Package 'bshazard'

Type Package

Title Nonparametric Smoothing of the Hazard Function

Version 1.2

Date 2024-05-10

Author Paola Rebora,Agus Salim, Marie Reilly

Maintainer Paola Rebora <paola.rebora@unimib.it>

Depends R(>= 3.3.3),splines,survival,Epi

Description The function estimates the hazard function non parametrically from a survival object (possibly adjusted for covariates).
The smoothed estimate is based on B-splines from the perspective of generalized linear mixed models. Left truncated and right censoring data are allowed.
The package is based on the work in Rebora P (2014) <doi:10.32614/RJ-2014-028>.

License GPL-2

RoxygenNote 7.3.1

NeedsCompilation no

Repository CRAN

Date/Publication 2024-05-12 21:03:32 UTC

R topics documented:

---

bshazard-package   *Nonparametric Smoothing of the Hazard Function*

Description

The function estimates the hazard function non parametrically from a survival object (possibly adjusted for covariates). The smoothed estimate is based on B-splines from the perspective of generalized linear mixed models. Left truncated and right censoring data are allowed.

Details

The DESCRIPTION file:

Package:	bshazard
Type:	Package
Title:	Nonparametric Smoothing of the Hazard Function
Version:	1.2
Date:	2024-05-10
Author:	Paola Rebora,Agus Salim, Marie Reilly
Maintainer:	Paola Rebora <paola.rebora@unimib.it>
Depends:	R(>= 3.3.3),splines,survival,Epi
Description:	The function estimates the hazard function non parametrically from a survival object (possibly adjusted
License:	GPL-2
RoxygenNote:	7.3.1
NeedsCompilation:	no
Packaged:	2024-05-10 11:06:47 UTC; paola.rebora
Repository:	CRAN
Date/Publication:	2024-05-10 10:01:45 UTC

Index of help topics:

```
bshazard              Nonparametric Smoothing of the Hazard Function
bshazard-package      Nonparametric Smoothing of the Hazard Function
bspois.basic          Functions for internal use of bshazard
plot.bshazard         Plot Method for 'bshazard'
print.bshazard        Print a short summary of the hazard rate
summary.bshazard      Summary of hazard curve
```

Author(s)

Paola Rebora, Agus Salim, Marie Reilly Maintainer: Paola Rebora <paola.rebora@unimib.it>

References

Rebora P, Salim A, Reilly M (2014) bshazard: A Flexible Tool for Nonparametric Smoothing of the Hazard Function.The R Journal Vol. 6/2:114-122.

Lee Y, Nelder JA, Pawitan Y (2006). Generalized Linear Models with Random Effects: Unified Analysis via H-likelihood, volume 106. Chapman & Hall/CRC.

Pawitan Y (2001). In All Likelihood: Statistical Modelling and Inference Using Likelihood. Oxford University Press

Examples

```
data(cancer,package="survival")
  fit<-bshazard(Surv(time, status==2) ~ 1,data=lung)
  plot(fit)
```

---

bshazard	*Nonparametric Smoothing of the Hazard Function*

Description

The function estimates the hazard function non parametrically from a survival object (possibly adjusted for covariates). The smoothed estimate is based on B-splines from the perspective of generalized linear mixed models. Left truncated and right censoring data are allowed.

Usage

```
## Default S3 method:
bshazard(formula,data,nbin=NULL,nk,degree,l0,lambda,phi,alpha,err,verbose)
```

Arguments

formula	a formula object, which must have a Surv object as the response on the left of the ~ operator. On the right, if desired are the names of the covariates in the Poisson model separated by + operators. For unadjusted estimates the right hand side should be ~1.
data	a data frame containing the variables named in the formula.
nbin	number of bins (equally spaced). If omitted, the function will split the data at each distinct time in the data (censoring or event).
nk	number of knots for B-splines (including minimum and maximum, default is 31): as a rule of thumb the number of observations divided by 4, but more than 40 knots are rarely needed.
degree	degree of B-splines, representing the effective number of parameters in the piecewise segments of splines (default is 1, corresponding to a linear segment)
l0	Starting value for the smoothing parameter lambda (default is 10). Relevant only if lambda=NULL
lambda	smoothing parameter. If not provided (default is NULL) it is estimated from the data as phi times the reciprocal of the variance of the random effect (by an iterative algorithm). A high value imposes a smoother estimate.

phi	overdispersion parameter. If not provided (default is NULL) it is estimated from the data.
alpha	1 minus the level for the two-sided confidence interval for the hazard function. Default is 0.05.
err	relative error for the iterative process in the lambda/phi estimation (default is 0.0001)
verbose	TRUE/FALSE Print each iteration step (default is T)

Details

The time axis is partitioned in a number of intervals equal to the number of different observations (if not fixed otherwise by the option nbin). The number of events in each interval is modeled by a Poisson model and the smoothing parameter (lambda) is estimated by a mixed model framework. The number of knots is 31 by default. The code also allows for overdispersion (phi). Adjustment for covariates can increase the computation time.

Value

The output of the bshazard function can be divided into three parts: 1. a data frame with the original data split in time intervals, 2. the set of vectors containing the estimated hazard and confidence limits and 3. the parameter estimates.

raw.data	data frame with original data split in bins (intervals of time), containing:
time	mid point of each bin
n.event	number of events in each bin
PY	total person-time in each bin (for each covariate value)
raw.hazard	number of events in each interval divided by PY
...	covariate values
nobs	number of records in input data
time	mid point of each bin at which the curve is computed
hazard	hazard estimate for each bin
lower.ci	lower limit of the hazard confidence interval (depends on alpha level)
upper.ci	upper limit of the hazard confidence interval (depends on alpha level)
(cov.value)	values of covariates at which hazard is computed (mean value)
fitted.values	estimated number of events at each time (from the Poisson model)
coefficients	coefficient estimates for each covariate
phi	overdispersion parameter
sv2	variance of the random effect corresponding to the inverse of the smoothing parameter multiplied by phi
df	degrees of freedom representing the effective number of smoothing parameters

Author(s)

Paola Rebora, Agus Salim, Marie Reilly Maintainer: Paola Rebora <paola.rebora@unimib.it>

References

Rebora P, Salim A, Reilly M (2014) bshazard: A Flexible Tool for Nonparametric Smoothing of the Hazard Function.The R Journal Vol. 6/2:114-122.

Lee Y, Nelder JA, Pawitan Y (2006). Generalized Linear Models with Random Effects: Unified Analysis via H-likelihood, volume 106. Chapman & Hall/CRC.

Pawitan Y (2001). In All Likelihood: Statistical Modelling and Inference Using Likelihood. Oxford University Press

Examples

```
data(cancer,package="survival")
  fit<-bshazard(Surv(time, status==2) ~ 1,data=lung)
  plot(fit$time, fit$hazard,xlab='Time', ylab='Rate/person-days',type='l')
  points(fit$raw.data$time,fit$raw.data$raw.hazard,cex=.3, lwd=3,col=1)
  lines(fit$time, fit$hazard, lty=1, lwd=3)
  lines(fit$time, fit$lower.ci, lty=1, lwd=1)
lines(fit$time, fit$upper.ci, lty=1, lwd=1)
# or alternatively
plot(fit)


#Example to graphically evaluate the Markov assumpion in an illness-death model
### data simulated under EXTENDED SEMI-MARKOV model
set.seed(123)
n  <- 500
beta<-log(3)
R  <- runif(n, min = 0, max =2)
cat <- cut(R, breaks = seq(0,2,0.5), labels = seq(1,4,1))
T12  <- 1/0.2*( (-log(runif(n, min = 0, max = 1))) / (exp(beta*R)))**(1/0.6)
C  <- runif(n, min =0, max = 10)
T12obs <- pmin(T12, C)
status  <- ifelse(T12obs < C, 1, 0)
T012obs <- R+T12obs
#R: simulated time to illness
#cat: time to illness categorised in 4 classes
#T12obs:  time from illness to death or censoring
#T012obs: time from origin to death or censoring
#status: indicator of death (1=death,0=censoring)

# Hazard estimate in the Clock Reset scale (time from illness) by time to illness class
fit <- bshazard(Surv(T12obs[cat == 1],status[cat==1]) ~ 1)
plot(fit,conf.int=FALSE,xlab='Time since illness',xlim=c(0,5),ylim=c(0,10),lwd=2,col=rainbow(1))
for(i in 2:4) {
  fit <- bshazard(Surv(T12obs[cat == i],status[cat==i]) ~ 1)
  lines(fit$time, fit$hazard, type = 'l', lwd = 2, col = rainbow(4)[i])
}
legend("top",title="Time to illness",c("<=0.5","0.5-|1","1-|1.5","1.5-|2"),col=c(rainbow(4)),lwd =2)

# Hazard estimate in the Clock Forward scale (time from origin) by time to illness class
fit <- bshazard(Surv(R[cat == 1],T012obs[cat == 1],status[cat==1]) ~ 1)
plot(fit,conf.int=FALSE,xlab='Time since origin',xlim=c(0,5),ylim=c(0,10),lwd=2,col=rainbow(1))
```

```
for(i in 2:4) {
  fit <- bshazard(Surv(R[cat == i],T012obs[cat == i],status[cat==i]) ~ 1)
  lines(fit$time, fit$hazard, type = 'l', lwd = 2, col = rainbow(4)[i])
}
legend("top",title="Time to illness",c("<=0.5","0.5-|1","1-|1.5","1.5-|2"),col=c(rainbow(4)),lwd =2)

#Alternatively an adjusted estimate can be performed, assuming proportionl hazard.
##This computation can be time consuming!
## Not run: fit.adj <- bshazard(Surv(T12obs,status) ~ cat )
plot(fit.adj,overall=FALSE, xlab = 'Time since illness',col=rainbow(4),lwd=2, xlim = c(0,5),
ylim = c(0,10))
legend("top",title="Time to illness",c("<=0.5","0.5-|1","1-|1.5","1.5-|2"),col=c(rainbow(4)),lwd =2)
## End(Not run)

### A more general setting with examples of Markov assumption evaluation can be found
### in Bernasconi et al. Stat in Med 2016
## The function is currently defined as
function (x, ...)
UseMethod("bshazard")
```

---

bshazard-internal   *Functions for internal use of bshazard*

Description

Functions for internal use of bshazard

Author(s)

Paola Rebora

---

plot.bshazard   *Plot Method for 'bshazard'*

Description

A plot of hazard rate is produced. The overall option allows to plot an hazard rate for each covariate value (assuming proportional hazard).

Usage

```
## S3 method for class 'bshazard'
plot(x, conf.int = T, overall = T, col = 1, lwd = 1, lty = 1, xlab = "Time",
ylab = "Hazard rate",border=NA,col.fill="lightgrey",...)
```

Arguments

x	the result of a call to the bshazard function.
conf.int	Determines whether confidence intervals will be plotted. The default is to do so if there is only 1 curve, i.e., no strata.
overall	Determines whether an overall curve is plotted (default overall=T) or a curve for each covariate value in the data (overall=F). It works only if there are covariates.
col	a vector of integers specifying colors for each curve. The default value is 1.
lwd	a vector of integers specifying line width for each curve. The default value is 1.
lty	a vector of integers specifying line types for each curve. The default value is 1.
xlab	label given to the x-axis.
ylab	label given to the y-axis.
border	the color to draw the border. The deafult value is NA.
col.fill	the color for filling the polygon. The default is lightgrey.
...	other arguments that will be passed forward to the underlying plot method, such as xlab or ylab.

See Also

bshazard,summary.bshazard,print.bshazard

Examples

```
fit<-bshazard(Surv(time, status==2) ~ 1,data=lung)
plot(fit)


#Example to graphically evaluate the Markov assumpion in an illness-death model
### data simulated under EXTENDED SEMI-MARKOV model
set.seed(123)
n  <- 500
beta<-log(3)
R  <- runif(n, min = 0, max =2)
cat <- cut(R, breaks = seq(0,2,0.5), labels = seq(1,4,1))
T12  <- 1/0.2*( (-log(runif(n, min = 0, max = 1))) / (exp(beta*R)))**(1/0.6)
C  <- runif(n, min =0, max = 10)
T12obs <- pmin(T12, C)
status  <- ifelse(T12obs < C, 1, 0)
T012obs <- R+T12obs
#R: simulated time to illness
#cat: time to illness categorised in 4 classes
#T12obs:  time from illness to death or censoring
#T012obs: time from origin to death or censoring
#status: indicator of death (1=death,0=censoring)

# Hazard estimate in the Clock Reset scale (time from illness) by time to illness class
fit <- bshazard(Surv(T12obs[cat == 1],status[cat==1]) ~ 1)
plot(fit,conf.int=FALSE,xlab='Time since illness',xlim=c(0,5),ylim=c(0,10),lwd=2,col=rainbow(1))
```

```
for(i in 2:4) {
  fit <- bshazard(Surv(T12obs[cat == i],status[cat==i]) ~ 1)
  lines(fit$time, fit$hazard, type = 'l', lwd = 2, col = rainbow(4)[i])
}
legend("top",title="Time to illness",c("<=0.5","0.5-|1","1-|1.5","1.5-|2"),col=c(rainbow(4)),lwd =2)

# Hazard estimate in the Clock Forward scale (time from origin) by time to illness class
fit <- bshazard(Surv(R[cat == 1],T012obs[cat == 1],status[cat==1]) ~ 1)
plot(fit,conf.int=FALSE,xlab='Time since origin',xlim=c(0,5),ylim=c(0,10),lwd=2,col=rainbow(1))
for(i in 2:4) {
  fit <- bshazard(Surv(R[cat == i],T012obs[cat == i],status[cat==i]) ~ 1)
  lines(fit$time, fit$hazard, type = 'l', lwd = 2, col = rainbow(4)[i])
}
legend("top",title="Time to illness",c("<=0.5","0.5-|1","1-|1.5","1.5-|2"),col=c(rainbow(4)),lwd =2)

#Alternatively an adjusted estimate can be performed, assuming proportionl hazard
# this computation can be time consuming!
## Not run: fit.adj <- bshazard(Surv(T12obs,status) ~ cat )
plot(fit.adj,overall=FALSE, xlab = 'Time since illness',col=rainbow(4),lwd=2, xlim = c(0,5),
ylim = c(0,10))
legend("top",title="Time to illness",c("<=0.5","0.5-|1","1-|1.5","1.5-|2"),col=c(rainbow(4)),lwd =2)
## End(Not run)

### A more general setting with examples of Markov assumption evaluation can be found
### in Bernasconi et al. Stat in Med 2016
```

| print.bshazard | *Print a short summary of the hazard rate* |

Description

Print number of observations, number of events, total person-time and overall rate of event.

Usage

```
## S3 method for class 'bshazard'
print(x,...)
```

Arguments

| x | the result of a call to the bshazard function. |
| ... | other arguments that will be passed forward |

Value

Number of records

number of records in input data

...   covariate values

`n.events`	total number of events (for each covariate value)
`person.time`	total person-time (for each covariate value)
`rate`	overall rate

See Also

bshazard,summary.bshazard,plot.bshazard

Examples

```
data(cancer,package="survival")
 fit<-bshazard(Surv(time, status==2) ~ 1,data=lung)
print(fit)
```

---

summary.bshazard	*Summary of hazard curve*

Description

Returns a list containing the hazard curve and confidence limits for the curve.

Usage

```
## S3 method for class 'bshazard'
summary(object, digits = 4,...)
```

Arguments

`object`	the result of a call to the bshazard function.
`digits`	Number of digits to print
`...`	other arguments that will be passed forward

Value

`(covariate.value)`	
	values of covariates at which hazard is computed (mean value)
`time`	mid point of each bin at which the curve is computed
`hazard`	hazard estimate for each bin
`lower.ci`	lower limit of the hazard confidence interval (depends on alpha level)
`upper.ci`	upper limit of the hazard confidence interval (depends on alpha level)
`lambda`	smoothing parameter
`df`	degrees of freedom representing the effective number of smoothing parameters
`phi`	overdispersion parameter

See Also

bshazard,print.bshazard,plot.bshazard

Examples

```
data(cancer,package="survival")
 summary(bshazard(Surv(time, status==2) ~ 1,data=lung))
```

Index