Journal of Data Science 14(2016), 701-712

A brief note on the simulation of survival data with a desired percentage of right-censored datas

Edson Zangiacomi Martinez[1*], Jorge Alberto Achcar[1],

Marcos Vinicius de Oliveira Peres[2] and Jose Andre Mota de Queiroz[2]

[1]Ribeir~ao Preto Medical School, University of S~ao Paulo, Brazi[l]
[2]Department of Statistics, State University of Maringa, Brazil

Abstract: Simulation studies are important statistical tools used to inves-tigate the performance, properties and adequacy of statistical models. The simulation of right-censored time-to-event data involves the generation of two independent survival distributions, where the rst distribution repre-sents the uncensored survival times and the second distribution represents the censoring mechanism. In this brief report we discuss how we can make it so that the percentage of censored data is previously de ned. The described method was used to generate data from a Weibull distribution, but it can be adapted to any other lifetime distribution. We further presented an R code function for generating random samples, considering the proposed approach.

Key words: Censored data, simulation, survival analysis, Weibull distribution

1. Introduction

Simulation studies use computer intensive procedures to assess the perfor-mance of a variety of statistical methods in relation to a known truth (Burton et al., 2006). In the context of time-to-event data, Bender et al. (2005) presented techniques to generate survival times for simulation studies regarding Cox pro-portional hazards models. Austin (2012) introduced data-generating processes for the Cox proportional hazards model with time-varying covariates when event times follow an exponential, Weibull or Gompertz distribution. A method for the simulation of survival times that follow a Cox proportional hazards model with time-dependent covariates was also developed by Hendry (2014). Royston (2012) described an approach for simulation based on creating pseudo-random sample from Royston-Parmar distributions, or to say, a distribution with baseline distri-bution function modelled as a restricted cubic spline function of the logarithms of the survival times. Mori~na and Navarro (2014) presented an R code package for the simulation of survival data including recurrent and multiple events.

The simulation of censored survival data in a general way requires the genera-tion of two survival distributions. The rst distribution represents the uncensored survival times and the second distribution represents the censoring mechanism. The two simulated distributions are thus combined so that the event of interest is considered to be observed when the uncensored survival

time is less than or equal to the censored time. Otherwise, the event is censored and the survival time corresponds to the censored time (Burton et al., 2006). In this procedure, the proportion of censored data depends on the censoring distribution. In this brief note, we show how to simulate right-censored survival data considering a desired percentage of censorship (0 < < 1). We considered the Weibull dis-tribution as a special case, but the algorithm described here can be adapted to other distributions. In an Appendix we present an R code function (Crawley, 2012) in order to facilitate this task.

2. Method

Let T be a nonnegative-valued random variable representing the time until an event occurs, with cumulative distribution function given by F_T (t) = P (T$\leqq$ t) = $\int_0^t fT$ (x)dx, where f_T (t) is its respective probability density function. The survival function describing the probability of surviving after time point t is given by S_T (t) = 1 -F_T (t) = P (T > t). In addition, suppose that C is the censoring variable, with distribution function F_C (t), and let us assume that T and C are independent. In this way, we de ne the random variable Y as follows:

$$\mathrm{Y} = \begin{cases} T \; if \; C \geq T \\ C \; if \; C \; < T \end{cases}$$

Let us de ne a censoring indicator variable d where d = 1 for a observed lifetime (C $\geq$T) and d = 0 for a censored observation (C < T). Given the assumption of independence between C and T , the probability P (C < T) is given by

$$P(C < T) = \int_0^\infty \int_0^t f_{T,C}(t,c)dedt = \int_0^\infty F_C(t)\, f_T(t)dt = \theta, \qquad (1)$$

Where is the desired percentage of censored data (0 < < 1) and fT;C (t; c) = fT (t)fC (c) is the joint probability density function of the random variables T and C. In a simulation study, the exponential distribution is a convenient choice for C, given that it is a continuous probability distribution that has a constant failure rate. However, other choices are also possible, according to the interest of the researcher.

Let us suppose, for example, C Exponential( ) and T W eibull(;), where > 0 and > 0 are parameters with values that are chosen by the researcher and is a parameter that should be determined so that the equality

$$P(C < T) = \lambda\alpha \int_0^\infty \left(1 - e^{-\beta t}\right)(t\lambda)^{\alpha-1}\, exp[-(t\lambda)^\alpha]\, dt = \theta \qquad (2)$$

holds. In general, this integral cannot be solved in closed form and numerical solutions have to be obtained. In the present study, we use the R code function \integrate" to numerically evaluate the integral in the equation (2) and the R function \uniroot" to nd the value of that satis es (2) for xed values of λ,α and θ.

The algorithm used to simulate a sample of size n from the lifetime distribu-tion of interest follows the steps:

Step 1. Fix values of λ,α and θ and replace them in the expression (2).

Step 2. From (2), find the value of β such as P (C < T) = θ .

Step 3. Generate *n* random samples from U $\sim$Uniform(0,1). The motivation for this is that if *T* has a distribution function F_T(t), then U = F_T (T) follows a uniform distribution on (0,1). Conversely, if U $\sim$Uniform(0,1) then T = F_T^{-1}(U) has distribution function F*T* .

Step 4. Using the parameters λ and a fixed in the first step, random samples for T are obtained from $F_T^{-1}(U) = \frac{1}{\lambda}[-\log(1-U)]^{1/\alpha}$. In this case, F_T (t) = 1-exp[-($t\lambda$)$^{\alpha}$] is the cumulative distribution function of the Weibull distribution with parameters and .

Step 5. Generate n random samples from W $\sim$Uniform(0,1).

Step 6. Random samples for C are obtained from $F_C^{-1}(W) = \frac{1}{\beta}[-\log(1-W)]$, where is given in the Step 2. Here, $F_C(t) = 1 - \exp(t\beta)$ is the cumulative distribution function of the exponential distribution with parameter .

Step 7. Random samples for Y are given by Y = min(T, C):

Step 8. Pairs of values (y1,d1), (y2,d2),…, (yn,dn) are thus obtained, where $d_i = 1$ if $c_i \geqq t_i$ and $d_i = 0$ if $c_i < t_i$, i = 1,…,n.

This algorithm can be adapted to accommodate other continuous distribu-tions for survival time that have explicit cumulative distribution function.

3. Results

Considering the proposed algorithm, let us simulate B random samples of size n from a Weibull distribution with a percentage of censored data given by θ. Let $\theta^*_{(b)}$ be the percentage of censored data observed in the b-th simulated sample (b = 1,…,B). The mean and the standard deviation of the values for $\theta^*_{(b)}$ are given respectively by

$$m(\theta^*) = \sum_{b=1}^{B} \frac{\theta^*_{(b)}}{B} \quad and \quad sd(\theta^*) = \sqrt{\sum_{b=1}^{B} \frac{[\theta^*_{(b)} - m(\theta^*)]^2}{B}}.$$

In addition, the percentiles 2.5% and 97.5% (denoted by $p_{2.5\%}$ and $p_{97.5\%}$, respectively) are used to describe a range in which 95% of the values for (b) are contained.

Table 1 shows the results for B = 100,000 random samples of sizes n = 15,20,30,50, 100 and 500, percentages of censored data given by = 0.05, 0.15, 0.25, 0.30, 0.50, 0.75 and 0.90, and the parameters and were arbitrarily set at 0.2 and 2.4, respectively.

Supposing C∼ Exponential(β) and T∼Weibull(λ,α), the appropriate values for β were determined according to the values of θ,λ and α (step 2 of the algorithm). For example, considering a percentage of censored data equal to θ = 0.05 and the parameters with xed values λ= 0.2 and α = 2.4, the value for that satisfies the expression

$$\int_0^{\infty} (1 - e^{-\beta t})(0.2t)^{1.4} \exp[-(0.2t)^{2.4}]\, dt = \frac{0.05}{0.2 \times 2.4}$$

is approximately β ≈0.01163 (see expression (2)). In the Appendix, we provide an R code function to get values for β. In Table 1, we observe in all simulations that the values obtained for m (θ*) are close to the respective nominal values of . In addition, the values for the standard deviation sd (θ*) decrease as the sample size n increases. In turn, values for sd (θ*) increase as the percentage of censored data approaches 0.5 for any sample size.

Let $\hat{\lambda}_{ML}^{(b)}$ and $\hat{\alpha}_{ML}^{(b)}$ be the maximum likelihood (M L) estimates for and , respectively, considering the b-th simulated sample (b = 1,…,B). Let us define $\mathrm{m}(\hat{\lambda}_{ML}^{*})$ and $sd(\hat{\lambda}_{ML}^{*})$ as respectively the mean and the standard deviation

Table 1: Computational results for the simulation study, considering λ= 0.2, α= 2.4, B = 100,000 random samples of sizes n = 15,20,30,50,100 and 500, and percentages of censored data given by = 0.05, 0.15, 0.25, 0.30, 0.50, 0.75 and 0.90.

θ	Statistics	$n=15$	$n=20$	$n=30$	$n=50$	$n=100$	$n=500$
0.05	$m(\theta^*)$	0.0502	0.0501	0.0502	0.0501	0.05	0.05
	$sd(\theta^*)$	0.0565	0.0486	0.0401	0.0310	0.0217	0.0098
	$p_{2.5\%}$	0	0	0	0	0.01	0.032
	$p_{97.5\%}$	0.2	0.15	0.1333	0.12	0.10	0.07
0.15	$m(\theta^*)$	0.1501	0.1501	0.1502	0.1498	0.15	0.15
	$sd(\theta^*)$	0.0924	0.0797	0.0651	0.0504	0.0358	0.0159
	$p_{2.5\%}$	0	0	0.0333	0.06	0.08	0.12
	$p_{97.5\%}$	0.3333	0.3	0.3	0.26	0.22	0.182
0.25	$m(\theta^*)$	0.2499	0.2499	0.2503	0.25	0.25	0.25
	$sd(\theta^*)$	0.1114	0.0968	0.0791	0.0611	0.0433	0.0193
	$p_{2.5\%}$	0.0667	0.1	0.1	0.14	0.17	0.212
	$p_{97.5\%}$	0.4667	0.45	0.4	0.38	0.34	0.288
0.30	$m(\theta^*)$	0.3006	0.3004	0.3006	0.3001	0.2999	0.3
	$sd(\theta^*)$	0.1183	0.1026	0.0838	0.0647	0.0456	0.0205
	$p_{2.5\%}$	0.0667	0.1	0.1333	0.18	0.21	0.26
	$p_{97.5\%}$	0.5333	0.5	0.4667	0.44	0.39	0.34
0.50	$m(\theta^*)$	0.4997	0.5001	0.5006	0.5	0.4997	0.5
	$sd(\theta^*)$	0.1291	0.1123	0.0909	0.0707	0.0498	0.0223
	$p_{2.5\%}$	0.2667	0.3	0.3333	0.36	0.4	0.456
	$p_{97.5\%}$	0.7333	0.7	0.6667	0.64	0.6	0.544
0.75	$m(\theta^*)$	0.7498	0.7503	0.7496	0.7502	0.7499	0.7501
	$sd(\theta^*)$	0.1113	0.0968	0.0789	0.0612	0.0433	0.0193
	$p_{2.5\%}$	0.5333	0.55	0.6	0.62	0.66	0.712
	$p_{97.5\%}$	0.9333	0.9	0.9	0.86	0.83	0.788
0.90	$m(\theta^*)$	0.8996	0.8999	0.9003	0.8999	0.8998	0.90
	$sd(\theta^*)$	0.0774	0.0669	0.0548	0.0425	0.0301	0.0134
	$p_{2.5\%}$	0.7333	0.75	0.7667	0.82	0.84	0.874
	$p_{97.5\%}$	1	1	1	0.98	0.95	0.926

of the B estimated values for $\hat{\lambda}_{ML}^{(b)}$ and, similarly, m ($\hat{\alpha}_{ML}^{*}$) and sd ($\hat{\alpha}_{ML}^{*}$) are respectively the mean and the standard deviation of the B values estimated for $\hat{\alpha}_{ML}^{(b)}$.Table 2 shows the obtained values for $\mathrm{m}(\hat{\lambda}_{ML}^{*})$, $sd(\hat{\lambda}_{ML}^{*})$, $\mathrm{m}(\hat{\alpha}_{ML}^{*})$ and $sd(\hat{\alpha}_{ML}^{*})$ considering B = 100,000, λ= 0.2, α= 2.4 and several choices for n and θ. In this simulation study, the maximum likelihood estimations were performed using the maxLik library in R software (Henningsen and Toomet, 2011), where the likelihood equations were solved by a method based on the Newton-Raphson algorithm.

In Table 2 we observe that the means $\mathrm{m}(\hat{\lambda}_{ML}^{*})$ are satisfactorily close to the nominal value λ= 0.2 in all performed simulations. The standard deviations $sd(\hat{\lambda}_{ML}^{*})$ decrease as the sample size n increases. However, we can observe that means $\mathrm{m}(\hat{\alpha}_{ML}^{*})$ are far from the nominal value and the respective values for sd ($\hat{\alpha}_{ML}^{*}$) are quite large, in the cases where the percentage of censored data is higher than 0.50 and the sample size is relatively small (say less than 100). This is due to the presence of simulated samples resulting on monotone likelihood and, in the present study, this was observed when all simulated observations are censored. Heinze and Schemper (2001) observed that monotone likelihood is primarily a problem of small sample bias which implies in the nonexistence of the maximum likelihood estimates. This situation can lead to spurious results, according to the computational algorithm used. Table 2 describes the incidence of monotone likelihood (abbreviated IM L) found in each simulation. We observe, for example, that in the case where θ= 0.90, the incidence of monotone likelihood was 20.39%, 11.96% and 4.69% when n = 15, 20 and 30, respectively.

4. Discusion

The proposed algorithm provides a practical method to simulate survival data with a desired percentage of right-censored data. This method is not intended to be regarded as unprecedented in the literature, given that a number of authors have used similar mechanisms to generate samples with censored data (see, as examples, Mantovani and Franco (2004) and Eudes et al. (2013)). However, the present report is intended to be useful to students and researchers, given that it describes in detail the algorithm used to simulate data samples and also provides a R code program that can be used for this purpose.

Based on the obtained results of Table 2, we recommend caution in using this simulation method when the desired percentage of censored data is relatively large and the sample size is relatively small. This can lead to problems due the possibility of all simulated observations be

censored and, in this case, the maximum likelihood estimates do not exist since the likelihood function has no local maximum.

Table 2: Summaries for the maximum likelihood estimates, considering λ= 0.2, α= 2.4, B = 100, 000 random samples of sizes n = 15, 20, 30, 50, 100 and 500, and percentages of censored data given by θ= 0.05, 0.15, 0.25, 0.30, 0.50, 0.75 and 0.90. *IM L* denotes the incidence of monotone likelihood.

θ	Statistics	$n=15$	$n=20$	$n=30$	$n=50$	$n=100$	$n=500$
0.05	$m\left(\widehat{\lambda}^{*}_{ML}\right)$	0.2036	0.2025	0.2018	0.2011	0.2004	0.2001
	$sd\left(\widehat{\lambda}^{*}_{ML}\right)$	0.0243	0.0206	0.0167	0.0128	0.009	0.004
	$m\left(\widehat{\alpha}^{*}_{ML}\right)$	2.6649	2.5905	2.5199	2.4716	2.436	2.407
	$sd\left(\widehat{\alpha}^{*}_{ML}\right)$	0.6297	0.5131	0.3932	0.2888	0.2002	0.085
	IML	0	0	0	0	0	0
0.15	$m\left(\widehat{\lambda}^{*}_{ML}\right)$	0.2041	0.2029	0.2021	0.2009	0.2006	0.2001
	$sd\left(\widehat{\lambda}^{*}_{ML}\right)$	0.0256	0.0218	0.0174	0.0134	0.0094	0.0041
	$m\left(\widehat{\alpha}^{*}_{ML}\right)$	2.6923	2.6026	2.5338	2.4821	2.4354	2.4083
	$sd\left(\widehat{\alpha}^{*}_{ML}\right)$	0.6686	0.5353	0.4093	0.3041	0.2076	0.0907
	IML	0	0	0	0	0	0
0.25	$m\left(\widehat{\lambda}^{*}_{ML}\right)$	0.2044	0.2030	0.2021	0.2013	0.2007	0.2001
	$sd\left(\widehat{\lambda}^{*}_{ML}\right)$	0.027	0.0229	0.0184	0.0139	0.0099	0.0044
	$m\left(\widehat{\alpha}^{*}_{ML}\right)$	2.7229	2.6444	2.5501	2.4838	2.442	2.4087
	$sd\left(\widehat{\alpha}^{*}_{ML}\right)$	0.7549	0.5951	0.4443	0.323	0.2187	0.0945
	IML	0	0	0	0	0	0
0.30	$m\left(\widehat{\lambda}^{*}_{ML}\right)$	0.2044	0.2033	0.2025	0.2011	0.2008	0.2001
	$sd\left(\widehat{\lambda}^{*}_{ML}\right)$	0.028	0.0238	0.0189	0.0146	0.0103	0.0045
	$m\left(\widehat{\alpha}^{*}_{ML}\right)$	2.7531	2.6389	2.5652	2.4869	2.4421	2.41
	$sd\left(\widehat{\alpha}^{*}_{ML}\right)$	0.7866	0.607	0.4616	0.332	0.2235	0.0971
	IML	0	0	0	0	0	0
0.50	$m\left(\widehat{\lambda}^{*}_{ML}\right)$	0.2062	0.2045	0.2028	0.2018	0.2008	0.2001
	$sd\left(\widehat{\lambda}^{*}_{ML}\right)$	0.0336	0.0284	0.0225	0.0172	0.0119	0.0053
	$m\left(\widehat{\alpha}^{*}_{ML}\right)$	2.9254	2.7513	2.6005	2.5157	2.4547	2.4096
	$sd\left(\widehat{\alpha}^{*}_{ML}\right)$	2.7341	0.8741	0.5650	0.3941	0.2623	0.1113
	IML	0	0	0	0	0	0
0.75	$m\left(\widehat{\lambda}^{*}_{ML}\right)$	0.2076	0.2058	0.2035	0.2018	0.2012	0.2004
	$sd\left(\widehat{\lambda}^{*}_{ML}\right)$	0.0607	0.0480	0.0385	0.0296	0.0198	0.0078
	$m\left(\widehat{\alpha}^{*}_{ML}\right)$	6.7385	4.0484	2.8805	2.5822	2.4858	2.4156
	$sd\left(\widehat{\alpha}^{*}_{ML}\right)$	25.2417	12.5656	4.0518	0.6078	0.3812	0.1484
	IML	1.3%	0.32%	0.02%	0.001%	0	0
0.90	$m\left(\widehat{\lambda}^{*}_{ML}\right)$	0.1913	0.1983	0.2028	0.2021	0.2005	0.2003
	$sd\left(\widehat{\lambda}^{*}_{ML}\right)$	0.1275	0.1074	0.0809	0.0517	0.0397	0.0158
	$m\left(\widehat{\alpha}^{*}_{ML}\right)$	21.6212	14.8605	7.3222	3.3665	2.5749	2.4308
	$sd\left(\widehat{\alpha}^{*}_{ML}\right)$	56.4377	42.6262	24.4186	8.0685	0.8181	0.236
	IML	20.44%	12.16%	4.18%	0.54%	0.003%	0

In the cases where IM L > 0, the results showed in Table 2 can be di erent when obtained from another statistical package, given that each software can consider a di erent stopping criteria for the iterative procedure of the Newton-Raphson method. When using the maxLik library in R software, the algorithm stops if the absolute di erence between successive iterations is less than 1×10^{-8}, However, the results in Table 2 are useful to illustrate the potential e ects of the incidence of monotone likelihood in a simulation study.

We also simulated random samples considering other values for the parameters and , but the results were analogous to those described in Tables 1 and 2.

Appendix - The R code

Assuming that T follows a Weibull distribution with probability density func-tion (pdf) fT (t) = (t) 1 exp [(t)], t > 0; and C follows a exponential distribution with pdf fC (t) = exp (t), the following function sol.beta pro-vides the value for that satis es P (C < T) = (equation (1)).

```
fint <- function(y,beta,lambda,alpha) {

fint <- (1-exp(-beta*y))*(y*lambda)^(alpha-1)

*exp(-(y*lambda)^alpha) }

fun <- function(x,lambda,alpha,theta) {

r <- integrate(fint,beta=x,lambda=lambda,alpha=alpha,lower = 0, upper = Inf)

fun <- lambda * alpha * r$value - theta }

sol.beta <- function(lambda=lambda,alpha=alpha,theta=theta) f sol <-
uniroot(fun,lambda=lambda,alpha=alpha,theta=theta, interval= c(0.01, 10))

return(as.numeric(sol$root)) }
```

For example, assuming = 0:2, = 2:4 and = 0:4, we have

```
> sol.beta(0.2,2.4,0.4)

[1] 0.1214826
```

The following function rcensWeib is used to generate random samples of size n from a Weibull distribution with parameters and and a desired percentage of right-censored data given by θ.

```
rcensWeib <- function(n,lambda,alpha,theta) {
beta <- sol.beta(lambda,alpha,theta)
w <- runif(n,0,1)
c0 <- (-log(1-w)/beta)
u <- runif(n,0,1)
t0 <- (1/lambda)*(-log(1-u))^(1/alpha)
t <- pmin(t0,c0)
d <- as.numeric(c0>=t0)
dados <- data.frame(t,d) return (dados) }
```

For example, assuming n = 20; = 0:2, = 2:4 and = 0:1, we have

```
> rcensWeib(20,0.2,2.4,0.1) t d
1 4.1775609 1
2 8.0388403 1
3 4.5675561 1
4 4.2659948 1
5 4.9293638 1
6 3.2307409 1
7 1.9213973 1
8 6.4329806 1
9 6.0117327 1
10 0.4023496 0
```

11 3.8465585 1

12 3.6301429 1

13 2.6819304 1

14 8.0879792 0

15 5.7055924 1

16 3.3186272 0

17 4.4106509 1

18 2.5296499 1

19 2.6484461 1

20 2.2352938 1

>

References

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

[2] Bender, R., Augustin, T. and Blettner, M. (2005). Generating survival times to simulate Cox proportional hazards models. Statistics in Medicine 24, 1713-1723.

[3] Burton, A., Altman, D. G., Royston, P. and Holder, R. L. (2006). The design of simulation studies in medical statistics. Statistics in Medicine 25, 4279-4292.

[4] Crawley, M. J. (2012) The R Book. 2nd Edition. Chichester: John Wiley & Sons. 1076p.

[5] Eudes, A. M., Tomazella, V. L. D. and Calsavara, V. F. (2013). Modelling survival with a cured fraction for lifetime data Weibull modi ed. Biometric Brazilian Journal 30, 326-342.

[6] Heinze, G. and Schemper, M. (2001). A solution to the problem of monotone likelihood in Cox regression. Biometrics 57, 114-119.

[7] Hendry, D. J. (2014). Data generation for the Cox proportional hazards model with time-dependent covariates: a method for medical researchers. Statis-tics in Medicine 33, 436-454.

[8] Henningsen, A. and Toomet, O. (2011). maxLik: A package for maximum likelihood estimation in R. Computational Statistics 26, 443-458.

[9] Mantovani, A. and Franco, M. A. P. (2004). A study on the asymptotic dis-tribution of maximum likelihood estimators for a two-parameter Weibull distribution in censored samples. Biometric Brazilian Journal 22, 7-20.

[10] Mori~na, D. and Navarro, A. (2014). The R package survsim for the simulation of simple and complex survival data. Journal of Statistical Software 59, 1-20.

[11] Royston, P. (2012). Tools to simulate realistic censored survival-time distribu-tions. The Stata Journal 12, 639-654

Edson Zangiacomi Martinez

Jorge Alberto Achcar

Department of Social Medicine Ribeir~ao Preto Medical School University of S~ao Paulo, USP Ribeir~ao Preto, SP, Brazil edson@fmrp.usp.br, achcar@fmrp.usp.br

Marcos Vinicius de Oliveira Peres

ose Andre Mota de Queiroz

Master Program in Biostatistics Department of Statistics
State University of Maringa,UEM,Maringa,PR,Brazil
marcosperes1991@hotmail.com, joseandrequeiroz@yahoo.com