Package ‘AalenJohansen’

March 1, 2023

Type Package

Title Conditional Aalen-Johansen Estimation

Version 1.0

Maintainer Martin Bladt <martinbladt@math.ku.dk>

Description Provides the conditional Nelson-Aalen and Aalen-Johansen estimators. The methods are based on Bladt & Furrer (2023), in preparation.

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.2.1

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Martin Bladt [aut, cre],
Christian Furrer [aut]

Repository CRAN

Date/Publication 2023-03-01 10:42:09 UTC

R topics documented:

aalen_johansen *Compute the conditional Aalen-Johansen estimator.*

Description

Compute the conditional Aalen-Johansen estimator.

Usage

```
aalen_johansen(
  data,
  x = NULL,
  a = NULL,
  p = NULL,
  alpha = 0.05,
  collapse = FALSE
)
```

Arguments

data	A list of trajectory data for each individual.
x	A numeric value for conditioning.
a	A bandwidth. Default uses an asymmetric version using alpha.
p	An integer representing the number of states. The absorbing state is last.
alpha	A probability around the point x, for asymmetric sub-sampling.
collapse	Logical, whether to collapse the last state of the model.

Value

A list containing the Aalen-Johansen estimator, the Nelson-Aalen estimator, and related quantities.

prodint *Calculate the product integral of a matrix function*

Description

Calculate the product integral of a matrix function

Usage

```
prodint(start, end, step_size, lambda)
```

Arguments

`start` Start time.

`end` End time.

`step_size` Step size of the grid.

`lambda` A given matrix function.

Value

The product integral of the given matrix function.

`sim_path` *Simulate the path of a time-inhomogeneous (semi-)Markov process until a maximal time*

Description

Simulate the path of a time-inhomogeneous (semi-)Markov process until a maximal time

Usage

```
sim_path(i, rates, dists, t = 0, u = 0, tn = Inf, abs = numeric(0), bs = NA)
```

Arguments

`i` The initial state, integer.

`rates` The total transition rates out of states, a function with arguments state (integer), time (numeric), and duration (numeric) returning a rate (numeric).

`dists` The distribution of marks, a function with arguments state (integer), time (numeric), and duration (numeric) returning a probability vector.

`t` The initial time, numeric.

`u` The initial duration (since the last transition), numeric. By default equal to zero

`tn` The maximal time, numeric. By default equal to inifinity

`abs` Vector indicating which states are absorbing. By default the last state is absorbing.

`bs` Vector of upper bounds on the total transition rates. By default the bounds are determined using optimize, which might only identify a local maximum.

Value

A list concerning jump times and states, with the first time being the initial time t and state and the last time being tn (if not absorbed)

Examples

```
jump_rate <- function(i, t, u){if(i == 1){3*t} else if(i == 2){5*t} else{0}}
mark_dist <- function(i, s, v){if(i == 1){c(0, 1/3, 2/3)} else if(i == 2){c(1/5, 0, 4/5)} else{0}}
sim <- sim_path(sample(1:2, 1), t = 0, tn = 2, rates = jump_rate, dists = mark_dist)
sim
```

Index