Package 'SplineHazardRegression'

June 16, 2023

Type Package

Title R code for spline hazard regression in survival analysis

Version 0.1.0

Author Ferran Espuny Pujol, Philip Rosenberg

Maintainer Ferran Espuny Pujol <f.pujol@ucl.ac.uk>

Description This code implements and further develops the methods from Philip S Rosenberg. ``Hazard Function Estimation Using B-Splines". \href{https://doi.org/10.2307/2532989}{Biometrics, Vol. 51, No. 3 (Sep., 1995), pp. 874-887}.

License GPL-3

Encoding UTF-8

LazyData true

URL https://github.com/fespuny/SplineHazardRegression

BugReports https://github.com/fespuny/SplineHazardRegression/issues

RoxygenNote 7.2.3

Depends splines2, pracma

R topics documented:

`bspline_regression_basis_functions`
Generate the basis functions needed for fitting the hazard B-spline model

---

Description

This function calculates all matrices needed for the estimation of a hazard function with censored time-to-event data using B-splines

Usage

```
bspline_regression_basis_functions(yd, entry, ORDER, knots, t)
```

Arguments

yd	- matrix of time-to-event data; pneumonic y is event time, d is for delta, the status indicator
entry	- late entry / left truncation times
ORDER	- 1 step, 2 linear, 3 quadratic, 4 cubic
knots	- sequence of knot locations
t	- vector of evaluation times

Details

When fitting a B-spline function $h(\alpha)(t) := B(t)\alpha$ to time-to-event-data, where $B(t)$ is a basis of B-splines and α are the coefficients to be estimated, we optimise a function of the likelihood

$$L(h(\theta)) =$$

.

Value

list(Wik, Zik, Eik, XH, Xh) - see details for definitions

---

etsim *Simulate time-to-event data*

Description

Generates survival time and status fields using hazard and censoring as inputs

Usage

```
etsim(INPUTS)
```

Arguments

INPUTS	- hazard and censoring distributions as generated by etsim_inputs()

Details

Times to event are defined as $t = \min(t_h, t_C)$, where t_h is the time to hazard (distributed according to the input S_h) and t_C is the time to censoring (distributed according to the input $S_{censoring}$)

Value

Returns a list of outputs (time, status, entry)

t = array of times to event status = 0 if alive at time t, 1 otherwise entry = NULL [IS IT OK TO IGNORE LATE ENTRY AT THIS STAGE?]

Examples

```
#Generate input for a b-spline hazard and piecewise exponential survival censoring
knots = c(0, 1, 3, 6, 10, NA, NA)
betac = 1 * c(0.05, 0.05, 0.05, 0.05, 0.40, 0.1, 0.05)
HParm = data.frame(knots, betac) # 'A Simple B-Spline'
cll = c(0, 5)
cup = c(5, 10)
cih = c(0.0125, 0.025)
CParm = data.frame(cll, cup, cih) # 'Light Censoring'
INPUTS = etsim_inputs( HParam=HParm, CParam=CParm)

#Then, we can generate the time-to event data
SimDat = etsim(INPUTS)
table( SimDat$status )
```

`etsim_inputs` *Generate hazard and censoring distributions*

Description

Generates hazard and censoring distributions and respective survival functions [IS IT OK TO IGNORE LATE ENTRY AT THIS STAGE?]

Usage

```
etsim_inputs(
  Hazard = "spline",
  HParam,
  Censor = "pe",
  CParam,
  Tmax = 10,
  SampleSize = 101
)
```

Arguments

`Hazard`	- hazard function for outcome variable. 'exp' for exponential, 'weib' for Weibull, 'pe' for piecewise-exponential, or 'spline' for B-Spline
`HParam`	- matrix of parameter values for outcome variable
`Censor`	- censoring hazard ('', 'exp', 'weib', 'pe' or 'bspline')
`CParam`	- matrix of parameter values for censoring variable
`Tmax`	- scalar, maximum follow-up time, default 10
`SampleSize`	- scalar, sample size

Details

For 'exp', HParm contains a scalar; For 'pe', HParm(:, 1) is left limit, HParm(:, 2) is right limit, and HParm(:, 3) is hazard over the interval For 'bspline', HParm(:, 1) lists knots, HParm(:, 2) lists spline coefficients. The number of rows of Hparm has to be equal to the number of degrees of freedom: number of interior knots plus order (=degree plus one) of the b-spline

Value

Returns a list of outputs (t, basis, h, hCensor, Sh, Scensor)

t = array of times basis = basis of b-spline cubic functions [WE NEED TO ADD ORDER/DEGREE AS PARAMETER IF WE WANT TO ALLOW DIFFERENT B-SPLINES] h = simulated hazard distribution hCensor = simulated censoring distribution Sh = cumulative hazard survival function SCensor = cumulative censoring survival funtion

Examples

```
#Generate input for a b-spline hazard simulation with piecewise exponential survival censoring
 knots = c(0, 1, 3, 6, 10, NaN, NaN )
 betac = 1 * c(0.05, 0.05, 0.05, 0.05, 0.40, 0.1, 0.05)
 HParm = data.frame(knots, betac) # 'A Simple B-Spline'
 cll = c(0, 5)
 cup = c(5, 10)
 cih = c(0.0125, 0.025)
 CParm = data.frame(cll, cup, cih) # 'Light Censoring'
 INPUTS = etsim_inputs( HParam=HParm, CParam=CParm )
```

---

generate_bspline_basis

Generate B-Spline basis

Description

Wraper function to a B-spline basis calculation function in R (currently from package bspline2)

Usage

```
generate_bspline_basis(time, Interior.knots, Boundary.knots, ORDER = 4)
```

Arguments

time	- x coordinates for the B-spline functions
Interior.knots	- interior knots of the B-spline basis
Boundary.knots	- boundary knots of the B-spline basis
ORDER	- 1 step, 2 linear, 3 quadratic, 4 cubic

Details

The current bSpline() function from the splines2 package extends the bs() function in the splines package for B-spline basis by allowing piecewise constant (left-closed and right-open except on the right boundary) spline basis of degree zero.

Value

B - matrix of basis of b-pline functions of the specified degree, evaluated at the "time" points B has dimensions length(time) x (number of interior points + degree + 1)

Examples

```
# A basis of cubic B-splines with no interior points
B = generate_bspline_basis( time = 0:10, Interior.knots=c(), Boundary.knots=c(0,10) )
B
# Note that B has 4=3+1 functions (as many as the order of cubic B-splines)
# we can plot the basis using matplot
matplot( 0:10, B, type="l")
```

`srllikb_fun` *Objective function for b-spline hazard regression*

Description

Calculates the objective function for b-spline hazard regression using as input the needed pre-calculated matrices

Usage

```
srllikb_fun(par.alpha, yd, Wik, Zik)
```

Arguments

`par.alpha`	- coefficients of a b-spline in a b-pline basis B (not provided)
`yd`	- time-to-event data observations matrix, with each row being a pair (time,status)
`Wik`	- matrix that multiplied by par.alpha gives the hazard function
`Zik`	- matrix that multiplied by par.alpha gives the cumulative hazard function

Value

l - objective function (-2 *log likelihood function)

Index