

2015

# MULTI-STATE MODELS FOR INTERVAL CENSORED DATA WITH COMPETING RISK

Shaoceng Wei
*University of Kentucky*, swe225@g.uky.edu

STUDENT AGREEMENT:

I represent that my thesis or dissertation and abstract are my original work. Proper attribution has been given to all outside sources. I understand that I am solely responsible for obtaining any needed copyright permissions. I have obtained needed written permission statement(s) from the owner(s) of each third-party copyrighted matter to be included in my work, allowing electronic distribution (if such use is not permitted by the fair use doctrine) which will be submitted to UKnowledge as Additional File.

I hereby grant to The University of Kentucky and its agents the irrevocable, non-exclusive, and royalty-free license to archive and make accessible my work in whole or in part in all forms of media, now or hereafter known. I agree that the document mentioned above may be made available immediately for worldwide access unless an embargo applies.

I retain all other ownership rights to the copyright of my work. I also retain the right to use in future works (such as articles or books) all or part of my work. I understand that I am free to register the copyright to my work.

REVIEW, APPROVAL AND ACCEPTANCE

The document mentioned above has been reviewed and accepted by the student's advisor, on behalf of the advisory committee, and by the Director of Graduate Studies (DGS), on behalf of the program; we verify that this is the final, approved version of the student's thesis including all changes required by the advisory committee. The undersigned agree to abide by the statements above.

Shaoceng Wei, Student

Dr. Richard J. Kryscio, Major Professor

Dr. Constance Wood, Director of Graduate Studies

# MULTI-STATE MODELS FOR INTERVAL CENSORED DATA WITH COMPETING RISK

ABSTRACT OF DISSERTATION

A dissertation submitted in partial
fulfillment of the requirements for
the degree of Doctor of Philosophy
in the College of Arts and Sciences
at the University of Kentucky

By
Shaoceng Wei
Lexington, Kentucky

Director: Dr. Richard J. Kryscio, Professor of Statistics
Lexington, Kentucky 2015

ABSTRACT OF DISSERTATION

MULTI-STATE MODELS FOR INTERVAL CENSORED DATA WITH COMPETING RISK

Multi-state models are often used to evaluate the effect of death as a competing event to the development of dementia in a longitudinal study of the cognitive status of elderly subjects. In this dissertation, both multi-state Markov model and semi-Markov model are used to characterize the flow of subjects from intact cognition to dementia with mild cognitive impairment and global impairment as intervening transient, cognitive states and death as a competing risk.

Firstly, a multi-state Markov model with three transient states: intact cognition, mild cognitive impairment (M.C.I.) and global impairment (G.I.) and one absorbing state: dementia is used to model the cognitive panel data. A Weibull model and a Cox proportional hazards (Cox PH) model are used to fit the time to death based on age at entry and the APOE4 status. A shared random effect correlates this survival time with the transition model.

Secondly, we further apply a Semi-Markov process in which we assume that the waiting times are Weibull distributed except for transitions from the baseline state, which are exponentially distributed and we assume no additional changes in cognition occur between two assessments. We implement a quasi-Monte Carlo (QMC) method to calculate the higher order integration needed for the likelihood based estimation.

At the end of this dissertation we extend a non-parametric "local EM algorithm" to obtain a smooth estimator of the cause-specific hazard function (CSH) in the presence of competing risk.

All the proposed methods are justified by simulation studies and applications to the Nun Study data, a longitudinal study of late life cognition in a cohort of 461 subjects.

KEYWORDS: multi-state Markov chain; competing event; Nun Study; semi-Markov model; Cause-specific hazard; Local EM algorithm

Author's signature: Shaoceng Wei

Date: May 6, 2015

# ACKNOWLEDGMENTS

I would like to sincerely thank my advisor Dr. Richard J. Kryscio for his guidance, support and patience and most importantly, his advice not only on my dissertation but also on my preparing for future career during my graduate studies at the University of Kentucky. He encouraged me to not only improve my skills in statistics but also the skills of communication and writing, which will benefit me for a long run during my life.

Also, I would like to thank my dissertation committee consisting of Dr. David Fardo, Dr. William Griffith, Dr. Constance Wood and Dr. Ruriko Yoshida. I am grateful to them for their time, and the careful and critical reading of this dissertation.

Most importantly, I would like to thank my husband Shihong. His support, encouragement, patience and steadfast love were undeniably the bedrock upon which the past six years of my life has been built. I thank my parents for their support and faith on me and for allowing me to pursue what I wanted.

In addition to the people I mentioned above, I wish to thank the faculty, students, and staffs from the Department of Statistics for all the knowledge, help, and support I received from them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

# Chapter 1 Introduction

A Medical study that allows a long follow-up period to assess a certain outcome typically generates rich yet complex data. In addition to the baseline diagnostic covariates obtained at the onset of the study, other important information may also be collected longitudinally as the study continues. A question of primary interest is to identify the risk factors associated with the outcome and quantify the effects of the identified risk factors. Any attempt to answer this question should recognize the complex data structure, such as a possible multi-state outcome variable or censored event times due to potential loss of follow-up, and incorporate these structures into the statistical models. We aim to develop flexible and powerful statistical models to address this question. The Nun dataset introduced in 1.1 is used as a working example for ease of presentation, but obviously the methodologies presented in this dissertation can be applied far beyond the Nun study itself.

## 1.1 Background of Nun Study Data

The Nun study is a well-known longitudinal study, initiated in 1986. Scientific interest in this study centers on examining the onset of dementia, a chronic disease, in relation to measureable risk factors. The study cohort consists of 672 members of the School Sisters of Notre Dame born before 1917 and living in retirement communities in the mid-western, eastern, and southern United States. Similar environmental influences and general lifestyles make the nuns an ideal population to study [52]. The participants were recruited in phases and received annual cognitive assessments with brain donation at death. The time independent covariates, such as gender, education level, gene-related factor (Apolipoprotein E4), were recorded at the baseline (first visit). The time dependent covariates, such as age, were recorded at each of the fol-

low up assessments. At each assessment, the cognitive status is categorized into one of the following three states:

**Intact Cognition:** The patient passes all the cognitive and activities of Daily Living tests.

**Mild Cognitive Impairment:** The patient passes the Delayed Word Recall, Mini-Mental State Exam, and Activities of Daily Living tests but fails one or more of the other three cognitive tests, including Boston Naming (participant is told to tell the examiner the name of 60 pictures and given about 20 seconds to response for each picture), Verbal Fluency (participant has to say as many words as possible from a category in a given time (usually 60 seconds)), and Constructional Praxis (check the participant's ability to build, assemble, or draw objects).

**Global Impairment:** The patient passes the Delayed Word Recall but fails the Mini-Mental State Exam, Activities of Daily Living test, and one or more of the other three cognitive tests (Boston Naming, Verbal Fluency, and Constructional Praxis) without meeting criteria for dementia.

The subjects were followed during the study period until terminal status, death or dementia, occurred. Some subjects were still at risk at the end of this study, which results in right-censored event times. At the meantime, the time to each cognitive state and dementia is subject to interval censoring, due to the periodical assessment. The time to death is exactly known. The flow diagram of the transition is summarized in Figure 1.1.

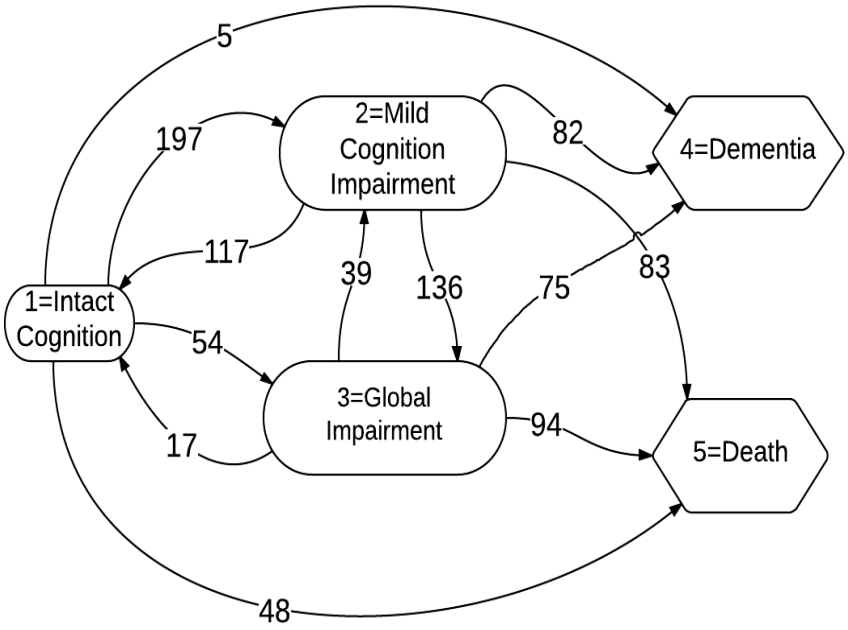


Figure 1.1: Frequency of the one-step transitions

## 1.2 Multi-State Models

For longitudinal study with categorical response variables, the multi-state Markov model is the most commonly used technique due to its simplicity. The central assumption, the Markov property, assumes that the subject's next status only depends probabilistically on the subject's current status given all of the subject's historical status. Under this assumption, the likelihood function can be easily formulated. See [52, 19, 45].

Of particular interest is the work by Salazar et al. [45] who proposed an elegant multi-state Markov models to model the progression to dementia in which death was treated as a competing absorbing state.

Specifically, the one-step transition probability $\mathrm{P}_{sv}$ from state $s$ to $v$ is expressed as

$$\log\left\{\frac{\mathrm{P}_{sv}(\theta_{sv}|\boldsymbol{X},\boldsymbol{\gamma})}{\mathrm{P}_{s1}(\theta_{s1}|\boldsymbol{X},\boldsymbol{\gamma})}\right\} = \alpha_v + \boldsymbol{X}^\top\boldsymbol{\beta}_v + \xi_v^s + \boldsymbol{W}^\top\boldsymbol{\gamma}, \qquad v = 2,3,4,5; s = 1,2,3.$$

where $\alpha_v$ is the intercept, $\boldsymbol{\beta}_v$ is the fixed effect for covariate vector $\boldsymbol{X}$, $\xi_v^s$ is the unknown fixed effect for the prior state $s$ and current state $v$ and $\boldsymbol{\gamma}$ is the random effect associated with each subject. The corresponding conditional likelihood function based on the baseline state is expressed as

$$L(\Theta|\boldsymbol{X}) = \int_\Omega \prod_{l=2}^{n} \times \prod_{\substack{1\le s\le 3\\ 2\le v\le 5}} \left\{\mathrm{P}_{sv}(\Theta|\boldsymbol{X},\boldsymbol{\gamma})\right\}^{\delta_{y_{l-1},s}\delta_{y_l,v}} \Pi_w(\Theta|\boldsymbol{Z},\boldsymbol{\gamma})h(\boldsymbol{\gamma})\mathrm{d}\boldsymbol{\gamma}, \tag{1.1}$$

where $\delta_{y_l,v}$ is an indicator function that takes value 1 if $y_l = v$ and 0 otherwise.

The adverse effect of ignoring the baseline distribution in the parameter estimation was investigated by Yu et al. [57]. In Yu et al. [56], they showed that the transition probability was time-dependent and further considered a non-homogeneous discrete time Markov chain consists of either absorbing states or transient states at any time.

However, in the longitudinal study, especially the chronic disease study, the progression of event is not only characterized by response status but also the corresponding waiting time (also known as holding time). Simply modeling the progression or the time-to-event data will result in loss of important information since these two processes may not be independent. Moreover, in the presence of censoring, solely modeling the transition status, without incorporating the waiting times, will introduce biases, since transitions involving worse cognitive status tend to be censored due to death or loss of follow-up [13]. Therefore, we aim to adopt firstly a joint modeling approach where time-to-death are explicitly accounted for and then a general semi-Markov model to incorporate the waiting times.

### 1.2.1 Joint Modeling of Longitudinal and Competing Risk Time-to-Event Data

Considerable recent literatures are focused on the so-called joint modeling, which models both the event time and a simultaneous longitudinal process by adding shared random effects and covariates to link those two components [55, 21]. The general approach for the likelihood function of the joint modeling for a longitudinal data $y$ along with the survival time $d$ is defined as

$$f(y,d) = \int_{\gamma} f(y|\boldsymbol{\gamma}) f(d|\boldsymbol{\gamma}) h(\boldsymbol{\gamma}) \mathrm{d}\boldsymbol{\gamma}, \tag{1.2}$$

where $\boldsymbol{\gamma}$ is the subject specific shared random effect, $f(y|\boldsymbol{\gamma})$ is the density function of the longitudinal data y and $f(d|\boldsymbol{\gamma})$ is the density function related to the survival time $d$. To find the MLE of 1.2, we should integrate out the shared random effect $\boldsymbol{\gamma}$.

Those joint models lead to correction of biases and enhanced efficiency. However, only the case of continuous longitudinal response variables was discussed.

Under the principle of joint modeling, we use a sub-model of four-state Markov chain to account the progression to dementia and another sub-model of parametric and semi-parametric Proportional Hazards models to account for the time-to-death, with these two sub-models linked by shared random effects. The commonly used parametric or semi-parametric distributions in modeling survival time are Weibull distribution, Generalized Weibull distribution and the Cox Proportional (Cox PH) hazard model.

**Weibull distribution** $W(r,\mu)$**:** The hazard function is defined as $\lambda(t) = r\mu t^{r-1}$, where $r > 0$ is the shape parameter and $\mu > 0$ is the scale parameter. In application, $\log(\mu)$ is always assumes to be a function of covariates.

**Generalized Weibull distribution** $WG(r, \mu, \theta)$**:** First introduced by Mudholkar et al. [41] and with the hazard function $\lambda(t) = r\mu\theta^{-1}(1+\mu t^r)^{1/\theta-1}t^{r-1}$. If $\theta = 1$, a Weibull formulation is obtained. This is a more flexible distribution due to it can have U shape or inverse U shape hazard function, see Figure 2.2.

**Cox PH model:** The hazard function has the form $\lambda(t|\boldsymbol{X}) = \lambda_0(t)\exp(\boldsymbol{X}^\top\boldsymbol{\beta})$, where $\lambda_0(t)$ is the unspecified baseline hazard and $\boldsymbol{\beta}$ is effect vector associated with the covariate vector $\boldsymbol{X}$.

In the real data analysis of Chapter 2, we applied both semi-parametric (Cox PH model) and parametric (Weibll model) to model the survival time, in which we use the semi-parametric model to check whether the parametric model assumption is appropriate here. The Generalized Weibull distribution is applied in the simulation study to check how reliable the proposed joint model in estimating the fixed effect when the assumption on the survival time is violated.

This joint modeling approach can address the following two objectives in the analysis of the Nun Study data: (1) understanding the within subject progression pattern of the cognitive status to dementia (2) understanding the relationship between the progression pattern to dementia and time to death. The detailed methodology, simulation study results, and results of the application to the Nun Study data are discussed in Chapter 2.

### 1.2.2 SEMI-MARKOV MODEL

The model introduced in 1.2.1 works well if assessment times are equally or approximately equally spaced. Biases may introduce if the observation time is not equally spaced. In order to further incorporate the time related to each transition, including

death, in the Nun study, we investigate the applicability of the semi-Markov model to the current problem. The semi-Markov model is more flexible than the Markov model in that it makes fewer assumptions, at the cost of higher model complexity.

In the semi-Markov setting, not only each transition status but also the waiting times between transitions are needed [9]. However, due to the periodical assessment schedule, the transition instants, except time to death, are interval censored, which further complicates the problem.

An important contribution is credited to Kang and Lagakos [29] who introduced a multi-state semi-Markov process with at least one state that has time homogenous transition intensity, namely, the holding time at that state is exponentially distributed. In that case, they were able to divide a long trajectory into smaller fragments according to the time homogenous transition intensity state. An alternative approach based on the use of phase type sojourn distributions and hidden Markov models is presented by Titman and Sharples [53].

More details about the challenges of the semi-Markov models and literature reviews are discussed in Section 3. We will discuss some numerical issue here. For interval-censored survival times, there are two main approaches, EM algorithm and direct integration over the intervals. Turnbull [50] derived a widely used EM-like iteration procedure "Product limit estimator". This yields an EM algorithm for computing the non-parametric maximum likelihood, which is not suitable in the parametric setting where directly integrating out the unobserved transition instants is the standard approach [16, 39]. However, the numerical implementation is quite computationally burdensome, especially for the multi-state model, where the orders of integrations are dictated by the number of states and whenever backward transitions are allowed.

For example, in the Nun Study data, we have 5 possible states with backward transitions between 3 transient states, which make the order of integration up to 8. The widely used numerical methods for high-dimensional integration are Adaptive Quadrature (AQ), Pseudo-Monte Carlo (PMC) and Quasi-Monte Carlo (QMC) methods.

Suppose, we want to approximate the integral

$$I(f) = \int_D f(x)dx,$$

where $D \in R^p$ is the pre-specified integration domain.

**AQ method:** The idea behind the AQ is to approximate an integral using a quadrature rule to adaptively redefine the sub-interval for the integral domain until the pre-specified tolerance reached. The procedures for AQ as follows:

- Step 0: Choose the tolerance $\tau_0$, a quadrature rule, such as the Trapezoidal Rule and a composite rule, on domain $D$ and set $k = 0$.
- Step 1: Estimate $I(f)$ using the quadrature rule and denote it as $I_0(f)$.
- Step 2: Set $k = k + 1$ and $I_k(f) = \sum_i \int_{D_i} f(x)dx$, where $D_i$ is the subdomain of domain $D$.
- Step 3: $\tau = I_k(f) - I_{k-1}(f)$, repeat step 2 and 3 until $\tau$ is less than $\tau_0$.

**PMC method:** The basic principle of the PMC method in evaluating the integral is to replace the continuous average by a discrete average over randomly choose points. It has the form of $I(f) = \frac{1}{n}\sum_{i=1}^{n} f(t_i)$, where $t_1, \cdots, t_n$ are independently and randomly chosen from the domain $D$.

**QMC method:** QMC method states in the same way as PMC excepting randomly selected points. QMC uses a low-discrepancy sequence such as the Halton

sequence, the Sobol sequence, or the Faure sequence, which is none-random but more uniformly distributed in the domain of integration. For example, the Halton sequence is generated by a prime number $r(r \geq 2)$ as base. To generate the sequence for base $r$, we start by dividing the interval (0,1) in $r$ths, then in $r^2$ths, $r^3$ths, etc. For $r = 2$, it generates

$$1/2, 1/4, 3/4, 1/8, 5/8, 3/8, 7/8, 1/16, 9/16, \cdots$$

For $r = 3$, it generates

$$1/3, 2/3, 1/9, 4/9, 7/9, 2/9, 5/9, 8/9, 1/27, \cdots$$

After combining and transferring multiple sequences with different primes, we get a sequence of points from high dimensional domain $D$.

The AQ leads to a very stable result and is highly recommended for low-dimension integrations [43]. The QMC method is a little less accurate than AQ, but is way more accurate than PMC (100 draws in QMC is more accurate than 1000 draws of PMC in the example of Brat [7] ). Regarding the computational time to the order of integration, it is approximately exponential for adaptive quadrature, while linear for QMC [7, 18]. Therefore, in this dissertation, we implement the QMC to calculate the higher order integration (up to 8).

### 1.3 Competing Risks

The term "competing risks" refers to the situation when more than one type of failure can occur, and only the smallest event time is recorded. At the same time, the event time is often subject to interval censoring due to the periodic assessment schedule.

Cause-specific hazard (CSH) and cumulative incidence function (CIF) are two functions of particular interest in the competing risk study.

The CSH of the $k$th ($k = 1, 2$) competing event, referring to the instantaneous rate of failure due to cause k at a specific time in the presence of all other failures types, is defined as

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

where $\varepsilon$ is an indicator for the failure type.

The CIF of the $k$th ($k = 1, 2$) competing event, referring to the probability of occurrence of failure due to cause $k$ by a specific time, is defined by

$$F_k(t) = \Pr(T \leq t, \varepsilon = k) = \int_0^t \lambda_k(s) \exp\Big\{ -\Lambda_1(s^-) + \Lambda_2(s^-) \Big\} \mathrm{d}s, \tag{1.3}$$

where $\Lambda_k(s) = \int_0^s \lambda_k(u) \mathrm{d}u$. It is clear that the summation of all the CIFs is less than or equal to 1.

The standard approach of modeling CIF is to make use of formula 1.3 and model the CSHs of all causes. Various models have been considered in the literature. Among others, the Cox proportional hazards (PH) model is unarguably the most popular choice. The PH model postulates that

$$\lambda_k(t|\boldsymbol{Z}) = \lambda_{k0}(t) \exp\Big\{ \boldsymbol{Z}^\top \boldsymbol{\beta}_k \Big\},$$

where $\lambda_{k0}(t)$ is the baseline hazard and $\boldsymbol{\beta}_k$ is the vector of regression coefficients associated with the vector of covariates $\boldsymbol{Z}$. This approach assumes different baseline hazards and different regression coefficients for different failure types. Anderson et al. [1] studied the estimator of the predicted CIF for a given set of covariates and derived a variance estimator.

Another popular hazard based model is the additive risk model, which assumes an additive covariate effect:

$$\lambda_k(t|\boldsymbol{Z}) = \lambda_{k0}(t) + \boldsymbol{Z}^\top \boldsymbol{\beta}_k.$$

Shen and Cheng [49] studied this model, presented an approach to constructing simultaneous confidence intervals of CIF and applied this model to analyze a melanoma data.

In randomized trials, where covariates adjustment is less of a concern, non-parametric estimators are appealing, but the current literature seems to be more focused on the CIF, because the CIF is easier to estimate non-parametrically as compared to the CSH.

A natural estimator of $F_k(t)$ is

$$\widehat{F}_k(t) = \Pr(T \leq t, \varepsilon = k) = \sum_{t_i \leq t} \widehat{S}(t_{i-1}) \mathrm{d}\widehat{\Lambda}_k(t_i),$$

where $\widehat{\Lambda}_k(s) = \sum_{t_i \leq t} d_i^k / Y_i$ is the Nelson-Aalen estimator, and $\widehat{S}(t) = \sum_{t_i \leq t} [1 - (d_i^1 + d_i^2)/Y_i]$ is the Kaplan-Meier estimator. Here, $d_i^1$ and $d_i^2$ are the number of failures, up to time $t_i$, of type 1 and 2, respectively; $Y_i$ is the size of the at risk set prior to time $t_i$. Lin [37] described the large sample properties of this estimator and demonstrated how to construct the simultaneous confidence band for the CIF curve through the counting process Martingale formulation.

When interval censoring is present, Frydman and Liu [17] extended the method in Turnbull [50] and developed a non-parametric maximum likelihood estimation procedure (NPMLE) for the CIF. This NPMLE maximizes the non-parametric likelihood with all the time intervals rearranged.

However, CIF and CSH represents two different aspects of competing risk study. The former refers to the probability of occurrence for a particular failure type, whereas the latter refers the instantaneous rate of a particular failure in the presence of other failure types. In other words, the CIF is a kind of average over CSHs. As a result, it is more insensitive than CSH to the covariate effect change. Therefore, Latouche et al. [36] and Hinchliffe and Paul [22] both proposed that a competing risk analysis should report results on both CSHs and CIFs side-by-side.

Most previous researches in CSH are focused on the right-censored data. It is of great interest to propose a method that can apply to the interval censored data. Motivated by Betensky et al. [4], who used a modified Expectation-Maximization (EM) algorithm based on the local likelihood, the so-called "local EM algorithm", to obtain a smooth estimate of the local hazard function for participants subject to only one type of failure with interval censored event times. We extended this method to the competing risk data. Details are presented in Chapter 4.

## 1.4 Outline of the Dissertation

The remainder of this dissertation is organized as follows.

In Chapter 2, we present the joint model of a four-state Markov chain and the time-to-death process, where correlation between these two processes is explained by common risk factors and a shared random effect. Robustness in certain aspects of the model is assessed by numerical studies.

Chapter 3 describes the semi-Markov model we proposed, where detailed model assumptions and model fitting procedures are provided. An approximate likelihood function is proposed to alleviate the computational burden. Numerical simulation

results are given followed by a summary of findings from the application to Nun's data.

We provide in Chapter 4 a non-parametric local-EM algorithm for smooth estimation of the CSH function in the presence of competing risks and interval censoring. Detailed methodology and a brief numerical study are included.

Finally in Chapter 5, we offer some potential areas for future study.

# Chapter 2 Markov Transition Model to Dementia with Death as a Competing Event

## 2.1 Introduction

In clinical trials and observational studies, it is common that the occurrence of the key event is censored by some competing risk such as disease-related dropout, which could cause non-ignorable missing data. More specifically, in most longitudinal studies on progression to a certain disease when the target population is elderly subjects, death is one of the competing risks. In the Nun study, among the total of 461 subjects of the final analytic sample for parameter estimating, almost half (n=225) died before converting to dementia. Several existing approaches have been developed in joint analysis of the longitudinal measurements and competing risks time-to-event data. Xu and Zeger [51] proposed a latent variable model to model the relationship between time-to-event data, longitudinal response, and covariates, in which covariates could only affect the longitudinal response through its influence on an assumed latent process. Elashoff et al. [13] suggested joint modeling of the repeated measures and competing risk failure time data by using latent random variables and common covariates to link the sub-models. However, few involve categorical responses that characterize these data.

Salazar el al. [45] proposed a suitable approach to the problem by defining a multistate Markov chain to model the progression of dementia in which death was treated as a competing absorbing state to dementia. A possible alternative is to model the competing risk of death without a dementia as a continuous variable. To this end this dissertation incorporates the Weibull model and Cox proportional hazards (PH) model into Salazar's Markov model assuming a shared random effect [2]. Specifically,

we introduced a random effect into the model to take into account for the correlation between the survival time and the transition states that is not explained by the model based solely on diagnostic effects in a similar spirit of Xu and Zeger [55]. The closed-form expressions for the conditional marginal likelihood function are derived. The model's stability to the violation of the assumption on the distributional form of survival is tested in simulation studies.

The dissertation is organized as follows: the model likelihood functions are constructed in Section 2.2; a simulation study is presented in Section 2.3; the application to the Nun Study data is presented in Section 2.4; and a summary of the findings is presented in Section 2.5.

## 2.2 Model and Estimation

### 2.2.1 Salazar's Multi-State Markov Model

Suppose there are $m$ subjects in the study. For a particular subject, let $\boldsymbol{Y} = (Y_1, Y_2, \cdots, Y_n)$ denote the random vector representing the observed cognitive states at n different ordered discrete occasions. Assume the Markov property holds [6, 24], that is, the conditional distribution $f(Y_k|Y_1, \cdots, Y_{k-1})$ is identical to the conditional distribution $f(Y_k|Y_{k-1})$ for $k = 2, \cdots, n$. Then conditioned on $Y_1$, the joint distribution of the random vector $\boldsymbol{Y}$ can be written as

$$f\Big(\boldsymbol{Y}\Big|Y_1\Big) = f\Big(Y_2, Y_3, \cdots, Y_n\Big|Y_1\Big) = f\Big(Y_2\Big|Y_1\Big) f\Big(Y_3\Big|Y_2\Big) \cdots f\Big(Y_n\Big|Y_{n-1}\Big).$$

In order to simplify the notation, we can use $\mathrm{P}_{Y_{k-1},Y_k} = f(Y_k|Y_{k-1})$ to denote the one step transition probability from state $Y_{k-1}$ to state $Y_k$. So for instance, if $Y_{k-1} = s$

and $Y_k = v$, then $P_{sv}$ represents the probability of transition from state $s$ to state $v$ in the $k$th visits.

In the example to be discussed later, the Nun study data, the status of a participant at each visit was recorded as being one of the states: 1=intact cognition, 2=mild cognitive impairments (M.C.I.), 3= global impairments (G.I.), or 4=dementia ([51]). The participants were followed during the study period until death occurred. The conditional distribution of the status of an individual participant at an arbitrary examination given her status at previous examinations was assumed to have the Markov property, i.e., that status at the examination depended on only the most recent previous examination and was independent of status at other previous examinations. Following Salazar et al. [45], a multi-state Markov chain was used to model transitions from one state to another, in which state 1-3 were considered transient states, whereas state 4 and death (state 5) were absorbing states as shown in Figure 2.1.

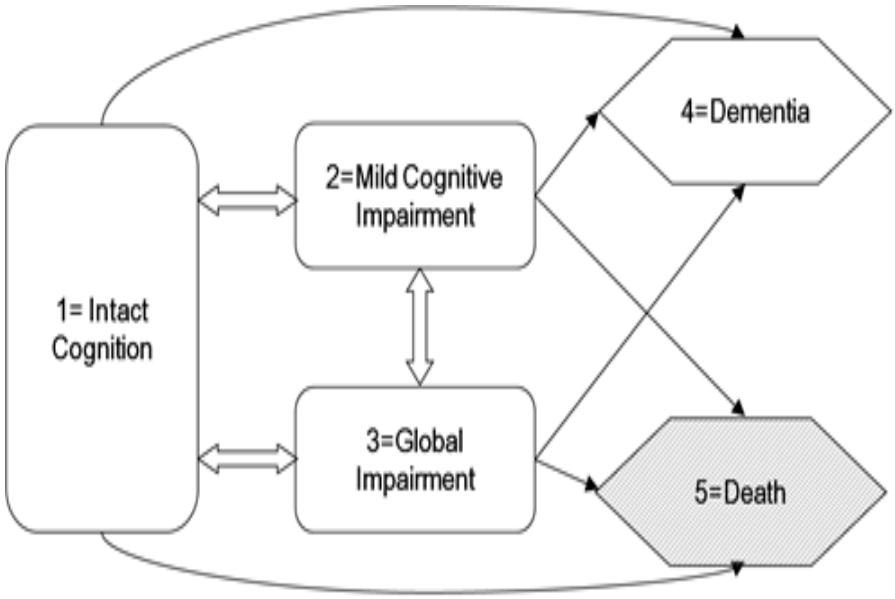


Figure 2.1: Possible one step transitions between three transient states (1) intact cognition (2) M.C.I. (3) G.I. and two absorbing states (4) dementia (5) death

Thus the one-step transition probability matrix could be presented in the form of

$$\begin{bmatrix} \mathrm{P}_{11}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{12}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{13}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{14}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{15}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) \\ \mathrm{P}_{21}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{22}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{23}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{24}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{25}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) \\ \mathrm{P}_{31}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{32}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{33}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{34}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{35}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Here $\boldsymbol{X}$ is the vector of covariates and $\boldsymbol{\gamma}$ is the vector of random effects.

According to Salazar et al. [45], a multinomial logit parameterization could be applied to link these transition probabilities with the fixed and random effects.

$$\log\left\{\frac{\mathrm{P}_{sv}(\theta_{sv}|\boldsymbol{X},\boldsymbol{\gamma})}{\mathrm{P}_{s1}(\theta_{s1}|\boldsymbol{X},\boldsymbol{\gamma})}\right\} = \alpha_v + \boldsymbol{X}^\top\boldsymbol{\beta_v} + \xi_{sv} + \boldsymbol{W}^\top\boldsymbol{\gamma}, \qquad v = 2,3,4,5; s = 1,2,3.$$

Here $\Theta$ represents the set of all the unknown parameters, $\boldsymbol{\alpha} = (\alpha_2, \alpha_3, \alpha_4, \alpha_5)$ is the vector of intercepts, $\boldsymbol{\beta}_v$ is the vector of unknown fixed effects for covariates $\boldsymbol{X}$ and $\xi_v^s$ is the set of unknown fixed effects for the prior state $s$ and current state $v$. Also, $\boldsymbol{\gamma}$ is the vector of unobserved random effects associated with the subject. The formulation of Salazar's model in terms of logit functions allows us to find the closed expression for each transition probability as follows

$$\mathrm{P}_{sv}(\theta|\boldsymbol{X},\boldsymbol{\gamma}) = \begin{cases} \frac{1}{1+\sum_{h=2}^{5}\exp(\alpha_h+\boldsymbol{X}^\top\boldsymbol{\beta}_h+\xi_h^s+\boldsymbol{W}^\top\boldsymbol{\gamma})}, & \text{for } v = 1 \\ \frac{\exp(\alpha_v+\boldsymbol{X}^\top\boldsymbol{\beta}_v+\xi_v^s+\boldsymbol{W}^\top\boldsymbol{\gamma})}{1+\sum_{h=2}^{5}\exp(\alpha_h+\boldsymbol{X}^\top\boldsymbol{\beta}_h+\xi_h^s+\boldsymbol{W}^\top\boldsymbol{\gamma})}, & \text{for } v > 1. \end{cases}$$

Therefore, based on the conditional distribution of $f(y_2, y_3, \cdots, y_n|y_1)$ the marginal likelihood function for the particular subject is

$$L(\Theta|X) = \int_{\Omega} \prod_{l=2}^{n} \prod_{\substack{s=1,\cdots,3 \\ v=1,\cdots,5}} \left\{ \mathrm{P}_{sv}(\Theta|\boldsymbol{X}, \boldsymbol{\gamma}) \right\}^{\delta_{y_{l-1},s}\delta_{y_l,v}} h(\boldsymbol{\gamma}) \mathrm{d}\boldsymbol{\gamma}, \tag{2.1}$$

with $\Omega$ denoting the support for the distribution of the random vector $\boldsymbol{\gamma}$ whose probability density function is denoted by $h(\cdot)$. Here $\delta_{y_{l-1},s}$ and $\delta_{y_l,v}$ are indicator functions valued at 1 if $y_{l-1} = s$ and $y_{l-1} = v$ , and 0 otherwise. The overall likelihood function can be obtained by evaluating the product of (2.1) across the subjects under study.

### 2.2.2 Models with Weibull and Cox Proportional Survival

In Salazar's model death is modeled as the competing absorbing state to dementia. A possible alternative approach is to incorporate information on the actual survival times from death of the subjects into the stochastic system. The data of interest involves multinomial responses and the parameterization of a polychotomous logit under a discrete time Markov framework complicating the problem. The hypothesis is that the survival time of those subjects who die without incurring a dementia come from certain parametric or semi-parametric distribution which shares the same random effects used in the Markov transition model. Additionally, these two pieces are conditionally independent given the random effects and their corresponding predictor variables.

In contrast with Salazar's model, the transition probabilities among cognitive states are modeled with a four-state Markov chain, same transient states but dementia being

the only absorbing state. The one-step transition probability matrix now becomes

$$\begin{bmatrix} \mathrm{P}_{11}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{12}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{13}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{14}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) \\ \mathrm{P}_{21}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{22}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{23}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{24}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) \\ \mathrm{P}_{31}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{32}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{33}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) & \mathrm{P}_{34}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Each transition probability $\mathrm{P}_{sv}$ could be postulated in the form of

$$\mathrm{P}_{sv}\Big(\Theta\Big|\boldsymbol{X},\boldsymbol{\gamma}\Big) = \begin{cases} \frac{1}{1+\sum_{h=2}^{4}\exp(\alpha_h+\boldsymbol{X}^\top\boldsymbol{\beta}_h+\xi_h^s+\boldsymbol{W}^\top\boldsymbol{\gamma})}, & \text{for v} = 1 \\ \frac{\exp(\alpha_v+\boldsymbol{X}^\top\boldsymbol{\beta}_v+\xi_v^s+\boldsymbol{W}^\top\boldsymbol{\gamma})}{1+\sum_{h=2}^{4}\exp(\alpha_h+\boldsymbol{X}^\top\boldsymbol{\beta}_h+\xi_h^s+\boldsymbol{W}^\top\boldsymbol{\gamma})}, & \text{for v} > 1. \end{cases} \tag{2.2}$$

Assume the survival time (that is, time on study) could be modeled by the parametric Weibull distribution or the semi-parametric Cox PH model. The semi-parametric Cox PH model is used to validate the parametric Weibull model assumption. Therefore, both the parametric and semi-parametric methods are applied to the Nun Study data and the corresponding simulation results and real data analysis results are compared in Section 2.3 and 2.4.

When the survival time follows the Weibull distribution, the survival time $S \sim \text{Weibull}(r,\mu)$, where $\mu = \exp(\eta_0 + \boldsymbol{Z}^\top\boldsymbol{\eta} + \boldsymbol{W}^\top\boldsymbol{\gamma})$. The probability of a subject failing from the competing risk of death is

$$\begin{aligned} \pi_w\Big(S=t\Big|\Theta,\boldsymbol{\gamma}\Big) &= \Big\{r\exp\big(\eta_0+\boldsymbol{Z}^\top\boldsymbol{\eta}+\boldsymbol{W}^\top\boldsymbol{\gamma}\big)t^{r-1}\Big\}^{\tau} \\ &\times \Big[\exp\big\{-\exp(\eta_0+\boldsymbol{Z}^\top\boldsymbol{\eta}+\boldsymbol{W}^\top\boldsymbol{\gamma})t^r\big\}\Big], r>0. \end{aligned}$$

Here $\tau$ is the indicator function valued at 1 if the $i$th subject died at time and 0 otherwise. $\Theta$ be the parameter vector associated with both the transition probability

and the probability of death. For each subject under study, the conditional marginal likelihood function for the $i$th subject can be rewritten as

$$L\Big(\Theta\Big|\boldsymbol{X}\Big) = \int_{\Omega} \prod_{l=2}^{n} \times \prod_{\substack{s=1,2,3 \\ v=1,\cdots,5}} \Big\{ \mathrm{P}_{sv}(\Theta|\boldsymbol{X},\boldsymbol{\gamma}) \Big\}^{\delta_{y_{l-1},s}\delta_{y_l,v}} \times \Pi_w\Big(\Theta\Big|\boldsymbol{Z},\boldsymbol{\gamma}\Big) h(\boldsymbol{\gamma}) \mathrm{d}\boldsymbol{\gamma}. \quad (2.3)$$

In the Cox proportional hazards model, we assume the hazard function has the form

$$\lambda(S = t|\Theta,\gamma) = \lambda_0(t) \exp\big(\eta_0 + Z'\eta + W'\gamma\big).$$

Here $\lambda_0(t)$ is the baseline hazard and $\mu = \exp\left(\eta_0 + Z'\eta + W'\gamma\right) > 0$ . According to Cox et al. [10], the contribution to the partial likelihood from the $i$th subject failing from the competing risk of death is

$$\pi_c(S = t|\Theta,\gamma) = \left( \frac{\mu}{\sum_{t_j > t} \mu_j} \right)^{I_{\{\tau=1\}}}.$$

The conditional on the baseline state likelihood function can be rewritten as

$$L_c(\Theta|X,Z) = \int \prod_{l=2}^{n} \prod_{s=1\ldots3, v=1\ldots4} P_{sv}(\Theta|X,\gamma)^{\delta_{y_{l-1},s}\delta_{y_{l},v}} \times \pi_c(\Theta|Z,\gamma) h(\gamma) d\gamma. \quad (2.4)$$

### 2.2.3 Parameter Estimation

The parameter estimation is implemented by maximizing the conditional likelihood $L(\Theta|X,Z)$ . In particular, all the calculations are approached by SAS PROC NLMIXED procedure. Assuming that the random effect is distributed as a $N(0,\sigma^2)$, both of the log likelihood functions (in (2.3) and (2.4)) can be maximized using the Double-Dogleg method combined with the adaptive Gauss-Hermite quadrature method (Raudenbush et al. [44]) to numerically evaluate the integrations and produce the parameter estimates. The likelihood function is not convex in the parameters, therefore convergence

of the optimization algorithm is not guaranteed for an arbitrary set of initial values. It is advisable to start with multiple sets of initial values and select the maximizers accordingly. The estimates of the standard errors are computed by Fisher's information method.

## 2.3 Simulations

The main purpose of the simulation study is to examine the sensitivity of the MLEs of $\beta$ to the violations of the Weibull model assumption or Cox PH model assumption on the survival time. The goal is to quantify how the distributional form for the survival term affects the model estimates associated with the fixed effects in Equation 2.2. The criteria are the bias and the mean squared errors of the MLEs.

Simulations were set to have 1000 iterations, with each containing either 200 or 500 subjects. The corresponding computation time of sample size 200 and 500 by using Intel i5-650 processor (4M Cache, 3.2 GHz) are 13.35 hours and 31.21 hours respectively. Each subject has up to ten follow-up waves starting from a baseline state of intact cognition. Four cases are considered:

1. Total of 200 subjects generated with prior distribution of survival being Weibull
2. Total of 500 subjects generated with prior distribution of survival being Weibull
3. Total of 200 subjects generated with prior distribution of survival being Generalized Weibull
4. Total of 500 subjects generated with prior distribution of survival being Generalized Weibull

The Generalized Weibull distribution $WG(r, \mu, \theta)$ has the hazard function, $h(t) = r\mu/\theta(1 + \mu t^r)^{1/\theta - 1} t^{r-1}$, where $t \geq 0$, $r > 0$ , $\mu > 0$ and $\theta > 0$ (Foucher et al. [16]).

If $\theta$ is 1, the Weibull formulation is obtained. In the simulation, set $r$ to be a fixed number 2.8593 and $log(\mu)$ be a linear function of current age and APOE4 status. The range of $\mu$ in the simulation lies between 0.0004 and 0.0103 and the mean value of $\mu$ is 0.0013. These choices are motivated by the application discussed in Section 2.4. Additionally, choose $\theta = 0.5$, 1, 2 and 4 separately. The plots of hazard functions of the Generalized Weibull distribution with $r = 2.8593$ and $\mu = 00013$ were shown on Figure 2.2. Note that the proportional hazards assumption holds only if $\theta = 1$.

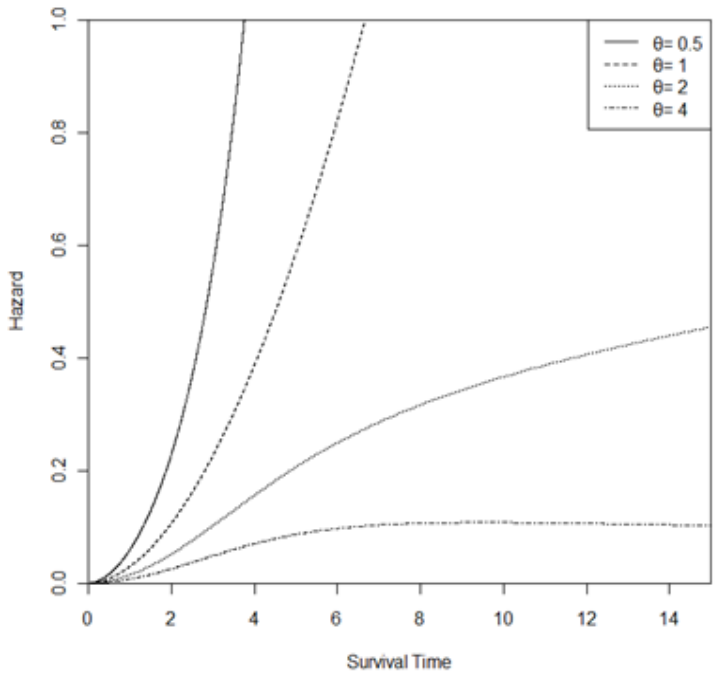


Figure 2.2: Hazard function of a Generalized Weibull Distribution with $r = 2.8593$ and $\mu = 0.0013$

Thus, two sets of comparisons could be explored: first, the effects of varying the sample size, and second, the effects of violating the original model assumption on the distributional form of survival term with a possible alternative.

In both situations, the transition probabilities were dependent on three covariates: current age (denoted as age), prior state (IC = intact cognition or M.C.I. or G.I. (the reference category)), and the presence/absence of an apolipoprotein E-4 allele (APOE4). The covariates entered in the survival model were age at entry and the APOE4 status of the subject. All the simulations were done using the IML procedure in SAS system. The results are presented in Table 2.1 and Table 2.2.

| | | | $\theta$=0.5 | | | | $\theta$=1 | | | | $\theta$=2 | | | | $\theta$=4 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Weibull | | Cox PH | | Weibull | | Cox PH | | Weibull | | Cox PH | | Weibull | | Cox PH | |
| Risk Factors | State | Para | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. |
| Markov chain | | | | | | | | | | | | | | | | | | |
| Age | 2 | 0.101 | -0.003 | 0.001 | -0.057 | 0.004 | -0.001 | 0.001 | -0.031 | 0.002 | 0.003 | 0.001 | -0.017 | 0.001 | 0.000 | 0.001 | -0.011 | 0.001 |
| | 3 | 0.181 | 0.001 | 0.002 | -0.059 | 0.005 | -0.001 | 0.002 | -0.028 | 0.003 | 0.002 | 0.001 | -0.015 | 0.002 | -0.003 | 0.001 | -0.010 | 0.001 |
| | 4 | 0.177 | 0.067 | 0.008 | 0.014 | 0.004 | 0.061 | 0.006 | 0.031 | 0.004 | 0.054 | 0.005 | 0.038 | 0.004 | 0.051 | 0.005 | 0.041 | 0.003 |
| APOE4 | 2 | 0.859 | 0.015 | 0.209 | -0.102 | 0.179 | -0.015 | 0.211 | -0.082 | 0.169 | -0.002 | 0.181 | -0.064 | 0.200 | -0.027 | 0.173 | -0.025 | 0.205 |
| | 3 | 1.313 | -0.001 | 0.287 | -0.106 | 0.245 | -0.050 | 0.297 | -0.092 | 0.253 | 0.010 | 0.211 | -0.044 | 0.249 | -0.036 | 0.200 | -0.044 | 0.249 |
| | 4 | 1.428 | 0.121 | 0.328 | -0.031 | 0.362 | -0.060 | 0.395 | -0.042 | 0.394 | 0.071 | 0.319 | -0.026 | 0.369 | 0.059 | 0.281 | 0.009 | 0.345 |
| Prior states: | | | | | | | | | | | | | | | | | | |
| IC | 2 | -1.11 | 0.028 | 0.148 | -0.001 | 0.172 | 0.004 | 0.160 | 0.009 | 0.155 | -0.001 | 0.164 | -0.009 | 0.148 | 0.003 | 0.150 | 0.009 | 0.119 |
| | 3 | -3.71 | 0.003 | 0.209 | -0.016 | 0.224 | 0.034 | 0.181 | 0.005 | 0.174 | 0.011 | 0.177 | -0.025 | 0.192 | -0.013 | 0.171 | 0.018 | 0.165 |
| | 4 | -5.23 | -0.892 | 6.221 | -0.928 | 5.858 | -0.617 | 4.269 | -0.987 | 5.698 | -0.659 | 4.396 | -0.646 | 4.088 | -0.653 | 4 | -0.619 | 4.109 |
| MCI | 2 | 0.74 | -0.026 | 0.148 | 0.075 | 0.168 | 0.012 | 0.150 | 0.067 | 0.167 | 0.054 | 0.161 | 0.058 | 0.160 | 0.050 | 0.130 | 0.072 | 0.134 |
| | 3 | -2.31 | -0.031 | 0.197 | 0.050 | 0.189 | 0.081 | 0.174 | 0.095 | 0.183 | 0.061 | 0.151 | 0.066 | 0.165 | 0.061 | 0.129 | 0.089 | 0.158 |
| | 4 | -1.93 | 0.155 | 0.407 | 0.174 | 0.322 | 0.171 | 0.28 | 0.174 | 0.316 | 0.193 | 0.268 | 0.13 | 0.236 | 0.138 | 0.213 | 0.177 | 0.224 |

Table 2.1: Bias and mean squared error of the model parameters based on the trajectories of 200 subjects when the likelihood for the survival assumes Weibull or Cox model

| | | | $\theta$=0.5 | | | | $\theta$=1 | | | | $\theta$=2 | | | | $\theta$=4 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Weibull | | Cox PH | | Weibull | | Cox PH | | Weibull | | Cox PH | | Weibull | | Cox PH | |
| Risk Factors | State | Para | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. | Bias | M.S.E. |
| Markov chain | | | | | | | | | | | | | | | | | | |
| Age | 2 | 0.101 | -0.001 | 0.000 | -0.054 | 0.003 | 0.000 | 0.000 | -0.034 | 0.001 | -0.002 | 0.000 | -0.018 | 0.001 | -0.001 | 0.000 | -0.010 | 0.000 |
| | 3 | 0.181 | 0.000 | 0.001 | -0.056 | 0.004 | 0.000 | 0.001 | -0.036 | 0.002 | -0.001 | 0.000 | -0.020 | 0.001 | 0.000 | 0.000 | -0.010 | 0.001 |
| | 4 | 0.177 | 0.068 | 0.006 | 0.011 | 0.001 | 0.057 | 0.004 | 0.022 | 0.002 | 0.050 | 0.003 | 0.032 | 0.002 | 0.049 | 0.003 | 0.039 | 0.002 |
| APOE4 | 2 | 0.859 | -0.020 | 0.086 | -0.140 | 0.085 | -0.067 | 0.083 | -0.084 | 0.075 | -0.026 | 0.080 | -0.074 | 0.079 | -0.038 | 0.078 | -0.046 | 0.076 |
| | 3 | 1.313 | -0.033 | 0.104 | -0.157 | 0.113 | -0.093 | 0.113 | -0.092 | 0.104 | -0.036 | 0.109 | -0.082 | 0.102 | -0.052 | 0.088 | -0.038 | 0.103 |
| | 4 | 1.428 | 0.058 | 0.167 | -0.111 | 0.182 | -0.063 | 0.144 | -0.031 | 0.144 | -0.016 | 0.130 | -0.030 | 0.134 | -0.011 | 0.119 | -0.001 | 0.136 |
| Prior states: | | | | | | | | | | | | | | | | | | |
| IC | 2 | -1.110 | 0.058 | 0.081 | -0.034 | 0.068 | -0.001 | 0.068 | -0.016 | 0.067 | 0.050 | 0.057 | -0.011 | 0.061 | 0.004 | 0.056 | -0.004 | 0.054 |
| | 3 | -3.708 | 0.063 | 0.080 | -0.027 | 0.074 | 0.031 | 0.087 | -0.012 | 0.076 | 0.036 | 0.056 | 0.005 | 0.065 | -0.002 | 0.059 | -0.014 | 0.069 |
| | 4 | -5.226 | 0.018 | 0.533 | -0.128 | 1.079 | -0.040 | 0.706 | -0.106 | 0.913 | 0.011 | 0.496 | -0.032 | 0.581 | 0.051 | 0.460 | -0.138 | 0.724 |
| MCI | 2 | 0.740 | 0.034 | 0.078 | 0.044 | 0.075 | 0.008 | 0.071 | 0.064 | 0.067 | 0.073 | 0.062 | 0.055 | 0.064 | 0.068 | 0.061 | 0.054 | 0.059 |
| | 3 | -2.305 | 0.017 | 0.074 | 0.058 | 0.080 | 0.040 | 0.078 | 0.059 | 0.066 | 0.077 | 0.066 | 0.077 | 0.067 | 0.079 | 0.055 | 0.061 | 0.064 |
| | 4 | -1.931 | 0.153 | 0.149 | 0.196 | 0.167 | 0.130 | 0.118 | 0.208 | 0.132 | 0.188 | 0.120 | 0.164 | 0.104 | 0.194 | 0.100 | 0.151 | 0.105 |

Table 2.2: Bias and mean squared error of the model parameters based on the trajectories of 500 subjects when the likelihood for the survival assumes Weibull or Cox model

As expected, increasing the sample size improves the estimates in terms of reducing mean square error (MSE). The main savings is in the variance of the estimates since the bias stays almost the same with only one exception, the effect of the transition from intact cognition into dementia. Those biases are reduced considerably when the sample size increased. For example, the bias is -0.892 when sample size is 200 reduced to 0.018 when sample size is 500 for the Weibull model when $\theta = 0.5$. The huge change is due to that the simulation parameter for the transition from intact cognition into dementia is very small, -5.226, which will increase the chance of observing few transitions. However, the chance of observing few transitions will be very rare when the sample size is larger than 300. Similar results were obtained for sample sizes of 300 and 400 (not shown). The results show that as long as the sample size is larger than 300, then the result will have acceptable small MSE and bias.

There is not much difference in term of the bias and MSE when fitting the data assuming a Weibull model or Cox model. The maximum differences between a Weibull model and a Cox model are 1.4289 for MSE and 0.3699(7.08%) for bias.

In all, the results indicate that the maximum likelihood estimates are not sensitive to violations of the assumed Weibull or Cox PH model in the case when the Generalized Weibull Distribution is the true distribution.

### 2.4 Application to the Nun Study

The Nun Study began enrollment in 1991. The data consists of a cohort of 672 members of the School Sisters of Notre Dame born before 1917 and living in retirement communities in the Midwestern, eastern, and southern United States. The subjects were recruited in phases and received annual cognitive assessments with brain donation at death. Analyses were based on data from ten successive examinations. A total

| | Current Visit | | | |
|---|---|---|---|---|
| Prior Visit | Intact Cognition | M.C.I. | G.I. | Dementia |
| Intact Cognition | 593 | 197 | 54 | 5 |
| | 69.9% | 23.2% | 6.3% | 0.6% |
| M.C.I. | 177 | 697 | 136 | 82 |
| | 16.2% | 63.8% | 12.5% | 7.5% |
| G.I. | 16 | 39 | 184 | 75 |
| | 5.1% | 12.4% | 58.6% | 23.9% |
| Dementia | 0 | 0 | 0 | 81 |
| | 0.0% | 0.0% | 0.0% | 100.0% |

Table 2.3: Number of transitions in the Nun study

of 211 subjects were excluded from the study due to: only one cognitive assessment (128), presence of dementia at baseline visit (61) or missing APOE4 (22). The final analytic sample consisted of 461 participants, of which 74 survived without dementia, 162 developed dementia and 225 died before converting to dementia. The transitions among the cognitive states are summarized in Table 2.3 .

The covariates of interest are age, education level, APOE4 status, and prior state. For simplicity, education was not included in the model simulations; but was considered here since it is a well-known risk factor and found to be significantly associated with dementia in previous studies. The covariates entering in both of the two survival models were age at entry and APOE4 status. As shown in Figure 2.3 below, subjects were sub-grouped based on their APOE4 status and age at entry, and thus four Weibull probability plots were created as a preliminary look at the model assumption. The estimated cumulative distribution function was computed by Kaplan-Meier estimator in the LIFEREG procedure in SAS. The straight line represents the maximum likelihood fit, with the point wise parametric confidence bands on each side. The plots indicate that the assumed Weibull model fits the data reasonably well although

not perfect since skewness arises in the tail of the distribution for some of the groups. Similar results were obtained for Cox PH model, which are not shown.

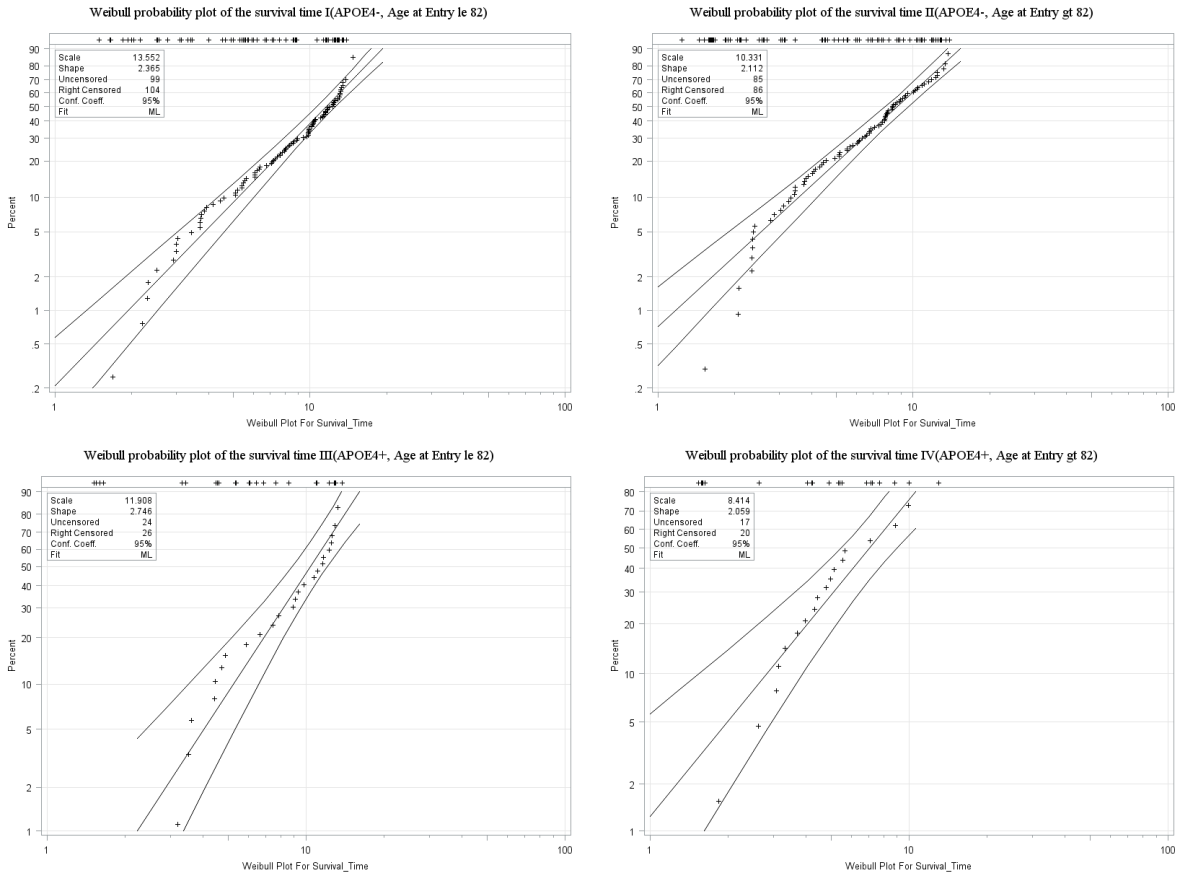


Figure 2.3: Weibull probability plots of the survival time for different cohorts in the Nun study

Since current age is the only interval level risk factor, there is interest in determining whether the linearity assumption between the logit of the transition probability and current age is adequate. To this end, we contrasted the linearity assumption against a piece wise constant assumption and test the adequacy of the linearity via the likelihood ratio test. Specifically, split the variable, current age, into 5, 10, 15 and 20 equally spaced bins, and estimate the effect of age for each bin. The resulting regression coefficients were then plotted against the age midpoint of each bin for the cases

| Bins | -2Log(Likelihood) | LRT | D.F. | P value |
|---|---|---|---|---|
| 5 | 5598.2 | 7.3 | 9 | 0.61 |
| 10 | 5577.9 | 27.6 | 24 | 0.28 |
| 15 | 5572.4 | 33.1 | 39 | 0.74 |
| 20 | 5554.2 | 51.3 | 54 | 0.58 |
| Linear | 5605.5 | | | |

Table 2.4: Fit statistics for linearity test of current age

of 10 and 20 bins given by initial state in Figure 2.4. For each initial state 2, 3, or 4 the coefficients appear to increase linearly with the age midpoints. The Likelihood Ratio Test for linearity is provided in Table 2.4 for 5, 10, 15 and 20 bins. Note that none of these tests are significant, supporting the linearity assumption for each state 2, 3, and 4. A similar analysis was conducted to check the linearity of baseline age in the survival component of the likelihood with the same result (which is not shown). In Table 2.4, the first and second column of each model lists the parameters and standard s of parameters obtained by SAS PROC NLIMXED. The third column lists the estimated standard error, which was obtained by using the bootstrap resampling method [11]. The two methods of estimating standard errors are almost the same. We found that the standard errors of transition parameter estimates of Weibull model are uniformly smaller than those of Cox PH model. This is likely due to the much larger estimate for the random effect in the Cox model (last line in Table 2.4).

Note that in either model, the regression coefficients for all three risk factors are positive and significant at the $P < 0.05$ level indicating that each factor promotes transitions into each impaired state at the next assessment with only one exception where the p-value for the regression coefficient is only marginally significant (P = 0.09). As noted above, the effect of age is linear. Referring to the Weibull model, the effect of an APOE 4 carrier is to promote transitions into M.C.I, G.I., and dementia as opposed to a transition into the intact cognition with estimated odds ratios (OR)

| | | Weibull Model | | | Cox PH Model | | |
|---|---|---|---|---|---|---|---|
| Risk Factors | State | Estimates | s.e. | e.s.e. | Estimates | s.e. | e.s.e. |
| Markov chain | | | | | | | |
| Age | 2 | 0.1010* | 0.017 | 0.016 | 0.1129* | 0.020 | 0.020 |
| | 3 | 0.1813* | 0.020 | 0.019 | 0.1955* | 0.023 | 0.022 |
| | 4 | 0.1772* | 0.024 | 0.021 | 0.1873* | 0.026 | 0.027 |
| APOE4 | 2 | 0.8585* | 0.244 | 0.307 | 1.1765* | 0.336 | 0.450 |
| | 3 | 1.3132* | 0.274 | 0.354 | 1.6383* | 0.358 | 0.492 |
| | 4 | 1.4282* | 0.306 | 0.353 | 1.7335* | 0.383 | 0.488 |
| Education: | | | | | | | |
| $<$ 16 years | 2 | 1.5658* | 0.361 | 0.345 | 2.0148* | 0.491 | 0.489 |
| vs. $>$ 16 years | 3 | 1.6105* | 0.402 | 0.421 | 2.0493* | 0.521 | 0.572 |
| | 4 | 1.4504* | 0.446 | 0.461 | 1.8779* | 0.555 | 0.620 |
| 16 years | 2 | 0.4969* | 0.164 | 0.178 | 0.7549* | 0.246 | 0.238 |
| vs. $>$ 16 years | 3 | 0.5276* | 0.199 | 0.204 | 0.7786* | 0.270 | 0.258 |
| | 4 | 0.4032 | 0.239 | 0.228 | 0.6528* | 0.300 | 0.277 |
| Prior states: | | | | | | | |
| Intact Cognition | 2 | -1.1103* | 0.337 | 0.369 | -0.7579* | 0.369 | 0.433 |
| | 3 | -3.7083* | 0.329 | 0.417 | -3.3338* | 0.362 | 0.479 |
| | 4 | -5.2264* | 0.548 | 1.650 | -4.8818* | 0.570 | 1.435 |
| Mild Cognitive | 2 | 0.7399* | 0.328 | 0.330 | 0.4734 | 0.354 | 0.354 |
| Impairment | 3 | -2.3053* | 0.307 | 0.322 | -2.5663* | 0.335 | 0.337 |
| | 4 | -1.9313* | 0.328 | 0.318 | -2.2025* | 0.354 | 0.323 |
| Survival Part: | | | | | | | |
| Age at Entry | - | 0.1206* | 0.019 | 0.020 | 0.0982* | 0.014 | 0.015 |
| APOE4 | - | 0.4794* | 0.231 | 0.250 | 0.3937* | 0.175 | 0.185 |
| Sigma | - | 1.0026* | 0.116 | 0.134 | 1.6409* | 0.200 | 0.233 |

States: 2=Mild cognitive impairment, 3=Global impairment, 4=Dementia;
e.s.e. is the estimated standard error from bootstrap resampling method
Significant at $P < 0.05$

Table 2.5: Maximum likelihood estimates (SE) of model parameters in the Nun study for two models (base state: 1=Intact Cognition)

2.36, 3.72 and 4.17, respectively. Low education (<16 years) versus high education ($>$ 16 years) is associated with even larger ORs of 4.79, 5.01, and 4.26 for similar transitions. More modest ORs are obtained when comparing 16 years of education to $>$ 16 years of education yielding ORs of 1.64, 1.69, 1.50 for similar transitions. The corresponding ORs are 0.33, 0.025 and 0.0054 for prior state intact cognitive and are 2.10, 0.10 and 0.14 for prior state mild cognitive indicating that subjects tend to remain in their prior state. For all three risk factors, the Cox model yields uniformly larger ORs but their statistical significance is about the same due to the increase in

the standard error of the regression coefficients. Only baseline age and APOE carrier status predict time to death without dementia.

## 2.5 Conclusion and Discussion

Considerable literature has focused on characterizing the relationship between longitudinal response process and time-to-event data. In contrast, relatively little research has been done to accommodate multinomial responses, with even fewer relying on a polychotomous logit parameterization under a discrete-time Markov chain.

As an improvement to Salazar's multi-state Markov model, this dissertation fits a Weibull distribution and a Cox PH ditribution to model the time to death without a dementia and correlate this with the Markov transition model by incorporating a shared random effect. The simulation study showed model stability in terms of violations of the distributional assumption on survival time. More specifically, the maximum likelihood estimates are not sensitive to violations of the assumed Weibull model or Cox PH model assumption when, in fact, a Generalized Weibull model should be used instead. Also, the semi-parametric model has almost the same effect as the parametric model.

The application to the Nun Study data found that Age, APOE 4 carrier status, and low education are significant predictors of a transition to an impaired state as opposed to a transition to cognitively normal because all the coefficients associated with Age and APOE4 are significant and positive. Remaining cognitively intact favors the highly educated ($>$ 16 years education) which also agrees with the results from the previous models. Age and APOE 4 status are also significant predictors for dying without incurring a dementia. Age at entry is a protective for subjects from the competing risk of death since older subjects are more likely to become demented before

death.

Yu et al. [57] incorporated the missing portion of the likelihood due to baseline demented individuals into the follow-up likelihood by assuming the two share the same random effect. The complete marginal likelihood function for a subject with baseline can be written as

$$L(\Theta|\boldsymbol{X},\boldsymbol{Z}) = \int \prod_{l=2}^{n} \prod_{s=1.2,3,v=1,\ldots,5} \mathrm{P}_{sv}(\Theta|\boldsymbol{X},\boldsymbol{\gamma})^{\delta_{y_{l-1,s}}\delta_{y_{l,v}}} \times \pi_{y_1}(\Theta|\boldsymbol{Z},\boldsymbol{\gamma})h(\boldsymbol{\gamma})d\boldsymbol{\gamma}.$$

Here $\Theta$ is the set of parameters associated with the baseline response components. The probability of the baseline state $\pi_{y_1}(\Theta|\boldsymbol{X}_B,\boldsymbol{\gamma})$ was similarly modeled by using multinomial logistic regression as for the one-step transition probability $P_{sv}(\Theta|\boldsymbol{X},\boldsymbol{\gamma})$ in the follow-up likelihood. It will also be interesting to combine this approach with our model to find a complete likelihood function that accommodates all the three pieces baseline, follow-up, and survival.

Due to the Markov property assumption, the proposed method works well when the follow-up assessments are evenly spaced, but may lead to biased estimators when the visit times are derivation from the predetermined visit times. Therefore, one potential limitation of our proposed methods is its inability to handle the uneven assessments or skipped visits. The general imputation approaches for the missing data can used to deal with skipped visits But those imputation methods are generally very complex. One simple and popular strategy is so called "last observation carry forward (LOCF)". However, it is not recommended to use since this approach will introduce bias in the result [40]. Uneven assessments call for use of more complex models as discussed by Huzurbazar [24]. Another possible drawback of the proposed method is that the computational burden will become heavier in the current model if a complicated form of the random effects is adopted.

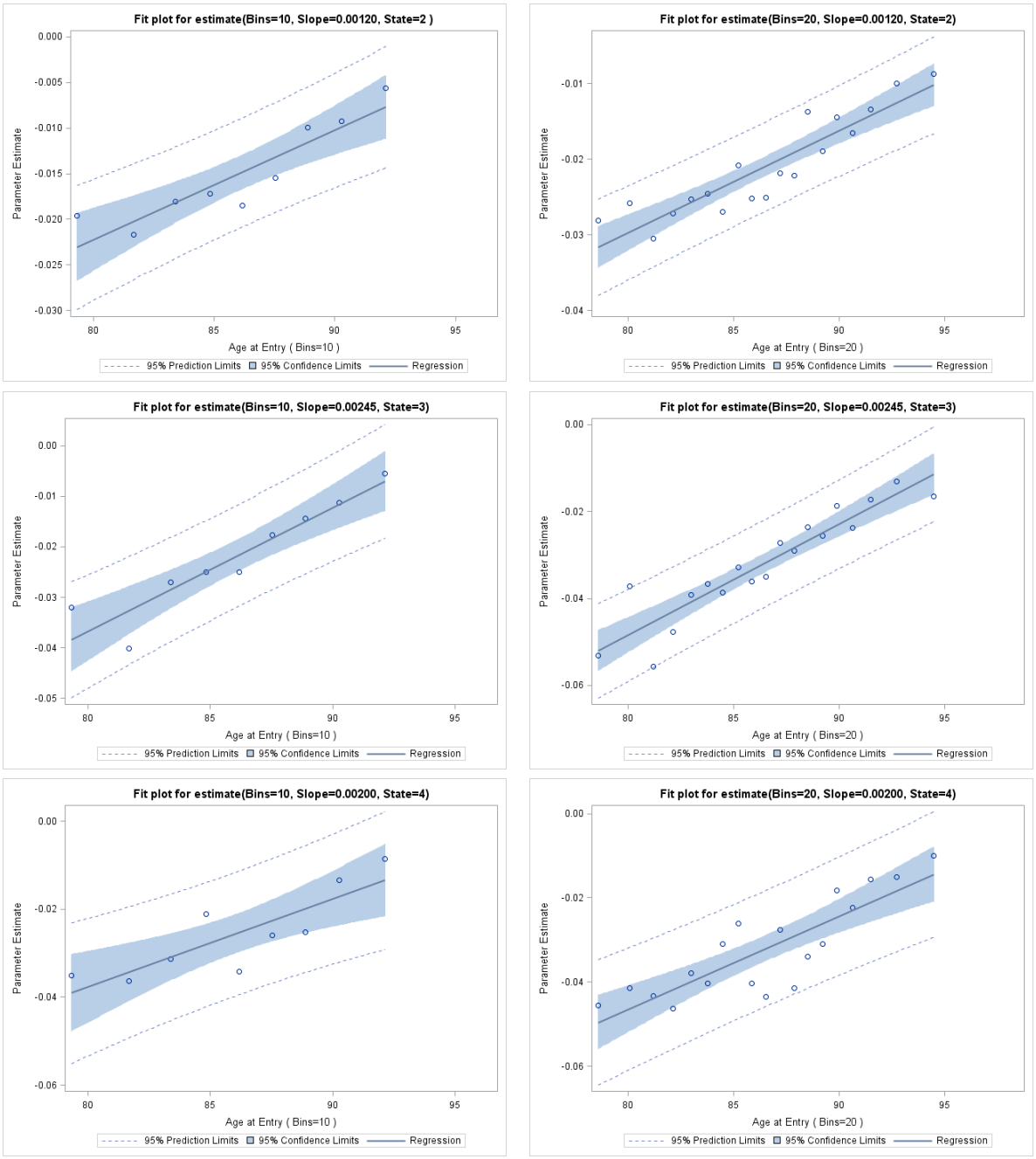


Figure 2.4: Assessment of linearity of current age in transition matrix using 10 and 20 age bins

# Chapter 3 Semi-Markov Models for Interval Censored Transient Cognitive States with Back Transitions and a Competing Risk

## 3.1 Introduction

In longitudinal analysis, the continuous-time multi-state stochastic process has a wide application in modeling the complex evolution of chronic diseases. Analysis of panel data is greatly simplified by the time homogeneous Markov assumption, especially when observations are made at some pre-specified evenly spaced time spots. Kalbfleisch and Lawless [27] proposed a quasi-Newton algorithm for maximum likelihood estimation that could effectively handle the case of unevenly spaced observation times.

Often it is the case that the transition intensities of the process depend on the time elapsed at the current state, which makes the process semi-Markov. There has been much literature on the application of semi-Markov models in very general statistical problems. When the exact transition times are fully observed, the likelihood function has a relatively elegant form, which also simplifies the subsequent maximization procedure [16]. The R package SemiMarkov recently developed by Listwon and Saint-Pierre [38] offers a convenient tool to implement general homogeneous semi-Markov models that could flexibly incorporate diagnostic covariates through parametric proportional hazards models. However, in many instances, the subjects are only periodically assessed resulting in interval censoring, with no information about the types of events between the observations and the associated transition instants. When the process only has right shift paths, namely, a subject can only visit a state at most once, and has only a small number of states, e.g., three or four, the length of all possible paths will be limited. In the parametric setting, the likelihood function will only involve in-

tegrations of low orders and therefore standard numerical methods such as Gaussian Quadrature or Monte-Carlo methods can be applied to approximate the likelihood [14, 15, 20, 39]. Nonparametric estimation is also possible via self-consistent estimators in the case of a unidirectional model without covariates [47]. Commenges [9] discusses the need to develop more stable and efficient algorithms when employing nonparametric inference for multistate models subject to interval censoring. A semi-parametric based on a penalized likelihood function for a three state progressive semi-Markov model with interval censored data is presented by Joly et al. [25, 26] Recently, Kapetanakis et al. [30] studied a three-state illness-death model with piecewise constant hazards in the presence of left, right and interval censoring.

Little work has been done to handle reverse transitions (namely, a subject can visit one state multiple times) in the presence of interval censoring, apparently due to the fact that reverse transitions will potentially lead to lengthy paths and hence prohibitively complicated high order integrations in the likelihood function. An important contribution is credited to Kang and Lagakos [29] who introduced a multi-state semi-Markov process with at least one state that has time homogeneous transition intensity, namely, the holding time at that state is exponentially distributed. In that case, they were able to divide a long trajectory into smaller fragments according to the time homogeneous transition intensity state. Although their method could be extended with minimal modification to incorporate time-independent covariates, dealing with time-dependent covariates may be problematic. An alternative approach based on the use of phase type sojourn distributions and hidden Markov models is presented by Titman and Sharples [53]. In the Nun study, one of our primary research interests is the effect of age (calendar time, with 15 years follow up period) on the holding time, which makes the approach of Kang and Lagakos inapplicable. We implement the quasi-Monte Carlo (QMC) method [7] which will provide considerably

better accuracy, with the expected integration error of the order of $N^{-1}$ ($N$ being the number of Halton sequence points from the high-dimensional integration space), to approximate the higher order integrations of the likelihood function.

A second issue in using a semi-Markov model is identifying the time origin, the exact time of entrance into the initial state (first observed state). For the semi-Markov model, the transition intensity of each state depends on the length of time at which each subject stayed at the current state. For the initial state, we do not have the exact time of entrance, which results in the left censoring for the holding time of the initial state. We identified some common strategies in the literature to deal with this problem. Kryscio and Abner [35] assume a unique time (age 60) as the time origin for all subjects. This works well if each subject is in the same initial state and all paths are right shift. Kapetanakis et al. [30] apply the EM inspired algorithm to find the unique age as the time origin. Satten and Sternberg [47] assume that the time elapsed before the first observation follows a given distribution and is independent of the time to the next transition from the first observation. Satten and Longini [46] develop a procedure to estimate Markov model parameters that conditions on the initiation time in order to remove dependence on this time. Kalbfleisch and Lawless [27] simply assume that the holding time of the initial state is exponentially distributed, rendering the time origin unnecessary due to the memory less property of the exponential distribution. In this dissertation, we use this strategy to simplify our model.

In this dissertation, a general approach to fitting the semi-Markov model to panel data is derived. The method, which allows for backward transition is used to model the unevenly spaced periodically observed transition data assuming no unobserved transitions. Different distributions for the holding time according to baseline state are assumed. There are two absorbing states, dementia (interval censored) and death

(competing risk to dementia and with exactly observed transition time). We expect that the incorporation of the time-dependent covariate (age, the calendar time) will lead to better parameter estimates. The advantage of this method is that it allows us to check which variables are related to the backwards transaction and the holding time of each state.

The remainder of the dissertation is structured as follows. Notation and likelihood of the semi-Markov is defined in Section 3.2. In Section 3.3, a simulation study is conducted to check the model robustness against certain violations of the distributional assumptions. Section 3.4 applies this new method to a real dataset, the Nun Study data. Conclusion and discussion are provided in Section 3.5.

## 3.2 Methodology

We first introduce the notation and establish the likelihood function of the semi-Markov process where sample paths are only periodically observed.

### 3.2.1 The Semi-Markov Process

Suppose there are $m$ subjects in the study, denoted by $i = 1, 2, 3, \cdots, m$. Let $SP = 1, 2, 3, \cdots, S$ be the finite state space representing the possible states of the evolution of a subject. For ease of exposition, the subsequent notations will be based on the Nun study data and extensions to more general semi-Markov process should be straightforward. In the Nun study, the status of a participant at each visit was recorded as being in one of the following states: 1 = intact cognition, 2 = mild cognitive impairments (M.C.I.), 3 = global impairments (G.I.), 4 = dementia,[55] and 5 = death, i.e. S=5 in this case. States 1-3 are transient while 4 and 5 are absorbing with 5 considered to be a competing risk to state 4. Transition times between the states 1, 2, 3 and 4 are not exactly known and a patient may begin his/her

evolution in any of these three transient states (Figure 3.1). The exact occurrence time of state 5 is known. The list of follow up states of a subject is denoted by $v_k$, where $k = 0, 1, 2, 3, \cdots, n$, $v_0$ is the baseline state, $n$ is the number of jumps for the subject and each $v_k$ is in SP. We assume the Markov property holds for the sequence $\boldsymbol{V} = (v_0, v_1, \cdots, v_n)$. In all subsequent exposition, we assume that $j = 1, 2, 3$, $j' = 1, 2, \cdots, 5$ and $j \neq j'$.

Let $d_k$ be the holding time in the state $v_k$, defined by $d_k = t_{k+1} - t_k$, where $t_k$ is the calendar time of entrance into state $v_k$. If $v_k$ is an absorbing state, we define $d_k = 0$. Let $\boldsymbol{Z} = (z_1, z_2, \cdots, z_p)^T$ be a vector of $p$ fixed (e.g. baseline) covariates. Let $w_k$ be age at time $t_k$; $w_0$ denotes baseline age.

The probability of one step transition from state $j$ to $j'$ at $k+1$th jump can be expressed as

$$\mathrm{P}_{kjj'} = \mathrm{P}(v_{k+1} = j' | v_k = j, \boldsymbol{Z}, w_k)$$

with the constraints $\sum_{j' \neq j} \mathrm{P}_{kjj'} = 1$ and $\mathrm{P}_{kjj'} \geq 0$.

Following Salazar et al. [45], a multinomial logit parameterization could be applied to link these transition probabilities in the following way:

$$\log\left(\frac{\mathrm{P}_{ksv}}{\mathrm{P}_{ks5}}\right) = \alpha_{sv} + \boldsymbol{Z}^\top \boldsymbol{\beta}_{1sv} + \boldsymbol{w}_k^\top \boldsymbol{\beta}_{2sv}, \qquad v = 1, 2, 3, 4; s = 1, 2, 3. \tag{3.1}$$

Here $\alpha_{sv}$ is the intercept; $\boldsymbol{\beta}_{1sv}$ and $\boldsymbol{\beta}_2 sv$ are the unknown regression coefficients. It follows that the transition probabilities are given by:

$$\mathrm{P}_{sv}(\Theta | \boldsymbol{X}, \boldsymbol{\gamma}) = \begin{cases} \frac{1}{1+\sum_{h=1}^{4} \exp(\alpha_{sh} + \boldsymbol{Z}^\top \boldsymbol{\beta}_{1sv} + \boldsymbol{w}_k^\top \boldsymbol{\beta}_{2sv})}, & \text{for v} = 5 \\ \frac{\exp(\alpha_{sv} + \boldsymbol{Z}^\top \boldsymbol{\beta}_{1sv} + \boldsymbol{w}_k^\top \boldsymbol{\beta}_{2sv})}{1+\sum_{h=1}^{4} \exp(\alpha_{sh} + \boldsymbol{Z}^\top \boldsymbol{\beta}_{1sv} + \boldsymbol{w}_k^\top \boldsymbol{\beta}_{2sv})}, & \text{for v} < 5. \end{cases} \tag{3.2}$$

We assume two types of distributions for the holding time according to the initial state, due to the left censoring of exact transition times to the initial state. Specifically,

- The distribution of the holding time for moving out of the initial state is exponential, with the hazard function

$$\lambda_{0jj'}(t) = \exp(-\boldsymbol{Z}^\top \boldsymbol{\gamma}_{1jj'} - w_0\gamma_{2jj'}). \tag{3.3}$$

- The distribution of the holding time for all other transitions is Weibull. The corresponding hazard function is given by

$$\lambda_{kjj'}(t) = \lambda_{0jj'}(t)\exp\Big(-\boldsymbol{Z}^\top \boldsymbol{\gamma}_{3jj'} - w_k\gamma_{4jj'}\Big), \tag{3.4}$$

  where $\alpha_{0jj'}(t) = k_{jj'}t^{k_{jj'}-1}$ and $k_{jj'}$ is an unknown fixed constant.

The corresponding survival function and density function are

$$\begin{aligned} S_{kjj'}(t) &= \mathrm{P}\Big(D_k \geq t | V_{k+1} = j', V_k = j, \boldsymbol{Z}, w_k\Big) = \exp\left(-\int_t^\infty \lambda_{kjj'}(s)\mathrm{d}s\right), \\ f_{kjj'}(t) &= \lambda_{kjj'}(t)\exp\left(-\int_t^\infty \lambda_{kjj'}(s)\mathrm{d}s\right). \end{aligned}$$

If the last observed state is a transient state, the holding time of that state will be right censored. Moreover, we do not even know what will be the next state, so for the last state we have:

$$\begin{aligned} S_{kj.}(t) &= \mathrm{P}\Big(D_k \geq t | V_k = j, \boldsymbol{Z}, w_k\Big) \\ &= \sum_{j'\neq j} \mathrm{P}\Big(V_{k+1} = j' | V_k = j, \boldsymbol{Z}, w_k\Big)\mathrm{P}\Big(D_k \geq t | V_{k+1} = j', V_k = j, \boldsymbol{Z}, w_k\Big) \\ &= \sum_{j'\neq j} \mathrm{P}_{kjj'}(t) S_{kjj'}(t). \end{aligned}$$

### 3.2.2 The Likelihood Function

Let $\mathbf{T} = (t_0, t_1, \cdots, t_n)$ be the vector of transition instants and $\mathbf{D} = (d_0, d_1, \cdots, d_n)$ be the vector of holding times. Let $u$ be the time of the last assessment. Note that if $v_n \in 1, 2, 3$, $d_n$ is right censored in the sense that it's only known up to $d_n \geq u - t_n$; if $v_n \in \{4, 5\}$, $d_n$ will not enter our likelihood function since we are not interested in the holding time of dementia or death. Then, conditional on the initial state, the contribution to the likelihood from the subject is

$$\ell(\boldsymbol{\theta}|Z, W, T, V) = \prod_{k=1}^{n} \mathrm{P}_{k-1,v_{k-1},v_k} \prod_{k=1}^{n} f_{k-1,v_{k-1},v_k}(t_k - t_{k-1}) \times S_{nv_n\cdot}(u - t_n)^{I[v_n \leq 3]}. \quad (3.5)$$

Here $\boldsymbol{\theta}$ represents the set of all the unknown parameters, $\mathbf{W} = (w_0, w_1, \cdots, w_n)$ and $I[\cdot]$ denotes the indicator function.

Due to the fact that a subject is only periodically assessed, we do not fully observe $\mathbf{T}$ or $\mathbf{D}$ and therefore the specification of $\ell$ in (3.5) needs some modifications. What we observe instead, except for the state at each assessment, is a sequence of lower bounds $\mathbf{L} = (0, l_1, \cdots, l_n)$ and upper bounds $\mathbf{U} = (u_0, u_1, \cdots, u_n)$ for any $t_k \in \mathbf{T}$ such that $l_k \leq t_k \leq u_k$, $k = 0, 1, \cdots, n$. As mentioned in subsection 2.1, we assume that the holding time of the initial state is exponentially distributed. Due to the memoryless property, we could simply treat the time of transition to the initial state as the time of the first assessment, namely the baseline age, or mathematically, $t_0 = l_0 = u_0$. To obtain the correct likelihood contribution, basically we could integrate out $\mathbf{T}$ in (3.5), where $\mathbf{T}$ falls in the domain implied by $\mathbf{U}$ and $\mathbf{L}$. Specifically, we propose to modify the likelihood as follows:

$$\begin{aligned} \ell(\theta|\mathbf{Z}, \mathbf{W}, \mathbf{U}, \mathbf{L}, \mathbf{V}) \;=\; & \int_A \prod_{k=1}^{n} \mathrm{P}_{k-1,v_{k-1},v_k} \prod_{k=1}^{n} f_{k-1,v_{k-1},v_k}(t_k - t_{k-1}), \\ & \times S_{nv_n\cdot}(u - t_n)^{I[v_n \leq 3]} \prod_{i=1}^{n} \mathrm{d}t_i \end{aligned} \quad (3.6)$$

where $u$ is the last assessment time if the subject was right censored. The integration domain $\mathbf{A}$ will incorporate the lower and upper bounds of $\mathbf{T}$ to reflect the fact that we only have partial information on $\mathbf{T}$. We have

$$A = \Big\{(t_1, t_2, \cdots, t_n)\Big| l_k \leq t_k \leq u_k, k = 1, 2, \cdots, n\Big\}.$$

When the last observation is a death, $t_n$ is exactly observed, and therefore the integration in (3.6) along the axis $t_n$ is with respect to the probability measure that puts unit mass on $t = t_n$. The integration in (3.6) can be lengthy, but the idea is straightforward.

One implication of the modeling assumption in (3.1) is that the transition probability $\mathrm{P}_{ksv}$ is conditioned on the value of the time-dependent covariate $w_k$ at the time of $t_k$. Therefore, the associated interpretation of the regression coefficient $\beta_{2sv}$ is conditioned on the unobserved random variable $t_k$. In order for the regression coefficient to have an interpretation that only depends on what we can actually observe, we replace $w_k$ in (3.1) by its value at the upper bound $u_k$. This results in our ability to predict the next state of a subject given his/her information at the current assessment. Moreover, under this modification, $\mathrm{P}_{ksv}$ does not depend on $t_k$, and therefore it allows us to pull $\mathrm{P}_{ksv}$ outside of the integration in (3.6) and thus significantly reduces the computational burden.

### 3.2.3 Parameter Estimation

The multi-dimensional integration in (3.6) could be approximated by numerical methods, including importance sampling, quasi-Monte Carlo (QMC) approximation [7] and so on. In this dissertation, we use the QMC method due to the fact that the highest order of integration is eight, which is relatively high. Estimation and inference on the parameters $\theta$ can be achieved by maximizing the likelihood function in Equation 3.6, where the optimization procedure could be implemented for example

by PROC NLMIXED in SAS. The likelihood function takes a complicated form and is not convex in the parameters, therefore convergence of the optimization algorithm is not guaranteed for an arbitrary set of initial values. It is advisable to start with multiple sets of initial values and select the maximizers accordingly.

### 3.3 Simulation Studies

In this section for simplicity we only consider the effects of baseline age and age as motivated by the Nun Study example. The purpose of the simulations is to determine how well the averaged odds ratios and hazard ratios for age in Tables $3.6 - 3.8$ will be estimated when the model assumed in Section 3.2.1 is correct and then when the assumption made on the distribution of the holding time for the initial state is violated. With respect to the latter a Generalized Weibull distribution $WG(r, \mu, \theta)$, with the hazard function $h(t) = \frac{r\mu}{\theta}(1 + \mu t^r)^{\frac{1}{\theta-1}} t^{(\gamma-1)}$, where $t > 0$, $r > 0$, $\mu > 0$ and $\theta > 0$, is used to check the robustness of the maximum likelihood estimate (MLE) to the violation of the holding time assumption. If we fix $\theta$ at 1, we obtain the Weibull formulation. We set $\theta$ and $r$ to be constants and $\log(\mu)$ to be a linear function of age. Different options of $\theta$ and $r$ with 1000 simulations was tested but the following tables only show the result for $\theta = 2$ and $r = 2$. Simulations were carried out using Intel i5-650 professor (4M Cache, 3.20 GHz). The computational time for 1000 simulations of sample size 300 and 1000 with 500 Halton numbers are 20.43 hours and 50.61 hours respectively.

Similar results are obtained when the assumption of Weibull distribution (versus a Generalized Weibull) is violated for the holding time in the non-initial states (results not shown).

Specific steps in the simulation process follow. For each subject:

1. Generate initial age $w_0$ using a truncated normal distribution that has the same range, mean, and standard deviation as age in the real dataset and generate the initial state $v_0$ using the probabilities 140/511, 272/511, and 99/511 for initial states 1, 2, and 3, respectively.

2. Then in the order $i = 0, \cdots, 9$ generate the next state $v_{i+1}$ given $v_i$ and $w_i$ according to the transition probabilities in Equation 3.2. Since the assessment times $b_1, \cdots, b_{10}$ are predetermined, note the values of $OS_i$ which denote the observed state of the process at $b_i$ for $i = 1, \cdots, 10$.

3. Generate holding time $d_i$ at state $v_i$ according to exponential ($i = 0$) or Weibull distribution ($i > 0$).

Repeat Steps 3 and 4 until either death, or dementia, or the summation of all holding times exceeds the largest planned observation time $b_{10}$.

Choice of the model parameters were made to come as close to those estimated from the real dataset of the next section without producing simulations that lead to non-estimable parameters (i.e. the likelihood function fails to converge). In the real dataset, the corresponding probabilities for the initial state are 140/461, 272/461, and 49/461 for states 1, 2, and 3. These were changed slightly in Step 1 to avoid convergence problems on too many simulations when using a smaller sample size (i.e. the simulated path of the process yields a likelihood that does not converge due to few transitions into some of the states). The selection of the regression coefficients required less trial and error. For Table 3.3 a hazard ratio of 0.905 was selected which is close to the average of the hazard ratios in Table 3.6 after log transformation (average -0.11 versus -0.10). For Tables 3.3 and 3.3 an odds ratio of 1.051 was selected which is to the average odds ratios in Tables 3.7 and 3.8 (after log transformation and after including the non-significant coefficients).

| | | Sample Size 300 | | | | | | Sample Size 1000 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Model | | Weibull | | G Weibull | | Model | | Weibull | | G Weibull | |
| | Odds Ratio (OR) | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ |
| 1→2 | 0.905 | 0.0298 | 1.31 | 0.0272 | 1.15 | 0.0285 | 1.15 | 0.0276 | 0.92 | 0.0266 | 0.90 | 0.0258 | 0.88 |
| 1→3 | 0.905 | 0.0295 | 1.24 | 0.0296 | 1.29 | 0.0298 | 1.27 | 0.0276 | 0.95 | 0.0275 | 0.95 | 0.0268 | 0.93 |
| 2→1 | 0.905 | 0.0290 | 1.18 | 0.0285 | 1.19 | 0.0294 | 1.12 | 0.0303 | 1.05 | 0.0306 | 1.11 | 0.0291 | 0.99 |
| 2→3 | 0.905 | 0.0267 | 1.06 | 0.0268 | 1.11 | 0.0269 | 1.11 | 0.0253 | 0.81 | 0.0253 | 0.82 | 0.0250 | 0.81 |
| 2→4 | 0.905 | 0.0439 | 3.03 | 0.0403 | 2.72 | 0.0409 | 2.65 | 0.0356 | 2.00 | 0.0310 | 1.88 | 0.0340 | 1.98 |
| 3→1 | 0.905 | 0.0242 | 0.92 | 0.0242 | 0.94 | 0.0260 | 0.95 | 0.0252 | 0.76 | 0.0244 | 0.74 | 0.0240 | 0.71 |
| 3→2 | 0.905 | 0.0239 | 0.96 | 0.0226 | 0.87 | 0.0247 | 0.95 | 0.0220 | 0.63 | 0.0213 | 0.64 | 0.0211 | 0.65 |
| 3→4 | 0.905 | 0.0339 | 2.09 | 0.0338 | 2.25 | 0.0357 | 2.17 | 0.0272 | 1.41 | 0.0225 | 1.37 | 0.0268 | 1.44 |

Model: the holding time satisfied our model assumption;

Weibull: the holding time for the baseline state follows Weibull distribution;

G Weibull: the holding time for the baseline state follows Generalized Weibull Distribution;

Table 3.1: Bias and mean square error (MSE) of the estimate of the age odds ratio, $\exp(\beta_{\text{age}})$, in Equation 3.2 for sample sizes 300 and 1000

The bias and mean square error (MSE) of age odds ratios in Equation 3.2 for sample sizes 300 and 1000 are shown in Table 3.3. From this table, we can see the effect of age on all forward transitions and the backward transitions can be well estimated, with the maximum bias 0.0439 (4.9%). Biases and MSEs stay the same when the initial holding time assumption is violated. MSE decreases as the sample size increases but biases stay almost the same, with the maximum difference 0.0113.

| | | Sample Size 300 | | | | | | Sample Size 1000 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Model | | Weibull | | G Weibull | | Model | | Weibull | | G Weibull | |
| | Hazard Ratio (HR) | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ |
| 1→2 | 1.051 | -0.0133 | 0.50 | -0.0127 | 0.54 | -0.0124 | 0.59 | -0.0105 | 0.23 | -0.009 | 0.21 | -0.0106 | 0.25 |
| 1→3 | 1.051 | -0.0135 | 0.58 | -0.0123 | 0.49 | -0.0135 | 0.46 | -0.0116 | 0.25 | -0.0104 | 0.23 | -0.0103 | 0.22 |
| 1→5 | 1.051 | -0.0166 | 0.47 | -0.0145 | 0.46 | -0.0153 | 0.45 | -0.0145 | 0.27 | -0.0138 | 0.28 | -0.0142 | 0.28 |
| 2→1 | 1.051 | -0.0216 | 0.98 | -0.0225 | 0.94 | -0.0217 | 0.93 | -0.0201 | 0.52 | -0.018 | 0.47 | -0.0191 | 0.50 |
| 2→3 | 1.051 | -0.0217 | 0.86 | -0.0215 | 0.81 | -0.0228 | 0.87 | -0.0206 | 0.55 | -0.0204 | 0.56 | -0.0221 | 0.61 |
| 2→4 | 1.051 | -0.0201 | 0.75 | -0.0203 | 0.79 | -0.0212 | 0.72 | -0.0202 | 0.52 | -0.0189 | 0.47 | -0.0195 | 0.50 |
| 2→5 | 1.051 | -0.0193 | 0.67 | -0.0181 | 0.64 | -0.0196 | 0.60 | -0.0186 | 0.44 | -0.0173 | 0.4 | -0.018 | 0.41 |
| 3→1 | 1.051 | 0.00290 | 0.17 | 0.00250 | 0.16 | 0.0034 | 0.15 | 0.0007 | 0.07 | -0.0005 | 0.12 | -0.0004 | 0.10 |
| 3→2 | 1.051 | -0.0211 | 0.86 | -0.0188 | 0.84 | -0.0197 | 0.75 | -0.0177 | 0.43 | -0.018 | 0.47 | -0.0175 | 0.40 |
| 3→4 | 1.051 | -0.0194 | 0.63 | -0.0183 | 0.72 | -0.0168 | 0.63 | -0.0175 | 0.4 | -0.0162 | 0.37 | -0.016 | 0.37 |

Table 3.2: Bias and MSE of the estimate of the exponential hazard ratio, $\exp(\gamma_{\text{age}})$, in Equation 3.3 for sample sizes 300 and 1000

Table 3.3 lists the bias and MSE of estimated exponential hazard ratios in Equation 3.3 under different sample sizes. In this table, most of the biases are negative and small. The changes in MSEs and biases are very small when the exponential assumption on the holding time for the initial state is violated.

| | | Sample Size 300 | | | | | | Sample Size 1000 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Model | | Weibull | | G Weibull | | Model | | Weibull | | G Weibull | |
| | Hazard Ratio (HR) | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ | bias | mse*$10^3$ |
| 1→2 | 1.051 | -0.0436 | 2.66 | -0.0533 | 3.67 | -0.0504 | 2.95 | -0.0434 | 2.15 | -0.05 | 2.95 | -0.0486 | 2.58 |
| 1→3 | 1.051 | -0.0475 | 2.78 | -0.0519 | 3.66 | -0.0488 | 2.78 | -0.0445 | 2.20 | -0.0491 | 3.06 | -0.0469 | 2.41 |
| 1→5 | 1.051 | -0.0401 | 2.90 | -0.0598 | 5.51 | -0.0446 | 2.66 | -0.0415 | 2.05 | -0.0577 | 4.01 | -0.0444 | 2.21 |
| 2→1 | 1.051 | -0.0323 | 1.33 | -0.0393 | 2.14 | -0.04 | 1.83 | -0.0302 | 1.11 | -0.0336 | 1.42 | -0.0359 | 1.47 |
| 2→3 | 1.051 | -0.0329 | 1.48 | -0.0369 | 1.97 | -0.0394 | 1.78 | -0.0302 | 1.11 | -0.0326 | 1.48 | -0.0358 | 1.45 |
| 2→4 | 1.051 | -0.0325 | 1.36 | -0.0373 | 2.12 | -0.0378 | 1.69 | -0.0292 | 1.05 | -0.0344 | 1.69 | -0.0333 | 1.35 |
| 2→5 | 1.051 | -0.0269 | 1.04 | -0.0319 | 1.65 | -0.0329 | 1.42 | -0.0261 | 0.80 | -0.0296 | 1.09 | -0.0319 | 1.13 |
| 3→1 | 1.051 | -0.0283 | 1.08 | -0.0336 | 1.35 | -0.0361 | 1.47 | -0.0286 | 0.90 | -0.0316 | 1.12 | -0.034 | 1.25 |
| 3→2 | 1.051 | -0.0272 | 1.03 | -0.0322 | 1.34 | -0.0358 | 1.46 | -0.0285 | 0.89 | -0.0299 | 1.07 | -0.0345 | 1.27 |
| 3→4 | 1.051 | -0.0262 | 0.89 | -0.0302 | 1.14 | -0.0345 | 1.36 | -0.0267 | 0.78 | -0.029 | 0.96 | -0.0329 | 1.17 |
| 3→5 | 1.051 | -0.0234 | 0.77 | -0.0232 | 0.83 | -0.0304 | 1.11 | -0.023 | 0.61 | -0.0254 | 0.73 | -0.0301 | 0.96 |

Table 3.3: Bias and MSE of the estimate of the Weibull hazard ratio, $\exp(\gamma_{\text{age}})$, in Equation 3.4 for sample size 300, 1000

Table 3.3 presents the bias and MSE of the estimated Weibull hazard ratios in Equation 3.4 under different sample sizes. Most of the biases are negative in this table, indicating that our proposed estimation method will slightly underestimate the effect of age. It is also clear that violations to the distributional assumptions on the holding time for the initial state lead to moderately worse biases and MSEs. Additionally, increasing sample size decreases the bias and MSEs. For examples, the bias and MSE are -0.0234 and 0.77 for model assumption with sample size 300 and 0.0230 and 0.61 with sample size 1000.

A clear pattern that is perceivable from these three tables, especially Table 3.3 and 3.3, is that the bias does not shrink to zero as the sample size increases from 300 to 1000, which suggests that our estimation method may yield slightly biased estimates. The systematic bias may have two sources. The first one is due to our data generating mechanism. Specifically, we use exponential, Weibull or generalized Weibull distribution, to generate the holding time and naturally we will occasionally encounter a transition with very short holding times, which will result in missing transitions under intermittent observation scheme. The second one relates to our treatment of $\mathrm{P}_{ksv}$ in (3.6) where we approximate the value of $\mathrm{P}_{ksv}$ at $w_k$ by its value at the upper bound $u_k$. While this approximation greatly facilitates our computation it affects our ability to precisely estimate the parameters.

These simulation studies indicate that estimation for the effect of an important covariate in Equations 3.2 and 3.3 is robust against a violation of the exponential assumption on the holding time for the initial state provided the sample size is large enough to assure adequate observations on all transitions. This is not true for estimation in Equation 3.4 where the lack of robustness is likely due to the effect of interval censoring on both ends of the estimation interval for the hazard function.

## 3.4 Application to the Nun Study

The Nun Study began enrolment in 1991. The cohort consists of 672 members of the School Sisters of Notre Dame born before 1917 and living in retirement communities in the midwestern, eastern, and southern United States. The participants were recruited in phases and received annual cognitive assessments with brain donation at death. Analyses were based on data from up to ten unevenly spaced examinations, with time spans between two assessment ranging from 0.421 to 3.911 and mean 1.441, made in fifteen-year period. The status of a participant at each visit was recorded as being one of the states: intact cognition, mild cognitive impairments (M.C.I.), global impairments (G.I.), or dementia [51]. A total of 211 subjects were excluded from the study due to: missing examinations, presence of dementia at baseline visit or missing APOE4 data. The final analytic sample consisted of 461 participants, of which 74 survived without dementia, 162 developed dementia and 225 died before converting to dementia. Among those final participants, 158 of them missed one examination and 7 of them had more than one missing examinations. The variables of interest include presence or absence of the APOE-4 allele (APOE4), education (no college, college and graduate education (reference)), and age. The transitions among the cognitive states are summarized in Figure 3.1.

### 3.4.1 Examples of Nun's Cognitive Paths

To better understand the data, we classified the trajectory of each Nun using three criteria. (i) Initial state: 1, 2, or 3. (ii) Final state: 1, 2, 3, 4, or 5 and (iii) Path type: non-terminal, right shift, or reversal. Non-terminal means final observed state is transient (i.e. 1, 2 or 3). Right shift means the final state is 4 or 5 and no back transition occurred. Reversal means the final state is 4 or 5 and at least one back transition occurred. The frequency of non-terminal, reversal and right shift paths, are 74 (16.1%), 131 (28.4%), and 256 (55.5%), respectively implying back transitions

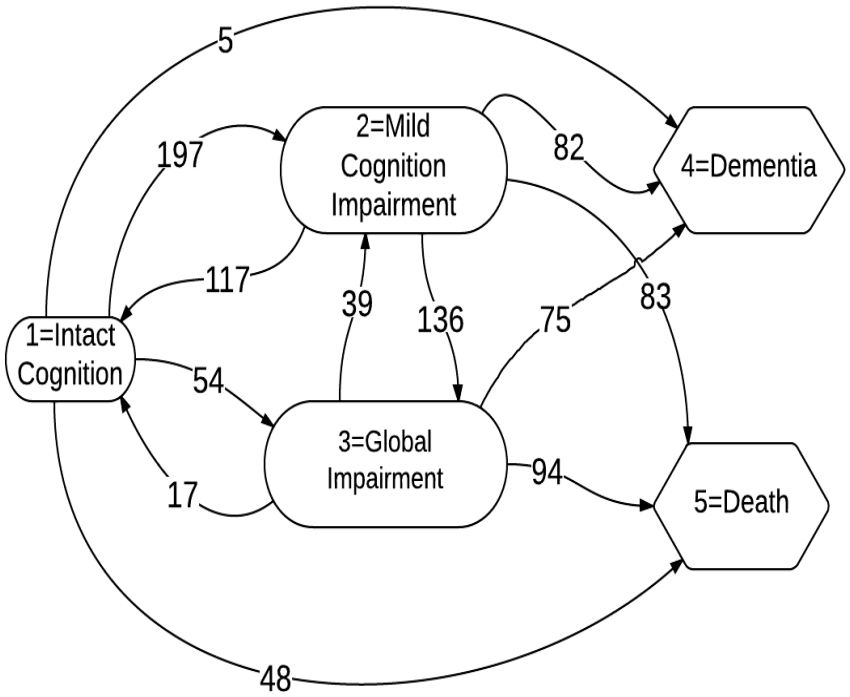


Figure 3.1: Frequency of the one-step transitions.

in these data frequently occurred. Table 3.4 shows some examples of these cognitive paths. The two most frequently observed paths are $2 \rightarrow 4$ and $2 \rightarrow 5$ with 100 out of 461 (21.7%) nuns having those trajectories. The total number of distinct paths observed in the Nun dataset is 84 but 32 of those have a frequency one (not shown in the table).

### 3.4.2 Risk Factors

The main purpose of this subsection is to identify the risk factors associated with the probability and the holding time of each transition.

Table 3.5 shows the frequency table for the integration orders of the likelihood 3.6. From the table, we can see the highest order of integration is 8 and the percentage of integration order higher than 3 is 14.31%, which makes the use of the traditional

| Visit Pattern | Freq | Initial State | Final State | Path Type |
|---|---|---|---|---|
| 1 | 11 | 1 | 1 | Non-terminal |
| 2→4 | 61 | 2 | 4 | Right Shift |
| 2→5 | 39 | 2 | 5 | Right Shift |
| 3→4 | 23 | 3 | 4 | Right Shift |
| 2→3→5 | 32 | 2 | 5 | Right Shift |
| 2→3→4 | 30 | 2 | 4 | Right Shift |
| 2→1→2→5 | 14 | 2 | 5 | Reversal |
| 3→2→3→5 | 3 | 3 | 5 | Reversal |
| 1→2→1→3→5 | 4 | 1 | 5 | Reversal |
| 1→3→1→3→5 | 2 | 1 | 5 | Reversal |
| 2→1→2→1→2 | 3 | 2 | 2 | Non-terminal |
| 2→3→2→3→4 | 5 | 2 | 4 | Reversal |

Table 3.4: Examples of Nun's cognitive path

GAUSS method difficult. Therefore, we implemented a quasi-random Monte Carlo (QMC) method [7] which will provide considerably better accuracy with much fewer draws and less computational time to estimate the likelihood function. In this dissertation, we choose 1000 draws with the average computational time about 30 minutes. The parameters associated with transition $1 \rightarrow 4$ are eliminated from our model since there are only 5 such transitions (Figure 3.1). Therefore, we have 942 transitions in the final analytic data which is a moderate number compared to the 145 potential parameters without interactions in our full model. Backward elimination with significance level to stay 0.05 was used to identify the covariates in the final reported model and only 57 remained after backward selection.

The odds ratios and 95% confidence intervals for the significant covariates affecting each transition probability are provided in Table 3.6 (base state: 5=death). The

| Order of integration | Count | Relative frequency | Cumulative Relative frequency |
|---|---|---|---|
| 0 | 63 | 13.67 | 13.67 |
| 1 | 177 | 38.39 | 52.06 |
| 2 | 108 | 23.43 | 75.49 |
| 3 | 47 | 10.20 | 85.69 |
| 4 | 36 | 7.83 | 93.50 |
| 5 | 13 | 2.82 | 96.32 |
| 6 | 8 | 1.74 | 98.06 |
| 7 | 7 | 1.52 | 99.58 |
| 8 | 2 | 0.43 | 100 |

Table 3.5: Frequency table of order of integration of the likelihood

reversal path is more likely in younger Nuns. For example, the odds ratios for the three possible backward transitions ($2 \to 1$, $3 \to 1$ and $3 \to 2$) with one year increases in age are 0.927, 0.804 and 0.900 respectively, as opposed to death. The effect of age for all the forward transitions is not significant as opposed to death except $3 \to 4$. Concerning the effect of APOE4 or education, the results show that the presence of APOE4 and no college education decreases the odds of the backward transition from mild impairments to intact cognition with the corresponding odds ratios being much less than 1, but the presence of APOE4 and college education increases the odds of a forward transition. The result is consistent with the historical result that the presence of APOE4 and college education promotes the probability of a right shift compared to reversal. Presence of APOE4 will promote the Nuns to dementia if the prior state is global impairment with OR=2.623 (p-value=0.0021).

The effects of covariates on the duration time for the initial state that follows an exponential distribution are also tested (see Table 3.7). Baseline age, APOE4 and education all have no significant influence on the holding time of transitions out of intact cognition or the global impairment. Baseline age increases the hazard ratio of

| Transition | Covariate | Odds Ratio | 95% Low | 95% Upper | P-value |
|---|---|---|---|---|---|
| 2→1 | Apoe4 | 0.363 | 0.1842 | 0.7137 | 0.0034 |
| 2→3 | Apoe4 | 1.726 | 1.0313 | 2.889 | 0.0379 |
| 3→4 | Apoe4 | 2.623 | 1.4232 | 4.8337 | 0.0021 |
| 2→1 | No College | 0.249 | 0.1051 | 0.5895 | 0.0016 |
| 1→2 | College | 1.661 | 1.0174 | 2.7118 | 0.0425 |
| 2→3 | College | 1.734 | 1.2459 | 2.4112 | 0.0011 |
| 2→1 | Age | 0.927 | 0.892 | 0.9616 | <.0001 |
| 3→1 | Age | 0.804 | 0.7653 | 0.8446 | <.0001 |
| 3→2 | Age | 0.900 | 0.8763 | 0.9231 | <.0001 |
| 3→4 | Age | 0.975 | 0.9573 | 0.9918 | 0.004 |

States: 1=Intact Cognition, 2=Mild Cognitive Impairment, 3=Global Impairment, 4=Dementia;

Table 3.6: The odds ratio and confidence interval for significant effects on each transition probability (base state: 5=Death)

transitions moving out of mild impairments to states 1, 3, and 4 with a hazard ratio 1.091, 1.101, and 1.063 respectively. Presence of APOE4 and no college education significantly promote the transition $2 \to 3$, by shorter the holding time with hazard ratio 2.386 and 3.191 respectively.

| Transition | Covariate | Hazard Ratio | 95% Low | 95% Upper | P-value |
|---|---|---|---|---|---|
| 2→3 | Apoe4 | 2.386 | 1.369 | 4.159 | 0.0022 |
| 2→3 | No College | 3.191 | 1.496 | 6.809 | 0.0028 |
| 2→1 | Baseline Age | 1.091 | 1.017 | 1.17 | 0.0158 |
| 2→3 | Baseline Age | 1.101 | 1.049 | 1.154 | 0.0001 |
| 2→4 | Baseline Age | 1.063 | 1.010 | 1.119 | 0.0213 |

Hazard ratio for baseline age is the hazard ratio for a one year increase in age.

Table 3.7: The hazard ratio and confidence interval for significant covariates in the exponential distribution

Table 3.8 lists the hazard ratio and 95% confidence interval estimate for significant effects in the non-initial transitions assuming Weibull distribution. Increasing in age increases the hazard ratio for almost all the transitions to the three transient states and death, except dementia. In other words, as the Nun gets older, the holding time at each state will be shorter on average, which makes the homogeneous semi-Markov model inappropriate. There are no differences on the holding time for transitions 2 to 3 and 2 to 5 when the Nun's gets one year older. The holding time will be shorter for transition from 1 to 5 than to 3 with the corresponding hazard ratio 1.349 versus 1.243. APOE promotes forward transitions by shorter the holding time from 1 to 2 and 1 to 3 with the corresponding hazard ratios 2.569 and 16.856 respectively. Also if nun is in state 3, APOE keeps a nun from being demented by longer the holding time. No college education has an influence on the transitions to dementia with hazard ratio 9.575 and 0.290 for prior state 2 and 3, respectively. Some of the hazard ratios are much larger than the majority of the hazard ratios partially due to the rare observations we have. For example, we only have 3 observations in the Nun Study data for transition from 1 to 3 with the presence of APOE4.

The estimates, standard deviation, and p-values of the shape parameters $k_{jj'}$ of the Weibull distributions are summarized in Table 3.9. The highly significance of these parameters justifies the use of Weibull distributions over exponential distributions.

## 3.5 Discussion and Conclusion

In this dissertation we implemented a quasi-Monte Carlo (QMC) method to evaluate the likelihood function in a semi-Markov process with interval censored observations and backward transitions. To the best of our knowledge few researchers consider the case of semi-Markov processes with backward transitions in the presence of interval censored data. We showed that use of the QMC makes the computation of the likelihood function possible provided we assume that the time interval from the

| Transition | Covariate | Hazard Ratio | 95% Low | 95% Upper | P-value |
|---|---|---|---|---|---|
| 1→2 | Apoe4 | 2.569 | 1.168 | 5.65 | 0.019 |
| 1→3 | Apoe4 | 16.856 | 2.072 | 137.14 | 0.008 |
| 3→4 | Apoe4 | 0.286 | 0.13 | 0.634 | 0.002 |
| 2→4 | No College | 9.575 | 1.315 | 69.735 | 0.026 |
| 3→4 | No College | 0.29 | 0.09 | 0.934 | 0.038 |
| 1→2 | Age | 1.06 | 1.012 | 1.11 | 0.014 |
| 1→3 | Age | 1.243 | 1.114 | 1.386 | ¡.001 |
| 1→5 | Age | 1.349 | 1.187 | 1.534 | ¡.001 |
| 2→1 | Age | 1.064 | 1.007 | 1.125 | 0.028 |
| 2→3 | Age | 1.09 | 1.027 | 1.158 | 0.005 |
| 2→5 | Age | 1.092 | 1.028 | 1.16 | 0.005 |
| 3→5 | Age | 1.055 | 1.013 | 1.097 | 0.01 |

Hazard ratio in age is the hazard ratio for one year increase in age.

Table 3.8: The hazard ratio and confidence interval estimate for significant effects in the Weibull distribution

initial state to the first transition is exponentially distributed and that no additional transitions occur between successive observations of the process.

Application of our method to the Nun Study data showed that older age diminishes the chances that any back transition occurs while less than a college education and presence of an APOE 4 allele diminishes the chance of a back transition to the normal cognitive state from the mild cognitive impairments state. Further, if the latter transition does occur the time interval associated with this transition is significantly abbreviated by older age. The reason additional factors are not significant for back transitions likely have to do with the small frequency of some of these transitions as shown in Figure 3.1. The use of a semi-Markov process in this application is motivated by up to ten serial assessments (approximately every 15 months apart) over

| Transition | Coeff. | Std. Dev. | p-value |
|---|---|---|---|
| 1→2 | 1.729 | 0.147 | <.0001 |
| 1→3 | 2.145 | 0.360 | <.0001 |
| 1→6 | 1.757 | 0.255 | 0.0001 |
| 2→1 | 1.863 | 0.173 | <.0001 |
| 2→3 | 1.752 | 0.211 | <.0001 |
| 2→4 | 2.384 | 0.454 | <.0001 |
| 2→6 | 1.595 | 0.194 | 0.0001 |
| 3→2 | 1.852 | 0.348 | 0.0011 |
| 3→4 | 2.240 | 0.320 | <.0001 |
| 3→6 | 1.352 | 0.122 | 0.0009 |

Table 3.9: The p-values of the significant shape parameters in the Weibull distribution

a fifteen-year period of the cognitive status of each participant in the study. It is possible but unlikely that the cognitive status of each nun fluctuated much in the interval between cognitive assessments meeting the assumptions of our model.

Simulation studies determined how the parameters will be estimated when the assumption made on the holding time is violated. The simulation result shows that the maximum likelihood estimates in Equations 3.2 and 3.3 are not sensitive to the violation of the assumption on the holding time for the initial state. But it is sensitive to the sample size due to the chance of observing few transitions. However, the change of observing few transitions will be very rare when the sample is larger than 500. Simulation results also show there is a persistent bias in Table 1-3. This is likely due to the replacement of $w_k$ with $u_k$ in Equation 3.6 for $P_{ksv}$. We recalculated the MLEs for the Nun study by making $w_k$ a function of $t^*$ in Equation 3.6; the resulting MLEs were no different than reported here.

Semi-Markov model has a wide application to be more accurately describing the process of interest. However, a general problem of panel data is lack of sufficient information for the progress, such as interval censoring data or some of the important transitions between two assessments are missing. Hence, despite the advantage of the semi-Markov process, the applications to the semi-Markov are limited as compared to Markov process.

# Chapter 4 Cause-Specific Hazard Estimation for the Interval Censored Competing Risk Data

Cause-specific hazard is an important function in competing risk studies. It describes the hazard associated with a certain cause after accounting for other competing causes. In this dissertation, a local polynomial function is used to approximate the log of the cause-specific hazard function when the data is subject to interval censoring. The so-called "local EM algorithm" is used to find the maximum likelihood estimator. The corresponding variance and confidence interval are obtained through a bootstrap calibration. The methodology is justified by a simulation study and illustrated by an application to the Nun study, a longitudinal study of late life cognition in a cohort of 461 subjects.

## 4.1 Introduction

Competing risks are common in medical research, in which a subject is often at risk of multiple events and the occurrence of one event prevents the happening of the others. Moreover, the exact time of some events are missing or censored due to periodic assessment, missing clinic visit or drop out from the study. One example is the Nun Study [51]. The follow-up ends when either death or dementia occurs. The time to dementia is interval censored, while the time to death without a dementia is exactly known.

Both the cumulative incidence function (CIF) and the cause-specific hazard (CSH) function are the two functions of particular interest. Many papers in the competing risk literature focus on the estimation of the CIF associated with a particular failure type. The CIF is a summary statistic used more routinely than CSH. A standard

approach of estimating and making inference on the CIF is, however, based on the CSH [31, 34]. For example, Benichou and Gail [3] provided inference procedures for CIF assuming the CSH follows a proportional hazards model with an unknown constant or piecewise constant baseline hazard function. Cheng and Fin [8] showed how to construct confidence intervals and bands for CIF under the Cox Proportional Hazards model assumption with unspecified baseline hazard. Both works rely on the Cox proportional hazards assumption, whereas neither provides any tools to check the validity of the assumption. In recent years, some new approaches that directly estimate the CIF without an assumption on the CSH were proposed, such as the Pseudo-value approach from jackknife statistics [33, 32] and the direct binomial modeling approach [48].

Since, the CIF of a particular failure accounts for CSHs of all failures, the effect of a covariate to the CSH is different from that to the CIF. Most previous works on estimation of the CSH assume right-censored data, with or without covariates. In this dissertation, it is of interest to study the CSH in the interval-censoring framework. In the case of no covariate adjustment, we propose a fully non-parametric method to estimate CSHs with interval censored competing risk data. The proposed method has the ability to incorporate covariates easily through, for example, a Proportional hazards model. The proposed methodology is motivated by Betensky et al. [4], who used a modified Expectation-Maximize (EM) algorithm based on the local likelihood, the so-called "local EM algorithm", to obtain a smooth estimate of the local hazard function for participants subject to only one type of failure with interval censored event times. The advantage of the proposed non-parametric hazard estimate is that it can best describe the data and provide us some new structures, which could not be found by parametric methods. Such a model-free approach is data driven and particularly useful for parametric model assumption checking, such as the proportional

hazard function assumption between treatment and control groups.

This dissertation is organized as follows. In Section 4.2, we introduced the notations and propose the modified EM algorithm. In Section 4.3, we discuss simulation studies. Application of the proposed method to Nun's data is applied in Section 4.4 and conclusion and discussion are presented in Section 4.5.

## 4.2 Local Log-Likelihood

Here, we will give a very brief introduction to the local EM estimation of the hazard function for interval censored data proposed by Betensky et al. [4] in Section 4.2.1. The modified local EM estimation of CSH for interval censored competing risk data is introduced in Section 4.2.2.

For ease of presentation, notations introduced in this dissertation will be tailored to the Nun Study data, which features two competing risks, dementia or death without a dementia. The extension to more general multivariate competing risk data should be straightforward. Let $T_{i1}$ be the failure time of dementia and $T_{i2}$ be the failure time of death for the $i$th subject, $i = 1, 2, \cdots, n$. In the presence of competing risk, we only observe the first failure time, that is $T_i = \min(T_{i1}, T_{i2})$. For right censored data, the observed competing risk time $X_i = \min(T_i, C_i) = \min(T_{i1}, T_{i2}, C_i)$, where $X_i$ and $C_i$ are the failure time and censoring time, respectively. Let $\varepsilon_i$ be an indicator function valued at 1 if $X_i = T_{i1}$, 2 if $X_i = T_{i2}$ and 0 otherwise. Let $(L_i, R_i)$ be the interval of $X_i$ if the failure time is interval censored. In the Nun study, time to dementia is interval or right censored and the time to death is exactly observed or right censored. If the event time is exactly known, we have $L_i = R_i$ and if $R_i = \infty$, then the failure time is said to be right censored.

### 4.2.1 Local EM Estimation of the Hazard Function for Interval Censored Data

Here, we will give a very brief introduction to the local EM estimation of the hazard function for interval-censored data. In this case, we only have one event. Therefore, $\varepsilon_i$ equals 1 if the failure time is interval censored and 0 otherwise.

The hazard function for the failure time at t is defined as

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{\Pr(t \leq T \leq t + \Delta t | T \geq t)}{\Delta t}.$$

A local polynomial approximation of order p of the log-hazard function at the neighborhood of point t is given by

$$\log\Big\{\lambda(s|t)\Big\} \approx \alpha_{t0} + \alpha_{t1}(s-t) + \cdots + \alpha_{tp}(s-t)^p, \quad \text{for } |s-t| \leq g(t), \tag{4.1}$$

where $g(t)$ is the bandwidth and $\boldsymbol{\alpha}_t = (\alpha_{t0}, \alpha_{t1}, \cdots, \alpha_{tp})$.

The contribution of a subject with failure time $s$ to the log likelihood at time point $t$ is

$$\ell\Big(s, t\Big|L, R\Big) = \int_L^R \log\lambda(s|t) K\left(\frac{s-t}{g}\right) \mathrm{d}s - \int_L^R \int_0^s \lambda(u|t) K\left(\frac{u-t}{g}\right) \mathrm{d}u\mathrm{d}s \tag{4.2}$$

where $K(\cdot)$ is a kernel function supported on a $g(t)$ neighborhood around $t$. It is assumed to be positive and symmetric about the origin. If $K(\cdot)$ has compact support on $[-1, 1]$, only the failure time falling into the interval $(t-g, t+g)$ will contribute to the log likelihood at time point $t$. Generally, $K(\cdot)$ is chosen such that $K(|s|)$ decays with $|s|$, so observations further from $t$ receive less weight in the likelihood formulation, which leads to the word *"local"* in the name "local EM algorithm".

The log of the local likelihood function at time t is given by

$$\ell(t) = \sum_{i=1}^{n} \ell(X_i, t|L_i, R_i). \tag{4.3}$$

In order to find the MLE of local log likelihood in formula 4.3, Betensky et al. [4] proposed the "Local EM estimation" in this situation. The main idea is to iterate between an E-step and a M-step. The E-step basically "imputes" the unobserved event times by calculating the conditional expectation of the local log likelihood given the observed data; the M-step maximizes the expected local log likelihood obtained in the E-step. See Betensky et al. [4] for more details.

### 4.2.2 Local EM Estimation of the Hazard Function for Interval Censored Competing Risk Data

Now, we extend the local EM algorithm in the Section 4.2.1 to allow for a competing risk. In our working example, the Nun study, we have two competing events, death, whose failure time is observed or right censored, and dementia, whose failure time is interval or right censored. Let $X_{i1}$ be the failure time of dementia, which is partially known and $X_{i2}$ be the failure time of death.

The cause-specific hazard (CSH) function of the $k$th competing event ($k = 1, 2$) refers to the instantaneous rate of a particular failure at a specific time in the presence of all other failures. A local polynomial approximation of order $p$ to the log of the CSH for the $i$th observation with failure time $x_i$ at the neighborhood of point $t$ is given by

$$\log\{\lambda_k(x_i|t)\} \approx \alpha_{kt_0} + \alpha_{kt_1}(x_i - t) + \cdots + \alpha_{kt_p}(x_i - t)^p, \qquad \text{for } |x_i - t| \leq g(t), \tag{4.4}$$

where $g(t)$ is the bandwidth. Let $\boldsymbol{\alpha}_{kt} = (\alpha_{kt_0}, \alpha_{kt_1}, \cdots, \alpha_{kt_p})$, for $k = 1, 2$.

To formulate the likelihood function, we need the density function of each competing event. We define the probability density function of the $k$-th competing event as $f_k(t)$. Due to the competing nature among the events, $f_k$ will involve not only $\lambda_k$ but also the CSHs of the other competing events, which is quite different from the case where no competing event is present. One can show the following connection (see also [28] ):

$$f_k(t) = \lambda_k(t) \exp\left( - \int_0^t \{\lambda_1(s) + \lambda_2(s)\} \mathrm{d}s \right).$$

The contribution of a subject at failure time s to the log likelihood at time point $t$ now takes the form of

$$\ell(s,t) = I_{(\varepsilon=1)} \times \log\Big\{\lambda_1(s|t)\Big\} + I_{(\varepsilon=2)} \times \log\Big\{\lambda_2(s|t)\Big\} - \int_0^s \Big\{\lambda_1(u|t) + \lambda_2(u|t)\Big\}\mathrm{d}u.$$

In the Nun study, time to death is observed, while the time to dementia is interval censored. In this case, we modify the local log likelihood of a failure time subject with failure time $x_i$ subject to interval censoring at time point $t$ as:

$$\begin{aligned}\int_L^R \ell_1(x_i,t)\mathrm{d}x_i &= \int_L^R K\left(\frac{x_i - t}{g}\right) \times \log\Big\{\lambda_1(x_i|t)\Big\}\mathrm{d}x_i \\ &\quad - \int_L^R \int_0^{x_i} \exp\Big\{ \log\lambda_1(u|t) + \log\lambda_2(u|t)\Big\} K\left(\frac{u-t}{g}\right)\mathrm{d}u\mathrm{d}x_i.\end{aligned}$$

If the failure time is observed, the contribution of a subject with failure time $s$ at time point $t$ is

$$\begin{aligned}\ell_2(x_i,t) &= K\left(\frac{x_i - t}{g}\right) \times \log\Big\{\lambda_2(X_i|t)\Big\} \\ &\quad - \int_0^{x_i} \exp\Big\{ \log\lambda_1(u|t) + \log\lambda_2(u|t)\Big\} K\left(\frac{u-t}{g}\right)\mathrm{d}u.\end{aligned}$$

If the failure time is right censored, the contribution of a subject with failure time $x_i$ to the local log likelihood at time $t$ is

$$\ell_3(x_i, t) = -\int_0^{x_i} \exp\Big\{ \log \lambda_1(u|t) + \log \lambda_2(u|t) \Big\} K\left(\frac{u-t}{g}\right) \mathrm{d}u.$$

Putting all three cases together, we have the following local log likelihood contributed from the $i$th subject at time $t$

$$\begin{aligned} \ell(x_i, t|L_i, R_i) &= \int_{L_i}^{R_i} I_{(\varepsilon_i=1)} \times \ell_1(x_i|t)\mathrm{d}x_i \\ &+ I_{(\varepsilon_i=2)} \times \ell_2(x_i|t) + I_{(\varepsilon_i=3)} \times \ell_3(x_i|t) \end{aligned} \tag{4.5}$$

The final log-local likelihood function is thus $\ell(t) = \sum_{i=1}^n \ell(x_i, t|L_i, R_i)$. One can maximize $\ell(t)$ with respect to the unknown $\alpha$ to get the MLE. But it is clear that there is no closed form maximizer, therefore, numerical procedures are needed. Despite a more complex likelihood function here, we find the "Local EM algorithm" in [4] is still applicable after some important modifications. Specifically we propose the following modified EM algorithm:

- Step 0: Choose the kernel function K, the bandwidths $g = g(t)$ and the grid of points of estimation $\varsigma$. For interval-censored data, set $T_i = (L_i + R_i)/2$. Using Nelson-Aalen to find the initial estimation of $\hat{\alpha}_{kt}$.

- Step 1: For each $t \in \varsigma$, set $\widehat{\lambda}_1(t) = \exp(\widehat{\alpha}_{1t_0})$ and $\widehat{\lambda}_2(t) = \exp(\widehat{\alpha}_{2t_0})$. The survival function $\widehat{S}(t)$ and density function $\widehat{f}_1(t)$ and $\widehat{f}_2(t)$ can be derived correspondingly.

- Step 2:For each $t \in \varsigma$,

$$\begin{aligned}(\widehat{\alpha}_{1t}, \widehat{\alpha}_{2t}) &= \underset{(\alpha_{1t},\alpha_{2t})}{\text{argmax}}\, \mathbb{E}_{\widehat{f}} \left[ \sum_{i=1}^{n} \ell(X_i, t|L_i, R_i) \right] \\ &= \underset{(\alpha_{1t},\alpha_{2t})}{\text{argmax}} \sum_{i=1}^{n} \left[ \frac{\int_{L_i}^{R_i} \ell(x_i, t)\widehat{f}_1(x_i)\mathrm{d}x_i}{\int_{L_i}^{R_i} \widehat{f}_1(x_i)\mathrm{d}x_i} \right]\end{aligned}$$

- Step 3: Repeat step 1 and 2 until convergence.

We have used here the notation

$$\begin{aligned}\ell(x_i, t) &= I_{(\varepsilon_i=1)} \times K\left(\frac{x_i - t}{g}\right) \times \log\Big\{\lambda_1(x_i|t)\Big\} + I_{(\varepsilon_i=2)} \times K\left(\frac{x_i - t}{g}\right) \\ &\times \log\Big\{\lambda_2(x_i|t)\Big\} - \int_0^{x_i} \exp\Big\{\log\lambda_1(u|t) + \log\lambda_2(u|t)\Big\} K\left(\frac{u - t}{g}\right) \mathrm{d}u,\end{aligned}$$

where $\log\big\{\lambda_k(x_i|t)\big\}$ is approximated by a polynomial function mentioned in (4.4). Although the formula of the M-step is very similar to the one in [4], the computational procedure is different. Under the competing risk framework, the two CSHs are coupled in the local log-likelihood function, which further complicates the computation.

Although it is possible to choose a polynomial of any degree in the local likelihood, it is always advisable to use $0, 1$ or $2$. If $p = 0$, there is an explicit expression of the M-step. Moreover, the computational burden to find the MLE of $(\hat{\alpha}_{1t}, \hat{\alpha}_{2t})$, although the idea is straightforward, will be heavy especially if $p \leq 1$ and it may result in a non-convergent algorithm. In this dissertation, instead of using a higher order of polynomial, we choose a low grid size, i.e. large number of grid points. By lowering the grid size, we can make sure the piecewise constant CSH (Wu and Tuma [54]) is suitable here. At the mean time, the algorithm can incorporate multiple competing risks with the combination of both observed and interval censored failure times straightforwardly.

The estimation of the standard error for CSH $\lambda_k(t)$ for interval-censored data is quite complicated. Resampling bootstrap method (Efron [12]), whose popularity is grounded in its simplicity and no assumption on the failure time, provides a simple yet flexible method to calculate the variance. In this dissertation, we use the quantiles of resampling bootstrap results to estimate the confidence interval of the CSH of two competing risks, dementia and death.

## 4.3 Numerical Studies

### 4.3.1 Simulation Studies

In this Section, we conducted some simulation studies to assess the reliability of the proposed EM algorithm in estimating the hazard function of a particular failure. We considered the case where both the failure times of dementia and death were generated using a Weibull hazard function $\lambda(t) = k\sigma^{-1}t^{(k-1)}$, where $k > 0$ and $\sigma > 0$. The different competing groups may have different choices of $k$ and $\sigma$, however, in this dissertation only results with same parameters for the two competing groups are shown here. In the simulation study, we chose $k$ to be 3 and $\sigma$ to be 20, which were selected to make sure the simulated failure times are close to the real data. Since the bandwidth $g$ is very important to smooth the hazard function and balance bias against variance, some simulation studies are also conducted to check the relationship among h, sample size, and the bias.

The competing risk data are generated as follows.

- Step 1: Generate two random samples from the Weibull distribution. Denote these as $S_1$ and $S_2$ where $S_1$ is the failure time from dementia and $S_2$ is the failure time from death.
- Step 2: Generate a random number from the uniform distribution supported

on $(0, q)$. Let the observed failure time $T$ be the minimum of $S_1$, $S_2$ and $C$. Let $\varepsilon$ be 1 if $T = S_1$ , 2 if $T = S_2$ and 0 otherwise. Here, the value of $q$ is chosen to ensure 20 percent of the data are right-censored.

- Step 3: If $\varepsilon$ is 1, then add some noise to the failure time $T$ to ensure the maximum length is 2.41 and minimum length is 0.75.

- Step 4: Repeat Step 1-3 M times.

For each simulation, we generated 500 samples with sample size M. The simulation results with sample size $M = 200, 400, 800$ and $g = 1, 2, 3$ are presented in Figure 4.1. The red lines are the true CSH curves of dementia: $\lambda_1(t) = \frac{3}{20}\frac{t}{20}^{(3-1)}$. The gray lines are the estimated $\lambda_1(t)$'s. The plots show that the variations of the $\lambda_1(t)$have a positive relationship with time t and a negative relationship with the sample size M for all three choices of h. For a small bandwidth ($g = 1$ or 2), the plots show little bias over the entire curve, whereas for a larger bandwidth ($g = 3$) the plots show large negative bias, especially when failure time t is large.

The result in Table 4.1 lists the relative bias and MSE of hazard function for the dementia hazard rate $\lambda_1(t)$ at time 10.0, 12.5, 15.0, 17.5 and 20.0 with different sample size M and bandwidth g. The bandwidth g has a dominated increasing influence in the magnitude of the bias. For example, when M=200, T=10, the relative biases are 0.68, -2.39 and 5.33 respectively for g=1,2 and 3. Sample size has an ignorable influence in the bias as compare to the MSE, which will decrease considerably with the increase in sample size. However, due to the existence of persistent biases, the percentage decreases of MSE are 77%, 65% and 12% for $g = 1, 2$ and 3, when the sample size increases from 200 to 800 respectively. Among the 3 options of bandwidth, g=2 gives us uniformly smallest MSE.

| | | g=1 | | | | | |
|---|---|---|---|---|---|---|---|
| time | CSH | M=200 | | M=400 | | M=800 | |
| | | Relative biases(%) | $MSE$ $\times 10^6$ | Relative biases(%) | $MSE$ $\times 10^6$ | Relative biases(%) | $MSE$ $\times 10^6$ |
| 10 | 0.0366 | 0.68 | 38.6 | 1.02 | 19.69 | 1.65 | 8.54 |
| 12.5 | 0.0575 | 0.52 | 81.2 | 1.35 | 43.22 | 1.05 | 17.25 |
| 15 | 0.083 | 1.56 | 169.82 | 1.68 | 90.69 | 1.29 | 45.55 |
| 17.5 | 0.1133 | 2.80 | 421.43 | 2.49 | 202.08 | 2.06 | 117.28 |
| 20 | 0.1482 | 3.49 | 1194.53 | 2.82 | 543.96 | 3.81 | 269.48 |
| | | g=2 | | | | | |
| time | CSH | M=200 | | M=400 | | M=800 | |
| 10.0 | 0.0366 | -2.39 | 19.13 | -2.53 | 9.21 | -1.88 | 5.62 |
| 12.5 | 0.0575 | -3.51 | 43.09 | -3.06 | 22.61 | -3.06 | 12.42 |
| 15.0 | 0.083 | -3.86 | 93.02 | -3.59 | 49.07 | -3.76 | 29.42 |
| 17.5 | 0.1133 | -4.39 | 204.69 | -4.91 | 111.79 | -4.61 | 71.65 |
| 20.0 | 0.1482 | -4.90 | 468.49 | -6.23 | 277.59 | -5.70 | 164.15 |
| | | g=3 | | | | | |
| time | CSH | M=200 | | M=400 | | M=800 | |
| 10.0 | 0.0366 | -5.33 | 15.3 | -5.80 | 10.09 | -5.62 | 6.90 |
| 12.5 | 0.0575 | -9.37 | 51.88 | -9.89 | 43.66 | -9.68 | 35.86 |
| 15.0 | 0.083 | -12.00 | 145.8 | -12.63 | 132.15 | -12.59 | 118.81 |
| 17.5 | 0.1133 | -14.21 | 348.07 | -14.84 | 327.64 | -14.92 | 306.21 |
| 20.0 | 0.1482 | -16.06 | 759.32 | -16.88 | 718.81 | -16.86 | 669.81 |

Table 4.1: Relative bias (%) and MSE of hazard function for dementia $\lambda_1(t)$ at time 10.0, 12.5, 15.0, 17.5 and 20.0 with different sample size M and bandwidth h

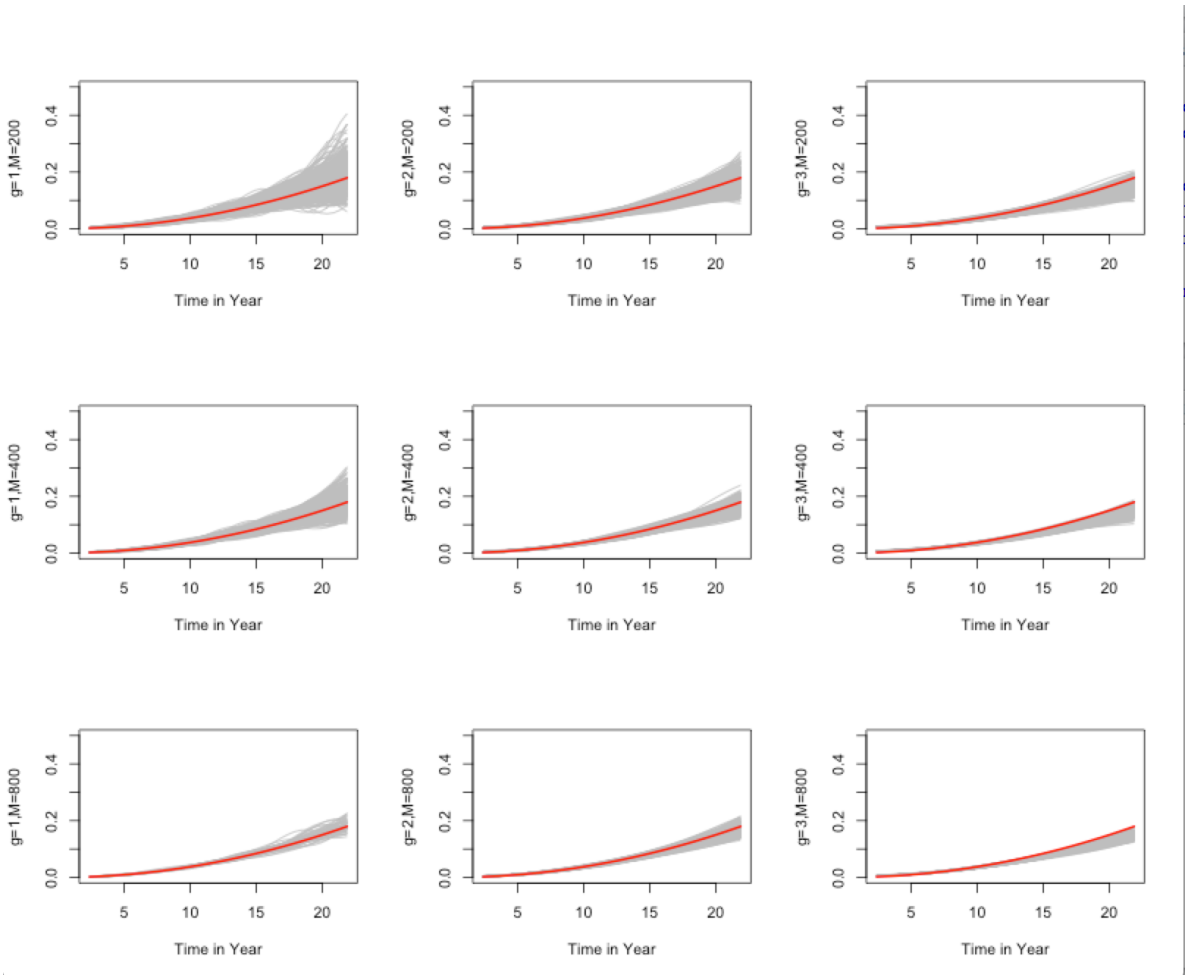


Figure 4.1: CSH function for dementia (Red lines: true CSH curves of dementia: $\lambda_1(t) = \frac{3}{20}\frac{t}{20}^{(3-1)}$. Gray lines: the estimated $\lambda_1(t)$'s.) with $M = 200, 400, 800$ and $g = 1, 2, 3$

Choice of the bandwidth for the local log likelihood estimation of CSH is critical and non-trivial. It is inherent in non-parametric research and has been researched considerably. Hufthammer and Tjostheim [23] showed that local likelihood density estimate has variance of order $O\{(ng)^{-1}\}$ and bias of order $O\{g^2 + (ng^3)^{-1}\}$, which is consistent with our simulation results, that both sample size and bandwidth have dominating influences in the variance while only bandwidth dominated the biases as long as the sample size is moderately large. For the bandwidth selection, some of the authors use the MSE criteria, which is to find the optimize bandwidth that will

minimize the estimated MSE. The fixed bandwidth is simple, however, it may introduce some bias at the boundary or sparse data area. A remedy to this problem is to use a dynamic bandwidth scheme, where the bandwidth varies with the sparseness of the data. In Betensky, et al. [5, 4], the bandwidth is choosen to make sure 40% of the data will contribute to the local log-likelihood. It can deal with the sparse data, however, boundary area is still a potential problem. In this dissertation, we use the simulation study to find the optimum bandwidth using the MSE criterion since the Nun Study data is only sparse in the right boundary area.

In summary, the simulation results show that the MSE of the CSH has a negative relationship with sample size, while the bias has almost no relationship with the sample size, as long as the sample size is moderately large. The biases increase with the bandwidths, while MSEs have a concave shape relationship with the bandwidths with a minimum at 2. For the data with a smooth increasing hazard function and a moderate sample size, $g = 2$ is an appropriate choice for bandwidth.

### 4.3.2 The Nun Study

We illustrate the local EM likelihood methodology with a prospective cohort study on dementia, the Nun Study data. Analyses were based on data from up to ten unevenly spaced examinations made in fifteen-year period. For a nun the study ends when either dementia or death occurred. Dementia is a key event subject to interval censored event times. The lengths of the corresponding intervals range from 0.75 to 2.41 years. In this application 211 of the 672 nuns were excluded from the analysis because, 61 (9.1%) had only one examination, 128 (19.0%) were demented at baseline visit, and 22 (3.3%) had missing APOE4 determinations. Therefore, the final analytic sample consisted of 461 participants, of which 74 are still at risk, 162 developed dementia and 225 died before converting to dementia.

One well known influential factor in the progression of dementia is APOE4. Most previous studies assumed that proportional hazards hold for the APOE4. Here, we want to use the non-parametric method proposed in Section 4.2 to check the proportional hazards assumption imposed on the effect of APOE4. At the same time, we want to confirm the proposal by some authors that both CSH and CIF should be reported in the competing risk study, since these may have different covariate effects.

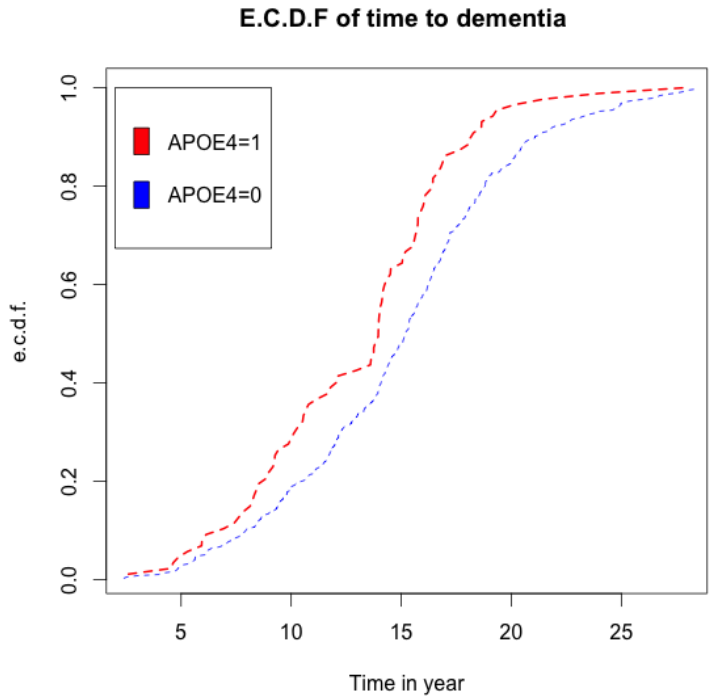


Figure 4.2: The empirical cumulative distribution function (E.C.D.F.) of time to dementia (time = age -75) by APOE4 carrier status

In the Nun Study, 87 of 461 or 18.9% are APOE4 carriers. The median failure time for those patients occurs near age 90. Figure 4.2 shows the empirical cumulative distribution function of time to dementia for APOE4 carrier status. The plot shows that the APOE4 carrier is more likely to becme demented at an earleri age than a non-carrier. Figure 4.3 presents the plot of CSH estimated and piecewise confidence

| Failure time | CSH Ratio | CI |
|---|---|---|
| 5 | 0.40 | (0.32,0.53) |
| 7.5 | 0.43 | (0.35,0.53) |
| 10 | 0.48 | (0.39,0.57) |
| 12.5 | 0.49 | (0.41,0.59) |
| 15 | 0.43 | (0.35,0.51) |
| 17.5 | 0.45 | (0.36,0.55) |
| 20 | 0.57 | (0.42,0.78) |
| 22.5 | 0.93 | (0.70,2.68) |

Table 4.2: Estimated CSH ratio of the APOE4 effect by failure time

interval of dementia with the presence APOE 4 and absence of APOE4. The choice of bandwidth parameter is 2, which is calculated from R package ('KernSmoonth'). With the choice of bandwidth equals 2, we can guarantee that 15 of the failure times contribute to the local log-likelihood except for the boundary area. From this plot, we can see the hazard to dementia increases as the Nuns ages for both groups. Moreover, there is a slope change for the hazard function at time of 12 years in both of those two APOE4 groups for dementia. Despite the small differences in those two competing events, those two plots show quit reasonable proportional hazard for presence and absence of APOE4 status for younger nuns. However, the two CSH curves almost cross each other when the nun gets older. The lack of proportionality in the covariate effect in a short of period may due to vulnerability to other diseases. Specifically, APOE 4 is a risk for Alzheimer's disease (AD) but as a nun ages she also becomes more susceptible to hippocampal sclerosis which is unaffected by APOE but is often clinically not different from AD [42] . The summary statistics of CSH Ratios of absence of APOE4 versus absence of APOE4 effects are presented in Table 4.2. Estimated CSH ratio as a function of time.

Figure 4.4 shows the CIF's of dementia with the present and absent of APOE4. The

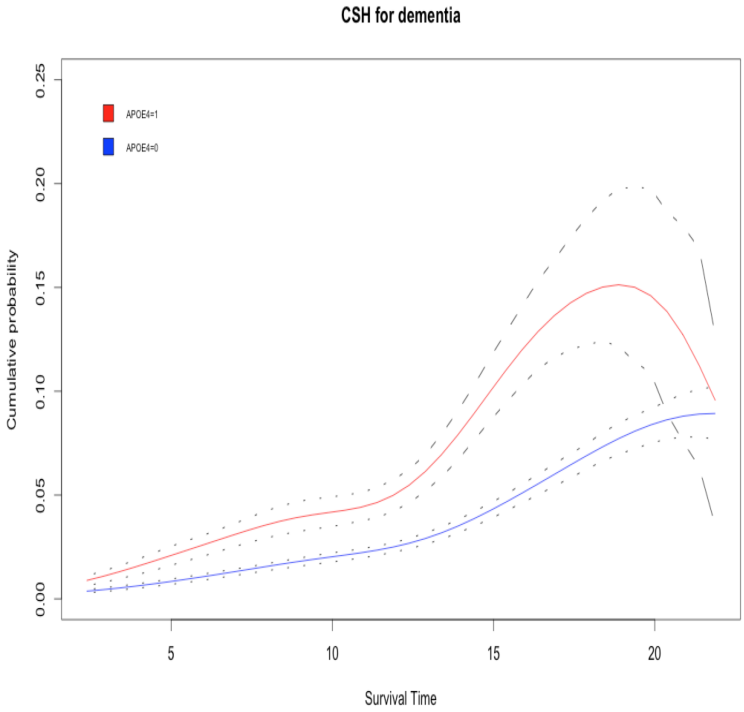


Figure 4.3: CSH function for dementia

two lines are different from the CSH in Figure 5 and are proportional to each other over all time. The difference is partially due to CIF is a function, average over all the CSHs before the specific time, which result in the insensitive to the change of covariate effects. This supports the proposal by Latouche et al. [36] and Hinchliffe and Paul [22], that a competing risk study should report both the CIFs and CSHs side by side.

## 4.4 Conclusion

We proposed a modified "Local EM algorithm" to estimate a smooth hazard function subject to a competing risk under the interval censoring setting. The proposed method provides a flexible non-parametric method to estimate the hazard function under interval censoring and competing risk. The proposed non-parametric local hazard estimation is useful in a variety of contexts. Firstly, because no restriction

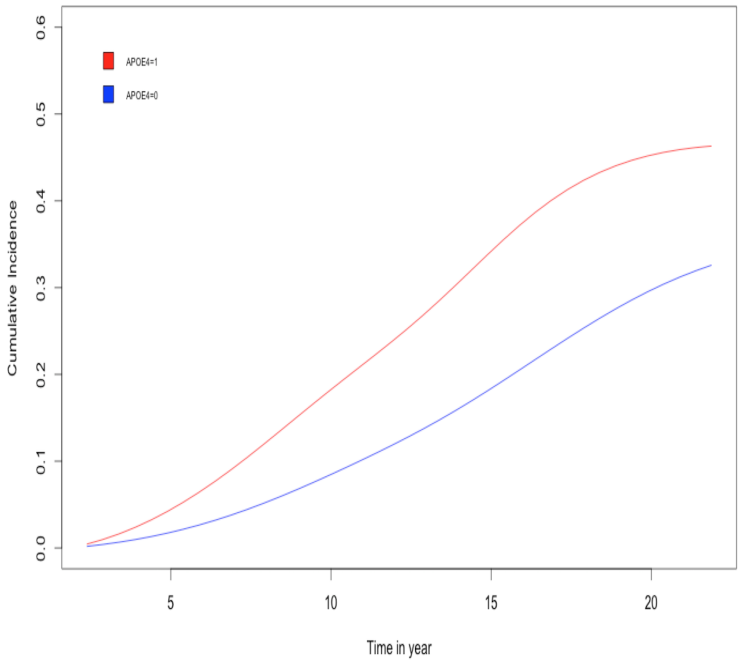


Figure 4.4: CSH function for dementia

or assumption on the shape of the hazard function, it can be used for exploratory research, such as graphically checking the effects of covariates or the shape of the baseline hazard and so on. Secondly, in the survival analysis, one always wants to check whether time-dependent covariates have an effect on the hazard function or not? Local hazard estimation provides a convenient way to check the assumption.

For example, the Nun's study data, the main purpose of this study is to examine the effects related the time and probability to dementia. Cumulative incidence function is a good statistic to exam the joint effect of competing risks. However, checking the proportional hazard assumption depends only on the CIF will result in biases. Therefore, both CIF and CSH should be reported side by side in order to better understand the structure of the data.

The proposed "Local EM algorithm" should be very straightforward to add the proportional hazard assumption for covariates at a single point, since it is a likelihood function based algorithm. However, the extension to the hazard function that allows for global parameters for proportional hazard is not trivial [5]. Making inferences about the effects of the covariates is also not trivial. More research in this area is needed. Further work is needed on the consequence for parameter estimators of choosing different bandwidths, i.e. choosing time-dependent bandwidth versus constant bandwidth.

## Chapter 5 Future Research

In this dissertation, we focus on the risk factors related to each transition and also that related to the survival component. We are pretty satisfied with what we do so far. However, there are some potential extensions for the future work.

The model proposed in the joint model has some obvious extensions. Firstly, only one competing risk event is considered in this dissertation. The extension to allow for multiple competing events is straightforward although the models will become more complex. Another extension of the model may include considering procedures that do not require a proportional hazard assumption. A Generalized Weibull model will be a good choice since the hazard can be U or inverse U shaped.

In the semi-Markov setting, we only consider the situation where there is no misclassification. However, misclassification is a problem, especially for subjects with very frequently jump between two states. Therefore a possible extension of semi-Markov model is to incorporate the information of misclassification.

Further investigation of the related model stability and verification of the model assumptions, such as Markov assumption for the transition component, the proportional hazard assumption on the survival time, and distribution of the holding time among two transitions are all of interest. Lastly, the application to the Nun Study data presented here emphasizes one step transition probabilities while clinically there is interest in the long run behavior of the process. That is, instead of estimating how a risk factor affects the odds of a transition into any impaired state at the next assessment there is also interest in determining how each risk factor affects the risk

of an eventual dementia diagnosis relative to dying without a dementia diagnosis. Results similar to those provided by [56] are needed for the model discussed here as well.

# Appendics

## 1. SAS Code for Halton Squence

```
proc fcmp outlib=sasuser.funcs.trial;
  function halton(base,I);
    J=I;  H=0;
    half=1/base;
    do while(J>0);
      digit=mod(J,base);
      H=H+digit*half;
      J=(J-digit)/base;
      half=half/base;
    end;
  return (H);
  endsub;
options cmplib=sasuser.funcs;
```

## 2. SAS Code for Higher Order Integration

```
%macro tranProb(pa1,ps1,cur11,cur12,cur13,cur14,cur16);
  if &ps1=0 then do;
   eta12=exp(intp2+apo12*apoe4+col12*ed12+grad12*ed3+age12*(&pa1+entrage));
   eta13= exp(intp3+apo13*apoe4+col13*ed12+grad13*ed3+age13*(&pa1+entrage));
   eta14=0;
   eta16= exp(0);
   den_eta1=eta12+eta13+eta14+eta16;
   p12=eta12/den_eta1; p13=eta13/den_eta1;
   p14=eta14/den_eta1; p16=eta16/den_eta1;
   eta21=exp(intp1+apo21*apoe4+col21*ed12+grad21*ed3+age21*(&pa1+entrage));
   eta23=exp(intp3+apo23*apoe4+col23*ed12+grad23*ed3+age23*(&pa1+entrage));
   eta24=exp(intp4+apo24*apoe4+col24*ed12+grad24*ed3+age24*(&pa1+entrage));
   eta26= exp(0);
   den_eta2=eta21+eta23+eta24+eta26;
   p21=eta21/den_eta2; p23=eta23/den_eta2;
   p24=eta24/den_eta2; p26=eta26/den_eta2;
   eta31=exp(intp1+apo31*apoe4+col31*ed12+grad31*ed3+age31*(&pa1+entrage));
   eta32= exp(intp2+apo32*apoe4+col32*ed12+grad32*ed3+age32*(&pa1+entrage));
   eta34= exp(intp4+apo34*apoe4+col34*ed12+grad34*ed3+age34*(&pa1+entrage));
   eta36= exp(0);
   den_eta3=eta31+eta32+eta34+eta36;
   p31=eta31/den_eta3 ; p32=eta32/den_eta3 ;
   p34=eta34/den_eta3 ; p36=eta36/den_eta3 ;
end;
  if &ps1=1 then do;
   eta12= exp(intp2+apo12*apoe4+col12*ed12+grad12*ed3+age12*(&pa1+entrage));
   eta13= exp(intp3+apo13*apoe4+col13*ed12+grad13*ed3+age13*(&pa1+entrage));
   eta14=0;
   eta16= exp(0);
   den_eta1=eta12+eta13+eta14+eta16;
   p12=eta12/den_eta1; p13=eta13/den_eta1;
   p14=eta14/den_eta1; p16=eta16/den_eta1;
   p2=p12*&cur12+p13*&cur13+p14*&cur14+p16*&cur16;
end;
  if &ps1=2 then do;
   eta21=exp(intp1+apo21*apoe4+col21*ed12+grad21*ed3+age21*(&pa1+entrage));
   eta23= exp(intp3+apo23*apoe4+col23*ed12+grad23*ed3+age23*(&pa1+entrage));
   eta24= exp(intp4+apo24*apoe4+col24*ed12+grad24*ed3+age24*(&pa1+entrage));
   eta26= exp(0);
   den_eta2=eta21+eta23+eta24+eta26; p21=eta21/den_eta2; p23=eta23/den_eta2;
   p24=eta24/den_eta2; p26=eta26/den_eta2;
   p2=p21*&cur11+p23*&cur13+p24*&cur14+p26*&cur16;
```

```
43     end;
44    if &ps1=3 then do;
45     eta31=exp(intp1+apo31*apoe4+col31*ed12+grad31*ed3+age31*(&pa1+entrage));
46     eta32= exp(intp2+apo32*apoe4+col32*ed12+grad32*ed3+age32*(&pa1+entrage));
47      eta34= exp(intp4+apo34*apoe4+col34*ed12+grad34*ed3+age34*(&pa1+entrage));
48     eta36= exp(0); den_eta3=eta31+eta32+eta34+eta36; p31=eta31/den_eta3 ;
49     p32=eta32/den_eta3 ; p34=eta34/den_eta3 ; p36=eta36/den_eta3 ;
50     p2=p31*&cur11+p32*&cur12+p34*&cur14+p36*&cur16;
51     end;
52 %mend tranProb;
53
54 %macro tranProbPara;
55 intp1,intp2,intp3,intp4,
56 apo12,apo13,apo21, apo23, apo24, apo31, apo32, apo34,
57 col12,col13,col21, col23, col24, col31, col32, col34,
58 grad12,grad13,grad21, grad23, grad24, grad31, grad32, grad34,
59 age12,age13,age21, age23, age24, age31, age32, age34
60 %mend tranProbPara;
61 %macro stateIndicator;
62 entrage,apoe4,ed12,ed3,survival,
63 ps1,pri11,pri12,pri13,
64 ps2,pri21,pri22,pri23,
65 ps3,pri31,pri32,pri33,
66 ps4,pri41,pri42,pri43,
67 ps5,pri51,pri52,pri53,
68 ps6,pri61,pri62,pri63,
69 ps7,pri71,pri72,pri73,
70 ps8,pri81,pri82,pri83,
71 cs1,cur11,cur12,cur13,cur14,cur16,
72 cs2,cur21,cur22,cur23,cur24,cur26,
73 cs3,cur31,cur32,cur33,cur34,cur36,
74 cs4,cur41,cur42,cur43,cur44,cur46,
75 cs5,cur51,cur52,cur53,cur54,cur56,
76 cs6,cur61,cur62,cur63,cur64,cur66,
77 cs7,cur71,cur72,cur73,cur74,cur76,
78 cs8,cur81,cur82,cur83,cur84,cur86,
79 pa1,pa2,pa3,pa4,pa5,pa6,pa7,pa8,
80 ca1,ca2,ca3,ca4,ca5,ca6,ca7,ca8
81 %mend stateIndicator;
82 %macro weibullPara;
83 k12, sigma12,k13, sigma13,k16, sigma16,
84 k21, sigma21,k23, sigma23,k24, sigma24,k26, sigma26,
85 k31, sigma31,k32, sigma32,k34, sigma34,k36, sigma36,
86 age112,age113,age116,
87 age121,age123,age124,age126,
88 age131,age132,age134,age136
89 %mend weibullPara;
90
91 %macro weibullParaAge(uu,age112,age113,age116,age121,age123,age124,age126,age131,
       age132,age134,age136);
92 sigmaf12=sigma12*exp(&age112*(&uu+entrage));
93 sigmaf13=sigma13*exp(&age113*(&uu+entrage));
94 sigmaf16=sigma16*exp(&age116*(&uu+entrage));
95 sigmaf21=sigma21*exp(&age121*(&uu+entrage));
96 sigmaf23=sigma23*exp(&age123*(&uu+entrage));
97 sigmaf24=sigma24*exp(&age124*(&uu+entrage));
98 sigmaf26=sigma26*exp(&age126*(&uu+entrage));
99 sigmaf31=sigma31*exp(&age131*(&uu+entrage));
100 sigmaf32=sigma32*exp(&age132*(&uu+entrage));
101 sigmaf34=sigma34*exp(&age134*(&uu+entrage));
102 sigmaf36=sigma36*exp(&age136*(&uu+entrage));
103 %mend weibullParaAge;
104 /*****************************/
105 proc fcmp outlib=sasuser.funcs.trial;
106 function  int_first1(n,sigma1,%weibullPara,%tranProbPara,%stateIndicator);
107    sum=0; h1=ca1-pa1; do i=1 to n;
108    uu=pa1+(i-0.5)*h1/n;
109    %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
```

```
110     pTran=p2;
111     %tranProb(uu,0,cur11,cur12,cur13,cur14,cur16);
112     sigma11=sigma1;
113   %weibullParaAge(uu,age112,age113,age116,age121,age123,age124,age126,age131,age132,
          age134,age136);
114     if cs1=1 then p1= pTran*p12*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k12/
            sigmaf12)+pTran*p13*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k13/
            sigmaf13)+pTran*p16*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k16/
            sigmaf16);
115    if cs1=2 then
116     p1= pTran*p21*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k21/sigmaf21)+pTran*
            p23*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k23/sigmaf23)+pTran*p24*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k24/sigmaf24)+pTran*p26*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k26/sigmaf26);
117    if cs1=3 then
118     p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k31/sigmaf31)+pTran*
            p32*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k32/sigmaf32)+pTran*p34*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k34/sigmaf34)+pTran*p36*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k36/sigmaf36);
119    if cs1=4 then
120     p1=pTran*1/sigma11*exp(-uu/sigma11);
121    if cs1=6 then
122     p1=pTran*1/sigma11*exp(-survival/sigma11);     sum=sum+p1;
123    end;
124    sum=sum*h1/n;
125    return (sum);
126    endsub;
127 options cmplib=sasuser.funcs;
128 plib=sasuser.funcs;
129
130 /*****************************/
131
132 proc fcmp outlib=sasuser.funcs.trial
133 function  int_first1(n,sigma1,%weibullPara,%tranProbPara,%stateIndicator)
134    sum=0; h1=ca1-pa1; do i=1 to n
135    uu=pa1+(i-0.5)*h1/n
136    %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16)
137    pTran=p2
138    %tranProb(uu,0,cur11,cur12,cur13,cur14,cur16)
139    sigma11=sigma1
140   %weibullParaAge(uu,age112,age113,age116,age121,age123,age124,age126,age131,age132,
          age134,age136)
141    if cs1=1 then
142     p1= pTran*p12*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k12/sigmaf12)+pTran*
            p13*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k13/sigmaf13)+pTran*p16*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k16/sigmaf16)
143    if cs1=2 then
144     p1= pTran*p21*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k21/sigmaf21)+pTran*
            p23*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k23/sigmaf23)+pTran*p24*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k24/sigmaf24)+pTran*p26*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k26/sigmaf26)
145    if cs1=3 then
146     p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k31/sigmaf31)+pTran*
            p32*1/sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k32/sigmaf32)+pTran*p34*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k34/sigmaf34)+pTran*p36*1/
            sigma11*exp(-uu/sigma11)*exp(-(survival-uu)**k36/sigmaf36)
147    if cs1=4 then
148     p1=pTran*1/sigma11*exp(-uu/sigma11)
149    if cs1=6 then
150     p1=pTran*1/sigma11*exp(-survival/sigma11);     sum=sum+p1
151    end
152    sum=sum*h1/n
153    return (sum)
154    endsub
155 options cmplib=sasuser.funcs
156 plib=sasuser.funcs
157
158 /*****************************/
```

```
159
160 proc fcmp outlib=sasuser.funcs.trial;
161 function int_first2(n,sigma1,k2,sigma2,age2,%weibullPara,%tranProbPara,%
     stateIndicator);
162    sum=0;
163    %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
164    pTran=p2;
165    %tranProb(ca1,ps2,cur21,cur22,cur23,cur24,cur26);
166    pTran=pTran*p2;
167    %tranProb(ca2,0,cur11,cur12,cur13,cur14,cur16);
168    h1=ca1-pa1;
169    h2=ca2-pa2;
170    do i=20 to n+19;
171    uu=pa1+halton(2,i)*h1;
172    vv=pa2+halton(3,i)*h2;
173    sigma11=sigma1;
174    sigma22=sigma2*exp(age2*(uu+entrage));
175    %weibullParaAge(vv,age112,age113,age116,age121,age123,age124,age126,age131,age132
         ,age134,age136);
176   if cs2=1 then
177    p1= pTran*p12*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
        **k2/sigma22)*exp(-(survival-vv)**k12/sigmaf12)+p13*1/sigma11*exp(-uu/sigma11)
        *k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*exp(-(survival-vv)**k13
        /sigmaf13)+p16*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv
        -uu)**k2/sigma22)*exp(-(survival-vv)**k16/sigmaf16);
178   if cs2=2 then
179    p1= pTran*p21*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
        **k2/sigma22)*exp(-(survival-vv)**k21/sigmaf21)+p23*1/sigma11*exp(-uu/sigma11)
        *k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*exp(-(survival-vv)**k23
        /sigmaf23)+p24*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv
        -uu)**k2/sigma22)*exp(-(survival-vv)**k24/sigmaf24)+p26*1/sigma11*exp(-uu/
        sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*exp(-(survival-
        vv)**k26/sigmaf26);
180   if cs2=3 then
181    p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
        **k2/sigma22)*exp(-(survival-vv)**k31/sigmaf31)+p32*1/sigma11*exp(-uu/sigma11)
        *k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*exp(-(survival-vv)**k32
        /sigmaf32)+p34*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv
        -uu)**k2/sigma22)*exp(-(survival-vv)**k34/sigmaf34)+p36*1/sigma11*exp(-uu/
        sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*exp(-(survival-
        vv)**k36/sigmaf36);
182   if cs2=4 then
183    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        );
184   if cs2=6 then
185    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(survival-uu)**(k2-1)*exp(-(survival-uu)
        **k2/sigma22);
186   sum=sum*pTran+p1;
187   end;
188   sum=sum*pTran*h1*h2/n;
189  return (sum);
190  endsub;
191 options cmplib=sasuser.funcs
192
193 /*******************************/
194
195 proc fcmp outlib=sasuser.funcs.trial;
196    function int_first3(n,sigma1,k2,sigma2,age2,k3,sigma3,age3,%weibullPara,%
        tranProbPara,%stateIndicator);
197    sum=0;
198    %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
199    pTran=p2;
200    %tranProb(ca1,ps2,cur21,cur22,cur23,cur24,cur26);
201    pTran=pTran*p2;
202    %tranProb(ca2,ps3,cur31,cur32,cur33,cur34,cur36);
203    pTran=pTran*p2;
204    %tranProb(ca3,0,cur41,cur42,cur43,cur44,cur46);
205    h1=ca1-pa1;
```

```
206     h2=ca2-pa2;
207     h3=ca3-pa3;
208     do i=20 to n+19;
209     uu=pa1+halton(2,i)*h1;
210     vv=pa2+halton(3,i)*h2;
211     ww=pa3+halton(5,i)*h3;
212     sigma11=sigma1;
213     sigma22=sigma2*exp(age2*(uu+entrage));
214     sigma33=sigma3*exp(age3*(vv+entrage));
215     %weibullParaAge(ww,age112,age113,age116,age121,age123,age124,age126,age131,age132
            ,age134,age136);
216    if cs3=1 then
217     p1=p12*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
            sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*exp(-(survival-
            ww)**k12/sigmaf12)+p13*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*
            exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33
            )*exp(-(survival-ww)**k13/sigmaf13)+p16*1/sigma11*exp(-uu/sigma11)*k2/sigma22
            *(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(
            ww-vv)**k3/sigma33)*exp(-(survival-ww)**k16/sigmaf16);
218    if cs3=2 then
219     p1=p21*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
            sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*exp(-(survival-
            ww)**k21/sigmaf21)+p23*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*
            exp(-(vvuu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)
            *exp(-(survival-ww)**k23/sigmaf23)+p24*1/sigma11*exp(-uu/sigma11)*k2/sigma22*
            (vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww
            -vv)**k3/sigma33)*exp(-(survival-ww)**k24/sigmaf24)+p26*1/sigma11*exp(-uu/
            sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-
            vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*exp(-(survival-ww)**k26/sigmaf26);
220    if cs3=3 then
221     p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
            *k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*exp(-(
            survival-ww)**k31/sigmaf31)+p32*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)
            **(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**
            k3/sigma33)*exp(-(survival-ww)**k32/sigmaf32)+p34*1/sigma11*exp(-uu/sigma11)*
            k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3
            -1)*exp(-(ww-vv)**k3/sigma33)*exp(-(survival-ww)**k34/sigmaf34)+p36*1/sigma11
            *exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/
            sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*exp(-(survival-ww)**k36/
            sigmaf36);
222    if cs3=4 then
223     p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
            )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33);
224    if cs3=6 then
225     p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
            )*k3/sigma33*(survival-vv)**(k3-1)*exp(-(survival-vv)**k3/sigma33);
226    sum=sum*pTran+p1;
227    end;
228    sum=sum*pTran*h1*h2*h3/n;
229    return (sum);
230   endsub;
231  options cmplib=sasuser.funcs;
232
233  /*****************************/
234
235  proc fcmp outlib=sasuser.funcs.trial;
236  function int_first4(n,sigma1,k2,sigma2,age2,k3,sigma3,age3,k4,sigma4,age4,%
         weibullPara,%tranProbPara,%stateIndicator);
237    sum=0;
238    %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
239     pTran=p2;
240     %tranProb(ca1,ps2,cur21,cur22,cur23,cur24,cur26);
241     pTran=pTran*p2;
242     %tranProb(ca2,ps3,cur31,cur32,cur33,cur34,cur36);
243     pTran=pTran*p2;
244     %tranProb(ca3,ps4,cur41,cur42,cur43,cur44,cur46);
245     pTran=pTran*p2;
246     %tranProb(ca4,0,cur51,cur52,cur53,cur54,cur56);
```

```
247     h1=ca1-pa1;
248     h2=ca2-pa2;
249     h3=ca3-pa3;
250     h4=ca4-pa4;
251     do i=20 to n+19;
252     uu=pa1+halton(2,i)*h1;
253     vv=pa2+halton(3,i)*h2;
254     ww=pa3+halton(5,i)*h3;
255     xx=pa4+halton(7,i)*h4;
256     sigma11=sigma1;
257     sigma22=sigma2*exp(age2*(uu+entrage));
258     sigma33=sigma3*exp(age3*(vv+entrage));
259     sigma44=sigma4*exp(age4*(ww+entrage));
260     %weibullParaAge(xx,age112,age113,age116,age121,age123,age124,age126,age131,age132
          ,age134,age136);
261    if cs4=1 then
262     p1=p12*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
          sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-
          ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k12/sigmaf12)+p13*1
          /sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
          )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(
          k4-1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k13/sigmaf13)+p16*1/
          sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)
          *k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4
          -1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k16/sigmaf16);
263    if cs4=2 then
264     p1=p21*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
          sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-
          ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k21/sigmaf21)+p23*1
          /sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
          )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(
          k4-1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k23/sigmaf23)+p24*1/
          sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)
          *k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4
          -1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k24/sigmaf24)+p26*1/sigma11
          *exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/
          sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*
          exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k26/sigmaf26);
265    if cs4=3 then
266     p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
          *k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*
          (xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k31/sigmaf31)+
          p32*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
          sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-
          ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k32/sigmaf32)+p34*1
          /sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
          )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(
          k4-1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k34/sigmaf34)+p36*1/
          sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)
          *k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4
          -1)*exp(-(xx-ww)**k4/sigma44)*exp(-(survival-xx)**k36/sigmaf36);
267    if cs4=4 then
268     p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
          )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(
          k4-1)*exp(-(xx-ww)**k4/sigma44);
269    if cs4=6 then
270     p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
          )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(
          k4-1)*exp(-(survival-ww)**k4/sigma44);
271    sum=sum*pTran+p1;
272    end;
273    sum=sum*pTran*h1*h2*h3*h4/n;
274   return (sum);
275   endsub;
276 options cmplib=sasuser.funcs;
277
278 /*****************************/
279
```

```
proc fcmp outlib=sasuser.funcs.trial;
   function int_first5(n,sigma1,k2,sigma2,age2,k3,sigma3,age3,k4,sigma4,age4,k5,
       sigma5,age5,%weibullPara,%tranProbPara,%stateIndicator);
   sum=0;
   %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
   pTran=p2;
   %tranProb(ca1,ps2,cur21,cur22,cur23,cur24,cur26);
   pTran=pTran*p2;
   %tranProb(ca2,ps3,cur31,cur32,cur33,cur34,cur36);
   pTran=pTran*p2;
   %tranProb(ca3,ps4,cur41,cur42,cur43,cur44,cur46);
   pTran=pTran*p2;
   %tranProb(ca4,ps5,cur51,cur52,cur53,cur54,cur56);
   pTran=pTran*p2;
   %tranProb(ca5,0,cur61,cur62,cur63,cur64,cur66);
   h1=ca1-pa1;
   h2=ca2-pa2;
   h3=ca3-pa3;
   h4=ca4-pa4;
   h5=ca5-pa5;
   do i=20 to n+19;
   uu=pa1+halton(2,i)*h1;
   vv=pa2+halton(3,i)*h2;
   ww=pa3+halton(5,i)*h3;
   xx=pa4+halton(7,i)*h4;
   yy=pa5+halton(11,i)*h5;
   sigma11=sigma1;
   sigma22=sigma2*exp(age2*(uu+entrage));
   sigma33=sigma3*exp(age3*(vv+entrage));
   sigma44=sigma4*exp(age4*(ww+entrage));
   sigma55=sigma5*exp(age5*(xx+entrage));
   %weibullParaAge(yy,age112,age113,age116,age121,age123,age124,age126,age131,age132
       ,age134,age136);
  if cs5=1 then
   p1=p12*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
       sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-
       ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)
       **k5/sigma55)*exp(-(survival-yy)**k12/sigmaf12)+p13*1/sigma11*exp(-uu/sigma11
       )*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(
       k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/
       sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*exp(-(survival-
       yy)**k13/sigmaf13)+p16*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*
       exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33
       )*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(
       k5-1)*exp(-(yy-xx)**k5/sigma55)*exp(-(survival-yy)**k16/sigmaf16);
  if cs5=2 then
   p1=p21*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
       sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-
       ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)
       **k5/sigma55)*exp(-(survival-yy)**k21/sigmaf21)+p23*1/sigma11*exp(-uu/sigma11
       )*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(
       k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/
       sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*exp(-(survival-
       yy)**k23/sigmaf23)+p24*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*
       exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33
       )*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(
       k5-1)*exp(-(yy-xx)**k5/sigma55)*exp(-(survival-yy)**k24/sigmaf24)+p26*1/
       sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)
       *k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4
       -1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
       sigma55)*exp(-(survival-yy)**k26/sigmaf26);
  if cs5=3 then
   p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
       *k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*
       (xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy
       -xx)**k5/sigma55)*exp(-(survival-yy)**k31/sigmaf31)+p32*1/sigma11*exp(-uu/
       sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-
       vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)
```

```
        **k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*exp(-(
        survival-yy)**k32/sigmaf32)+p34*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)
        **(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**
        k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(
        yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*exp(-(survival-yy)**k34/sigmaf34)+
        p36*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
        sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-
        ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)
        **k5/sigma55)*exp(-(survival-yy)**k36/sigmaf36);
317   if cs5=4 then
318    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(
        k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
        sigma55);
319   if cs5=6 then
320    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(
        k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(survival-xx)**(k5-1)*exp(-(
        survival-xx)**k5/sigma55);
321   sum=sum*pTran+p1;
322   end;
323   sum=sum*pTran*h1*h2*h3*h4*h5/n;
324   return (sum);
325  endsub;
326 options cmplib=sasuser.funcs;
327
328 /*****************************/
329
330 proc fcmp outlib=sasuser.funcs.trial;
331    function int_first6(n,sigma1,k2,sigma2,age2,k3,sigma3,age3,k4,sigma4,age4,k5,
        sigma5,age5,k6,sigma6,age6,%weibullPara,%tranProbPara,%stateIndicator);
332    sum=0;
333    %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
334    pTran=p2;
335    %tranProb(ca1,ps2,cur21,cur22,cur23,cur24,cur26);
336    pTran=pTran*p2;
337    %tranProb(ca2,ps3,cur31,cur32,cur33,cur34,cur36);
338    pTran=pTran*p2;
339    %tranProb(ca3,ps4,cur41,cur42,cur43,cur44,cur46);
340    pTran=pTran*p2;
341    %tranProb(ca4,ps5,cur51,cur52,cur53,cur54,cur56);
342    pTran=pTran*p2;
343    %tranProb(ca5,ps6,cur61,cur62,cur63,cur64,cur66);
344    pTran=pTran*p2;
345    %tranProb(ca6,0,cur71,cur72,cur73,cur74,cur76);
346    h1=ca1-pa1;
347    h2=ca2-pa2;
348    h3=ca3-pa3;
349    h4=ca4-pa4;
350    h5=ca5-pa5;
351    h6=ca6-pa6;
352    do i=20 to n+19;
353    uu=pa1+halton(2,i)*h1;
354    vv=pa2+halton(3,i)*h2;
355    ww=pa3+halton(5,i)*h3;
356    xx=pa4+halton(7,i)*h4;
357    yy=pa5+halton(11,i)*h5;
358    zz=pa6+halton(13,i)*h6;
359    sigma11=sigma1;
360    sigma22=sigma2*exp(age2*(uu+entrage));
361    sigma33=sigma3*exp(age3*(vv+entrage));
362    sigma44=sigma4*exp(age4*(ww+entrage));
363    sigma55=sigma5*exp(age5*(xx+entrage));
364    sigma66=sigma6*exp(age6*(yy+entrage));
365    %weibullParaAge(zz,age112,age113,age116,age121,age123,age124,age126,age131,age132
        ,age134,age136);
366   if cs6=1 then
367    p1=p12*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
```

```
        sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-
        ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)
        **k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* exp(-(
        survival-zz) **k12/sigmaf12)+p13*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu
        )**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)*
        *k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*
        (yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz
        -yy)**k6/sigma66)* exp(-(survival-zz)**k13/sigmaf13)+p16*1/sigma11*exp(-uu/
        sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-
         vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww
        )**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/
        sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* exp(-(survival-zz)**k16/
        sigmaf16);
368   if cs6=2 then
369    p1=p21*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
        sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(- (ww-vv)**k3/sigma33)*k4/sigma44*(xx
        -ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx
        )**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* exp(-(
        survival-zz) **k21/sigmaf21)+p23*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu
        )**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)*
        *k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*
        (yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz
        -yy)**k6/sigma66)* exp(-(survival-zz)**k23/sigmaf23)+p24*1/sigma11*exp(-uu/
        sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-
         vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww
        )**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/
        sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* exp(-(survival-zz)**k24/
        sigmaf24)+p26*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-
        uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/
        sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*
        exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66
        )* exp(-(survival-zz)**k26/sigmaf26);
370   if cs6=3 then
371    p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
        *k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww- vv)**k3/sigma33)*k4/sigma44
        *(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(
        yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* exp
        (-(survival-zz)**k31/sigmaf31)+p32*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-
        uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv
        )**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/
        sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*
        exp(-(zz-yy)**k6/sigma66)* exp(-(survival-zz)**k32/sigmaf32)+p34*1/sigma11*
        exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/
        sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)
        *exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
        sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* exp(-(survival
        -zz)**k34/sigmaf34)+p36*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)
        *exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/
        sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-
        xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)
        **k6/sigma66)* exp(-(survival-zz)**k36/sigmaf36);
372   if cs6=4 then
373    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv) **k3/sigma33)*k4/sigma44*(xx-ww)**(
        k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
        sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66);
374   if cs6=6 then
375    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv) **k3/sigma33)*k4/sigma44*(xx-ww)**(
        k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
        sigma55)*k6/sigma66*(survival-yy)**(k6-1)*exp(-(survival-yy)**k6/sigma66);
376   sum=sum*pTran+p1;
377   end;
378   sum=sum*pTran*h1*h2*h3*h4*h5*h6/n;
379   return (sum);
380  endsub;
381 options cmplib=sasuser.funcs;
382
```

```
383 /*****************************/
384
385 proc fcmp outlib=sasuser.funcs.trial;
386    function int_first7(n,sigma1,k2,sigma2,age2,k3,sigma3,age3,k4,sigma4,age4,k5,
           sigma5,age5,k6,sigma6,age6,k7,sigma7,age7,%weibullPara,%tranProbPara,%
           stateIndicator);
387    sum=0;
388    %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
389    pTran=p2;
390    %tranProb(ca1,ps2,cur21,cur22,cur23,cur24,cur26);
391    pTran=pTran*p2;
392    %tranProb(ca2,ps3,cur31,cur32,cur33,cur34,cur36);
393    pTran=pTran*p2;
394    %tranProb(ca3,ps4,cur41,cur42,cur43,cur44,cur46);
395    pTran=pTran*p2;
396    %tranProb(ca4,ps5,cur51,cur52,cur53,cur54,cur56);
397    pTran=pTran*p2;
398    %tranProb(ca5,ps6,cur61,cur62,cur63,cur64,cur66);
399    pTran=pTran*p2;
400    %tranProb(ca6,ps7,cur71,cur72,cur73,cur74,cur76);
401    pTran=pTran*p2;
402    %tranProb(ca7,0,cur81,cur82,cur83,cur84,cur86);
403    h1=ca1-pa1;
404    h2=ca2-pa2;
405    h3=ca3-pa3;
406    h4=ca4-pa4;
407    h5=ca5-pa5;
408    h6=ca6-pa6;
409    h7=ca7-pa7;
410    do i=20 to n+19;
411    uu=pa1+halton(2,i)*h1;
412    vv=pa2+halton(3,i)*h2;
413    ww=pa3+halton(5,i)*h3;
414    xx=pa4+halton(7,i)*h4;
415    yy=pa5+halton(11,i)*h5;
416    zz=pa6+halton(13,i)*h6;
417    aa=pa7+halton(17,i)*h7;
418    sigma11=sigma1;
419    sigma22=sigma2*exp(age2*(uu+entrage));
420    sigma33=sigma3*exp(age3*(vv+entrage));
421    sigma44=sigma4*exp(age4*(ww+entrage));
422    sigma55=sigma5*exp(age5*(xx+entrage));
423    sigma66=sigma6*exp(age6*(yy+entrage));
424    sigma77=sigma7*exp(age7*(zz+entrage));
425    %weibullParaAge(aa,age112,age113,age116,age121,age123,age124,age126,age131,age132
           ,age134,age136);
426   if cs7=1 then
427    p1=p12*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
           sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(- (ww-vv)**k3/sigma33)*k4/sigma44*(xx
           -ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx
           )**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/
           sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(survival-aa) **k12/
           sigmaf12)+p13*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-
           uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/
           sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*
           exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66
           )* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(survival-aa)**
           k13/sigmaf13)+p16*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp
           (-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*
           k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5
           -1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/
           sigma66)* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(
           survival-aa)**k16/sigmaf16);
428   if cs7=2 then
429    p1=p21*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
           sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(- (ww-vv)**k3/sigma33)*k4/sigma44*(xx
           -ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx
           )**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/
```

```
        sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(survival-aa) **k21/
        sigmaf21)+p23*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-
        uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/
        sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*
        exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66
        )* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(survival-aa)**
        k23/sigmaf23)+p24*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp
        (-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*
        k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5
        -1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/
        sigma66)* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(
        survival-aa)**k24/sigmaf24)+p26*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)
        **(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**
        k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(
        yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-
        yy)**k6/sigma66)* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp
        (-(survival-aa)**k26/sigmaf26);
430   if cs7=3 then
431    p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
        *k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww- vv)**k3/sigma33)*k4/sigma44
        *(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(
        yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/
        sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(survival-aa)**k31/
        sigmaf31)+p32*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-
        uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/
        sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*
        exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66
        )* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(survival-aa)**
        k32/sigmaf32)+p34*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp
        (-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*
        k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5
        -1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/
        sigma66)* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp(-(
        survival-aa)**k34/sigmaf34)+p36*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)
        **(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**
        k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(
        yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-
        yy)**k6/sigma66)* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* exp
        (-(survival-aa)**k36/sigmaf36);
432   if cs7=4 then
433    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv) **k3/sigma33)*k4/sigma44*(xx-ww)**(
        k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
        sigma55)*k6/sigma66*(zz-yy)**(k6-1)*k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**
        k7/sigma77)* exp(-(survival-aa)**k6/sigma66);
434   if cs7=6 then
435    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv) **k3/sigma33)*k4/sigma44*(xx-ww)**(
        k4-1)*exp(-(survival-ww)**k4/sigma44)*k5/sigma55*(survival-xx)**(k5-1)*exp(-(
        yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*k7/sigma77*(aa-zz)**(k7-1)*exp
        (-(aa-zz)**k7/sigma77)* exp(-(survival-aa)**k6/sigma66);
436   sum=sum*pTran+p1;
437   end;
438   sum=sum*pTran*h1*h2*h3*h4*h5*h6*h7/n;
439   return (sum);
440  endsub;
441 options cmplib=sasuser.funcs;
442
443 /*****************************/
444
445 proc fcmp outlib=sasuser.funcs.trial;
446    function int_first8(n,sigma1,k2,sigma2,age2,k3,sigma3,age3,k4,sigma4,age4,k5,
        sigma5,age5,k6,sigma6,age6,k7,sigma7,age7,k8,sigma8,age8,%weibullPara,%
        tranProbPara,%stateIndicator);
447    sum=0;
448    %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
449    pTran=p2;
450    %tranProb(ca1,ps2,cur21,cur22,cur23,cur24,cur26);
```

```
451     pTran=pTran*p2;
452     %tranProb(ca2,ps3,cur31,cur32,cur33,cur34,cur36);
453     pTran=pTran*p2;
454     %tranProb(ca3,ps4,cur41,cur42,cur43,cur44,cur46);
455     pTran=pTran*p2;
456     %tranProb(ca4,ps5,cur51,cur52,cur53,cur54,cur56);
457     pTran=pTran*p2;
458     %tranProb(ca5,ps6,cur61,cur62,cur63,cur64,cur66);
459     pTran=pTran*p2;
460     %tranProb(ca6,ps7,cur71,cur72,cur73,cur74,cur76);
461     pTran=pTran*p2;
462     %tranProb(ca7,ps8,cur81,cur82,cur83,cur84,cur86);
463     pTran=pTran*p2;
464     %tranProb(ca8,0,cur81,cur82,cur83,cur84,cur86);
465     h1=ca1-pa1;
466     h2=ca2-pa2;
467     h3=ca3-pa3;
468     h4=ca4-pa4;
469     h5=ca5-pa5;
470     h6=ca6-pa6;
471     h7=ca7-pa7;
472     h8=ca8-pa8;
473     do i=20 to n+19;
474     uu=pa1+halton(2,i)*h1;
475     vv=pa2+halton(3,i)*h2;
476     ww=pa3+halton(5,i)*h3;
477     xx=pa4+halton(7,i)*h4;
478     yy=pa5+halton(11,i)*h5;
479     zz=pa6+halton(13,i)*h6;
480     aa=pa7+halton(17,i)*h7;
481     bb=pa8+halton(19,i)*h8;
482     sigma11=sigma1;
483     sigma22=sigma2*exp(age2*(uu+entrage));
484     sigma33=sigma3*exp(age3*(vv+entrage));
485     sigma44=sigma4*exp(age4*(ww+entrage));
486     sigma55=sigma5*exp(age5*(xx+entrage));
487     sigma66=sigma6*exp(age6*(yy+entrage));
488     sigma77=sigma7*exp(age7*(zz+entrage));
489     sigma88=sigma8*exp(age8*(aa+entrage));
490     %weibullParaAge(bb,age112,age113,age116,age121,age123,age124,age126,age131,age132
            ,age134,age136);
491   if cs8=1 then
492     p1=p12*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
            sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(- (ww-vv)**k3/sigma33)*k4/sigma44*(xx
            -ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx
            )**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/
            sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)
            *exp(-(bb-aa)**k8/sigma88)* exp(-(survival-bb) **k12/sigmaf12)+p13*1/sigma11*
            exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/
            sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)
            *exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
            sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/sigma77*(aa
            -zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-
            aa)**k8/sigma88)* exp(-(survival-bb)**k13/sigmaf13)+p16*1/sigma11*exp(-uu/
            sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-
             vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww
            )**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/
            sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/sigma77*(aa-zz)**(k7-1)
            *exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-aa)**k8/
            sigma88)* exp(-(survival-bb)**k16/sigmaf16);
493   if cs8=2 then
494     p1=p21*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/
            sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(- (ww-vv)**k3/sigma33)*k4/sigma44*(xx
            -ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx
            )**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/
            sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)
            *exp(-(bb-aa)**k8/sigma88)* exp(-(survival-bb) **k21/sigmaf21)+p23*1/sigma11*
            exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/
```

```
        sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)
        *exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
        sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/sigma77*(aa
        -zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-
        aa)**k8/sigma88)* exp(-(survival-bb)**k23/sigmaf23)+p24*1/sigma11*exp(-uu/
        sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-
         vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww
        )**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/
        sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/sigma77*(aa-zz)**(k7-1)
        *exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-aa)**k8/
        sigma88)* exp(-(survival-bb)**k24/sigmaf24)+p26*1/sigma11*exp(-uu/sigma11)*k2
        /sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3
        -1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/
        sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-
        yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz
        )**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-aa)**k8/sigma88)* exp(-(
        survival-bb)**k26/sigmaf26);
495   if cs8=3 then
496    p1= pTran*p31*1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)*
        *k2/sigma22)*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww- vv)**k3/sigma33)*k4/sigma44
        *(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(
        yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/
        sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)
        *exp(-(bb-aa)**k8/sigma88)* exp(-(survival-bb)**k31/sigmaf31)+p32*1/sigma11*
        exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/
        sigma33*(ww- vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)
        *exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
        sigma55)*k6/sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/sigma77*(aa
        -zz)**(k7-1)*exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-
        aa)**k8/sigma88)* exp(-(survival-bb)**k32/sigmaf32)+p34*1/sigma11*exp(-uu/
        sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww-
         vv)**(k3-1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww
        )**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/
        sigma66*(zz-yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/sigma77*(aa-zz)**(k7-1)
        *exp(-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-aa)**k8/
        sigma88)* exp(-(survival-bb)**k34/sigmaf34)+p36*1/sigma11*exp(-uu/sigma11)*k2
        /sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22)*k3/sigma33*(ww- vv)**(k3
        -1)*exp(-(ww-vv)**k3/sigma33)*k4/sigma44*(xx-ww)**(k4-1)*exp(-(xx-ww)**k4/
        sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/sigma55)*k6/sigma66*(zz-
        yy)**(k6-1)*exp(-(zz-yy)**k6/sigma66)* k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz
        )**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-aa)**k8/sigma88)* exp(-(
        survival-bb)**k36/sigmaf36);
497   if cs8=4 then
498    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv) **k3/sigma33)*k4/sigma44*(xx-ww)**(
        k4-1)*exp(-(xx-ww)**k4/sigma44)*k5/sigma55*(yy-xx)**(k5-1)*exp(-(yy-xx)**k5/
        sigma55)*k6/sigma66*(zz-yy)**(k6-1)*k7/sigma77*(aa-zz)**(k7-1)*exp(-(aa-zz)**
        k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-aa)**k8/sigma88)* exp(-(
        survival-bb)**k6/sigma66);
499   if cs8=6 then
500    p1=1/sigma11*exp(-uu/sigma11)*k2/sigma22*(vv-uu)**(k2-1)*exp(-(vv-uu)**k2/sigma22
        )*k3/sigma33*(ww-vv)**(k3-1)*exp(-(ww-vv) **k3/sigma33)*k4/sigma44*(xx-ww)**(
        k4-1)*exp(-(survival-ww)**k4/sigma44)*k5/sigma55*(survival-xx)**(k5-1)*exp(-(
        yy-xx)**k5/sigma55)*k6/sigma66*(zz-yy)**(k6-1)*k7/sigma77*(aa-zz)**(k7-1)*exp
        (-(aa-zz)**k7/sigma77)* k8/sigma88*(bb-aa)**(k8-1)*exp(-(bb-aa)**k8/sigma88)*
         exp(-(survival-bb)**k6/sigma66);
501   sum=sum*pTran+p1;
502   end;
503   sum=sum*pTran*h1*h2*h3*h4*h5*h6*h7*h8/n;
504   return (sum);
505  endsub;
506 options cmplib=sasuser.funcs;
```

## 3. SAS Code for Semi-Markove Model Fitting

```
1 %macro semiMarkov;
```

```
ods output ParameterEstimates=est CovMatParmEst=cov FitStatistics=fit;
proc nlmixed data=temp2 cov  MAXITER=1000  tech=DBLDOG;
parms
age12=-0.1  age13=-0.1  age21=-0.1
age23=-0.1  age24=-0.1  age31=-0.1
age32=-0.1  age34=-0.1  age112=-0.1
age113=-0.1 age116=-0.1 age121=-0.1
age123=-0.1 age124=-0.1 age126=-0.1
age131=-0.1 age132=-0.1 age134=-0.1
age136=-0.1 bage12=-0.1 bage13=-0.1
bage16=-0.1 bage21=-0.1 bage23=-0.1
bage24=-0.1 bage26=-0.1 bage31=-0.1
bage32=-0.1 bage34=-0.1 bage36=-0.1
int12=1   int13=1   int16=1
int21=1   int23=1   int24=1
int26=1   int32=1   int34=1
int36=1   intp1=1   intp2=1
intp3=1   intp4=1
k112=1    k113=1    k116=1
k121=1    k123=1    k124=1
k126=1    k132=1    k134=1
k136=1
sigma112=1  sigma113=1  sigma116=1
sigma121=1  sigma123=1  sigma124=1
sigma126=1  sigma131=1  sigma132=1
sigma134=1  sigma136=1
;
k131=0;   apo14 =-5;  col14 =-5;
grad14=-5;  age14 =-5;
sigma114=0; fsigma14=0;

n=500;
lambda12= int12 + bage12*entrage;
lambda13= int13 + bage13*entrage;
lambda16= int16 + bage16*entrage;
fsigma12=exp(lambda12);
fsigma13=exp(lambda13);
fsigma16=exp(lambda16);
lambda21= int21 +bage21*entrage;
lambda23= int23 +bage23*entrage;
lambda24= int24 +bage24*entrage;
lambda26= int26 +bage26*entrage;
fsigma21=exp(lambda21);
fsigma23=exp(lambda23);
fsigma24=exp(lambda24);
fsigma26=exp(lambda26);
lambda31= int31 + bage31*entrage;
lambda32= int32 + bage32*entrage;
lambda34= int34 + bage34*entrage;
lambda36= int36 + bage36*entrage;
fsigma31=exp(lambda31);
fsigma32=exp(lambda32);
fsigma34=exp(lambda34);
fsigma36=exp(lambda36);
k12 =exp(k112); k13 =exp(k113);
k16 =exp(k116); k21=exp(k121);
k23=exp(k123);  k24=exp(k124);
k26=exp(k126);  k31=exp(k131);
k32=exp(k132);  k34=exp(k134);
k36=exp(k136);
sigma12=exp(sigma112);
sigma13=exp(sigma113);
sigma16=exp(sigma116);
sigma21=exp(sigma121);
sigma23=exp(sigma123);
sigma24=exp(sigma124);
sigma26=exp(sigma126);
sigma31=exp(sigma131);
```

```
70 sigma32=exp(sigma132);
71 sigma34=exp(sigma134);
72 sigma36=exp(sigma136);
73 %tranProb(0,ps1,cur11,cur12,cur13,cur14,cur16);
74 if ps1=1 & cs1=1 then
75  p1=p12*exp(-survival/fsigma12)
76    +p13*exp(-survival/fsigma13)
77    +p16*exp(-survival/fsigma16);
78 if ps1=2 & cs1=2 then
79  p1=p21*exp(-survival/fsigma21)
80    +p23*exp(-survival/fsigma23)
81    +p24*exp(-survival/fsigma24)
82    +p26*exp(-survival/fsigma26);
83 if (indxi=1) then do;
84   if ps1=1 & cs1=2   then
85    p1=int_first1(n,fsigma12,%weibullPara,%tranProbPara,%stateIndicator);
86   if ps1=1 & cs1=3   then
87    p1=int_first1(n,fsigma13,%weibullPara,%tranProbPara,%stateIndicator);
88   if ps1=1 & cs1=6   then
89    p1=int_first1(n,fsigma16,%weibullPara,%tranProbPara,%stateIndicator);
90   if ps1=2 & cs1=1   then
91    p1=int_first1(n,fsigma21,%weibullPara,%tranProbPara,%stateIndicator);
92   if ps1=2 & cs1=3   then
93    p1=int_first1(n,fsigma23,%weibullPara,%tranProbPara,%stateIndicator);
94   if ps1=2 & cs1=4   then
95    p1=int_first1(n,fsigma24,%weibullPara,%tranProbPara,%stateIndicator);
96   if ps1=2 & cs1=6   then
97    p1=int_first1(n,fsigma26,%weibullPara,%tranProbPara,%stateIndicator);
98   if ps1=3 & cs1=1   then
99    p1=int_first1(n,fsigma31,%weibullPara,%tranProbPara,%stateIndicator);
100  if ps1=3 & cs1=2   then
101   p1=int_first1(n,fsigma32,%weibullPara,%tranProbPara,%stateIndicator);
102  if ps1=3 & cs1=4   then
103   p1=int_first1(n,fsigma34,%weibullPara,%tranProbPara,%stateIndicator);
104  if ps1=3 & cs1=6   then
105   p1=int_first1(n,fsigma36,%weibullPara,%tranProbPara,%stateIndicator);
106 end;
107 if (indxi=2) then do;
108   if ps1=1 & ps2=2 & cs2=1 then
109     p1=int_first2(n,fsigma12,k21,sigma21,age121,%weibullPara,%tranProbPara,%
          stateIndicator);
110   if ps1=1 & ps2=2 & cs2=3 then
111    p1=int_first2(n,fsigma12,k23,sigma23,age123,%weibullPara,%tranProbPara,%
         stateIndicator);
112   if ps1=1 & ps2=2 & cs2=4 then
113    p1=int_first2(n,fsigma12,k24,sigma24,age124,%weibullPara,%tranProbPara,%
         stateIndicator);
114   if ps1=1 & ps2=2 & cs2=6 then
115     p1=int_first2(n,fsigma12,k26,sigma26,age126,%weibullPara,%tranProbPara,%
          stateIndicator);
116   if ps1=1 & ps2=3 & cs2=1 then
117    p1=int_first2(n,fsigma13,k31,sigma31,age131,%weibullPara,%tranProbPara,%
         stateIndicator);
118   if ps1=1 & ps2=3 & cs2=2 then
119    p1=int_first2(n,fsigma13,k32,sigma32,age132,%weibullPara,%tranProbPara,%
         stateIndicator);
120   if ps1=1 & ps2=3 & cs2=4 then
121    p1=int_first2(n,fsigma13,k34,sigma34,age134,%weibullPara,%tranProbPara,%
         stateIndicator);
122   if ps1=1 & ps2=3 & cs2=6 then
123    p1=int_first2(n,fsigma13,k36,sigma36,age136,%weibullPara,%tranProbPara,%
         stateIndicator);
124   if ps1=2 & ps2=1 & cs2=2 then
125    p1=int_first2(n,fsigma21,k12,sigma12,age112,%weibullPara,%tranProbPara,%
         stateIndicator);
126   if ps1=2 & ps2=1 & cs2=3 then
127    p1=int_first2(n,fsigma21,k13,sigma13,age113,%weibullPara,%tranProbPara,%
         stateIndicator);
```

```
128   if ps1=2 & ps2=1 & cs2=6 then
129    p1=int_first2(n,fsigma21,k16,sigma16,age116,%weibullPara,%tranProbPara,%
           stateIndicator);
130   if ps1=2 & ps2=3 & cs2=1 then
131    p1=int_first2(n,fsigma23,k31,sigma31,age131,%weibullPara,%tranProbPara,%
           stateIndicator);
132   if ps1=2 & ps2=3 & cs2=2 then
133    p1=int_first2(n,fsigma23,k32,sigma32,age132,%weibullPara,%tranProbPara,%
           stateIndicator);
134   if ps1=2 & ps2=3 & cs2=4 then
135    p1=int_first2(n,fsigma23,k34,sigma34,age134,%weibullPara,%tranProbPara,%
           stateIndicator);
136   if ps1=2 & ps2=3 & cs2=6 then
137    p1=int_first2(n,fsigma23,k36,sigma36,age136,%weibullPara,%tranProbPara,%
           stateIndicator);
138   if ps1=3 & ps2=1 & cs2=2 then
139    p1=int_first2(n,fsigma31,k12,sigma12,age112,%weibullPara,%tranProbPara,%
           stateIndicator);
140   if ps1=3 & ps2=1 & cs2=3 then
141    p1=int_first2(n,fsigma31,k13,sigma13,age113,%weibullPara,%tranProbPara,%
           stateIndicator);
142   if ps1=3 & ps2=1 & cs2=6 then
143    p1=int_first2(n,fsigma31,k16,sigma16,age116,%weibullPara,%tranProbPara,%
           stateIndicator);
144   if ps1=3 & ps2=2 & cs2=1 then
145    p1=int_first2(n,fsigma32,k21,sigma21,age121,%weibullPara,%tranProbPara,%
           stateIndicator);
146   if ps1=3 & ps2=2 & cs2=3 then
147    p1=int_first2(n,fsigma32,k23,sigma23,age123,%weibullPara,%tranProbPara,%
           stateIndicator);
148   if ps1=3 & ps2=2 & cs2=4 then
149    p1=int_first2(n,fsigma32,k24,sigma24,age124,%weibullPara,%tranProbPara,%
           stateIndicator);
150   if ps1=3 & ps2=2 & cs2=6 then
151    p1=int_first2(n,fsigma32,k26,sigma26,age126,%weibullPara,%tranProbPara,%
           stateIndicator);
152 end;
153 if (indxi=3) then do;
154   if ps1=1 & ps2=2 & ps3=1& cs3=2 then
155    p1=int_first3(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,%weibullPara,%
           tranProbPara,%stateIndicator);
156   if ps1=1 & ps2=2 & ps3=1& cs3=3 then
157    p1=int_first3(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,%weibullPara,%
           tranProbPara,%stateIndicator);
158   if ps1=1 & ps2=2 & ps3=1& cs3=6 then
159    p1=int_first3(n,fsigma12,k21,sigma21,age121,k16,sigma16,age116,%weibullPara,%
           tranProbPara,%stateIndicator);
160   if ps1=1 & ps2=2 & ps3=3& cs3=1 then
161     p1=int_first3(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,%weibullPara,%
            tranProbPara,%stateIndicator);
162   if ps1=1 & ps2=2 & ps3=3& cs3=2 then
163    p1=int_first3(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,%weibullPara,%
           tranProbPara,%stateIndicator);
164   if ps1=1 & ps2=2 & ps3=3& cs3=4 then
165    p1=int_first3(n,fsigma12,k23,sigma23,age123,k34,sigma34,age134,%weibullPara,%
           tranProbPara,%stateIndicator);
166   if ps1=1 & ps2=2 & ps3=3& cs3=6 then
167    p1=int_first3(n,fsigma12,k23,sigma23,age123,k36,sigma36,age136,%weibullPara,%
           tranProbPara,%stateIndicator);
168   if ps1=1 & ps2=3 & ps3=1& cs3=2 then
169    p1=int_first3(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,%weibullPara,%
           tranProbPara,%stateIndicator);
170   if ps1=1 & ps2=3 & ps3=1& cs3=3 then
171    p1=int_first3(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,%weibullPara,%
           tranProbPara,%stateIndicator);
172   if ps1=1 & ps2=3 & ps3=1& cs3=6 then
173    p1=int_first3(n,fsigma13,k31,sigma31,age131,k16,sigma16,age116,%weibullPara,%
           tranProbPara,%stateIndicator);
```

```
174  if ps1=1 & ps2=3 & ps3=2& cs3=1 then
175   p1=int_first3(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,%weibullPara,%
          tranProbPara,%stateIndicator);
176  if ps1=1 & ps2=3 & ps3=2& cs3=3 then
177   p1=int_first3(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,%weibullPara,%
          tranProbPara,%stateIndicator);
178  if ps1=1 & ps2=3 & ps3=2& cs3=4 then
179   p1=int_first3(n,fsigma13,k32,sigma32,age132,k24,sigma24,age124,%weibullPara,%
          tranProbPara,%stateIndicator);
180  if ps1=1 & ps2=3 & ps3=2& cs3=6 then
181   p1=int_first3(n,fsigma13,k32,sigma32,age132,k26,sigma26,age126,%weibullPara,%
          tranProbPara,%stateIndicator);
182  if ps1=2 & ps2=1 & ps3=2& cs3=1 then
183   p1=int_first3(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,%weibullPara,%
          tranProbPara,%stateIndicator);
184  if ps1=2 & ps2=1 & ps3=2& cs3=3 then
185   p1=int_first3(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,%weibullPara,%
          tranProbPara,%stateIndicator);
186  if ps1=2 & ps2=1 & ps3=2& cs3=4 then
187   p1=int_first3(n,fsigma21,k12,sigma12,age112,k24,sigma24,age124,%weibullPara,%
          tranProbPara,%stateIndicator);
188  if ps1=2 & ps2=1 & ps3=2& cs3=6 then
189   p1=int_first3(n,fsigma21,k12,sigma12,age112,k26,sigma26,age126,%weibullPara,%
          tranProbPara,%stateIndicator);
190  if ps1=2 & ps2=1 & ps3=3& cs3=1 then
191   p1=int_first3(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,%weibullPara,%
          tranProbPara,%stateIndicator);
192  if ps1=2 & ps2=1 & ps3=3& cs3=2 then
193   p1=int_first3(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,%weibullPara,%
          tranProbPara,%stateIndicator);
194  if ps1=2 & ps2=1 & ps3=3& cs3=4 then
195   p1=int_first3(n,fsigma21,k13,sigma13,age113,k34,sigma34,age134,%weibullPara,%
          tranProbPara,%stateIndicator);
196  if ps1=2 & ps2=1 & ps3=3& cs3=6 then
197   p1=int_first3(n,fsigma21,k13,sigma13,age113,k36,sigma36,age136,%weibullPara,%
          tranProbPara,%stateIndicator);
198  if ps1=2 & ps2=3 & ps3=1& cs3=2 then
199   p1=int_first3(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,%weibullPara,%
          tranProbPara,%stateIndicator);
200  if ps1=2 & ps2=3 & ps3=1& cs3=3 then
201   p1=int_first3(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,%weibullPara,%
          tranProbPara,%stateIndicator);
202  if ps1=2 & ps2=3 & ps3=1& cs3=6 then
203   p1=int_first3(n,fsigma23,k31,sigma31,age131,k16,sigma16,age116,%weibullPara,%
          tranProbPara,%stateIndicator);
204  if ps1=2 & ps2=3 & ps3=2& cs3=1 then
205   p1=int_first3(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,%weibullPara,%
          tranProbPara,%stateIndicator);
206  if ps1=2 & ps2=3 & ps3=2& cs3=3 then
207   p1=int_first3(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,%weibullPara,%
          tranProbPara,%stateIndicator);
208  if ps1=2 & ps2=3 & ps3=2& cs3=4 then
209   p1=int_first3(n,fsigma23,k32,sigma32,age132,k24,sigma24,age124,%weibullPara,%
          tranProbPara,%stateIndicator);
210  if ps1=2 & ps2=3 & ps3=2& cs3=6 then
211   p1=int_first3(n,fsigma23,k32,sigma32,age132,k26,sigma26,age126,%weibullPara,%
          tranProbPara,%stateIndicator);
212  if ps1=3 & ps2=1 & ps3=2& cs3=1 then
213   p1=int_first3(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,%weibullPara,%
          tranProbPara,%stateIndicator);
214  if ps1=3 & ps2=1 & ps3=2& cs3=3 then
215   p1=int_first3(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,%weibullPara,%
          tranProbPara,%stateIndicator);
216  if ps1=3 & ps2=1 & ps3=2& cs3=4 then
217   p1=int_first3(n,fsigma31,k12,sigma12,age112,k24,sigma24,age124,%weibullPara,%
          tranProbPara,%stateIndicator);
218  if ps1=3 & ps2=1 & ps3=2& cs3=6 then
219   p1=int_first3(n,fsigma31,k12,sigma12,age112,k26,sigma26,age126,%weibullPara,%
```

```
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=3& cs3=1 then
   p1=int_first3(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=3& cs3=2 then
   p1=int_first3(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=3& cs3=4 then
   p1=int_first3(n,fsigma31,k13,sigma13,age113,k34,sigma34,age134,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=3& cs3=6 then
   p1=int_first3(n,fsigma31,k13,sigma13,age113,k36,sigma36,age136,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=1& cs3=2 then
   p1=int_first3(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=1& cs3=3 then
   p1=int_first3(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=1& cs3=6 then
   p1=int_first3(n,fsigma32,k21,sigma21,age121,k16,sigma16,age116,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& cs3=1 then
   p1=int_first3(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& cs3=2 then
   p1=int_first3(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,%weibullPara,%
        tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& cs3=6 then
   p1=int_first3(n,fsigma32,k23,sigma23,age123,k36,sigma36,age136,%weibullPara,%
        tranProbPara,%stateIndicator);
end;
if (indxi=4) then do;
  if ps1=1 & ps2=2 & ps3=1& ps4=2 & cs4=1 then
   p1=int_first4(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=1& ps4=2 & cs4=3 then
   p1=int_first4(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=1& ps4=2 & cs4=4 then
   p1=int_first4(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k24,sigma24,age124
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=1& ps4=2 & cs4=6 then
   p1=int_first4(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k26,sigma26,age126
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=1& ps4=3 & cs4=1 then
   p1=int_first4(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k31,sigma31,age131
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=1& ps4=3 & cs4=2 then
   p1=int_first4(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=1& ps4=3 & cs4=4 then
   p1=int_first4(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k34,sigma34,age134
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=1& ps4=3 & cs4=6 then
   p1=int_first4(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k36,sigma36,age136
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=3& ps4=1 & cs4=2 then
   p1=int_first4(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=3& ps4=1 & cs4=3 then
   p1=int_first4(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=3& ps4=1 & cs4=6 then
   p1=int_first4(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k16,sigma16,age116
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=1 & ps2=2 & ps3=3& ps4=2 & cs4=1 then
   p1=int_first4(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k21,sigma21,age121
```

```
      ,%weibullPara,%tranProbPara,%stateIndicator);
266  if ps1=1 & ps2=2 & ps3=3& ps4=2 & cs4=3 then
267   p1=int_first4(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
      ,%weibullPara,%tranProbPara,%stateIndicator);
268  if ps1=1 & ps2=2 & ps3=3& ps4=2 & cs4=4 then
269   p1=int_first4(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k24,sigma24,age124
      ,%weibullPara,%tranProbPara,%stateIndicator);
270  if ps1=1 & ps2=2 & ps3=3& ps4=2 & cs4=6 then
271   p1=int_first4(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k26,sigma26,age126
      ,%weibullPara,%tranProbPara,%stateIndicator);
272  if ps1=1 & ps2=3 & ps3=1& ps4=2 & cs4=1 then
273   p1=int_first4(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k21,sigma21,age121
      ,%weibullPara,%tranProbPara,%stateIndicator);
274  if ps1=1 & ps2=3 & ps3=1& ps4=2 & cs4=3 then
275   p1=int_first4(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
      ,%weibullPara,%tranProbPara,%stateIndicator);
276  if ps1=1 & ps2=3 & ps3=1& ps4=2 & cs4=4 then
277   p1=int_first4(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k24,sigma24,age124
      ,%weibullPara,%tranProbPara,%stateIndicator);
278  if ps1=1 & ps2=3 & ps3=1& ps4=2 & cs4=6 then
279   p1=int_first4(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k26,sigma26,age126
      ,%weibullPara,%tranProbPara,%stateIndicator);
280  if ps1=1 & ps2=3 & ps3=1& ps4=3 & cs4=1 then
281   p1=int_first4(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131
      ,%weibullPara,%tranProbPara,%stateIndicator);
282  if ps1=1 & ps2=3 & ps3=1& ps4=3 & cs4=2 then
283   p1=int_first4(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
      ,%weibullPara,%tranProbPara,%stateIndicator);
284  if ps1=1 & ps2=3 & ps3=1& ps4=3 & cs4=4 then
285   p1=int_first4(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k34,sigma34,age134
      ,%weibullPara,%tranProbPara,%stateIndicator);
286  if ps1=1 & ps2=3 & ps3=1& ps4=3 & cs4=6 then
287   p1=int_first4(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k36,sigma36,age136
      ,%weibullPara,%tranProbPara,%stateIndicator);
288  if ps1=1 & ps2=3 & ps3=2& ps4=1 & cs4=2 then
289   p1=int_first4(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
      ,%weibullPara,%tranProbPara,%stateIndicator);
290  if ps1=1 & ps2=3 & ps3=2& ps4=1 & cs4=3 then
291   p1=int_first4(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
      ,%weibullPara,%tranProbPara,%stateIndicator);
292  if ps1=1 & ps2=3 & ps3=2& ps4=1 & cs4=6 then
293   p1=int_first4(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k16,sigma16,age116
      ,%weibullPara,%tranProbPara,%stateIndicator);
294  if ps1=1 & ps2=3 & ps3=2& ps4=3 & cs4=1 then
295   p1=int_first4(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k31,sigma31,age131
      ,%weibullPara,%tranProbPara,%stateIndicator);
296  if ps1=1 & ps2=3 & ps3=2& ps4=3 & cs4=2 then
297   p1=int_first4(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
      ,%weibullPara,%tranProbPara,%stateIndicator);
298  if ps1=1 & ps2=3 & ps3=2& ps4=3 & cs4=4 then
299   p1=int_first4(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k34,sigma34,age134
      ,%weibullPara,%tranProbPara,%stateIndicator);
300  if ps1=1 & ps2=3 & ps3=2& ps4=3 & cs4=6 then
301   p1=int_first4(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k36,sigma36,age136
      ,%weibullPara,%tranProbPara,%stateIndicator);
302  if ps1=2 & ps2=1 & ps3=2& ps4=1 & cs4=2 then
303   p1=int_first4(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
      ,%weibullPara,%tranProbPara,%stateIndicator);
304  if ps1=2 & ps2=1 & ps3=2& ps4=1 & cs4=3 then
305   p1=int_first4(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
      ,%weibullPara,%tranProbPara,%stateIndicator);
306  if ps1=2 & ps2=1 & ps3=2& ps4=1 & cs4=6 then
307   p1=int_first4(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k16,sigma16,age116
      ,%weibullPara,%tranProbPara,%stateIndicator);
308  if ps1=2 & ps2=1 & ps3=2& ps4=3 & cs4=1 then
309   p1=int_first4(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
      ,%weibullPara,%tranProbPara,%stateIndicator);
310  if ps1=2 & ps2=1 & ps3=2& ps4=3 & cs4=2 then
```

```
311     p1=int_first4(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
            ,%weibullPara,%tranProbPara,%stateIndicator);
312   if ps1=2 & ps2=1 & ps3=2& ps4=3 & cs4=4 then
313     p1=int_first4(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k34,sigma34,age134
            ,%weibullPara,%tranProbPara,%stateIndicator);
314   if ps1=2 & ps2=1 & ps3=2& ps4=3 & cs4=6 then
315     p1=int_first4(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k36,sigma36,age136
            ,%weibullPara,%tranProbPara,%stateIndicator);
316   if ps1=2 & ps2=1 & ps3=3& ps4=1 & cs4=2 then
317     p1=int_first4(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
            ,%weibullPara,%tranProbPara,%stateIndicator);
318   if ps1=2 & ps2=1 & ps3=3& ps4=1 & cs4=3 then
319     p1=int_first4(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
            ,%weibullPara,%tranProbPara,%stateIndicator);
320   if ps1=2 & ps2=1 & ps3=3& ps4=1 & cs4=6 then
321     p1=int_first4(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k16,sigma16,age116
            ,%weibullPara,%tranProbPara,%stateIndicator);
322   if ps1=2 & ps2=1 & ps3=3& ps4=2 & cs4=1 then
323     p1=int_first4(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k21,sigma21,age121
            ,%weibullPara,%tranProbPara,%stateIndicator);
324   if ps1=2 & ps2=1 & ps3=3& ps4=2 & cs4=3 then
325     p1=int_first4(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
            ,%weibullPara,%tranProbPara,%stateIndicator);
326   if ps1=2 & ps2=1 & ps3=3& ps4=2 & cs4=4 then
327     p1=int_first4(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k24,sigma24,age124
            ,%weibullPara,%tranProbPara,%stateIndicator);
328   if ps1=2 & ps2=1 & ps3=3& ps4=2 & cs4=6 then
329     p1=int_first4(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k26,sigma26,age126
            ,%weibullPara,%tranProbPara,%stateIndicator);
330   if ps1=2 & ps2=3 & ps3=1& ps4=2 & cs4=1 then
331     p1=int_first4(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k21,sigma21,age121
            ,%weibullPara,%tranProbPara,%stateIndicator);
332   if ps1=2 & ps2=3 & ps3=1& ps4=2 & cs4=3 then
333     p1=int_first4(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
            ,%weibullPara,%tranProbPara,%stateIndicator);
334   if ps1=2 & ps2=3 & ps3=1& ps4=2 & cs4=4 then
335     p1=int_first4(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k24,sigma24,age124
            ,%weibullPara,%tranProbPara,%stateIndicator);
336   if ps1=2 & ps2=3 & ps3=1& ps4=2 & cs4=6 then
337     p1=int_first4(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k26,sigma26,age126
            ,%weibullPara,%tranProbPara,%stateIndicator);
338   if ps1=2 & ps2=3 & ps3=1& ps4=3 & cs4=1 then
339     p1=int_first4(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131
            ,%weibullPara,%tranProbPara,%stateIndicator);
340   if ps1=2 & ps2=3 & ps3=1& ps4=3 & cs4=2 then
341     p1=int_first4(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
            ,%weibullPara,%tranProbPara,%stateIndicator);
342   if ps1=2 & ps2=3 & ps3=1& ps4=3 & cs4=4 then
343     p1=int_first4(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k34,sigma34,age134
            ,%weibullPara,%tranProbPara,%stateIndicator);
344   if ps1=2 & ps2=3 & ps3=1& ps4=3 & cs4=6 then
345     p1=int_first4(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k36,sigma36,age136
            ,%weibullPara,%tranProbPara,%stateIndicator);
346   if ps1=2 & ps2=3 & ps3=2& ps4=1 & cs4=2 then
347     p1=int_first4(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
            ,%weibullPara,%tranProbPara,%stateIndicator);
348   if ps1=2 & ps2=3 & ps3=2& ps4=1 & cs4=3 then
349     p1=int_first4(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
            ,%weibullPara,%tranProbPara,%stateIndicator);
350   if ps1=2 & ps2=3 & ps3=2& ps4=1 & cs4=6 then
351     p1=int_first4(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k16,sigma16,age116
            ,%weibullPara,%tranProbPara,%stateIndicator);
352   if ps1=2 & ps2=3 & ps3=2& ps4=3 & cs4=1 then
353     p1=int_first4(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k31,sigma31,age131
            ,%weibullPara,%tranProbPara,%stateIndicator);
354   if ps1=2 & ps2=3 & ps3=2& ps4=3 & cs4=2 then
355     p1=int_first4(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
            ,%weibullPara,%tranProbPara,%stateIndicator);
```

```
356  if ps1=2 & ps2=3 & ps3=2& ps4=3 & cs4=4 then
357   p1=int_first4(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k34,sigma34,age134
          ,%weibullPara,%tranProbPara,%stateIndicator);
358  if ps1=2 & ps2=3 & ps3=2& ps4=3 & cs4=6 then
359   p1=int_first4(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k36,sigma36,age136
          ,%weibullPara,%tranProbPara,%stateIndicator);
360  if ps1=3 & ps2=1 & ps3=2& ps4=1 & cs4=2 then
361   p1=int_first4(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
          ,%weibullPara,%tranProbPara,%stateIndicator);
362  if ps1=3 & ps2=1 & ps3=2& ps4=1 & cs4=3 then
363   p1=int_first4(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
          ,%weibullPara,%tranProbPara,%stateIndicator);
364  if ps1=3 & ps2=1 & ps3=2& ps4=1 & cs4=6 then
365   p1=int_first4(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k16,sigma16,age116
          ,%weibullPara,%tranProbPara,%stateIndicator);
366  if ps1=3 & ps2=1 & ps3=2& ps4=3 & cs4=1 then
367   p1=int_first4(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
          ,%weibullPara,%tranProbPara,%stateIndicator);
368  if ps1=3 & ps2=1 & ps3=2& ps4=3 & cs4=2 then
369   p1=int_first4(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
          ,%weibullPara,%tranProbPara,%stateIndicator);
370  if ps1=3 & ps2=1 & ps3=2& ps4=3 & cs4=4 then
371   p1=int_first4(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k34,sigma34,age134
          ,%weibullPara,%tranProbPara,%stateIndicator);
372  if ps1=3 & ps2=1 & ps3=2& ps4=3 & cs4=6 then
373   p1=int_first4(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k36,sigma36,age136
          ,%weibullPara,%tranProbPara,%stateIndicator);
374  if ps1=3 & ps2=1 & ps3=3& ps4=1 & cs4=2 then
375   p1=int_first4(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
          ,%weibullPara,%tranProbPara,%stateIndicator);
376  if ps1=3 & ps2=1 & ps3=3& ps4=1 & cs4=3 then
377   p1=int_first4(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
          ,%weibullPara,%tranProbPara,%stateIndicator);
378  if ps1=3 & ps2=1 & ps3=3& ps4=1 & cs4=6 then
379   p1=int_first4(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k16,sigma16,age116
          ,%weibullPara,%tranProbPara,%stateIndicator);
380  if ps1=3 & ps2=1 & ps3=3& ps4=2 & cs4=1 then
381   p1=int_first4(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k21,sigma21,age121
          ,%weibullPara,%tranProbPara,%stateIndicator);
382  if ps1=3 & ps2=1 & ps3=3& ps4=2 & cs4=3 then
383   p1=int_first4(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
          ,%weibullPara,%tranProbPara,%stateIndicator);
384  if ps1=3 & ps2=1 & ps3=3& ps4=2 & cs4=4 then
385   p1=int_first4(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k24,sigma24,age124
          ,%weibullPara,%tranProbPara,%stateIndicator);
386  if ps1=3 & ps2=1 & ps3=3& ps4=2 & cs4=6 then
387   p1=int_first4(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k26,sigma26,age126
          ,%weibullPara,%tranProbPara,%stateIndicator);
388  if ps1=3 & ps2=2 & ps3=1& ps4=2 & cs4=1 then
389   p1=int_first4(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
          ,%weibullPara,%tranProbPara,%stateIndicator);
390  if ps1=3 & ps2=2 & ps3=1& ps4=2 & cs4=3 then
391   p1=int_first4(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
          ,%weibullPara,%tranProbPara,%stateIndicator);
392  if ps1=3 & ps2=2 & ps3=1& ps4=2 & cs4=4 then
393   p1=int_first4(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k24,sigma24,age124
          ,%weibullPara,%tranProbPara,%stateIndicator);
394  if ps1=3 & ps2=2 & ps3=1& ps4=2 & cs4=6 then
395   p1=int_first4(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k26,sigma26,age126
          ,%weibullPara,%tranProbPara,%stateIndicator);
396  if ps1=3 & ps2=2 & ps3=1& ps4=3 & cs4=1 then
397   p1=int_first4(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k31,sigma31,age131
          ,%weibullPara,%tranProbPara,%stateIndicator);
398  if ps1=3 & ps2=2 & ps3=1& ps4=3 & cs4=2 then
399   p1=int_first4(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
          ,%weibullPara,%tranProbPara,%stateIndicator);
400  if ps1=3 & ps2=2 & ps3=1& ps4=3 & cs4=4 then
401   p1=int_first4(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k34,sigma34,age134
```

```
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=1& ps4=3 & cs4=6 then
   p1=int_first4(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k36,sigma36,age136
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& ps4=1 & cs4=2 then
   p1=int_first4(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& ps4=1 & cs4=3 then
   p1=int_first4(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& ps4=1 & cs4=6 then
   p1=int_first4(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k16,sigma16,age116
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& ps4=2 & cs4=1 then
   p1=int_first4(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k21,sigma21,age121
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& ps4=2 & cs4=3 then
   p1=int_first4(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& ps4=2 & cs4=4 then
   p1=int_first4(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k24,sigma24,age124
        ,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=2 & ps3=3& ps4=2 & cs4=6 then
   p1=int_first4(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k26,sigma26,age126
        ,%weibullPara,%tranProbPara,%stateIndicator);
end;
if (indxi=5) then do;
  if ps1=3 & ps2=1 & ps3=2& ps4=1& ps5=2 & cs5=1 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
        ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=1& ps5=2 & cs5=3 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
        ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=1& ps5=2 & cs5=4 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
        ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=1& ps5=2 & cs5=6 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
        ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=1& ps5=3 & cs5=1 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
        ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=1& ps5=3 & cs5=2 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
        ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=1& ps5=3 & cs5=4 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
        ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=1& ps5=3 & cs5=6 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
        ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=3& ps5=1 & cs5=2 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
        ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=3& ps5=1 & cs5=3 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
        ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=3& ps5=1 & cs5=6 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
        ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=3& ps5=2 & cs5=1 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
        ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=3& ps5=2 & cs5=3 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
        ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
  if ps1=3 & ps2=1 & ps3=2& ps4=3& ps5=2 & cs5=4 then
   p1=int_first5(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
```

```
        ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
448   if ps1=3 & ps2=1 & ps3=2& ps4=3& ps5=2 & cs5=6 then
449    p1=int_first5(n,fsigma31,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
        ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
450   if ps1=3 & ps2=1 & ps3=3& ps4=1& ps5=2 & cs5=1 then
451    p1=int_first5(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
        ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
452   if ps1=3 & ps2=1 & ps3=3& ps4=1& ps5=2 & cs5=3 then
453    p1=int_first5(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
        ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
454   if ps1=3 & ps2=1 & ps3=3& ps4=1& ps5=2 & cs5=4 then
455    p1=int_first5(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
        ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
456   if ps1=3 & ps2=1 & ps3=3& ps4=1& ps5=2 & cs5=6 then
457    p1=int_first5(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
        ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
458   if ps1=3 & ps2=1 & ps3=3& ps4=1& ps5=3 & cs5=1 then
459    p1=int_first5(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
        ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
460   if ps1=3 & ps2=1 & ps3=3& ps4=1& ps5=3 & cs5=2 then
461    p1=int_first5(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
        ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
462 if ps1=3 & ps2=1 & ps3=3& ps4=1& ps5=3 & cs5=4 then
463    p1=int_first5(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
        ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
464   if ps1=3 & ps2=1 & ps3=3& ps4=1& ps5=3 & cs5=6 then
465    p1=int_first5(n,fsigma31,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
        ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
466   if ps1=3 & ps2=1 & ps3=3& ps4=2& ps5=1 & cs5=2 then
467    p1=int_first5(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k21,sigma21,age121
        ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
468   if ps1=3 & ps2=1 & ps3=3& ps4=2& ps5=1 & cs5=3 then
469    p1=int_first5(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k21,sigma21,age121
        ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
470   if ps1=3 & ps2=1 & ps3=3& ps4=2& ps5=1 & cs5=6 then
471    p1=int_first5(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k21,sigma21,age121
        ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
472   if ps1=3 & ps2=1 & ps3=3& ps4=2& ps5=3 & cs5=1 then
473    p1=int_first5(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
        ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
474   if ps1=3 & ps2=1 & ps3=3& ps4=2& ps5=3 & cs5=2 then
475    p1=int_first5(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
        ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
476   if ps1=3 & ps2=1 & ps3=3& ps4=2& ps5=3 & cs5=4 then
477    p1=int_first5(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
        ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
478   if ps1=3 & ps2=1 & ps3=3& ps4=2& ps5=3 & cs5=6 then
479    p1=int_first5(n,fsigma31,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
        ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
480   if ps1=3 & ps2=2 & ps3=1& ps4=2& ps5=1 & cs5=2 then
481    p1=int_first5(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
        ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
482   if ps1=3 & ps2=2 & ps3=1& ps4=2& ps5=1 & cs5=3 then
483    p1=int_first5(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
        ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
484   if ps1=3 & ps2=2 & ps3=1& ps4=2& ps5=1 & cs5=6 then
485    p1=int_first5(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
        ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
486   if ps1=3 & ps2=2 & ps3=1& ps4=2& ps5=3 & cs5=1 then
487    p1=int_first5(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
        ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
488   if ps1=3 & ps2=2 & ps3=1& ps4=2& ps5=3 & cs5=2 then
489    p1=int_first5(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
        ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
490   if ps1=3 & ps2=2 & ps3=1& ps4=2& ps5=3 & cs5=4 then
491    p1=int_first5(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
        ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
492   if ps1=3 & ps2=2 & ps3=1& ps4=2& ps5=3 & cs5=6 then
```

```
493     p1=int_first5(n,fsigma32,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
            ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
494   if ps1=3 & ps2=2 & ps3=1& ps4=3& ps5=1 & cs5=2 then
495     p1=int_first5(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k31,sigma31,age131
            ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
496   if ps1=3 & ps2=2 & ps3=1& ps4=3& ps5=1 & cs5=3 then
497     p1=int_first5(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k31,sigma31,age131
            ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
498   if ps1=3 & ps2=2 & ps3=1& ps4=3& ps5=1 & cs5=6 then
499     p1=int_first5(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k31,sigma31,age131
            ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
500   if ps1=3 & ps2=2 & ps3=1& ps4=3& ps5=2 & cs5=1 then
501     p1=int_first5(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
            ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
502   if ps1=3 & ps2=2 & ps3=1& ps4=3& ps5=2 & cs5=3 then
503     p1=int_first5(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
            ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
504   if ps1=3 & ps2=2 & ps3=1& ps4=3& ps5=2 & cs5=4 then
505     p1=int_first5(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
            ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
506   if ps1=3 & ps2=2 & ps3=1& ps4=3& ps5=2 & cs5=6 then
507     p1=int_first5(n,fsigma32,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
            ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
508   if ps1=3 & ps2=2 & ps3=3& ps4=1& ps5=2 & cs5=1 then
509     p1=int_first5(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
            ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
510   if ps1=3 & ps2=2 & ps3=3& ps4=1& ps5=2 & cs5=3 then
511     p1=int_first5(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
            ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
512   if ps1=3 & ps2=2 & ps3=3& ps4=1& ps5=2 & cs5=4 then
513     p1=int_first5(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
            ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
514   if ps1=3 & ps2=2 & ps3=3& ps4=1& ps5=2 & cs5=6 then
515     p1=int_first5(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
            ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
516   if ps1=3 & ps2=2 & ps3=3& ps4=1& ps5=3 & cs5=1 then
517     p1=int_first5(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
            ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
518   if ps1=3 & ps2=2 & ps3=3& ps4=1& ps5=3 & cs5=2 then
519     p1=int_first5(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
            ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
520   if ps1=3 & ps2=2 & ps3=3& ps4=1& ps5=3 & cs5=6 then
521     p1=int_first5(n,fsigma32,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
            ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
522   if ps1=3 & ps2=2 & ps3=3& ps4=2& ps5=1 & cs5=2 then
523     p1=int_first5(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k21,sigma21,age121
            ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
524   if ps1=3 & ps2=2 & ps3=3& ps4=2& ps5=1 & cs5=3 then
525     p1=int_first5(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k21,sigma21,age121
            ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
526   if ps1=3 & ps2=2 & ps3=3& ps4=2& ps5=1 & cs5=6 then
527     p1=int_first5(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k21,sigma21,age121
            ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
528   if ps1=3 & ps2=2 & ps3=3& ps4=2& ps5=3 & cs5=1 then
529     p1=int_first5(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
            ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
530   if ps1=3 & ps2=2 & ps3=3& ps4=2& ps5=3 & cs5=2 then
531     p1=int_first5(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
            ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
532   if ps1=3 & ps2=2 & ps3=3& ps4=2& ps5=3 & cs5=4 then
533     p1=int_first5(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
            ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
534   if ps1=3 & ps2=2 & ps3=3& ps4=2& ps5=3 & cs5=6 then
535     p1=int_first5(n,fsigma32,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
            ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
536   if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=1 & cs5=2 then
537     p1=int_first5(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
            ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
```

```
538  if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=1 & cs5=3 then
539   p1=int_first5(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
          ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
540  if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=1 & cs5=6 then
541   p1=int_first5(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
          ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
542  if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=3 & cs5=1 then
543   p1=int_first5(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
          ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
544  if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=3 & cs5=2 then
545   p1=int_first5(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
          ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
546  if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=3 & cs5=4 then
547   p1=int_first5(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
          ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
548  if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=3 & cs5=6 then
549   p1=int_first5(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
          ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
550  if ps1=1 & ps2=2 & ps3=1& ps4=3& ps5=1 & cs5=2 then
551   p1=int_first5(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k31,sigma31,age131
          ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
552  if ps1=1 & ps2=2 & ps3=1& ps4=3& ps5=1 & cs5=3 then
553   p1=int_first5(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k31,sigma31,age131
          ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
554  if ps1=1 & ps2=2 & ps3=1& ps4=3& ps5=1 & cs5=6 then
555   p1=int_first5(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k31,sigma31,age131
          ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
556  if ps1=1 & ps2=2 & ps3=1& ps4=3& ps5=2 & cs5=1 then
557   p1=int_first5(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
          ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
558  if ps1=1 & ps2=2 & ps3=1& ps4=3& ps5=2 & cs5=3 then
559   p1=int_first5(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
          ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
560  if ps1=1 & ps2=2 & ps3=1& ps4=3& ps5=2 & cs5=4 then
561   p1=int_first5(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
          ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
562  if ps1=1 & ps2=2 & ps3=1& ps4=3& ps5=2 & cs5=6 then
563   p1=int_first5(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
          ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
564  if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=2 & cs5=1 then
565   p1=int_first5(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
          ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
566  if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=2 & cs5=3 then
567   p1=int_first5(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
          ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
568  if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=2 & cs5=4 then
569   p1=int_first5(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
          ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
570  if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=2 & cs5=6 then
571   p1=int_first5(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
          ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
572  if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=3 & cs5=1 then
573   p1=int_first5(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
          ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
574  if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=3 & cs5=2 then
575   p1=int_first5(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
          ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
576  if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=3 & cs5=4 then
577   p1=int_first5(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
          ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
578  if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=3 & cs5=6 then
579   p1=int_first5(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k13,sigma13,age113
          ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
580  if ps1=1 & ps2=2 & ps3=3& ps4=2& ps5=1 & cs5=2 then
581   p1=int_first5(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k21,sigma21,age121
          ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
582  if ps1=1 & ps2=2 & ps3=3& ps4=2& ps5=1 & cs5=3 then
583   p1=int_first5(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k21,sigma21,age121
```

```
      ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
584  if ps1=1 & ps2=2 & ps3=3& ps4=2& ps5=1 & cs5=6 then
585   p1=int_first5(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k21,sigma21,age121
      ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
586  if ps1=1 & ps2=2 & ps3=3& ps4=2& ps5=3 & cs5=1 then
587   p1=int_first5(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
      ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
588  if ps1=1 & ps2=2 & ps3=3& ps4=2& ps5=3 & cs5=2 then
589   p1=int_first5(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
      ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
590  if ps1=1 & ps2=2 & ps3=3& ps4=2& ps5=3 & cs5=4 then
591   p1=int_first5(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
      ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
592  if ps1=1 & ps2=2 & ps3=3& ps4=2& ps5=3 & cs5=6 then
593   p1=int_first5(n,fsigma12,k23,sigma23,age123,k32,sigma32,age132,k23,sigma23,age123
      ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
594  if ps1=1 & ps2=3 & ps3=1& ps4=2& ps5=1 & cs5=2 then
595   p1=int_first5(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k21,sigma21,age121
      ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
596  if ps1=1 & ps2=3 & ps3=1& ps4=2& ps5=1 & cs5=3 then
597   p1=int_first5(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k21,sigma21,age121
      ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
598  if ps1=1 & ps2=3 & ps3=1& ps4=2& ps5=1 & cs5=6 then
599   p1=int_first5(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k21,sigma21,age121
      ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
600  if ps1=1 & ps2=3 & ps3=1& ps4=2& ps5=3 & cs5=1 then
601   p1=int_first5(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
      ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
602  if ps1=1 & ps2=3 & ps3=1& ps4=2& ps5=3 & cs5=2 then
603   p1=int_first5(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
      ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
604  if ps1=1 & ps2=3 & ps3=1& ps4=2& ps5=3 & cs5=4 then
605   p1=int_first5(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
      ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
606  if ps1=1 & ps2=3 & ps3=1& ps4=2& ps5=3 & cs5=6 then
607   p1=int_first5(n,fsigma13,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
      ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
608  if ps1=1 & ps2=3 & ps3=1& ps4=3& ps5=1 & cs5=2 then
609   p1=int_first5(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131
      ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
610  if ps1=1 & ps2=3 & ps3=1& ps4=3& ps5=1 & cs5=3 then
611   p1=int_first5(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131
      ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
612  if ps1=1 & ps2=3 & ps3=1& ps4=3& ps5=1 & cs5=6 then
613   p1=int_first5(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131
      ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
614  if ps1=1 & ps2=3 & ps3=1& ps4=3& ps5=2 & cs5=1 then
615   p1=int_first5(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
      ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
616  if ps1=1 & ps2=3 & ps3=1& ps4=3& ps5=2 & cs5=3 then
617   p1=int_first5(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
      ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
618  if ps1=1 & ps2=3 & ps3=1& ps4=3& ps5=2 & cs5=4 then
619   p1=int_first5(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
      ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
620  if ps1=1 & ps2=3 & ps3=1& ps4=3& ps5=2 & cs5=6 then
621   p1=int_first5(n,fsigma13,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
      ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
622  if ps1=1 & ps2=3 & ps3=2& ps4=1& ps5=2 & cs5=1 then
623   p1=int_first5(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
      ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
624  if ps1=1 & ps2=3 & ps3=2& ps4=1& ps5=2 & cs5=3 then
625   p1=int_first5(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
      ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
626  if ps1=1 & ps2=3 & ps3=2& ps4=1& ps5=2 & cs5=4 then
627   p1=int_first5(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
      ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
628  if ps1=1 & ps2=3 & ps3=2& ps4=1& ps5=2 & cs5=6 then
```

```
629     p1=int_first5(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
            ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
630   if ps1=1 & ps2=3 & ps3=2& ps4=1& ps5=3 & cs5=1 then
631     p1=int_first5(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
            ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
632   if ps1=1 & ps2=3 & ps3=2& ps4=1& ps5=3 & cs5=2 then
633     p1=int_first5(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
            ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
634   if ps1=1 & ps2=3 & ps3=2& ps4=1& ps5=3 & cs5=4 then
635     p1=int_first5(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
            ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
636   if ps1=1 & ps2=3 & ps3=2& ps4=1& ps5=3 & cs5=6 then
637     p1=int_first5(n,fsigma13,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
            ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
638   if ps1=1 & ps2=3 & ps3=2& ps4=3& ps5=1 & cs5=2 then
639     p1=int_first5(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k31,sigma31,age131
            ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
640   if ps1=1 & ps2=3 & ps3=2& ps4=3& ps5=1 & cs5=3 then
641     p1=int_first5(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k31,sigma31,age131
            ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
642   if ps1=1 & ps2=3 & ps3=2& ps4=3& ps5=1 & cs5=6 then
643     p1=int_first5(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k31,sigma31,age131
            ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
644   if ps1=1 & ps2=3 & ps3=2& ps4=3& ps5=2 & cs5=1 then
645     p1=int_first5(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
            ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
646   if ps1=1 & ps2=3 & ps3=2& ps4=3& ps5=2 & cs5=3 then
647     p1=int_first5(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
            ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
648   if ps1=1 & ps2=3 & ps3=2& ps4=3& ps5=2 & cs5=4 then
649     p1=int_first5(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
            ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
650   if ps1=1 & ps2=3 & ps3=2& ps4=3& ps5=2 & cs5=6 then
651     p1=int_first5(n,fsigma13,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
            ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
652   if ps1=2 & ps2=3 & ps3=1& ps4=2& ps5=1 & cs5=2 then
653     p1=int_first5(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k21,sigma21,age121
            ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
654   if ps1=2 & ps2=3 & ps3=1& ps4=2& ps5=1 & cs5=3 then
655     p1=int_first5(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k21,sigma21,age121
            ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
656   if ps1=2 & ps2=3 & ps3=1& ps4=2& ps5=1 & cs5=6 then
657     p1=int_first5(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k21,sigma21,age121
            ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
658   if ps1=2 & ps2=3 & ps3=1& ps4=2& ps5=3 & cs5=1 then
659     p1=int_first5(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
            ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
660   if ps1=2 & ps2=3 & ps3=1& ps4=2& ps5=3 & cs5=2 then
661     p1=int_first5(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
            ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
662   if ps1=2 & ps2=3 & ps3=1& ps4=2& ps5=3 & cs5=4 then
663     p1=int_first5(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
            ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
664   if ps1=2 & ps2=3 & ps3=1& ps4=2& ps5=3 & cs5=6 then
665     p1=int_first5(n,fsigma23,k31,sigma31,age131,k12,sigma12,age112,k23,sigma23,age123
            ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
666   if ps1=2 & ps2=3 & ps3=1& ps4=3& ps5=1 & cs5=2 then
667     p1=int_first5(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131
            ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
668   if ps1=2 & ps2=3 & ps3=1& ps4=3& ps5=1 & cs5=3 then
669     p1=int_first5(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131
            ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
670   if ps1=2 & ps2=3 & ps3=1& ps4=3& ps5=1 & cs5=6 then
671     p1=int_first5(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131
            ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
672   if ps1=2 & ps2=3 & ps3=1& ps4=3& ps5=2 & cs5=1 then
673     p1=int_first5(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
            ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
```

```
674  if ps1=2 & ps2=3 & ps3=1& ps4=3& ps5=2 & cs5=3 then
675   p1=int_first5(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
          ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
676  if ps1=2 & ps2=3 & ps3=1& ps4=3& ps5=2 & cs5=4 then
677   p1=int_first5(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
          ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
678  if ps1=2 & ps2=3 & ps3=1& ps4=3& ps5=2 & cs5=6 then
679   p1=int_first5(n,fsigma23,k31,sigma31,age131,k13,sigma13,age113,k32,sigma32,age132
          ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
680  if ps1=2 & ps2=3 & ps3=2& ps4=1& ps5=2 & cs5=1 then
681   p1=int_first5(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
          ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
682  if ps1=2 & ps2=3 & ps3=2& ps4=1& ps5=2 & cs5=3 then
683   p1=int_first5(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
          ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
684  if ps1=2 & ps2=3 & ps3=2& ps4=1& ps5=2 & cs5=4 then
685   p1=int_first5(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
          ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
686  if ps1=2 & ps2=3 & ps3=2& ps4=1& ps5=2 & cs5=6 then
687   p1=int_first5(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k12,sigma12,age112
          ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
688  if ps1=2 & ps2=3 & ps3=2& ps4=1& ps5=3 & cs5=1 then
689   p1=int_first5(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
          ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
690  if ps1=2 & ps2=3 & ps3=2& ps4=1& ps5=3 & cs5=2 then
691   p1=int_first5(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
          ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
692  if ps1=2 & ps2=3 & ps3=2& ps4=1& ps5=3 & cs5=4 then
693   p1=int_first5(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
          ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
694  if ps1=2 & ps2=3 & ps3=2& ps4=1& ps5=3 & cs5=6 then
695   p1=int_first5(n,fsigma23,k32,sigma32,age132,k21,sigma21,age121,k13,sigma13,age113
          ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
696  if ps1=2 & ps2=3 & ps3=2& ps4=3& ps5=1 & cs5=2 then
697   p1=int_first5(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k31,sigma31,age131
          ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
698  if ps1=2 & ps2=3 & ps3=2& ps4=3& ps5=1 & cs5=3 then
699   p1=int_first5(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k31,sigma31,age131
          ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
700  if ps1=2 & ps2=3 & ps3=2& ps4=3& ps5=1 & cs5=6 then
701   p1=int_first5(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k31,sigma31,age131
          ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
702  if ps1=2 & ps2=3 & ps3=2& ps4=3& ps5=2 & cs5=1 then
703   p1=int_first5(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
          ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
704  if ps1=2 & ps2=3 & ps3=2& ps4=3& ps5=2 & cs5=3 then
705   p1=int_first5(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
          ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
706  if ps1=2 & ps2=3 & ps3=2& ps4=3& ps5=2 & cs5=4 then
707   p1=int_first5(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
          ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
708  if ps1=2 & ps2=3 & ps3=2& ps4=3& ps5=2 & cs5=6 then
709   p1=int_first5(n,fsigma23,k32,sigma32,age132,k23,sigma23,age123,k32,sigma32,age132
          ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
710  if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=2 & cs5=1 then
711   p1=int_first5(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
          ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
712  if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=2 & cs5=3 then
713   p1=int_first5(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
          ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
714  if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=2 & cs5=4 then
715   p1=int_first5(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
          ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
716  if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=2 & cs5=6 then
717   p1=int_first5(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
          ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
718  if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=3 & cs5=1 then
719   p1=int_first5(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
```

```
      ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
720   if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=3 & cs5=2 then
721    p1=int_first5(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
      ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
722   if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=3 & cs5=4 then
723    p1=int_first5(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
      ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
724   if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=3 & cs5=6 then
725    p1=int_first5(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
      ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
726   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=1 & cs5=2 then
727    p1=int_first5(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
      ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
728   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=1 & cs5=3 then
729    p1=int_first5(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
      ,k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
730   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=1 & cs5=6 then
731    p1=int_first5(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
      ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
732   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=2 & cs5=1 then
733    p1=int_first5(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
      ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
734   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=2 & cs5=3 then
735    p1=int_first5(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
      ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
736   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=2 & cs5=4 then
737    p1=int_first5(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
      ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
738   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=2 & cs5=6 then
739    p1=int_first5(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k32,sigma32,age132
      ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
740   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=2 & cs5=1 then
741    p1=int_first5(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
      ,k21,sigma21,age121,%weibullPara,%tranProbPara,%stateIndicator);
742   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=2 & cs5=3 then
743    p1=int_first5(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
      ,k23,sigma23,age123,%weibullPara,%tranProbPara,%stateIndicator);
744   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=2 & cs5=4 then
745    p1=int_first5(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
      ,k24,sigma24,age124,%weibullPara,%tranProbPara,%stateIndicator);
746   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=2 & cs5=6 then
747    p1=int_first5(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
      ,k26,sigma26,age126,%weibullPara,%tranProbPara,%stateIndicator);
748   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=3 & cs5=1 then
749    p1=int_first5(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
      ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
750   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=3 & cs5=2 then
751    p1=int_first5(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
      ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
752   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=3 & cs5=4 then
753
754  p1=int_first5(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113,
       k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
755   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=3 & cs5=6 then
756    p1=int_first5(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k13,sigma13,age113
      ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
757   if ps1=2 & ps2=1 & ps3=3& ps4=2& ps5=1 & cs5=2 then
758    p1=int_first5(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k21,sigma21,age121
      ,k12,sigma12,age112,%weibullPara,%tranProbPara,%stateIndicator);
759   if ps1=2 & ps2=1 & ps3=3& ps4=2& ps5=1 & cs5=3 then
760
761  p1=int_first5(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k21,sigma21,age121,
       k13,sigma13,age113,%weibullPara,%tranProbPara,%stateIndicator);
762   if ps1=2 & ps2=1 & ps3=3& ps4=2& ps5=1 & cs5=6 then
763    p1=int_first5(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k21,sigma21,age121
      ,k16,sigma16,age116,%weibullPara,%tranProbPara,%stateIndicator);
764   if ps1=2 & ps2=1 & ps3=3& ps4=2& ps5=3 & cs5=1 then
765    p1=int_first5(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
```

```
        ,k31,sigma31,age131,%weibullPara,%tranProbPara,%stateIndicator);
766   if ps1=2 & ps2=1 & ps3=3& ps4=2& ps5=3 & cs5=2 then
767    p1=int_first5(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
        ,k32,sigma32,age132,%weibullPara,%tranProbPara,%stateIndicator);
768   if ps1=2 & ps2=1 & ps3=3& ps4=2& ps5=3 & cs5=4 then
769    p1=int_first5(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
        ,k34,sigma34,age134,%weibullPara,%tranProbPara,%stateIndicator);
770   if ps1=2 & ps2=1 & ps3=3& ps4=2& ps5=3 & cs5=6 then
771    p1=int_first5(n,fsigma21,k13,sigma13,age113,k32,sigma32,age132,k23,sigma23,age123
        ,k36,sigma36,age136,%weibullPara,%tranProbPara,%stateIndicator);
772 end;
773 if (indxi=6) then do;
774   if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=1 & ps6=2& cs6=3 then
775    p1=int_first6(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
        ,k12,sigma12,age112,k23,sigma23,age123,%weibullPara,%tranProbPara,%
        stateIndicator);
776   if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=1 & ps6=3& cs6=2 then
777    p1=int_first6(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
        ,k13,sigma13,age113,k32,sigma32,age132,%weibullPara,%tranProbPara,%
        stateIndicator);
778   if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=2 & ps6=1& cs6=2 then
779    p1=int_first6(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
        ,k21,sigma21,age121,k12,sigma12,age112,%weibullPara,%tranProbPara,%
        stateIndicator);
780   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=1 & ps6=3& cs6=6 then
781    p1=int_first6(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
        ,k13,sigma13,age113,k36,sigma36,age136,%weibullPara,%tranProbPara,%
        stateIndicator);
782   if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=2 & ps6=1& cs6=2 then
783    p1=int_first6(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
        ,k21,sigma21,age121,k12,sigma12,age112,%weibullPara,%tranProbPara,%
        stateIndicator);
784   if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=3 & ps6=2& cs6=4 then
785    p1=int_first6(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k13,sigma13,age113
        ,k32,sigma32,age132,k24,sigma24,age124,%weibullPara,%tranProbPara,%
        stateIndicator);
786   if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=2 & ps6=3& cs6=6 then
787    p1=int_first6(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
        ,k23,sigma23,age123,k36,sigma36,age136,%weibullPara,%tranProbPara,%
        stateIndicator);
788   if ps1=2 & ps2=1 & ps3=2& ps4=1& ps5=2 & ps6=1& cs6=2 then
789    p1=int_first6(n,fsigma21,k12,sigma12,age112,k21,sigma21,age121,k12,sigma12,age112
        ,k21,sigma21,age121,k12,sigma12,age112,%weibullPara,%tranProbPara,%
        stateIndicator);
790   if ps1=2 & ps2=1 & ps3=2& ps4=3& ps5=1 & ps6=3& cs6=6 then
791    p1=int_first6(n,fsigma21,k12,sigma12,age112,k23,sigma23,age123,k31,sigma31,age131
        ,k13,sigma13,age113,k36,sigma36,age136,%weibullPara,%tranProbPara,%
        stateIndicator);
792 end;
793   if (indxi=7) then
794    do;
795   if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=1 & ps6=2& ps7=3& cs7=6 then
796    p1=int_first7(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121
        ,k12,sigma12,age112,k23,sigma23,age123,k36,sigma36,age136,%weibullPara,%
        tranProbPara,%stateIndicator);
797   if ps1=1 & ps2=2 & ps3=3& ps4=1& ps5=2 & ps6=1& ps7=2& cs7=1 then
798    p1=int_first7(n,fsigma12,k23,sigma23,age123,k31,sigma31,age131,k12,sigma12,age112
        ,k21,sigma21,age121,k12,sigma12,age112,k21,sigma21,age121,%weibullPara,%
        tranProbPara,%stateIndicator);
799   if ps1=1 & ps2=2 & ps3=1& ps4=3& ps5=2 & ps6=1& ps7=3& cs7=6 then
800    p1=int_first7(n,fsigma12,k21,sigma21,age121,k13,sigma13,age113,k32,sigma32,age132
        ,k21,sigma21,age121,k13,sigma13,age113,k36,sigma36,age136,%weibullPara,%
        tranProbPara,%stateIndicator);
801   if ps1=2 & ps2=1 & ps3=3& ps4=1& ps5=2 & ps6=1& ps7=3& cs7=4 then
802    p1=int_first7(n,fsigma21,k13,sigma13,age113,k31,sigma31,age131,k12,sigma12,age112
        ,k21,sigma21,age121,k13,sigma13,age113,k34,sigma34,age134,%weibullPara,%
        tranProbPara,%stateIndicator);
803 end;
```

```
804 if (indxi=8) then do;
805   if ps1=1 & ps2=2 & ps3=1& ps4=2& ps5=3 & ps6=1& ps7=3& ps8=1& cs8=6 then
806    p1=int_first8(n,fsigma12,k21,sigma21,age121,k12,sigma12,age112,k23,sigma23,age123
          ,k31,sigma31,age131,k13,sigma13,age113,k31,sigma31,age131,k16,sigma16,age116
          ,%weibullPara,%tranProbPara,%stateIndicator);
807 end;
808 if p1=0 then p1=1;
809 ll=log(p1);
810 model id~ general (ll);
811 run;
812 %mend;
```

## Bibliography


[1] Anderson, P.K., Borgan, O., Gill, R.D., Keiding, N., 1993. Statistical models based on the counting process. *Spring*, New York.

[2] Albert, P.S., Follmann, D.A., 2003. A random effects transition model for longitudinal binary data with informative missingness. *Statistica Neerlandica* 57, 100-111.

[3] Benichou, J., Gail, M.H., 1990. Estimates of absolute cause-specific risk in cohort studies. *Biometrics* 46(3), 813-816.

[4] Betensky, R.A., Lindsey, J.C., Ryan, L.M., Wand, M.P., 1999. Local EM sstimation of the hazard function for interval-censored data. *Biometrics* 55(1), 238-245.

[5] Betensky, R.A., Lindsey, J.C., Ryan, L.M., Wand, M.P., 2002. A local likelihood proportional hazards model for interval censored data. *Statistics in Medicine* 21(2), 263-275.

[6] Bhat, U.N., Miller, G.K., 2002. Elements of Applied Stochastic Processes, 3rd ed. *Wiley*, New York.

[7] Chandra, R.B., 2001. Quasi-random maximum simulated likelihood estimation of the mixed multinomial logit model. *Transportation Research* 35B, 677-693.

[8] Cheng, S. C., Fine, J.P., Wei, L. J., 1998. Prediction of cumulative incidence function under the proportional hazards model. *Biometrics* 54(1), 219-228.

[9] Commenges, D., 2002. Inference for multi-state models from interval censored data. *Statistical Methods in Medical Research* 2002; 11: 167-182.

[10] Cox, D.R., Oakes, D., 1984. Analysis of Survival Data. *Chapman & Hall*, London.

[11] Efron, B., 1981. Nonparametric estimates of standard error, the jackknife, the bootstrap, and other methods. *Biometrika* 68, 589-599.

[12] Efron, B., Tibshirani,R.J., 1984. An Introduction to the Bootstrap. *Chapman and Hall/CRC* Financial Mathematics Series.

[13] Elashoff, R.M., Li, G., Li, N., 2007. An approach to joint analysis of longitudinal measurements and competing risks failure time data. *Statistics In Medicine* 26, 2813-2835.

[14] Foucher, Y., Giral, M., Soulillou, J.P., Daures, J.P., 2007. A semi-Markov model for multistate and interval-censored data with multiple terminal events. Application in renal transplantation. *Statistics in Medicine* 26(30), 5381-5393.

[15] Foucher, Y., Giral, M., Soulillou, J.P., Daures, J.P., 2010. A flexible semi-Markov model for interval-censored data and goodness-of-fit testing. *Statistical Methods in Medical Research* 19(2), 127-145.

[16] Foucher, Y., Mathieu, E., Saint-Pierre, P., Durand, J.F., Daures, J.P., 2005. A semi-Markov model based on Generalized Weibull distribution with an illustration for HIV disease. *Biometrical Journal* 47, 825-833.

[17] Frydman, H., Liu, J., 2013. Nonparametric estimation of the cumulative intensities in an interval censored competing risks model. *Lifetime Data Analysis* 19(1), 79-99.

[18] Haan, P., Uhlendorff, A., 2006. Markov chains with measurement error: estimating the 'true' course of a marker of the progression of human immunodeficiency virus disease. *The Stata Journal* 6(2), 229-245.

[19] Harezlak, J., Gao, S., Hui, S.L., 2003. An illness-death stochastic model in the analysis of longitudinal dementia data. *Statistics in medicine* 22(9), 1465-1475.

[20] Healy, B., Degruttola, V., 2007. Hidden Markov models for settings with interval-censored transition times and uncertain time origin: application to HIV genetic analyses. *Biostatistics* 8(2), 438-452.

[21] Henderson, R., Diggle, P., Dobson, A., 2000. Joint modelling of longitudinal measurements and event time data. *Biostatistics* 1(4), 465-480.

[22] Hinchliffe, S.R., Lambert, P.C., 2013. Flexible parametric modeling of cause-specific hazards to estimate cumulative incidence functions. *BMC Medical Research Methodology* 13:13.

[23] Hufthammer, K., Tjstheim, D., 2008. Local gaussian likelihood and local gaussian correlation. *Doctoral Thesis*, Chapter 3, University of Bergen.

[24] Huzurbazar, A.V., 2005. Flowgraph models for multistate time to event data. *Wiley*, New York.

[25] Joly, P., Commenges, D., 1999. A penalized likelihood approach for a progressive three-state model with censored and truncated data: Application to AIDS. *Biometrics* 55, 887-890.

[26] Joly, P., Commenges, D., Helmer, C., Letenneur, L., 2002. A penalized likelihood approach for an illness-death model with interval-censored data: application to age-specific incidence of dementia. *Biostatistics* 3(3), 433-443.

[27] Kalbfleisch, J.D., Lawless, J.F., 1985. The analysis of panel data under a Markov assumption. *Journal of the American Statistical Association* 80, 863-871.

[28] Kalbfleisch, J.D., Prentice, R.L., 2002. The Statistical Analysis of Failure Time Data, Second Edition. *Wiley Series in Probability and Statistics* .

[29] Kang, M., Lagakos, S.W., 2007. Statistical methods for panel data from a semi-markov process, with application to HPV. *Biostatistics* 8(2), 252-264.

[30] Kapetanakis, V., Matthews, F.E., van den Hout, A., 2012. A semi-Markov model for stroke with piecewise-constant hazards in the presence of left, right and interval censoring. *Statistics in Medicine* 32(4), 697-713.

[31] Keiding, M., Klein, J.P., Horowitz, M.M., 2001. Multi-state models and outcome prediction in bone marrow transplantation. *Statistics in Medicine* 20, 1871-1885.

[32] Klein, J.P., 2006. Modeling competing risks in cancer studies. *Statistics in Medicine* 25(6), 1015-1034.

[33] Klein, J.P., Andersen, K.P., 2005. Regression modeling of competing risks data based on pseudovalues of the cumulative incidence function. *Biometrics* 61(1), 223-229.

[34] Klein, J.P., Rizzo, J.D., Zhang, M.J., Keiding, N., 2001. Statistical methods for the analysis and presentation of the results of bone marrow transplants. Part 2: Regression modeling. *Bone Marrow Transplants* 28(11), 1001-1011.

[35] Kryscio, R.J., Abner, E.L., Lin, Y., Cooper, G.E., Fardo, D.W., Jicha, G.A., Nelson, P.T., Smith, C.D., Van Eldik, L.J., Wan, L., Schmitt, F.A., 2013. Adjusting for mortality when identifying risk factors for transitions to mild cognitive impairment and dementia. *Journal of Alzheimer's Disease* 35(4), 823-832.

[36] Latouche, A., Allignol, A., Beyersmann, J., Labopin, M., Fine, J.P., 2013. A competing risks analysis should report results on all cause-specific hazards and cumulative incidence functions. *Journal of Clinical Epidemiology* 66(6), 648-653.

[37] Lin, D.Y., 1997. Non-parametric inference for cumulative incidence functions in competing risk studies. *Statistics in Medicine* 16, 901-910.

[38] Listwon, A., Saint-Pierre, P., 2013. SemiMarkov: An R Package for Parametric Estimation in Multi-State Semi-Markov Models. Version 1. URL http://cran.r project.org/web/packages/SemiMarkov/.

[39] Mathieu, E., Foucher, Y., Dellamonica, P., Daures, J.P., 2007. Parametric and Non Homogeneous Semi-Markov Process for HIV Control. *Methodology and Computing in Applied Probability*, 9(3), 389-397.

[40] Molnar, F.J., Hutton, B., Fergusson, D., 2008. Does analysis using 'last observation carried forward' introduce bias in dementia research? *CMAJ* 179(8), 751−753.

[41] Mudoholkar, G., Srivastava, D., Kollia, G. D., 1996. A Generalization of the Weibull distribution with application. *Journal of the American Statistical Association* 91, 1575-1583.

[42] Nelson, P.T., Schmitt, F.A., Lin,Y., Abner, E.L., Jicha, G.A., Patel, E., Thomason, P.C., Neltner, KJ.H., Smith C.D., Santacruz,K.S., Sonnen, J.A., Poon, L.W., Gearing, M., Green, R.C., Woodard, J.L., VanEldik, L.J., Kryscio R.J., 2011. Hippocampal sclerosis in advanced age: clinical and pathological features. *Brain* 134: 1506-1518.

[43] Rabe-Hesketh, Skrondal, A., 2002. Reliable estimation of generalized linear mixed models using adaptive quadrature. *The Stata Journal* 2(1), 1-21.

[44] Raudenbush, S.W., Yang, M., Yosef, M., 2000. Maximum Likelihood for Generalized Linear Models with Nested Random Effects via High-Order, Multivariate Laplace Approximation. *Journal of Computational and Graphical Statistics* 9, 141-157.

[45] Salazar, J.C., Schmitt, F.A., Yu, L., Mendiondo, M.M., Kryscio, R.J., 2007. Shared random effects analysis of multi-state Markov models: application to a longitudinal study of transitions to dementia. *Statistics in Medicine* 26, 568-580.

[46] Satten, G.A., Longini, I.M., 1996. Markov chains with measurement error: estimating the 'true' course of a marker of the progression of human immunodeficiency virus disease. *Applied Statistics* 45, 275-309.

[47] Satten, G.A., Sternberg, M., 1999. Fitting semi-Markov models to interval-censored data with unknown initiation times. *Biometrics* 55(2), 507-513.

[48] Scheike, T.H., Zhang, M., Gerds, T.A., 2008. Predicting cumulative incidence probability by direct binomial regression. *Biometrika* 95 (1), 205-220.

[49] Shen, Y., Cheng, S.C., 1999. Confidence bands for cumulative incidence function under the additive risk model. *Biometrics* 55, 1093-1100.

[50] Turnbulla, B.W., 1974. Nonparametric estimation of a survivorship function with doubly censored data. *Journal of the American Statistical Association* 69,169-173.

[51] Tyas, S.L., Salazar, J.C., Snowdon, D., Derosiers, M.F., Riley, K.P., Mendiondo, M.S., Kryscio, R.J., 2007. Transitions to mild cognitive impairments, dementia, and death: finding from the Nun Study. *American Journal of Epidemiology* 165, 1231-1238.

[52] Tyas, S.L., Snowdon, D.A., Desrosiers, M.F., Riley, K.P., Markesbery, W.l., 2007. Healthy ageing in the Nun Study: Definition and neuropathologic correlates. *Age and Ageing* 36(6), 650-655.

[53] Titman, A.C., Sharples, L.D., 2010. Semi-Markov models with phase-type sojourn distributions. *Biometrics* 66, 742-752.

[54] Wu, L. L., Tuma, N.,B., 1990. Local hazard models. *Sociological Methodology* 20, 141-180.

[55] Xu, J., Zeger, S.L., 2001. Joint analysis of longitudinal data comprising repeated measures and times to events. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 50, 375−387.

[56] Yu, L., Griffith, W.S., Tyas, S.L., Snowdon, D.A., Kryscio, R.J., 2010. A non-stationary Markov transition model for computing the relative risk of dementia before death. *Statistics in Medicine* 29, 639−648.

[57] Yu, L., Tyas, S.L., Snowdon, D., Kryscio, R.J., 2009. Effects of ignoring baseline on modeling transitions from intact cognition to dementia. *Computational Statistics and Data Analysis* 53, 3334−3343.

## Vita

Shaoceng Wei

### Education

**M.S. in Statistics** Nankai University 2007-2009
**B.S. in Statistics** East China Normal University 2003-2007

### Employment

Teaching Assistant 2009, 2012-2014
Statistical Department, University of Kentucky

Research Assistant Jan 2011-Aug 2012
Statistical Department, University of Kentucky

Summer Internship Summer 2011 & 2013
Energy and Environment Cabinet, Frankfort, Ky

### Publication & Manuscript

**Wei, S.**, Kryscio, R.J., 2014. Semi-Markov models for interval censored transient cognitive states with back transitions and a competing risk. Accepted by *Statistical Methods in Medical Research.*

**Wei, S.**, Xu, L., Kryscio, R.J., 2014. Markov transition model to dementia with death as a competing event. *Computational Statistics & Data Analysis* 80, 78-88.