Joint Models with Splines in the Longitudinal Submodel

by
Victoria P. Silverman

A Thesis
presented to
The University of Guelph

In partial fulfilment of requirements
for the degree of
Master of Science
in
Mathematics & Statistics

Guelph, Ontario, Canada

ABSTRACT

JOINT MODELS WITH SPLINES
IN THE LONGITUDINAL SUBMODEL

Victoria Silverman
University of Guelph, 2022

Advisors:
Dr. Julie Horrocks
Dr. Gerarda Darlington

Joint models are used to model data that has both time-to-event and longitudinal components. It is not always reasonable to assume a linear trajectory for longitudinal data, especially data from applications in the biological and medical fields. A spline is a good way to introduce flexibility to account for the non-linearity that is present. This thesis assesses the use of quadratic and cubic B-splines within the longitudinal submodel of a joint model and compares them to a joint model with a longitudinal submodel that only has linear terms. These methods were demonstrated on two datasets from the field of medicine. A simulation study was also conducted to compare three models that employ B-splines in the longitudinal submodel to the traditional linear longitudinal submodel in a joint model. The results of the simulation study suggest that the introduction of B-splines into the longitudinal model marginally impacts results. However, one must be careful not to incorporate too high a degree of B-splines to not over-fit the data. Overall, splines can be a valuable way to incorporate flexibility into joint models.

Dedication

In memory of John Aaron Sussman, whose light shone bright but was extinguished too soon.

Acknowledgements

I would like to thank my advisors Dr. Julie Horrocks and Dr. Gerarda Darlington for their continued guidance and support throughout my masters program. Next, I would like to thank Dr. Zeny Feng for agreeing to be on my examination committee and for taking the time to read my thesis and provide valuable feedback and suggestions. In addition, I would like to thank Susan McCormick for always being able to answer my questions regarding the administrative aspects of the program.

Finally, I would like to thank my parents, Deborah & Michael Silverman, and my sisters, Meredith & Jillian Silverman, for always being there to answer my calls home and for supporting me through my studies. I would not have been able to do this or be where I am without all of you! I would also like to thank my grandparents, Eva & David Sussman and Netta & the late David Silverman, for always being proud of me and my accomplishments.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Data with a time-to-event and a longitudinal component often arise naturally in several fields such as biology or medicine. It is not always correct to assume that the longitudinal component has a linear trend. For example, a biomarker, such as a component of an individual's blood, can fluctuate both up and down throughout a person's lifetime and during the duration of a study. With that in mind, not taking non-linearity into account in the analysis can impact results. Therefore, there is a need to incorporate flexibility into models used to include non-linear aspects in the statistical analyses. This thesis will use the concept of splines. Splines are piece-wise polynomial functions where at points called "knots", the locally fitted polynomials are joined together [Perperoglou et al., 2019]. This thesis will use basis splines, otherwise known as B-splines, as an attempt to account for the non-linearity of this type of data.

There are several packages that can be used to fit joint models in the statistical software R [R Core Team, 2022]. Some of these explored in this thesis include JM [Rizopoulos, 2010], joineR [Philipson et al., 2018] and joineRML [Hickey et al., 2018]. Ultimately, the joint models in Chapter 3 and Chapter 4 will be fit using only functions from the package JM [Rizopoulos, 2010]. To fit the longitudinal submodel the function lme from the package nlme [Pinheiro et al., 2022] is used. This function allows for the use of B-splines.

Splines have the flexibility to follow the data and be of any degree desired. Typically splines of degree three (cubic) or degree four (quartic) are used [Hastie et al., 2017]. The function bs from the package splines [R Core Team, 2022] which is used to calculate the B-splines in this thesis, have only boundary knots at either end of the data as the default [James et al., 2017]. Additional knots may be specified to increase the flexibility of the spline. This can be done through directly specifying the location of the knots or by specifying the degrees

of freedom and then the function will choose the knots at uniformly distributed quantiles [James et al., 2017]. These are referred to as interior knots. In this thesis, both cubic and quadratic models are implemented as well as a cubic B-spline with additional interior knots within the longitudinal submodel.

This thesis is organized as follows: a general background of joint models and their components (the longitudinal and survival submodels) are explained in Chapter 2. In addition, the theory behind B-splines and the software that can implement them are also explained in Chapter 2. Three types of longitudinal submodels will be applied to two datasets in Chapter 3. The first dataset comes from a study of bipolar disorder [Duffy et al., 2007] and the second dataset comes from the Framingham Heart study [Mahmood et al., 2014]. In Chapter 4, the results of a simulation study are presented. Lastly, conclusions and suggestions for future research are explored in Chapter 5. The code for the analyses conducted in this thesis are available in the appendices.

Chapter 2

Methodology

2.1 Joint Models for Longitudinal and Survival Data

In this chapter, the topic of joint models, specifically joint models for longitudinal and survival data, will be introduced. These models are useful tools in data analysis when the goal is to model the association between time dependent covariates and survival [Rizopoulos, 2012]. Joint models use both the covariates available and the survival data simultaneously, while also minimizing bias as would be seen with using the traditional Cox model when the longitudinal covariate is measured infrequently and with measurement error [Wulfsohn and Tsiatis, 1997]. Joint models have two components, the longitudinal submodel and the survival submodel, which will be be discussed in the next few sections.

2.1.1 Longitudinal Submodel

Let, $y_i(t)$ be the observed value of the longitudinal variable for an individual $i = 1, ..., n$ at time t, which is comprised of the unobserved mean or smoothed trajectory ($m_i(t)$) and the error (ϵ_i(t)) for $i = 1, 2, \ldots, n$ [Rizopoulos, 2012]. The error is assumed to be normally distributed with mean 0 and variance σ^2. A linear mixed effects model will be used such that

$$\begin{aligned} y_i(t) &= m_i(t) + \epsilon_i(t) \\ m_i(t) &= x_i^T(t)\beta + z_i^T(t)b_i \\ b_i &\sim \mathcal{N}(0, \sigma_b^2) \\ \epsilon_i(t) &\sim \mathcal{N}(0, \sigma^2). \end{aligned} \tag{2.1}$$

In this model, β is a $p \times 1$ vector of fixed effects parameters and b_i is a $q \times 1$ vector of random effects, therefore, $x_i(t)$, and $z_i(t)$ are the design matrices of size $p \times 1$ and $q \times 1$ for the fixed and random effects for individual i, respectively [Rizopoulos, 2012]. The random effects are assumed to be normally distributed with mean 0 and variance matrix σ_b^2.

The model above is linear in β, yet it is important to have an accurate model for the longitudinal data. Not always does the longitudinal data have a linear trajectory. In fact, it can have a highly non-linear trajectory. Therefore, in this situation it is suggested to incorporate flexible representations for $x_i(t)$ and $z_i(t)$ such as splines [Rizopoulos, 2012]. Splines will be discussed further in section 2.2.2

2.1.2 Survival Submodel

The second component of the joint model is the survival submodel. The survival submodel requires that the complete longitudinal history of an individual, $\mathcal{M}_i = m_i(r), 0 \leq r < t$, be estimated [Rizopoulos, 2012]. The hazard function for individual i used for the survival submodel is defined as [Rizopoulos, 2012],

$$h_i(t|\mathcal{M}_i(t), w_i) = h_0(t) \exp\{\gamma^T w_i + \alpha m_i(t)\}, \ t > 0. \tag{2.2}$$

Here, $h_0(t)$ is the baseline hazard function, α is the parameter that quantifies the effect of the longitudinal component [Rizopoulos, 2012], and $m_i(t)$ is estimated from the longitudinal submodel defined in section 2.1.1. The estimated $m_i(t)$ is the term that inherently links the survival and longitudinal components of the joint model. The w_i is a vector of time fixed covariates and γ is the vector of corresponding regression coefficients [Rizopoulos, 2012]. The assumed baseline hazard function $h_0(\cdot)$ must be chosen along with the software to implement model fitting. It is clear from Equation 2.2 that the complete history is required, since the value of $m_i(t)$ must be known for all $t > 0$. More detail on the functions and packages used in this thesis are presented in section 2.2.1.

2.2 Non Linear Models for Longitudinal Data

As discussed previously, it is important to have an accurate estimate of the longitudinal history and some longitudinal data can have non linear trajectories. It is suggested that flexible representations for the design matrices in the longitudinal submodel be incorporated

to account for this non-linearity [Rizopoulos, 2012]. One such representation used in this thesis is the application of splines, specifically B-splines, which are discussed in section 2.2.2.

2.2.1 Software for Joint Models

There are several packages in the software R [R Core Team, 2022] which allow for joint longitudinal and survival modelling: joineRML [Hickey et al., 2018], joineR [Philipson et al., 2018] and JM [Rizopoulos, 2010] can fit these models. The assumed baseline hazard function $h_0(\cdot)$ must be selected leading to a choice of software to implement model fitting. In joineRML, the baseline hazard function is left unspecified [Hickey et al., 2018] and this is also the case in JoineR [Williamson et al., 2008]. However, in JM the baseline hazard function can be specified by the user [Rizopoulos, 2012]. The choices for a baseline hazard in JM are a Weibull baseline hazard function, a piecewise constant baseline hazard function and a log baseline hazard function that is approximated using B-splines [Rizopoulos, 2010]. It should be noted that leaving the baseline hazard model unspecified can lead to underestimation of standard errors for the regression coefficients [Rizopoulos, 2012; Hsieh et al., 2006].

JM has the function jointModel which requires the user to create a longitudinal mixed effects model object, which can be done using the function lme from the package nlme [Pinheiro et al., 2022] and a Cox model for the survival submodel, which can be done using the function coxph from the package survival [Therneau, 2021]. The function also has an argument method which is where the user specifies the type of baseline hazard function ($h_0(\cdot)$) to be used. Ultimately, the data analysis in Chapter 3 and the simulation study in Chapter 4 use the function jointModel from the package JM [Rizopoulos, 2010]. Additionally, in the data analysis section of this thesis, the baseline hazard used will be based on a Weibull distribution. For the simulation study in Chapter 4 the baseline hazard will be based on an exponential distribution.

2.2.2 Splines

As mentioned earlier, joint models include a submodel for the longitudinal data, which models the longitudinal data as a function of time. It is not always reasonable to assume a linear trajectory over time for longitudinal data, especially data from applications in the biological and medical fields. Splines are a good way to introduce flexibility. Splines are piece-wise polynomial functions where at points called "knots" the locally fitted polynomials are joined together [Perperoglou et al., 2019]. Splines have the flexibility to follow the data

and to be of any degree desired, and usually in practice, they are of degree 3 or 4 which are cubic or quartic splines [Hastie et al., 2017]. Cubic splines are preferred because they are the lowest degree where the break points are not visible [Hastie et al., 2017].

Regression splines are the types of splines that will be used in this thesis. The first task in creating a regression spline is to define a set of K knots $\tau_1 < ... < \tau_K$ in the range of the data. The knots will divide the data into $K + 1$ intervals in which a polynomial of degree at most d will be estimated within each interval [James et al., 2017]. The knots will join the polynomial curves together allowing the data to be represented with piece-wise curves which will better fit the data than a single curve. These polynomials are required to join continuously at each knot [James et al., 2017]. The more knots introduced, the more flexible the spline will be. However, the bias-variance trade-off comes into play: adding too many knots increases the variance, while having too few increases the bias [Perperoglou et al., 2019]. It is important to choose a reasonable number of knots to minimize both bias and variance.

For a cubic spline, the polynomials used in the piece-wise function for the spline will be of degree at most 3 [Johnson, 2013]. A cubic spline can be defined using basis functions ϕ_s, where $1 \leq s \leq K + 3$ and K is the number of knots. In this thesis, regression splines will be used to model the longitudinal response, y, over time, t. The cubic regression spline can be represented as [James et al., 2017].

$$y(t) = \eta_0 + \eta_1\phi_1(t) + \eta_2\phi_2(t) + ... + \eta_{K+3}\phi_{K+3}(t) + \epsilon(t). \tag{2.3}$$

where ϵ represents the error term. The coefficients of the spline, denoted in this case with $\eta_s, 1 \leq s \leq K + 3$, can be estimated using ordinary least squares [James et al., 2017]. A simple set of basis functions is X, X^2, X^3 and a truncated power basis function for each knot ξ_k, is $f(t, \xi_k)$, for $k = 1, 2, ..., K$ [James et al., 2017], where

$$f(t, \xi_k) = (t - \xi_k)^3_+ = \begin{cases} (t - \xi_k)^3 & \text{if } t > \xi_k \\ 0 & \text{otherwise.} \end{cases} \tag{2.4}$$

In total, $K + 4$ regression coefficients are estimated [James et al., 2017].

B-splines are comprised of an orthonormal basis of recursive functions [Chaudhuri, 2019]. These splines require an extension of the sequence of knots for the cubic spline. First let the boundary knots be ξ_0 and ξ_{K+1}, where $\xi_0 \leq \xi_1$ and $\xi_K \leq \xi_{K+1}$ [Hastie et al., 2017]. Next set $\phi_{l,d}(t)$, the l^{th} B-spline basis function of order d, for the knot-sequence $\tau = \tau_1, ..., \tau_{K+2D}$,

where $d \leq D$. The knots $\tau_1, ...\tau_D$ are outside of the boundary knots and are usually set to be equal to the boundary knot ξ_0. Similarly, $\tau_{K+D+1}, ..., \tau_{K+2D}$ are also outside of the boundary and are usually set to be equal to ξ_{K+1} [Hastie et al., 2017]. The interior knots are set to be $\xi_j = \tau_{j+d}$ for $j = 1, ..., K$. The l^{th} basis function of order d can be represented using a recursive framework [Hastie et al., 2017] such that

$$\phi_{l,1}(t) = \begin{cases} 1 & \text{if } \tau_l \leq t \leq \tau_{l+1} \\ 0 & \text{otherwise} \end{cases} \tag{2.5}$$

for $l = 1, ..., K + 2D - 1$

$$\phi_{l,d}(t) = \frac{t - \tau_i}{\tau_i + d - 1 - \tau_l}\phi_{l,d-1}(t) + \frac{\tau_{l+d} - t}{\tau_{l+d} - \tau_{l+1}}\phi_{l+1,d-1}(t), \tag{2.6}$$

for $l = 1, ..., K + 2D - d$.

The recursion formula above can generate any dimension of B-spline basis functions. For a cubic B-spline, take D to be 4, and use Equations 2.5 and 2.6 to generate $\phi_{l,4}$ with $l = 1, ..., K + 4$ which creates $K + 4$ cubic B-spline basis functions necessary for the knot sequence ξ [Hastie et al., 2017].

Splines can be estimated using the splines package in R [R Core Team, 2022], and this package will be used to apply splines in this thesis. The type used will be B-splines and they can be applied using the function bs().

Chapter 3

Data Analysis

3.1 Bipolar Data Set

The first dataset utilized to demonstrate the methods mentioned previously is from a longitudinal study of individuals and their offspring with bipolar disorder (BD). This dataset was provided by Dr. Anne Duffy at Queen's University. BD usually begins in adolescence or early adulthood [McCormick et al., 2015]. While offspring of individuals have an increased risk of BD, the majority of them do not go on to be diagnosed with BD [Goldstein et al., 2010]. The study follows individuals who are at high-risk for BD and have exactly one parent who is diagnosed through the Schedule for Affective Disorders and Schizophrenia - Lifetime version (SADS-L) interviews and Diagnostic and Statistical Manual of Mental Disorders, 4th edition (DSM-IV) [American Psychiatric Association, 1994] diagnostic criteria [Duffy et al., 2007]. The study uses a diagnosis from the DSM [American Psychiatric Association, 1994] as the event of interest.

3.1.1 Introduction and Exploratory Analysis

Initially, the dataset consisted of 305 individuals from 121 families. The outcome of interest is the time to diagnosis of BD, major depressive disorder (MDD) or schizoaffective disorder. The time-varying longitudinal covariate available is a measurement of anxiety using the Hamilton Anxiety (HAMA) scale [Hamilton, 1959], which has a range of 0 (low) to 56 (high). There are 10 time-independent variables including: sex, socioeconomic status (SES) on a Hollingshead Scale [Hollingshead, 2011], an indicator for whether the parent of the individual responded to lithium treatment (LITHRESP), the onset age of the parent's diagnosis

(PARONSAGE), and some measurements for sleep assessment which will not be included in this thesis.

Only individuals that had at least one HAMA measurement were included. This led to the removal of 58 individuals, making the number of individuals in the dataset equal to 247. Next, only individuals who had a diagnosis after the first recorded HAMA score were included. As a result, the number of individuals in the dataset was 186 individuals from 93 families. The family structure of individuals was ignored in this thesis because current R functions cannot handle it. Clustering in joint models was studied by Gaudet [2021].

Among the 186 individuals, 19 experienced the event of interest and 167 were censored. Table 3.1 contains descriptive statistics for the time-independent variables sex, lithium response of the parent with BD and the parent's SES. The sample of individuals is nearly evenly split between female and male individuals. There were more parents who did not respond to lithium than who did. The SES variable has 5 levels, where a higher level corresponds to higher SES. There are few parents who fall into the first 3 levels, and even combining the first 3, there are less of them that fall into the combined 1-3 category than in each of level 4 and 5. Table 3.2 contains descriptive statistics related to the length of the study and parental onset age. The average number of years each individual was followed in the study was 5.28, with a maximum of 17.25 and a minimum of 0, which occurs when the individual dropped out after the initial HAMA score collection or experienced the event at the same time. As seen in Table 3.2, the average parental onset of BD was 26.32 years and the average age for the event outcome was 23.71 years.

In this thesis, the longitudinal covariate of interest is the measurement recorded on the HAMA scale. Figure 3.1 provides the distribution of HAMA scores and Table 3.3 contains descriptive statistics of HAMA scores. In Figure 3.1 A, a histogram of the HAMA scores is plotted. It can be seen from this plot that HAMA scores are skewed to the right. A linear mixed effects model was estimated based on

$$\begin{aligned} HAMA_i(t) = \beta_0 + \beta_1 HAMAAGE_i(t) + \beta_2 SEX_i + \\ \beta_3 SES_{4i} + \beta_4 SES_{1,2,3i} + \beta_4 LITHRESP_i + \beta_5 PARONSAGE_i + b_i + \epsilon_i(t). \end{aligned} \tag{3.1}$$

where LITHRESP is parent's lithium response and PARONSAGE is parent's onset age. The residuals from this model have some curvature at the tails of the normal QQplot in Figure 3.2 A. For a linear mixed-effects model, the residuals are assumed to be normal, therefore the HAMA scores violate that assumption. The HAMA scores were transformed by taking

the log of the score plus 1 to account for instances where the score was 0. The histogram in Figure 3.1 B and the normal QQplot in Figure 3.2 B for the transformed scores look better although not exactly normal. The residuals of the model shown in Equation 3.1 are also used in Figure 3.3. In Figure 3.3 B the transformed scores display that the plot is better than Figure 3.3 A. The descriptive statistics of the HAMA scores are presented in Table 3.3. The number of measurements per individual differ, where the average number of recorded HAMA scores is 3.42, with the most being 13 and the least being 1. Figure 3.4 A contains the longitudinal trajectories for HAMA scores and Figure 3.4 B contains the longitudinal trajectories for log transformed HAMA scores.

	Frequency (%)
Event Outcome:	
Yes	19 (10.2)
No	167 (89.8)
Sex:	
Male	83 (44.6)
Female	103 (55.4)
Lithium Response:	
Yes	77 (41.4)
No	109 (58.6)
Hollingshead index for SES:	
1,2,3	30 (16.1)
4	68 (36.6)
5	88 (47.3)

Table 3.1: Descriptive statistics for discrete variables

	Mean	SD	Median	Min	Max
Age of first HAMA score (years)	18.43	7.17	17.15	6.56	42.12
Age at event outcome (years)	23.71	7.79	23.91	7.68	45.81
Years to follow up	5.28	4.10	4.61	0.00	17.25
Parental onset age (years)	26.32	10.21	25.43	5.12	49.63

Table 3.2: Descriptive statistics for continuous variables

	Mean	SD	Median	Min	Max
HAMA scores	5.80	5.50	4.00	0.00	30.00
Log Transformed HAMA scores	1.58	0.87	1.61	0.00	3.43
Number of HAMA scores per individual	3.42	2.45	3.00	1.00	13.00

Table 3.3: Descriptive statistics of HAMA measurements

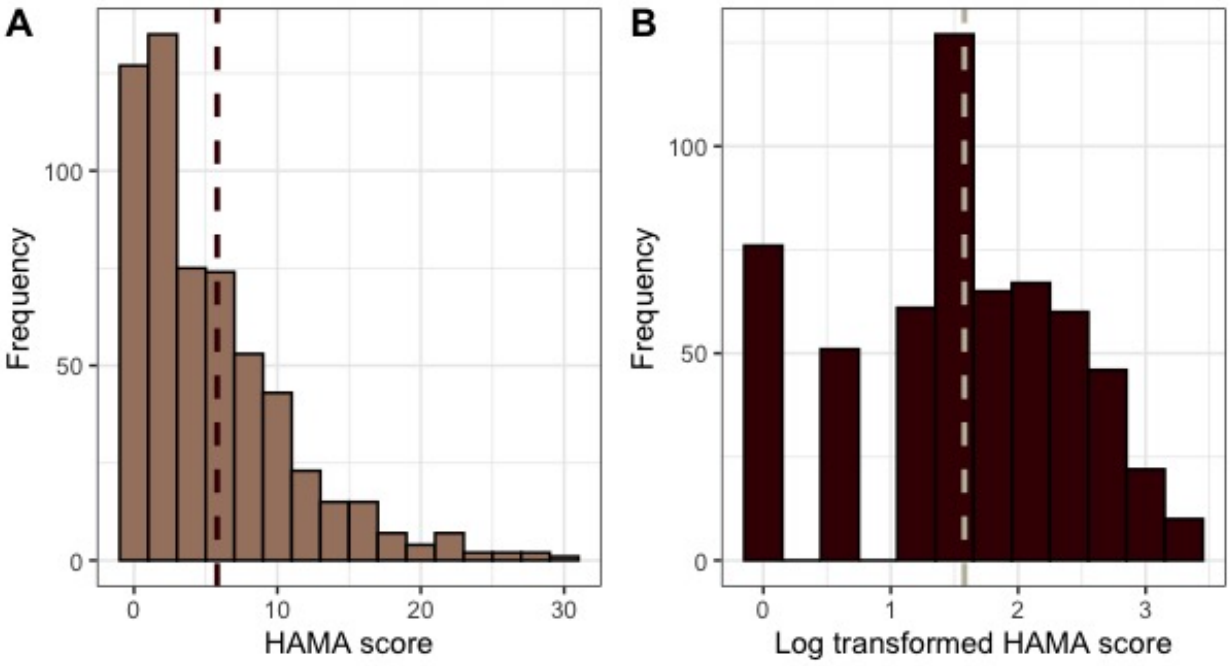

Figure 3.1: Histogram for (A) HAMA score and (B) log transformed HAMA score

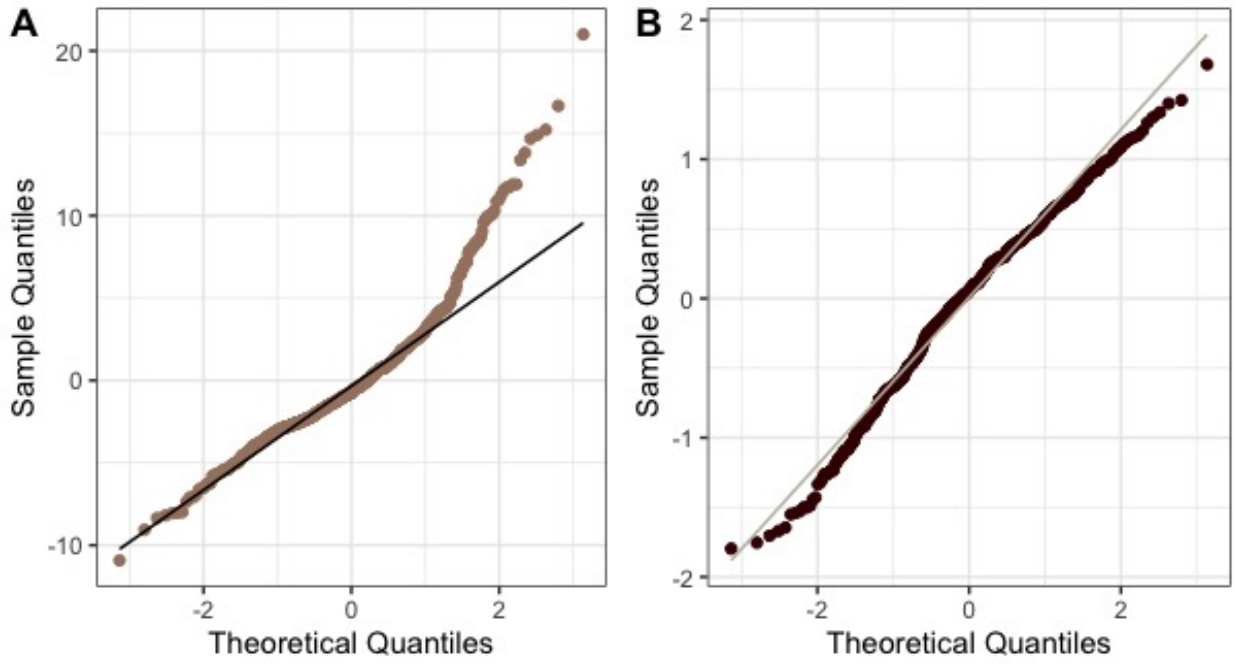

Figure 3.2: Normal QQplot for residuals of linear mixed effects model with (A) HAMA score and (B) log transformed HAMA score

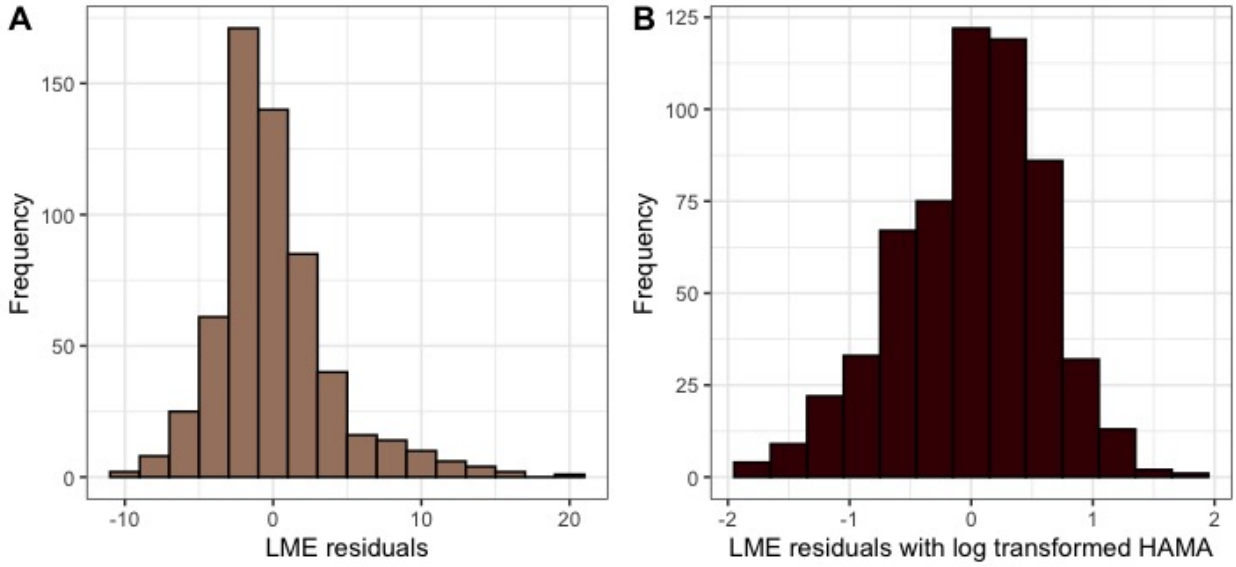

Figure 3.3: Histogram of residuals of linear mixed effects model with (A) HAMA score and (B) log transformed HAMA score

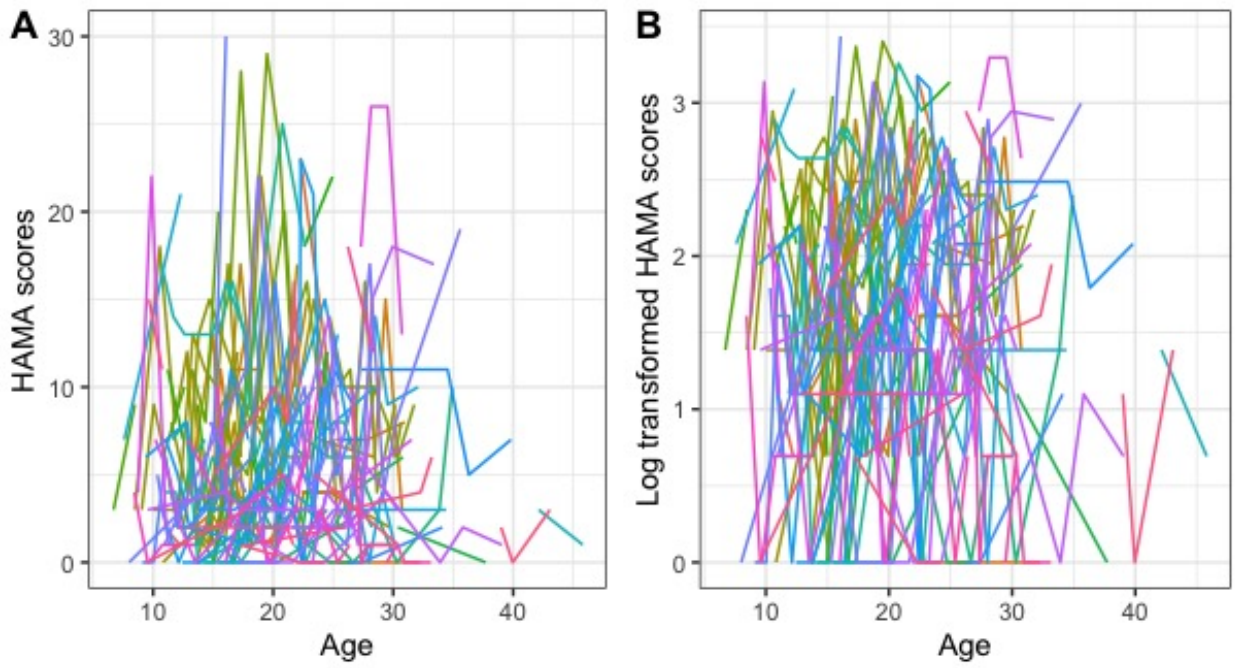

Figure 3.4: Longitudinal trajectory plot for (A) HAMA scores and (B) log transformed HAMA scores

3.1.2 Models and Analysis

Joint models were fit with the JM package [Rizopoulos, 2010] in R [R Core Team, 2022]. The code used to conduct the analysis in this section can be found in Appendix A.1. The following coding was used for the time independent variables:

$$
\begin{aligned}
FEMALE &= \begin{cases} 1 & \text{Female} \\ 0 & \text{Male} \end{cases} \\
SES_4 &= \begin{cases} 1 & \text{SES score of 4} \\ 0 & \text{otherwise} \end{cases} \\
SES_{1,2,3} &= \begin{cases} 1 & \text{SES score of 1,2 or 3} \\ 0 & \text{otherwise} \end{cases} \\
LITHRESP &= \begin{cases} 1 & \text{responds to lithium} \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}
\tag{3.2}
$$

Four joint models were fit, each with a unique longitudinal submodel that will be discussed in this section. All of the joint models have the same survival submodel. The first longitudinal submodel is a random intercept mixed-effects model, where the log transformed HAMA score is used as the longitudinal response variable and time, t, in Equation 2.1 is represented by the individual's age at the time of HAMA measurement ($HAMAAGE$). The first longitudinal submodel, a linear model for the log transformed HAMA score for the i^{th} individual where $i = 1, ..., 186$ is

$$log(HAMA_i(t) = \beta_0 + \beta_1 HAMAAGE_i(t) + b_i + \epsilon_i(t). \tag{3.3}$$

As mentioned earlier, it is not always reasonable to assume a linear trajectory for longitudinal data. In this thesis, B-splines are implemented to assess the impact of increased flexibility on estimating the longitudinal trajectory. The next longitudinal model will use a quadratic B-spline with no interior knots. Therefore, the basis functions of the quadratic B-spline for $HAMAAGE$ are used as the longitudinal covariates in the random intercept longitudinal submodel. This model, which will be referred to as the quadratic longitudinal model, is as follows:

$$\begin{aligned} log(HAMA_i(t) = m_i(t) + \epsilon_i(t) = \beta_0 + \beta_1 bs_1(HAMAAGE_i(t)) \\ +\beta_2 bs_2(HAMAAGE_i(t)) + b_i + \epsilon_i(t) \end{aligned} \tag{3.4}$$

where $bs_k(t)$, $k = 1, 2$, is the k^{th} basis function at time t.

The next longitudinal submodel used employed a cubic B-spline with no interior knots. This model, which will be referred to as the cubic longitudinal model, is as follows:

$$\begin{aligned} log(HAMA_i(t) = m_i(t) + \epsilon_i(t) = \beta_0 + \beta_1 bs_1(HAMAAGE_i(t)) \\ +\beta_2 bs_2(HAMAAGE_i(t)) + \beta_3 bs_3(HAMAAGE_i(t)) + b_i + \epsilon_i(t) \end{aligned} \tag{3.5}$$

where $bs_k(t)$, $k = 1, 2, 3$, is the k^{th} basis function for the cubic model.

Lastly, a model using the cubic B-spline with 3 additional knots was used. The knots were placed at the 1^{st} quartile, median and 3^{rd} quartile of the age variable ($HAMAAGE$). This model will be referred to as the longitudinal model with B-spline and is defined as follows:

$$log(HAMA_i(t) = m_i(t) + \epsilon_i(t) = \beta_0 + \beta_1 bs_1(HAMAAGE_i(t))$$
$$+\beta_2 bs_2(HAMAAGE_i(t)) + \beta_3 bs_3(HAMAAGE_i(t)) + \beta_4 bs_4(HAMAAGE_i(t))+ \quad (3.6)$$
$$\beta_5 bs_5(HAMAAGE_i(t)) + \beta_6 bs_6(HAMAAGE_i(t)) + b_i + \epsilon_i(t$$

where $bs_k(t)$, $k = 1, ..., 6$, is the k^{th} basis function for the B-spline model.

Each of the survival submodels, utilizes one of the longitudinal trajectories, $m_i(t)$, defined above in Equations 3.3, 3.4, 3.5 and 3.6 and the time independent covariates $LITHRESP$, $FEMALE$, and SES:

$$h_i(t|\mathcal{M}_i(t), w_i) = h_0(t)exp\{\gamma_1 LITHRESP_i + \gamma_2 FEMALE_i +$$
$$\gamma_3 SES_{4i} + \gamma_4 SES_{1,2,3_i} + \alpha m_i(t)\},\ t > 0. \quad (3.7)$$

The function $h_0(t)$ is assumed to have the form of a Weibull baseline hazard function, namely

$$h_0(t) = \lambda \omega t^{\omega - 1} \quad (3.8)$$

where λ is the scale parameter and ω is the shape parameter [Rizopoulos, 2012].

The results for the four joint models are shown in Tables 3.4, 3.5, 3.6 and 3.7. The tables include the coefficient estimates, their standard errors and p-values.

As shown in Table 3.4, the linear longitudinal submodel indicates that there is a significant relationship (p-value: 0.0001) between the log transformed HAMA scores and the age of assessment. This model estimates that for a one year increase in age at assessment the log HAMA score will increase by 0.027 units.

The results of the model fit with a quadratic B-spline is in Table 3.5. Only the 1^{st} component of the spline on the age covariate ($HAMAAGE$) is significant (p-value: <0.0001).

The output of the cubic longitudinal model is presented in Table 3.6 where only the 2^{nd} component of the spline on the age covariate ($HAMAAGE$) is significant (p-value: 0.0001).

In Table 3.7, where the results of the the B-spline longitudinal model are presented, none of the components of the spline on the age covariate are significant. In this model, only the intercept is significant (p-value: 0.023).

With respect to the survival submodel for each of the joint models in Tables 3.4, 3.5, 3.6 and 3.7, using a significance level of 0.05, there is no evidence of an association between any of the time independent covariates and the time to diagnosis. The association parameter in the output represents an estimate of α in Equation 3.7 which quantifies the relationship

between the longitudinal trajectory of the log transformed HAMA scores and time to the event of interest. In each joint model, the survival submodel estimates are similar. In all three of the models, there is strong evidence to suggest that the association parameter is not 0.

In Table 3.4, corresponding to the linear longitudinal model, the association parameter is estimated to be 1.164 (p-value: 0.034). In Table 3.6, corresponding to the cubic longitudinal model, the association parameter is estimated to be 1.290 (p-value: 0.022). For the B-spline longitudinal model results presented in Table 3.7 the association parameter is estimated to be 1.181 (p-value: 0.031).

Based on the Akaike information criterion (AIC) [Faraway, 2016] the best model fit is the one with the lowest AIC. A model with likelihood L and number of parameters r has AIC: [Faraway, 2016]

$$AIC = -2logL + 2r. \tag{3.9}$$

The best model fit with respect to AIC for the data is the longitudinal with B-spline (AIC: 1689.711) displayed in Table 3.7. The AIC for the quadratic longitudinal (AIC: 1689.912) is 0.2 higher than the longitudinal with B-spline model. The AIC for cubic longitudinal (AIC: 1690.085) is not much higher than the two previously mentioned models, however, there is a larger increase for the linear longitudinal model (AIC: 1697.067).

Figure 3.5 displays the predicted HAMA score at each age as a black line for each of the 4 models. In Figure 3.5 D the vertical lines are at the interior knots. It can be seen that the black line in Figure 3.5 A is linear (as expected). Figure 3.5 B looks like a concave down quadratic function. Figure 3.5 C looks like part of a cubic function. Panels B and C of Figure 3.5 look quite similar which makes sense given that their results are quite similar. The notable difference is the curvature around 10-20 years. Figure 3.5 D fits a cubic between each knot.

Submodel	Variable	Est	Std. Error	p-value
	β_0	1.022	0.154	<0.0001
Linear Longitudinal	β_1	0.027	0.007	0.0001
	$\log(\lambda)$	14.636	2.390	<0.0001
	Sex	0.143	0.492	0.771
	SES_4	-0.029	0.581	0.960
Survival	$SES_{1,2,3}$	0.866	0.592	0.143
	Lithium response	-0.571	0.521	0.273
	α	1.164	0.549	0.034
	ω	1.153	0.199	<0.0001
	AIC	1697.067		

Table 3.4: Summary of the estimated model parameters of the joint model for HAMA scores with linear longitudinal submodel

Submodel	Variable	Est	Std. Error	p-value
	β_0	0.867	0.157	<0.0001
Quadratic Longitudinal	β_1	1.620	0.384	<0.0001
	β_2	0.506	0.325	0.120
	$\log(\lambda)$	-14.634	2.562	<0.0001
	Sex	0.125	0.495	0.800
	SES_4	-0.037	0.583	0.950
Survival	$SES_{1,2,3}$	0.903	0.596	0.130
	Lithium response	-0.586	0.523	0.263
	α	1.257	0.552	0.023
	ω	1.136	0.221	<0.0001
	AIC	1689.912		

Table 3.5: Summary of the estimated model parameters of the joint model for HAMA scores with quadratic longitudinal submodel

Submodel	Variable	Est	Std. Error	p-value
Cubic Longitudinal	β_0	1.063	0.213	<0.0001
	β_1	0.364	0.588	0.536
	β_2	1.772	0.463	0.0001
	β_3	-0.205	0.617	0.740
Survival	$\log(\lambda)$	14.746	2.506	<0.0001
	Sex	0.116	0.497	0.816
	SES_4	-0.025	0.584	0.966
	$SES_{1,2,3}$	0.919	0.598	0.124
	Lithium response	-0.593	0.524	0.258
	α	1.290	0.564	0.022
	ω	1.139	0.214	<0.0001
	AIC	1690.085		

Table 3.6: Summary of the estimated model parameters of the joint model for HAMA scores with cubic longitudinal submodel

Submodel	Variable	Est	Std. Error	p-value
Longitudinal with B-spline	β_0	1.011	0.443	0.023
	β_1	0.503	0.654	0.441
	β_2	-0.061	0.417	0.882
	β_3	0.833	0.480	0.082
	β_4	0.826	0.497	0.098
	β_5	0.723	0.715	0.312
	β_6	0.261	0.793	0.742
Survival	$\log(\lambda)$	-14.710	2.580	<0.0001
	Sex	0.131	0.494	0.791
	SES_4	-0.032	0.583	0.957
	$SES_{1,2,3}$	0.912	0.595	0.125
	Lithium response	-0.587	0.521	0.260
	α	1.181	0.548	0.031
	ω	1.155	0.219	<0.0001
	AIC	1689.711		

Table 3.7: Summary of the estimated model parameters of the joint model for HAMA scores with longitudinal submodel with B-spline

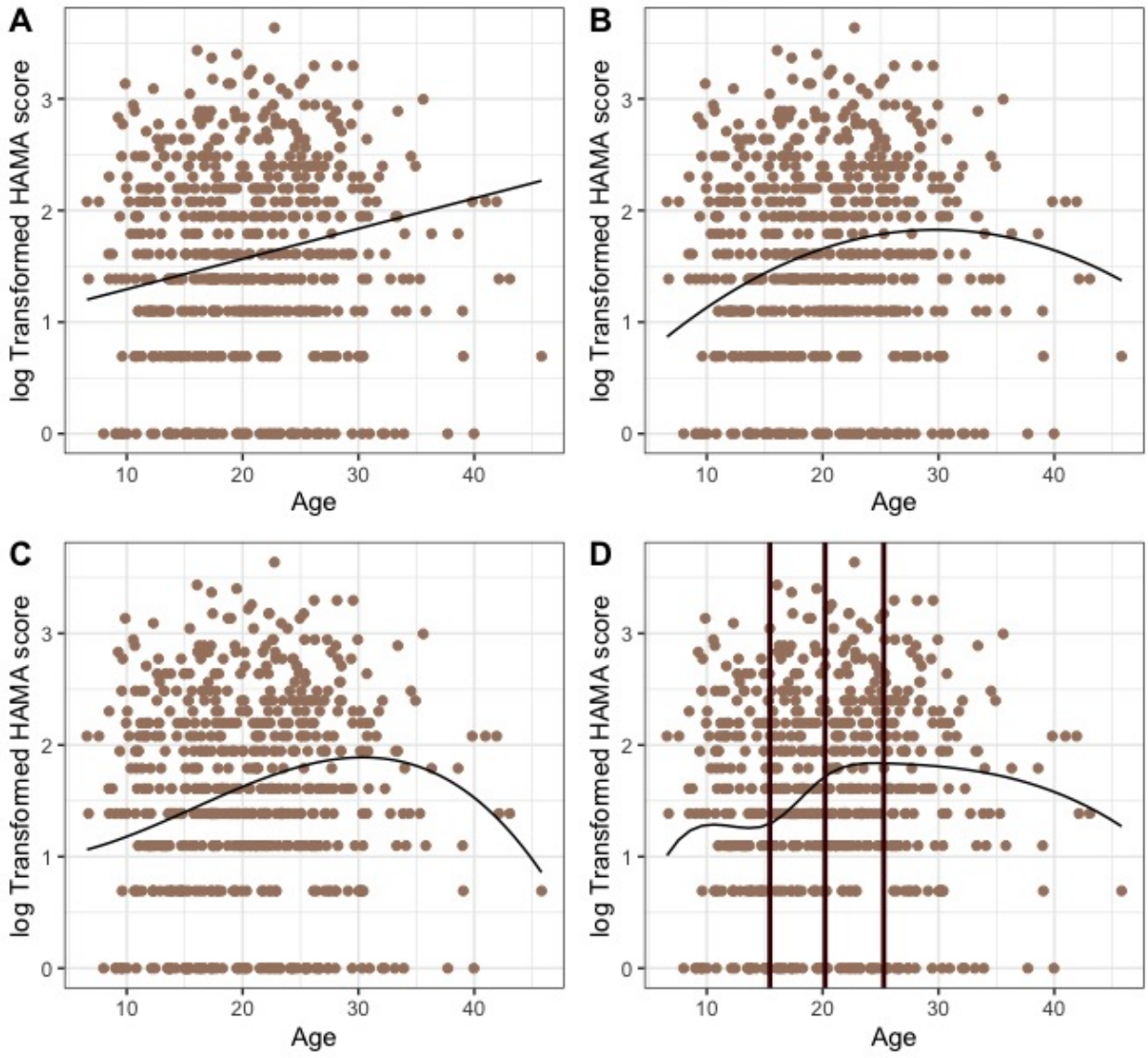

Figure 3.5: Predicted HAMA scores with a (A) linear longitudinal submodel, (B) quadratic longitudinal submodel, (C) cubic longitudinal submodel, (D) longitudinal submodel with B-spline

3.2 Framingham Heart Study Data Set

The second dataset utilized to demonstrate the methods mentioned previously is from a longitudinal study of individuals. This dataset was provided by Dr. You and Dr. Qiu at the University of South Florida [You and Qiu, 2021]. The data is a subset from the Framingham

Heart Study conducted at the University of Boston. The study follows 3 generations of individuals from the original cohort established in 1948 who lived in Framingham, Massachusetts [Mahmood et al., 2014]. The study sought to discover the risk factors of myocardial infarction and strokes [Lüscher, 2018]. The data provided has systolic and diastolic blood pressure measurements for each individual as the longitudinal covariates and uses the experience of a stroke as the event of interest [You and Qiu, 2021].

3.2.1 Introduction and Exploratory Analysis

The dataset consisted of 1055 individuals. The event outcome of interest is the time to the experience of a stroke. There are two time-varying longitudinal covariates available: a measurement of diastolic blood pressure, and a measurement of systolic blood pressure. There are no time-independent variables provided in this dataset. Two individuals were removed from the dataset because they had a diastolic blood pressure of 0, which is only possible in very specific circumstances and since the medical history of the individuals was not available, it was assumed to be a device malfunction.

From the remaining 1053 individuals, 27 experienced a stroke and 1026 (97.4%) were censored as can be seen in Table 3.10. In this thesis, the longitudinal covariate of interest is the measurement of systolic blood pressure. Both Figure 3.6 and Table 3.8 examine the distribution of the systolic blood pressure measurements and event times. In Figure 3.6 A, a histogram of systolic blood pressure measurements is plotted. It can be seen from this plot that measurements of systolic blood pressure look approximately normal. Figure 3.6 B shows the QQplot of the residuals where a linear mixed effects model was estimated such that

$$systolic_i(t) = \beta_0 + \beta_1 t_i + b_i + \epsilon_i(t). \tag{3.10}$$

The residuals from this model have some curvature at the tails of the normal QQplot, however, nothing of concern as seen in Figure 3.6. The average of systolic blood pressure measurements is 124.12, with a maximum of 220 and a minimum of 75. The average age of an individual in this dataset is 50.49 years with a maximum of 85 and a minimum of 14. Figure 3.7 A has the longitudinal trajectories for a random sample of 100 individuals. In the dataset provided, there are 7 measurements for each individual and the average time between measurements is 4.3 years, with the minimum being 1 and the maximum being 11. Figure 3.7 B is a histogram of the time between measurements. On average, the time

between the first and second measurement for an individual is longer than the time between any other set of measurements. Conversely, the interval between the last two measurements is on average smaller than the rest. These trends can be seen in Table 3.9.

	Mean	SD	Median	Min	Max
Systolic pressure	124.12	17.24	122	75	220
Diastolic pressure	76.90	10.10	76	40	178
Ages of individuals (years)	50.49	12.18	51	14	85
Time elapsed before a stroke (years)	26.74	1.17	26.50	23.50	30.50

Table 3.8: Descriptive statistics of blood pressure measurements

Time between measurements (years)	Mean	SD	Median	Min	Max
All	4.37	1.76	4	1	11
First and second	7.84	0.67	8	5	11
Second and third	4.40	0.66	4	2	7
Third and fourth	3.45	0.57	3	1	6
Fourth and fifth	3.65	0.55	4	2	6
Fifth and sixth	4.04	0.64	4	2	7
Sixth and seventh	2.85	0.92	3	1	6

Table 3.9: Descriptive statistics of measurement intervals

Event Outcome	Frequency (%)
Stroke	27 (2.6)
No Stroke	1026 (97.4)

Table 3.10: Event outcome summary (incidence of a stroke)

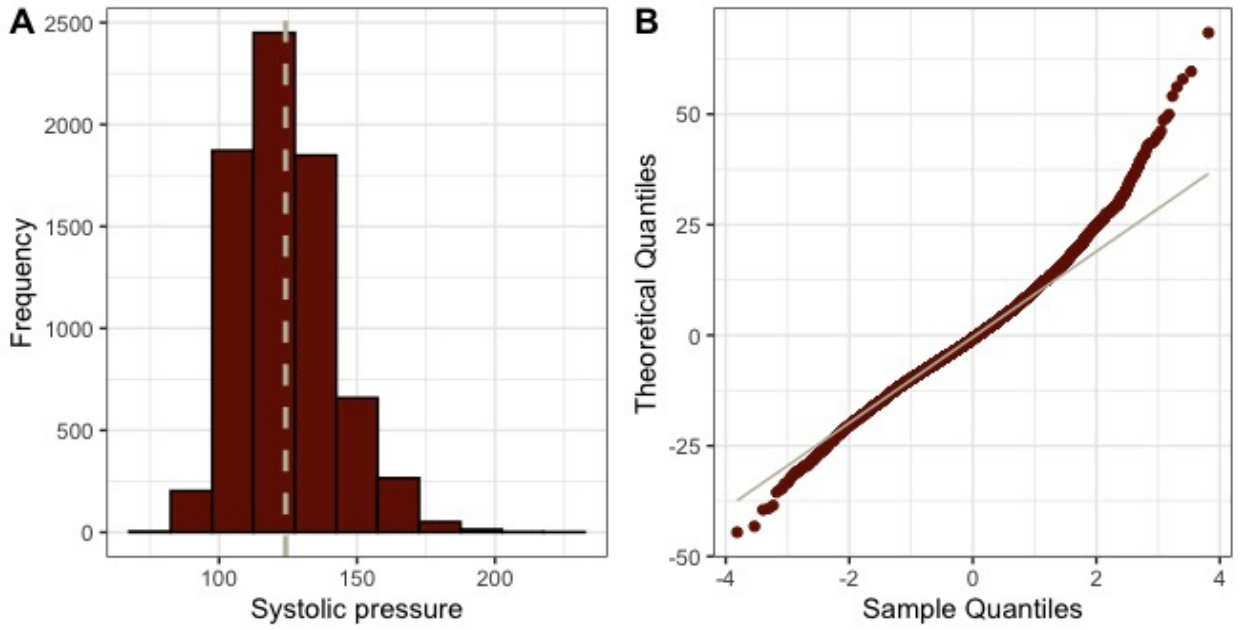

Figure 3.6: (A) Histogram for systolic blood pressure measurements and (B) Normal QQplot for residuals of linear mixed effects model with systolic blood pressure measurements

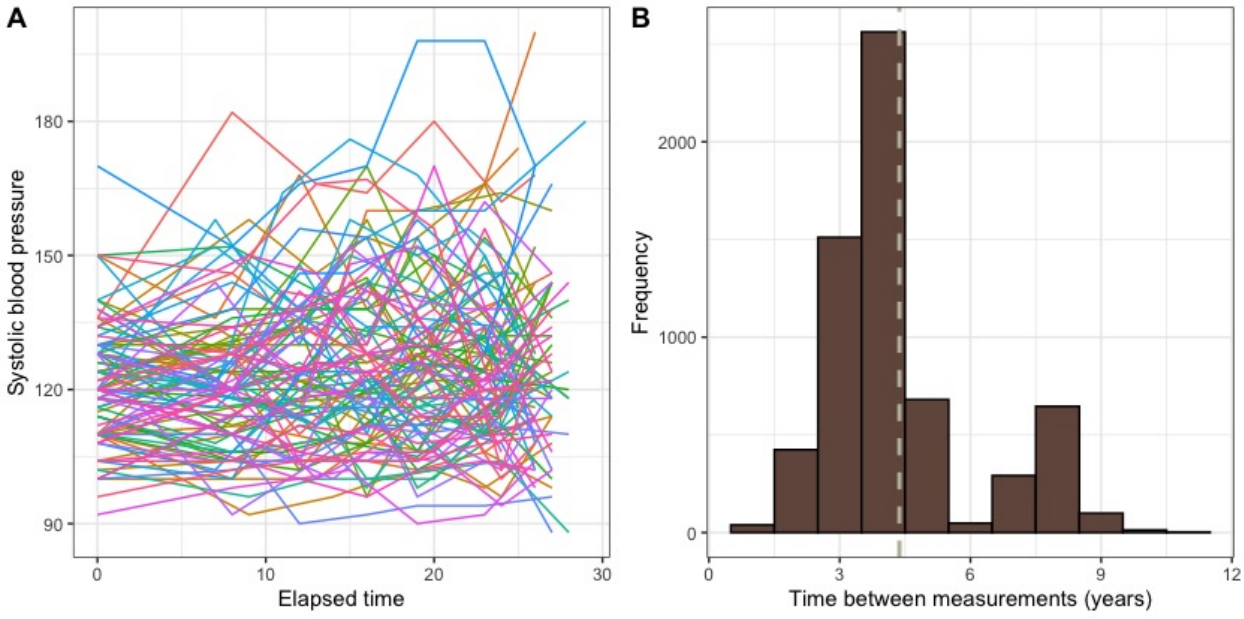

Figure 3.7: (A) Longitudinal trajectory plot for systolic blood pressure measurements for a random sample of 100 individuals and (B) histogram of the time between measurements

3.2.2 Models and Analysis

Joint models were fit with the JM package [Rizopoulos, 2010] in R [R Core Team, 2022]. The code used to conduct the analysis in this section can be found in Appendix A.2.To account for ties between event/censoring times and measurement times, it was assumed that the individuals experienced the event or were censored 0.5 units after the last measurement.

Four joint models with unique longitudinal submodels were fit. The survival submodel for each of these models was the same. The longitudinal submodel used is a random intercept mixed-effects model, where the systolic blood pressure measurement is used as the longitudinal response variable and time t in Equation 2.1 is represented by the time (t) from the start of the study to the time of the event or censoring. The first longitudinal submodel for the systolic blood pressure measurement for the i^{th} individual, where $i = 1, ..., 1053$ is estimated such that

$$systolic_i(t) = m_i(t) + \epsilon_i(t) = \beta_0 + \beta_1 t_i + b_i + \epsilon_i(t). \tag{3.11}$$

This model will be referred to as the linear longitudinal model.

In the remaining longitudinal submodels for this dataset, the basis functions of the B-splines for systolic blood pressure are used as the longitudinal covariates in the random intercept longitudinal submodel. The first B-spline model employs a quadratic basis function. This model, which will be referred to as the quadratic longitudinal model, is as follows:

$$systolic_i(t) = m_i(t) + \epsilon_i(t) = \beta_0 + \beta_1 bs_1(t_i) + \beta_2 bs_2(t_i) + b_i + \epsilon_i(t), \tag{3.12}$$

where $bs_k(t)$, $k = 1, 2$, is the k^{th} basis function for the quadratic model.

The next B-spline model uses a cubic basis function with no interior knots and will be referred to as the cubic longitudinal model. This model is as follows:

$$systolic_i(t) = m_i(t) + \epsilon_i(t) = \beta_0 + \beta_1 bs_1(t_i) + \beta_2 bs_2(t_i) + \beta_3 bs_3(t_i) + b_i + \epsilon_i(t). \tag{3.13}$$

where $bs_k(t)$, $k = 1, 2, 3$, is the k^{th} basis function for the cubic model.

The last model uses a cubic B-spline with 3 additional knots. The knots will be placed at the 1^{st} quartile, median and 3^{rd} quartile of the time variable ($time$). This model will be referred to as the longitudinal model with B-spline and is defined as follows:

$$systolic_i(t) = m_i(t) + \epsilon_i(t) = \beta_0 + \beta_1 bs_1(t_i) + \beta_2 bs_2(t_i) + \beta_3 bs_3(t_i) + \beta_4 bs_4(t_i) + \beta_5 bs_5(t_i) + \beta_6 bs_6(t_i) + b_i + \epsilon_i(t). \tag{3.14}$$

where $bs_k(t)$, $k = 1, ..., 6$, is the k^{th} basis function for the longitudinal model with B-spline.

The survival submodel used in all three joint models has one of the longitudinal trajectories $m_i(t)$ discussed previously as shown in Equations 3.11, 3.12, 3.13 and 3.14, and has no time independent covariates. The outcome of interest is the time to an individual's experience of a stroke with hazard function

$$h_i(t|\mathcal{M}_i(t), w_i) = h_0(t) exp\{\alpha m_i(t)\},\ t > 0 \tag{3.15}$$

where the function $h_0(t)$ is assumed to have the form of the Weibull baseline hazard function displayed in Equation 3.8.

The results for the four joint models are found in Tables 3.11, 3.12, 3.13, and 3.14. The tables include the coefficient estimates, their standard errors and p-values.

As shown in Table 3.11, the linear longitudinal submodel displays that there is a significant relationship (p-value: < 0.0001) between the systolic blood pressure measurement and the time to stroke. This model estimates that after one year elapses, systolic blood pressure will increase by 0.321 units. For the quadratic longitudinal model results in Table 3.12, all of the components of the spline are statistically significant with p-values less than 0.0001. For the cubic longitudinal model, both the 2^{nd} (p-value: < 0.0001) and 3^{rd} (p-value: < 0.0001) components of the spline on the covariate for time since start of study are significant. In the B-spline longitudinal model, the 3^{rd} (p-value: < 0.0001), 4^{th} (p-value: < 0.0001) and 5^{th} (p-value: < 0.0001) components are significant while the rest are not.

The association parameter in the output represents an estimate of α in Equation 3.15 which quantifies the relationship between the longitudinal trajectory of an individual's systolic blood pressure and the survival time of an individual. The estimated association for the joint models with the linear longitudinal in Table 3.11, the quadratic longitudinal in Table 3.12 and the cubic longitudinal in Table 3.13 are the same (0.075). The B-spline model in Table 3.14 is only slightly higher at 0.077. Both the survival submodels in Tables 3.13 and 3.14 have very similar estimates in their outputs.

Lowest AIC was used as the criterion for the best model fit for the data. The model with the lowest AIC is the joint model in Table 3.13, the cubic longitudinal model (AIC: 59708.210). The AIC for the B-spline longitudinal model in Table 3.14 is slightly higher

(AIC: 59710.490). The linear longitudinal model (AIC: 59734.400) had an AIC 1.01 less than the quadratic longitudinal model (AIC: 59735.410). The quadratic longitudinal model in, Table 3.12, had the highest AIC.

Figure 3.8 displays the predicted systolic blood pressure versus time since start of study as the black line for all 4 models. In Figure 3.8 D the vertical lines are placed at the interior knots. It can be seen that the black line in Figure 3.8 A is linear (as expected), while C and D show some non-linearity. Figure 3.8 B the predicted model does not look much different than Figure 3.8 A.

Submodel	Variable	Est	Std. Error	p-value
	β_0	119.319	0.471	<0.0001
Linear Longitudinal	β_1	0.321	0.016	<0.0001
	$\log(\lambda)$	-81.978	10.031	<0.0001
Survival	α	0.075	0.016	<0.0001
	ω	3.028	0.144	<0.0001
	AIC	59734.400		

Table 3.11: Summary of the estimated model parameters of the joint model for systolic blood pressure measurements with a linear longitudinal submodel

Submodel	Variable	Est	Std. Error	p-value
	β_0	119.521	0.513	<0.0001
Quadratic Longitudinal	β_1	4.036	0.805	<0.0001
	β_2	9.800	0.520	<0.0001
	$\log(\lambda)$	-81.824	10.02	<0.0001
Survival	α	0.075	0.016	<0.0001
	ω	3.025	0.144	<0.0001
	AIC	59735.410		

Table 3.12: Summary of the estimated model parameters of the joint model for systolic blood pressure measurements with a quadratic longitudinal submodel

Submodel	Variable	Est	Std. Error	p-value
Cubic Longitudinal	β_0	120.071	0.523	<0.0001
	β_1	-3.369	1.245	0.007
	β_2	12.581	1.346	<0.0001
	β_3	5.548	0.941	<0.0001
Survival	$\log(\lambda)$	-84.209	10.277	<0.0001
	α	0.075	0.016	<0.0001
	ω	3.060	0.141	<0.0001
	AIC	59708.210		

Table 3.13: Summary of the estimated model parameters of the joint model for systolic blood pressure measurements with a cubic longitudinal submodel

Submodel	Variable	Est	Std. Error	p-value
Longitudinal with B-spline	β_0	120.121	0.524	<0.0001
	β_1	0.835	3.180	0.793
	β_2	-1.739	1.668	0.297
	β_3	6.212	1.037	<0.0001
	β_4	6.060	1.157	<0.0001
	β_5	8.830	1.396	<0.0001
	β_6	3.853	3.019	0.202
Survival	$\log(\lambda)$	-84.451	10.559	<0.0001
	α	0.077	0.016	<0.0001
	ω	3.061	0.144	<0.0001
	AIC	59710.490		

Table 3.14: Summary of the estimated model parameters of the joint model for systolic blood pressure measurements with longitudinal submodel with B-spline

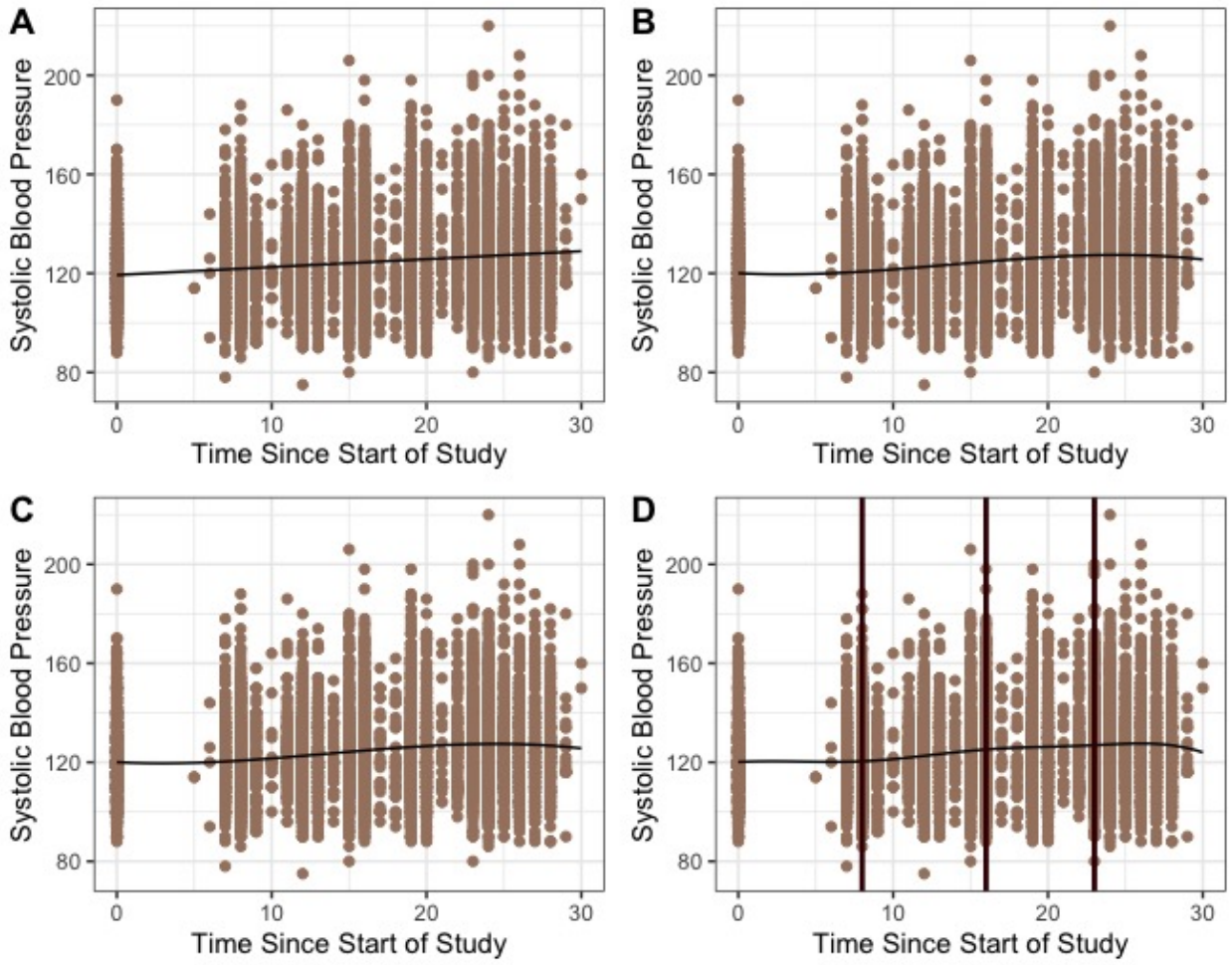

Figure 3.8: Predicted systolic blood pressure measurements with a (A) linear longitudinal submodel, (B) quadratic longitudinal submodel, (C) cubic longitudinal model, (D) longitudinal submodel with B-spline

Chapter 4

Simulation Study

4.1 Introduction

To assess the methods presented in this thesis, a simulation study was conducted. The simulation study data was generated based on the results from the analysis for the Framingham dataset [Mahmood et al., 2014] in Section 3.2. The event times for the simulated data were generated based on work by Bender et al. [2005] and Austin [2012]. Bender et al. [2005] published a method to simulate event times utilizing parametric distributions (exponential, Weibull or Gompertz). This, however, was done with only time independent covariates. Austin [2012] extended this to include a continuous time dependent covariate. Additionally, Stefan [2019] and Lowe [2020] extended the work by Austin [2012] to generate event times from a model with a time dependent covariate and random effects.

Due to the simulation being computationally intensive, the simulation was run on the network of computers from the Digital Research Alliance of Canada (known as SHARCNET: www.sharcnet.ca). The R [R Core Team, 2022] code used to conduct the simulation study can be found in Appendix A.3 and an example of the bash script to submit the code to the network is in Appendix A.4.

4.2 Simulation Design

In order to be able to fit the joint models presented in Chapter 3, both longitudinal and time-to-event data must be generated. Longitudinal data was generated based on the linear mixed effects model results presented in Table 3.11. The dataset that the simulation was

based on only had one time-varying covariate. Therefore for the simulation only one time-varying covariate was generated and for the time-to-event data the baseline hazard was kept constant at the scale parameter $\lambda = 0.000005$.

Recall from section 2.2, the Cox model with time fixed covariates of the form

$$h(t) = h_0(t)exp(\gamma^T w_i) \tag{4.1}$$

where $h_0(t)$ is the baseline hazard, γ is a vector of regression coefficients, and w_i is a vector of time fixed covariates for an individual $i = 1, .., n$ [Rizopoulos, 2012]. It can be derived that the survival function of the Cox model is

$$S_i(t) = -H_0(t)exp(\gamma^T w_i) \tag{4.2}$$

where $H_0(t)$ is the cumulative baseline hazard function which is defined as $H_0(t) = \int_0^t h_0(x)dx$. The cumulative distribution function is [Austin, 2012; Rizopoulos, 2012]

$$\begin{aligned} F_i(t) &= 1 - S_i(t) \\ F_i(t) &= 1 - exp(-H_0(t)exp(\gamma^T w_i)). \end{aligned} \tag{4.3}$$

Bender et al. [2005] noted that given a survival time T_i, for individual $i = 1, ..., n$, with distribution function F_i in Equation 4.3, the random variable $V_i = F_i(T_i)$ has a uniform distribution, $V_i \sim$ U[0,1]. It follows that $1 - V_i$ is uniformly distributed as well [Bender et al., 2005]. Using this information combined with Equation 4.3 [Bender et al., 2005]

$$F_i(T_i) = V_i = exp[-H_0(t)exp(\gamma^T w_i)] \sim \text{U}[0,1]. \tag{4.4}$$

$H_0(t)$ can be inverted if $h_0(t) > 0$ for all t and the survival time T_i of the Cox model can be generated as [Bender et al., 2005]

$$T_i = H_0^{-1}[-log(V_i)exp(-\gamma^T w_i)]. \tag{4.5}$$

where V_i is a U(0,1) pseudo-random number.

As mentioned, Austin [2012] extended the work by Bender et al. [2005] to incorporate a longitudinal covariate and Stefan [2019] and Lowe [2020] further extended it to accommodate individual random effects. Austin [2012] sets the baseline hazard h_0, to a constant, λ for an exponential.

In the simulation there are no fixed effects in the survival submodel. The hazard for an

individual $i = 1, ..., 1000$ used in this simulation is

$$h_i(t) = \lambda exp(\alpha(\beta_0 + \beta_1 t + b_i)). \tag{4.6}$$

Following the logic laid out by Stefan [2019] and Lowe [2020] and Equation 4.4, the event times for a joint model can be generated using

$$T_i = \frac{1}{\alpha\beta_1} log(1 + \frac{\alpha\beta_1(-log(V_i))}{\lambda exp(\alpha(\beta_0 + b_i))} \tag{4.7}$$

where β_0 is the fixed intercept, β_1 is the fixed slope, V_i is a U(0,1) pseudo-random number, and b_i is the random intercept as defined in Equation 2.1 in section 2.1.1. The variable α represents the association between a time dependent covariate and an individual's survival referenced in section 2.1.2.

The simulation data was generated from the following joint model

$$\begin{aligned} h_i(t) &= \lambda \exp\{\alpha(\beta_0 + \beta_1 t + b_i)\} \\ y_i(t) &= \beta_0 + \beta_1 t_i(t) + b_i + \epsilon_i(t) \\ b_i &\sim \mathcal{N}(0, \sigma_b^2) \\ \epsilon_i(t) &\sim \mathcal{N}(0, \sigma^2). \end{aligned} \tag{4.8}$$

Originally n_i=7 observations $y_i(t)$ were generated for each individual $i = 1, ..., 1000$ at times $0, 1, ..., 6$. However, observations with $t < T_i$ (the event time) were discarded (filtered). The true parameters used for the simulation are displayed in Table 4.1 where n is the number of individuals and n_i is the number of observations per individual i before filtering and N is the number of iterations of the simulation. Table 4.2 contains the descriptive statistics of the number of measurements per individual for 1000 iterations of the simulation. Table 4.3 contains descriptive statistics of the event times and systolic blood pressures from 1000 iterations of the simulation after filtering, compared to descriptive statistics of these parameters from the Framingham data set used in Section 3.2. Figure 4.1 shows the distribution of systolic blood pressure, event time and number of measurements of an individual as well as a subset of longitudinal trajectories from a single iteration of the simulation. Figure 4.2 displays the predicted systolic blood pressure measurements from a single iteration of the simulation at each time as a black line.

N	n	n_i	β_0	β_1	α	λ	σ_b	σ
1000	1000	7	119.30	0.30	0.07	0.000005	11.70	12.40

Table 4.1: True values of parameters used to generate simulated data

	Mean	SD	Median	Min	Max
Number of observations per individual	6.41	1.50	7	1	7

Table 4.2: Descriptive statistics of number observations per individual for 1000 iterations of the simulation

	Mean	SD	Median	Min	Max
Systolic pressure	124.12	17.24	122	75	220
Systolic pressure simulated	119.30	16.80	23.10	119.34	204.51
Event time	26.74	1.17	26.05	23.50	30.50
Event time simulated	34.18	27.91	0.00001	26.44	240.35

Table 4.3: Descriptive statistics of simulated blood pressure measurements and event times for 1000 iterations compared to the Framingham data set

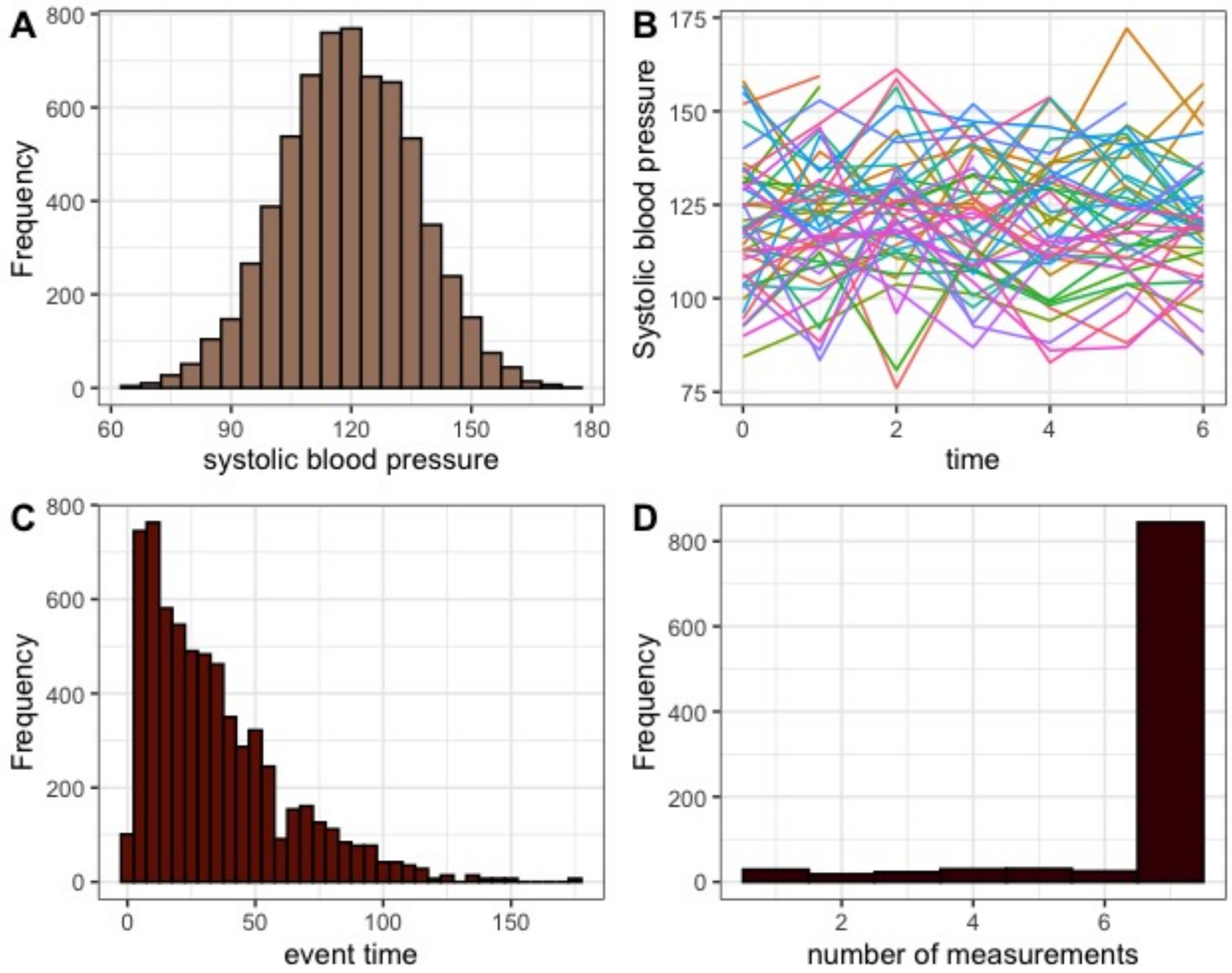

Figure 4.1: Descriptive plots for a single iteration of the simulation (A) Histogram of systolic blood pressure measurements (B) longitudinal trajectory plot for systolic blood pressure measurements (C) histogram of event times (D) histogram of number of measurements per individual after filtering

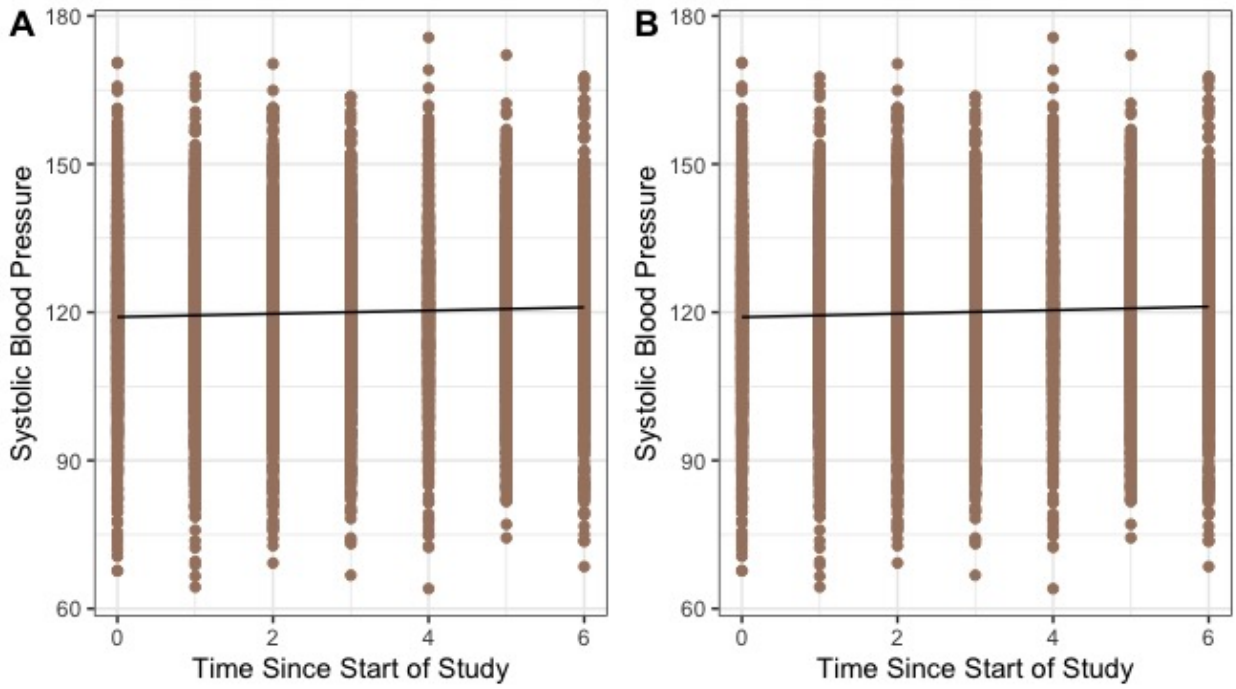

Figure 4.2: Predicted systolic blood pressure measurements for a single iteration with a (A) linear longitudinal submodel, (B) quadratic longitudinal submodel

4.3 Analysis

To aggregate the results of the simulations, an average of the estimates across the iterations was used. This can be expressed using δ to represent an arbitrary model parameter. The mean of the parameter estimates can be denoted by $\bar{\delta}_{sim}$ which is represented as

$$\bar{\delta}_{sim} = \frac{\sum_{c=1}^{c=n_{sim}} \hat{\delta}_c}{n_{sim}} \tag{4.9}$$

where n_{sim} represents the number of simulation iterations conducted and $c = 1, ..., n_{sim}$ indexes the iterations and $\hat{\delta}_c$ is the estimate from the cth iteration. The number of simulation iterations conducted in this thesis is 1000. The empirical standard deviation of the parameter estimates, denoted as SD_e can be expressed as

$$SD_e = \frac{\sum_{c=1}^{c=n_{sim}} (\hat{\delta}_c - \bar{\delta}_{sim})^2}{n_{sim} - 1}. \tag{4.10}$$

The percent relative bias (PRB) is calculated using the true value of the parameter, δ_{true}, as

$$PRB = \frac{\bar{\delta}_{sim} - \delta_{true}}{\delta_{true}} \times 100. \tag{4.11}$$

The simulation was conducted so that 1000 of the linear longitudinal models were successfully fit. It should be noted that not every iteration that led to a successful fitting of a linear longitudinal model led to a successful fit of the quadratic, cubic and B-spline longitudinal models. The models were either unsuccessful due to failed convergence of the longitudinal submodel or resulted in a non positive-definite Hessian in the joint model. These models were not included in the summaries in Tables 4.4 and 4.5. The quadratic longitudinal model had 890 successful models fit, the cubic longitudinal model had 26 successful models, and the longitudinal with B-spline model had 3 models successfully fit. The results of the linear and quadratic longitudinal models are displayed in Tables 4.4 and 4.5. The results of the cubic and longitudinal with B-spline models are based on very few iterations and as such their results are not shown. The true model in the simulation has a single linear covariate (*time*) in the longitudinal submodel and no time-fixed covariates. One reason that the cubic longitudinal and longitudinal model with B-spline encountered difficulty in fitting could be due to over-fitting. These models had between 3 and 6 parameters in the longitudinal submodel that were used to model the covariate *time*.

The joint model with linear longitudinal submodel results displayed in Table 4.4 include a column of the true values of the parameters since the data is generated based on the linear longitudinal model from Section 3.2. The mean estimates of the parameters in this model are quite close to the true values. This is demonstrated through the PRB being less than 1%.

The results for the joint model with a quadratic longitudinal submodel are presented in Table 4.5. Comparing the model estimates from the quadratic longitudinal model to the linear longitudinal model, the intercepts of the longitudinal models are quite similar. Additionally, the average standard error of the intercept is also very similar in both models. This is also the case for the estimates of the variance components of the longitudinal models, σ and σ_b. The estimate of the association parameter, α, in the quadratic model is slightly smaller than in the linear model. The estimate of $\log(\lambda)$ of the survival model is slightly larger in the quadratic model than the linear model. Neither of the regression coefficient estimates in the quadratic longitudinal model are similar to the single regression coefficient estimate in the linear longitudinal model. The average standard error of $\hat{\beta}_1$ is 10 times larger in the quadratic model than the linear model. When using lowest AIC as the criterion for

choosing the better model fit, the true model (linear model) was selected 85% of the time in 890 iterations while the quadratic model was selected 15% of the time.

Submodel	Variable	True Value	Mean Est	PRB(%)	SD_e	Mean SE_{JM}
	β_0	119.300	119.285	-0.012	0.411	0.424
Linear Longitudinal	β_1	0.300	0.302	0.662	0.017	0.017
	σ	11.800	11.699	0.106	-0.863	
	σ_b	12.400	12.392	0.329	-0.065	
	$\log(\lambda)$	-12.200	-12.235	0.503	0.004	0.004
Survival	α	0.070	0.070	0.000	0.004	0.504

Table 4.4: Summary of the estimated model parameters of the joint model for simulation measurements with a linear longitudinal submodel for 1000 iterations

Submodel	Variable	Mean Est	SD_e	Mean SE_{JM}
	β_0	119.280	0.470	0.440
	β_1	1.058	0.554	0.170
Quadratic Longitudinal	β_2	1.743	0.274	0.247
	σ	11.696	0.108	
	σ_b	12.386	0.328	
	$\log(\lambda)$	-11.413	2.582	0.004
Survival	α	0.064	0.021	0.468

Table 4.5: Summary of the estimated model parameters of the joint model for simulation measurements with a quadratic longitudinal submodel for 890 iterations

Chapter 5

Conclusion And Future Work

Joint models are often used in many applications that have both a longitudinal and a time-to-event component. There are situations where a linear longitudinal model is insufficient and misses capturing some of the trajectory of the longitudinal covariate. This thesis explored the behaviour of joint models with a non-linear longitudinal submodel by specifically utilizing B-splines in the longitudinal submodels.

Three joint model with B-splines in the longitudinal submodels were compared to a joint model with a linear longitudinal submodel. All four models were applied to two real world data sets. The first was data regarding the onset of BD, major depressive disorder (MDD) or schizoaffective disorder, in individuals who had at least one parent that was diagnosed with BD [Duffy et al., 2007]. The bipolar data set analysis found that the model that fit the data best using the criterion of lowest AIC was the longitudinal model with B-spline. This model had 3 interior knots in addition to the 2 boundary knots. The second data set that was analyzed involved the onset of stroke in individuals who were participants in a study in Framingham, Massachusetts [Mahmood et al., 2014]. The Framingham data set analysis found that the model that fit the data best using the criterion of lowest AIC was the cubic longitudinal model. This model employed cubic B-spline basis functions as the regression coefficients. It had just the two default boundary knots.

The simulation study conducted to assess the performance of incorporating B-splines into the longitudinal submodel showed that the introduction of B-splines into the longitudinal submodel had marginal impact on the parameter estimates. When using lowest AIC as the criterion for choosing the better model fit, the true model (linear model) was selected 85% of the time while the quadratic model was selected 15% of the time. However, it highlighted the possibility of over-fitting the data through the difficulty in successful fitting

of models with higher flexibility (cubic longitudinal model and longitudinal model with B-spline). Data was generated according to methods developed by Austin [2012], Bender et al. [2005], Stefan [2019] and Lowe [2020]. The λ parameter of the simulation was chosen to ensure realistic event times that were similar to the data the simulation was based on, namely the Framingham data.

When estimating models to assess the impact of including B-splines in the longitudinal submodel in the data analysis section, no notable issues arose. However, when applying the same four longitudinal submodels to data generated in the simulation, not every model fit would converge. This was due to either non-convergence of the longitudinal submodel or a non positive-definite Hessian being produced when the joint model was fit. As a result, the impact of the cubic longitudinal model and the longitudinal model with B-spline on the simulated data could not be assessed in this thesis.

For future work, simulating data with a higher degree of non-linearity should be used in order to assess the impact of higher degree B-splines in a longitudinal submodel. In addition, investigating and developing methods for simulating event times with time-varying covariates that have a non-linear effect on the response should also be explored. Additionally, incorporating the impact of the family structure in the model for the BD data should also be investigated. Submodels for the longitudinal data that incorporate zero-inflation could be explored.

In conclusion, this thesis exemplified that incorporating the appropriate degree of B-splines in the longitudinal model can improve the model fit. Model fit was assessed using the criterion of lowest AIC. In the future, other methods to assess model fit and the selection of an appropriate degree of B-spline should also be considered.

Bibliography

American Psychiatric Association (1994). *Diagnostic and Statistical Manual of Mental Disorders, 4th edition (DSM-IV)*. APA.

Austin, P. C. (2012). Generating survival times to simulate Cox proportional hazards models with time-varying covariates. *Statistics in Medicine*, 31:3946–3958.

Bender, R., Augustin, T., and Blettner, M. (2005). Generating survival times to simulate Cox proportional hazards models. *Wiley InterScience*, 24:1713–1723.

Chaudhuri, A. (2019). B-splines. *Samsung R & D Institute Delhi*, pages 1–11.

Duffy, A., Alda, M., Crawford, L., Milin, R., and Grof, P. (2007). The early manifestations of bipolar disorder: a longitudinal prospective study of the offspring of bipolar parents. *Bipolar Disorders*, 9:828–838.

Faraway, J. (2016). *Extending the Linear Model with R Generalized Linear, Mixed Effects and Nonparametric Regression Models*. CRC Press, 2nd edition.

French, M. (2021). Joint models for multivariate longitudinal data and time-to-event data. Master's thesis, University of Guelph.

Gaudet, D. (2021). Marginal approaches for joint models with clustered data. Master's thesis, University of Guelph.

Goldstein, B. I., Shamseddeen, W., Axelson, D. A., Kalas, C., Monk, K., Brent, D. A., Kupfer, D. J., and Birmaher, B. (2010). Clinical, demographic, and familial correlates of bipolar spectrum disorders among school-aged offspring of parents with bipolar disorder. *Journal of the American Academy of Child and Adolescent Psychiatry*, 49:388–396.

Hamilton, M. (1959). The assessment of anxiety states by rating. *British Journal of Medical Psychology*, 32:50–55.

Hastie, T., Tibshirani, R., and Friedman, J. (2017). *Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer, 2st edition.

Hickey, G., Philipson, P., and Jorgensen, A. (2018). joineRML: a joint model and software package for time-to-event and multivariate longitudinal outcomes. *BMC Medical Research Methodology*, 18(1).

Hollingshead, A. B. (2011). Four factor index of social status. *Yale Journal of Sociology*, 8:21–51.

Hsieh, F., Tseng, Y.-K., and Wang, J.-L. (2006). Joint modeling of survival and longitudinal data: Likelihood approach revisited. *Biometrics*, 62.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2017). *Introduction to Statistical Learning: with Applications in R*. Springer, 1st edition.

Johnson, S. A. (2013). *Encyclopedia of Operations Research and Management Science*, pages 1443–1446. Springer.

Lowe, M. (2020). The cumulative effects of time-varying covariates in survival analysis. Master's thesis, University of Guelph.

Lüscher, T. F. (2018). What is a normal blood pressure? *European Heart Journal*, 39(24):2233–2240.

Mahmood, S. S., Levy, D., Vasan, R. S., and Wang, T. J. (2014). The Framingham heart study and the epidemiology of cardiovascular disease: a historical perspective. *Lancet*, 383.

McCormick, U., Murray, B., and McNew, B. (2015). Diagnosis and treatment of patients with bipolar disorder: A review for advanced practice nurses. *Journal of the American Association of Nurse Practitioners*, 27:530–542.

Perperoglou, A., Sauerbrei, W., Abrahamowicz, M., and Schmid, M. (2019). A review of spline function procedures in R. *BMC Medical Research Methodology*.

Philipson, P., Sousa, I., Diggle, P. J., Williamson, P., Kolamunnage-Dona, R., Henderson, R., and Hickey, G. L. (2018). *joineR: Joint Modelling of Repeated Measurements and Time-to-Event Data*. R package version 1.2.6.

Pinheiro, J., Bates, D., DebRoy, S., Sarkar, D., and R Core Team (2022). *nlme: Linear and Nonlinear Mixed Effects Models*. R package version 3.1-155.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

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.

Stefan, G. (2019). A comparison of Cox and joint models for time-to-event data. Master's thesis, University of Guelph.

Therneau, T. M. (2021). *A Package for Survival Analysis in R.* R package version 3.2-13.

Williamson, P. R., Kolamunnage-Dona, R., Philipson, P., and Marson, A. G. (2008). Joint modelling of longitudinal and competing risks data. *Statistics in Medicine*, 27:6426–6438.

Wulfsohn, M. and Tsiatis, A. (1997). A joint model for survival and longitudinal data measured with error. *Biometrics*, 53:330–339.

You, L. and Qiu, P. (2021). Joint modeling of multivariate nonparametric longitudinal data and survival data: A local smoothing approach. *Statistics in Medicine*, 40:6689–6706.

Appendix A

Source Code

A.1 Bipolar Data Analysis

```
library(readxl)
library(ggplot2)
library(tidyverse)
library(cowplot)
library(tayloRswift)
library(xtable)#to be able to convert tables to latex

HAMA <- BD.data %>% names() %>% keep(~ str_detect(.,"HAMATOT"))

`%nin%` = Negate(`%in%`)

#get all ID's for completely null HAMA and completely null BDI scores
IS.Null<-BD.data%>%
  select(contains(c("ID","HAMA")))%>%
  filter_at(vars(HAMA), all_vars(is.na(.))) #removes 32

# do some cleanup
Data.set<-BD.data%>%
  filter(ID %nin% IS.Null$ID)%>% #remove any of the ID's identified above
  mutate(OUTCOME1BAGE= case_when(OUTCOME1B == 0  ~ AGELASTINT,
  #replace the age of outcome with age last interview if obs is censored
```

```
                                OUTCOME1B == 1 ~ OUTCOME1BAGE),
        SES = case_when(PARENTSES_1 %in% c(1,2,3)~3,
                        PARENTSES_1 ==4 ~2,
                        PARENTSES_1==5 ~ 1))%>%
  filter(HAMAAGE_1<=OUTCOME1BAGE)%>%
  #keep only obs where the outcome occurs after the start of the study
  select(-contains(c("SLEEP","OUTCOME1A", "BDI", "OUTCOME2", "BDIY")))

Long.2<-Data.set %>%  pivot_longer(
  cols = -1:-11,
  names_to = c('var', "level"),
  names_sep = "_",
  values_to = c("value")) %>%
  pivot_wider(names_from = var, values_from = value,values_fn = list) %>%
  unnest(c(HAMATOT, HAMAAGE)) %>%
  mutate(HAMATOT.2 = case_when(OUTCOME1BAGE < HAMAAGE ~ as.numeric(NA),
                               TRUE ~ HAMATOT),
        logHAMA = log(HAMATOT.2 + 1),
        SES = case_when(PARENTSES_1 %in% c(1,2,3) ~ 3,
                        PARENTSES_1 == 4 ~ 2,
                        TRUE ~ 1),
                      level=as.numeric(level))%>%
  drop_na(logHAMA)

#Summary Statistics
#check how many people were censored versus had the event
cens<-Data.set%>%group_by(OUTCOME1B)%>%count()

#average number of observations per person
avg<-Long.2%>%
  drop_na(HAMATOT.2)%>% count(ID)%>%
  summarize(" "="Number of HAMA scores per individual",
  Mean=round(mean(n),2),
  SD=round(sd(n),2), Median = round(median(n),2),
```

```
  Min=round( min(n),2),Max=round( max(n),2))

avglogHAMA<-Long.2%>%
  drop_na(HAMATOT.2)%>%
  summarize(" "="Transformed HAMA scores", Mean=round(mean(logHAMA),2),
  SD=round(sd(logHAMA),2), Median = round(median(logHAMA),2),
  Min=round( min(logHAMA),2),Max=round( max(logHAMA),2))

avgHAMA<-Long.2%>%
  drop_na(HAMATOT.2)%>%
  summarize(" "="HAMA scores", Mean=round(mean(HAMATOT.2),2),
  SD=round(sd(HAMATOT.2),2), Median = round(median(HAMATOT.2),2),
  Min=round( min(HAMATOT.2),2),Max=round( max(HAMATOT.2),2))%>%
  rbind(avglogHAMA, avg)

#how many unique families
length(unique(Data.set$FAMILYID))
#how many unique individuals
length(unique(Data.set$ID))

SEX<-Data.set%>% group_by(SEX)%>%count(SEX)%>%
  summarize(Frequency=n, Percent=(n/186)*100)%>%
  mutate(SEX=case_when(SEX==1~"Male",
                       TRUE ~ "Female"))

LI<-Data.set%>%group_by(LITHRESP)%>% count(LITHRESP)%>%
  summarize(Frequency=n, Percent=(n/186)*100) %>%
  mutate(LITHRESP=case_when(LITHRESP==0~"No",
                       TRUE ~ "Yes"))

SES<-Data.set%>%
  mutate(SES=case_when(PARENTSES_1==5~"5",
                       PARENTSES_1==4~"4",
                       TRUE ~ "1,2,3"))%>%
```

```
  group_by(SES)%>% count(SES)%>%
  summarize(Frequency=n, Percent=(n/186)*100)

ParentsOnset<-Long.2%>%drop_na(HAMATOT.2)%>%
distinct(ID, .keep_all=TRUE)%>%
  summarize(" "="Parents Onset", Mean=round(mean(PARONSAGE),2),
  SD=round(sd(PARONSAGE),2), Median = round(median(PARONSAGE),2),
  Min=round( min(PARONSAGE),2),Max=round( max(PARONSAGE),2),n())

FirstInterview<-Data.set%>%
  summarize(" "="Age of First Interview", Mean=round(mean(HAMAAGE_1),2),
  SD=round(sd(HAMAAGE_1),2), Median = round(median(HAMAAGE_1),2),
  Min=round( min(HAMAAGE_1),2),Max=round( max(HAMAAGE_1),2),n())%>%
  rbind(LastInterview,YearsFollowed,ParentsOnset)

YearsFollowed<-Data.set%>%
  select(HAMAAGE_1,OUTCOME1BAGE,ID)%>%
  mutate(YearsFollowed=OUTCOME1BAGE-round(HAMAAGE_1,2))%>%
  distinct(ID, .keep_all=TRUE)%>%
  summarize(" "="Years Followed", Mean=round(mean(YearsFollowed),2),
  SD=round(sd(YearsFollowed),2), Median = round(median(YearsFollowed),2),
  Min=round( min(YearsFollowed),2),Max=round( max(YearsFollowed),2),n())

#histograms for HAMA
mean(Long.2$HAMATOT.2) #5.80
mean(Long.2$logHAMA) #1.578

p1<-ggplot(Long.2, aes(x=HAMATOT.2)) +
  geom_histogram (color="black" , fill="#A6836F", binwidth =2) +
  labs( x="HAMA score" , y= "Frequency") + theme_bw()+
  geom_vline(xintercept = 5.80 ,
  color="#400303", size=1, linetype = "dashed")

p2<-ggplot(Long.2, aes(x=logHAMA)) +
```

```
  geom_histogram (color="black" , fill="#400303", binwidth =0.3) +
  labs( x="Log transformed HAMA score" , y= "Frequency") + theme_bw()
  +geom_vline(xintercept = 1.578, color="#BFBCAA",size=1,
  linetype = "dashed")

plot_grid(p1, p2 ,labels = c('A', 'B'))


#QQ plots for HAMA

#lme models
lmeFit.hl <- lme(logHAMA ~ HAMAAGE + factor(SEX)+ factor(SES) +
                factor(LITHRESP),
                 random = ~ 1 | ID, data = Long.2)

lmeFit.h <- lme(HAMATOT.2 ~HAMAAGE + factor(SEX)+ factor(SES) +
                factor(LITHRESP),
                random = ~ 1 | ID, data = Long.2)

#residuals
resids.h<-data.frame(resids=residuals(lmeFit.h))
resids.hl<-data.frame(resids=residuals(lmeFit.hl))

#QQplots
p1<-ggplot(resids.h, aes(sample = resids))+ stat_qq(color="#A6836F") +
stat_qq_line(color="black") +labs( x="Theoretical quartiles" ,
y= "Sample quartiles")
+theme_bw()

p2<-ggplot(resids.hl, aes(sample = resids))+ stat_qq(color="#400303") +
stat_qq_line(color="#BFBCAA") +labs( x="Theoretical quartiles" ,
y= "Sample quartiles")
+theme_bw()
```

```
plot_grid(p1, p2 ,labels = c('A', 'B'))

#histograms
p1<-ggplot(resids.h, aes(x=resids)) +
  geom_histogram (color="black" , fill="#A6836F", binwidth =2) +
  labs( x="LME residuals" , y= "Frequency") + theme_bw()

p2<-ggplot(resids.hl, aes(x=resids)) +
  geom_histogram (color="black" , fill="#400303", binwidth =0.3) +
  labs( x="LME residuals with log transformed HAMA" , y= "Frequency") +
  theme_bw()

plot_grid(p1, p2 ,labels = c('A', 'B'))

#spaghetti plots
momsspaghetti.1= ggplot(Long.2, aes( x=HAMAAGE , y=HAMATOT.2,
color=factor(ID))) +
  geom_line(show.legend = FALSE) + theme_bw()+
  labs( y="HAMA scores",x="Age" )

momsspaghetti.2= ggplot(Long.2, aes( x=HAMAAGE , y=logHAMA,
color=factor(ID))) +
    geom_line(show.legend = FALSE) + theme_bw()+
  labs( y="Log transformed HAMA scores",x="Age" )

plot_grid(momsspaghetti.1, momsspaghetti.2 ,labels = c('A', 'B'))

#cant have MASS load before i clean
library(JM)

#four joint models

lmeFit.hle <- lme(logHAMA ~ HAMAAGE,
                random = ~ 1 | ID, data = Long.2)
```

```
lmeFit.h <- lme(logHAMA ~ bs(HAMAAGE,degree=2),
                random = ~ 1 | ID, data = Long.2)

lmeFit.bk <- lme(logHAMA ~ bs(HAMAAGE,
                 knots=c(15.44722, 20.21111, 25.27917)),
                 random = ~ 1 | ID, data = Long.2)

lmeFit.b <- lme(logHAMA ~ bs(HAMAAGE),
                random = ~ 1 | ID, data = Long.2)



# Cox PH model for the survival model.
coxFit <- coxph(Surv(OUTCOME1BAGE,OUTCOME1B) ~ factor(SEX)+  factor(SES) +
factor(LITHRESP), data = Data.set, x=TRUE)

### JM ###
jointFit.hle <- jointModel(lmeFit.hle, coxFit,
                           timeVar = "HAMAAGE")

jointFit.h <- jointModel(lmeFit.h, coxFit,
                         timeVar = "HAMAAGE")

jointFit.b <- jointModel(lmeFit.b, coxFit,
                        timeVar = "HAMAAGE")

jointFit.bk <- jointModel(lmeFit.bk, coxFit,
                        timeVar = "HAMAAGE")



#prediction plots
##plots of model predictions
#linear
```

```
pframe <- data.frame(HAMAAGE=seq(min(Long.2$HAMAAGE), max(Long.2$HAMAAGE),
length.out=40))
pframe$logHAMA <- predict(jointFit.hle, newdata = pframe, level = 0)
p1<-ggplot(Long.2, aes(HAMAAGE, logHAMA)) +
  geom_point(color="#A6836F") +
  geom_line(data=pframe)+
  labs( x="Age" , y= "log Transformed HAMA score")+theme_bw()

#bspline
pframe <- data.frame(HAMAAGE=seq(min(Long.2$HAMAAGE), max(Long.2$HAMAAGE),
length.out=40))
pframe$logHAMA <- predict(jointFit.bk, newdata = pframe, level = 0)
p3<-ggplot(Long.2, aes(HAMAAGE, logHAMA)) +
  geom_point(color="#A6836F") +
  geom_line(data=pframe)+
  labs( x="Age" , y= "log Transformed HAMA score") +
  geom_vline(xintercept = c(15.44722 ,20.21111, 25.27917), color="#400303",
  size=1)+theme_bw()



#quadratic
pframe <- data.frame(HAMAAGE=seq(min(Long.2$HAMAAGE), max(Long.2$HAMAAGE),
length.out=100))
pframe$logHAMA <- predict(jointFit.h, newdata = pframe, level = 0)
p2<-ggplot(Long.2, aes(HAMAAGE, logHAMA)) +
  geom_point(color="#A6836F") +
  geom_line(data=pframe)+
  labs( x="Age" , y= "log Transformed HAMA score") +theme_bw()

#cubic
pframe <- data.frame(HAMAAGE=seq(min(Long.2$HAMAAGE), max(Long.2$HAMAAGE),
length.out=100))
pframe$logHAMA <- predict(jointFit.b, newdata = pframe, level = 0)
```

```
p4<-ggplot(Long.2, aes(HAMAAGE, logHAMA)) +
  geom_point(color="#A6836F") +
  geom_line(data=pframe)+
  labs( x="Age" , y= "log Transformed HAMA score") +theme_bw()

plot_grid(p1, p2,p4,p3, nrow=2 ,labels = c('A', 'B', 'C','D'))
```

A.2 Framingham Heart Study Analysis

```
dir='my_path'

#read in each of the files.
diastolic_case=read_xls(paste(dir,"case_diastolic1.xls",sep="/"),
col_names=FALSE,
.name_repair = ~ LETTERS[seq_along(.x)])
systolic_case=read_xls(paste(dir,"case_systolic1.xls",sep="/"),
col_names=FALSE,
.name_repair = ~ LETTERS[seq_along(.x)])
age_case= read_xls(paste(dir,"case_age1.xls",sep="/") ,
col_names=FALSE,
.name_repair = ~ LETTERS[seq_along(.x)])

diastolic_cont=read_xls(paste(dir,"control_diastolic1.xls",sep="/"),
col_names=FALSE,
.name_repair = ~ LETTERS[seq_along(.x)])
systolic_cont=read_xls(paste(dir,"control_systolic1.xls",sep="/"),
col_names=FALSE,
.name_repair = ~ LETTERS[seq_along(.x)])
age_cont=read_xls(paste(dir,"control_age1.xls" ,sep="/"),
col_names=FALSE,
.name_repair = ~ LETTERS[seq_along(.x)])

#now reshape the whole thing
```

```
## The control data is the censored data, ie the data that didnt have
the survival event.

# create an id variable
diastolic_case$id<- c(1:nrow(diastolic_case))
systolic_case$id<- c(1:nrow(diastolic_case))
age_case$id<- c(1:nrow(diastolic_case))

diastolic_cont$id<- c(nrow(diastolic_case)+1:nrow(diastolic_cont))
systolic_cont$id<- c(nrow(diastolic_case)+1:nrow(diastolic_cont))
age_cont$id<- c(nrow(diastolic_case)+1:nrow(diastolic_cont))

#create long tables
diastolic_case<-diastolic_case%>%pivot_longer(!id,values_to = "diastolic")
systolic_case<-systolic_case%>%pivot_longer(!id,values_to = "systolic")
age_case<-age_case%>%pivot_longer(!id,values_to = "age")

diastolic_cont<-diastolic_cont%>%pivot_longer(!id,values_to = "diastolic")
systolic_cont<-systolic_cont%>%pivot_longer(!id,values_to = "systolic")
age_cont<-age_cont%>%pivot_longer(!id,values_to = "age")

#merge the control and case data
data_list<-list(diastolic_case,systolic_case,age_case)

case<-data_list %>% reduce(inner_join, by = c("id", "name"))

#add in survival status
case$surv<-c(rep(1,nrow(case)))

data_list.2<-list(diastolic_cont,systolic_cont,age_cont)

cont<-data_list.2 %>% reduce(inner_join, by = c("id", "name"))
```

```
cont$surv<-c(rep(0,nrow(cont)))

#merge both the contol and case data
Framingham<-rbind(case,cont)

`%nin%`= Negate(`%in%`)

Framingham<-Framingham%>% #cant have 0 pressure on either end
  filter(id %nin% c(725,995))

# calculate the survival time and elapsed time
avgtimefollow<-Framingham%>%
  group_by(id)%>%
  mutate(add=age-lag(age))%>%
  mutate(avgtime= case_when(name=="A"~ 0,
         TRUE ~ add),
         lags=lag(age))%>%
  mutate(etime = case_when (name=="A"~ avgtime,
    TRUE ~ cumsum(avgtime)))%>%ungroup(id)

entry<- Framingham%>%group_by(id)%>%
filter(name%in% c("A"))%>%mutate(entry=age)%>%drop_na()%>%
select(c(id,entry))

exit<- Framingham%>%group_by(id)%>%
  filter(name%in% c("G"))%>%mutate(exit=age)%>%drop_na()%>%
  select(c(id,exit,surv))

results<-merge(x=entry, y=exit, by="id",all.x=TRUE)

results$survtime<-(results$exit-results$entry)+0.5

test<- left_join(avgtimefollow,results, by=c("id","surv"))
```

```
#plots
#histogram for systolic

p3<-ggplot(Framingham, aes(x=systolic)) +
  geom_histogram (color="black" , fill="#A6836F", binwidth =15) +
  labs( x="Systolic pressure" , y= "Frequency")+ theme_bw()+
  geom_vline(xintercept = 124.12, color="#400303",size=1,
  linetype = "dashed")

#QQ plots for systolic

lmeFit.f <- lme(systolic ~  etime,
                random = ~ 1 | id, data = avgtimefollow)

resids.f<-data.frame(resids=residuals(lmeFit.f))

p4<-ggplot(resids.f, aes(sample = resids))+ stat_qq(color="#A6836F") +
stat_qq_line(color="black") +labs( x="Sample quartiles" ,
y= "Theoretical quartiles")+theme_bw()

plot_grid(p3, p4 ,labels = c('A', 'B'))

subset<-sample(unique(Framingham$id),100,replace=F)
subset.data<-Framingham%>%
  filter(id %in% subset)

#spaghetti plots
momsspaghetti= ggplot(subset.data, aes( x=age , y=systolic,
color=factor(id))) +
  geom_line(show.legend = FALSE) + theme_bw()+
 labs( y="Systolic blood pressure", x="Age" )

timeplot<-ggplot(time, aes(x=avgtime)) +
  geom_histogram (color="black" , fill="#73564C", binwidth =1) +
```

```
  labs( x="Time between measurements (years)" , y= "Frequency")+
  theme_bw()+
  geom_vline(xintercept = 4.37, color="#BFBCAA",size=1,
  linetype = "dashed")

#spaghetti plots
momsspaghetti= ggplot(subset.data, aes( x=age , y=systolic,
color=factor(id))) +
  geom_line(show.legend = FALSE) + theme_bw()+
 labs( y="Systolic blood pressure", x="Age" )

plot_grid(momsspaghetti,timeplot,labels=c("A","B"))

#Descriptive statistics
ages<-Framingham%>%
summarize(Mean=round(mean(age),2),SD=round(sd(age),2),
median=median(age),Min=min(age),Max=max(age))

counting<-Framingham%>%
# confirm all individuals have 7 obs.
  count(id)%>% distinct(n)

avgdiastolic<-Framingham%>%
  summarize(" "="diastolic pressure", Mean=round(mean(diastolic),2),
  SD=round(sd(diastolic),2), Median = round(median(diastolic),2),
  Min=round( min(diastolic),2),Max=round( max(diastolic),2))

avgsystolic<-Framingham%>%
  summarize(" "="systolic pressure", Mean=round(mean(systolic),2),
  SD=round(sd(systolic),2), Median = round(median(systolic),2),
  Min=round( min(systolic),2),Max=round( max(systolic),2))%>%
  rbind(avgslogsystolic,avgdiastolic)

  time<-Framingham%>%
```

```
  group_by(id)%>% mutate(avgtime=age-lag(age))%>% drop_na()%>%
  summarize(Mean=mean(avgtime),SD=sd(avgtime),median=median(avgtime),
  Min=min(avgtime),Max=max(avgtime))

time.2<-Framingham%>% # between 1st and 2nd
  group_by(id)%>% mutate(avgtime=age-lag(age))%>%drop_na()%>%
  slice_head(n=1)%>%
  summarize(Mean=mean(avgtime),SD=sd(avgtime),median=median(avgtime),
  Min=min(avgtime),Max=max(avgtime))

time.3<-Framingham%>% # between 6 and 7
  group_by(id)%>%mutate(avgtime=age-lag(age))%>%drop_na()%>%
  slice_tail(n=1)%>%

time.4<-Framingham%>% #between and 2 obs
  group_by(id)%>%mutate(avgtime=age-lag(age))%>%drop_na()%>%
  group_by(name)%>%
  summarize(Mean=mean(avgtime),SD=sd(avgtime),median=median(avgtime),
  Min=min(avgtime),Max=max(avgtime))



#methods
#cant have MASS load before i clean
library(JM)

lmeFit.f <- lme(systolic ~  etime,
                random = ~ 1 | id, data = avgtimefollow)

lmeFit.fbq <- lme(systolic ~  bs(etime, degree=2),
                  random = ~ 1 | id, data = avgtimefollow)

lmeFit.fb <- lme(systolic ~  bs(etime),
                 random = ~ 1 | id, data = avgtimefollow)
```

```
lmeFit.fbk <- lme(systolic ~  bs(etime,
                knots=c(8,16,23)),
                random = ~ 1 | id, data = avgtimefollow)

# Cox PH model for the survival model.
fitSURV <- coxph(Surv(survtime,surv) ~ 1, data = results, x = TRUE)

### JM ###
jointFit.f <- jointModel(lmeFit.f, fitSURV,
                         timeVar = "etime")
### JM ###
jointFit.fb <- jointModel(lmeFit.fb, fitSURV,
                         timeVar = "etime")

jointFit.fbk <- jointModel(lmeFit.fbk, fitSURV,
                          timeVar = "etime")

jointFit.fbq <- jointModel(lmeFit.fbq, fitSURV,
                           timeVar = "etime")

#prediction plots
pframe.f <- data.frame(etime=seq(min(avgtimefollow$etime),
max(avgtimefollow$etime),
length.out=100))
pframe.f$systolic <- predict(jointFit.fb, newdata = pframe.f, level = 0)
p2<-ggplot(avgtimefollow, aes(etime,systolic)) +
  geom_point(color="#A6836F") +
  geom_line(data=pframe.f)+
  labs(x="Time Since Start of Study", y="Systolic Blood Pressure")
  +theme_bw()


pframe.f <- data.frame(etime=seq(min(avgtimefollow$etime),
```

```
max(avgtimefollow$etime),
length.out=100))
pframe.f$systolic <- predict(jointFit.f, newdata = pframe.f, level = 0)
p1<-ggplot(avgtimefollow, aes(etime,systolic)) +
  geom_point(color="#A6836F") +
  geom_line(data=pframe.f)+
  labs(x="Time Since Start of Study", y="Systolic Blood Pressure")
  +theme_bw()


pframe.f <- data.frame(etime=seq(min(avgtimefollow$etime),
max(avgtimefollow$etime),
length.out=100))
pframe.f$systolic <- predict(jointFit.fbk, newdata = pframe.f, level = 0)
p3<-ggplot(avgtimefollow, aes(etime,systolic)) +
  geom_point(color="#A6836F") +
  geom_line(data=pframe.f)+
  labs(x="Time Since Start of Study", y="Systolic Blood Pressure")
  geom_vline(xintercept = c(8,16,23) , color="#400303", size=1)
  +theme_bw()

pframe.f <- data.frame(etime=seq(min(avgtimefollow$etime),
max(avgtimefollow$etime),
length.out=100))
pframe.f$systolic <- predict(jointFit.fb, newdata = pframe.f, level = 0)
p4<-ggplot(avgtimefollow, aes(etime,systolic)) +
  geom_point(color="#A6836F") +
  geom_line(data=pframe.f)+
  labs(x="Time Since Start of Study", y="Systolic Blood Pressure")
  +theme_bw()


plot_grid(p1, p4,p2, p3, nrow=2 ,labels = c('A', 'B', 'C','D'))
```

A.3 Simulation Study Analysis

Code for the simulation study was adapted from Lowe [2020], French [2021] and Gaudet [2021]

```
### its a simulation
library(tidyverse)
library(survival)
library(JM)
library(nlme)
library(purrr)
simulation.perf = function() {
  safe_jointModel <- purrr::safely(.f = jointModel)
  set.seed(1000) #set the seed
  # initialize parameter
  iterations <- 1000 # number of iterations
  n <- 1000 # number of undividuals
  ni <- 7 # number of observations per individual
  beta0 <- 119.3 #intercept for longitudinal model
  beta1 <- 0.3 #covariate estimate
  sig <- 11.7 #SD for individual
  sigb <- 12.4  #SD for random effect b sqrt(153.5669)
  alpha <- 0.07 #association parameter estimate
  lambda <- 0.000005 #baseline hazard

  # vectors to save things
  coefficients <- matrix(ncol = 3, nrow = iterations)
  stderr <- matrix(ncol = 3, nrow = iterations)
  pvalue <- matrix(ncol = 3, nrow = iterations)
  AIC<-vector()

  coefficients.f <- matrix(ncol = 2, nrow = iterations)
  stderr.f <- matrix(ncol = 2, nrow = iterations)
  pvalue.f <- matrix(ncol = 2, nrow = iterations)
```

```
AIC.f<-vector()

number.indv<-vector()
systolic<-vector()
event.time<-vector()

Event.process<- matrix(ncol = 3, nrow = iterations)
Event.process.f<- matrix(ncol = 3, nrow = iterations)
Event.process.stderr<-matrix(ncol = 3, nrow = iterations)
Event.process.stderr.f<-matrix(ncol = 3, nrow = iterations)
Event.process.pval<- matrix(ncol = 3, nrow = iterations)
Event.process.pval.f<- matrix(ncol = 3, nrow = iterations)
D<-vector()
D.f<-vector()
sigma0<-vector()
sigma0.f<-vector()

coefficients.fb.2 <- matrix(ncol = 4, nrow = iterations)
stderr.fb.2 <- matrix(ncol = 4, nrow = iterations)
pvalue.fb.2 <- matrix(ncol = 4, nrow = iterations)
AIC.fb.2<-vector()
Event.process.fb.2<- matrix(ncol = 3, nrow = iterations)
Event.process.stderr.fb.2<-matrix(ncol = 3, nrow = iterations)
Event.process.pval.fb.2<- matrix(ncol = 3, nrow = iterations)
D.fb.2<-vector()
sigma0.fb.2<-vector()

coefficients.fb.3 <- matrix(ncol = 7, nrow = iterations)
stderr.fb.3 <- matrix(ncol = 7, nrow = iterations)
pvalue.fb.3 <- matrix(ncol = 7, nrow = iterations)
AIC.fb.3<-vector()
Event.process.fb.3<- matrix(ncol = 3, nrow = iterations)
Event.process.stderr.fb.3<-matrix(ncol = 3, nrow = iterations)
Event.process.pval.fb.3<- matrix(ncol = 3, nrow = iterations)
```

```
D.fb.3<-vector()
sigma0.fb.3<-vector()
r <- 1
work<-0
while (r <= iterations) {
  work=work+1
  time <- rep(seq(0 , ni - 1 , by = 1), n)
  df <-
    data.frame (individual = sort (rep(c(1:n), ni)), time = time)
  bi <- rep(rnorm(n, 0, sigb), rep(ni, n))
  eij <- rnorm(n * ni, 0, sig)
  yij <- beta0 + beta1 * time + bi + eij
  df$response = yij
  eventtime <- ((1 / (beta1 * alpha)) * log(1 - beta1 * alpha *
  log(runif(n, 0, 1))
  / (lambda * exp(alpha * (beta0 + unique(bi))))))
  df$eventtime <- rep(eventtime, rep(ni, n))
  df$event <- rep(1, n * ni)
  df <- df %>% filter(time < eventtime)
  surv <- df %>% distinct(individual, .keep_all = TRUE)
  print(head(surv))
  print(r)
  print(head(df))
  #fit models
  skip_to_next <- FALSE
  fitSURV <-coxph(Surv(eventtime, event) ~ 1, data = surv, x = TRUE)
  lmeFit.f <- try(lme(response ~  time,
                      random = ~ 1 | individual,
                      data = df))
  if (class(lmeFit.f) == "lme")  {
print("I knew")
jointFit.f <-
  safe_jointModel(lmeFit.f, fitSURV, scaleWB=1, timeVar = "time")
  # linear joint model
```

```
}
if(is.null(jointFit.f$error) &&
!is.nan(summary(jointFit.f$result)
$`CoefTable-Long`[7]))
{
  print("you")

  coefficients.f[r, 1] <- jointFit.f$result$coefficients$betas[1]
  coefficients.f[r,2]<-jointFit.f$result$coefficients$betas[2]
  AIC.f[r]<-summary(jointFit.f$result)$AIC
  pvalue.f[r,1]<-summary(jointFit.f$result)$`CoefTable-Long`[7]
  pvalue.f[r,2]<-summary(jointFit.f$result)$`CoefTable-Long`[8]
  stderr.f[r,1]<-summary(jointFit.f$result)$`CoefTable-Long`[3]
  stderr.f[r,2]<-summary(jointFit.f$result)$`CoefTable-Long`[4]
  Event.process.f[r,1]<-jointFit.f$result$coefficients$alpha
  Event.process.f[r,2]<-jointFit.f$result$coefficients$gammas
  Event.process.f[r,3]<-summary(jointFit.f$result)$`CoefTable-Event`[3]
  Event.process.stderr.f[r,1]<-summary(jointFit.f$result)
  $`CoefTable-Event`[4]
  Event.process.stderr.f[r,2]<-summary(jointFit.f$result)
  $`CoefTable-Event`[5]
  Event.process.stderr.f[r,3]<-summary(jointFit.f$result)
  $`CoefTable-Event`[6]
  Event.process.pval.f[r,1]<-summary(jointFit.f$result)
  $`CoefTable-Event`[10]
  Event.process.pval.f[r,2]<-summary(jointFit.f$result)
  $`CoefTable-Event`[11]
  Event.process.pval.f[r,3]<-summary(jointFit.f$result)
  $`CoefTable-Event`[12]
  D.f[r]<-summary(jointFit.f$result)$D
  sigma0.f[r]<-summary(jointFit.f$result)$sigma

  iter.indv<-df%>%group_by(individual)%>%summarise(n = n())
  number.indv<-c(number.indv,iter.indv$n)
```

```
systolic<-c(systolic, df$response)
# keep the systolic blood pressure and event times
#to make plots for their distributions
event.time<-c(event.time, df$eventtime)

lmeFit.fb <- try(lme(response ~  bs(time, degree = 2),
                     random = ~ 1 | individual,
                     data = df))
lmeFit.fb.2 <- try(lme(response ~  bs(time, degree = 3),
                       random = ~ 1 | individual,
                       data = df))
lmeFit.fb.3 <- try(lme(response ~  bs(time, degree=3,
knots=c(1.5, 3.0 ,4.5)),
                       random = ~ 1 | individual,
                       data = df))

if (class(lmeFit.fb) == "lme"){
  print("were")
  jointFit.fbk <-
    safe_jointModel(lmeFit.fb, fitSURV, scaleWB=1,  timeVar = "time")
    # spline joint model

}
if (is.null(jointFit.fbk$error) &&
(!is.nan(summary(jointFit.fbk$result)
$`CoefTable-Long`[10]))){
  print("trouble")

  print(r)
  coefficients[r, 1] <- jointFit.fbk$result$coefficients$betas[1]
  coefficients[r,2]<-jointFit.fbk$result$coefficients$betas[2]
  coefficients[r,3]<-jointFit.fbk$result$coefficients$betas[3]
  AIC[r]<-summary(jointFit.fbk$result)$AIC
```

```
  pvalue[r,1]<-summary(jointFit.fbk$result)$`CoefTable-Long`[10]
  pvalue[r,2]<-summary(jointFit.fbk$result)$`CoefTable-Long`[11]
  pvalue[r,3]<-summary(jointFit.fbk$result)$`CoefTable-Long`[12]
  stderr[r,1]<-summary(jointFit.fbk$result)$`CoefTable-Long`[4]
  stderr[r,2]<-summary(jointFit.fbk$result)$`CoefTable-Long`[5]
  stderr[r,3]<-summary(jointFit.fbk$result)$`CoefTable-Long`[6]
  Event.process[r,1]<-jointFit.fbk$result$coefficients$alpha
  Event.process[r,2]<-jointFit.fbk$result$coefficients$gammas
  Event.process[r,3]<-summary(jointFit.fbk$result)$`CoefTable-Event`[3]
  Event.process.stderr[r,1]<-summary(jointFit.fbk$result)
  $`CoefTable-Event`[4]
  Event.process.stderr[r,2]<-summary(jointFit.fbk$result)
  $`CoefTable-Event`[5]
  Event.process.stderr[r,3]<-summary(jointFit.fbk$result)
  $`CoefTable-Event`[6]
  Event.process.pval[r,1]<-summary(jointFit.fbk$result)
  $`CoefTable-Event`[10]
  Event.process.pval[r,2]<-summary(jointFit.fbk$result)
  $`CoefTable-Event`[11]
  Event.process.pval[r,3]<-summary(jointFit.fbk$result)
  $`CoefTable-Event`[12]
  D[r]<-summary(jointFit.fbk$result)$D
  sigma0[r]<-summary(jointFit.fbk$result)$sigma
  #name <- paste("df", r, sep = ".")
  #name.s <- paste("surv", r, sep = ".")
  #assign(name, df, envir = .GlobalEnv)
  #assign(name.s, surv, envir = .GlobalEnv)

}
if (class(lmeFit.fb.2) == "lme"){
  print("when")

  jointFit.fbk.2 <-
    safe_jointModel(lmeFit.fb.2, fitSURV, scaleWB=1,timeVar = "time")
```

```
    # spline joint model
}
if (is.null(jointFit.fbk.2$error) &&
(!is.nan(summary(jointFit.fbk.2$result)
$`CoefTable-Long`[14]))){
  print("you")
  print(r)
  coefficients.fb.2[r, 1] <- jointFit.fbk.2$result$coefficients$betas[1]
  coefficients.fb.2[r,2]<-jointFit.fbk.2$result$coefficients$betas[2]
  coefficients.fb.2[r,3]<-jointFit.fbk.2$result$coefficients$betas[3]
  coefficients.fb.2[r,4]<-jointFit.fbk.2$result$coefficients$betas[4]
  AIC.fb.2[r]<-summary(jointFit.fbk.2$result)$AIC
  pvalue.fb.2[r,1]<-summary(jointFit.fbk.2$result)$`CoefTable-Long`[13]
  pvalue.fb.2[r,2]<-summary(jointFit.fbk.2$result)$`CoefTable-Long`[14]
  pvalue.fb.2[r,3]<-summary(jointFit.fbk.2$result)$`CoefTable-Long`[15]
  pvalue.fb.2[r,4]<-summary(jointFit.fbk.2$result)$`CoefTable-Long`[15]
  stderr.fb.2[r,1]<-summary(jointFit.fbk.2$result)$`CoefTable-Long`[5]
  stderr.fb.2[r,2]<-summary(jointFit.fbk.2$result)$`CoefTable-Long`[6]
  stderr.fb.2[r,3]<-summary(jointFit.fbk.2$result)$`CoefTable-Long`[7]
  stderr.fb.2[r,4]<-summary(jointFit.fbk.2$result)$`CoefTable-Long`[8]
  Event.process.fb.2[r,1]<-jointFit.fbk.2$result$coefficients$alpha
  Event.process.fb.2[r,2]<-jointFit.fbk.2$result$coefficients$gammas
  Event.process.fb.2[r,3]<-
  summary(jointFit.fbk.2$result)$`CoefTable-Event`[3]
  Event.process.stderr.fb.2[r,1]<-
  summary(jointFit.fbk.2$result)$`CoefTable-Event`[4]
  Event.process.stderr.fb.2[r,2]<-
  summary(jointFit.fbk.2$result)$`CoefTable-Event`[5]
  Event.process.stderr.fb.2[r,3]<-
  summary(jointFit.fbk.2$result)$`CoefTable-Event`[6]
  Event.process.pval.fb.2[r,1]<-
  summary(jointFit.fbk.2$result)$`CoefTable-Event`[10]
  Event.process.pval.fb.2[r,2]<-
  summary(jointFit.fbk.2$result)$`CoefTable-Event`[11]
```

```
  Event.process.pval.fb.2[r,3]<-
  summary(jointFit.fbk.2$result)$`CoefTable-Event`[12]
  D.fb.2[r]<-summary(jointFit.fbk.2$result)$D
  sigma0.fb.2[r]<-summary(jointFit.fbk.2$result)$sigma
}
if (class(lmeFit.fb.3) == "lme"){
  print("walked")

  jointFit.fbk.3 <-
    safe_jointModel(lmeFit.fb.3, fitSURV, scaleWB=1, timeVar = "time")
    # spline joint model
}
if (is.null(jointFit.fbk.3$error) &&
(!is.nan(summary(jointFit.fbk.3$result)
$`CoefTable-Long`[23]))){
  print("in")
  print(r)
  coefficients.fb.3[r,1]<-jointFit.fbk.3$result$coefficients$betas[1]
  coefficients.fb.3[r,2]<-jointFit.fbk.3$result$coefficients$betas[2]
  coefficients.fb.3[r,3]<-jointFit.fbk.3$result$coefficients$betas[3]
  coefficients.fb.3[r,4]<-jointFit.fbk.3$result$coefficients$betas[4]
  coefficients.fb.3[r,5]<-jointFit.fbk.3$result$coefficients$betas[5]
  coefficients.fb.3[r,6]<-jointFit.fbk.3$result$coefficients$betas[6]
  coefficients.fb.3[r,7]<-jointFit.fbk.3$result$coefficients$betas[7]
  AIC.fb.3[r]<-summary(jointFit.fbk.3$result)$AIC
  pvalue.fb.3[r,1]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[22]
  pvalue.fb.3[r,2]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[23]
  pvalue.fb.3[r,3]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[24]
  pvalue.fb.3[r,4]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[25]
  pvalue.fb.3[r,5]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[26]
  pvalue.fb.3[r,6]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[27]
  pvalue.fb.3[r,7]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[28]
  stderr.fb.3[r,1]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[8]
  stderr.fb.3[r,2]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[9]
```

```
    stderr.fb.3[r,3]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[10]
    stderr.fb.3[r,4]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[11]
    stderr.fb.3[r,5]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[12]
    stderr.fb.3[r,6]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[13]
    stderr.fb.3[r,7]<-summary(jointFit.fbk.3$result)$`CoefTable-Long`[14]
    Event.process.fb.3[r,1]<-jointFit.fbk.3$result$coefficients$alpha
    Event.process.fb.3[r,2]<-jointFit.fbk.3$result$coefficients$gammas
    Event.process.fb.3[r,3]<-
    summary(jointFit.fbk.3$result)$`CoefTable-Event`[3]
    Event.process.stderr.fb.3[r,1]<-
    summary(jointFit.fbk.3$result)$`CoefTable-Event`[4]
    Event.process.stderr.fb.3[r,2]<-
    summary(jointFit.fbk.3$result)$`CoefTable-Event`[5]
    Event.process.stderr.fb.3[r,3]<-
    summary(jointFit.fbk.3$result)$`CoefTable-Event`[6]
    Event.process.pval.fb.3[r,1]<-
    summary(jointFit.fbk.3$result)$`CoefTable-Event`[10]
    Event.process.pval.fb.3[r,2]<-
    summary(jointFit.fbk.3$result)$`CoefTable-Event`[11]
    Event.process.pval.fb.3[r,3]<-
    summary(jointFit.fbk.3$result)$`CoefTable-Event`[12]
    D.fb.3[r]<-summary(jointFit.fbk.3$result)$D
    sigma0.fb.3[r]<-summary(jointFit.fbk.3$result)$sigma
  }
  r=r+1
}else {
  print("I've got a blank space baby")
  skip_to_next<-TRUE
}
if (skip_to_next) {
  print("and I'll write your name")
  next
}
} # end of loop
```

```
  coeffs<-data.frame(mean=c(mean(coefficients[,1]),mean(coefficients[,2]),
            mean(coefficients[,3])),sd=c(sd(coefficients[,1]),
            sd(coefficients[,2]),sd(coefficients[,3])))

  coeffs.f<-data.frame(mean=c(mean(coefficients.f[,1]),
  mean(coefficients.f[,2])),
            sd=c(sd(coefficients.f[,1]),sd(coefficients.f[,2])))

  indiv.count<-data.frame(mean=c(mean(number.indv)), sd=c(sd(number.indv)),
              min=c(min(number.indv)), max=c(max(number.indv)))

    surv.summ<-data.frame(mean=c(mean(systolic),mean(event.time)),
                sd=c(sd(systolic),sd(event.time)),
                     min=c(min(systolic),min(event.time)),
                     median=c(median(systolic),
                     median(event.time)), max=c(max(systolic),
                     max(event.time)))

  freq<-data.frame(table(number.indv))
  #frequency table of obs per individual
return(list(efficiency=(r-1)/work, work=work, summard.f= coeffs.f,
            summard=coeffs,coeff.f=coefficients.f, coeff=coefficients,
            Aic=AIC,Aic.f=AIC.f,r=r,pval=pvalue,pval.f=pvalue.f,
            std=stderr,
            std.f=stderr.f, indiv.count=indiv.count, freq=freq,
            Event.process.f=Event.process.f, Event.process=Event.process,
            Event.process.pval.f=Event.process.pval.f,
            Event.process.pval=Event.process.pval,
            Event.process.stderr=Event.process.stderr,
            Event.process.stderr.f=Event.process.stderr.f,
            systolic=systolic, event.time=event.time,
            surv.summ=surv.summ, coeffs.fb.2=coeffs.fb.2,
            coeffs.fb.3=coeffs.fb.3,coefficients.fb.2=coefficients.fb.2,
            coefficients.fb.3=coefficients.fb.3, stderr.fb.2=stderr.fb.2,
```

```
            stderr.fb.3=stderr.fb.3,pvalue.fb.2=pvalue.fb.2,
            pvalue.fb.3=pvalue.fb.3,
            Event.process.fb.2=Event.process.fb.2,
            Event.process.fb.3=Event.process.fb.3,
            Event.process.pval.fb.3=Event.process.pval.fb.3,
            Event.process.pval.fb.2=Event.process.pval.fb.2,
            Event.process.stderr.fb.3=Event.process.stderr.fb.3,
            Event.process.stderr.fb.2=Event.process.stderr.fb.2,
            AIC.fb.2=AIC.fb.2,AIC.fb.3=AIC.fb.3, sigma0=sigma0,
            sigma0.f=sigma0.f,
            sigma0.fb.2=sigma0.fb.2,sigma0.fb.3=sigma0.fb.3, D=D,
            D.f=D.f,
            D.fb.2=D.fb.2, D.fb.3=D.fb.3))
} # end of function

perfectly.fine<-simulation.perf()

#save output when using Sharcnet so you can read it
#into R on personal device
saveRDS(perfectly.fine, file="/home/vsilverm/Too.Late.RData")

#want to see the contents of the simulation
perfectly.fine

#read in results from sharcnet into R on personal device
perfectly.fine<- readRDS("my path/Too.Late.RData")

#create a vector with the frequency table

counted<-c(rep(1,tested.3$freq[1,2]), rep(2,tested.3$freq[2,2]),
rep(3, tested.3$freq[3,2]), rep(4, tested.3$freq[4,2]),
rep(5,tested.3$freq[5,2]),
rep(6,tested.3$freq[6,2]), rep(7,tested.3$freq[7,2])))
```

```
counted.d<-data.frame(counted)

plot.hist.3<-ggplot(counted.d ,aes(x=counted)) +
geom_histogram (color="black" ,
fill="#A6836F", binwidth =5) +
+ labs( x="Number of observations per individual" , y= "Frequency")
+ theme_bw()

etime<-data.frame(perfectly.fine$event.time)
plot.hist<-ggplot(etime, aes(x=tested.2.event.time))+
geom_histogram
(color="black" , fill="#A6836F", binwidth =5) +
+ labs( x="event time" , y= "Frequency")
+ theme_bw()

systol<-data.frame(perfectly.fine$systolic)
plot.hist.2<-ggplot(systol, aes(x=tested.3.systolic))+
geom_histogram (color="black" , fill="#731803", binwidth =5) +
+ labs( x="systolic blood pressure" , y= "Frequency") + theme_bw()

plot_grid(plot.hist.2, plot.hist,plot.hist.3, nrow=1 ,
labels = c('A', 'B','C'))

#descriptive statistics work for a single iteration
subset<-sample(unique(df.1$individual),50,replace=F)
subset.data.sim<-df.1%>%
  filter(individual%in% subset)

#spaghetti plot
momsspaghetti= ggplot(subset.data.sim, aes( x=time , y=response,
color=factor(individual))) +geom_line(show.legend = FALSE) + theme_bw()+
  labs( y="Systolic blood pressure", x="time" )

#histogram
```

```
ggplot(df.1, aes(x=response)) + geom_histogram
(color="black" , fill="#A6836F",
binwidth =5) + labs( x="systolic blood pressure" , y= "Frequency") +
theme_bw()

ggplot(df.1, aes(x=eventtime)) + geom_histogram (color="black" ,
fill="#A6836F", binwidth =1) + labs( x="event time" , y= "Frequency") +
theme_bw()

#tables
avgtable<-df.1%>%
  summarize(" "="systolic blood pressure", Mean=round(mean(response),2),
  SD=round(sd(response),2), Median = round(median(response),2),
  Min=round( min(response),2),Max=round( max(response),2))

avgtable.e<-df.1%>%
  summarize(" "="event time", Mean=mean(eventtime),
  SD=sd(eventtime), Median = median(eventtime),
  Min= min(eventtime),Max= max(eventtime))
```

A.4 Example SHARCNET code

```
#!/bin/bash
#SBATCH --time=04:15:00
#SBATCH -A <advisorsaccount>
#SBATCH --mem=2GB
#SBATCH --output=testingoutput.out
#SBATCH --mail-user=<youremail>
#SBATCH --mail-type=END
#SBATCH --mail-type=FAIL
module load gcc/9.3.0 r/4.1.2
export R_LIBS=~/local/R_LIBS/
R -f simstudy.sharc.R
```