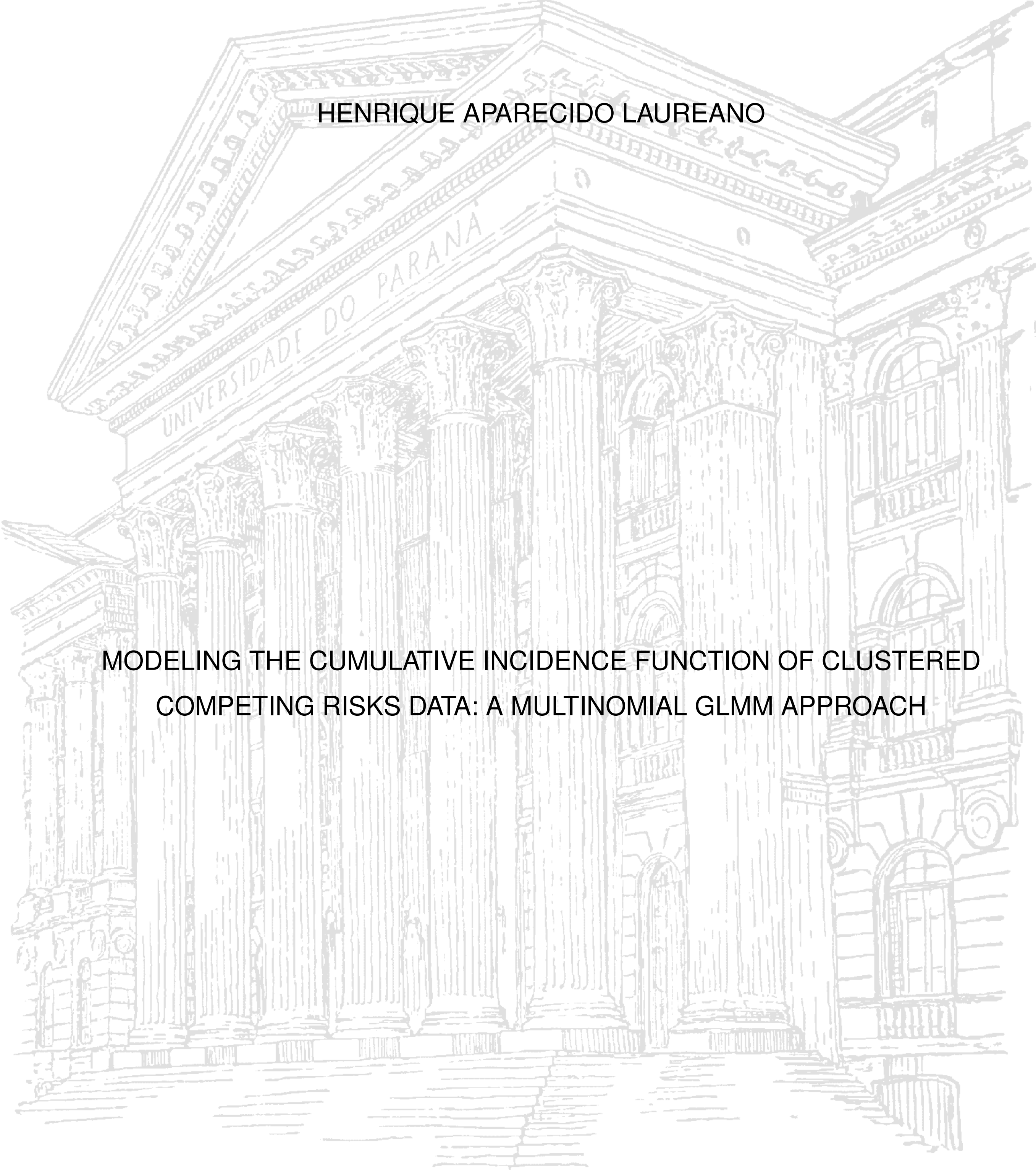
FEDERAL UNIVERSITY OF PARANÁ



HENRIQUE APARECIDO LAUREANO

MODELING THE CUMULATIVE INCIDENCE FUNCTION OF CLUSTERED COMPETING RISKS DATA: A MULTINOMIAL GLMM APPROACH

CURITIBA

2020

HENRIQUE APARECIDO LAUREANO

MODELING THE CUMULATIVE INCIDENCE FUNCTION OF CLUSTERED COMPETING RISKS DATA: A MULTINOMIAL GLMM APPROACH

Thesis presented to the Graduate Program of Numerical Methods in Engineering, Concentration Area in Mathematical Programming: Statistical Methods Applied in Engineering, Federal University of Paraná, as part of the requirements to the obtention of the Master′s Degree in Sciences.

Supervisor: Prof. PhD Wagner Hugo Bonat

Co-supervisor: Prof. PhD Paulo Justiniano Ribeiro Jr

CURITIBA

2020

HENRIQUE APARECIDO LAUREANO

MODELING THE CUMULATIVE INCIDENCE FUNCTION OF CLUSTERED COMPETING RISKS DATA: A MULTINOMIAL GLMM APPROACH

Thesis presented to the Graduate Program of Numerical Methods in Engineering, Concentration Area in Mathematical Programming: Statistical Methods Applied in Engineering, Federal University of Paraná, as part of the requirements to the obtention of the Master's Degree in Sciences.

Master thesis approved. XXX XX, 2020.

Prof. PhD Wagner Hugo Bonat
Supervisor

Prof. PhD Paulo Justiniano Ribeiro Jr
Co-supervisor

Prof. PhD . . .
Internal Examinator - PPGMNE

Prof. PhD . . .
Internal Examinator - PPGMNE

Prof. PhD . . .
External Examiner -

CURITIBA
2020

To Celita and Olivio

ACKNOWLEDGEMENTS

As Moro said once, I’m thankful for everything and everyone.

"It's not supposed to be easy."
(Gregg Popovich)

ABSTRACT

Failure time data …

Keywords: Competing risks.

RESUMO

Dados de tempos de falha …

Palavras-chave: Riscos competitivos.

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

CONTENTS

1 INTRODUCTION

Consider a cluster of random variables representing the time until the occurrence of some event. The random variables that compose that cluster are assumed to be correlated, i.e. for some biological or environmental reason it is not adequate to assume independence between them. Also, we may be interested in the occurrence of not only one specific event, having in practice a competition of events to see which one happens first, if it happens. Such events may also be of low probability but with severe consequences, that is the moment when the correlation ingredient makes its difference: the occurrence of an event in a subject should affect the probability of the same happening in the others.

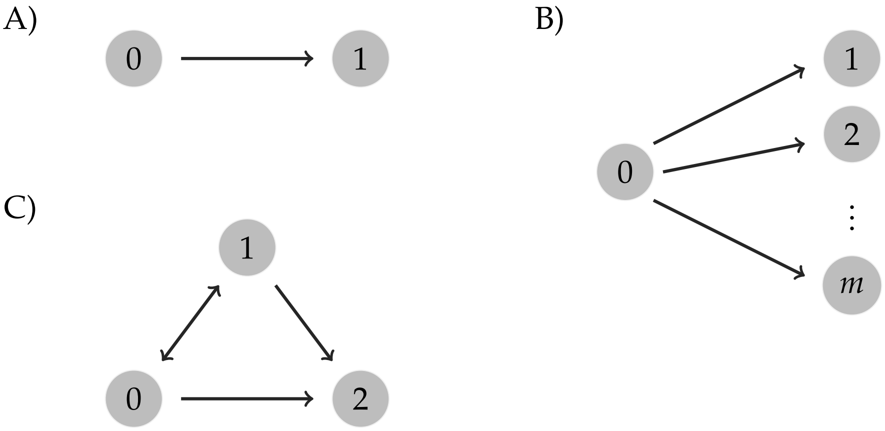
A realistic context that fits perfectly with the framework described above is the study of disease incidence in family members, where each member is indexed by a random variable and each cluster consists of a family. The inspiration to the study of these kinds of problems came from the work developed in Cederkvist et al. (2019), where they studied breast cancer incidence in mothers and daughters. Based on that, the aim of this thesis is to propose a simpler framework to make inference on the gender-specific cancer incidence in twins. The twins' case is just one more in the range of possible applications to our model, but given its intrinsic particularities, becomes the focus application. Until now we just contextualized, we still need to introduce the methodology. To this, some definitions and theoretical contexts are welcome.

When the object under study is random variables representing the time until some event occurs, we're in the field of *failure time data* (KALBFLEISCH; PRENTICE, 2002). The occurrence of an event is generally called as a *failure*, and major areas of application are biomedical studies and industrial life testing. In this thesis, we maintain our focus on the former.

As is common in science, same methodologies can receive different names in different areas. In industrial life testing applications is performed what is called a *reliability analysis*; in biomedical studies is performed what is called *survival analysis*. Generally, the term survival analysis is applied when we're interested in the occurrence of only one event, a *failure time process*. When we're interested in the occurrence of more than one event, like now, we enter in the yard of *competing risks* and *multistate* models. A visual aid is presented on Figure 1 and a comprehensive reference is Kalbfleisch & Prentice (2002).

Failure time and competing risk processes may be seen as particular cases of a multistate model. Besides the number of events (states) of interest, the main difference between a multistate model and its particular cases is that only in the multistate scenario we may have transient states, using a *stochastic process* language. In the particular cases, all the states besides the initial state 0, are absorbents - once you reached it you don't leave. The simplest multistate model that exemplify this behavior is the so-called illness-death model, Figure 1 C), i.e. a patient enters the study (state 0) and it can get sick (state 1) or die (state 2); if sick it can recover (returns to state 0) or die. We will work only with competing risk processes. For each individual, we have the time (age) until the occurrence, or not, of cancer.

FIGURE 1 – BEHAVIOR ILLUSTRATIONS OF MULTISTATE MODELS FOR A) FAILURE TIME PROCESS; B) COMPETING RISKS; AND C) ILNESS-DEATH MODEL, THE SIMPLEST MULTISTATE MODEL



SOURCE: The author (2020).

When for some know or unknown reason we are not able to see the occurrence of an event, we have what is called *censorship*. Still in the illness-death model: during the period of follow up the patient may not get sick or die, staying at state 0, this is called a *right-censorship*; The same for state 1. If a patient is in state 1 at the end of the study, we're *censored* to see him reaching the state 2 or returning to state 0. This is the inherent idea to censorship and must be present in the modeling framework, in this manner, arriving in the so-called *survival models*.

A survival model deals with survival experience. Usually, survival experience is modeled in the *hazard* (failure rate) scale and for an individual *i* can be written as

$$\lambda(t \mid \mathbf{x}_i) = \lambda_0(t) \times c(\mathbf{x}_i \boldsymbol{\beta}) \quad \text{at time } t, \tag{1.1}$$

i.e. a product of $\lambda_0(\cdot)$, an arbitrary baseline hazard function, and c, a specific function form that will depend on the chosen probability distribution for the failure time and on a covariates (explanatory/independent variables) vector $\mathbf{x}_i = [x_1 \ \dots \ x_p]$, where $\boldsymbol{\beta}^\top = [\beta_1 \ \dots \ \beta_p]$ is a vector of regression parameters. This structure is specified for a simple failure time process, as in Figure 1 A). Nevertheless, its idea is easy to extend.

We basically have the Equation 1.1 model for each cause-specific (in a competing risks process) or transition (in a multistate process). A complete and extensive detailing can be, again, found in Kalbfleisch & Prentice (2002).

In this work we approach the case of clustered competing risks. Besides the cause-specific structure, we have to deal with the fact that the events are happening in related individuals (twins). This configures what is called *family studies*, i.e. we have a cluster (a family/pair of twins) dependence that needs to be considered and modeled. This, possible, dependence is something that we do not actually measure, but know (or just suppose) that exists. In the statistical modeling language this characteristic receives the name of *random* or *latent* effect. A survival model with a latent effect, association, or unobserved heterogeneity, is called a *frailty model* (CLAYTON, 1978; VALPEL; MANTON; STALLARD, 1979). In its simplest form, a frailty is an unobserved random proportionality factor that modifies the hazard function of an individual, or of related individuals. Frailty models are extensions of the Equation 1.1 model.

In the competing risks setting, the hazard scale (focusing on the cause-specific hazard) is not the only possible scale to work on. A more attractive possibility is to work on the probability scale, focusing on the cause-specific cumulative incidence. Both may complement each other (ANDERSEN et al., 2012). However, in family studies there is often a strong interest in: describing age at disease onset and account for within-family dependence. The point to be made is that the distribution of age at disease onset is directly described by the cause-specific cumulative incidence. Therefore, the probability scale is the logical choice.

To work with competing risks data, on the probability scale, and with a latent effect structure that allows for within-cluster dependence of both risk and timing, Cederkvist et al. (2019) proposed a pairwise composite likelihood approach based on a linear model with multinomial response distribution and multivariate normal latent effects (in a frailty model the common choice for the latent effects is the gamma distribution). This approach sounds complicated and unnecessary, so we try to reach the same goal but with a much simpler framework, a generalized linear mixed model (GLMM). Instead of concentrating on failure time data and consequently having a survival/frailty model based on the hazard scale, or using a complicated composite approach, we just build the joint likelihood function (a multinomial model with a link function based on the cause-specific cumulative incidence function accouting for an appropriate latent effects structure), marginalize (integrate out the latent effects) and optimize it.

To better contextualize our GLMM approach, lets first just define it starting by the start. For a random individual i, in a standard linear model we assume that the response variable, Y_i, conditioned on the (possible) covariates follows a normal

(Gaussian) distribution and, what we do is to model its mean, $\mu_i \equiv \mathbb{E}(Y_i)$, via a linear combination. As much well explained in Nelder & Wedderburn (1972), with the support of a "link function", $g(\cdot)$, we are able to generalize this idea to non-Gaussian distributions. This extended framework received the name of generalized linear model (GLM) and is characterized by the following mean structure

$$g(\mu_i) = \mathbf{x}_i \boldsymbol{\beta},$$

where $\mathbf{x}_i$ is the i^{th} row of a model matrix $\mathbf{X}$, and $\boldsymbol{\beta}$ is a vector of unknown parameters. Also, in a GLM the Y_i are independent and

$$Y_i \sim \text{some exponential family distribution.}$$

The *exponential family* of distributions includes many distributions that are useful for practical modelling, such as the Poisson (for counting data), binomial (dichotomic data), gamma (continuous but positive) and normal (continuous data) distributions. A comprehensive reference for GLMs is McCullagh & Nelder (1989).

What makes a GLM into a GLMM is the addition of a latent (*m*ixed) effect, $\mathbf{u}$. The mean structure becomes

$$g(\mu_i) = \mathbf{x}_i \boldsymbol{\beta} + \mathbf{z}_i \mathbf{u}, \quad \mathbf{u} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$$

where the latent effect is assumed to follow a multivariate normal distribution of zero mean and a given variance-covariance matrix.

1.1 GOALS

1.1.1 General goals

Propose a multinomial generalized linear mixed model (multiGLMM) to the cause-specific cumulative incidence function (CIF) of clustered competing risks data.

1.1.2 Specific goals

1. Simulate from the multiGLMM to the cause-specific CIF, i.e. generate synthetic data from the model.

2. Write the model in the Template Model Builder (TMB) software, developed by Kristensen et al. (2016) and possibly the most efficient way of doing so, taking advantage of its functionalities: computation of all necessary gradients and Hessians via Automatic Differentiation (AD) and integration of the joint likelihood latent effects, marginalization, via an efficient Laplace approximation implementation.

3. Study the model identifiability through the proposition of models with different complexity levels in terms of parametric space.

4. Apply the model to synthetic data and a real-based dataset.

5. Compare the results of our multiGLMM to the cause-specific CIF with the results obtained with the pairwise composite approach of Cederkvist et al. (2019), the probably state-of-art to the study of clustered competing risks data.

1.2 JUSTIFICATION

In the biomedical statistical modeling literature, the examination of disease occurrence in related individuals receives the name of family studies. Key points of interest are the within-family dependence and determining the role of different risk factors. The within-family dependence may reflect both disease heritability and the impact of shared environmental effects. The role of different risk factors arrives in the class of multivariate models, which options are limited in the statistical literature. Thus, the number of statistical models for competing risks data that accommodate the within-cluster (family) dependence is even more limited. Some modeling options are briefly commented in Cederkvist et al. (2019), with his pairwise composite approach being proposed as a new and better option to model the cause-specific cumulative incidence function (CIF, that describes age at disease onset) of clustered competing risks data on the probability scale. We propose to model the cause-specific CIF and accommodate the within-family dependence in the same fashion (via a latent structure that allows the absolute risk and the failure time distribution to vary between families) but with an easier framework, based on a multinomial generalized linear mixed model (multiGLMM) approach.

1.3 LIMITATION

This work restraint to the proposition and application of a specific multinomial model for the cause-specific cumulative incidence function of competing risks data with a latent effect structure to accommodate within-family dependence with regard to both risk and timing. Given the elevated model complexity, hypothesis tests; residual analysis; and good-of-fit measures are not contemplated.

1.4 THESIS ORGANIZATION

This thesis contains 6 chapters including this introduction. Chapter 2 presents a systematic review of the main aspects involved in the construction and optimzation of a generalized linear mixed model (GLMM). Given the modeling framework overview, Chapter 3 presents our multinomial GLMM (multiGLMM) to model the cause-specific cumulative incidence function (CIF) of clustered competing risks data. In Chapter 4 we describes how to simulate some synthetic data from the proposed model, and presents a real-based dataset as an application. In Chapter 5 the obtained results are presented, and in Chapter 6 we discuss the contributions of this thesis and present some suggestions for future work.

2 GENERALIZED LINEAR MIXED MODELS: CONSTRUCTION AND OPTIMIZATION

This chapter presents a systematic review of the main theoretical aspects involved in the construction, estimation and implementation of a generalized linear mixed model (GLMM). We start in Section 2.1 with the model construction framework, concluding with the so-called joint likelihood function. Section 2.2 address the marginalization of that joint likelihood, performed here in terms of a Laplace approximation technique. Section 2.3 discusses available alternatives for the parameters optimization of the marginal likelihood obtained through that marginalization. Section 2.4 talks about automatic differentiation (AD), the most efficent manner of computing derivatives, and a key point for us. Last but not least, in Section 2.5 we present the computational tool used to peform all the discussed procedure, the TMB: Template Model Builder. A very exciting R (R DEVELOPMENT CORE TEAM, 2018) package developed by Kristensen et al. (2016).

2.1 CONSTRUCTION: JOINT LIKELIHOOD FUNCTION

A GLMM models an n-vector of exponential family random variables, $\mathbf{Y}$, in terms of its conditional expected value, $\boldsymbol{\mu} \equiv \mathbb{E}(\mathbf{Y} \mid \mathbf{X}, \mathbf{u})$, via a linear combination called of linear predictor and generally expressed by

$$g(\boldsymbol{\mu}) = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u}, \quad \mathbf{u} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}). \tag{2.1}$$

In other words, a GLMM is a generalized linear model (GLM) in which the linear predictor depends on some Gaussian latent effects, $\mathbf{u}$, times a latent effects model matrix $\mathbf{Z}$. Since we do not observe the latent component, an exemplification of the idea embedded in matrix $\mathbf{Z}$ is welcome. Suppose, e.g. three individuals (or clusters) and that each one has two measures. This configures a repeated measures context, the most common latent structure in family studies.. Also, it is reasonable to admit that each individual has its particular latent effect value. Consequently,

$$\mathbf{Z}\mathbf{u} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_1 \\ u_2 \\ u_2 \\ u_3 \\ u_3 \end{bmatrix},$$

where $\mathbf{u}^\top = [u_1 \; u_2 \; u_3]$ and $\mathbf{Z}$ has the role of projecting the values of $\mathbf{u}$ to match the number of measures.

In a mixed model the mean structure is approached into a combination of probability distributions. It is a combination since we have to assume probabilistic structures for the observed and non-observed (latent) data. To each observed variable y_{ij} we have a probability distribution of the exponential family, denoted by $f(y_{ij} \mid \mathbf{u}_i, \boldsymbol{\theta})$. To the non-observed latent effect we have, generally, a (multivariate) Gaussian distribution, denoted by $f(\mathbf{u}_i \mid \boldsymbol{\Sigma})$. To each individual or unity under study, i, and to each measure, j, we have the product of these probability densities, a likelihood contribution.

Our goal is to estimate the parameter vector $\boldsymbol{\theta} = [\boldsymbol{\beta}\ \boldsymbol{\Sigma}]^\top$ of a mean structure, as in Equation 2.1. Besides the role of emphasizing the fact that $\boldsymbol{\mu}$ is a function of $\boldsymbol{\theta}$ and that we want to estimate that $\boldsymbol{\theta}$, the likelihood function ties the probability densities. i.e. the likelihood is the product of the product of probability densities, to each subject i. Since Y_i are mutually independent, the likelihood for $\boldsymbol{\theta}$ can be written as

$$L(\boldsymbol{\theta} \mid \mathbf{y}, \mathbf{u}) = \prod_{i=1}^{I} \prod_{j=1}^{n_i} f(y_{ij} \mid \mathbf{u}_i, \boldsymbol{\beta}, \boldsymbol{\Sigma})\ f(\mathbf{u}_i \mid \boldsymbol{\Sigma}). \tag{2.2}$$

From standard probability theory is easy to see that in the right-hand side (r.h.s.) we have a joint density, consequently, Equation 2.2 represents what is called a joint likelihood function. What makes problematic working with this joint likelihood is that we do not have all the necessary information to just maximize it and get the desired parameter estimates. The latent effect $\mathbf{u}$ is *latent*, i.e. we do not observe it. To handle this we have basically two available paths.

2.2 MARGINALIZATION: LAPLACE APPROXIMATION

To deal with a joint likelihood function as in Equation 2.2 we have a choice to make. Be or not to be Bayesian. Each choice has its own difficulties, advantages, and characteristics.

The Bayesian path assumes that all $\boldsymbol{\theta}$ components are random variables. With all parameters being treated as random variables and, since we do not observe them, what the Bayesian framework does is try to compute the mode of each "parameter" marginal distribution, generally, via a sampling algorithm called MCMC: Markov chain Monte Carlo (GELFAND; SMITH, 1990; DIACONIS, 2009). The advantage of being Bayesian is that we can reach an MCMC algorithm to basically any statistical model, the disadvantage is that this approach is very time consuming and we have to propose prior distributions to each "parameter". These prior proposals are not always easy to make and, the resulting marginal distributions can be very depending of it.

A Bayesian approach can be applied in basically any context, without guarantees that will work - obtain convergence to all parameters is not a straightforward task. However, in complex scenarios they can be the only available method to "maximize" the likelihood function. This is not the case here. We have a joint density where one of the random variables is not observed, but we are not interested in it, only in the variance parameters inherent in it. Again, from standard probability theory, if we have a joint density we can just integrate out the undesired variable resulting in

$$\begin{aligned} L(\boldsymbol{\theta} \mid \mathbf{y}) &= \prod_{i=1}^{I} \int_{\mathcal{R}^{\mathbf{u}_i}} \left[\prod_{j=1}^{n_i} f(y_{ij} \mid \mathbf{u}_i, \boldsymbol{\beta}, \boldsymbol{\Sigma})\ f(\mathbf{u}_i \mid \boldsymbol{\Sigma}) \right] \mathrm{d}\mathbf{u}_i \\ &= \prod_{i=1}^{I} \int_{\mathcal{R}^{\mathbf{u}_i}} f(\mathbf{y}_i, \mathbf{u}_i \mid \boldsymbol{\theta})\ \mathrm{d}\mathbf{u}_i, \end{aligned} \tag{2.3}$$

a marginal density that keeps the parameters $\boldsymbol{\Sigma}$ of the integrated variable.

When the response distribution of a mixed model is Gaussian, is analytically tractable to integrate $\mathbf{u}$ out of the joint density. Consequently, it is possible to evaluate the marginal likelihood exactly. This is the case of the linear mixed models (LMMs) and the main difference to the GLMMs. When the response distribution is not Gaussian, generally, it is not anymore analytically tractable to integrate out the latent effect. So what do we do? Well, we have basically two options.

We can avoid the integrals in Equation 2.3 replacing it by integrals that are more analytically tractable. This can be performed via an algorithm called Expectation-Maximization (EM) proposed by Dempster, Laird & Rubin (1977). This approach is considered a little bit naive and generally is not recommended if you have a better option. The other option consists of performing a numerical integration, i.e. approximating the integral. The most common way of doing that in the statistical modeling literature is via an adaptive Gaussian quadrature rule (PINHEIRO; CHAO, 2006). In general, adaptive Gaussian quadratures are not so simple of using (may be unstable, computationally expensive and we have the problem of choosing how many integration points should be used). To us, the better option consists in take advantage of the exponential family structure and the fact that we are dealing with Gaussian latent effects. These ideas converge to an adaptive Gaussian quadrature with one integration point, also called as *Laplace approximation* (MOLENBERGHS; VERBEKE, 2005; SHUN; MCCULLAGH, 1995; TIERNEY; KADANE, 1986; WOOD, 2015).

With an integral that is analytically intractable, we may approximate it to obtain a tractable closed-form expression allowing then the numerical maximization of the resulting marginal likelihood function (BONAT; RIBEIRO, 2016). The Laplace approximation has been designed to approximate integrals in the form

$$\int_{\mathcal{R}^{\mathbf{u}_i}} \exp\{Q(\mathbf{u}_i)\} \mathrm{d}\mathbf{u}_i \approx (2\pi)^{n_{\mathbf{u}}/2} \left| Q''(\hat{\mathbf{u}}_i) \right|^{-1/2} \exp\{Q(\hat{\mathbf{u}}_i)\}, \tag{2.4}$$

where $Q(\mathbf{u}_i)$ is a known, unimodal bounded function and $\hat{\mathbf{u}}_i$ is the value for which $Q(\mathbf{u}_i)$ is maximized. As Wood (2015) shows, a Laplace approximation consists of a second order Taylor expansion of $\log f(\mathbf{y}_i, \mathbf{u}_i \mid \boldsymbol{\theta})$, about $\hat{\mathbf{u}}_i$, that gives

$$\log f(\mathbf{y}_i, \mathbf{u}_i \mid \boldsymbol{\theta}) \approx \log f(\mathbf{y}_i, \hat{\mathbf{u}}_i \mid \boldsymbol{\theta}) - \frac{1}{2}(\mathbf{u}_i - \hat{\mathbf{u}}_i)^\top \mathbf{H}\ (\mathbf{u}_i - \hat{\mathbf{u}}_i),$$

where $\mathbf{H} = -\nabla_u^2 \log f(\mathbf{y}_i, \hat{\mathbf{u}}_i \mid \boldsymbol{\theta})$. Hence, we can approximate the joint by

$$f(\mathbf{y}_i, \mathbf{u}_i \mid \boldsymbol{\theta}) \approx f(\mathbf{y}_i, \hat{\mathbf{u}}_i \mid \boldsymbol{\theta})\ \exp\left\{-\frac{1}{2}(\mathbf{u}_i - \hat{\mathbf{u}}_i)^\top \mathbf{H}\ (\mathbf{u}_i - \hat{\mathbf{u}}_i)\right\}. \tag{2.5}$$

From here we start to take advantage of the points mentioned above. First, the fact that we are dealing with Gaussian distributed latent effects. In Equation 2.5 we have the core of a Gaussian density, that complete is

$$\int_{\mathcal{R}^{\mathbf{u}_i}} \frac{1}{(2\pi)^{n_\mathbf{u}/2}\ |\mathbf{H}^{-1}|^{1/2}}\ \exp\left\{-\frac{1}{2}(\mathbf{u}_i - \hat{\mathbf{u}}_i)^\top \mathbf{H}\ (\mathbf{u}_i - \hat{\mathbf{u}}_i)\right\} \mathrm{d}\mathbf{u}_i = 1,$$

i.e. integrates to 1. Integrating Equation 2.5 follows that

$$\begin{aligned}\int_{\mathcal{R}^{\mathbf{u}_i}} f(\mathbf{y}_i, \mathbf{u}_i \mid \boldsymbol{\theta})\ \mathrm{d}\mathbf{u}_i &\approx f(\mathbf{y}_i, \hat{\mathbf{u}}_i \mid \boldsymbol{\theta}) \int_{\mathcal{R}^{\mathbf{u}_i}} \exp\left\{-\frac{1}{2}(\mathbf{u}_i - \hat{\mathbf{u}}_i)^\top \mathbf{H}\ (\mathbf{u}_i - \hat{\mathbf{u}}_i)\right\} \mathrm{d}\mathbf{u}_i \\ &= (2\pi)^{n_\mathbf{u}/2}\ |\mathbf{H}|^{-1/2}\ f(\mathbf{y}_i, \hat{\mathbf{u}}_i \mid \boldsymbol{\theta}),\end{aligned}$$

i.e. we get Equation 2.4, a first order Laplace approximation to the integral. Careful accounting of the approximation error shows it to generally be $\mathcal{O}(n^{-1})$, where n is the sample size and assuming a fixed length for $\mathbf{u}_i$ (WOOD, 2015).

The second advantage of a Laplace approximation approach in a GLMM is the exponential family structure. In a usual GLMM, the response follows a one-parameter exponential family distribution that can be written as

$$f(\mathbf{y}_i \mid \mathbf{u}_i, \boldsymbol{\theta}) = \exp\left\{\mathbf{y}_i^\top (\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i) - \mathbf{1}_i^\top b(\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i) + \mathbf{1}_i^\top c(\mathbf{y}_i)\right\},$$

where $b(\cdot)$ and $c(\cdot)$ are known functions. This general and easy to compute expression, together with a (multivariate) Gaussian distribution, highlights the convenience of the Laplace method. The $Q(\mathbf{u}_i)$ function to be maximized can be expressed as

$$\begin{aligned}Q(\mathbf{u}_i) = \mathbf{y}_i^\top (\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i) - \mathbf{1}_i^\top b(\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i) + \mathbf{1}_i^\top c(\mathbf{y}_i) \\ - \frac{n_\mathbf{u}}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}\mathbf{u}_i^\top \boldsymbol{\Sigma}^{-1}\ \mathbf{u}_i.\end{aligned} \tag{2.6}$$

The approximation in Equation 2.4 requires the maximum $\hat{\mathbf{u}}_i$ of the function $Q(\mathbf{u}_i)$. As we assume a Gaussian distribution with a known mean for the latent effects, we have the perfect initial guess for a gradient-based maximization method as the Newton-Raphson (NR) algorithm. The NR method consists of an iterative scheme as follows:

$$\mathbf{u}_i^{(k+1)} = \mathbf{u}_i^{(k)} - Q''(\mathbf{u}_i^{(k)})^{-1}\ Q'(\mathbf{u}_i^{(k)}), \quad k = 0,\ 1,\ \ldots$$

until convergence, which gives $\hat{\mathbf{u}}_i$. At this stage, all parameters $\boldsymbol{\theta}$ are considered known. Bonat & Ribeiro (2016) presents the generic expressions for the derivatives required by the NR method, given by the following:

$$
\begin{aligned}
Q'(\mathbf{u}_i^{(k)}) &= \{\mathbf{y}_i - b'(\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i^{(k)})\}^\top - \mathbf{u}_i^{(k)\top}\boldsymbol{\Sigma}^{-1},\\
Q''(\mathbf{u}_i^{(k)}) &= -\text{diag}\{b''(\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i^{(k)})\} - \boldsymbol{\Sigma}^{-1}.
\end{aligned}
$$

At $k = 0$ we have the initial guess.

Finally, the marginal log-likelihood function returned by the Laplace approximation, to each invividual or unit under study i, is as follows:

$$
\begin{aligned}
l(\boldsymbol{\theta} \mid \mathbf{y}_i) = \log L(\boldsymbol{\theta} \mid \mathbf{y}_i) = \frac{n}{2}\log(2\pi) - \frac{1}{2}\log\left|\text{diag}\{b''(\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\hat{\mathbf{u}}_i)\} + \boldsymbol{\Sigma}^{-1}\right|\\
+ \mathbf{y}_i^\top(\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\hat{\mathbf{u}}_i) - \mathbf{1}_i^\top b(\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\hat{\mathbf{u}}_i) + \mathbf{1}_i^\top c(\mathbf{y}_i)\\
- \frac{n_{\mathbf{u}}}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}\hat{\mathbf{u}}_i^\top\boldsymbol{\Sigma}^{-1}\,\hat{\mathbf{u}}_i,
\end{aligned}
$$

that can now be numerically maximized over the model parameters $\boldsymbol{\theta} = [\boldsymbol{\beta}\ \boldsymbol{\Sigma}]^\top$.

2.3 OPTIMIZATION: MARGINAL LIKELIHOOD FUNCTION

At this point it is already clear that we have two optimizations to be performed, an "inside" and an "outside" optimization. The inside one is made into the Laplace approximation layer via a Newton-Raphson algorithm, a Newton's method. The outside optimization is made with the Laplace approximation outputs, i.e. the maximization step Equation 2.3's marginal log-likelihood over its parameters $\boldsymbol{\theta}$. This task is usually performed via a quasi-Newton method, we focus on two of the most traditional ones: the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm and the PORT routines.

The inside optimization is the numerical maximization of the joint log-likelihood w.r.t. its latent effects. This is kind of a simple task since all model parameters are considered as fixed and we "know" that the latent effects are distributed with zero mean, i.e. we have the perfect initial guess. In this context, the use of a Newton's method is straightforward. When we talk about the outside optimization it is a completely different scenario, it is not straightforward to find a good initial guess or reach convergence. Thus, more robust methods are a good choice.

In optimization, Newton methods are algorithms for finding local maxima and minima of functions, i.e. the search for the zeroes of the gradient of that function. Newton methods are characterized by the use of a symmetric matrix of function's second derivatives, the Hessian matrix. Quasi-Newton methods are based on Newton's method and are seen as an alternative to it. They can be used if the Hessian is unavailable or if is too expensive to compute it at every iteration.

As shown in Nocedal & Wright (2006), major advantages of quasi-Newton methods over Newton's method are that the Hessian matrix does not need to be computed, it is approximated; and it also does not need to be inverted. Newton's method requires the Hessian to be inverted, typically by solving a system of linear equations - often quite costly. In contrast, quasi-Newton methods usually generate an estimate of it directly. As in Newton's method, they use a second-order approximation to find the minimum of a function $f(\mathbf{x})$. The Taylor series of $f(\mathbf{x})$ around an iterate is

$$f(\mathbf{x}_k + \Delta\mathbf{x}) \approx f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^\top \Delta\mathbf{x} + \frac{1}{2}\Delta\mathbf{x}^\top \mathbf{B}\,\Delta\mathbf{x},$$

where $\nabla f(\cdot)$ is the gradient, and $\mathbf{B}$ an approximation to the Hessian matrix. The gradient of this approximation w.r.t. $\Delta\mathbf{x}$ is

$$\nabla f(\mathbf{x}_k + \Delta\mathbf{x}) \approx \nabla f(\mathbf{x}_k) + \mathbf{B}\,\Delta\mathbf{x},$$

setting this gradient to zero provides the Newton step:

$$\Delta\mathbf{x} = -\mathbf{B}^{-1}\nabla f(\mathbf{x}_k).$$

The Hessian approximation $\mathbf{B}$ is chosen to satisfy

$$\nabla f(\mathbf{x}_k + \Delta\mathbf{x}) = \nabla f(\mathbf{x}_k) + \mathbf{B}\,\Delta\mathbf{x},$$

which is called the *secant* equation, i.e. the Taylor series of the gradient itself. Solving for $\mathbf{B}$ and applying the Newton's step with the updated value is equivalent to the *secant* method. Quasi-Newton methods are a generalization of the secant method to find the root of the first derivative for multidimensional problems. The various quasi-Newton methods differ in their choice of the solution to the secant equation.

In a general quasi-Newton method, the unknown $\mathbf{x}_k$ is updated applying the Newton's step calculated using the current approximate Hessian matrix $\mathbf{B}_k$ in the following fashion:

- $\Delta\mathbf{x}_k = -\alpha_k \mathbf{B}_k^{-1}\nabla f(\mathbf{x}_k)$, with α chosen to satisfy the so called Wolfe conditions (NOCEDAL; WRIGHT, 2006, p. 34);
- $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta\mathbf{x}_k$;
- The gradient computed at the new point $\nabla f(\mathbf{x}_{k+1})$, and $\mathbf{y}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k)$ is used to update the approximate Hessian $\mathbf{B}_{k+1}$, or directly its inverse $\mathbf{H}_{k+1} = \mathbf{B}_{k+1}^{-1}$.

The most popular quasi-Newton method is the BFGS algorithm, named for its discoverers Broyden, Fletcher, Goldfarb, and Shanno. It has the following update formula

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\mathbf{y}_k \mathbf{y}_k^\top}{\mathbf{y}_k^\top \Delta\mathbf{x}_k} - \frac{\mathbf{B}_k \Delta\mathbf{x}_k (\mathbf{B}_k \Delta\mathbf{x}_k)^\top}{\Delta\mathbf{x}_k^\top \mathbf{B}_k \Delta\mathbf{x}_k},$$

$$\mathbf{H}_{k+1} = \mathbf{B}_{k+1}^{-1} = \left(\mathbf{I} - \frac{\Delta\mathbf{x}_k \mathbf{y}_k^\top}{\mathbf{y}_k^\top \Delta\mathbf{x}_k}\right) \mathbf{H}_k \left(\mathbf{I} - \frac{\mathbf{y}_k \Delta\mathbf{x}_k^\top}{\mathbf{y}_k^\top \Delta\mathbf{x}_k}\right) + \frac{\Delta\mathbf{x}_k \Delta\mathbf{x}_k^\top}{\mathbf{y}_k^\top \Delta\mathbf{x}_k}.$$

Another quasi-Newton method, popular in the statistical modeling literature, is the one based on the PORT routines ⟨http://www.netlib.org/port/⟩. A Fortran mathematical subroutine library designed to be *portable* over different types of computers, and developed by David Gay in the Bell Labs (GAY, 1990). It is a quasi-Newton adaptive nonlinear least-squares algorithm (DENNIS; GAY; WELSCH, 1981) with the following update formula

$$\begin{aligned}\mathbf{B}_{k+1} = \mathbf{B}_k & \\ &+ \frac{(\mathbf{y}_k - \mathbf{B}_k \Delta\mathbf{x}_k)\, \Delta\mathbf{x}_k^\top \mathbf{B}_k + \mathbf{B}_k \Delta\mathbf{x}_k\, (\mathbf{y}_k - \mathbf{B}_k \Delta\mathbf{x}_k)^\top}{\Delta\mathbf{x}_k^\top \mathbf{B}_k \Delta\mathbf{x}_k} \\ &- \frac{\Delta\mathbf{x}_k^\top\, (\mathbf{y}_k - \mathbf{B}_k \Delta\mathbf{x}_k)\, \mathbf{B}_k \Delta\mathbf{x}_k \Delta\mathbf{x}_k^\top \mathbf{B}_k}{(\Delta\mathbf{x}_k^\top \mathbf{B}_k \Delta\mathbf{x}_k)^\top\, \Delta\mathbf{x}_k^\top \mathbf{B}_k \Delta\mathbf{x}_k}.\end{aligned}$$

As Nocedal & Wright (2006) points out, each quasi-Newton method iteration can be performed at a cost of $\mathcal{O}(n^2)$ arithmetic operations (plus the cost of function and gradient evaluations); there are no $\mathcal{O}(n^3)$ operations such as linear system solves or matrix-matrix operations. In the BFGS algorithm is known that the rate of convergence is superlinear, which is a valid assumption to any quasi-Newton method and is fast enough for most practical purposes. Even though Newton's method converges more rapidly, quadratically, its cost per iteration usually is higher, because of its need for second derivatives and solution of a linear system.

In this thesis, the used BFGS implementation is the one in the `R` (R DEVELOPMENT CORE TEAM, 2018) function `optim()`, and the PORT routine used is the one implemented in the `R` function `nlminb()`.

2.4 A DIFFERENTIAL: AUTOMATIC DIFFERENTIATION

Computing gradients, $\nabla f(\mathbf{x})$, are a fundamental and crucial task but also the main computational bottleneck to any Newton and quasi-Newton method. We choose to use the most efficient manner of computing gradients, one of the best scientific computing techniques but still not so famous in the statistical modeling literature, the *automatic differentiation* (AD) procedure. AD has two modes, the so-called forward and reverse mode. We will talk a bit about both but we will use only the reverse mode. The reason can be illustrated by a simple example, given later.

Automatic differentiation, also called algorithmic differentiation or computational differentiation, is a set of techniques to numerically and recursively evaluate the derivative of a function specified by a computer program. AD techniques are based on the observation that any function, no matter how complicated, is evaluated by performing a sequence of simple elementary operations involving just one or two arguments at a time. Derivatives of arbitrary order can be computed automatically, automatized and accurately to working precision. Most of the information in this section was taken of Peyré (2020), but Wood (2015, p. 120) and Nocedal & Wright (2006, p. 204) are also very good references.

The most common differentiation approaches are finite differences (FD) and symbolic calculus. Considering a function $f : \mathbb{R}^p \to \mathbb{R}$ and the goal of deriving a method to evaluate $\nabla f : \mathbb{R}^p \to \mathbb{R}^p$, the approximation of this vector field via FD would require $p+1$ evaluations of f. The same task via reverse mode AD has in most cases a cost proportional to a single evaluation of f. AD is similar to symbolic calculus in the sense that it provides an exact gradient computation, up to machine precision. However, symbolic calculus does not takes into account the underlying algorithm which compute the function, while AD factorizes the computation of the derivative according to an efficient algorithm.

The use of AD is inherent to the use of a computational graph, Figure 2. Assuming that f is implemented in an algorithm, the goal is to compute the derivatives

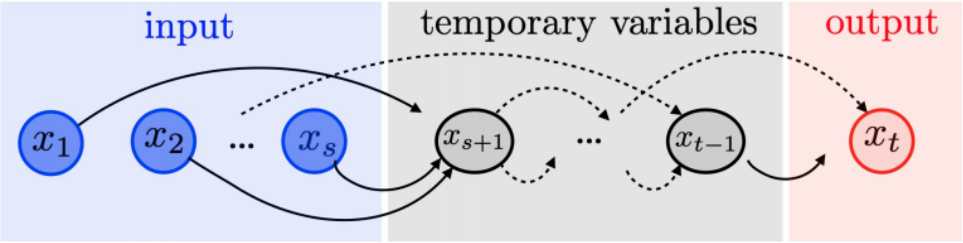
$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_k} \in \mathbb{R}^{n_t \times n_k},$$

for a numerical algorithm (succession of functions) of the form

$$\forall\, k = s+1, \ldots, t, \quad \mathbf{x}_k = f_k(\mathbf{x}_1, \ldots, \mathbf{x}_{k-1}),$$

where f_k is a function which only depends on the previous variables.

FIGURE 2 – A COMPUTATIONAL GRAPH



SOURCE: Peyré (2020, p. 31).

The computational graph, Figure 2, has the role of represent the linking of the variables involved in f_k to $\mathbf{x}_k$. The evaluation of $f(\mathbf{x})$ corresponds to a forward traversal of this graph. Now, how we evaluate f through the graph? Via one of the AD modes.

2.4.1 Forward Mode

The forward mode correspond to the usual way of computing differentials. The method initialize with the derivative of the input nodes

$$\frac{\partial \mathbf{x}_1}{\partial \mathbf{x}_1} = \mathrm{Id}_{n_1 \times n_1}, \quad \frac{\partial \mathbf{x}_2}{\partial \mathbf{x}_1} = \mathbf{0}_{n_2 \times n_1}, \quad \frac{\partial \mathbf{x}_s}{\partial \mathbf{x}_1} = \mathbf{0}_{n_s \times n_1},$$

and then iteratively make use of the following recursion formula

$$\begin{aligned}&\forall\, k = s+1, \ldots, t, \\ &\frac{\partial \mathbf{x}_k}{\partial \mathbf{x}_1} = \sum_{l \,\in\, \mathrm{father}(k)} \frac{\partial \mathbf{x}_k}{\partial \mathbf{x}_l} \times \frac{\partial \mathbf{x}_l}{\partial \mathbf{x}_1} = \sum_{l \,\in\, \mathrm{father}(k)} \frac{\partial}{\partial \mathbf{x}_l} f_k(\mathbf{x}_1, \ldots, \mathbf{x}_{k-1}) \times \frac{\partial \mathbf{x}_l}{\partial \mathbf{x}_1}.\end{aligned}$$

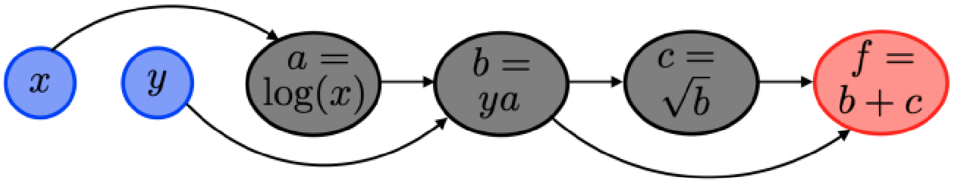
The notation “father(k)” denotes the nodes $l < k$ of the graph that are connected to k. We make use of Peyré (2020, p. 32)’s simple example.

Example. Consider the function

$$f(x,y) = y\log(x) + \sqrt{y\log(x)}$$

with the corresponding computational graph being displayed in Figure 3.

FIGURE 3 – EXAMPLE OF A SIMPLE COMPUTATIONAL GRAPH



SOURCE: Peyré (2020, p. 33).

The forward mode iterations to compute the derivative w.r.t. x, following the computational graph, is given by

$$\begin{aligned}
&\frac{\partial x}{\partial x} = 1, \quad \frac{\partial y}{\partial x} = 0 \\
&\frac{\partial a}{\partial x} = \frac{\partial a}{\partial x}\frac{\partial x}{\partial x} = \frac{1}{x}\frac{\partial x}{\partial x} && \{x \mapsto a = \log(x)\} \\
&\frac{\partial b}{\partial x} = \frac{\partial b}{\partial a}\frac{\partial a}{\partial x} + \frac{\partial b}{\partial y}\frac{\partial y}{\partial x} = y\frac{\partial a}{\partial x} + 0 && \{(y,a) \mapsto b = ya\} \\
&\frac{\partial c}{\partial x} = \frac{\partial c}{\partial b}\frac{\partial b}{\partial x} = \frac{1}{2\sqrt{b}}\frac{\partial b}{\partial x} && \{b \mapsto c = \sqrt{b}\} \\
&\frac{\partial f}{\partial x} = \frac{\partial f}{\partial b}\frac{\partial b}{\partial x} + \frac{\partial f}{\partial c}\frac{\partial c}{\partial x} = 1\frac{\partial b}{\partial x} + 1\frac{\partial c}{\partial x} && \{(b,c) \mapsto f = b + c\}
\end{aligned}$$

To compute the derivative w.r.t. y we run another forward process

$$\begin{aligned}
&\frac{\partial x}{\partial y} = 0, \quad \frac{\partial y}{\partial y} = 1 \\
&\frac{\partial a}{\partial y} = \frac{\partial a}{\partial x}\frac{\partial x}{\partial y} = 0 && \{x \mapsto a = \log(x)\} \\
&\frac{\partial b}{\partial y} = \frac{\partial b}{\partial a}\frac{\partial a}{\partial y} + \frac{\partial b}{\partial y}\frac{\partial y}{\partial y} = 0 + a\frac{\partial y}{\partial y} && \{(y,a) \mapsto b = ya\} \\
&\frac{\partial c}{\partial y} = \frac{\partial c}{\partial b}\frac{\partial b}{\partial y} = \frac{1}{2\sqrt{b}}\frac{\partial b}{\partial y} && \{b \mapsto c = \sqrt{b}\} \\
&\frac{\partial f}{\partial y} = \frac{\partial f}{\partial b}\frac{\partial b}{\partial y} + \frac{\partial f}{\partial c}\frac{\partial c}{\partial y} = 1\frac{\partial b}{\partial y} + 1\frac{\partial c}{\partial y} && \{(b,c) \mapsto f = b + c\}
\end{aligned}$$

2.4.2 Reverse Mode

Instead of evaluating the differentials for all the input nodes, which is problematic for a large number of nodes, the reverse mode evaluates the differentials of the output node w.r.t. all the inner nodes.

The method initialize with the derivative of the final node

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_t} = \mathrm{Id}_{n_y \times n_y},$$

and then, from the last to the first node, iteratively make use of the following recursion formula

$$\begin{aligned}
&\forall\, k = t-1, t-2, \ldots, 1, \\
&\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \sum_{m\,\in\,\mathrm{son}(k)} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_m} \times \frac{\partial \mathbf{x}_m}{\partial \mathbf{x}_k} = \sum_{m\,\in\,\mathrm{son}(k)} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_m} \times \frac{\partial}{\partial \mathbf{x}_k} f_m(\mathbf{x}_1, \ldots, \mathbf{x}_m).
\end{aligned}$$

The notation "son(k)" denotes the nodes $m < k$ of the graph that are connected to k. To be clear, the same simple example.

Example. Consider, again, the function

$$f(x,y) = y\log(x) + \sqrt{y\log(x)}.$$

The iterations of the reverse mode is given by

$$\begin{aligned}
\frac{\partial f}{\partial f} &= 1 \\
\frac{\partial f}{\partial c} &= \frac{\partial f}{\partial f}\frac{\partial f}{\partial c} = \frac{\partial f}{\partial f}1 && \{c \mapsto f = b + c\} \\
\frac{\partial f}{\partial b} &= \frac{\partial f}{\partial c}\frac{\partial c}{\partial b} + \frac{\partial f}{\partial f}\frac{\partial f}{\partial b} = \frac{\partial f}{\partial c}\frac{1}{2\sqrt{b}} + \frac{\partial f}{\partial f}1 && \{b \mapsto c = \sqrt{b},\, b \mapsto f = b + c\} \\
\frac{\partial f}{\partial a} &= \frac{\partial f}{\partial b}\frac{\partial b}{\partial a} = \frac{\partial f}{\partial b}y && \{a \mapsto b = ya\} \\
\frac{\partial f}{\partial y} &= \frac{\partial f}{\partial b}\frac{\partial b}{\partial y} = \frac{\partial f}{\partial b}a && \{y \mapsto b = ya\} \\
\frac{\partial f}{\partial x} &= \frac{\partial f}{\partial a}\frac{\partial a}{\partial x} = \frac{\partial f}{\partial a}\frac{1}{x} && \{x \mapsto a = \log(x)\}
\end{aligned}$$

This is the advantage of reverse mode over the forward mode. A single traversal over the computational graph allows to compute both derivatives w.r.t. x, y, while the forward mode necessities two processes.

An drawback of the reverse mode is the need to store the entire computational graph, which is needed for the reverse sweep. In principle, storage of this graph is not too difficult to implement. However, the main benefit of AD is higher accuracy, and in many applications the cost is not critical.

2.5 TMB: TEMPLATE MODEL BUILDER

Note that the goal of AD is not to define an efficient computational graph, it is up to the user to provide it. However, computing an efficient graph associated to a mathematical formula is a complicated combinatorial problem. Thus, since our goal is to be able to fit our desired statistical models, a computational tool able to efficiently define and implement this computational graph is make necessary. To solve this and many other tasks, we have the Template Model Builder (TMB) (KRISTENSEN et al., 2016).

TMB ⟨http://tmb-project.org⟩ is an R (R DEVELOPMENT CORE TEAM, 2018) package for fitting statistical latent variable models to data, inpired by AD Model Builder (ADMB) (FOURNIER et al., 2012). ADMB is a statistical application for fitting nonlinear statistical models and solve optimization problems, that implements AD using C++ classes and a native template language. Unlike most R packages, in TMB the model is formulated in C++. This characteristic provides great flexibility, but requires some familiarity with the C/C++ programming language. With TMB a user should be able to quickly implement complex latent effect models through simple C++ templates.

In this chapter we describe step-by-step all the processes involved in the creation and parameter estimation of a GLMM. With the TMB, all this is put in practice in an efficient and robust fashion.

A user needs to provide just the joint likelihood function writing in a C++ template. If the model presents latent effects, during the compilation the latent effects will be integrated out via an efficient Laplace approximation routine, with a Newton algorithm inside, and the marginal log-likelihood gradient will be also computed. These marginal log-likelihood will be returned into an R object, that can then be optimized using the user's favorite quasi-Newton routine, available in R. To do all that, TMB combines some state of the art software

- CppAD, a C++ AD package ⟨https://coin-or.github.io/CppAD/⟩;
- Eigen (GUENNEBAUD; JACOB et al., 2010), a C++ templated matrix-vector library;
- CHOLMOD, sparse matrix routines available from R, used to obtain an efficient implementation of the Laplace approximation with exact derivatives ⟨https://developer.nvidia.com/cholmod⟩;
- Parallelism through BLAS: Basic Linear Algebra Subprograms ⟨http://www.netlib.org/blas/⟩.

Also, some of its key characteristics are

- TMB employs AD to calculate first and second order derivatives of the likelihood function or any objective function in C++;
- The objective function, and its derivatives, can be called from R. Hence, parameter estimation via `optim()` or `nlminb()` is easy to be performed;
- Standard deviations of any parameter, or derived parameter, can be obtained via the *delta method*.

Here we focus on GLMMs, but basically any statistical model with a latent structure (or not), linear (or not), can be fitted with TMB. In times of *big data*, and with the TMB's authors having a professional preference for state-of-space and spatial models, TMB has also automatic sparseness detection.and some other nice built tools. Pre and post-processing of data should be done in R.

A TMB Users' mailing list exists, and it is extremely helpful for taking doubts and questions ⟨https://groups.google.com/g/tmb-users⟩. Also, a very didactic and comprehensive documentation with several examples is available online ⟨https://kaskr.github.io/adcomp/_book/Tutorial.html⟩.

3 MULTIGLMM: A MULTINOMIAL GENERALIZED LINEAD MIXED MODEL

We are dealing with a complex survival data structure, the clustered competing risks setting. But we are using a general statistical modeling framework, the generalized linear mixed models (GLMMs), that was not made for this purpose.

To model competing risks data, one has to choose in which scale to work. We can work on the hazard scale dealing with the cause-specific hazard or on the probability scale dealing with the cause-specific cumulative incidence function (CIF). With the correct link function, we can make an appropriate GLMM to work on that probability scale.

Our focus in this thesis is to be able to deal with complex family studies, where there is generally a strong interest in describing age at disease onset in the scenarios of within-cluster dependence. The distribution of age at disease onset is directly described by the cause-specific CIF. To make a GLMM work for this type of data we need to accommodate the cause-specific CIFs and the censorings. Assuming the conditional distribution for our model response as multinomial already deals with both left-truncation and right-censoring, avoiding the specification of a censoring distribution. The cause-specific CIFs can be modeled via the link function of our, then, multinomial GLMM (multiGLMM). The multinomial distribution also guarantees that the CIFs of all causes are modeled.

Our choice of a general framework tries to make the inference of this complex model, easier. Besides, taking advantage of all the procedures mentioned in the previous chapter.

3.1 CLUSTER-SPECIFIC CUMULATIVE INCIDENCE FUNCTION (CIF)

Consider that the observed follow-up time of an individual is given by $T = \min(T^*,\, C)$, where T^* denote the failure time and C denote the censoring time. Given the possible covariates X (that can be time-dependent), for a cause-specific of failure k the CIF is defined as

$$\begin{aligned} F_k(t \mid X) &= \mathbb{P}[T \leq t, K = k \mid X] \\ &= \int_0^t f_k(z \mid X)\, \mathrm{d}z \\ &= \int_0^t \lambda_k(z \mid X)\, S(z \mid X)\, \mathrm{d}z, \quad t > 0, \quad k = 1, \ldots, K. \end{aligned}$$

where $f_k(t \mid X)$ is the (sub)density for the time to a type k failure. This is the general definition of a CIF, and to define it we need to define the functions that compose the subdensity.

The first is the cause-specific hazard function or process

$$\lambda_k(t \mid X) = \lim_{h \to 0} h^{-1} \mathbb{P}[t \leq T < t + h, K = k \mid T \geq t, X], \quad t > 0, \quad k = 1, \ldots, K.$$

In words, the cause-specific hazard function, $\lambda_k(t \mid X)$, represents the instantaneous rate for failures of type k at time t given X and all other failure types (competing causes). If we sum up all cause-specific hazard function we get the overall hazard function,

$$\lambda(t \mid X) = \sum_{k=1}^{K} \lambda_k(t \mid X).$$

From the overall hazard function we arrive in the overall survival function,

$$S(t \mid X) = \mathbb{P}[T > t \mid X] = \exp\left\{ - \int_0^t \lambda(z \mid X) \, \mathrm{d}z \right\},$$

the second function that compose the subdensity $f_k(t \mid X)$. A comprehensive reference for all these definitions is the book of Kalbfleisch & Prentice (2002).

Until this point, we were talking about a general CIF's definition. We need now a precise framework telling how to take into consideration our clustered/family structure. We use the same CIF specification of Cederkvist et al. (2019), i.e. the approach that motivated this thesis.

For two competing causes of failure, the cause-specific CIFs are specified in the following manner,

$$F_k(t \mid X, u_1, u_2, \eta_k) = \underbrace{\pi_k(X, u_1, u_2)}_{\substack{\text{cluster-specific} \\ \text{risk level}}} \times \underbrace{\Phi[w_k g(t) - X^\top \gamma_k - \eta_k]}_{\substack{\text{cluster-specific} \\ \text{failure time trajectory}}}, \quad t > 0, \quad k = 1, 2. \tag{3.1}$$

i.e. as a product of a cluster-specific risk level and a cluster-specific failure time trajectory, resulting in a cluster-specific CIF.

What makes the components in Equation 3.1 cluster-specific are $\boldsymbol{u} = \{u_1, u_2\}$ and $\boldsymbol{\eta} = \{\eta_1, \eta_2\}$, Gaussian distributed latent effects with zero mean and potentially correlated, i.e.

$$\begin{bmatrix} u_1 \\ u_2 \\ \eta_1 \\ \eta_2 \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_{u_1}^2 & \text{cov}(u_1, u_2) & \text{cov}(u_1, \eta_1) & \text{cov}(u_1, \eta_2) \\ & \sigma_{u_2}^2 & \text{cov}(u_2, \eta_1) & \text{cov}(u_2, \eta_2) \\ & & \sigma_{\eta_1}^2 & \text{cov}(\eta_1, \eta_2) \\ & & & \sigma_{\eta_2}^2 \end{bmatrix} \right).$$

The cluster-specific survival function is given as $S(t \mid X, \boldsymbol{u}, \boldsymbol{\eta}) = 1 - F_1(t \mid X, \boldsymbol{u}, \eta_1) - F_2(t \mid X, \boldsymbol{u}, \eta_2)$.

Since we use the same CIF specification of Cederkvist et al. (2019), the following descriptions and details are essentially the same encountered in the paper.

Focusing first on the second component of Equation 3.1. The cluster-specific failure time trajectory

$$\Phi[w_k g(t) - X^\top \gamma_k - \eta_k], \quad t > 0, \quad k = 1,\, 2,$$

where $\Phi(\cdot)$ is the cumulative distribution function of a standard Gaussian distribution.

Instead of $w_k g(t)$, in Cederkvist et al. (2019) is specified $\alpha_k(g(t))$, where $\alpha_k(\cdot)$ are monotonically increasing functions known up to a finite-dimensional parameter vector, w_k. Examples are monotonically increasing B-spline or piecewise lienar functions. However, to try to simplify the model structure we consider just the finite-dimensional parameter vector. The bottom line is that the authors do the same approach in their applications.

With regard to the function $g(t)$, it plays a crucial role since the separation of the CIF in Equation 3.1 is only possible with it. A time t transformation given by

$$g(t) = \text{arctanh}\left(\frac{t - \delta/2}{\delta/2}\right), \quad t \in]0,\, \delta[, \quad g(t) \in]-\infty,\, \infty[,$$

where δ depends on the data and cannot exceed the maximum observed follow-up time τ, i.e. $\delta \leq \tau$. With this transformation, based on a Fisher transformation, the value of the cluster-specific failure time trajectory is equal 1, at time δ. Consequently, $F_k(\delta \mid X, \boldsymbol{u}, \eta_k) = \pi_k(X \mid \boldsymbol{u})$ and, we can interpret $\pi_1(X \mid \boldsymbol{u})$ and $\pi_2(X \mid \boldsymbol{u})$ as the cause-specific cluster-specific risk levels, at time δ.

The cluster-specific risk levels are modeled by a multinomial logistic regression model with latent effects, i.e.

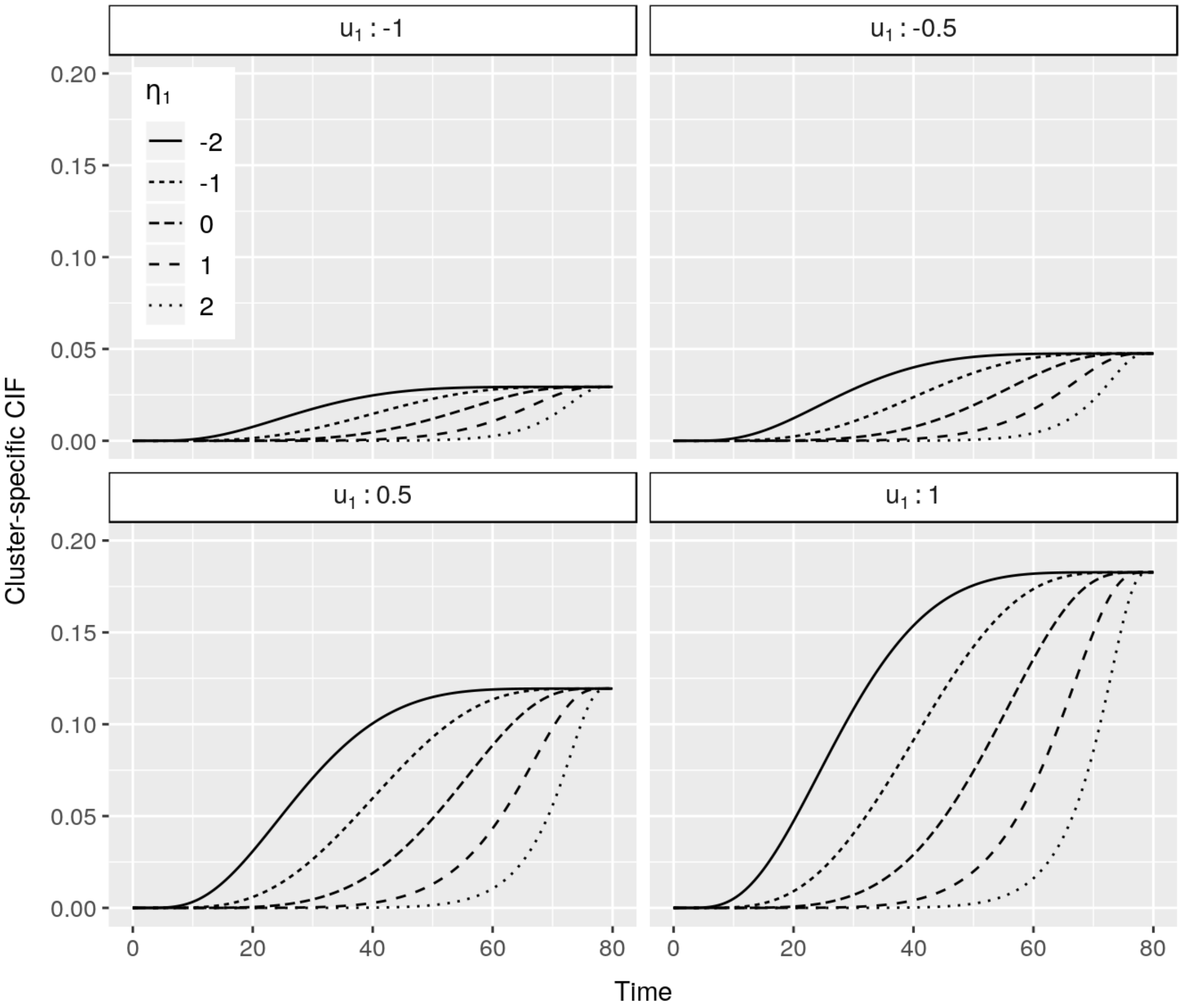
$$\pi_k(X, \boldsymbol{u}) = \frac{\exp\{X^\top \beta_k + u_k\}}{1 + \exp\{X^\top \beta_1 + u_1\} + \exp\{X^\top \beta_2 + u_2\}}, \quad k = 1,\, 2, \tag{3.2}$$

where β_k's are the parameters responsible for quantifying the impact of the covariates in the cause-specific risk levels. For individuals from the same chuster/family, at the same time point, the β_ks have the well-known odds ratio interpretation.

The γ_k's are the parameters responsible for quantifying the impact of the covariates in the cause-specific failure time trajectories, i.e. the shape of the cumulative incidence, and consequently how quickly the cluster-specific risk levels observed at time δ are reached. The fact that γ_k enters negatively in the cluster-specific failure time trajectory makes that a negative value causes an advance towards the cluster-specific risk level, whereas a covariate with a positive effect causes a delay.

Within-cluster dependence is induced by the latent effects in $\boldsymbol{u}$ and $\boldsymbol{\eta}$, but they don't have an easy interpretation. To help in the discussion, Figure 4 illustrates the cluster-specific CIF for a given failure cause, let's call it failure cause 1 (in total we have two).

FIGURE 4 – ILLUSTRATION OF A CLUSTER-SPECIFIC CUMULATIVE INCIDENCE FUNCTION (CIF), PROPOSED BY Cederkvist et al. (2019), FOR A GIVEN FAILURE CAUSE 1. FROM A CONFIGURATION WITH $X = 1$ FOR ALL SUBJECTS AND WITH $\beta_1 = -1.9$, $\beta_2 = -0.2$, $\gamma_1 = 1$, $w_1 = 3$ AND $u_2 = 0$. THE VARIATION BETWEEN FRAMES IS GIVEN BY THE LATENT EFFECTS u_1 AND η_1



SOURCE: The author (2020).

The latent effects u_1 and u_2 always appear together in the cluster-specific risk level, as consequency they have a joint effect on the cumulative incidence of both causes. Nevertheless, as we can see in Figure 4, an increase in u_k will increase the risk of failure from cause k and vice versa. The interpretation of $\text{cov}(\eta_1, \eta_2)$ and $\text{cov}(u_1, u_2)$ is more or less straightforward. With regard to $\text{cov}(u_k, \eta_k)$, a negative correlation between η_k and u_k imply that when η_k decreases, u_k increases and conversely when η_K increases, u_k decreases. In other words, an increased risk level is reached quickly and a decreased risk level is reached later, respectively.

Practical situations with a positive within-cause correlation are hard to find, i.e. where an increased risk level is associated with a late onset and vice versa. However, a positive cross-cause correlation between η and u sounds more realistic. i.e. where late onset of one failure cause is associated with a high absolute risk of another failure cause.

The latent effects are assumed independent across clusters and shared by individuals within the same cluster/family.

3.2 MODEL SPECIFICATION

Our generalized linear mixed model (GLMM) is specified in the following fashion. For two competing causes of failure, a subject i, with cluster j, in the time t, we have

$$y_{ijt} \mid \{u_{1j},\, u_{2j},\, \eta_{1j},\, \eta_{2j}\} \sim \text{Multinomial}(p_{1ijt},\, p_{2ijt},\, p_{3ijt})$$

$$\begin{bmatrix} u_1 \\ u_2 \\ \eta_1 \\ \eta_2 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma^2_{u_1} & \text{cov}(u_1,\, u_2) & \text{cov}(u_1,\, \eta_1) & \text{cov}(u_1,\, \eta_2) \\ & \sigma^2_{u_2} & \text{cov}(u_2,\, \eta_1) & \text{cov}(u_2,\, \eta_2) \\ & & \sigma^2_{\eta_1} & \text{cov}(\eta_1,\, \eta_2) \\ & & & \sigma^2_{\eta_2} \end{bmatrix} \right)$$

$$\begin{aligned} p_{kijt} &= \frac{\partial}{\partial t} F_k(t \mid X, u_1, u_2, \eta_k) \\ &= \frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki} + u_{kj}\}}{1 + \sum_{m=1}^{K-1} \exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi} + u_{mj}\}} \\ &\quad \times w_k \frac{\delta}{2\delta t - 2t^2}\, \phi\left(w_k \text{arctanh}\left(\frac{t - \delta/2}{\delta/2} \right) - \boldsymbol{x}_{kij}\gamma_{ki} - \eta_{kj} \right), \\ k &= 1,\, 2. \end{aligned} \tag{3.3}$$

The chosen link function to represent the probabilities is given by the derivative w.r.t. time t of the cluster-specific CIF. The choice of a multinomial logistic regression model ensures that the sum of the predicted cause-specific CIFs does not exceed 1.

Considering two competing causes of failure, we have a multinomial with three classes. The third class exists to handle the censorship and its probability is given by the complementary to reach 1. This framework in Equation 3.3 results in what we call multiGLMM, a multinomial GLMM.

For a random sample, the corresponding marginal likelihood functions in given by

$$\begin{aligned}
L(\boldsymbol{\theta}\,;\,y) &= \prod_{j=1}^{J} \int_{\Re^4} \pi(y_j \mid \boldsymbol{r}_j) \times \pi(\boldsymbol{r}_j)\, \mathrm{d}\boldsymbol{r}_j \\
&= \prod_{j=1}^{J} \int_{\Re^4} \Bigg\{ \underbrace{\prod_{i=1}^{n_j} \prod_{t=1}^{n_{ij}} \left(\frac{(\sum_{k=1}^{K} y_{kijt})!}{y_{1ijt}!\, y_{2ijt}!\, y_{3ijt}!} \prod_{k=1}^{K} p_{kijt}^{y_{kijt}} \right)}_{\text{fixed effect component}} \Bigg\} \times \\
&\qquad \underbrace{(2\pi)^{-2} |\Sigma|^{-1/2} \exp\left\{ -\frac{1}{2} \boldsymbol{r}_j^{\top} \Sigma^{-1} \boldsymbol{r}_j \right\} \mathrm{d}\boldsymbol{r}_j}_{\text{latent effect component}} \\
&= \prod_{j=1}^{J} \int_{\Re^4} \Bigg\{ \underbrace{\prod_{i=1}^{n_j} \prod_{t=1}^{n_{ij}} \prod_{k=1}^{K} p_{kijt}^{y_{kijt}}}_{\text{fixed effect}} \Bigg\} \underbrace{(2\pi)^{-2} |\Sigma|^{-1/2} \exp\left\{ -\frac{1}{2} \boldsymbol{r}_j^{\top} \Sigma^{-1} \boldsymbol{r}_j \right\}}_{\text{latent effect component}} \mathrm{d}\boldsymbol{r}_j, \qquad (3.4)
\end{aligned}$$

where $\boldsymbol{\theta} = [\boldsymbol{\beta}\ \boldsymbol{\gamma}\ \boldsymbol{w}\ \boldsymbol{\sigma}^2\ \boldsymbol{\varrho}]^{\top}$ is the parameters vector to be maximized. In our framework, a subject can fail from just one competing cause or get censor, at a given time. Thus, the fraction of factorials in the fixed effect component is made only by 0's and 1's. Finally, returning the value 1 .The matrix Σ is the variance-covariance matrix, which components are given by $\boldsymbol{\sigma}^2$ and $\boldsymbol{\varrho}$.

Now, Equation 3.4 in words. To each cluster (family) j we have a product of two components. The fixed effect component, given by a multinomial distribution with its probabilities specified through the cluster-specific CIF (Equation 3.1) and, the latent effect component, given by a multivariate Gaussian distribution.

To each subject i that composes a cluster j we have its specific fixed effects contribution. The likelihood in Equation 3.4 is the most general as possible, allowing for repeated measures to each subject. Since all subjects of a given cluster shares the same latent effect, we have just one latent effect contribution multiplying the product of fixed effects contribution. As we don't observe the latent effect variables, $\boldsymbol{r}_j$, we integrate out in it. With two competing causes of failure, we have four latent effects (a multivariate Gaussian distribution in four dimensions). As consequence, for each cluster, we approximate an integral in four dimensions. The product of these approximated integrals results in the called marginal likelihood, to be maximized in $\boldsymbol{\theta}$.

4 DATASETS

This chapter describes how to simulate from our multiGLMM, and describes a real-based dataset used as an application example. The simulation procedure addressed in Section 4.1 is not straightforward, since we need to simulate the outputs (failure or censor) and the failure/censoring times, controlling for a pre-specified censorship level. In Section 4.2 a simulated dataset based on the Nordic Cancer Union (NCU) twins data is presented as an application example.

4.1 SIMULATING FROM A multiGLMM

Being able to simulate some data from a model is a key task, fundamental to assess the finite-sample properties and the estimation procedure of a given statistical model.

The core of the simulation procedure for our multiGLMM, Equation 3.3, was proposed by Cederkvist et al. (2019) and is basically a four-step procedure.

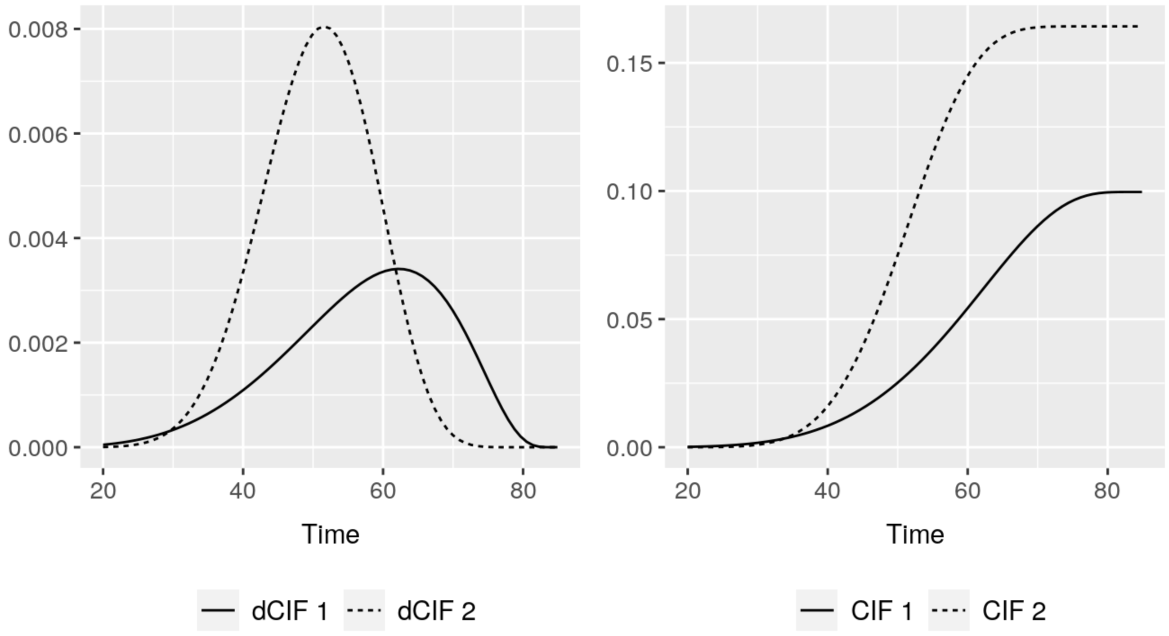
In the first step, based on the set model parameters and sampled latent effects, we compute the cause-specific failure times. The second step (the only when that differs from the simulation procedure proposed by Cederkvist et al. (2019)) consists of the computation of all competing risk probabilities as defined in Equation 3.3, with the censorship probability being computed as the complementary to guarantee that the probabilities sum up 1 to each individual. Based on that probabilities, we sample the outputs, i.e. the subject fails or censors, from a multinomial probability distribution. The last and fourth step is a somehow output overwriting to have better control of the proportion of censorship. In this last step which individuals should be censor are chosen by a Bernoulli process, with a fixed probability, and its censoring times are sampled from a given Uniform probability distribution. Algorithm 1 describes step-by-step all that process for an arbitrary number of competing causes of failure.

ALGORITHM 1 SIMULATING FROM THE multiGLMM

```
Set $J$, the number of clusters/families
Set $n_j$, the number of individuals in each family    ▷ can be of different sizes
Set values to the model parameters $\boldsymbol{\theta} = [\boldsymbol{\beta}\ \boldsymbol{\gamma}\ \boldsymbol{w}\ \boldsymbol{\sigma}^2\ \boldsymbol{\varrho}]^\top$
Sample $J$ vectors of latent effects from $\mathcal{N}_{(K-1)\times(K-1)}(\mathbf{0},\ \Sigma(\boldsymbol{\sigma}^2, \boldsymbol{\varrho}))$
Set $\delta$    ▷ cannot exceed the maximum follow-up time
Sample $\varsigma \sim \mathrm{U}(0,\ 1)$
Compute the cause-specific failure times by solving
    $\varsigma = \Phi[w_k g(t_k) - \mathrm{X}^\top \gamma_k - \eta_k] \quad \text{for } t_k, \quad k = 1,\ 2,\ \ldots,\ K-1$
Compute the competing risk probabilities
    $p_{kijt} = \frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki} + u_{kj}\}}{1 + \sum_{m=1}^{K-1} \exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi} + u_{mj}\}}$
    $\quad \times w_k \frac{\delta}{2\delta t - 2t^2}\, \phi\left(w_k \operatorname{arctanh}\left(\frac{t - \delta/2}{\delta/2}\right) - \boldsymbol{x}_{kij}\gamma_{ki} - \eta_{kj}\right),$
    $p_{Kijt} = 1 - \sum_{k=1}^{K-1} p_{kijt}, \quad k = 1,\ 2,\ \ldots,\ K-1$
Sample $J \times n_j$ vectors from a Multinomial$(p_{1ijt},\ p_{2ijt},\ \ldots,\ p_{Kijt})$
If $t_{kij} = \delta$, subject moves to class K    ▷ any failure at time $\delta$ is censored
Set a censoring probability, $cp$    ▷ larger than actual proportion of censored
Sample which individuals will be censored from a Bernoulli$(cp)$
For the censored individuals, sample the censoring times from a U$(0,\ \delta + 30)$
return To each individual, its failure/censoring time and from which cause-specific
    it is
```

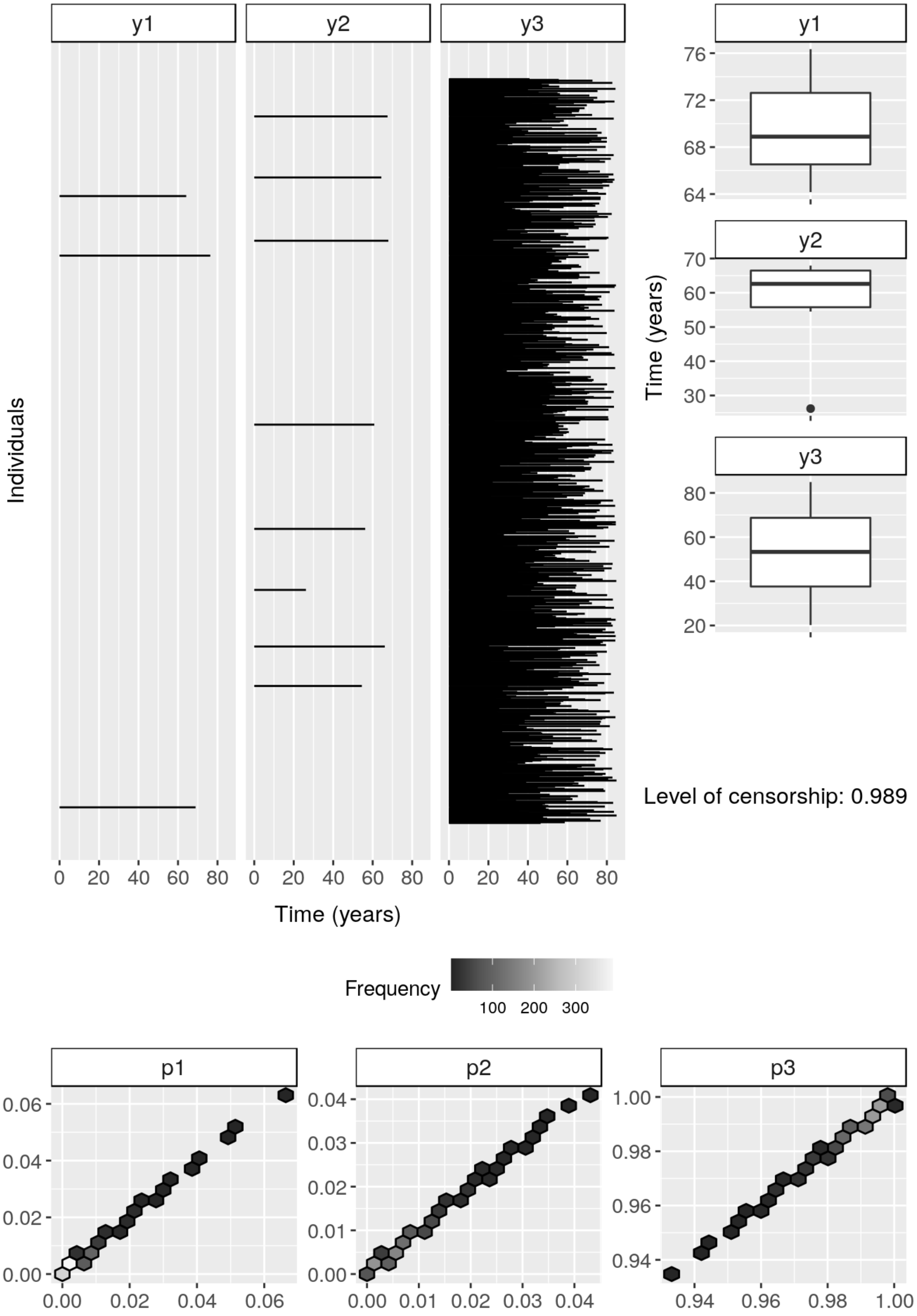
SOURCE: The author (2020).

FIGURE 5 – FAILURE (y_1, y_2) AND CENSORING (y_3) TIMES FOR 200 PAIRS OF TWINS SAMPLED FROM A MODEL ...



SOURCE: The author (2020).

FIGURE 6 – FAILURE (y_1, y_2) AND CENSORING (y_3) TIMES FOR 200 PAIRS OF TWINS SAMPLED FROM A MODEL . . .



SOURCE: The author (2020).

FIGURE 7 – CÓDIGOS EM LINGUAGEM R PARA GERAÇÃO DE VARIÁVEIS ALEATÓRIAS BETA CORRELACIONADAS

```
R = 1000  # tamanho da amostra
mu = 0.5  # parâmetro de média
phi = 9  # parâmetro de dispersão
cor_matrix <- matrix(c(1.0,0.75,0.75,1.0),2,2) # matriz de correlação
require(MASS)  # carrega o pacote com a função mvrnorm()
Z <- mvrnorm(n = R, mu = c(0,0), Sigma = cor_matrix)  # passo 1
Y <- qbeta(pnorm(Z), shape1 = mu*phi, shape2 = (1 - mu)*phi) # passo 2
```

SOURCE: The author (2020).

4.2 REAL-BASED DATASET

5 RESULTS

6 FINAL CONSIDERATIONS

6.1 FUTURE WORKS

BIBLIOGRAPHY

ANDERSEN, P. K.; GESKUS, R. B.; WITTE, T. de; PUTTER, H. Competing risks in epidemiology: possibilities and pitfalls. *International Journal of Epidemiology*, v. 31, n. 1, p. 861–870, 2012. Cited on page 15.

BONAT, W. H.; RIBEIRO, P. J. Practical likelihood analysis for spatial generalized linear mixed models. *Environmetrics*, v. 27, n. 1, p. 83–89, 2016. Cited 2 times on pages 21 and 23.

CEDERKVIST, L.; HOLST, K. K.; ANDERSEN, K. K.; SCHEIKE, T. H. Modeling the cumulative incidence function of multivariate competing risks data allowing for within-cluster dependence of risk and timing. *Biostatistics*, v. 20, n. 2, p. 199–217, 2019. Cited 8 times on pages 8, 13, 15, 17, 32, 33, 34, and 37.

CLAYTON, D. G. A model for association in bivariate life tables and its application in epidemiological studies of familial rendency in chronic disease incidence. *Biometrika*, v. 65, n. 1, p. 141–151, 1978. Cited on page 15.

DEMPSTER, A. P.; LAIRD, N. M.; RUBIN, D. B. Maximum likelihood estimation from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society, Series B (Methodological)*, v. 39, n. 1, p. 1–38, 1977. Cited on page 21.

DENNIS, J. E.; GAY, D. M.; WELSCH, R. E. An Adaptive Nonlinear Least-Squares Algorithm. *ACM Transactions on Mathematical Software*, v. 7, n. 3, p. 348–368, 1981. Cited on page 25.

DIACONIS, P. The Markov chain Monte Carlo revolution. *Bulletin (New Series) of the American Mathematical Society*, v. 46, n. 2, p. 179–205, 2009. Cited on page 20.

FOURNIER, D. A.; SKAUG, H. J.; ANCHETA, J.; IANELLI, J.; MAGNUSSON, A.; MAUNDER, M. N.; NIELSEN, A.; SIBERT, J. AD Model Builder: using automatic differentiation for statistical inference of highly parameterized complex nonlinear models. *Optimization Methods and Software*, v. 27, n. 2, p. 233–249, 2012. Cited on page 29.

GAY, D. M. *Usage summary for selected optimization routines*. Computing Science Technical Report 153, AT&T Bell Laboratories. Murray Hill, NJ, 1990. Cited on page 25.

GELFAND, A. E.; SMITH, A. F. M. Sampling-Based Approaches to Calculating Marginal Densities. *Journal of the American Statistical Association*, v. 85, n. 410, p. 398–409, 1990. Cited on page 20.

GUENNEBAUD, G.; JACOB, B. et al. *Eigen v3*. 2010. ⟨http://eigen.tuxfamily.org⟩. Cited on page 30.

KALBFLEISCH, J. D.; PRENTICE, R. L. *The Statistical Analysis of Failure Time Data*. Second Edition. Hoboken, New Jersey: John Wiley & Sons, Inc., 2002. Cited 3 times on pages 13, 15, and 32.

KRISTENSEN, K.; NIELSEN, A.; BERG, C. W.; SKAUG, H. J.; BELL, B. M. TMB: Automatic Differentiation and Laplace Approximation. *Journal of Statistical Software*, v. 70, n. 5, p. 1–21, 2016. Cited 3 times on pages 16, 19, and 29.

MCCULLAGH, P.; NELDER, J. A. *Generalized linear models*. Second edition. London: Chapman & Hall, 1989. Cited on page 16.

MOLENBERGHS, G.; VERBEKE, G. *Models for Discrete Longitudinal Data*. New York: Springer, 2005. Cited on page 21.

NELDER, J. A.; WEDDERBURN, R. W. M. Generalized linear models. *Journal of the Royal Statistical Society, Series A*, v. 135, n. 3, p. 370–384, 1972. Cited on page 16.

NOCEDAL, J.; WRIGHT, S. J. *Numerical Optimization*. Second Edition. New York: Springer, 2006. (Springer Series in Operations Research and Financial Engineering). Cited 3 times on pages 24, 25, and 26.

PEYRé, G. *Course notes on Optimization for Machine Learning*. 2020. May 10, ⟨https://mathematical-tours.github.io/book-sources/optim-ml/OptimML.pdf⟩. CNRS & DMA, École Normale Supérieure. Cited 2 times on pages 26 and 27.

PINHEIRO, J. C.; CHAO, E. C. Efficient Laplacian and Adaptive Gaussian Quadrature Algorithms for Multilevel Generalized Linear Mixed Models. *Journal of Computational and Graphical Statistics*, v. 15, n. 1, p. 58–81, 2006. Cited on page 21.

R DEVELOPMENT CORE TEAM. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2018. ⟨http://www.R-project.org/⟩. Cited 3 times on pages 19, 25, and 29.

SHUN, Z.; MCCULLAGH, P. Laplace approximation of high dimentional integrals. *Journal of the Royal Statistical Society, Series B (Methodological)*, v. 57, n. 4, p. 749–760, 1995. Cited on page 21.

TIERNEY, L.; KADANE, J. Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, v. 81, n. 393, p. 82–86, 1986. Cited on page 21.

VALPEL, J. W.; MANTON, K. G.; STALLARD, E. The impact of heterogeneity in Individual Frailty on the Dynamics of Mortality. *Demography*, v. 16, n. 1, p. 439–454, 1979. Cited on page 15.

WOOD, S. N. *Core Statistics*. IMS: Institute of Mathematical Statistics, Textbooks, 2015. Cited 3 times on pages 21, 22, and 26.

Appendix

APPENDIX A – GRADIENTS AND HESSIAN MATRIX OF THE MULTINOMIAL GENERALIZED LINEAR MIXED MODEL (GLMM) WITH A LINK FUNCTION BASED ON THE CUMULATIVE INCIDENCE FUNCTION (CIF)

$$\boldsymbol{l}'(u_{kj}) = y_{kij}\frac{1+\sum_{m\neq k}^{K-1}\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u_{mj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}$$

$$-\left(\sum_{m\neq k}^{K-1} y_{mij}\right)\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}$$

$$-\, y_{Kij}\frac{1}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}\Bigg($$

$$\frac{w_k\frac{\delta}{2\delta t-2t^2}\phi[w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\gamma_{ki}-\eta_{kj}]\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\Big(1+\sum_{m\neq k}^{K-1}\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}(1-w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{nij}\gamma_{ni}-\eta_{nj}])}$$

$$-\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\sum_{m\neq k}^{K-1}w_m\frac{\delta}{2\delta t-2t^2}\phi[w_m\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{mij}\gamma_{mi}-\eta_{mj}]\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}(1-w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{nij}\gamma_{ni}-\eta_{nj}])}$$

$$-\, e_k^\top\boldsymbol{Qr},$$

$$\boldsymbol{l}'(\eta_{kj}) = y_{kij}(w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\gamma_{ki}-\eta_{kj})$$

$$-\, y_{Kij}\frac{\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_k\frac{\delta}{2\delta t-2t^2}(w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\gamma_{ki}-\eta_{kj})\phi[w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)}{1-\sum_{n=1}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{nij}\gamma_{ni}-}$$

$$-\, e_k^\top\boldsymbol{Qr}.$$

and the hessian,

$$l^{''}(u_{kj}) = -\frac{\Big(\sum_{k=1}^{K-1} y_{kij}\Big)\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\Big(1+\sum_{m\neq k}^{K-1}\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u_{mj}\}\Big)}{\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\Big)^2}$$

$$+\frac{y_{Kij}\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\sum_{m\neq k}^{K-1} w_m\frac{\delta}{2\delta t-2t^2}\phi[w_m\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{mij}\gamma_{mi}-\eta_{mj}]\exp}{\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\Big)\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}(1-w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}}$$

$$-\frac{y_{Kij}w_k\frac{\delta}{2\delta t-2t^2}\phi[w_k\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{kij}\gamma_{ki}-\eta_{kj}]\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\Big(1+\sum_{m\neq k}^{K-1}\exp}{\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\Big)\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}(1-w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}}$$

$$-\frac{y_{Kij}}{\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\Big)^2}\Bigg($$

$$\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\sum_{m\neq k}^{K-1} w_m\frac{\delta}{2\delta t-2t^2}\phi[w_m\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{mij}\gamma_{mi}-\eta_{mj}]\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u_{mj}}{\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}(1-w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{nij}\gamma_{ni}-\eta_{nj}])\Big)^2}$$

$$-\frac{w_k\frac{\delta}{2\delta t-2t^2}\phi[w_k\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{kij}\gamma_{ki}-\eta_{kj}]\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\Big(1+\sum_{m\neq k}^{K-1}\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}-}{\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}(1-w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{nij}\gamma_{ni}-\eta_{nj}])\Big)}$$

$$\times\Bigg(\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}(1-w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)$$

$$+\Big(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\Big)\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}(1-w_k\frac{\delta}{2\delta t-2t^2}\phi[w_k\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)$$

$$-e_k^\top\boldsymbol{Q},$$

$$l^{''}(\eta_{kj}) = -y_{kij}$$

$$-y_{Kij}\frac{\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_k\frac{\delta}{2\delta t-2t^2}\phi[w_k\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{kij}\gamma_{ki}-\eta_{kj}]\Big(\Big(w_k\text{arctanh}\Big(\frac{t-\delta/}{\delta/2}}{1-\sum_{n=1}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{nij}\gamma}$$

$$-y_{Kij}\frac{\Big(\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_k\frac{\delta}{2\delta t-2t^2}(w_k\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{kij}\gamma_{ki}-\eta_{kj})\phi[w_k\text{arctanh}\Big(\frac{t-\delta/}{\delta/2}}{\Big(1-\sum_{n=1}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\Big(\frac{t-\delta/2}{\delta/2}\Big)-\boldsymbol{x}_{nij}\gamma_{ni}-}$$

$$-e_k^\top\boldsymbol{Q},$$

$$\boldsymbol{l}''(u_{kj}u_{mj}) = \left(\sum_{k=1}^{K-1} y_{kij}\right) \frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki} + u_{kj}\}\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi} + u_{mj}\}}{\left(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}\right)^2}$$

$$+ \frac{y_{Kij}\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki} + u_{kj}\} w_m \frac{\delta}{2\delta t - 2t^2}\phi[w_m \mathrm{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right) - \boldsymbol{x}_{mij}\gamma_{mi} - \eta_{mj}]\,\mathrm{ex}}{\left(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}\right)\bigg(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}(1 - w_n \frac{\delta}{2\delta t - 2t^2}\phi[w_n \mathrm{arcta}}$$

$$- \frac{y_{Kij} w_k \frac{\delta}{2\delta t - 2t^2}\phi[w_k \mathrm{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right) - \boldsymbol{x}_{kij}\gamma_{ki} - \eta_{kj}]\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki} + u_{kj}\}\,\mathrm{exp}}{\left(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}\right)\bigg(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}(1 - w_n \frac{\delta}{2\delta t - 2t^2}\phi[w_n \mathrm{arcta}}$$

$$- \frac{y_{Kij}}{\left(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}\right)^2}\Bigg($$

$$\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki} + u_{kj}\}\sum_{m\neq k}^{K-1} w_m \frac{\delta}{2\delta t - 2t^2}\phi[w_m \mathrm{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right) - \boldsymbol{x}_{mij}\gamma_{mi} - \eta_{mj}]\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi} + }{\left(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}(1 - w_n \frac{\delta}{2\delta t - 2t^2}\phi[w_n \mathrm{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right) - \boldsymbol{x}_{nij}\gamma_{ni} - \eta_{nj}])\right)^2}$$

$$- \frac{w_k \frac{\delta}{2\delta t - 2t^2}\phi[w_k \mathrm{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right) - \boldsymbol{x}_{kij}\gamma_{ki} - \eta_{kj}]\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki} + u_{kj}\}\bigg(1 + \sum_{m\neq k}^{K-1}\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}}{\bigg(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}(1 - w_n \frac{\delta}{2\delta t - 2t^2}\phi[w_n \mathrm{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right) - \boldsymbol{x}_{nij}\gamma_{ni} - \eta_{n}}$$

$$\times \Bigg(\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi} + u_{mj}\}\bigg(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}(1 - w_n \frac{\delta}{2\delta t - 2t^2}\phi[w_n \mathrm{arctanh}\Bigg(\frac{t -}{\delta/}$$

$$+ \left(1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}\right)\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi} + u_{mj}\}(1 - w_m \frac{\delta}{2\delta t - 2t^2}\phi[w_m \mathrm{arctanh}\Bigg(\frac{t -}{\delta/}$$

$$- e_k^\top \boldsymbol{Q},$$

$$\boldsymbol{l}''(\eta_{kj}\eta_{mj}) = -y_{Kij}\frac{\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}} w_k \frac{\delta}{2\delta t - 2t^2}(w_k \mathrm{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right) - \boldsymbol{x}_{kij}\gamma_{ki} - \eta_{kj})\phi[w_k \mathrm{arctanh}\bigg(\frac{t}{}}{\bigg(1 - \sum_{n=1}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}} w_n \frac{\delta}{2\delta t - 2t^2}\phi[w_n \mathrm{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right) - \boldsymbol{x}_{nij}\gamma}$$

$$\times \frac{\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi} + u_{mj}\}}{1 + \sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni} + u_{nj}\}} w_m \frac{\delta}{2\delta t - 2t^2}(w_m \mathrm{arctanh}\left(\frac{t - \delta/2}{\delta/2}\right) - \boldsymbol{x}_{mij}\gamma_{mi} - \eta_{mj})$$

$$\times \phi[w_m \mathrm{arctanh}\left(\frac{t - \delta/2}{\delta/2}\right) - \boldsymbol{x}_{mij}\gamma_{mi} - \eta_{mj}]$$

$$- e_k^\top \boldsymbol{Q},$$

$$l''(\eta_{kj}u_{kj}) = y_{Kij}\frac{\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_k\frac{\delta}{2\delta t-2t^2}(w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\boldsymbol{\gamma}_{ki}-\eta_{kj})\phi[w_k\text{arctanh}\left(\frac{t-\delta}{\delta/}\right.}{\left(1-\sum_{n=1}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{nij}\boldsymbol{\gamma}_{ni}\right.}$$

$$\times\left(\sum_{n\neq k}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}}{\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)^2}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{nij}\right.$$

$$-\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\left(\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)-\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\right)}{\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)^2}$$

$$\left.\times w_k\frac{\delta}{2\delta t-2t^2}\phi[w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\boldsymbol{\gamma}_{ki}-\eta_{kj}]\right)$$

$$-y_{Kij}\frac{\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\left(\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)-\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\right)}{\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)^2}}{1-\sum_{n=1}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{nij}\boldsymbol{\gamma}_{ni}-\eta_{nj}]}$$

$$\times w_k\frac{\delta}{2\delta t-2t^2}(w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\boldsymbol{\gamma}_{ki}-\eta_{kj})\phi[w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\boldsymbol{\gamma}_{ki}-$$

$$-e_k^\top\boldsymbol{Q},$$

$$l''(\eta_{kj}u_{mj}) = y_{Kij}\frac{\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u_{mj}\}}{\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)^2}w_k\frac{\delta}{2\delta t-2t^2}(w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\boldsymbol{\gamma}_{ki}-\eta_{kj})\phi[w_k\text{arcta}}{1-\sum_{n=1}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}}$$

$$+y_{Kij}\frac{\frac{\exp\{\boldsymbol{x}_{kij}\boldsymbol{\beta}_{ki}+u_{kj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_k\frac{\delta}{2\delta t-2t^2}(w_k\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{kij}\boldsymbol{\gamma}_{ki}-\eta_{kj})\phi[w_k\text{arctanh}\left(\frac{t-\delta}{\delta/}\right.}{\left(1-\sum_{n=1}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}{1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{nij}\boldsymbol{\gamma}_{n}\right.}$$

$$\times\left(\sum_{n\neq m}^{K-1}\frac{\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u_{mj}\}}{\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)^2}w_n\frac{\delta}{2\delta t-2t^2}\phi[w_n\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}\right.$$

$$-\frac{\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u_{mj}\}\left(\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)-\exp\{\boldsymbol{x}_{mij}\boldsymbol{\beta}_{mi}+u_{mj}\}\right)}{\left(1+\sum_{n=1}^{K-1}\exp\{\boldsymbol{x}_{nij}\boldsymbol{\beta}_{ni}+u_{nj}\}\right)^2}$$

$$\left.\times w_m\frac{\delta}{2\delta t-2t^2}\phi[w_m\text{arctanh}\left(\frac{t-\delta/2}{\delta/2}\right)-\boldsymbol{x}_{mij}\boldsymbol{\gamma}_{mi}-\eta_{mj}]\right)$$

$$-e_k^\top\boldsymbol{Q}.$$