R Package: TBSSurvival

Adriano Polpo Cassio de Campos Debajyoti Sinha
Stuart Lipsitz Jianchang Lin

April 17, 2013

Abstract

The purpose of this text is to provide a simple explanation about the main features of `TBSSurvival` package for `R` language. In short, we give some examples on how to use the package.

Keywords: TBSSurvival, R package, Generalized Power Transformation, reliability, error distribution, median estimation.

1 Introduction

We assume the reader to be familiar with reliability and/or survival estimation. For details we suggest the technical paper: *Transform Both Sides Model: A Parametric Approach*, Polpo et al. 2013. In this document we use as example a reliability analysis situation, but the methods will work just as fine with survival analysis.

The problem targeted here can be defined by `time`, an array of times of failure (survival) or components (patients) and `delta` the event indicator, that is, `delta` equals to one if an event happened at that time, or zero in case of right-censoring. Furthermore, covariates can be present in the data set, which can be used in a regression analysis, in the very same way as done by well-known functions such as `survreg`. First, we exemplify the use of the methods without covariates.

Example 1 In this example we simulate a data set with right-censored failure time. The problem regards an experiment with 30 machines. We observe the failure time of these machines, however our experiment has a maximum observation time of 6 week. After that, all machines that have not failed yet are considered to be censored with respect to their failure time. For simplicity, we define the failure time by a random variable with the Gamma distribution.

```
> time <- pmin(rgamma(30,10,2),rep(6,30))
> delta <- rep(1,30)
> for (i in 1:30) {
+   if (time[i] == 6) delta[i] <- 0
+ }
> data <- cbind(time,delta)
> data
```

```
          time delta
 [1,] 1.373189     1
 [2,] 3.552366     1
 [3,] 5.701566     1
 [4,] 5.136519     1
 [5,] 4.582540     1
 [6,] 4.871431     1
 [7,] 3.947642     1
 [8,] 5.026587     1
 [9,] 5.131757     1
[10,] 5.463954     1
[11,] 1.963734     1
[12,] 3.669229     1
[13,] 4.741880     1
[14,] 6.000000     0
[15,] 4.803878     1
[16,] 4.865529     1
[17,] 5.918272     1
[18,] 3.819599     1
[19,] 2.842224     1
[20,] 4.703623     1
[21,] 6.000000     0
[22,] 3.099829     1
[23,] 6.000000     0
[24,] 5.143568     1
[25,] 6.000000     0
[26,] 5.755176     1
[27,] 1.319985     1
[28,] 5.023511     1
```

```
[29,] 4.687110     1
[30,] 5.037901     1
```

2 TBS Model

The TBS Model is defined as

$$g_\lambda(\log(T)) = g_\lambda(\theta(\mathbf{X})) + \varepsilon, \tag{1}$$

where the function $g_\lambda(\cdot)$ is the TBS defined by

$$g_\lambda(u) = \frac{\mathrm{sign}(u)|u|^\lambda}{\lambda}, \tag{2}$$

$\mathrm{sign}(u) = 1$ if $u \geq 0$ and $\mathrm{sign}(u) = -1$ if $u < 0$, $\lambda > 0$, $\theta(\mathbf{X})$ is a function of co-variables and the error has some distribution ($\varepsilon \sim F_\varepsilon$) with parameter ξ. The error distributions currently implemented in the TBSSurvival package are: normal (used with `dist=dist.error('norm')`, t-student (`dist.error('t')`), Cauchy (`dist.error('cauchy')`), double-exponential (`dist.error('doubexp')`) and logistic (`dist.error('logistic')`) distributions. It is also possible to simply use the name of the distribution in the function call. Before continuing, in case you have not yet done so, the first thing to do before using the functions is to install and load the library.

```
> install.packages("TBSSurvival_VERSION.tar.gz",
+                  repos=NULL,type="source") ## from local file
> install.packages("TBSSurvival")  ## or from CRAN

> library("TBSSurvival")
```

Example 2 Density, distribution, quantile, random generation and hazard functions for the TBS model are available using the commonly used notation: `dtbs, ptbs, qtbs, rtbs, htbs`. The error distribution is passed as argument, as well some other parameters such as λ (for the TBS), ξ (parameter of the error distribution) and covariates' weights $\boldsymbol{\beta}$.

```
> x <- seq(0.1,10,0.1)
> plot(x, (1-ptbs(x,lambda=1,xi=sqrt(2),beta=1,
+                 dist=dist.error("norm"))),
+       type="l",lwd=2,lty=1,col=2,ylim=c(0,1),xlab="t",
+       ylab="R(t)",main="Reliability",cex.lab=1.2)
```

```
> plot(x, (dtbs(x,lambda=1,xi=sqrt(2),beta=1,
+                 dist=dist.error("norm"))),
+       type="l",lwd=2,lty=1,col=2,xlab="t",
+       ylab="f(t)",main="Density",cex.lab=1.2)

> plot(x, (htbs(x,lambda=1,xi=sqrt(2),beta=1,
+                 dist=dist.error("norm"))),
+       type="l",lwd=2,lty=1,col=2,xlab="t",
+       ylab="h(t)",main="Hazard",cex.lab=1.2)
```

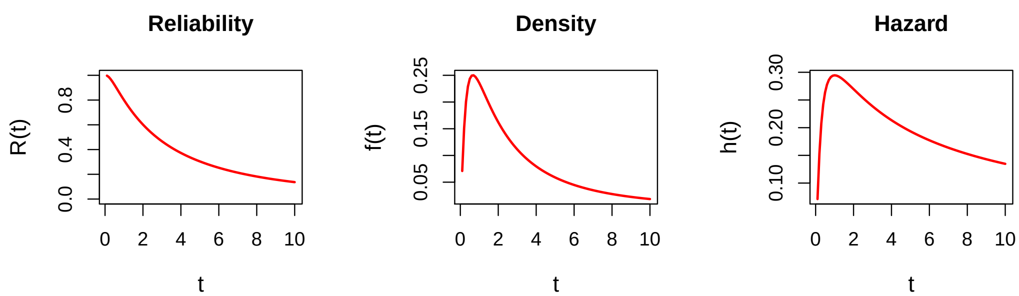

Figure 1: Density, reliability and Hazard functions
($\lambda = 1$; $\epsilon \sim \text{Normal}(0, 2^2)$; $\beta_0 = 1$).

```
> x <- seq(0.1,10,0.1)
> plot(x, (1-ptbs(x, lambda=1, xi=1, beta=1, dist=dist.error("doubexp"))),
+       type="l", lwd=2, lty=1, col=2, ylim=c(0,1), xlab="t",
+       ylab="R(t)", main="Reliability", cex.lab=1.2)

> plot(x, (dtbs(x, lambda=1, xi=1, beta=1, dist=dist.error("doubexp"))),
+       type="l", lwd=2, lty=1, col=2, xlab="t", ylab="f(t)",
+       main="Density", cex.lab=1.2)

> plot(x, (htbs(x, lambda=1, xi=1, beta=1,
+                       dist=dist.error("doubexp"))), type="l", lwd=2,
+       lty=1, col=2, xlab="t", ylab="h(t)",
+       main="Hazard", cex.lab=1.2)
```

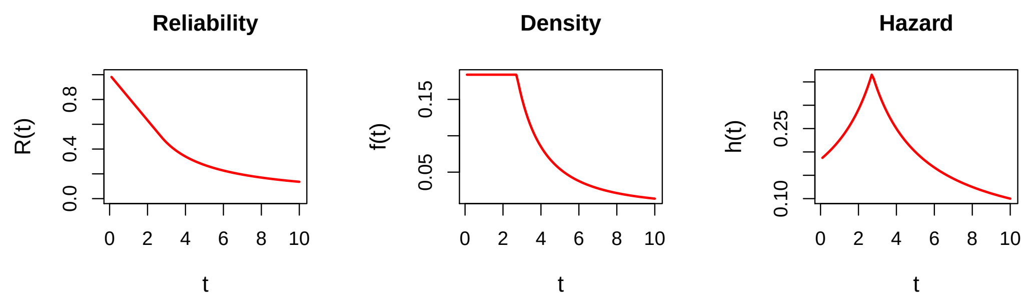

Figure 2: Density, reliability and Hazard functions
($\lambda = 1$; $\epsilon \sim$ doubexp(1); $\beta_0 = 1$).

2.1 Error distribution

Table 1 presents the available error distributions. Note that the meaning of the parameter changes according to the distribution being used. Moreover, for the normal distribution, the parameter is σ (standard deviation) and not σ^2 (variance). Besides these implemented functions, the user can easily specify their own to be used within TBS, as long as the distribution is symmetric unimodal, centered at zero, and has a single parameter (multiple parameters will be implemented in the future). To do so, instead of using `dist.error` as argument of the TBS functions, one can replace this call with a 5-element list containing: (density function, distribution function, quantile function, random generation function, and a name string). In fact, the function `dist.error` simply constructs such a list:

```
> mynormal = list(
+   d = function(x,xi) dnorm(x,mean=0,sd=sqrt(xi)), # density
+   p = function(x,xi) pnorm(x,mean=0,sd=sqrt(xi)), # distr
+   q = function(x,xi) qnorm(x,mean=0,sd=sqrt(xi)), # quantile
+   r = function(x,xi) rnorm(x,mean=0,sd=sqrt(xi)), # generation
+   name = "norm"
+   )
```

Table 1: Internally implemented error distributions.

Distribution	Parameter	Density function ($f_\varepsilon(\epsilon\|\xi)$)
Normal	$\xi = \sigma$	$(2\pi\sigma^2)^{-1/2}\exp\left\{-\epsilon^2/(2\sigma^2)\right\}$
DoubExp	$\xi = b$	$(2b)^{-1}\exp\left\{-\|\epsilon\|/b\right\}$
t-Student	$\xi = \eta(d.f.)$	$\frac{\Gamma((\eta+1)/2)}{\Gamma(\eta/2)\sqrt{\pi\eta}}\left(1+\frac{\epsilon^2}{\eta}\right)^{-(\eta+1)/2}$
Cauchy	$\xi = c$	$\left[\pi c\left(1+(\epsilon/c)^2\right)\right]^{-1}$
Logistic	$\xi = s$	$\frac{\exp\{-\epsilon/s\}}{s\left[(1+\exp\{-\epsilon/s\})^2\right]}$

The parametric space for ξ is $(0, +\infty)$ in all cases.

3 Estimation

The most important piece of code in the package is the parameter estimation of the TBS model. There are two procedures available: Maximum Likelihood Estimation (MLE) and Bayesian Estimation (BE). For illustration purposes, we will perform the estimation with both methods for the simulated data in Example 1. Remember that we defined the observed failure time as `time` and the censor indicator δ as `delta`.

3.1 MLE

In order to perform the estimation, following the common practice with other survival analysis packages, the user has to build a formula with time and censoring indication (delta), which is then used to call the estimation method itself:

```
> formula <- Surv(data[,1],data[,2]) ~ 1
> tbs.mle <- tbs.survreg.mle(formula,dist=mynormal, nstart=3,
+                            method="Nelder-Mead")
> tbs.mle
```

```
--------------------------------------------------------
 TBS model with norm error distribution (MLE).

         Estimates Std. Error
 lambda:    2.3710     0.5883
     xi:    0.2881     0.1178
```

```
   beta:    1.5385    0.0576

    AIC: 110.2771
   AICc: 111.2002
    BIC: 114.4807

Run time: 0.00 min
--------------------------------------------------------
```

Here, we compare the TBS model with the non-parametric Kaplan-Meier estimator. The result is presented in the Figure 3.

```
> # Kaplan-Meier estimation
> km <- survfit(formula)
> plot(tbs.mle,lwd=2,col="gray20",ylab="R(t)",
+      xlab="t: number of cycles (in thousands)",
+      main="Reliability function (MLE)",cex.lab=1.2)
> lines(km)
```

In the following, we present a simple example on how to define covariates to be used with TBS. In fact, we use the same framework of formulas as other survival packages. We also point out that different optimization methods are available in order to maximize the likelihood with the TBS model. If the argument `method` is not specified, all of them will be tried, and eventually the best one will be returned. The output, stored in the variable `s` in the following example, contains some score criteria and other useful information. Furthermore, standard R functions such as `summary(s)`, `print(s)` and `plot(s)` are all implemented to facilitate the visualization of the results.

```
> library(survival)
> data(colon)
> ## Running MLE on colon (from survival package) with a covariate
> colon$age60=as.numeric(colon$age>60) #threshold defined from medical papers
> s=tbs.survreg.mle(Surv(colon$time,colon$status==1) ~ colon$age60,
+       dist=mynormal,method=c("Nelder-Mead"),nstart=3,verbose=FALSE)
> summary(s)
```

3.2 BE

When dealing with the Bayesian estimation, it is necessary to consider an informative choice of priors, as well as the convergence rate of the MCMC

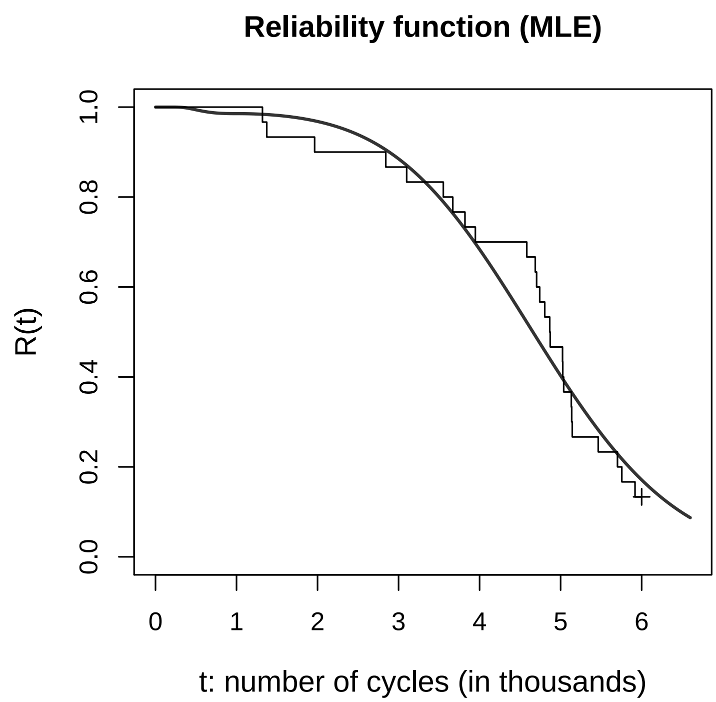

Figure 3: TBS model estimation result using MLE.

method, on which the estimation is based. However, we have experimented with some reasonable non-informative priors, which have shown adequate results. The user is entitled to choose `prior.mean` and `prior.sd` for the normal distribution that is used as prior for the $boldsymbol\beta$, a `scale` parameter, and initial guesses for the previously mentioned parameters of the model. Other well known arguments that are used my the Metropolis Hastings can also be set. We refer to the man pages/help files and the technical paper for more details about the priors. All these parameters can also be left unspecified, in which case some default values will be used.

```
> tbs.be <- tbs.survreg.be(formula,dist=mynormal,
+                          guess.lambda=2,guess.xi=4,
+                          guess.beta=1.5,burn=1000,
+                          jump=10,size=1000,scale=0.06)
```

As a visual example, we construct the plots with the estimated reliability and the 95% credible interval of High Posterior Density of the reliability function. The result is presented in the Figure 4.

```
> plot(tbs.be,ylab="R(t)",
+       xlab="t: number of cycles (in thousands)",
+       main="Reliability function (BE)",
+       lwd=2,lty=1,col=2,lwd.HPD=2,lty.HPD=2,col.HPD=2)
> lines(km)
```

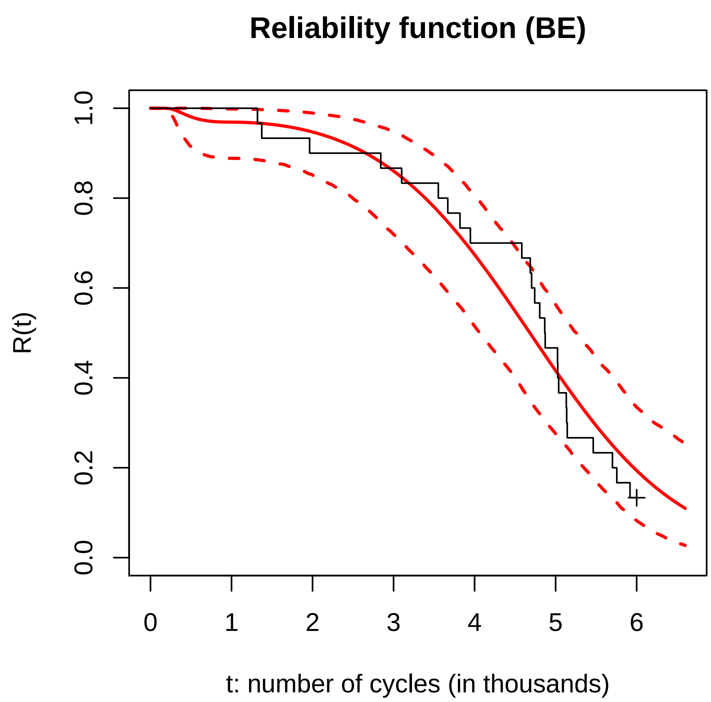

Figure 4: TBS model (BE).

4 Remarks

This "manual" describes the basics of the TBSSurvival package, which is currently developed around the estimation methods. We invite the user to the functions' help pages (available with the package) and to the technical paper mentioned in the beginning of this document for further details.