Continuous - Time Survival Analysis in Mplus

Tihomir Asparouhov and Bengt Muthén

Version 3

June 29, 2018

1 Overview

Here we will describe the basic continuous time survival model implemented in Mplus and will provide some details on the basic modeling options that are available. Introduction to continuous time survival modeling can be found in Singer & Willett (2003), Hougaard (2000) or Klein & Moeschberger (1997). Survival analysis: techniques for. The survival models implemented in Mplus includes many extensions of this basic model such as mixture survival models, survival models with random effects (frailty models), multilevel survival models, time varying covariate models, competing risk models, non-proportional hazard models etc. Describing the details of these models is beyond the scope of this document. In most cases however the material presented here applies to these extensions as well. More details on the models and algorithms implemented in Mplus can be found in Larsen (2004, 2005) and Asparouhov, Masyn & Muthén (2006). Practical applications of the Mplus methodology for continuous time survival modeling can be found in Muthén et al. (2009).

Let the variable T_0 be a time-to-event variable such as time to death for example. Let C be the time when the individual leaves the target cohort due to death or other types of censoring such as lost to follow up etc. The survival variable T and the censoring indicator δ are defined by

$$T = \min\{T_0, C\} \tag{1}$$

$$\delta = \begin{cases} 1 & \text{if } T_0 > C \\ 0 & \text{if } T_0 \leq C \end{cases} . \tag{2}$$

Both variables T and δ have to be constructed and used in the survival analysis in Mplus. The T variable is specified via the *survival=* command while the δ variable is specified via the *timecensored=* command. Details on the specification options can be found in Muthén & Muthén (2012). Let X be an observed predictor of T.

2 The Proportional Hazard Model

The proportional hazard (PH) model specifies that the hazard function is proportional to the baseline hazard function, i.e.,

$$h(t) = h_0(t)Exp(\beta X) \tag{3}$$

where $h(t)$ is the hazard function and $h_0(t)$ is the baseline hazard function at time t. Two proportional hazard models are implemented in Mplus. One of the models assumes a completely non-parametric shape for the baseline hazard function. This model is known as the Cox regression model. The other model is based on a parametric model for the baseline hazard function. This model is known as the parametric PH model. The general parametric model for the baseline hazard function in Mplus is a step function with arbitrary number of steps, however through parameter constraints this parametric model can serve as an approximation to any other parametric model, including models such as Exponential, Weibull and Gompertz models. This approximation is based on the fact that any continuous function can be closely approximated by a step function. Note also that because of the parameter constraints the number of parameters that are freely estimated in the approximation model will remain the same, see Section 7 for a detailed example. First we will describe the parametric PH model implementation.

2.1 Parametric PH model

To estimate $h_0(t)$ as a step function with L intervals the survival variable is declared as *survival=T (L interval lengths)*. For example if *survival=T(2*1 2)*, the length of the intervals in the baseline hazard step function are 1, 1, 2 and ∞ in that order, i.e., the intervals in the definition of the baseline hazard function are $[0, 1), [1, 2), [2, 4), [4, \infty)$ over which we assume that the baseline

hazard is constant

$$h_0(t) = \begin{cases} h_1 & \text{if } 0 < t \leq 1 \\ h_2 & \text{if } 1 < t \leq 2 \\ h_3 & \text{if } 2 < t \leq 4 \\ h_4 & \text{if } 4 < t < \infty \end{cases} . \quad (4)$$

The analysis command option *basehazard* determines how the parameters h_1, h_2, ..., h_L are treated. If *basehazard=on* these parameters are estimated as regular parameters. Thus standard error will be computed for the baseline hazard function. Such standard errors can be used to obtain standard errors for the survival rates for example. They are also included as parameters in the model and can be held equal across class for example or they can be used in model constraint to impose certain parametric shape. Starting values can be given for these parameters and these starting values can be perturbed just as for other parameters. Acceptable starting values are between 0 and ∞, i.e., negative values are not acceptable baseline hazard function values. The *basehazard=on* option should be used if relatively few steps are used (small L) or there are enough restrictions in the model to compensate for a large number of steps. If L is large however, even with many restrictions on the h_i parameters, it may be difficult to estimate the model. The more parameters are in the model the more difficult the maximization will be, i.e., the estimation will be very computationally demanding. In addition to that the asymptotic approximation used with MLE requires larger sample size for models with larger number of parameters. Both the parameter estimates and the standard errors may have larger biases for models with larger number of parameters. These undesired effects can be avoided by specifying *basehazard=off* and in that case the parameters h_1, h_2, ..., h_L are treated as nuisance parameters. The profile likelihood is formed by explicitly maximizing the full likelihood over these parameters. The profile likelihood is then treated as regular maximum likelihood, see Murphy and van der Vaart (2000). Standard errors are not computed for the baseline hazard function, however the values of the nuisance parameters can be obtained by including *basehazard* option in the output command. In mixture models Mplus will estimate class varying baseline hazard and thus the mean of the survival variable will be unidentified. With *basehazard=off* the estimation will typically be less computationally demanding.

2.2 Cox Regression Model

There are a number of different methods for estimating this model. The method that Mplus uses is based on PH parametric model estimation described in the previous section. To obtain a fully non-parametric baseline hazard function we just need to select sufficiently detailed step function estimation. This can be accomplished for example by settings such as *survival=T(500*0.02)* or *survival=T(1000*0.01)* if the T value ranges between 0 and 10. The exact specification of the step size typically will have a minimal effect on the estimates. The step size however affects the log-likelihood value. It is important that when LRT is conducted between two models the step function framework is the same. Mplus also implements an automatic option, *survival=T(all)*, which will construct the step intervals from the data, by making the steps as detailed as needed. With this option Mplus estimates a baseline hazard step function which is constant between every two consecutive event times. If all event times, including censored observations are $t_1 < t_2 < .... < t_n$ then Mplus estimates

$$h_0(t) = \begin{cases} h_1 & \text{if } 0 < t \leq t_1 \\ h_2 & \text{if } t_1 < t \leq t_2 \\ ... & \\ h_{n+1} & \text{if } t_n < t < \infty \end{cases} . \tag{5}$$

Equal event times are treated as one event time. There is a direct relation between the *survival=T(all)* specification of Cox regression and the *survival=T(M * h)* specification with M large and h small. If h is smaller than the distance between any two distinct event times and Mh is greater than the biggest event time in the data, the parameter estimates and their standard errors will be the same. When estimating the Cox regression model the parameters h_i should be estimated as nuisance (unrestricted) parameters, i.e., with the settings *basehazard=off*. It is possible to estimate Cox regression with *basehazard=on* however this combination should be used with great care as the number of parameters may be too large.

For a discussion on the different ways to estimate the Cox regression model and the equivalence of the profile likelihood and the traditional partial likelihood methods see Clayton (1988).

3 The Cumulative Baseline Hazard Function

Suppose that the baseline hazard function is

$$h_0(t) = \begin{cases} h_1 & \text{if } t_0 = 0 < t \leq t_1 \\ h_2 & \text{if } t_1 < t \leq t_2 \\ ... & \\ h_{L+1} & \text{if } t_L < t < \infty \end{cases} . \tag{6}$$

The cumulative baseline hazard function at time t represents the total hazard an individual is exposed to up to time t. If $t_k < t < t_{k+1}$ the cumulative baseline hazard function is

$$H_0(t) = \int_0^t h_0(x)dx = \sum_{i=1}^{k-1} h_i(t_i - t_{i-1}) + h_k(t - t_{k-1}) \tag{7}$$

4 The Survival Function

The survival function is the probability that the survival variable T is greater than t

$$S(t) = P(T > t) = Exp(-Exp(\beta X)H_0(t)). \tag{8}$$

The survival function complements the distribution function

$$F(t) = P(T \leq t) = 1 - S(t). \tag{9}$$

5 The Likelihood Function

The likelihood function of the survival variable T is

$$L(T) = (h_0(T)Exp(\beta X))^{(1-\delta)}S(T) \tag{10}$$

where δ is the censoring variable.

6 Survival Variable in Monte Carlo Simulations

Survival variables can be used with Mplus simulation facilities. The step sizes m_1 ... m_L used for the generation process are specified in *generate = T (s* m_1

... m_L). The values of the baseline hazard function are specified in the Model Population section. These parameters are referred as $T\#1, ..., T\#L+1$ and should be specified in the Model Population section regardless of whether or not they are available in the Model section. These parameters are available in the Model section if the option *basehazard=ON*, however they are always available and should be specified in the Model Population section.

The simplest example for a survival variable specification is *generate=T(s).* With this specification $L = 0$. The hazard function has a single step, which is the infinite interval $[0, \infty)$. In this case only one baseline hazard function value $T\#1$ has to be specified in the Model Population section. Thus the specification $T\#1 * \lambda$ defines a survival variable with constant hazard function λ over the entire $[0, \infty)$ interval. Suppose that there are no X variables in the model. In this case equation (3) says that $h(t) = h_0(t) = \lambda$. Equation (7) reduces to $H_0(t) = t\lambda$. Equations (8) and (9) then give us the distribution function of T

$$F(t) = P(T \leq t) = 1 - e^{-t\lambda}, \tag{11}$$

i.e., T is exponentially distributed with density function $\lambda e^{-t\lambda}$, mean $1/\lambda$ and variance $1/\lambda^2$, where $0 < \lambda < \infty$. This implies that the smaller the λ, the longer the survival time. Such considerations can be used for selecting proper values in the simulation study. For example if a predominant range of T between 0 and 30 is desired then the hazard should be set to $T\#1 * 0.1$. Using the distribution function of T in this case we get

$$P(0 < T < 10) = 1 - e^{-1} \approx 63\%$$

$$P(10 < T < 20) = e^{-1} - e^{-2} \approx 23\%$$

$$P(20 < T < 30) = e^{-2} - e^{-3} \approx 9\%$$

$$P(30 < T) = e^{-3} \approx 5\%.$$

7 Right Censoring of Survival Variables in Monte Carlo Simulations

The command *gentcensoring = T (λ_1)* specifies that the hazard for the censoring process is λ_1, i.e., an exponential variable C with mean $1/\lambda_1$ is generated as well as the uncensored survival variable T_0 following the survival

variable specification. Censoring occurs if $C < T_0$. In that case we set $T = C$ and the censoring indicator δ to 1, i.e.,

$$T = \min\{T_0, C\} \tag{12}$$

$$\delta = \begin{cases} 1 & \text{if } T_0 > C \\ 0 & \text{if } T_0 \leq C \end{cases} . \tag{13}$$

Suppose that *gentcensoring = T (λ_1)* and the baseline hazard function is set to λ by setting *generate = T (s)* and within Model Population $T\#1 * \lambda$. Then T_0 and C are independent exponentially distributed random variables with distribution $1 - e^{-\lambda t}$ and $1 - e^{-\lambda_1 t}$ respectively. In this case the variable T is also exponentially distributed with distribution function $1 - e^{-(\lambda+\lambda_1)t}$ because

$$P(T_0 > t) = e^{-\lambda t} \tag{14}$$

$$P(C > t) = e^{-\lambda_1 t} \tag{15}$$

$$P(T > t) = P(T_0 > t)P(C > t) = e^{-(\lambda+\lambda_1)t} \tag{16}$$

If $\lambda = \lambda_1$ about 50% of the observations will be censored because T_0 and C would be identically distributed and the two variables are equally likely to be the smallest.

8 Weibull PH Model Specification

The Weibull model assumes that the baseline hazard function is

$$h_0(t) = \lambda s(\lambda t)^{s-1}, \tag{17}$$

for some parameters λ and s, see Bradburn et alt. (2003). Below we describe how to set up an approximation for this model via the Mplus step function baseline model. The precision of the approximation depends on the number of intervals L used in the baseline step function. The more intervals are used the better the approximation. Typically however $L = 50$ will be sufficient. Suppose that most of the T values range from 0 to 5. We can split this range into equal intervals of length 0.1 and specify the baseline step function estimation by turning the option *basehazard=ON* and by setting *survival=T(50*0.1)*. With this setup however the baseline function will assume an unrestricted shape. We can add labels for the basehazard parameters by adding this line to the model $[T\#1 - T\#50](p1 - p50)$. Using these

labels we specify the Weibull shape by adding the following *Model Constraint* section to the input file:

model constraint:
new (s lambda);
*p1=lambda*s*(lambda*0.05)**(s-1);*
*p2=lambda*s*(lambda*0.15)**(s-1);*
...
*p50=lambda*s*(lambda*4.95)**(s-1);*

The value for t has been substituted with the midpoint for each of the time intervals. Take for example the first interval $[0, 0.1]$. To make the approximation as close as possible we use the midpoint of this interval 0.05 and we substitute that for t in equation (17). Better approximation can be accomplished by specifying smaller intervals for more dense time segments and larger intervals for time segments with fewer events. For example *survival=T(20*0.05 40*0.1)* will lead to better approximation if for many individuals $T < 1$. The LRT test can be used to test the model constraint equations, i.e., to test the assumption of Weibull baseline hazard.

9 Types of survival variables in Mplus

Starting with Mplus version 7.2 there are four types of survival variables, non-parametric, semi-parametric, parametric and constant-hazard. These survival variables are specified differently and depending on the model the Mplus program will use an optimal default choice for the modeling type. If the survival variable is T, one can obtain the Mplus default setting simply by specifying the survival variable by name *survival=T*. Note that prior to Mplus version 7.2 that command was used to specify constant hazard survival variable.

The second important option is the *BASEHAZARD* option. If this option is turned ON then the baseline hazard parameters are treated as actual model parameter. If this option is turned OFF then the baseline hazard parameters are treated as auxiliary parameter.

The four different types of survival variables are specified as follows.

9.1 Non-parametric survival

The non-parametric survival variable can be specified as *survival=T(all)*. This is the original Cox specification where the baseline hazard function is completely saturated. The baseline hazard function is a stepwise hazard function with different hazard values between every two consecutive T values that occur in the data. This was described in more details in Section 2.2.

9.2 Semi-parametric survival

The semi-parametric survival variable is specified as *survival=T(10)* or *survival=T(50)* where the numeric value can be chosen to be any positive integer value. The semi-parametric survival variable uses as a baseline hazard function a stepwise function with 10 or 50 jumps. The intervals are chosen by Mplus internally as to approximate completely the non-parametric Cox stepwise function. As the number of jumps increases the approximation becomes better. If the number of jumps is set to a value N that is as big as the number of different points T that occur in the data the semi-parametric model becomes identical to the Cox fully parametric model. Typically, 10 jumps can be used to approximate the Cox function sufficiently well, however if extra precisions is needed we would recommend 30 jumps. In general it is important to ensure that the model estimates don't depend on the number of jumps and thus some sensitivity investigation should be conducted. The above approximation is similar to how the sample distribution is approximated and represented by a histogram based on a number of different bins. The semi-parametric model is an important alternative to the Cox model because in the situations where the BASEHAZARD option is turned ON the semi-parametric model will have much fewer parameters than the Cox model. That is, the combination of options *BASEHAZARD=ON* and *survival=T(all)* can be computationally very demanding, particularly when the sample size is also large. In fact, if by default Mplus has to turn on the *BASEHAZARD*, Mplus will also default to the semi-parametric model.

9.2.1 BASEHAZARD turned on by default

Now let's focus on why Mplus might turn on by default the *BASEHAZARD* option. This happens precisely when a survival variable is regressed on a latent variable. The reason for this is because the standard errors for the

parameters currently are not implemented in that situation if the *BASEHAZARD* option is off. The auxiliary basehazard parameter maximization will depend on the latent variables and from there on any parameter that is involved in the latent variable distribution. Typically in the EM algorithm, the complete data log-likelihood splits in multiple branches that can be maximized separately and the various derivatives can be computed separately. That is, the model for the latent variables separates from the model of the survival variables in the complete log-likelihood. This assumption, however, breaks down due to the above dependence of the baseline hazard parameters on the latent variables. The second problem that occurs is due to the fact that the envelope theorem typically used to deal with the auxiliary parameters does not hold for the parameter derivatives for individual (one sample point) log-likelihood derivatives needed for the computation of the standard errors and those derivatives become intractable for a general model when the the auxiliary parameters depend on the latent variables and all parameters involved in the latent variable distribution. The dependence of the latent variable on its variance covariance parameters would also have to involve the Cholesky decomposition derivatives with respect to the variance covariance parameters which are also not explicitly available and would have to involve the integration points from the numerical quadrature.

In some special cases where survival variables are regressed on a latent variable the above problems do not exist since the latent variable has a fixed standard normal distribution and therefore no parameters are involved in this cross branch dependence. One such example is a model where a survival variable is regressed on a latent variable measured by an IRT model, such as in Larsen (2005). For other models Mplus can reparameterize the model internally and still avoid the *BASEHAZARD=OFF* problems described above. If this standard error problem with *BASEHAZARD=OFF* can not be avoided then Mplus will still compute the point estimates but will not report any standard errors.

None of these problems exist when the *BASEHAZARD* option is on since the algorithm does not deal with auxiliary parameters and that is why Mplus will default to that algorithm if the model involves a regression from a survival variable on a latent variable. Note also that in this special case when the semi-parametric model is used and the *BASEHAZARD* option is used the baseline hazard parameters are not reported in the Mplus output as regular parameters even though they are. They are reported as auxiliary parameters, i.e., they can be obtained using the *BASEHAZARD* option of the *OUTPUT*

command.

9.3 Parametric survival

The next type of survival variable is the parametric model described in Section 2.1. In this model Mplus estimates a baseline hazard step function where the number of jumps and their precise location are specified by the user. For example *survival=T(10*1)* specifies that there are 10 jumps in the baseline hazard that occur at t=1,2,3,...,10, i.e. there are 10 consecutive intervals with constant (but different) hazard function.

9.4 Constant hazard

The final type of survival variable is the variable where we assume a constant hazard function. This model is the simplest model to estimate and is specified by *survival=T(constant)*. This type should be used only is situations when there are convergence problems with the Mplus default selection.

9.5 Likelihood-ratio testing

It is important to understand that likelihood ratio testing (LRT) between two models should be conducted in general only if the models are based on the same survival type. That is, the log-likelihood values for model 1 and 2 can be used for LRT if both models specify the same type of survival variable. LRT should not in general be used to compare a nonparametric model with a semi-parametric model.

One additional complication occurs when conducting robust LRT with the log-likelihood values for a semi-parametric model obtained with the MLR estimator. Typically the correction number when using the LRT is obtained by

$$cd = \frac{p_0 c_0 - p_1 c_1}{p_0 - p_1}$$

where p_0 and p_1 are the number of parameters in the two models and c_0 and c_1 are the corresponding likelihood correction numbers. When the semi-parametric models is used the values p_0 and p_1 are not the actual model parameters but they should be the number of model parameters + all baseline hazard parameters. This number is typically the largest parameter number found in the technical output 1.

10 Estimating the baseline hazard function

In a recent article, Wong et al. (2018) attempt to estimate the baseline hazard value $h_0(t)$ for particular values of t and the corresponding standard error. They found that the point estimates are consistent but the standard errors are biased and the coverage is underestimated. These issues were encountered due to inappropriately using the Mplus methodology for that purpose. The authors of this article used the saturated baseline hazard approach discussed in Section 2.2, using the option *survival=T(all)*. When there are no ties in the data, the baseline hazard function with this option will have as many parameters as there are observations in the data. As a result of that, the number of parameters in the model will be larger than the number of observations. For example, in the Wong et al. (2018) article, the model has 430 parameters estimated with 400 observations. Such a setup of course yields poor standard errors because these standard errors are based on asymptotic theory, which typically requires many more number of observations than number of parameters. In addition, typically the information matrix in such circumstances would be singular (or near singular) and Mplus would not be able to estimate standard errors for all the parameters. In that case, Mplus will fix parameters that appear unidentified to their point estimates and compute standard errors for the remaining parameters. Ample warnings are provided in the Mplus output in such circumstances. This will also result in many parameters having zero standard errors as the parameters are treated as fixed. In a simulation study, this will of course bias the accumulated results downwards as a portion of the accumulated results are zeros.

When the Cox approach is used for the saturated baseline hazard model, the results are difficult to accumulate over multiple replications as the intervals of the step function change over replications. This also applies to the semi-parametric approach implemented in Mplus where the intervals of the step function vary over replications. The most appropriate way to conduct a simulation study that is focused on computing standard errors on the baseline hazard function is to use a parametric step function where the intervals are fixed across replications.

The most problematic outcome, however, caused by the lack of parsimony in the Wong et al. (2018) approach, is the fact that the parameter estimates become very inefficient. That inefficiency, as will be shown below, extends to a smaller degree into the structural parameters as well and it is not restricted

only to the baseline hazard function parameters.

Note here that the current discussion applies to the situation where there are no ties in the data, i.e., individual survival times are all different. If there are many ties in the data, the saturated step function will not have a large number of parameters and as a result of that the asymptotic theory standard errors and the efficiency of the estimation will not deteriorate. In situations where the survival times take 20 or 30 different values, using the saturated baseline hazard function would yield satisfactory results.

In this section, we discuss optimal ways to estimate a survival model with an arbitrary baseline hazard function. Wong et al. (2018) use the following structural model in a simulation study. The model has two covariates Z_1 and Z_2, seven dependent variables Y_i, i=1,...,7, measuring two latent factors η_1 and η_2 and a single survival variable T. The first five dependent variables are normally distributed, while the last two are binary. The model is given by the following equations

$$Y_i = \mu_i + \lambda_i \eta_1 + \varepsilon_i, i = 1, 2, 3 \tag{18}$$

$$Y_i = \mu_i + \lambda_i \eta_2 + \varepsilon_i, i = 4, 5 \tag{19}$$

$$\eta_2 = \beta_1 \eta_2 + \xi_2 \tag{20}$$

$$Prob(Y_6 = 1) = 1/(1 + Exp(\mu_6 + \beta_2 Z_1 + \beta_2 Z_2 + \beta_4 \eta_2)) \tag{21}$$

$$Prob(Y_7 = 1) = 1/(1 + Exp(\mu_7 + \beta_5 Z_1 + \beta_6 Z_2 + \beta_7 \eta_2 + \beta_8 Y_6)). \tag{22}$$

The survival variable T follows the proportional hazard model

$$h(t) = h_0(t) Exp(\gamma_1 Z_1 + \gamma_2 Z_2 + \gamma_3 Y_6 + \gamma_4 Y_7 + \gamma_5 \eta_2), \tag{23}$$

where the baseline hazard function $h_0(t) = 2t$ or equivalently the cumulative baseline hazard function $H_0(t) = t^2$. The covariate Z_1 is generated from a standard normal distribution, while the covariate Z_2 is a binary variable with odds set to 1. The structural parameter values β_i and γ_i used for the data generation can be found in Table 1 in Wong et al. (2018). The sample size used for the simulation study is $N = 400$.

To estimate a this model in Mplus with a non-parametric baseline hazard function, we estimate the model using a step function baseline hazard. A step function can be used to approximate any function and the precision of the approximation depends on the number and the sizes of the steps. In a general context, setting up a step function that has many small steps

is generally preferred as the approximation would be more precise. This, however, is not the case in statistical applications, where there is a steep penalty for adding many steps in the function because that increases the number of parameters in the model and results in overparametrization and poor quality of the baseline hazard estimates. The worst case scenario is used in Wong et al. (2018) where the fully saturated step function is used.

To be more clear, using the fully saturated baseline hazard is the preferred approach when we are estimating only a structural model without estimating the baseline hazard function. This is of course the basis of the Cox proportional hazard model estimation. In that case, however, the base hazard parameters are treated as auxiliary parameters which are eliminated from the likelihood and ultimately only the structural parameters are included in the information matrix. The difference between the Cox proportional hazard model estimation and the estimation used in Wong et al. (2018) is that the the first one does not estimate the baseline hazard function while the second does.

Because we want to evaluate the estimation method in a simulation study, we use the parametric step function approach in Mplus where the number of steps and the step sizes are specified before the estimation. This guarantees that the step function has the same number of steps and the same step sizes across the replications. The semi-parametric approach and the saturated step-function are not feasible in a simulation study due to the fact that the size of the intervals change across replications. In the following discussion we consider step functions with equal steps as follows. If the step function has L steps of size δ the baseline hazard function is defined as follows

$$h_0(t) = \begin{cases} h_i & \text{if } (i-1)\delta < t \leq i\delta \text{ for } i = 1, ..., L \\ h_{L+1} & \text{if } L\delta < t \end{cases} . \tag{24}$$

The above function has $L+1$ parameters h_i for $i = 1, ..., L+1$. The function is constant in each of the first L intervals of size δ and the last parameter is the value of the remaining infinite interval. The step function $h_0(t)$ can be thought of as a non-parametric function because it can be used to approximate any baseline hazard function. Typically, the step function is smoothed after estimation so that it does not have discontinuities at the times $i\delta$. One simple way of doing this is to assume that the value h_i is the baseline hazard function value at the mid-point of the i-th interval $t_i = (2i-1)\delta/2$. We can then fit a spline through the points (t_i, h_i), for $i = 0, ..., L+1$, assuming $t_0 = 0$ and $h_0 = 0$. For our illustration purposes, we use a linear spline smoothing

but other splines such as quadratic or cubic can be utilized similarly. The linear spline method gives the following smoothed baseline hazard function

$$h_0^*(t) = \begin{cases} h_{i-1}\frac{t_i - t}{t_i - t_{i-1}} + h_i \frac{t - t_{i-1}}{t_i - t_{i-1}} & \text{if } t_{i-1} < t \leq t_i \text{ for } i = 1, ..., L+1 \\ h_L \frac{t_{L+1} - t}{t_{L+1} - t_L} + h_{L+1}\frac{t - t_L}{t_{L+1} - t_L} & \text{if } t_{L+1} < t \end{cases} . \tag{25}$$

The above formula simply says that for a point t between t_{i-1} and t_i the smoothed baseline hazard function $h_0^*(t)$ is a weighted average between h_{i-1} and h_i where the weights are based on the distance of t to t_{i-1} and t_i. Smoothing the baseline hazard function would be particularly important if the number of intervals L is small (such as 5 or 10) and we are interested in estimating the baseline hazard at values that are not among the nodes t_i.

The choice of the baseline hazard function is essentially determined by the number of intervals L and by the value $L\delta$ where the infinite interval starts. To estimate well the parameter h_{L+1} in the infinite interval we generally want to have between 20 and 50 observations in that interval but not more than 10% of the total number of observations. Assessing the data generated in Wong et al. (2018), we see that about 10% of the observations (i.e. 40 observations) are larger than 1.5. Thus, we use as the start of the infinite interval $L\delta = 1.5$. We consider three different methods for determining the number of intervals L. The first method is to choose L as the value which minimizes the BIC criterion. Denote this value by L_{BIC}. The second method is to choose L as the value which minimizes the AIC criterion. Denote this value by L_{AIC}. The AIC criterion tends to overparameterize the model but this could result in good approximation for the true baseline hazard function. The BIC criterion, on the other hand, uses a penalty for overparameterization and can be expected to yield a more parsimonious model, but still, a model that extracts enough information from the data to appropriately model the change of hazard across time. The third method is simply choosing $L = 30$ which would allow us to approximate the true baseline hazard function to a reasonably detailed level while keeping the model within identifiable bounds.

To determine the L_{AIC} and L_{BIC} we estimate the above model with $L = 1, 2, ...$ and evaluate AIC and BIC. The results, averaged over 100 replications, are presented in Table 1. These results imply that $L_{BIC} = 5$ and therefore $\delta_{BIC} = 0.3$, and $L_{AIC} = 12$ and therefore $\delta_{AIC} = 0.125$.

In Wong et al. (2018) the baseline hazard function is evaluated at three particular points in time $t = 0.45, 0.72$, and 1.05 which represent 25% quantile, 50% quantile and 75% quantile in the distribution of T. We evaluate the

Table 1: BIC and AIC for different values of L

L	BIC	AIC
0	6099	5975
1	6085	5957
2	6021	5889
3	5993	5857
4	5984	5844
5	5979	5836
6	5979	5831
7	5981	5830
8	5985	5829
9	5987	5828
10	5991	5828
11	5995	5827
12	5999	5827
13	6004	5828
14	6008	5829
15	6013	5829
30	6084	5841

baseline hazard function at those three time points as well. The smoothed values $h_0^*(t)$ can be computed in Mplus as new parameters in Model Constraint using equation (25).

Next, we present the estimation results using the above three specifications for the baseline hazard function. Table 2 contains the results based on the L_{BIC} estimation, averaged over 500 replications, for the baseline hazard parameters as well as the regression parameters for the survival variable T. In addition to the bias and coverage, the table contains the square root of the mean squared error (SMSE) for the estimates. Table 3 contains the results based on the L_{AIC} estimation. Table 4 contains the results based on the $L = 30$ estimation. In this case, since the estimated baseline hazard is quite detailed, we can use also the baseline hazard values without smoothing. These parameters are also included in the Table 4.

The results indicate that the bias is minimal with all three estimations. The coverage is near the nominal level as well. The results also point out that there is a substantial benefit in keeping the model as parsimonious as possible. The SMSE for the baseline hazard parameters is smallest for the L_{BIC} estimation, followed by the L_{AIC} estimation, followed by the $L = 30$ estimation. For the structural parameters the differences between the three methods are minimal, however, a small SMSE advantage for the L_{BIC} estimation can be seen. Comparing the results for the smoothed baseline hazard function $h_0^*(t)$ and the baseline hazard step-function $h_0(t)$, we see that even for the $L = 30$ estimation there is a benefit of smoothing the estimates. The smoothed estimates have a smaller bias and a better SMSE. This simulation study illustrates the fact that the standard errors for the baseline hazard parameters are computed correctly in Mplus and that the poor coverage reported in Table S1 in Wong et al. (2018) are due to inappropriately setting up the baseline hazard function with a poorly identified model.

Next, we demonstrate how to use the baseline hazard step-function approach to model actual parametric models. In the Wong et al. (2018) simulation study the baseline hazard function is linear $h_0(t) = 2t$. To estimate such a linear model we simply constrain the baseline hazard parameters h_i to follow a linear model

$$h_i = a + b \cdot i \tag{26}$$

for $i = 1, ..., L$. Introducing such a constraint in the model is beneficial not just in terms of providing an actual parametric model for the baseline hazard function, but also in further reducing the number of parameters in

Table 2: Model results using L_{BIC}

Parameter	absolute bias	coverage	SMSE
$h_0^*(0.45)$	0.00	0.93	0.22
$h_0^*(0.72)$	0.00	0.91	0.34
$h_0^*(1.05)$	0.04	0.93	0.56
γ_1	0.00	0.96	0.07
γ_2	0.00	0.94	0.18
γ_3	0.01	0.94	0.14
γ_4	0.00	0.95	0.16
γ_5	0.03	0.94	0.15

Table 3: Model results using L_{AIC}

Parameter	absolute bias	coverage	SMSE
$h_0^*(0.45)$	0.02	0.92	0.25
$h_0^*(0.72)$	0.03	0.92	0.38
$h_0^*(1.05)$	0.11	0.93	0.69
γ_1	0.00	0.96	0.07
γ_2	0.00	0.94	0.18
γ_3	0.01	0.95	0.14
γ_4	0.00	0.95	0.16
γ_5	0.01	0.96	0.16

Table 4: Model results using $L = 30$

Parameter	absolute bias	coverage	SMSE
$h_0^*(0.45)$	0.02	0.92	0.28
$h_0^*(0.72)$	0.05	0.93	0.51
$h_0^*(1.05)$	0.15	0.94	0.78
γ_1	0.00	0.96	0.07
γ_2	0.00	0.94	0.18
γ_3	0.01	0.94	0.14
γ_4	0.00	0.95	0.16
γ_5	0.01	0.96	0.17
$h_0(0.45)$	0.03	0.90	0.32
$h_0(0.72)$	0.07	0.93	0.54
$h_0(1.05)$	0.22	0.95	0.98

the model and making it more parsimonious. With the above constraint the baseline hazard function would have only 3 parameters: a, b and h_{L+1}. Table 5 contains the results for L_{BIC} and Table 6 contains the results for L_{AIC} estimated with linear baseline hazard function. The results indicate that the bias is minimal for all parameter estimates and coverage is near the nominal levels. We also see here that introducing the linear constraint in the model has further improved the SMSE for the baseline hazard parameters. In addition, we see that the advantages of the L_{BIC} method over the L_{AIC} method are now minimal. Note that the two models under the linearity constraint have the same number of parameters. This is further evidence that making the baseline hazard model more parsimonious leads to substantial improvements in the precision of the estimates.

Wong et al. (2018) also report that they were unable to obtain cumulative hazard estimates in Mplus. Using formula (7) we can obtain the cumulative hazard values as follows. If $(i-1)\delta < t \leq i\delta$ the cumulative hazards is

$$H_0(t) = \delta \sum_{j=1}^{i-1} h_j + h_i(t - (i-1)\delta). \tag{27}$$

If $\delta(2i-1)/2 < t \leq \delta(2i+1)/2$, the smoothed cumulative hazard function is

Table 5: Model results using L_{BIC} with linear baseline hazard

Parameter	absolute bias	coverage	SMSE
$h_0^*(0.45)$	0.00	0.93	0.20
$h_0^*(0.72)$	0.01	0.93	0.32
$h_0^*(1.05)$	0.02	0.93	0.47
γ_1	0.00	0.95	0.07
γ_2	0.00	0.94	0.18
γ_3	0.01	0.94	0.14
γ_4	0.00	0.95	0.16
γ_5	0.03	0.94	0.14

Table 6: Model results using L_{AIC} with linear baseline hazard

Parameter	absolute bias	coverage	SMSE
$h_0^*(0.45)$	0.02	0.94	0.21
$h_0^*(0.72)$	0.04	0.94	0.33
$h_0^*(1.05)$	0.06	0.94	0.49
γ_1	0.00	0.95	0.07
γ_2	0.00	0.93	0.18
γ_3	0.01	0.94	0.14
γ_4	0.00	0.95	0.16
γ_5	0.00	0.95	0.15

Table 7: Cumulative hazard

Parameter	absolute bias	coverage	SMSE
$H_0(0.45)$	0.02	0.94	0.06
$H_0(0.72)$	0.02	0.93	0.12
$H_0(1.05)$	0.03	0.94	0.25
$H_0^*(0.45)$	0.02	0.94	0.06
$H_0^*(0.72)$	0.02	0.94	0.12
$H_0^*(1.05)$	0.03	0.94	0.25

computed as follows

$$H_0^*(t) = \delta \sum_{j=1}^{i-1} h_j + \delta h_i/2 + (t - \delta(2i-1)/2)(h_0^*(t) + h_i)/2. \qquad (28)$$

Both of these can be obtained in Mplus by setting the above formulas as new parameters in the Model Constraint command. To illustrate this computation we use the L_{BIC} estimation. The results, presented in Table 7, show that the bias is minimal and the coverage is near the nominal level. The difference between the smoothed and unsmoothed versions is negligible. The smoothing advantage we saw for the baseline hazard function may not exist for the cumulative hazard function. This is most likely due to the fact that the cumulative baseline hazard function $H_0(t)$ is already a continuous function, unlike the baseline hazard function $h_0(t)$.

To conclude this section, we reiterate that the problems reported in Wong et al. (2018) are due to inappropriate use of the methodology implemented in Mplus. Using the same simulation study as in Wong et al. (2018), we illustrate above that the correct estimation setup yields satisfactory results. All Mplus input and output files used in the above simulation studies can be found at statmodel.com.

References

Asparouhov, T., Masyn, K. & Muthén, B. (2006). Continuous time survival in latent variable models. Proceedings of the Joint Statistical Meeting in Seattle, August 2006. ASA section on Biometrics, 180-187.

Bradburn, M. J., Clark T.G., Love S.B., and Altman D.G. (2003), Survival Analysis Part II: Multivariate Data Analysis - an introduction to concepts and methods, British Journal of Cancer v. 89, 431-436.

Clayton, D. G. (1988), The analysis of event history data: A review of progress and outstanding problems, Statistics in Medicine, v. 7, 819-841.

Hougaard, P. (2000) Analysis of Multivariate Survival Data. Springer, New York.

Klein J.P. & Moeschberger, M.L. (1997) Survival analysis: techniques for censored and truncated data. New York: Springer.

Larsen, K. (2004) Joint Analysis of Time-to-Event and Multiple Binary Indicators of Latent Classes. Biometrics 60 (1), 85-92.

Larsen, K. (2005) The Cox Proportional Hazards Model with a Continuous Latent Variable Measured by Multiple Binary Indicators. Biometrics 61:4, 1049-1055

Murphy, S.A. & van der Vaart, A.W. (2000). On profile likelihood. Journal of the American Statistical Association. 95, 449-465.

Muthén, B., Asparouhov, T., Boye, M., Hackshaw, M. & Naegeli, A. (2009). Applications of continuous-time survival in latent variable models for the analysis of oncology randomized clinical trial data using Mplus. Technical Report. http://www.statmodel.com/download/lilyFinalReportV6.pdf

Muthén, L.K. & Muthén, B.O. (1998-2018) Mplus User's Guide. Eight Edition. Los Angeles, CA: Muthén & Muthén.

Singer, J. D. & Willett, J. B. (2003) Applied Longitudinal Data Analysis: Modeling Change and Event Occurrence. New York: Oxford University Press.

Wong K., Zeng D. & Lin D. Y. (2018) Efficient Estimation for Semiparametric Structural Equation Models With Censored Data. Journal of the American Statistical Association.
https://doi.org/10.1080/01621459.2017.1299626