regression coeffs

vector<lower=0>[K] gamma; // Weibull shape param

vector[K] theta; // association params

}

model{

real mul;

matrix[N,K] mus;



// Priors

for(i in 1:P){

beta[i] ~ normal(mbeta[i],sbeta);

}

sigmau ~ gamma(au,bu);

sigmaeps ~ gamma(aeps,beps);
```

```
u0 ~ normal(0,sigmau);

for(i in 1:K){

for(j in 1:Q){

alpha[i,j] ~ normal(malpha[i,j],salpha);

}

theta[i] ~ normal(mtheta[i],stheta);

gamma[i] ~ gamma(agamma,bgamma);

}

gamma ~ gamma(agamma,bgamma);


// Longutudinal Model

for(i in 1:Ntot){

mul = XL[i]*beta + u0[subj[i]];

Y[i] ~ normal(mul,sigmaeps);

}

// Survival Model

for(i in 1:N){

for(j in 1:K){

mus[i,j] = XS[i]*(alpha[j]') + theta[j]*u0[i];

}

}

surt ~ mysurv(D, mus, gamma);


}
```

E.3 Stan Code for Joint Models with Dependent Competing Risks in Chapter 4

E.3.1 For Model Fitting

```
functions{

real mysurv_lpdf(vector t, matrix d, matrix chi, vector gam,

real delta){

vector[num_elements(t)] probs;


real lprob;


for(i in 1:num_elements(t)){

probs[i] = 0;

}

for(i in 1:num_elements(t)){          // num elements t = N

for(j in 1:num_elements(gam)){   // num elements gam = K


probs[i] = probs[i] +

d[i,j]*( (delta-1)*log( (chi[i,1]*(t[i]^gam[1]))^(1/delta) +

(chi[i,2]*(t[i]^gam[2]))^(1/delta) ) +

((1/delta)-1)*log( chi[i,j]*(t[i]^gam[j]) ) + log(chi[i,j]) +

log(gam[j]) + (gam[j]-1)*log(t[i]) ) -

 ( (chi[i,1]*(t[i]^gam[1]))^(1/delta) +

 (chi[i,2]*(t[i]^gam[2]))^(1/delta) )^delta

;

}
```

```stan
}
lprob = sum(probs);
return lprob;
}
}
data{
int K; // Number of competing risks
//(not including independent censoring)
int Ntot; // length of vector of all
//longitudinal outcome observations
int N;    // number of subjects
int P;    // number of covars in longitudinal model
int<lower=1,upper=N> subj[Ntot]; // subject ID
  vector[Ntot] Y;      // longitudinal outcome
int Q; // num covars in survival model
vector[N] surt; // survival times
matrix[N,K] D; // failure indicators
matrix[Ntot,P] XL; // covariates
matrix[N,Q] XS;
vector[P] mbeta;  // Hyper priors
real<lower=0> sbeta;
vector[K] mtheta;
real<lower=0> stheta;
//vector[K] theta;
real<lower=0> au;
real<lower=0> bu;
real<lower=0> aeps;
```

```stan
real<lower=0> beps;

real<lower=0> su;

matrix[K,Q] malpha;

real<lower=0> salpha;

real<lower=0> agamma;

real<lower=0> bgamma;

real<lower=0> adelta;

real<lower=0> bdelta;

}

parameters{

vector[P] beta;  // longitudinal regression coeffs

real<lower=0> sigmau;  // std dev of random intercept

real<lower=0> sigmaeps;  // std dev of longitudinal error

vector[N] u0;  // subject-specific random intercept

matrix[K,Q] alpha;

vector[K] theta;

vector[K] gamma;

real<lower=0, upper=1> delta;

}

model{

real mul;

matrix[N,K] chik;



// Priors

for(i in 1:P){

beta[i] ~ normal(mbeta[i],sbeta);
```

```
}

sigmau ~ gamma(au,bu);

sigmaeps ~ gamma(aeps,beps);

u0 ~ normal(0,sigmau);


for(i in 1:K){

for(j in 1:Q){

alpha[i,j] ~ normal(malpha[i,j],salpha);

}

theta[i] ~ normal(mtheta[i],stheta);

gamma[i] ~ gamma(agamma,bgamma);

}


delta ~ beta(adelta,bdelta);


// Longutudinal Model

for(i in 1:Ntot){

mul = XL[i]*beta + u0[subj[i]];

Y[i] ~ normal(mul,sigmaeps);

}


// Survival Model

for(i in 1:N){

chik[i,1] = exp(XS[i,1]*alpha[1,1] +

XS[i,2]*alpha[1,2] + theta[1]*u0[i]);

chik[i,2] = exp(XS[i,1]*alpha[2,1] +

XS[i,2]*alpha[2,2] + theta[2]*u0[i]);
```

```
}

surt ~ mysurv(D, chik, gamma, delta);



}
```

E.3.2   For Bayes Factors

Code for Model with $0 < \delta < 1$:

```
functions{

real mysurv_lpdf(vector t, matrix d, matrix chi, vector gam,

real delta){

vector[num_elements(t)] probs;



real lprob;



for(i in 1:num_elements(t)){

probs[i] = 0;

}

for(i in 1:num_elements(t)){          // num elements t = N

for(j in 1:num_elements(gam)){    // num elements gam = K



probs[i] = probs[i] +

d[i,j]*( (delta-1)*log( (chi[i,1]*(t[i]^gam[1]))^(1/delta) +

(chi[i,2]*(t[i]^gam[2]))^(1/delta) ) +

((1/delta)-1)*log( chi[i,j]*(t[i]^gam[j]) ) +

log(chi[i,j]) + log(gam[j]) + (gam[j]-1)*log(t[i])

) -
```

```
( (chi[i,1]*(t[i]^gam[1]))^(1/delta) +

(chi[i,2]*(t[i]^gam[2]))^(1/delta) )^delta

;


}
}
lprob = sum(probs);

return lprob;

}
}
data{

int K; // Number of competing risks

//(not including independent censoring)

int Ntot; // length of vector of all

//longitudinal outcome observations

int N;   // number of subjects

int P;   // number of covars in longitudinal model

int<lower=1,upper=N> subj[Ntot]; // subject ID

  vector[Ntot] Y;     // longitudinal outcome

int Q; // num covars in survival model

vector[N] surt; // survival times

matrix[N,K] D; // failure indicators

matrix[Ntot,P] XL; // covariates

matrix[N,Q] XS;

vector[P] mbeta;  // Hyper priors

real<lower=0> sbeta;

vector[K] mtheta;
```

```
real<lower=0> stheta;

//vector[K] theta;

real<lower=0> au;

real<lower=0> bu;

real<lower=0> aeps;

real<lower=0> beps;

real<lower=0> su;

matrix[K,Q] malpha;

real<lower=0> salpha;

real<lower=0> agamma;

real<lower=0> bgamma;

real<lower=0> adelta;

real<lower=0> bdelta;

}

parameters{

vector[P] beta;  // longitudinal regression coeffs

real<lower=0> sigmau;  // std dev of random intercept

real<lower=0> sigmaeps;  // std dev of longitudinal error

vector[N] u0;  // subject-specific random intercept

matrix[K,Q] alpha;

vector[K] theta;

vector[K] gamma;

real<lower=0, upper=1> delta;

}

model{

real mul;

matrix[N,K] chik;
```

```
// Priors

for(i in 1:P){

target += normal_lpdf(beta[i] | mbeta[i],sbeta);

}

target += gamma_lpdf(sigmau | au,bu);

target += gamma_lpdf(sigmaeps | aeps,beps);

target += normal_lpdf(u0 | 0,sigmau);


for(i in 1:K){

for(j in 1:Q){

target += normal_lpdf(alpha[i,j] | malpha[i,j],salpha);

}

target += normal_lpdf(theta[i] | mtheta[i],stheta);

target += gamma_lpdf(gamma[i] | agamma,bgamma);

}


target += beta_lpdf(delta | adelta,bdelta);


// Longutudinal Model

for(i in 1:Ntot){

mul = XL[i]*beta + u0[subj[i]];

target += normal_lpdf(Y[i] | mul,sigmaeps);

}


// Survival Model
```

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

chik[i,1] = exp(XS[i,1]*alpha[1,1] +

XS[i,2]*alpha[1,2] + theta[1]*u0[i]);

chik[i,2] = exp(XS[i,1]*alpha[2,1] +

XS[i,2]*alpha[2,2] + theta[2]*u0[i]);

}

target += mysurv_lpdf(surt | D, chik, gamma, delta);


}
```

Code for Model with $\delta = 1$:

```
functions{

real mysurv_lpdf(vector t, matrix d, matrix chi, vector gam){

vector[num_elements(t)] probs;


real lprob;


for(i in 1:num_elements(t)){

probs[i] = 0;

}

for(i in 1:num_elements(t)){          // num elements t = N

for(j in 1:num_elements(gam)){    // num elements gam = K


probs[i] = probs[i] +

d[i,j]*( log(chi[i,j]) + log(gam[j]) +

(gam[j]-1)*log(t[i])) -

 ( (chi[i,1]*(t[i]^gam[1])) +

 (chi[i,2]*(t[i]^gam[2])) )
```

```
;


}
}
lprob = sum(probs);
return lprob;
}
}
data{
int K; // Number of competing risks
//(not including independent censoring)
int Ntot; // length of vector of all
//longitudinal outcome observations
int N;   // number of subjects
int P;   // number of covars in longitudinal model
int<lower=1,upper=N> subj[Ntot]; // subject ID
  vector[Ntot] Y;     // longitudinal outcome
int Q; // num covars in survival model
vector[N] surt; // survival times
matrix[N,K] D; // failure indicators
matrix[Ntot,P] XL; // covariates
matrix[N,Q] XS;
vector[P] mbeta;  // Hyper priors
real<lower=0> sbeta;
vector[K] mtheta;
real<lower=0> stheta;
//vector[K] theta;
```

```
real<lower=0> au;

real<lower=0> bu;

real<lower=0> aeps;

real<lower=0> beps;

real<lower=0> su;

matrix[K,Q] malpha;

real<lower=0> salpha;

real<lower=0> agamma;

real<lower=0> bgamma;

}

parameters{

vector[P] beta;  // longitudinal regression coeffs

real<lower=0> sigmau;  // std dev of random intercept

real<lower=0> sigmaeps;  // std dev of longitudinal error

vector[N] u0;  // subject-specific random intercept

matrix[K,Q] alpha;

vector[K] theta;

vector[K] gamma;

}

model{

real mul;

matrix[N,K] chik;



// Priors

for(i in 1:P){

target += normal_lpdf(beta[i] | mbeta[i],sbeta);
```

```
}

target += gamma_lpdf(sigmau | au,bu);

target += gamma_lpdf(sigmaeps | aeps,beps);

target += normal_lpdf(u0 | 0,sigmau);


for(i in 1:K){

for(j in 1:Q){

target += normal_lpdf(alpha[i,j] | malpha[i,j],salpha);

}

target += normal_lpdf(theta[i] | mtheta[i],stheta);

target += gamma_lpdf(gamma[i] | agamma,bgamma);

}



// Longutudinal Model

for(i in 1:Ntot){

mul = XL[i]*beta + u0[subj[i]];

target += normal_lpdf(Y[i] | mul,sigmaeps);

}


// Survival Model

for(i in 1:N){

chik[i,1] = exp(XS[i,1]*alpha[1,1] +

XS[i,2]*alpha[1,2] + theta[1]*u0[i]);

chik[i,2] = exp(XS[i,1]*alpha[2,1] +

XS[i,2]*alpha[2,2] + theta[2]*u0[i]);

}
```

```
target += mysurv_lpdf(surt | D, chik, gamma);


}
```

E.4 Stan Code for Joint Models with Multiple Longitudinal Outcomes and Multi-state Data in Chapter 5

```
functions{

real myintegrand(real x, real delt, real m1, real g1,

real m2, real g2){

// integrate over x


real intval;


intval = (1/delt)*(

(m1^(1/delt))*g1*( x^( (g1/delt)-1 ) )*(

(1-delt)*( ( ( m1*( x^g1 ) )^(1/delt) )^(-1) ) +

delt*( ( ( m1*( x^g1 ) )^(1/delt) )^(delt-1) )

)

);

return intval;

}

real mysurv_lpdf(vector t, int Nsurv, vector D, matrix mu,

real[] gam, real delta, int[] lastobs){

vector[Nsurv] probs;


real lprob;
```

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

probs[i] = 0;

}

for(i in 2:Nsurv){

probs[i] = probs[i] +

(lastobs[i-1] != 1)*(

(D[i]==1)*(D[i-1]==2)*log(

(1/delta)*(mu[i,1]^(1/delta))*gam[1]*(

t[i]^( (gam[1]/delta)-1 ) )*

( (1-delta)*(

( (mu[i,1]*(t[i]^(gam[1]) ))^(1/delta) +

(mu[i,2]*(t[i-1]^(gam[2]) ))^(1/delta) )^(-1)

) + delta*(((mu[i,1]*(t[i]^(gam[1]) ))^(1/delta) +

(mu[i,2]*(t[i-1]^(gam[2]) ))^(1/delta))^(delta-1) )

)

)+

(D[i]==1)*(D[i-1]==3)*log(

(1/delta)*(mu[i,1]^(1/delta))*gam[1]*(

t[i]^( (gam[1]/delta)-1 ) )*

( (1-delta)*(

( (mu[i,1]*(t[i]^(gam[1]) ))^(1/delta) +

(mu[i,3]*(t[i-1]^(gam[3]) ))^(1/delta) )^(-1)

) + delta*(((mu[i,1]*(t[i]^(gam[1]) ))^(1/delta) +

(mu[i,3]*(t[i-1]^(gam[3]) ))^(1/delta))^(delta-1) )

)

)+

(D[i]==2)*(D[i-1]==1)*log(
```

```
(1/delta)*(mu[i,2]^(1/delta))*gam[2]*(

t[i]^( (gam[2]/delta)-1 ) )*

( (1-delta)*(

( (mu[i,2]*(t[i]^(gam[2]) ))^(1/delta) +

(mu[i,1]*(t[i-1]^(gam[1]) ))^(1/delta) )^(-1)

) + delta*(((mu[i,2]*(t[i]^(gam[2]) ))^(1/delta) +

(mu[i,1]*(t[i-1]^(gam[1]) ))^(1/delta))^(delta-1) )

)

)+

(D[i]==2)*(D[i-1]==3)*log(

(1/delta)*(mu[i,2]^(1/delta))*gam[2]*(

t[i]^( (gam[2]/delta)-1 ) )*

( (1-delta)*(

( (mu[i,2]*(t[i]^(gam[2]) ))^(1/delta) +

(mu[i,3]*(t[i-1]^(gam[3]) ))^(1/delta) )^(-1)

) + delta*(((mu[i,2]*(t[i]^(gam[2]) ))^(1/delta) +

(mu[i,3]*(t[i-1]^(gam[3]) ))^(1/delta))^(delta-1) )

)

)+

(D[i]==3)*(D[i-1]==1)*log(

(1/delta)*(mu[i,3]^(1/delta))*gam[3]*(

t[i]^( (gam[3]/delta)-1 ) )*

( (1-delta)*(

( (mu[i,3]*(t[i]^(gam[3]) ))^(1/delta) +

(mu[i,1]*(t[i-1]^(gam[1]) ))^(1/delta) )^(-1)

) + delta*(((mu[i,3]*(t[i]^(gam[3]) ))^(1/delta) +

(mu[i,1]*(t[i-1]^(gam[1]) ))^(1/delta))^(delta-1) )
```

```
)

)+

(D[i]==3)*(D[i-1]==2)*log(

(1/delta)*(mu[i,3]^(1/delta))*gam[3]*(

t[i]^( (gam[3]/delta)-1 ) )*

( (1-delta)*(

( (mu[i,3]*(t[i]^(gam[3]) ))^(1/delta) +

(mu[i,2]*(t[i-1]^(gam[2]) ))^(1/delta) )^(-1)

) + delta*(((mu[i,3]*(t[i]^(gam[3]) ))^(1/delta) +

(mu[i,2]*(t[i-1]^(gam[2]) ))^(1/delta))^(delta-1) )

)

)+

(D[i]==4)*(D[i-1]==1)*log(

(1/delta)*(mu[i,4]^(1/delta))*gam[4]*(

t[i]^( (gam[4]/delta)-1 ) )*

( (1-delta)*(

( (mu[i,4]*(t[i]^(gam[4]) ))^(1/delta) +

(mu[i,1]*(t[i-1]^(gam[1]) ))^(1/delta) )^(-1)

) + delta*(((mu[i,4]*(t[i]^(gam[4]) ))^(1/delta) +

(mu[i,1]*(t[i-1]^(gam[1]) ))^(1/delta))^(delta-1) )

)

)+

(D[i]==4)*(D[i-1]==2)*log(

(1/delta)*(mu[i,4]^(1/delta))*gam[4]*(

t[i]^( (gam[4]/delta)-1 ) )*

( (1-delta)*(

( (mu[i,4]*(t[i]^(gam[4]) ))^(1/delta) +
```

```
(mu[i,2]*(t[i-1]^(gam[2]) ))^(1/delta) )^(-1)
) + delta*(((mu[i,4]*(t[i]^(gam[4]) ))^(1/delta) +
(mu[i,2]*(t[i-1]^(gam[2]) ))^(1/delta))^(delta-1) )
)
)+
(D[i]==4)*(D[i-1]==3)*log(
(1/delta)*(mu[i,4]^(1/delta))*gam[4]*(
t[i]^( (gam[4]/delta)-1 ) )*
( (1-delta)*(
( (mu[i,4]*(t[i]^(gam[4]) ))^(1/delta) +
(mu[i,3]*(t[i-1]^(gam[3]) ))^(1/delta) )^(-1)
) + delta*(((mu[i,4]*(t[i]^(gam[4]) ))^(1/delta) +
(mu[i,3]*(t[i-1]^(gam[3]) ))^(1/delta))^(delta-1) )
)
)
-
(t[i]-t[i-1])*(
(D[i-1] == 1)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,2],
gam[2], mu[i,1], gam[1])+
(D[i-1] == 1)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,3],
gam[3], mu[i,1], gam[1])+
(D[i-1] == 1)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,4],
gam[4], mu[i,1], gam[1])+
(D[i-1] == 2)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,1],
gam[1], mu[i,2], gam[2])+
(D[i-1] == 2)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,3],
gam[3], mu[i,2], gam[2])+
```

```
(D[i-1] == 2)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,4],
gam[4], mu[i,2], gam[2])+
(D[i-1] == 3)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,1],
gam[1], mu[i,3], gam[3])+
(D[i-1] == 3)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,2],
gam[2], mu[i,3], gam[3])+
(D[i-1] == 3)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,4],
gam[4], mu[i,3], gam[3])+
(D[i-1] == 4)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,1],
gam[1], mu[i,4], gam[4])+
(D[i-1] == 4)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,2],
gam[2], mu[i,4], gam[4])+
(D[i-1] == 4)*myintegrand((t[i-1]+t[i])/2, delta, mu[i,3],
gam[3], mu[i,4], gam[4])+
(D[i] == 5)*(D[i-1] == 1)*myintegrand((t[i-1]+t[i])/2, delta,
mu[i,2],gam[2], mu[i,1], gam[1])+
(D[i] == 5)*(D[i-1] == 1)*myintegrand((t[i-1]+t[i])/2, delta,
mu[i,3],gam[3], mu[i,1], gam[1])+
(D[i] == 5)*(D[i-1] == 1)*myintegrand((t[i-1]+t[i])/2, delta,
mu[i,4], gam[4], mu[i,1], gam[1])+
(D[i] == 5)*(D[i-1] == 2)*myintegrand((t[i-1]+t[i])/2, delta,
mu[i,1], gam[1], mu[i,2], gam[2])+
(D[i] == 5)*(D[i-1] == 2)*myintegrand((t[i-1]+t[i])/2, delta,
mu[i,3], gam[3], mu[i,2], gam[2])+
(D[i] == 5)*(D[i-1] == 2)*myintegrand((t[i-1]+t[i])/2, delta,
mu[i,4], gam[4], mu[i,2], gam[2])+
(D[i] == 5)*(D[i-1] == 3)*myintegrand((t[i-1]+t[i])/2, delta,
```

```
mu[i,2], gam[2], mu[i,3], gam[3])+

(D[i] == 5)*(D[i-1] == 3)*myintegrand((t[i-1]+t[i])/2, delta,

mu[i,1], gam[1], mu[i,3], gam[3])+

(D[i] == 5)*(D[i-1] == 3)*myintegrand((t[i-1]+t[i])/2, delta,

mu[i,4], gam[4], mu[i,3], gam[3])

)

)

;

}

lprob = sum(probs);

return lprob;

}

}

data{

int K; // Number of possible states

//(not including independent censoring)

int Ntot1; // length of vector of

// all longitudinal outcome observations

int Ntot2;

int N;   // number of subjects

int Nsurv; // length of survival vectors

int P1;   // number of covars in longitudinal model

int P2;

int<lower=1> subj1[Ntot1]; // subject ID,

//each repeated for the number of that subject's

// longitudinal measurements

int<lower=1> subj2[Ntot2];
```

```
int<lower=1> subj3[Nsurv]; // subjid repeated
//number of transition times
   vector[Ntot1] Y1;      // longitudinal outcome
vector[Ntot2] Y2;
int Q; // num covars in survival model
vector[Nsurv] surt; // survival times
vector[Nsurv] D; // failure indicators
matrix[Ntot1,P1] XL1; // covariates
matrix[Ntot2,P2] XL2;
matrix[Nsurv,Q] XS;
int<lower=0,upper=1> lastobs[Nsurv];
vector[P1] mbeta1; // Hyper priors
vector[P2] mbeta2;
real<lower=0> sbeta;
vector[K] mtheta;
real<lower=0> stheta;
real<lower=0> aeps;
real<lower=0> beps;
matrix[K,Q] malpha;
real<lower=0> salpha;
real<lower=0> agamma;
real<lower=0> bgamma;
real<lower=0> adelta;
real<lower=0> bdelta;
real<lower=0> nu;
matrix[2,2] hypSigma;
vector[2] muzero;
```

```
}

parameters{

vector[P1] beta1;  // longitudinal regression coeffs

vector[P2] beta2;

real<lower=0> sigmaeps1;  // std dev of longitudinal error

real<lower=0> sigmaeps2;

matrix[N,2] u0;  // subject-specific random intercept

matrix[K,Q] alpha;

vector[K] theta1;

vector[K] theta2;

real<lower=0> gamma[K];

real<lower=0, upper=1> delta;

cov_matrix[2] Sigma;

}

transformed parameters{

real<lower=0> sigmau1;

real<lower=0> sigmau2;

real rho;

sigmau1 = Sigma[1,1]^(0.5);

sigmau2 = Sigma[2,2]^(0.5);

rho = Sigma[1,2]/((Sigma[1,1]^(0.5))*(Sigma[2,2]^(0.5)));

}

model{

real mul1;

real mul2;

matrix[Nsurv,K] muk;
```

```
// Priors
for(i in 1:P1){
beta1[i] ~ normal(mbeta1[i],sbeta);
}
for(i in 1:P2){
beta2[i] ~ normal(mbeta2[i],sbeta);
}
sigmaeps1 ~ gamma(aeps,beps);
sigmaeps2 ~ gamma(aeps,beps);
Sigma ~ wishart(nu,hypSigma);


for(i in 1:N){
u0[i,] ~ multi_normal(muzero,Sigma);
}
for(i in 1:K){
for(j in 1:Q){
alpha[i,j] ~ normal(malpha[i,j],salpha);
}
theta1[i] ~ normal(mtheta[i],stheta);
theta2[i] ~ normal(mtheta[i],stheta);
gamma[i] ~ gamma(agamma,bgamma);
}
delta ~ beta(adelta,bdelta);


// Longutudinal Model
for(i in 1:Ntot1){
mul1 = XL1[i]*beta1 + u0[subj1[i],1];
```

```
Y1[i] ~ normal(mul1,sigmaeps1);

}

for(i in 1:Ntot2){

mul2 = XL2[i]*beta2 + u0[subj2[i],2];

Y2[i] ~ normal(mul2,sigmaeps2);

}


// Survival Model

for(i in 1:Nsurv){

muk[i,1] = exp(XS[i,1]*alpha[1,1] + XS[i,2]*alpha[1,2] +

theta1[1]*u0[subj3[i],1] + theta2[1]*u0[subj3[i],2]);

muk[i,2] = exp(XS[i,1]*alpha[2,1] + XS[i,2]*alpha[2,2] +

theta1[2]*u0[subj3[i],1] + theta2[2]*u0[subj3[i],2]);

muk[i,3] = exp(XS[i,1]*alpha[3,1] + XS[i,2]*alpha[3,2] +

theta1[3]*u0[subj3[i],1] + theta2[3]*u0[subj3[i],2]);

muk[i,4] = exp(XS[i,1]*alpha[4,1] + XS[i,2]*alpha[4,2] +

theta1[4]*u0[subj3[i],1] + theta2[4]*u0[subj3[i],2]);

}

surt ~ mysurv(Nsurv,D,muk,gamma,delta,lastobs);

}
```

Appendix F

Additional Data Analysis Results

F.1 Data Analysis Trace Plots for Joint Model with Conditionally Independent Competing Risks in Chapter 3
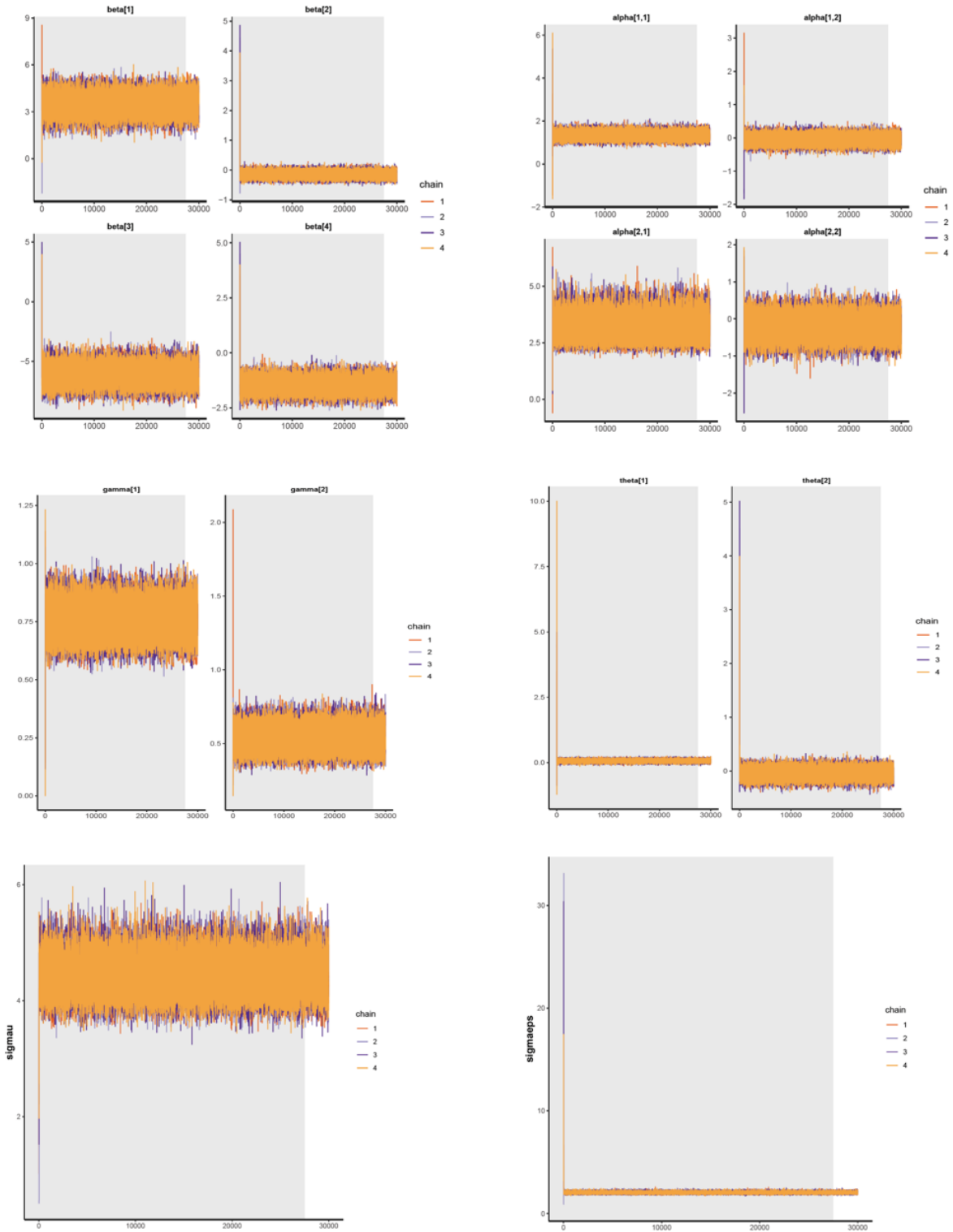
Figure F.1: Trace plots for Model W from ACC data.
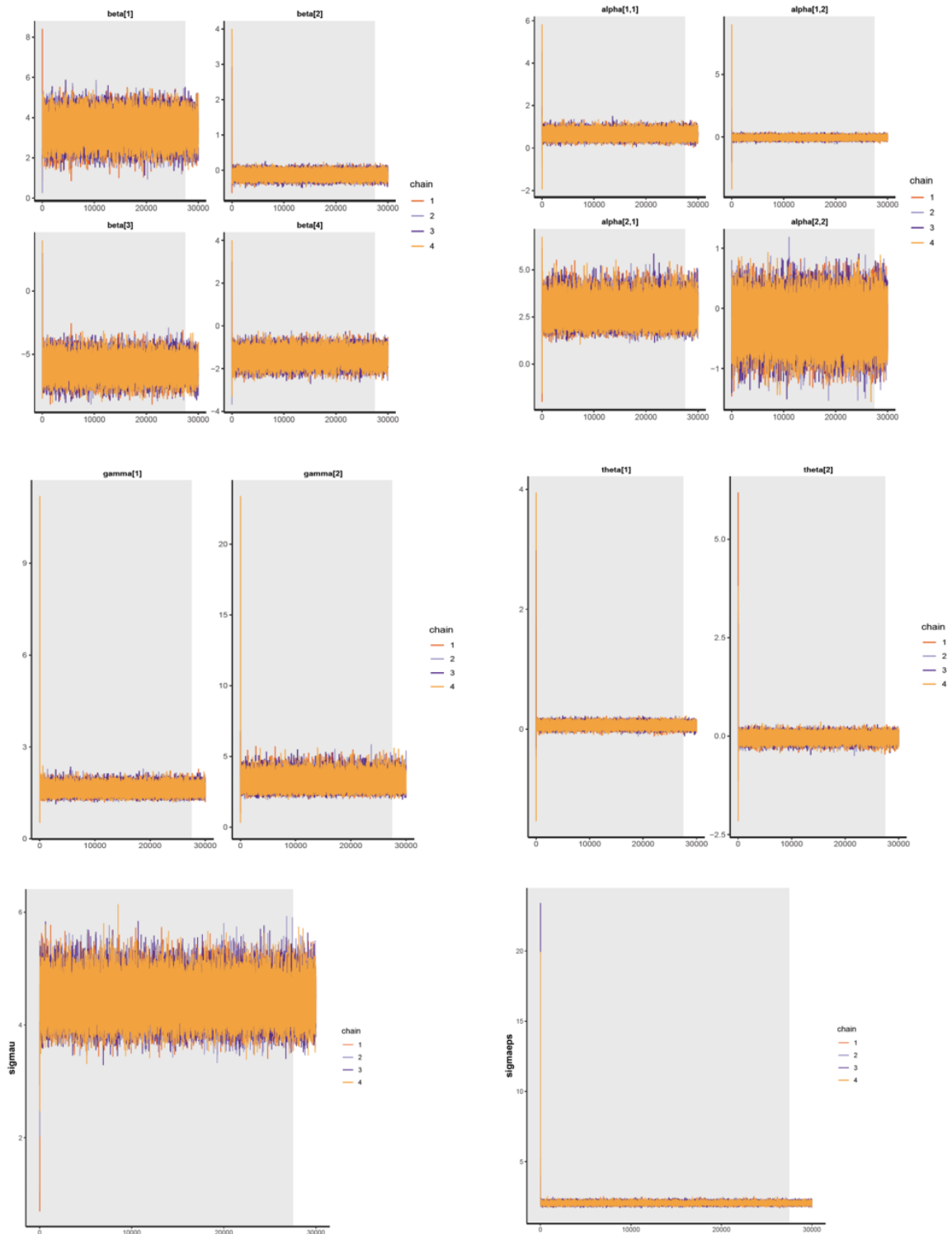
Figure F.2: Trace plots for Model L from ACC data.

F.2 Chapter 4 Chain Initial Values for Data Analysis

Parameter	Chain 1	Chain 2	Chain 3	Chain 4
β_0	3.5	2.5	1.5	2
β_1	0.15	0.55	-0.15	0.05
β_2	2	1	3	0.5
β_3	3	2	4	1.5
σ_U	0.5	1	2	1.5
σ_ϵ	2	1	3	0.5
$\alpha_{0,1}$	1.2	2.2	0.2	0.6
$\alpha_{1,1}$	-0.4	0.4	-1.4	-0.2
$\alpha_{0,2}$	1.1	2.1	0.1	0.5
$\alpha_{1,2}$	-0.5	0.5	-1.5	-0.1
θ_1	-1.1	1.1	-2.1	-0.1
θ_2	-0.9	0.9	-1.9	-0.5
γ_1	5	6	4	5.5
γ_2	8	7	6	9
δ	0.4	0.1	0.7	0.6
U_{0i}	1	-1	2	-2

Table F.1: Initial values for four chains used to fit Model W to the ACC data.

F.3 Data Analysis Trace Plots for Joint Longitudinal and Dependent Competing Risks in Chapter 4
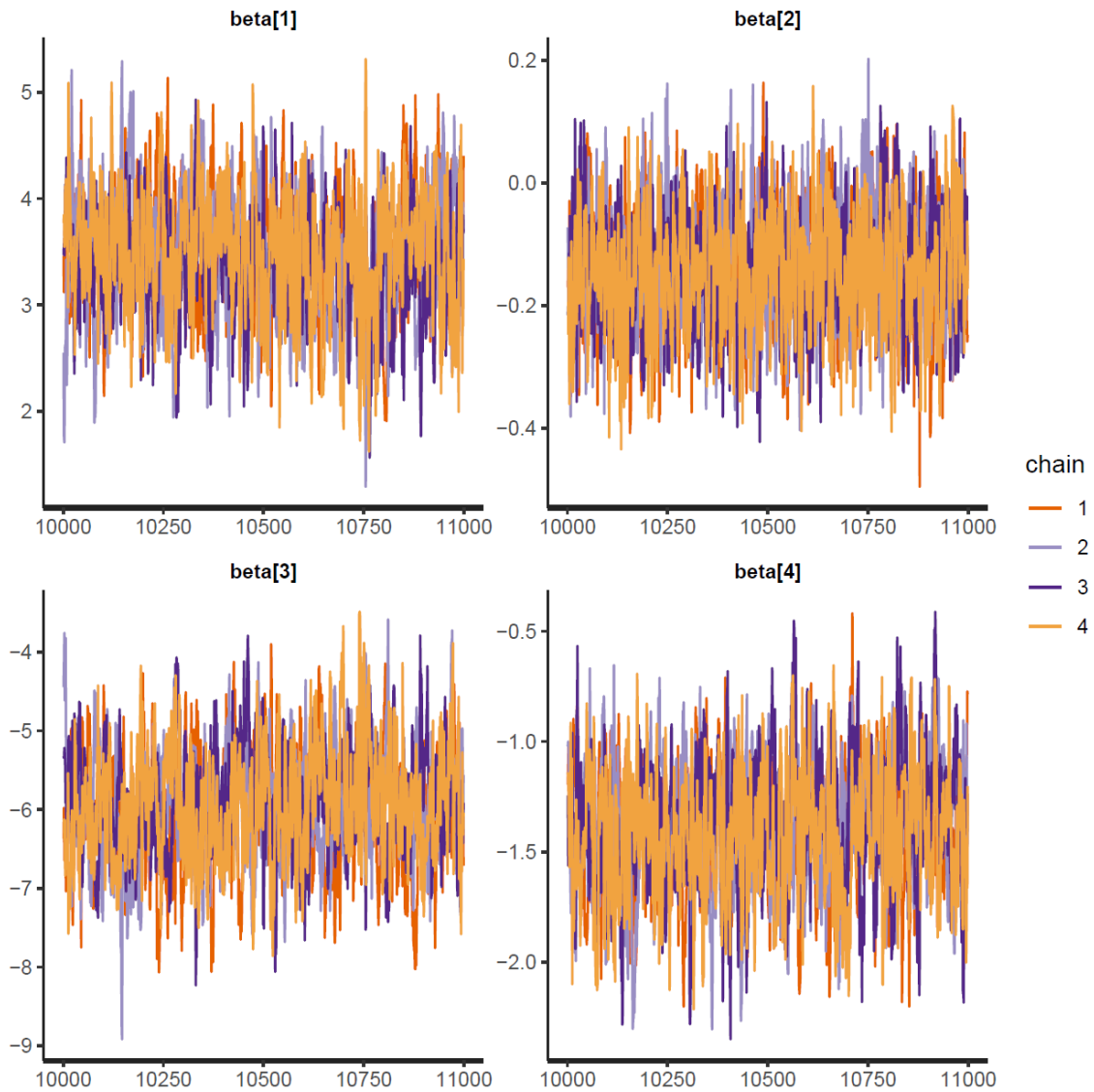
Figure F.3: Trace plots for beta parameters.
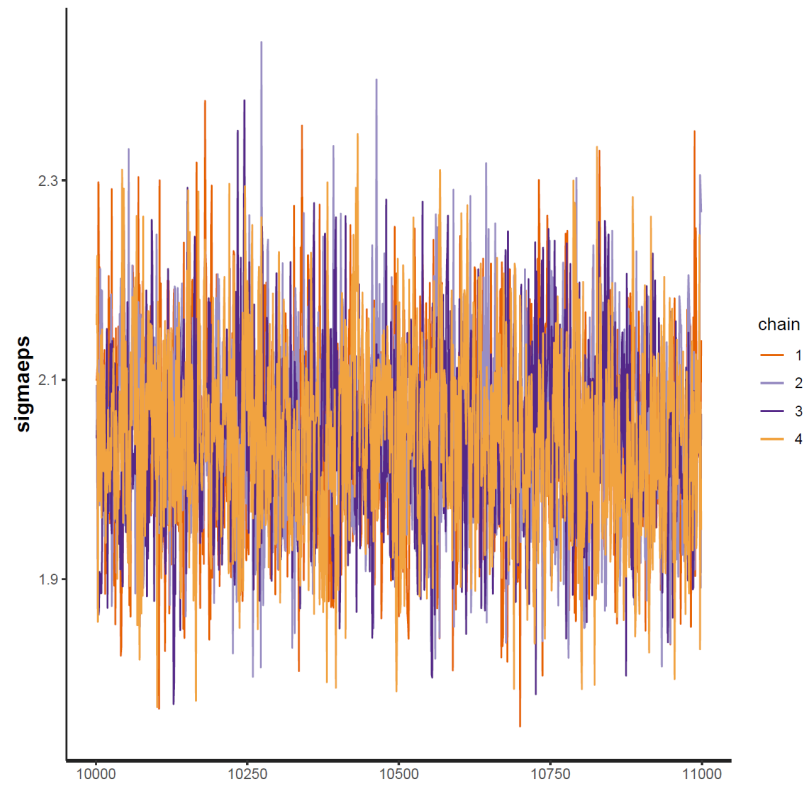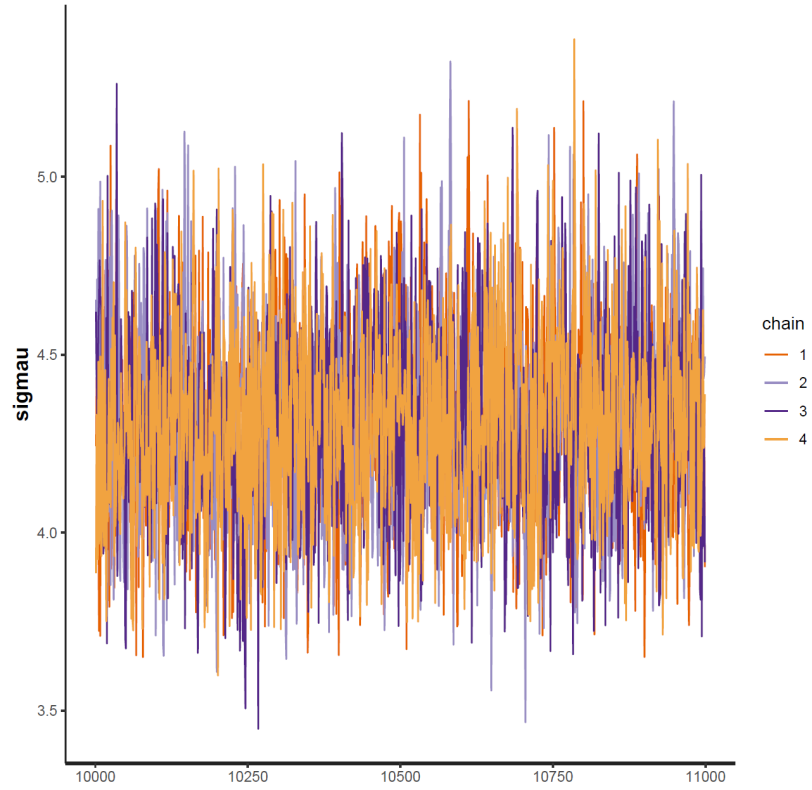
Figure F.4: Trace plots for standard deviation parameters.
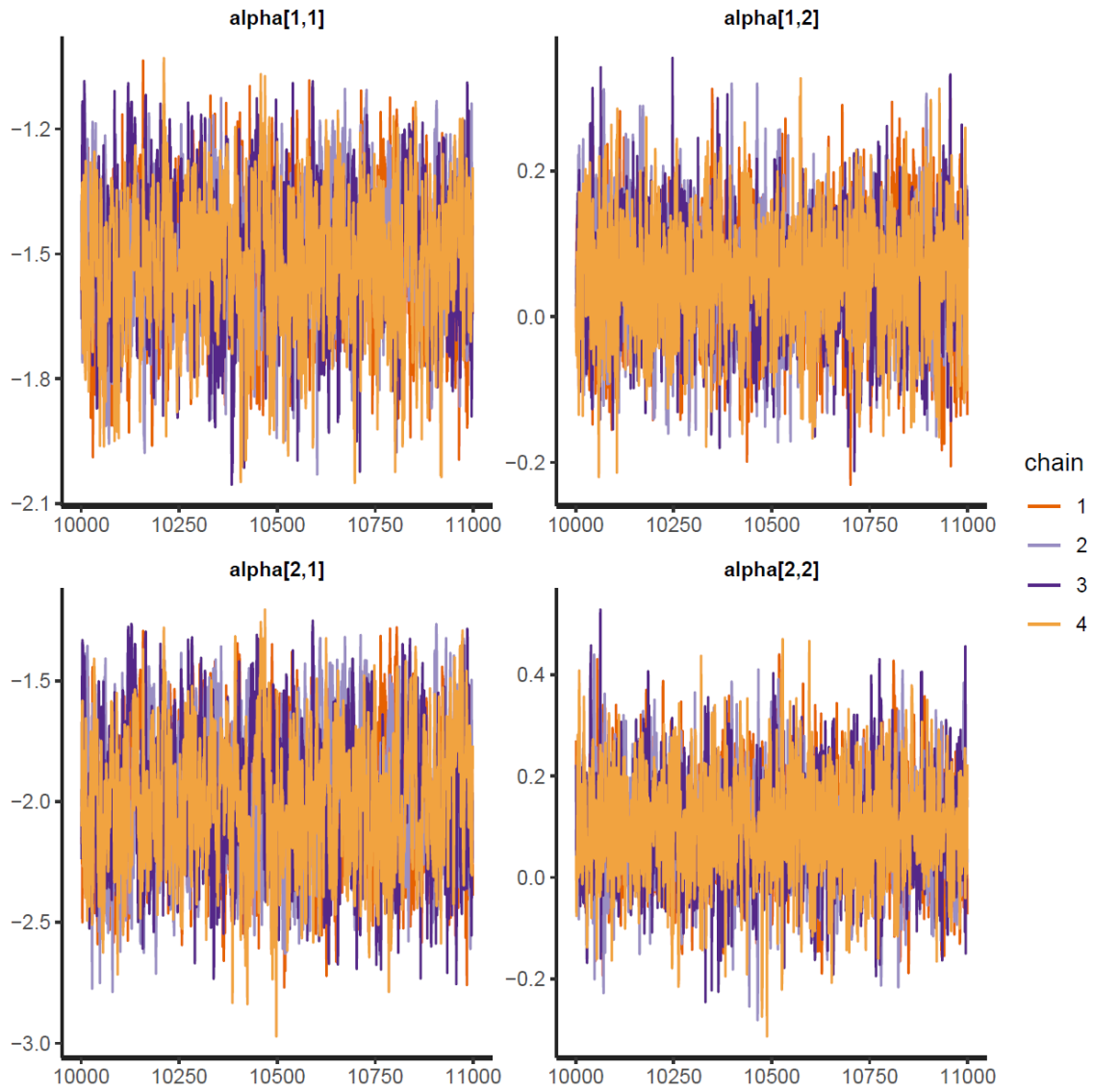
Figure F.5: Trace plots for alpha parameters.
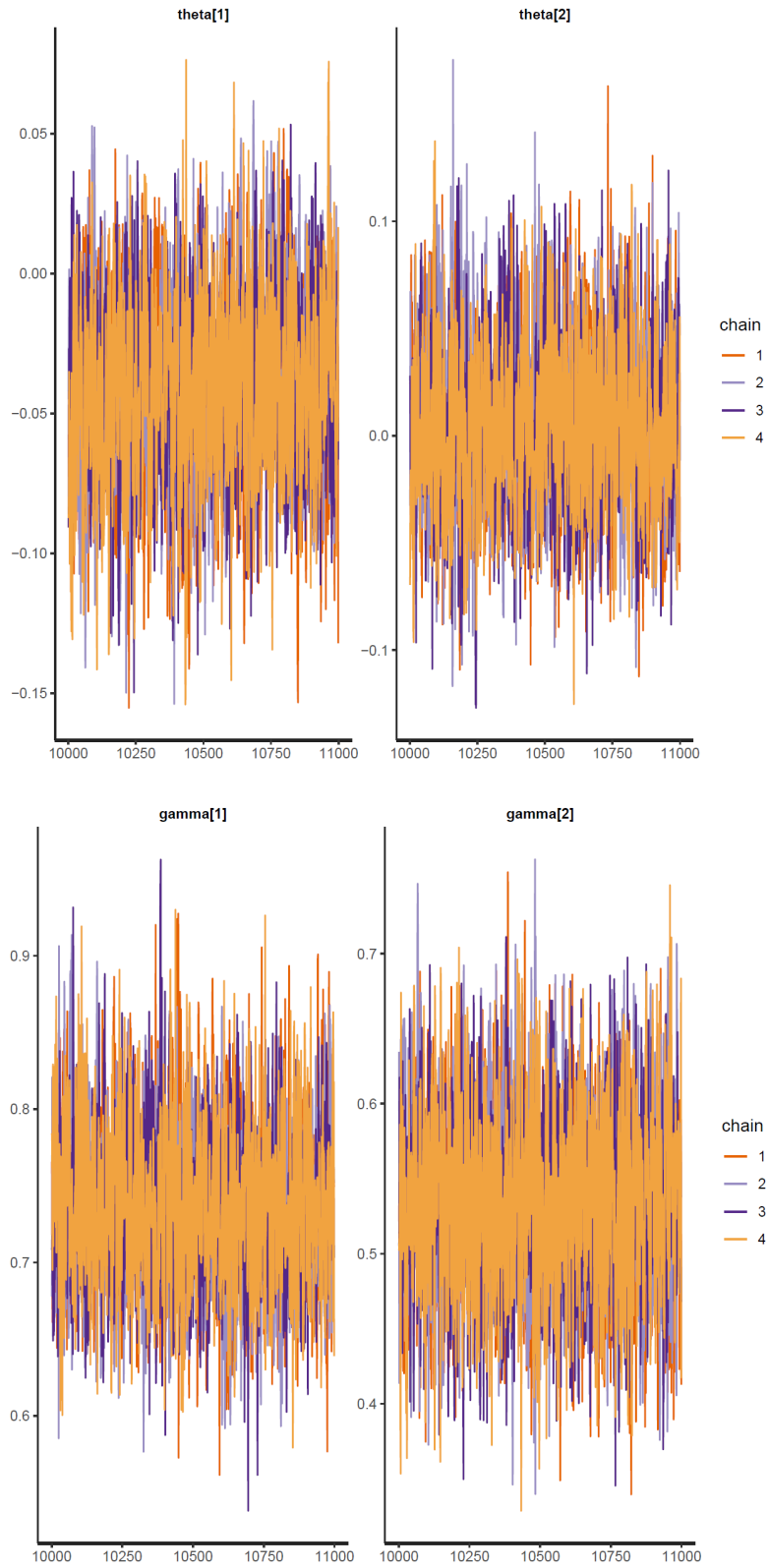
Figure F.6: Trace plots for theta and gamma parameters.
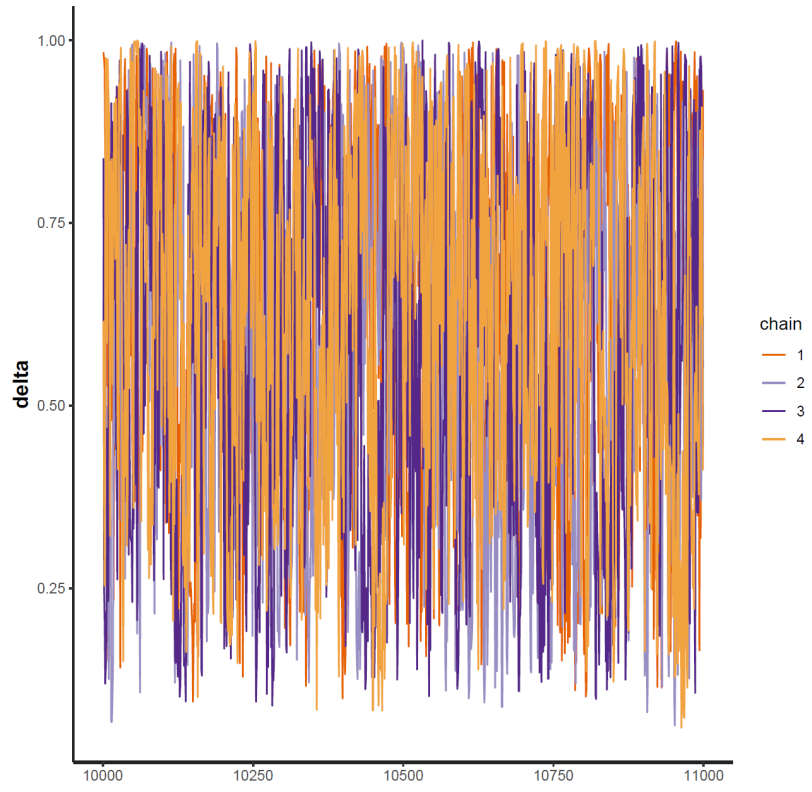
Figure F.7: Trace plot for delta parameter with Beta(1,1) prior.
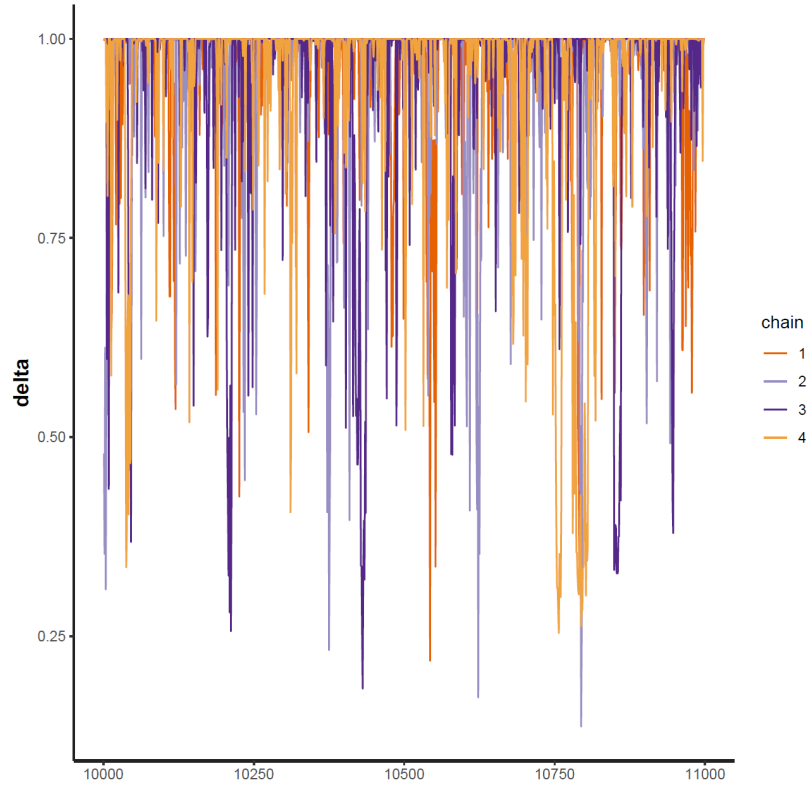
Figure F.8: Trace plot for delta parameter with Beta(0.03,0.07) prior.
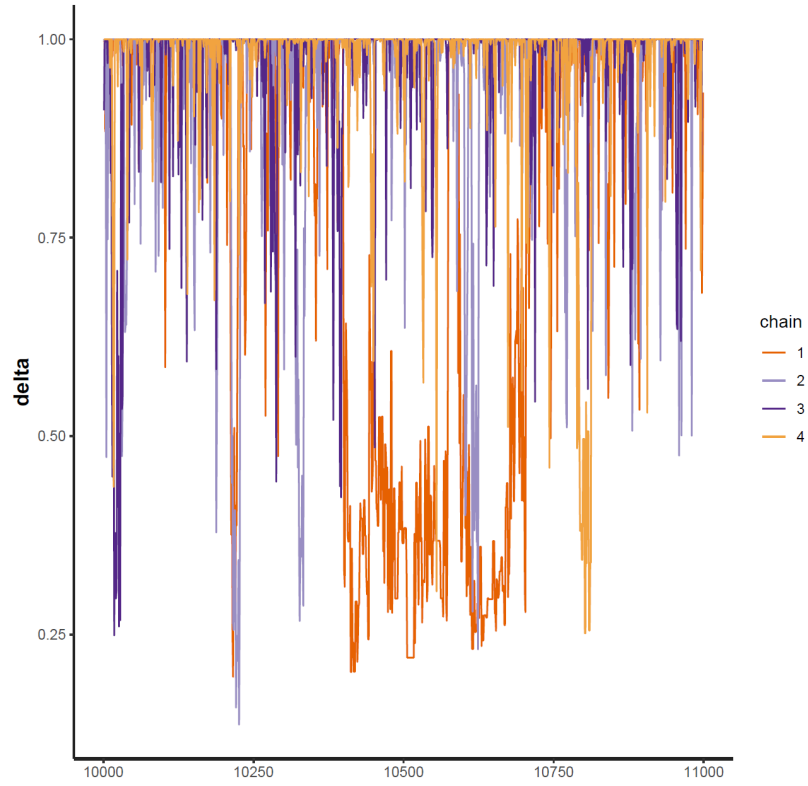
Figure F.9: Trace plot for delta parameter with Beta(0.05,0.05) prior.
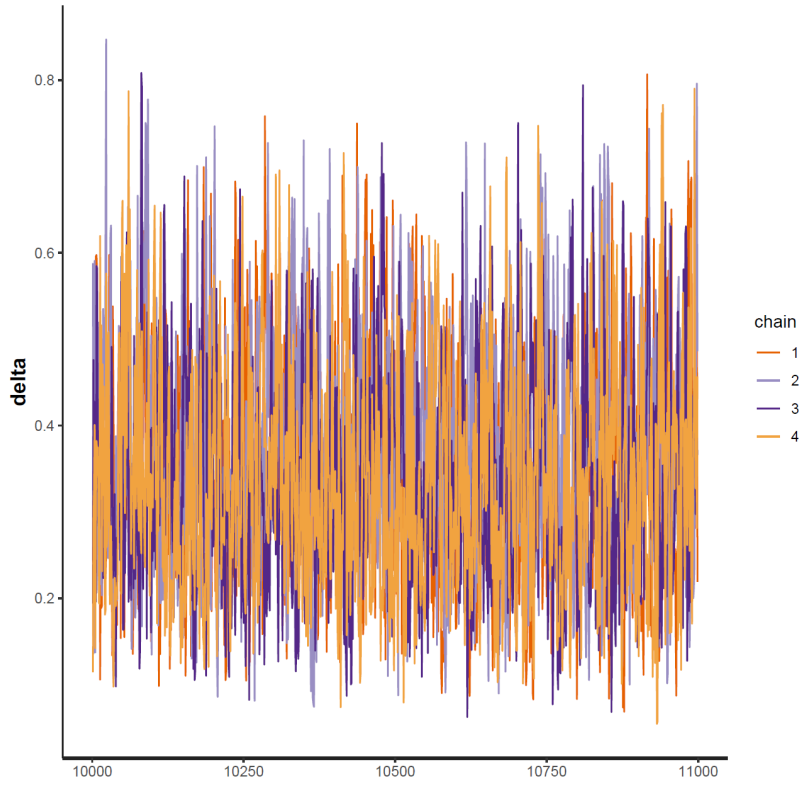
Figure F.10: Trace plot for delta parameter with Beta(3,7) prior.
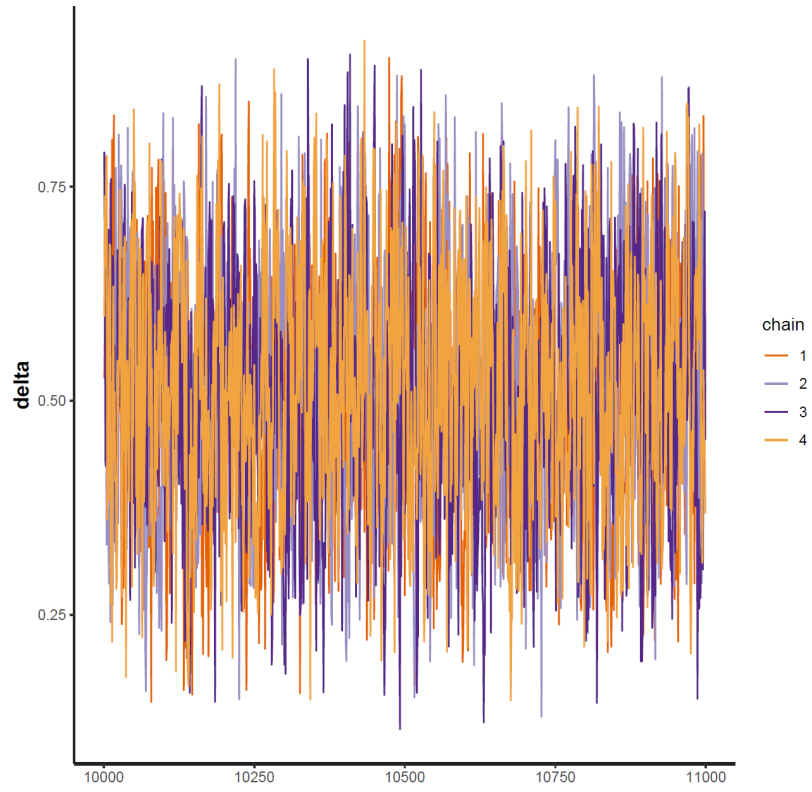
Figure F.11: Trace plot for delta parameter with Beta(5,5) prior.

F.4 Additional Information from Data Analysis with Multistate Model in Chapter 5

Parameter	N eff	Rhat	Parameter	N eff	Rhat	Parameter	N eff	Rhat
$\beta_0^{(1)}$	570	1.00	$\alpha_{0,1}$	1952	1.00	$\alpha_{0,3}$	2222	1.00
$\beta_1^{(1)}$	1683	1.00	$\alpha_{1,1}$	2341	1.00	$\alpha_{1,3}$	2705	1.00
$\beta_2^{(1)}$	460	1.01	$\theta_1^{(1)}$	3422	1.00	$\theta_3^{(1)}$	2619	1.00
$\beta_3^{(1)}$	553	1.00	$\theta_1^{(2)}$	3269	1.00	$\theta_3^{(2)}$	4403	1.00
$\beta_0^{(2)}$	367	1.00	γ_1	3081	1.00	γ_3	2660	1.00
$\beta_1^{(2)}$	1984	1.00	$\alpha_{0,2}$	1571	1.00	$\alpha_{0,4}$	1411	1.00
$\beta_2^{(2)}$	448	1.00	$\alpha_{1,2}$	1511	1.00	$\alpha_{1,4}$	1498	1.00
$\beta_3^{(2)}$	402	1.00	$\theta_2^{(1)}$	2827	1.00	$\theta_4^{(1)}$	2056	1.00
σ_1	1919	1.00	$\theta_2^{(2)}$	4013	1.00	$\theta_4^{(2)}$	3072	1.00
σ_2	2345	1.00	γ_2	3733	1.00	γ_4	2593	1.00
ρ	1993	1.00	$\sigma_\epsilon^{(1)}$	3445	1.00	δ	4289	1.00
			$\sigma_\epsilon^{(2)}$	3428	1.00			

Table F.2: Effective sample size (N eff) and Rhat values for the joint model fit to the ACC data.

Appendix G

Full Conditionals

G.1 Full Conditionals for Joint Longitudinal and Conditionally Independent Competing Risks in Chapter 3

$p(\beta|\cdot) \propto$

$$\sigma_\epsilon^{-\sum_{i=1}^N J_i} \exp\left\{ -\frac{1}{2\sigma_\epsilon^{2\sum_{i=1}^N J_i}} \sum_{i=1}^N (Y_i - X_i\beta - Z_iU_i)^T(Y_i - X_i\beta - Z_iU_i) \right\}$$

$$\times \sigma_\beta^{-p} \exp\left\{ -\frac{1}{2\sigma_\beta^{2p}}(\beta - m_\beta)^T(\beta - m_\beta) \right\}$$

$$= \sigma_\epsilon^{-\sum_{i=1}^N J_i} \sigma_\beta^{-p}.$$

$$\exp\left\{ -\frac{1}{2\sigma_\epsilon^{2\sum_{i=1}^N J_i}} \sum_{i=1}^N (Y_i - X_i\beta - Z_iU_i)^T(Y_i - X_i\beta - Z_iU_i) - \frac{1}{2\sigma_\beta^{2p}}(\beta - m_\beta)^T(\beta - m_\beta) \right\}$$

$$\sim N(A, B)$$

where

$$B = \left(\frac{1}{\sigma_\beta^{2p}} I_p + \frac{1}{\sigma_\epsilon^{2\sum_{i=1}^N J_i}} \sum_{i=1}^N X_i^T X_i \right)^{-1}$$

and

$$A = B^{-1}\left(\frac{1}{\sigma_\epsilon^{2\sum_{i=1}^N J_i}} \left(\sum_{i=1}^N U_i^T Z_i X_i - \sum_{i=1}^N Y_i^T X_i \right) - \frac{1}{\sigma_\beta^{2p}} m_\beta \right)$$

$$p(\sigma_\epsilon|\cdot) \propto$$

$$\sigma_\epsilon^{-\sum_{i=1}^N J_i} \exp\left\{-\frac{1}{2\sigma_\epsilon^{2J_i}}\sum_{i=1}^N (Y_i - X_i\beta - Z_iU_i)^T(Y_i - X_i\beta - Z_iU_i)\right\}$$

$$\times \frac{1}{\Gamma(a)b^a}\sigma_\epsilon^{a-1}\exp\left\{-\frac{\sigma_\epsilon}{b}\right\}$$

$$p(\sigma_U|\cdot) \propto$$

$$\sigma_U^{-1}\exp\left\{-\frac{1}{2\sigma_U^2}U_i^2\right\}\frac{1}{\Gamma(a)b^a}\sigma_U^{a-1}\exp\left\{-\frac{\sigma_U}{b}\right\}$$

G.1.1   Weibull model

$$p(\gamma_k|\cdot) \propto$$

$$\prod_{i=1}^N\left(\prod_{k=1}^K\left[\left(\gamma_k\left(\exp\{W_i\alpha_k + \theta_k^TU_i\}\right)^{-1}t^{\gamma_k-1}\right)^{D_{ki}}\exp\left\{-t^{\gamma_k}\left(\exp\{W_i\alpha_k + \theta_k^TU_i\}\right)^{-1}\right\}\right]\right)$$

$$\times \prod_{k=1}^K\left[b^{-a}\gamma_k^{a-1}\exp\left\{-\frac{\gamma_k}{b}\right\}\right]$$

$$p(\alpha_k|\cdot) \propto$$

$$\prod_{i=1}^N\left(\prod_{k=1}^K\left[\left(\gamma_k\left(\exp\{W_i\alpha_k + \theta_k^TU_i\}\right)^{-1}t^{\gamma_k-1}\right)^{D_{ki}}\exp\left\{-t^{\gamma_k}\left(\exp\{W_i\alpha_k + \theta_k^TU_i\}\right)^{-1}\right\}\right]\right)$$

$$\times \prod_{k=1}^K\left[s_\alpha^{-1}\exp\left\{-\frac{1}{2s_\alpha^{2q}}(\alpha_k - m_\alpha)^T(\alpha_k - m_\alpha)\right\}\right]$$

$$p(\theta_k|\cdot) \propto$$

$$\prod_{i=1}^{N} \left(\prod_{k=1}^{K} \left[\left(\gamma_k \left(\exp\{W_i \alpha_k + \theta_k^T U_i\} \right)^{-1} t^{\gamma_k - 1} \right)^{D_{ki}} \exp\left\{ -t^{\gamma_k} \left(\exp\{W_i \alpha_k + \theta_k^T U_i\} \right)^{-1} \right\} \right] \right)$$

$$\times \prod_{k=1}^{K} \left[s_\theta^{-1} \exp\left\{ -\frac{1}{2s_\theta^2} (\theta_k - m_\theta)^T (\theta_k - m_\theta) \right\} \right]$$

G.1.2 Log-Normal Model

$$p(\gamma_k|\cdot) \propto$$

$$\prod_{i=1}^{N} \left(\prod_{k=1}^{K} \left[(h_k(T_i|W_i, U_i, \gamma_k, \alpha_k, \theta_k))^{D_{ki}} \tau_k(T_i|W_i, U_i, \gamma_k, \alpha_k, \theta_k) \right] \right)$$

$$\times \prod_{k=1}^{K} \left[b^{-a} \gamma_k^{a-1} \exp\left\{ -\frac{\gamma_k}{b} \right\} \right]$$

$$p(\alpha_k|\cdot) \propto$$

$$\prod_{i=1}^{N} \left(\prod_{k=1}^{K} \left[(h_k(T_i|W_i, U_i, \gamma_k, \alpha_k, \theta_k))^{D_{ki}} \tau_k(T_i|W_i, U_i, \gamma_k, \alpha_k, \theta_k) \right] \right)$$

$$\times \prod_{k=1}^{K} \left[s_\alpha^{-1} \exp\left\{ -\frac{1}{2s_\alpha^{2q}} (\alpha_k - m_\alpha)^T (\alpha_k - m_\alpha) \right\} \right]$$

$$p(\theta_k|\cdot) \propto$$

$$\prod_{i=1}^{N} \left(\prod_{k=1}^{K} \left[(h_k(T_i|W_i, U_i, \gamma_k, \alpha_k, \theta_k))^{D_{ki}} \tau_k(T_i|W_i, U_i, \gamma_k, \alpha_k, \theta_k) \right] \right)$$

$$\times \prod_{k=1}^{K} \left[s_\theta^{-1} \exp\left\{ -\frac{1}{2s_\theta^2} (\theta_k - m_\theta)^T (\theta_k - m_\theta) \right\} \right]$$

Here the hazard and survival functions are:

$$h_k(t|U_i) = \frac{\left(\frac{\gamma_k}{2\pi}\right)^{1/2} x^{-1} \exp\left(-\frac{\gamma_k}{2}(\log x - \left[W_i\alpha_k + \theta_k^T U_i\right])^2\right)}{\int_t^\infty \left(\frac{\gamma_k}{2\pi}\right)^{1/2} x^{-1} \exp\left(-\frac{\gamma_k}{2}(\log x - \left[W_i\alpha_k + \theta_k^T U_i\right])^2\right)}$$

$$\tau_k(t|U_i) = \int_t^\infty \left(\frac{\gamma_k}{2\pi}\right)^{1/2} x^{-1} \exp\left(-\frac{\gamma_k}{2}(\log x - \left[W_i\alpha_k + \theta_k^T U_i\right])^2\right)$$

G.2 Full Conditionals for Joint Longitudinal and Dependent Competing Risks in Chapter 4

The posterior is proportional to the following.

$$p(\Omega \mid Y, T, D, X, W) \propto \prod_{i=1}^N (2\pi\sigma_\epsilon)^{-\frac{J_i}{2}} \exp\left\{-\frac{1}{2\sigma_\epsilon^{2J_i}}(Y_i - X_i\beta - U_i)^T (Y_i - X_i\beta - U_i)\right\} \cdot$$

$$s_\beta^{-p} \exp\left(-\frac{1}{2s_\beta^{2p}}(\beta - m_\beta)^T(\beta - m_\beta)\right) \sigma_U^{-1} \exp\left(-\frac{1}{2\sigma_U^2}U_i^2\right) \sigma_U^{a_U-1} \exp(-\sigma_U/b_U)\sigma_\epsilon^{a_\epsilon-1} \exp(-\sigma_\epsilon/b_\epsilon) \cdot$$

$$\prod_{k=1}^K \left(\gamma_k \exp\left(W_i\alpha_k - \theta_k^T U_i\right)^{\frac{1}{\delta}} T_i^{\frac{\gamma_k}{\delta}-1}\left((\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1})^{\frac{1}{\delta}} + (\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2})^{\frac{1}{\delta}}\right)^{\delta-1}\right)^{D_{ik}} \cdot$$

$$\exp\left(-\left((\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1})^{\frac{1}{\delta}} + (\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2})^{\frac{1}{\delta}}\right)^\delta\right) \cdot$$

$$\gamma_k^{a_\gamma-1} \exp(-\gamma_k/b_\gamma)s_\alpha^{-1} \exp\left(-\frac{1}{2s_\alpha^{2q}}(\alpha_k - m_\alpha)^T(\alpha_k - m_\alpha)\right) s_\theta^{-1} \exp\left(-\frac{1}{2s_\theta^{2p}}(\theta_k - m_\theta)^T(\theta_k - m_\theta)\right) \cdot$$

$$\delta^{a_\delta-1}(1-\delta)^{b_\delta-1}$$

And the full conditionals are below.

$p(\beta|\text{ all}) \propto$

$$\prod_{i=1}^{N} (2\pi\sigma_\epsilon)^{-\frac{J_i}{2}} \exp\left\{-\frac{1}{2\sigma_\epsilon^{J_i}} (Y_i - X_i\beta - U_i)^T (Y_i - X_i\beta - U_i)\right\} \cdot$$

$$s_\beta^{-p} \exp\left(-\frac{1}{2s_\beta^{2p}} (\beta - m_\beta)^T (\beta - m_\beta)\right)$$

$$= \sigma_\epsilon^{-\sum_{i=1}^{N} J_i/2} s_\beta^{-p} \exp\left\{-\frac{1}{2\sigma_\epsilon^{2\sum_{i=1}^{N} J_i}} \sum_{i=1}^{N} (Y_i - X_i\beta - U_i)^T (Y_i - X_i\beta - U_i)\right.$$

$$\left.-\frac{1}{2s_\beta^{2p}} (\beta - m_\beta)^T (\beta - m_\beta)\right\}$$

$$\sim N(A, B)$$

where

$$B = \left(\frac{1}{s_\beta^{2p}} I_p + \frac{1}{\sigma_\epsilon^{2\sum_{i=1}^{N} J_i/2}} \sum_{i=1}^{N} X_i^T X_i\right)^{-1}$$

and

$$A = B^{-1} \left(\frac{1}{\sigma_\epsilon^{2\sum_{i=1}^{N} J_i/2}} \left(\sum_{i=1}^{N} U_i X_i - \sum_{i=1}^{N} Y_i^T X_i\right) - \frac{1}{s_\beta^{2p}} m_\beta\right)$$

$p(\sigma_U|\text{ all}) \propto$

$$\prod_{i=1}^{N} \sigma_U^{-1} \exp\left(-\frac{1}{2\sigma_U^2} U_i^2\right) \sigma_U^{a_U-1} \exp(-\sigma_U/b_U) \cdot$$

$$= \sigma_U^{a_U-2} \exp\left(-\frac{1}{2\sigma_U^2} \sum_{i=1}^{N} U_i^2 - \sigma_U/b_U\right)$$

$p(\sigma_\epsilon|\text{ all}) \propto$

$$\prod_{i=1}^{N} (2\pi\sigma_\epsilon)^{-\frac{J_i}{2}} \exp\left\{-\frac{1}{2\sigma_\epsilon^{2J_i}} (Y_i - X_i\beta - U_i)^T (Y_i - X_i\beta - U_i)\right\} \sigma_\epsilon^{a_\epsilon-1} \cdot$$

$$\exp(-\sigma_\epsilon/b_\epsilon)$$

228

$p(\gamma_k|\text{ all})$

$$\propto \left(\gamma_k \exp\left(W_i\alpha_k - \theta_k^T U_i\right)^{\frac{1}{\delta}} T_i^{\frac{\gamma_k}{\delta}-1}\cdot\right.$$

$$\left(\left(\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1}\right)^{\frac{1}{\delta}} + \left(\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2}\right)^{\frac{1}{\delta}}\right)^{\delta-1}\right)^{D_{ik}}\cdot$$

$$\exp\left(-\left(\left(\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1}\right)^{\frac{1}{\delta}} + \left(\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2}\right)^{\frac{1}{\delta}}\right)^{\delta}\right)\cdot$$

$$\gamma_k^{a_\gamma-1} \exp(-\gamma_k/b_\gamma)$$

$p(\alpha_k|\text{ all})$

$$\propto \left(\gamma_k \exp\left(W_i\alpha_k - \theta_k^T U_i\right)^{\frac{1}{\delta}} T_i^{\frac{\gamma_k}{\delta}-1}\cdot\right.$$

$$\left(\left(\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1}\right)^{\frac{1}{\delta}} + \left(\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2}\right)^{\frac{1}{\delta}}\right)^{\delta-1}\right)^{D_{ik}}\cdot$$

$$\exp\left(-\left(\left(\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1}\right)^{\frac{1}{\delta}} + \left(\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2}\right)^{\frac{1}{\delta}}\right)^{\delta}\right) s_\alpha^{-1}\cdot$$

$$\exp\left(-\frac{1}{2s_\alpha^{2q}}(\alpha_k - m_\alpha)^T(\alpha_k - m_\alpha)\right)$$

$p(\theta_k|\text{ all})$

$$\propto \left(\gamma_k \exp\left(W_i\alpha_k - \theta_k^T U_i\right)^{\frac{1}{\delta}} T_i^{\frac{\gamma_k}{\delta}-1}\cdot\right.$$

$$\left(\left(\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1}\right)^{\frac{1}{\delta}} + \left(\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2}\right)^{\frac{1}{\delta}}\right)^{\delta-1}\right)^{D_{ik}}\cdot$$

$$\exp\left(-\left(\left(\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1}\right)^{\frac{1}{\delta}} + \left(\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2}\right)^{\frac{1}{\delta}}\right)^{\delta}\right) s_\theta^{-1}\cdot$$

$$\exp\left(-\frac{1}{2s_\theta^{2p}}(\theta_k - m_\theta)^T(\theta_k - m_\theta)\right)$$

$p(\delta| \text{ all})$

$$\propto \left(\gamma_k \exp\left(W_i\alpha_k - \theta_k^T U_i\right)^{\frac{1}{\delta}} T_i^{\frac{\gamma_k}{\delta}-1} \cdot \right.$$

$$\left(\left(\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1}\right)^{\frac{1}{\delta}} + \left(\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2}\right)^{\frac{1}{\delta}}\right)^{\delta-1} \right)^{D_{ik}} \cdot$$

$$\exp\left(-\left(\left(\exp\left(W_i\alpha_1 - \theta_1^T U_i\right) T_i^{\gamma_1}\right)^{\frac{1}{\delta}} + \left(\exp\left(W_i\alpha_2 - \theta_2^T U_i\right) T_i^{\gamma_2}\right)^{\frac{1}{\delta}}\right)^{\delta}\right) \cdot$$

$$\delta^{a_\delta-1}(1-\delta)^{b_\delta-1}$$

Appendix H

Proof of Representation Theorem from Chapter 4

Extending a distribution in Lu and Bhattacharyya's paper (Lu and Bhattacharyya, 1990), consider the following multivariate (joint) survival distribution for K variables $T_1, ..., T_K$:

$$S_T(t_1, t_2, ..., t_K) = \exp\left(-\left[H_1(t_1)^{\frac{1}{\delta}} + ... + H_K(t_K)^{\frac{1}{\delta}}\right]^\delta\right) \tag{H.1}$$

where $0 < \delta \leq 1$, and $H_1, ..., H_K$ are arbitrary cumulative hazard functions. If each $H_k(t)$ is a cumulative hazard function then there exists an inverse function $H_k^{-1}(t)$ which is also non-decreasing. Additionally, for any cumulative hazard function $H_k(t)$ and δ in $(0, 1]$, $H_k(t)^{\frac{1}{\delta}}$ is also a valid cumulative hazard function.

Let $Z_k = H_k(T_k)^{\frac{1}{\delta}}$, $k = 1, ..., K$. We can write the joint survival function of the Z_k using the joint survival function of the T_k in (H.1).

$$
\begin{aligned}
S_Z(z_1, ..., z_K) &= Pr(Z_1 > z_1, ..., Z_K > z_K) \\
&= Pr(T_1 > H_1^{-1}(z_1^\delta), ..., T_K > H_K^{-1}(z_K^\delta)) \\
&= S_T(H_1^{-1}(z_1^\delta), ..., H_K^{-1}(z_K^\delta)) \\
&= \exp\left(-\left[H_1(H_1^{-1}(z_1^\delta))^{\frac{1}{\delta}} + ... + H_K(H_K^{-1}(z_K^\delta))^{\frac{1}{\delta}}\right]^\delta\right) \\
&= \exp\left(-[z_1 + ... + z_K]^\delta\right)
\end{aligned}
$$

Define the random variable $Y_{(K)} = Z_1 + ... + Z_K$, with realization $y_{(K)} = z_1 + ... + z_K$. The

joint density of the Z's is given by

$$
\begin{aligned}
f_Z(z_1, ..., z_K) &= \frac{\partial^K}{\partial z_1 \cdots \partial z_K} S_Z(z_1, ..., z_K) \\
&= (-1)^K \frac{\partial^K}{\partial y_{(K)}^K} \exp\left(-y_{(K)}^\delta\right)
\end{aligned}
$$

See Section H.1 for details. Define the $g_K(y)$ functions as

$$
g_K(y) = (-1)^K \frac{\partial^K}{\partial y^K} \exp\left(-y^\delta\right) \tag{H.2}
$$

So that

$$
f_Z(z_1, ..., z_K) = g_K\left(y_{(K)}\right) \tag{H.3}
$$

We can show that $g_K(y)$ can be written in the form in (H.4),

$$
\begin{aligned}
g_K(y) = \exp(-y^\delta) \Big(& (-1)^{K-K} a_{K,K} y^{K\delta-K} \\
& + (-1)^{K-(K-1)} a_{K,K-1} y^{(K-1)\delta-K} \\
& + ... \\
& + (-1)^{K-k} a_{K,k} y^{k\delta-K} \\
& + ... \\
& + (-1)^{K-2} a_{K,2} y^{2\delta-K} \\
& + (-1)^{K-1} a_{K,1} y^{\delta-K} \Big)
\end{aligned} \tag{H.4}
$$

The $a_{K,k}$ coefficients are defined recursively.

$$
\begin{aligned}
& a_{1,1} = \delta; \\
& a_{K,K} = \delta a_{K-1,K-1}; \\
& a_{K,1} = (\delta - (K-1)) a_{K-1,1}; \\
& a_{K,k} = (k\delta - (K-1)) a_{K-1,k} + \delta a_{K-1,k-1}, \text{ for } 2 \le k \le K-1
\end{aligned} \tag{H.5}
$$

See Section H.2.1 of the appendix for the proof of (H.3) - (H.5).

Define the random variables $V_1, ..., V_K$

$$V_1 = (Z_1 + ... + Z_K)^\delta = Y_{(K)}^\delta$$

$$V_2 = \frac{Z_1}{Z_1 + Z_2}$$

$$V_3 = \frac{Z_1 + Z_2}{Z_1 + Z_2 + Z_3}$$

$$...$$

$$V_K = \frac{Z_1 + ... + Z_{K-1}}{Z_1 + ... + Z_{K-1} + Z_K}$$

We can write the $Z_1, ..., Z_K$ in terms of the V variables.

$$Z_1 = V_2 \cdots V_K V_1^{1/\delta}$$

$$Z_2 = (1 - V_2)V_3 \cdots V_K V_1^{1/\delta}$$

$$Z_3 = (1 - V_3)V_4 \cdots V_K V_1^{1/\delta} \tag{H.6}$$

$$...$$

$$Z_K = (1 - V_K)V_1^{1/\delta}$$

Note that $y_{(K)} = v_1^{\frac{1}{\delta}}$. Call the Jacobian matrix of this transformation $J_{(K)}$. The determinant of the Jacobian is below. See Section H.3 in the appendix for the proof of (H.7).

$$det\left(J_{(K)}\right) = \frac{(-1)^{K-1}}{\delta} v_1^{\frac{K}{\delta}-1} v_3 v_4^2 v_5^3 \cdots v_K^{K-2} \tag{H.7}$$

The joint density of $V_1, ..., V_K$ is derived in Section H.4 and is equal to

$$f_V(v_1, v_2, ..., v_K) = \left| det\left(J_{(K)}\right)\right| f_Z(z_1, ..., z_K)$$

$$= \frac{1}{\delta} v_2^0 v_3 v_4^2 v_5^3 \cdots v_K^{K-2} \cdot \exp(-v_1)\Big(a_{K,K} v_1^{K-1} + (-1)^{K-(K-1)} a_{K,K-1} v_1^{K-2} + ...$$

$$+ (-1)^{K-k} a_{K,k} v_1^{k-1} + ... + (-1)^{K-2} a_{K,2} v_1 + (-1)^{K-1} a_{K,1}\Big)$$

(H.8)

Since $f_V(v_1, v_2, ..., v_K)$ factors, $V_1, ..., V_K$ are independent. The kernel of the marginal density for each V_k will have the same form as the terms with v_k in the joint density. So we can see that

$$V_2 \sim Beta(1,1)$$

$$V_3 \sim Beta(2,1)$$

$$\vdots$$

$$V_K \sim Beta(K-1,1)$$

and V_1 is a mixture of the Gamma distributions $\Gamma(K,1), \Gamma(K-1,1), ..., \Gamma(2,1), \Gamma(1,1)$ with the following distribution

$$f_{V_1}(v) = \frac{1}{\delta \Gamma(K)} \exp(-v_1)\Big(a_{K,K} v_1^{K-1} + (-1)^{K-(K-1)} a_{K,K-1} v_1^{K-2} + ...$$

$$+ (-1)^{K-k} a_{K,k} v_1^{k-1} + ...$$

$$+ (-1)^{K-2} a_{K,2} v_1 + (-1)^{K-1} a_{K,1}\Big)$$

(H.9)

where $\Gamma(x)$ is the Gamma function, i.e. $\Gamma(k) = \int_0^\infty x^{k-1} e^{-x} dx$. See Section H.4 of the appendix for proof that $f_{V_1}(v)$ is a valid distribution.

H.1 Proof of the Joint density of the Z Variables

Define the random variable $Y_{(K)} = Z_1 + ... + Z_K$, with realization $y_{(K)} = z_1 + ... + z_K$. Note that the derivative of $y_{(K)}$ with respect to any z_k is 1, i.e. $\frac{\partial y_{(K)}}{\partial z_k} = 1, \quad \forall k = 1, ..., K$. And so

$$
\begin{aligned}
\frac{\partial}{\partial z_k} S_Z(z_1, ..., z_K) &= \frac{\partial}{\partial z_k} \exp\left(-(z_1 + ... + z_K)^\delta\right) \\
&= \exp\left(-(z_1 + ... + z_K)^\delta\right)(-\delta)(z_1 + ... + z_K)^{\delta-1} \\
&= \exp\left(-y_{(K)}^\delta\right)(-\delta)\left(y_{(K)}\right)^{\delta-1} \\
&= \frac{\partial}{\partial y_{(K)}} \exp\left(-y_{(K)}^\delta\right)
\end{aligned}
$$

Extending this, for any $k_1, k_2, ..., k_m \in \{1, 2, ..., K\}$

$$
\frac{\partial^m}{\partial z_{k_1} \partial z_{k_2} \cdots \partial z_{k_m}} S_Z(z_1, ..., z_K) = \frac{\partial^m}{\partial y_{(K)}^m} \exp\left(-y_{(K)}^\delta\right) \tag{H.10}
$$

H.2 Verifying (H.1) is a Survival Function

It is easy to see that for all $t_1, ..., t_K$, (H.1) is non-negative, so $S_T(t_1, ..., t_K) \geq 0$.

Now $t_k \geq 0$ and H_k is a cumulative hazard function so it has range $[0, \infty)$ for $k = 1, 2, ..., K$,

and $0 < \delta \leq 1$. So

$$H_k(t_k) \geq 0, \forall k = 1, ..., K, \forall t_k \in [0, \infty)$$

$$\Rightarrow (H_k(t_k))^{\frac{1}{\delta}} \geq 0$$

$$\Rightarrow (H_k(t_k))^{\frac{1}{\delta}} + ... (H_k(t_k))^{\frac{1}{\delta}} \geq 0$$

$$\Rightarrow \left((H_k(t_k))^{\frac{1}{\delta}} + ... (H_k(t_k))^{\frac{1}{\delta}} \right)^{\delta} \geq 0$$

$$\Rightarrow - \left((H_k(t_k))^{\frac{1}{\delta}} + ... (H_k(t_k))^{\frac{1}{\delta}} \right)^{\delta} \leq 0$$

$$\Rightarrow \exp \left(- \left((H_k(t_k))^{\frac{1}{\delta}} + ... (H_k(t_k))^{\frac{1}{\delta}} \right)^{\delta} \right) \leq 1$$

$$\Rightarrow S_T(t_1, ..., t_K) \leq 1$$

Suppose $a \leq b$. Then since $H_k(t)$ is a cumulative hazard function for any k

$$H_1(a) \leq H_1(b)$$

$$\Rightarrow (H_1(a))^{\frac{1}{\delta}} \leq (H_1(b))^{\frac{1}{\delta}}$$

$$\Rightarrow (H_1(a))^{\frac{1}{\delta}} + (H_2(t_2))^{\frac{1}{\delta}} ... + (H_K(t_K))^{\frac{1}{\delta}} \leq (H_1(b))^{\frac{1}{\delta}} + (H_2(t_2))^{\frac{1}{\delta}} ... + (H_K(t_K))^{\frac{1}{\delta}}$$

$$\Rightarrow \left((H_1(a))^{\frac{1}{\delta}} + (H_2(t_2))^{\frac{1}{\delta}} ... + (H_K(t_K))^{\frac{1}{\delta}} \right)^{\delta} \leq \left((H_1(b))^{\frac{1}{\delta}} + (H_2(t_2))^{\frac{1}{\delta}} ... + (H_K(t_K))^{\frac{1}{\delta}} \right)^{\delta}$$

$$\Rightarrow - \left((H_1(a))^{\frac{1}{\delta}} + (H_2(t_2))^{\frac{1}{\delta}} ... + (H_K(t_K))^{\frac{1}{\delta}} \right)^{\delta} \geq - \left((H_1(b))^{\frac{1}{\delta}} + (H_2(t_2))^{\frac{1}{\delta}} ... + (H_K(t_K))^{\frac{1}{\delta}} \right)^{\delta}$$

$$\Rightarrow \exp \left(- \left((H_1(a))^{\frac{1}{\delta}} + (H_2(t_2))^{\frac{1}{\delta}} ... + (H_K(t_K))^{\frac{1}{\delta}} \right)^{\delta} \right) \geq$$

$$\exp \left(- \left((H_1(b))^{\frac{1}{\delta}} + (H_2(t_2))^{\frac{1}{\delta}} ... + (H_K(t_K))^{\frac{1}{\delta}} \right)^{\delta} \right)$$

$$\Rightarrow S_T(a, t_2, ..., t_K) \geq S_T(b, t_2, ..., t_K)$$

This also holds for if a and b are substituted for any of $t_2, ..., t_K$. Hence S_T is non-increasing.

Note that $H_k(0) = 0$ for all k so

$$S_T(0, ..., 0) = \exp\left(-\left((H_1(0))^{\frac{1}{\delta}} + ... (H_K(0))^{\frac{1}{\delta}}\right)^{\delta}\right)$$

$$= \exp(0)$$

$$= 1$$

Also,

$$\lim_{t_1 \to \infty, ..., t_K \to \infty} S_T(t_1, ..., t_K) = \lim_{t_1 \to \infty, ..., t_K \to \infty} \exp\left(-\left((H_1(t_1))^{\frac{1}{\delta}} + ... (H_K(t_K))^{\frac{1}{\delta}}\right)^{\delta}\right)$$

$$= \lim_{y \to \infty} \exp\left(-y^{\delta}\right)$$

$$= 0$$

And

$$\lim_{t_1 \to \infty} S_T(t_1, ..., t_K) = \lim_{t_1 \to \infty} \exp\left(-\left((H_1(t_1))^{\frac{\gamma_1}{\delta}} + ... (H_K(t_K))^{\frac{\gamma_K}{\delta}}\right)^{\delta}\right)$$

$$= \lim_{y \to \infty} \exp\left(-y^{\delta}\right)$$

$$= 0$$

This holds for $t_k \to \infty$ for any $k = 1, ..., K$. Finally

$$S_T(t_1, 0, ..., 0) = \exp\left(-H_1(t_1)\right) = \exp\left(-\left(-\log(S_1(t_1))\right)\right) = S_1(t_1)$$

Similarly for any $k = 2, ..., K$, $t_k \neq 0$ and $t_l = 0$, for all $l \neq k$.

So S_T in (H.1) is a valid survival function.

H.2.1   Proof of Formula for Joint Density of $Z_1, ..., Z_K$ in (H.3 - H.5)

We will prove by induction.

We will write $f_{Z^{(K)}}(z)$ for the joint density of the K random variables $(Z_1, .., Z_K)$. For ease of notation we will drop the subscripts on $Y_{(K)}$ and $y_{(K)}$.

For $K = 1, Y = Y_{(1)} = Z_1$ and

$$
\begin{aligned}
f_{Z^{(1)}}(z_1) &= g_1(y) \\
&= (-1)^1 \frac{\partial}{\partial y} \exp(-y^\delta) \\
&= -\left(-\delta y^{\delta-1} \exp(-y^\delta)\right) \\
&= \delta y^{\delta-1} \exp(-y^\delta) \\
&= \exp(-y^\delta)(-1)^0 a_{1,1} y^{\delta-1}
\end{aligned}
$$

We have $a_{1,1} = \delta$ and the form for $g_K(y)$ holds for $K = 1$.

For $K = 2, Y = Y_{(2)} = Z_1 + Z_2$ and

$$
\begin{aligned}
f_{Z^{(2)}}(z_1, z_2) &= g_2(y) \\
&= (-1)^2 \frac{\partial^2}{\partial y^2} \exp(-y^\delta) \\
&= \frac{\partial}{\partial y} \left[-\delta y^{\delta-1} \exp(-y^\delta) \right] \\
&= (-\delta y^{\delta-1})(-\delta y^{\delta-1}) \exp(-y^\delta) + \exp(-y^\delta)(-\delta(\delta-1)y^{\delta-2}) \\
&= \exp(-y^\delta) \left(\delta^2 y^{2\delta-2} - \delta(\delta-1)y^{\delta-2} \right) \\
&= \exp(-y^\delta) \left((-1)^0 a_{2,2} y^{2\delta-2} + (-1)a_{2,1} y^{\delta-2} \right)
\end{aligned}
$$

Hence $a_{2,2} = \delta^2 = \delta a_{1,1}$, and $a_{2,1} = (\delta-1)\delta = (\delta-(2-1))a_{1,1}$. The formula holds for $K = 2$.

For $K = 3$, $Y = Y_{(3)} = Z_1 + Z_2 + Z_3$.

$$f_{Z^{(3)}}(z_1, z_2, z_3) = g_3(y)$$

$$= (-1)^3 \frac{\partial^3}{\partial y^3} \exp(-y^\delta)$$

$$= (-1)\frac{\partial}{\partial y}\left[\exp(-y^\delta)\delta^2 y^{2\delta-2} - \exp(-y^\delta)\delta(\delta-1)y^{\delta-2}\right]$$

$$= (-1)\Big(\exp(-y^\delta)\left(\delta^2(2\delta-2)y^{2\delta-3}\right) + \delta^2 y^{2\delta-2}\left(-\delta y^{\delta-1}\right)\exp(-y^\delta)$$

$$- \left[\exp(-y^\delta)\left(\delta(\delta-1)(\delta-2)y^{\delta-3}\right) + \delta(\delta-1)y^{\delta-2}\left(-\delta y^{\delta-1}\right)\exp(-y^\delta)\right]\Big)$$

$$= \exp(-y^\delta)\Big(-\delta^2(2\delta-2)y^{2\delta-3} + \delta^3 y^{3\delta-3} + \delta(\delta-1)(\delta-2)y^{\delta-3}$$

$$- \delta^2(\delta-1)y^{2\delta-3}\Big)$$

$$= \exp(-y^\delta)\Big((-1)^0\delta^3 y^{3\delta-3} + (-1)^1\left[\delta^2(2\delta-2) + \delta^2(\delta-1)\right]y^{2\delta-3}$$

$$+ (-1)^2\delta(\delta-1)(\delta-2)y^{\delta-3}\Big)$$

$$= \exp(-y^\delta)\Big((-1)^0 a_{3,3} y^{3\delta-3} + (-1)^1 a_{3,2} y^{2\delta-3} + (-1)^2 a_{3,1} y^{\delta-3}\Big)$$

and the formula holds for $K = 3$.

Assume that (H.4) holds for some K. Note that $\frac{\partial^K}{\partial y^K}\exp(-y^\delta) = (-1)^K g_K(y)$. Then for $K+1$

variables

$$f_{Z^{(K+1)}}(z_1, ..., z_{K+1}) = g_{K+1}(y)$$

$$= (-1)^{K+1} \frac{\partial^{K+1}}{\partial y^{K+1}} \exp(-y^\delta)$$

$$= (-1)^{K+1} \frac{\partial}{\partial y} \left[\frac{\partial^K}{\partial y^K} \exp(-y^\delta) \right]$$

$$= (-1)^{K+1} \frac{\partial}{\partial y} \left[(-1)^K g_K(y) \right]$$

$$= (-1)^{2K+1} \frac{\partial}{\partial y} \Bigg(\exp(-y^\delta) \Big[(-1)^{K-K} a_{K,K} y^{K\delta-K}$$

$$+ (-1)^{K-(K-1)} a_{K,K-1} y^{(K-1)\delta-K}$$

$$+ ...$$

$$+ (-1)^{K-k} a_{K,k} y^{k\delta-K}$$

$$+ ...$$

$$+ (-1)^{K-2} a_{K,2} y^{2\delta-K}$$

$$+ (-1)^{K-1} a_{K,1} y^{\delta-K} \Big] \Bigg)$$

$$\Rightarrow f_{Z^{(K+1)}}(z_1, ..., z_{K+1}) = (-1) \Bigg(\exp(-y^\delta) \Big[a_{K,K}(K\delta - K)y^{K\delta - K - 1}$$

$$+ (-1)^{K-(K-1)} a_{K,K-1} \left[(K-1)\delta - K \right] \cdot$$

$$y^{(K-1)\delta - K - 1}$$

$$+ ...$$

$$+ (-1)^{K-k} a_{K,k}(k\delta - K)y^{k\delta - K - 1}$$

$$+ ...$$

$$+ (-1)^{K-2} a_{K,2}(2\delta - K)y^{2\delta - K - 1}$$

$$+ (-1)^{K-1} a_{K,1}(\delta - K)y^{\delta - K - 1} \Big]$$

$$+ \exp(-y^\delta)\left(-\delta y^{\delta - 1}\right) \Big[a_{K,K} y^{K\delta - K}$$

$$+ (-1)^{K-(K-1)} a_{K,K-1} y^{(K-1)\delta - K}$$

$$+ ...$$

$$+ (-1)^{K-k} a_{K,k} y^{k\delta - K}$$

$$+ ...$$

$$+ (-1)^{K-2} a_{K,2} y^{2\delta - K}$$

$$+ (-1)^{K-1} a_{K,1} y^{\delta - K} \Big] \Bigg)$$

$$\Rightarrow f_{Z^{(K+1)}}(z_1, ..., z_{K+1}) = (-1)\exp(-y^\delta)\Big(a_{K,K}(K\delta - K)y^{K\delta - K - 1}$$

$$+ (-1)^{K-(K-1)}a_{K,K-1}\left[(K-1)\delta - K\right] \cdot$$
$$y^{(K-1)\delta - K - 1}$$

$$+ ...$$

$$+ (-1)^{K-k}a_{K,k}(k\delta - K)y^{k\delta - K - 1}$$

$$+ ...$$

$$+ (-1)^{K-2}a_{K,2}(2\delta - K)y^{2\delta - K - 1}$$

$$+ (-1)^{K-1}a_{K,1}(\delta - K)y^{\delta - K - 1}$$

$$+ (-\delta)a_{K,K}y^{(K+1)\delta - K - 1}$$

$$+ (-1)^{K-(K-1)+1}\delta a_{K,K-1}y^{K\delta - K - 1}$$

$$+ ...$$

$$+ (-1)^{K-k+1}\delta a_{K,k}y^{(k+1)\delta - K - 1}$$

$$+ ...$$

$$+ (-1)^{K-2+1}\delta a_{K,2}y^{3\delta - K - 1}$$

$$+ (-1)^{K-1+1}\delta a_{K,1}y^{2\delta - K - 1}\Big)$$

$$\Rightarrow f_{Z^{(K+1)}}(z_1, ..., z_{K+1}) = (-1)\exp(-y^\delta)\Big((-\delta)a_{K,K}\cdot$$

$$y^{(K+1)\delta-(K+1)}$$

$$+ (-1)^{K-(K-1)+1}(\delta + (K\delta - K))a_{K,K-1}\cdot$$

$$y^{K\delta-(K+1)}$$

$$+ (-1)^{K-(K-1)}\left[(K-1)\delta - K\right]a_{K,K-1}\cdot$$

$$y^{(K-1)\delta-(K+1)}$$

$$+ ...$$

$$+ (-1)^{K+1-k}\delta a_{K,k}y^{(k+1)\delta-(K+1)}$$

$$+ (-1)^{K-k}(k\delta - K)a_{K,k}y^{k\delta-(K+1)}$$

$$+ ...$$

$$+ (-1)^{K+1-2}\left[\delta a_{K,2} + (3\delta - K)a_{K,3}\right]y^{3\delta-(K+1)}$$

$$+ (-1)^{K+1-1}\left[\delta a_{K,1} + (2\delta - K)a_{K,2}\right]y^{2\delta-(K+1)}$$

$$+ (-1)^{K-1}(\delta - K)a_{K,1}y^{\delta-(K+1)}\Big)$$

$$\Rightarrow f_{Z^{(K+1)}}(z_1, ..., z_{K+1}) = \exp(-y^\delta)\Big(\delta a_{K,K}y^{(K+1)\delta-(K+1)}$$

$$+ (-1)^{(K+1)-K}\left[(K\delta - K)a_{K,K} + \delta a_{K,K-1}\right]y^{K\delta-(K+1)}$$

$$+ ...$$

$$+ (-1)^{K+1-k}\left[(k\delta - K)a_{K,k} + \delta a_{K,k-1}\right]y^{k\delta-(K+1)}$$

$$+ ...$$

$$+ (-1)^{K+1-1+1}\left[(2\delta - K)a_{K,2} + \delta a_{K,1}\right]y^{2\delta-(K+1)}$$

$$+ (-1)^{K+1-1}(\delta - K)a_{K,1}y^{\delta-(K+1)}\Big)$$

$$\Rightarrow f_{Z^{(K+1)}}(z_1, ..., z_{K+1}) = \exp(-y^\delta)\Big((-1)^{(K+1)-(K+1)}a_{K+1,K+1}y^{(K+1)\delta-(K+1)}$$

$$+ (-1)^{(K+1)-K}a_{K+1,K}y^{K\delta-(K+1)}$$

$$+ ...$$

$$+ (-1)^{(K+1)-k}a_{K+1,k}y^{k\delta-(K+1)}$$

$$+ ...$$

$$+ (-1)^{(K+1)-2}a_{K+1,2}y^{2\delta-(K+1)}$$

$$+ (-1)^{(K+1)-1}a_{K+1,1}y^{\delta-(K+1)}\Big)$$

The right hand side is (H.4) for $K+1$ variables and the result is shown.

H.3 Proof of Transformation Jacobian in (H.7)

We will show that

$$det\,(J_K) = \frac{(-1)^{K-1}}{\delta}v_1^{\frac{K}{\delta}-1}v_3v_4^2v_5^3\cdots v_K^{K-2}$$

The determinant of the Jacobian matrix for the transformation in (H.6) is

$$\det(J_K) = \begin{vmatrix} \dfrac{\partial z_1}{\partial v_1} & \dfrac{\partial z_1}{\partial v_2} & \dfrac{\partial z_1}{\partial v_3} & \dfrac{\partial z_1}{\partial v_4} & \cdots & \dfrac{\partial z_1}{\partial v_{K-1}} & \dfrac{\partial z_1}{\partial v_K} \\[2mm] \dfrac{\partial z_2}{\partial v_1} & \dfrac{\partial z_2}{\partial v_2} & \dfrac{\partial z_2}{\partial v_3} & \dfrac{\partial z_2}{\partial v_4} & \cdots & \dfrac{\partial z_2}{\partial v_{K-1}} & \dfrac{\partial z_2}{\partial v_K} \\[2mm] \cdots & \cdots & \cdots & \cdots & \ddots & \cdots & \cdots \\[2mm] \dfrac{\partial z_{K-1}}{\partial v_1} & \dfrac{\partial z_{K-1}}{\partial v_2} & \dfrac{\partial z_{K-1}}{\partial v_3} & \dfrac{\partial z_{K-1}}{\partial v_4} & \cdots & \dfrac{\partial z_{K-1}}{\partial v_{K-1}} & \dfrac{\partial z_{K-1}}{\partial v_K} \\[2mm] \dfrac{\partial z_K}{\partial v_1} & \dfrac{\partial z_K}{\partial v_2} & \dfrac{\partial z_K}{\partial v_3} & \dfrac{\partial z_K}{\partial v_4} & \cdots & \dfrac{\partial z_K}{\partial v_{K-1}} & \dfrac{\partial z_K}{\partial v_K} \end{vmatrix}$$

$$\det(J_K) = \begin{vmatrix} \frac{1}{\delta} v_1^{\frac{1}{\delta}-1} v_2 v_3 \ldots v_K & v_1^{\frac{1}{\delta}} v_3 \ldots v_K & v_1^{\frac{1}{\delta}} v_2 v_4 \ldots v_K & v_1^{\frac{1}{\delta}} v_2 v_3 v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}} v_2 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}} v_2 \ldots v_{K-1} \\[2mm] \frac{1}{\delta} v_1^{\frac{1}{\delta}-1}(1-v_2) v_3 \ldots v_K & -v_1^{\frac{1}{\delta}} v_3 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2) v_4 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2) v_3 v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_2) v_3 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_2) v_3 \ldots v_{K-1} \\[2mm] \frac{1}{\delta} v_1^{\frac{1}{\delta}-1}(1-v_3) v_4 \ldots v_K & 0 & -v_1^{\frac{1}{\delta}} v_4 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_3) v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_3) v_4 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_3) v_4 \ldots v_{K-1} \\[2mm] \frac{1}{\delta} v_1^{\frac{1}{\delta}-1}(1-v_4) v_5 \ldots v_K & 0 & 0 & -v_1^{\frac{1}{\delta}} v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_4) v_5 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_4) v_5 \ldots v_{K-1} \\[2mm] \ldots & \ldots & \ldots & \ldots & \ddots & \ldots & \ldots \\[2mm] \frac{1}{\delta} v_1^{\frac{1}{\delta}-1}(1-v_{K-1}) v_K & 0 & 0 & 0 & \ldots & -v_1^{\frac{1}{\delta}} v_K & v_1^{\frac{1}{\delta}}(1-v_{K-1}) \\[2mm] \frac{1}{\delta} v_1^{\frac{1}{\delta}-1}(1-v_K) & 0 & 0 & 0 & \ldots & 0 & -v_1^{\frac{1}{\delta}} \end{vmatrix}$$

245

Factor out $\frac{1}{\delta}v_1^{\frac{1}{\delta}-1}$ from the first column.

$$
det\left(J_K\right) = \frac{1}{\delta}v_1^{\frac{1}{\delta}-1}
\begin{vmatrix}
v_2 v_3 \ldots v_K & v_1^{\frac{1}{\delta}} v_3 \ldots v_K & v_1^{\frac{1}{\delta}} v_2 v_4 \ldots v_K & v_1^{\frac{1}{\delta}} v_2 v_3 v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}} v_2 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}} v_2 \ldots v_{K-1} \\
(1-v_2) v_3 \ldots v_K & -v_1^{\frac{1}{\delta}} v_3 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2) v_4 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2) v_3 v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_2) v_3 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_2) v_3 \ldots v_{K-1} \\
(1-v_3) v_4 \ldots v_K & 0 & -v_1^{\frac{1}{\delta}} v_4 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_3) v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_3) v_4 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_3) v_4 \ldots v_{K-1} \\
(1-v_4) v_5 \ldots v_K & 0 & 0 & -v_1^{\frac{1}{\delta}} v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_4) v_5 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_4) v_5 \ldots v_{K-1} \\
\ldots & \ldots & \ldots & \ldots & \ddots & \ldots & \ldots \\
(1-v_{K-1}) v_K & 0 & 0 & 0 & \ldots & -v_1^{\frac{1}{\delta}} v_K & v_1^{\frac{1}{\delta}}(1-v_{K-1}) \\
(1-v_K) & 0 & 0 & 0 & \ldots & 0 & -v_1^{\frac{1}{\delta}}
\end{vmatrix}
$$

Subtract row 2 from row 1. (Subtracting a multiple of row j from row i, $i \neq j$, does not change the value of the determinant.)

$$\det(J_K) = \frac{1}{\delta} v_1^{\frac{1}{\delta}-1}$$

$$
\begin{vmatrix}
v_3 \ldots v_K & 0 & v_1^{\frac{1}{\delta}} v_4 \ldots v_K & v_1^{\frac{1}{\delta}} v_3 v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}} v_3 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}} v_3 \ldots v_{K-1} \\
(1-v_2)v_3 \ldots v_K & -v_1^{\frac{1}{\delta}} v_3 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_4 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_3 v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_2)v_3 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_3 \ldots v_{K-1} \\
(1-v_3)v_4 \ldots v_K & 0 & -v_1^{\frac{1}{\delta}} v_4 \ldots v_K & v_1^{\frac{1}{\delta}}(1-v_3)v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_3)v_4 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_3)v_4 \ldots v_{K-1} \\
(1-v_4)v_5 \ldots v_K & 0 & 0 & -v_1^{\frac{1}{\delta}} v_5 \ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_4)v_5 \ldots v_{K-2} v_K & v_1^{\frac{1}{\delta}}(1-v_4)v_5 \ldots v_{K-1} \\
\ldots & \ldots & \ldots & \ldots & \ddots & \ldots & \ldots \\
(1-v_{K-1})v_K & 0 & 0 & 0 & \ldots & -v_1^{\frac{1}{\delta}} v_K & v_1^{\frac{1}{\delta}}(1-v_{K-1}) \\
(1-v_K) & 0 & 0 & 0 & \ldots & 0 & -v_1^{\frac{1}{\delta}}
\end{vmatrix}
$$

Repeat this, subtracting row 3, then row 4, ..., row K all from row 1 and see that we end up with

247

the following,

$$det\left(J_K\right)=\frac{1}{\delta}v_1^{\frac{1}{\delta}-1}\begin{vmatrix}
1 & 0 & 0 & 0 & \cdots & 0 & 0 \\
(1-v_2)v_3\ldots v_K & -v_1^{\frac{1}{\delta}}v_3\ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_4\ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_3v_5\ldots v_K & \cdots & v_1^{\frac{1}{\delta}}(1-v_2)v_3\ldots v_K-2v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_3\ldots v_{K-1} \\
(1-v_3)v_4\ldots v_K & 0 & -v_1^{\frac{1}{\delta}}v_4\ldots v_K & v_1^{\frac{1}{\delta}}(1-v_3)v_5\ldots v_K & \cdots & v_1^{\frac{1}{\delta}}(1-v_3)v_4\ldots v_K-2v_K & v_1^{\frac{1}{\delta}}(1-v_3)v_4\ldots v_{K-1} \\
(1-v_4)v_5\ldots v_K & 0 & 0 & -v_1^{\frac{1}{\delta}}v_5\ldots v_K & \cdots & v_1^{\frac{1}{\delta}}(1-v_4)v_5\ldots v_K-2v_K & v_1^{\frac{1}{\delta}}(1-v_4)v_5\ldots v_{K-1} \\
\cdots & \cdots & \cdots & \cdots & \ddots & \cdots & \cdots \\
(1-v_{K-1})v_K & 0 & 0 & 0 & \cdots & -v_1^{\frac{1}{\delta}}v_K & v_1^{\frac{1}{\delta}}(1-v_{K-1}) \\
(1-v_K) & 0 & 0 & 0 & \cdots & 0 & -v_1^{\frac{1}{\delta}}
\end{vmatrix}$$

So calculating the determinant above from the minors of the first row, only the (1,1) minor is

248

non-zero.

$$\det(J_K) = \frac{1}{\delta}v_1^{\frac{1}{\delta}-1}
\begin{vmatrix}
-v_1^{\frac{1}{\delta}}v_3\ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_4\ldots v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_3v_5\ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_2)v_3\ldots v_{K-2}v_K & v_1^{\frac{1}{\delta}}(1-v_2)v_3\ldots v_{K-1} \\
0 & -v_1^{\frac{1}{\delta}}v_4\ldots v_K & v_1^{\frac{1}{\delta}}(1-v_3)v_5\ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_3)v_4\ldots v_{K-2}v_K & v_1^{\frac{1}{\delta}}(1-v_3)v_4\ldots v_{K-1} \\
0 & 0 & -v_1^{\frac{1}{\delta}}v_5\ldots v_K & \ldots & v_1^{\frac{1}{\delta}}(1-v_4)v_5\ldots v_{K-2}v_K & v_1^{\frac{1}{\delta}}(1-v_4)v_5\ldots v_{K-1} \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
0 & 0 & 0 & \ldots & -v_1^{\frac{1}{\delta}}v_K & v_1^{\frac{1}{\delta}}(1-v_{K-1}) \\
0 & 0 & 0 & \ldots & 0 & -v_1^{\frac{1}{\delta}}
\end{vmatrix}$$

249

This is an upper triangular matrix so the determinant is the product of the diagonal entries. We see a pattern where v_3 is only in the first diagonal term, v_4 is in the first two diagonal terms, v_5 in the first three, and so on until v_K which is in K-2 terms. And $-v_1^{\frac{1}{\delta}}$ in all $K-1$ diagonal terms. So

$$
\begin{aligned}
det\left(J_K\right) &= \frac{1}{\delta}v_1^{\frac{1}{\delta}-1}v_3v_4^2v_5^3\ldots v_{K-1}^{K-3}v_K^{K-2}\left(-v_1^{\frac{1}{\delta}}\right)^{K-1} \\
&= \frac{(-1)^{K-1}}{\delta}v_1^{\frac{K}{\delta}-1}v_3v_4^2v_5^3\ldots v_{K-1}^{K-3}v_K^{K-2}
\end{aligned}
$$

Which shows the result

$$
det\left(J_K\right) = \frac{(-1)^{K-1}}{\delta}v_1^{\frac{K}{\delta}-1}v_3v_4^2v_5^3\cdots v_K^{K-2}
$$

H.4   Proof of Mixture Distribution for V_1

The joint density of $V_1, ..., V_K$ is

$$
\begin{aligned}
f_V(v_1, v_2, ..., v_K) &= \left|det\left(J_{(K)}\right)\right| f_Z(z_1, ..., z_K) \\
&= \left|det\left(J_{(K)}\right)\right| f_Z(v_2\cdots v_K v_1^{1/\delta}, (1-v_2)v_3\cdots v_K v_1^{1/\delta}, ..., (1-v_K)v_1^{1/\delta}) \\
&= \left|det\left(J_{(K)}\right)\right| g_K(v_2\cdots v_K v_1^{1/\delta} + (1-v_2)v_3\cdots v_K v_1^{1/\delta} + ... + (1-v_K)v_1^{1/\delta}) \\
&= \left|det\left(J_{(K)}\right)\right| g_K(v_1^{1/\delta}(v_2\cdots v_K + -v_2v_3\cdots v_K + v_3\cdots v_K + ... + 1 - v_K)) \\
&= \left|det\left(J_{(K)}\right)\right| g_K(v_1^{1/\delta})
\end{aligned}
$$

$$
\begin{aligned}
\Rightarrow f_V(v_1, v_2, ..., v_K) = \frac{1}{\delta}v_1^{\frac{K}{\delta}-1}v_3v_4^2v_5^3\cdots v_K^{K-2}\cdot \exp(-(v_1^{1/\delta})^\delta)\Big(a_{K,K}v_1^{\frac{K\delta-K}{\delta}} \\
+ (-1)^{K-(K-1)}a_{K,K-1}v_1^{\frac{(K-1)\delta-K}{\delta}} + ... \\
+ (-1)^{K-k}a_{K,k}v_1^{\frac{k\delta-K}{\delta}} + ... \\
+ (-1)^{K-2}a_{K,2}v_1^{\frac{2\delta-K}{\delta}} \\
+ (-1)^{K-1}a_{K,1}v_1^{\frac{\delta-K}{\delta}}\Big)
\end{aligned}
$$

$$\Rightarrow f_V(v_1, v_2, ..., v_K) = \frac{1}{\delta} v_1^{\frac{K}{\delta}-1} v_3 v_4^2 v_5^3 \cdots v_K^{K-2} \cdot \exp(-v_1)\Big(a_{K,K} v_1^{K-\frac{K}{\delta}}$$

$$+ (-1)^{K-(K-1)} a_{K,K-1} v_1^{K-1-\frac{K}{\delta}} + \dots$$

$$+ (-1)^{K-k} a_{K,k} v_1^{k-\frac{K}{\delta}} + \dots$$

$$+ (-1)^{K-2} a_{K,2} v_1^{2-\frac{K}{\delta}}$$

$$+ (-1)^{K-1} a_{K,1} v_1^{1-\frac{K}{\delta}}\Big)$$

$$\Rightarrow f_V(v_1, v_2, ..., v_K) = \frac{1}{\delta} v_2^0 v_3 v_4^2 v_5^3 \cdots v_K^{K-2} \cdot \exp(-v_1)\Big(a_{K,K} v_1^{K-1}$$

$$+ (-1)^{K-(K-1)} a_{K,K-1} v_1^{K-2} + \dots$$

$$+ (-1)^{K-k} a_{K,k} v_1^{k-1} + \dots$$

$$+ (-1)^{K-2} a_{K,2} v_1^{1}$$

$$+ (-1)^{K-1} a_{K,1} v_1^{0}\Big)$$

This is (H.8).

We will show that f_{V_1} in (H.9) (and repeated below) is a valid distribution function. Note that $V_1 \geq 0$.

$$f_{V_1}(v) = \frac{1}{\delta \Gamma(K)} \exp(-v_1)\Big(a_{K,K} v_1^{K-1} + (-1)^{K-(K-1)} a_{K,K-1} v_1^{K-2} + \dots$$

$$+ (-1)^{K-k} a_{K,k} v_1^{k-1} + \dots + (-1)^{K-2} a_{K,2} v_1 + (-1)^{K-1} a_{K,1}\Big)$$

This needs to integrate to 1.

$$\int_0^\infty f_{V_1}(v)dv = \int_0^\infty \left[\frac{1}{\delta\Gamma(K)} \exp(-v_1)\Big(a_{K,K}v_1^{K-1}\right.$$
$$+ (-1)^{K-(K-1)}a_{K,K-1}v_1^{K-2} + ...$$
$$+ (-1)^{K-k}a_{K,k}v_1^{k-1} + ...$$
$$+ (-1)^{K-2}a_{K,2}v_1$$
$$\left.+ (-1)^{K-1}a_{K,1}\Big)\right]dv$$

$$\Rightarrow \int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta\Gamma(K)}\left[a_{K,K}\int_0^\infty v_1^{K-1}\exp(-v_1)dv\right.$$
$$+ (-1)^{K-(K-1)}a_{K,K-1}\int_0^\infty v_1^{K-2}\exp(-v_1)dv + ...$$
$$+ (-1)^{K-k}a_{K,k}\int_0^\infty v_1^{k-1}\exp(-v_1)dv + ...$$
$$+ (-1)^{K-2}a_{K,2}\int_0^\infty v_1\exp(-v_1)dv$$
$$\left.+ (-1)^{K-1}a_{K,1}\int_0^\infty \exp(-v_1)v_1dv\right]$$

The integrands are kernels of Gamma densities $Gamma(k,1), k = K,...,1$. $\Gamma(k)$ denotes the gamma function, i.e. $\Gamma(k) = \int_0^\infty x^{k-1}e^{-x}dx$. Note that $\Gamma(k) = (k-1)!$ if k is an integer.

$$\Rightarrow \int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta\Gamma(K)}\left[\Gamma(K)a_{K,K}\right.$$
$$+ (-1)^{K-(K-1)}\Gamma(K-1)a_{K,K-1} + ...$$
$$+ (-1)^{K-k}\Gamma(k)a_{K,k} + ...$$
$$+ (-1)^{K-2}\Gamma(2)a_{K,2}$$
$$\left.+ (-1)^{K-1}\Gamma(1)a_{K,1}\right]$$

$$\Rightarrow \int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\left[\frac{(K-1)!}{(K-1)!}a_{K,K}\right.$$
$$+ (-1)^{K-(K-1)}\frac{(K-2)!}{(K-1)!}a_{K,K-1} + ...$$
$$+ (-1)^{K-k}\frac{(k-1)!}{(K-1)!}a_{K,k} + ... \tag{H.11}$$
$$+ (-1)^{K-2}\frac{(2-1)!}{(K-1)!}a_{K,2}$$
$$\left.+ (-1)^{K-1}\frac{(1-1)!}{(K-1)!}a_{K,1}\right]$$

We will prove by induction that the sum on the RHS of (H.4) equals 1. Recall the definition of the $a_{K,k}$ in (H.5).

For $K = 1$,

$$\int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\frac{(-1)^{1-1}(1-1)!}{(1-1)!}a_{1,1}$$
$$= \frac{1}{\delta}a_{1,1}$$
$$= \frac{1}{\delta}\delta$$
$$= 1$$

So it holds for $K = 1$.

For $K = 2$,

$$\int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\left[(-1)^{2-2}\frac{(2-1)!}{(2-1)!}a_{2,2} + (-1)^{2-1}\frac{(1-1)!}{(2-1)!}a_{2,1}\right]$$

$$= \frac{1}{\delta}\left[a_{2,2} - a_{2,1}\right]$$

$$= \frac{1}{\delta}\left[\delta a_{1,1} - (\delta - 1)a_{1,1}\right]$$

$$= \frac{1}{\delta}\left[\delta^2 - (\delta - 1)\delta\right]$$

$$= \delta - \delta + 1$$

$$= 1$$

And the integral equals 1 for $K = 2$.

For $K = 3$,

$$\int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\left[\frac{(3-1)!}{(3-1)!}a_{3,3}\right.$$

$$+ (-1)^{3-2}\frac{(2-1)!}{(3-1)!}a_{3,2}$$

$$\left. + (-1)^{3-1}\frac{(1-1)!}{(3-1)!}a_{3,1}\right]$$

$$= \frac{1}{\delta}\left[\delta a_{2,2}\right.$$

$$- \frac{1}{2}\left[(2\delta - 2)a_{2,2} + \delta a_{2,1}\right]$$

$$\left. + \frac{1}{2}(\delta - 2)a_{2,1}\right]$$

$$= \frac{1}{\delta}\left[\delta a_{2,2} - \delta a_{2,2} + a_{2,2} - \frac{\delta}{2}a_{2,1} + \frac{\delta}{2}a_{2,1} - a_{2,1}\right]$$

$$= \frac{1}{\delta}\left[a_{2,2} - a_{2,1}\right]$$

$$= \frac{1}{\delta}\left[\delta^2 - (\delta - 1)\delta\right]$$

$$= 1$$

It holds for $K = 3$.

Assume the integral of $f_{V_1}(v)dv$ is 1 for arbitrary K. Meaning we are assuming that

$$\int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\Big[\frac{(K-1)!}{(K-1)!}a_{K,K}$$
$$+ (-1)^{K-(K-1)}\frac{(K-2)!}{(K-1)!}a_{K,K-1} + ...$$
$$+ (-1)^{K-k}\frac{(k-1)!}{(K-1)!}a_{K,k} + ...$$
$$+ (-1)^{K-2}\frac{(2-1)!}{(K-1)!}a_{K,2}$$
$$+ (-1)^{K-1}\frac{(1-1)!}{(K-1)!}a_{K,1}\Big]$$
$$= 1$$

(H.12)

Consider the integral for $K+1$.

$$\int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\Big[\frac{(K+1-1)!}{(K+1-1)!}a_{K+1,K+1}$$
$$+ (-1)^{K+1-K}\frac{(K+1-2)!}{(K+1-1)!}a_{K+1,K+1-1} + ...$$
$$+ (-1)^{K+1-k}\frac{(k-1)!}{(K+1-1)!}a_{K+1,k} + ...$$
$$+ (-1)^{K+1-2}\frac{(2-1)!}{(K+1-1)!}a_{K+1,2}$$
$$+ (-1)^{K+1-1}\frac{(1-1)!}{(K+1-1)!}a_{K+1,1}\Big]$$

$$\Rightarrow \int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\Big[\frac{(K+1-1)!}{(K+1-1)!}\delta a_{K,K}$$
$$+ (-1)^{K+1-K}\frac{(K+1-2)!}{(K+1-1)!}\big((K\delta - K)a_{K,K} + \delta a_{K,K-1}\big) + ...$$
$$+ (-1)^{K+1-k}\frac{(k-1)!}{(K+1-1)!}\big((k\delta - K)a_{K,k} + \delta a_{K,k-1}\big) + ...$$
$$+ (-1)^{K+1-2}\frac{(2-1)!}{(K+1-1)!}\big((2\delta - K)a_{K,2} + \delta a_{K,1}\big)$$
$$+ (-1)^{K+1-1}\frac{(1-1)!}{(K+1-1)!}(\delta - K)a_{K,1}\Big]$$

$$\Rightarrow \int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\Big[\frac{K!}{K!}\delta a_{K,K}$$

$$+ (-1)^{K+1-K}\frac{K!}{K!}\delta a_{K,K} + (-1)^{K+1-K+1}\frac{(K-1)!}{(K-1)!}a_{K,K}+$$

$$(-1)^{K+1-K}\frac{(K-1)!}{K!}\delta a_{K,K-1}$$

$$+ (-1)^{K+1-(K-1)}\frac{(K-1)!}{K!}\delta a_{K,K-1}+$$

$$(-1)^{K+1-(K-2)}\frac{(K-2)!}{(K-1)!}a_{K,K-1} + ...$$

$$+ (-1)^{K+1-k}\frac{k!}{K!}\delta a_{K,k} + (-1)^{K+1-(k-1)}\frac{k!}{(K-1)!}a_{K,k}+$$

$$(-1)^{K+1-k}\frac{(k-1)!}{K!}\delta a_{K,k-1}$$

$$+ ... + (-1)^{K-1}\frac{2!}{K!}\delta a_{K,2} + (-1)^{K-2}\frac{(2-1)!}{(K-1)!}a_{K,2} + (-1)^{K-1}\delta a_{K,1}$$

$$+ (-1)^K\frac{1}{K!}\delta a_{K,1} + (-1)^{K-1}\frac{1}{(K-1)!}a_{K,1}\Big]$$

$$\Rightarrow \int_0^\infty f_{V_1}(v)dv = \frac{1}{\delta}\Big[(-1)^{K-K}\frac{(K-1)!}{(K-1)!}a_{K,K}$$

$$+ (-1)^{K-(K-1)}\frac{(K-2)!}{(K-1)!}a_{K,K-1} + ...$$

$$+ (-1)^{K-k}\frac{(k-1)!}{(K-1)!}a_{K,k} + ...$$

$$+ (-1)^{K-2}\frac{(2-1)!}{(K-1)!}a_{K,2}$$

$$+ (-1)^{K-1}\frac{(1-1)!}{(K-1)!}a_{K,1}\Big]$$

$$=1 \quad \text{(by the induction hypothesis)}$$

We have proved that $\int f_{V_1}(v)dv = 1$ for any K. Therefore $f_{V_1}(v)$ is a valid density function $\forall K$.

Bibliography

Alafchi, B., H. Mahjub, L. Tapak, G. Roshanaei, and M. A. Amirzargar (2021). Two-Stage Joint Model for Multivariate Longitudinal and Multistate Processes , with Application to Renal Transplantation Data. *Journal of Probability and Statistics 2021*, 1–10.

Andersen, P. K. and N. Keiding (2002). Multi-state models for event history analysis. *Statistical Methods in Medical Research 11*(2), 91–115.

Andrinopoulou, E.-R., P. H. C. Eilers, J. J. M. Takkenberg, and D. Rizopoulos (2018). Improved Dynamic Predictions from Joint Models of Longitudinal and Survival Data with Time-Varying Effects Using P-Splines. *Biometrics 74*, 685–693.

Andrinopoulou, E.-R., K. Nasserinejad, R. D. Szczesniak, and D. Rizopoulos (2018). Integrating latent classes in the bayesian shared parameter joint model of longitudinal and survival outcomes.

Andrinopoulou, E.-R., D. Rizopoulos, J. J. Takkenberg, and E. Lesaffre (2013). Joint modeling of two longitudinal outcomes and competing risk data. *Statistics in Medicine 33*, 3167–3178.

Andrinopoulou, E.-R., D. Rizopoulos, J. J. Takkenberg, and E. Lesaffre (2017). Combined dynamic predictions using joint models of two longitudinal outcomes and competing risk data. *Statistical Methods in Medical Research 26*(4), 1787–1801.

Armero, C., A. Forte, H. Perpiñán, M. J. Sanahuja, and S. Agustí (2016). Bayesian joint modeling for assessing the progression of chronic kidney disease in children. *Statistical Methods in Medical Research*, 1–17.

Ayantobo, O. O., Y. Li, and S. Song (2019). Multivariate Drought Frequency Analysis using Four-Variate Symmetric and Asymmetric Archimedean Copula Functions. *Water Resources Management 33*, 103–127.

Baghfalaki, T., M. Ganjali, and R. Hashemi (2014). Bayesian joint modeling of longitudinal measurements and time-to-event data using robust distributions. *Journal of Biopharmaceutical Statistics 24*, 834–855.

Bakoyannis, G., F. Siannisb, and G. Touloumia (2010). Modelling competing risks data with missing cause of failure. *Statistics in Medicine 29*, 3172–3185.

Balentine, C. J., J. Enriquez, W. Fisher, S. Hodges, V. Bansal, S. Sansgiry, N. J. Petersen, and D. H. Berger (2010). Intra-abdominal Fat Predicts Survival in Pancreatic Cancer. *Journal of Gastrointestinal Surgery 14*, 1832–1837.

Basu, S., P. B. Asit, and C. Mukhopadhyay (1999). Bayesian analysis for masked system failure data using non-identical Weibull models. *Journal of Statistical Planning and Inference 78*, 255–275.

Basu, S., A. Sen, and M. Banerjee (2003). Bayesian analysis of competing risks with partially masked cause of failure. *Journal of the Royal Statistical Society. Series C: Applied Statistics 78*, 77–93.

Bayar, M. A., S. Antoun, and E. Lanoy (2017). Statistical approaches for evaluating body composition markers in clinical cancer research. *Expert Review of Anticancer Therapy 17*(4), 311–318.

Bekele, B. N. and Y. Shen (2005). A Bayesian Approach to Jointly Modeling Toxicity and Biomarker Expression in a Phase I/II Dose-Finding Trial. *Biometrics 61*, 344–354.

Blanche, P., C. Proust-Lima, L. Loubère, C. Berr, J.-F. Dartigues, and H. Jacqmin-Gadda (2015). Quantifying and Comparing Dynamic Predictive Accuracy of Joint Models for Longitudinal Marker and Time-to-Event in Presence of Censoring and Competing Risks. *Biometrics 71*, 102–113.

Brown, E. R. (2005). Software. Accessed: 2018-02-05.

Brown, E. R. and J. G. Ibrahim (2003). A Bayesian Semiparametric Joint Hierarchical Model for Longitudinal and Survival Data. *Biometrics 59*(2), 221–228.

Cai, Q., M.-C. Wang, and K. C. G. Chan (2017). Joint modeling of longitudinal, recurrent events and failure time data for survivor's population. *Biometrics 73*, 1150–1160.

Cao, G., D. Todem, L. Yang, and J. P. Fine (2013). Evaluating Statistical Hypotheses Using Weakly-Identifiable Estimating Functions. *Scan Stat Theory Appl 40*(2), 256–273.

Carrière, J. F. (1995). Removing Cancer when it is Correlated with other Causes of Death. *Biometrical Journal 37*, 339–350.

Chen, Y. and Y. Wang (2017). Variable selection for joint models of multivariate longitudinal measurements and event time data. *Statistics in Medicine 36*, 3820–3829.

Craiu, R. V. and B. Reiser (2006). Inference for the dependent competing risks model with masked causes of failure. *Lifetime Data Analysis 12*, 21–33.

Crowder, M. (2012). *Multivariate Survival Analysis and Competing Risks*. CRC Press.

Crowther, M. J., K. R. Abrams, and P. C. Lambert (2013). Joint modeling of longitudinal and survival data. *The Stata Journal 13*(1), 165–184.

Dantan, E., P. Joly, J.-F. Dartigues, and J. Jacqmin-Gadda (2011). Joint model with latent state for longitudinal and multistate data. *Biostatistics 12*(4), 723–736.

De Gruttola, V. and X. M. Tu (1994). Modelling Progression of CD4-Lymphocyte Count and Its Relationship to Survival Time. *Biometrics 50*(4), 1003.

Deslandes, E. and S. Chevret (2010). Joint modeling of multivariate longitudinal data and the dropout process in a competing risk setting: application to ICU data. *Medical Research Methodology 10*(69), 1–13.

Dessie, Z. G., T. Zewotir, H. Mwambi, and D. North (2020). Modelling of viral load dynamics and CD4 cell count progression in an antiretroviral naive cohort: Using a joint linear mixed and multistate Markov model. *BMC Infectious Diseases 20*, 1–14.

Diao, L. and R. J. Cook (2014). Composite likelihood for joint analysis of multiple multistate processes via copulas. *Biostatistics 15*(4), 690–705.

Dil, E. and D. Karasoy (2020). Joint Modeling of a Longitudinal Measurement and Parametric Survival Data with Application to Primary Biliary Cirrhosis Study. *Pakistan Journal of Statistics and Operation Research 16*(1), 295–304.

Dobson, A. and R. Henderson (2003). Diagnostics for Joint Longitudinal and Dropout Time Modeling. *Biometrics 59*, 741–751.

Efron, B. and R. J. Tibshirani (1993). *An Introduction to the Bootstrap*. Chapman & Hall. pages 184-187.

Elashoff, R. M., G. Li, and N. Li (2007). An approach to joint analysis of longitudinal measurements and competing risks failure time data. *Statistics in Medicine 26*(14), 2813–2835.

Elashoff, R. M., G. Li, and N. Li (2017). *Joint Modeling of Longitudinal and Time-to-Event Data*. CRC Press.

Escarela, G. and J. F. Carrière (2003). Fitting competing risks with an assumed copula. *Statistical Methods in Medical Research 12*, 333–349.

Faucett, C. L. and D. C. Thomas (1996). Simultaneously Modelling Censored Survival Data And Repeatedly Measured Covariates: A Gibbs Sampling Approach. *Statistics in Medicine 15*(15), 1663–1685.

Fenech, J. P., H. Vosgha, and S. Shafik (2015). Loan default correlation using an archimedean copula approach: A case for recalibration. *Economic Modelling 47*, 340–354.

Ferrer, L., V. Rondeau, J. Dignam, T. Pickles, H. Jacqmin-Gaddaa, and C. Proust-Lima (2016). Joint modelling of longitudinal and multi-state processes: application to clinical progressions in prostate cancer. *Statistics in Medicine 35*, 3933–3948.

Fine, J. P. and R. J. Gray (1999). A Proportional Hazards Model for the Subdistribution of a Competing Risk. *Journal of the American Statistical Association 94*, 432–445.

Furgal, A. K. C., A. Sen, and J. M. G. Taylor (2019). Review and Comparison of Computational Approaches for Joint Longitudinal and Time-to-Event Models. *International Statistical Review 87*(2), 393–418.

Garcia-Hernandez, A. and D. Rizopoulos (2015). %JM Macro Reference Manual. Online; accessed December 1, 2015.

Genest, C. and J. Nešlehová (2006). Copulas and Copula Models. *Encyclopedia of Environmetrics*, 1–12.

Gould, A. L., M. E. Boye, M. J. Crowther, J. G. Ibrahim, G. Quartey, S. Micallef, and F. Y. Bois (2014). Joint modeling of survival and longitudinal non-survival data: current methods and issues. Report of the DIA Bayesian joint modeling working group. *Statistics in Medicine 34*(14), 2181–2195.

Gronau, Q. F., H. Singmann, and E.-J. Wagenmakers (2020). bridgesampling: An R package for estimating normalizing constants. *Journal of Statistical Software 92*(10), 1–29.

Guo, J., J. Gabry, and B. G. [and Sebastian Weber (2020). rstan: R Interface to Stan. Online; accessed November 1, 2020.

Guo, X. (2003). WinBUGS 1.4 code for joint model (Model XI in Guo and Carlin paper). Online; accessed December 1, 2015.

Guo, X. and B. P. Carlin (2004). Separate and Joint Modeling of Longitudinal and Event Time Data Using Standard Computer Packages. *The American Statistician 58*(1), 16–24.

Han, J., E. H. Slate, and E. A. P. na (2007). Parametric latent class joint model for a longitudinal biomarker and recurrent events. *Statistics in Medicine 26*, 5285–5302.

Hanson, T. E., A. J. Branscum, and W. O. Johnson (2011). Predictive comparison of joint longitudinal-survival modeling : a case study illustrating competing approaches. *Lifetime Data Analysis 17*, 3–28.

Hatfield, L. A., M. E. Boye, and B. P. Carlin (2011). Joint modeling of multiple longitudinal patient-reported outcomes and survival. *J Biopharm Stat. 21*(5), 971–991.

Heckman, J. J. and B. E. Honorè (1989). The identifiability of the competing risks model. *Biometrika 76*(2), 361–372.

Henderson, R., P. Diggle, and A. Dobson (2000, Jan). Joint modelling of longitudinal measurements and event time data. *Biostatistics 1*(4), 465–480.

Hickey, G. L., P. Philipson, and A. Jorgensen (2018). Package 'joineRML': Joint modeling of multivariate longitudinal data and time-to-event outcomes. Accessed: 2018-02-05.

Hickey, G. L., P. Philipson, A. Jorgensen, and R. Kolamunnage-Dona (2016). Joint modelling of time-to-event and multivariate longitudinal outcomes: recent developments and issues. *BMC Medical Research Methodology 16*(117), 1–15.

Ho, N. and X. Nguyen (2016). Convergence rates of parameter estimation for some weakly identifiable finite mixtures. *The Annals of Statistics 44*(6), 2726–2755.

Holcombe, S., B. Enchakalody, P. Zhang, and S. Wang (2016). Morphomics Data Dictionary.

Hong, Y., C. King, Y. Zhang, and W. Q. Meeker (2015). Bayesian Life Test Planning for Log-Location-Scale Family of Distributions. *Journal of Quality Technology 47*(4), 336–350.

Hougaard, P. (1986). A class of multivariate failure time distributions. *Biometrika 73*(3), 671–678.

Hougaard, P. (1999). Multi-state Models: A Review. *Lifetime Data Analysis 5*, 239–264.

Hounsfield, G. N. (1980). Computed medical imaging. *Medical Physics 7*(4), 283–290.

Hseih, F., Y.-K. Tseng, and J.-L. Wang (2006). Joint modeling of survival and longitudinal data: a likelihood approach revisited. *Biometrics 62*, 1037–1043.

Hu, B., L. Li, X. Wanga, and T. Greene (2012). Nonparametric multistate representations of survival and longitudinal data with measurement error. *Statistics in Medicine 31*, 2303–2317.

Hu, W., G. Li, and N. Li (2009). A Bayesian approach to joint analysis of longitudinal measurements and competing risks failure time data. *Statistics in Medicine 28*(11), 1601–1619.

Huang, X., G. Li, R. M. Elashoff, and J. Pan (2010). A general joint model for longitudinal measurements and competing risks survival data with heterogeneous random effects. *Lifetime Data Analysis 17*(1), 80–100.

Huang, X., L. A. Stefanski, and M. Davidian (2009). Latent-Model Robustness in Joint Models for a Primary Endpoint and a Longitudinal Process. *Biometrics 65*(3), 719–727.

Huang, X.-W., W. Wang, and T. Emura (2020). A copula-based Markov chain model for serially dependent event times with a dependent terminal event. *Japanese Journal of Statistics and Data Science*, 1–35.

Huang, Y., G. Dagne, and L. Wu (2011). Bayesian inference on joint models of HIV dynamics for time-to-event and longitudinal data with skewness and covariate measurement errors. *Statistics in Medicine 30*(24), 2930–2946.

Hwang, B. S. (2013). Semiparametric Bayesian Joint Modeling with Applications in Toxicological Risk Assessment. *The Ohio State University Biostatistics Dissertations*, 1–172.

Ibrahim, J. G., H. Chu, and L. M. Chen (2010, Mar). Basic Concepts and Methods for Joint Models of Longitudinal and Survival Data. *Journal of Clinical Oncology 28*(16), 2796–2801.

Jia, X. (2018). Survival analysis using archimedean copulas. Accessed: 2021-04-06.

Kass, R. E. and A. E. Raftery (1995). Bayes Factors. *Journal of the American Statistical Association 90*(430), 773–795.

Kim, S., D. Zeng, L. Chambless, and Y. Li (2012). Joint Models of Longitudinal Data and Recurrent Events with Informative Terminal Event. *Statistics in Biosciences 4*, 262–281.

Król, A., L. Ferrer, J.-P. Pignon, C. Proust-Lima, M. Ducreux, O. Bouché, S. Michiels, and d Virginie Rondeau (2016). Joint model for left-censored longitudinal data, recurrent events and terminal event: Predictive abilities of tumor burden for cancer evolution with application to the FFCD 2000-05 trial. *Biometrics 72*, 907–916.

Lakhal, L., L.-P. Rivest, and B. Abdous (2008). Estimating Survival and Association in a Semicompeting Risks Model. *Biometrics 64*, 180–188.

Lee, L. (1979). Multivariate Distributions having Weibull Properties. *Journal of Multivariate Analysis 9*, 267–277.

Lehmann, A. (2009). Joint modeling of degradation and failure time data. *Journal of Statistical Planning and Inference 139*(5), 1693–1706.

Li, E., N. Wang, and N.-Y. Wang (2007). Joint Models for a Primary Endpoint and Multiple Longitudinal Covariate Processes. *Biometrics 63*(4), 1068–1078.

Li, H. and Y. Lu (2019). Modeling cause-of-death mortality using hierarchical Archimedean copula. *Scandinavian Actuarial Journal 3*, 247–272.

Li, K. and S. Luo (2019). Dynamic predictions in Bayesian functional joint models for longitudinal and time-to-event data: An application to Alzheimer's disease. *Statistical Methods in Medical Research 28*(2), 327–342.

Li, N., R. M. Elashoff, G. Li, and J. Saver (2010). Joint modeling of longitudinal ordinal data and competing risks survival times and analysis of the NINDS rt-PA stroke trial. *Statistica Sinica 29*, 546–557.

Lin, H., B. W. Turnbull, C. E. McCulloch, and E. H. Slate (2002). Latent Class Models for Joint Analysis of Longitudinal Biomarker and Event Process Data: Application to Longitudinal Prostate-Specific Antigen Readings and Prostate Cancer. *Journal of the American Statistical Association 97*(457), 53–65.

Littell, R., G. Milliken, W. Stroup, R. Wolfinger, and S. O. (2006). *SAS for Mixed Models*. SAS Institute.

Liu, L. and X. Huang (2009). Joint analysis of correlated repeated measures and recurrent events processes in the presence of death, with application to a study on acquired immune deficiency syndrome. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 58*(1), 65–81.

Lo, S. M. S. and R. A. Wilke (2010). A copula model for dependent competing risks. *Journal of the Royal Statistical Society. Series C: Applied Statistics 59*, 359–376.

Locke, J. E., J. J. Carr, S. Nair, J. G. Terry, R. D. Reed, G. D. Smith, D. L. Segev, V. Kumar, and C. E. Lewis (2017). Abdominal lean muscle is associated with lower mortality among kidney waitlist candidates. *Clinical Transplantation 31*, 1–8.

Long, J. D. and J. A. Mills (2018). Joint modeling of multivariate longitudinal data and survival data in several observational studies of Huntington's disease. *BMC Medical Research Methodology 18*(183), 1–15.

Lu, J.-C. and Bhattacharyya (1990). Some New Constructions of Bivariate Weibull Models. *Annals of the Institute of Statistical Mathematics 42*(3), 543–559.

Lu, J.-C. and G. K. Bhattacharyya (1991). Inference procedures for bivariate exponential model of Gumbel. *Statistics & Probability Letters 12*, 37–50.

Malehi, A. S., E. Hajizadeh, K. A. Ahmadi, and P. Mansouri (2015). Joint modelling of longitudinal biomarker and gap time between recurrent events: copula-based dependence. *Journal of Applied Statistics 42*(9), 1931–1945.

Marchenko, Y. (2016). Joint modeling of longitudinal and survival data. Online; accessed November 1, 2016.

Mauff, K., E. Steyerberg, I. Kardys, E. Boersma, and D. Rizopoulos (2020). Joint models with multiple longitudinal outcomes and a time-to-event outcome: a corrected two-stage approach. *Statistics and Computing 30*, 999–1014.

Mccrink, L. M., A. H. Marshall, and K. J. Cairns (2013). Advances in Joint Modelling: A Review of Recent Developments with Application to the Survival of End Stage Renal Disease Patients. *International Statistical Review 81*(2), 249–269.

McDuffie, L. A. and R. D. Aufforth (2016). Adrenocortical carcinoma: modern management and evolving treatment strategies. *International Journal of Endocrine Oncology 3*(2), 161–174.

Meng, X.-L. and W. H. Wong (1996). Simulating ratios of normalizing constants via a simple identity: A theoretical exploration. *Statistica Sinica 6*, 831–860.

Miller, B. S., K. M. Ignatoski, S. Daignault, C. L. M. Doherty, P. G. Gauger, G. D. Hammer, S. C. Wang, G. M. Doherty, and T. U. of Michigan Analytical Morphomics Group (2012). Worsening Central Sarcopenia and Increasing Intra-Abdominal Fat Correlate with Decreased Survival in Patients with Adrenocortical Carcinoma. *World Journal of Surgery 36*, 1509–1516.

Mukhopadhyay, C. and S. Basu (2007). Bayesian analysis of masked series system lifetime data. *Communications in Statistics - Theory and Methods 36*, 329–348.

Mukhopadhyay, C. and S. Roy (2016). Bayesian accelerated life testing under competing log-location-scale family of cause of failure. *Computational Statistics 31*, 89–119.

Musoro, J. Z., R. B. Geskus, and A. H. Zwinderman (2015). A joint model for repeated events of different types and multiple longitudinal outcomes with application to a follow-up study of patients after kidney transplant. *Biometrical Journal 57*, 185–200.

Musoro, J. Z., A. H. Zwinderman, A. Abu-Hanna, R. Bosman, and R. B. Geskus (2018). Dynamic prediction of mortality among patients in intensive care using the sequential organ failure assessment (SOFA) score: a joint competing risk survival and longitudinal modeling approach. *Statistica Neerlandica 72*, 34–47.

Mwanyekange, J., S. M. Mwalili, and O. Ngesa (2019). Bayesian joint models for longitudinal and multi-state survival data. *International Journal of Statistics and Probability 8*(2), 34–55.

Papageorgiou, G., K. Mauff, A. Tomer, and D. Rizopoulos (2019). An Overview of Joint Modeling of Time-to-Event and Longitudinal Outcomes. *Annual Review of Statistics and Its Application 6*(1), 223–240.

Park, K. Y. and P. Qiu (2014). Model selection and diagnostics for joint modeling of survival and longitudinal data with crossing hazard rate functions. *Statistics in Medicine 33*, 4532–4546.

Philipson, P., I. Sousa, P. Diggle, P. Williamson, R. Kolamunnage-Dona, and R. Henderson (2012). Package 'joineR': Joint modeling of repeated measurements and time-to-event data. Accessed: 2016-10-12.

Pinheiro, J., D. Bates, S. DebRoy, D. Sarkar, EISPACK, S. Heisterkamp, and B. Van Willigen (2016). Package 'nlme': Linear and Nonlinear Mixed Effects Models. Accessed: 2016-10-12.

Proust-lima, C., J.-f. Dartigues, and H. Jacqmin-Gadda (2015). Joint modeling of repeated multivariate cognitive measures and competing risks of dementia and death : a latent process and latent class approach. *Statistics in Medicine*.

Proust-Lima, C., V. Philips, and B. Liquet (2016). Estimation of Extended Mixed Models Using Latent Classes and Latent Processes: The R Package lcmm. Online; accessed November 1, 2016.

Proust-Lima, C., M. Se'ne, J. M. Taylor, and H. Jacqmin-Gadda (2014). Joint latent class models for longitudinal and time-to-event data: A review. *Statistical Methods in Medical Research 23*(1), 74–90.

Proust-Lima, C. and J. M. G. Taylor (2009). Development and validation of a dynamic prognostic tool for prostate cancer recurrence using repeated measures of posttreatment PSA: a joint modeling approach. *Biostatistics 10*(3), 535–549.

Ren, X., J. Wang, and S. Luo (2019). Dynamic prediction using joint models of longitudinal and recurrent event data: a Bayesian perspective. *Biostatistics & Epidemiology*.

Rizopoulos, D. (2010). JM: An R Package for the Joint Modelling of Longitudinal and Time-to-Event Data. *Journal of Statistical Software 35*(9), 1–33.

Rizopoulos, D. (2012). *Joint models for longitudinal and time-to-event data: with applications in R*. CRC Press.

Rizopoulos, D. (2016a). Package 'JM': Joint modeling of Longitudinal and Survival Data. Accessed: 2016-09-30.

Rizopoulos, D. (2016b). Package 'JMbayes': Joint Modeling of Longitudinal and Time-to-Event Data under a Bayesian Approach. Accessed: 2016-10-30.

Rizopoulos, D. (2016c). The R Package JMbayes for Fitting Joint Models for Longitudinal and Time-to-Event Data Using MCMC. *Journal of Statistical Software 72*(7), 1–46.

Rizopoulos, D. and P. Ghosh (2011). A Bayesian semiparametric multivariate joint model for multiple longitudinal outcomes and a time-to-event. *Statistics in Medicine 30*(12), 1366–1380.

Rizopoulos, D., L. A. Hatfield, B. P. Carlin, and J. J. M. Takkenberg (2014, Feb). Combining Dynamic Predictions From Joint Models for Longitudinal and Time-to-Event Data Using Bayesian Model Averaging. *Journal of the American Statistical Association 109*(508), 1385–1397.

Rizopoulos, D., G. Verbeke, and E. Lesaffre (2009). Fully exponential Laplace approximations for the joint modelling of survival and longitudinal data. *Journal of the Royal Statistical Society, Series B 71*, 637–654.

Rizopoulos, D., G. Verbeke, and G. Molenberghs (2010). Multiple Imputation-Based Residuals and Diagnostic Plots for Joint Models of Longitudinal and Survival Outcomes. *Biometrics 66*, 20–29.

Rondeau, V., J. R. Gonzalez, Y. Mazroui, A. Mauguen, A. Krol, A. Diakite, A. Laurent, and M. Lopez (2017). General Frailty Models: Shared, Joint and Nested Frailty Models (Package 'frailtypack'). Online; accessed August 29, 2017.

Rondeau, V., Y. Mazroui, and J. R. Gonzalez (2012). frailtypack: An R Package for the Analysis of Correlated Survival Data with Frailty Models Using Penalized Likelihood Estimation or Parametrical Estimation. *Journal of Statistical Software 47*(4), 1–28.

Rotolo, F., C. Legrand, and I. V. Keilegom (2013). A simulation procedure based on copulas to generate clustered multi-state survival data. *Computer Methods and Programs in Biomedicine 109*, 305–312.

Rouanet, A., P. Joly, J.-F. Dartigues, C. Proust-Lima, and H. Jacqmin-Gadda (2016). Joint Latent Class Model for Longitudinal Data and Interval-Censored Semi-Competing Events : Application to Dementia. *Biometrics 72*, 1123–1135.

Rue, M., E.-R. Andrinopoulou, D. Alvares, C. Armero, A. Forte, and L. Blanch (2017). Bayesian joint modeling of bivariate longitudinal and competing risks data : An application to study patient-ventilator asynchronies in critical care patients. *Biometrical Journal 59*, 1184–1203.

Sabel, M. S., J. Lee, A. Wang, C. Lao, S. Holcombe, and S. Wang (2015). Morphomics Predicts Response to Ipilimumab in Patients With Stage IV Melanoma. *Journal of Surgical Oncology 112*, 333–337.

Sabel, M. S., M. Terjimanian, A. S. C. Conlon, K. A. Griffith, A. M. Morris, M. W. Mulholland, M. J. Englesbe, S. Holcombe, and S. C. Wang (2013). Analytic Morphometric Assessment of Patients Undergoing Colectomy for Colon Cancer. *Journal of Surgical Oncology 108*, 169–175.

SAS Institute (2015a). Example 61. 5 Failure Time and Frailty Model: SAS User's Guide: PROC NLMIXED. Online; accessed December 1, 2015.

SAS Institute (2015b). The NLMIXED Procedure (in SAS 9.2 Help and Documentation). Online; accessed December 1, 2015.

Savu, C. and M. Trede (2010). Hierarchies of Archimedean copulas. *Quantitative Finance 10*(3), 295–304.

Schabenberger, O. (2004). Model XI in Carlin and Guo with PROC NLMIXED in SAS. Online; accessed December 1, 2015.

Schwarz, M., G. Jongbloed, and I. Van Keilegom (2013). On the identifiability of copulas in bivariate competing risks models. *The Canadian Journal of Statistics 41*(2), 291–303.

Sen, A., M. Banerjee, Y. Li, and A.-M. Noone (2010). A Bayesian approach to competing risks analysis with masked cause of death. *Statistics in Medicine 29*, 1681–1695.

Shaked, M. and J. G. Shanthikumar (1987). The Multivariate Hazard Construction. *Stochastic Processes and their Applications 24*, 241–258.

Shih, J.-H. and T. Emura (2018). Likelihood-based inference for bivariate latent failure time models with competing risks under the generalized FGM copula. *Computational Statistics 33*, 1293–1323.

shou Ko, F. (2019). An issue of identifying longitudinal biomarkers for competing risks data with masked causes of failure considering frailties model. *Statistical Methods in Medical Research 0*, 1–14.

Singal, A. G., P. Zhang, A. K. Waljee, L. Ananthakrishnan, N. D. Parikh, P. Sharma, P. Barman, V. Krishnamurthy, L. Wang, S. C. Wang, , and G. L. Su (2016). Body Composition Features Predict Overall Survival in Patients With Hepatocellular Carcinoma. *Clinical and Translational Gastroenterology 7*(172), 1–8.

Singh, V. P. and L. Zhang (2007). IDF Curves Using the Frank Archimedean Copula. *Journal of Hydrologic Engineering 12*(6), 651–662.

Stata Corp LP (2015). Stata Structural Equation Modeling Reference Manual. Version 14, Online; accessed November 1, 2016.

Stidham, R. W., A. K. Waljee, N. M. Day, C. L. Bergmans, K. M. Zahn, P. D. R. Higgins, S. C. Wang, and G. L. Su (2015). Body Fat Composition Assessment Using Analytic Morphomics Predicts Infectious Complications After Bowel Resection in Crohn's Disease. *Inflammatory Bowel Diseases 21*(6), 1306–1313.

Sudell, M. (2018). joineRmeta. Accessed: 2018-04-05.

Sudell, M., R. Kolamunnage-Dona, and C. T. Smith (2018). joineRmeta: Joint Modelling for Meta-Analytic (Multi-Study) Data. Accessed: 2018-04-05.

Sudell, M., R. Kolamunnage-Dona, and C. Tudur-Smith (2016). Joint models for longitudinal and time-to-event data: a review of reporting quality with a view to meta-analysis. *BMC Medical Research Methodology 16*(168), 1–11.

Sun, J., J. D. Herazo-Maya, P. L. Molyneaux, T. M. Maher, N. Kaminski, and H. Zhao (2019). Regularized Latent Class Model for Joint Analysis of High-Dimensional Longitudinal Biomarkers and a Time-to-Event Outcome. *Biometrics 75*, 69–77.

Suresh, K., J. M. G. Taylor, and A. Tsodikov (2019). A Gaussian copula approach for dynamic prediction of survival with a longitudinal biomarker. *Biostatistics*.

Sweeting, M. J. and S. G. Thompson (2011). Joint modelling of longitudinal and time-to-event data with application to predicting abdominal aortic aneurysm growth and rupture. *Biometrical Journal 53*(5), 750–763.

Tang, A.-M. and N.-S. Tang (2015). Semiparametric Bayesian inference on skew-normal joint modeling of multivariate longitudinal and survival data. *Statistics in Medicine 34*, 824–843.

Tao, Y., J. Liu, Z. Li, J. Lin, T. Lu, and F. Yan (2013). Dose-Finding Based on Bivariate Efficacy-Toxicity Outcome Using Archimedean Copula. *PLoS ONE 8*(11), 1–6.

Taylor, J. M. G., Y. Park, D. P. Ankerst, C. Proust-Lima, S. Williams, L. Kestin, K. Bae, T. Pickles, and H. Sandler (2013). Real-Time Individual Predictions of Prostate Cancer Recurrence Using Joint Models. *Biometrics 69*, 206–213.

Therneau, T. M. and T. Lumley (2016). Package 'survival': Survival Analysis. Accessed: 2016-10-12.

Thomasson, A. M. (2012). A Joint Longitudinal-Survival Model with Possible Cure: An Analysis of Patient Outcomes on the Liver Transplant Waiting List. *Publicly Accessible Penn Dissertations*, 1–84.

Tseng, Y.-K., F. Hsieh, and J.-L. Wang (2005). Joint modelling of accelerated failure time and longitudinal data. *Biometrika 92*(3), 587–603.

Tsiatis, A. (1975). A Nonidentifiability Aspect of the Problem of Competing Risks. *PNAS 72*(1), 20–22.

Tsiatis, A. A. and M. Davidian (2004). Joint modeling of longitudinal and time-to-event data: an overview. *Statistica Sinica 14*, 809–834.

Tsiatis, A. A., V. Degruttola, and M. S. Wulfsohn (1995). Modeling the Relationship of Survival to Longitudinal Data Measured with Error. Applications to Survival and CD4 Counts in Patients with AIDS. *Journal of the American Statistical Association 90*(429), 27–37.

Wang, A. (2012). On the nonidentifiability property of Archimedean copula models under dependent censoring. *Statistics and Probability Letters 82*, 621–625.

Wang, A. (2014). Properties of the marginal survival functions for dependent censored data under an assumed Archimedean copula. *Journal of Multivariate Analysis 129*, 57–68.

Wang, A., K. Chandra, R. Xu, and J. Sun (2015). The Identifiability of Dependent competing Risks Models Induced by Bivariate Frailty Models. *Scandinavian Journal of Statistics 42*, 427–437.

Wang, C., J. M. Vainshtein, M. Veksler, P. E. Rabban, J. A. Sullivan, S. C. Wang, A. Eisbruch, and S. Jolly (2016). Investigating the clinical significance of body composition changes in patients undergoing chemoradiation for oropharyngeal cancer using analytic morphomics. *SpringerPlus 5*(429), 1–9.

Wang, C.-P. and M. Ghosh (2000). Bayesian Analysis of Bivariate Competing Risks Models. *The Indian Journal of Statistics 62*, 388–401.

Wang, W., W. Wang, T. H. Mosley, and M. E. Griswold (2017). A SAS macro for the joint modeling of longitudinal outcomes and multiple competing risk dropouts. *Computer Methods and Programs in Biomedicine 138*, 23–30.

Wang, Y. and J. M. G. Taylor (2001). Jointly Modeling Longitudinal and Event Time Data With Application to Acquired Immunodeficiency Syndrome. *Journal of the American Statistical Association 96*(455), 895–905.

Williamson, P., R. Kolamunnage-Dona, P. Philipson, and A. Marson (2008). Joint modelling of longitudinal and competing risks data. *Statistics in Medicine 27*(30), 6426–6438.

Wu, B.-H., H. Michimae, and T. Emura (2020). Meta-analysis of individual patient data with semi-competing risks under the Weibull joint frailty–copula model. *Computational Statistics 35*, 1525–1552.

Wu, C., L. Li, and R. Li (2019). Dynamic Prediction of Competing Risk Events using Landmark Sub-distribution Hazard Model with Multiple Longitudinal Biomarkers. *Biometrics*. Online https://arxiv.org/pdf/1906.05647.pdf; accessed September 6, 2019.

Wu, M., Y. Shi, and C. Zhang (2017). Statistical analysis of dependent competing risks model in accelerated life testing under progressively hybrid censoring using copula function. *Communication in Statistics: Simulation and Computation 46*(5), 4004–4017.

Wulfsohn, M. S. and A. A. Tsiatis (1997). A Joint Model for Survival and Longitudinal Data Measured with Error. *Biometrics 53*(1), 330–339.

Xiao, J., V. C. Mazurak, T. A. Olobatuyi, B. J. Caan, and C. M. Prado (2018). Visceral adiposity and cancer survival: a review of imaging studies. *European Journal of Cancer Care 27*, 1–9.

Yang, L., M. Yu, and S. Gao (2016). Joint models for multiple longitudinal processes and time-to-event outcome. *Statistical Computation and Simulation 86*, 3682–3700.

Yu, M., N. J. Law, J. M. Taylor, and H. M. Sandler (2004). Joint Longitudinal-Survival-Cure Models and their Application to Prostate Cancer. *Statistica Sinica 14*, 835–862.

Yuen, H. P. and A. Mackinnon (2016). Performance of joint modelling of time-to-event data with time-dependent predictors: an assessment based on transition to psychosis data. *PeerJ 4*, 1–20.

Zhang, D., M.-H. Chen, J. G. Ibrahim, M. E. Boye, P. Wang, and W. Shen (2014). Assessing Model Fit in Joint Models of Longitudinal and Survival Data with Applications to Cancer Clinical Trials. *Statistics in Medicine 33*(27), 4715–4733.

Zhang, D., M.-H. Chen, J. G. Ibrahim, M. R. Boye, and W. Shen (2016). JMfit: A SAS Macro for Joint Models of Longitudinal and Survival Data. *Journal of Statistical Software 71*(3), 1–24.

Zhang, H., J. Zhang, and M. Kuwano (2012). An integrated model of tourists' time use and expenditure behaviour with self-selection based on a fully nested Archimedean copula function. *Tourism Management 33*(6), 1562–1573.

Zheng, M. and J. P. Klein (1995). Estimates of marginal survival for dependent competing risks based on an assumed copula. *Biometrika 1*, 127–138.