Package 'IntervalCensoredMultistateR2'

August 6, 2026

Type Package

Title Regression Analysis in Interval-Censored Multistate Models

Version 1.0.0

Date 2026-07-28

Author Lu You [aut, cre]

Maintainer Lu You <lu.you@epi.usf.edu>

Description Estimates regression coefficients in proportional hazards models for interval-censored multistate data. Individuals may be observed at irregular times and their states may be partially observed. Allowable transitions and transition-specific covariate effects can be specified. The numerical estimation is implemented in 'C++' using 'RcppArmadillo'. The method implemented in this package is described in You, Liu, and Krischer (2024) <doi:10.1002/sim.10079>.

License GPL-3

Imports Rcpp (>= 1.0.11)

LinkingTo Rcpp, RcppArmadillo

Depends R (>= 3.6.0)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

LazyData true

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation yes

Repository CRAN

Date/Publication 2026-08-06 13:10:28 UTC

Contents

`examplesimple_competing_risks`
A simpler example for interval-censored competing risks dataset

---

Description

A simulated dataset (with three causes) used to illustrate interval-censored competing risks models.

Usage

```
examplesimple_competing_risks
```

Format

A list:

left A numeric vector of length 200. Left endpoint of the censoring interval

right A numeric vector of length 200. Right endpoint of the censoring interval

cause An integer vector of length 200. Event type indicator for competing risks. 0 indicates censoring; 1, 2, and 3 indicate event from causes 1, 2, and 3.

zzi A numeric matrix of dimension 200×3. Three covariates in the competing risks model.

F_zidx A 3-element list of integer vectors. ‘F_zidx[[i]]‘ contains the indices of covariates used in the model for cause ‘i‘.

Source

Simulated by the authors

examplesimple_multistate

A simpler example for interval-censored multistate dataset

Description

A simulated dataset (with three states) used to illustrate interval-censored multistate models.

Usage

```
examplesimple_multistate
```

Format

A list:

TTij A numeric matrix of dimension 200×20. ‘TTij[i,j]‘ is the ‘j‘th visit time of the ‘i‘th individual.

ssij A logical array of dimension 200×20×3. ‘ssij[i,j,s]‘ specifies if state ‘s‘ can be occupied at the ‘j‘th visit for the ‘i‘th individual.

nobss An integer vector of length 200. ‘nobss[i]‘ is the number of visits of the ‘i‘th individual.

zzi A numeric matrix of dimension 200×2. Three covariates in the multistate model.

F_zidx A 3×3 matrix of integer vectors. ‘F_zidx[[i,j]]‘ contains the indices of covariates used in the model for transitions from state ‘i‘ to state ‘j‘.

possible_transition A 3×3 logical matrix. ‘possible_transition[i,j]==TRUE‘ if and only if transitions from state ‘i‘ to state ‘j‘ are possible.

Source

Simulated by the authors

examplesimple_single_event

A simpler example for interval-censored single-event dataset

Description

A simulated dataset used to illustrate interval-censored single-event models.

Usage

```
examplesimple_single_event
```

Format

A data.frame:

left Left endpoint of the censoring interval.

right Right endpoint of the censoring interval.

event Event indicator. The event indicator equals 1 or TRUE if an event is observed; the event indicator equals 0 or FALSE if the event is right-censored.

X1 Covariate 1

X2 Covariate 2

X3 Covariate 3

Source

Simulated by the authors

`example_competing_risks`

Example interval-censored competing risks dataset

Description

A simulated dataset (with three causes) used to illustrate interval-censored competing risks models.

Usage

```
example_competing_risks
```

Format

A list:

left A numeric vector of length 500. Left endpoint of the censoring interval

right A numeric vector of length 500. Right endpoint of the censoring interval

cause An integer vector of length 500. Event type indicator for competing risks. 0 indicates censoring; 1, 2, and 3 indicate event from causes 1, 2, and 3.

zzi A numeric matrix of dimension 500×3. Three covariates in the competing risks model.

F_zidx A 3-element list of integer vectors. 'F_zidx[[i]]' contains the indices of covariates used in the model for cause 'i'.

Source

Simulated by the authors

example_multistate *Example interval-censored multistate dataset*

Description

A simulated dataset (with three states) used to illustrate interval-censored multistate models.

Usage

```
example_multistate
```

Format

A list:

TTij A numeric matrix of dimension 1000×25. ‘TTij[i,j]‘ is the ‘j‘th visit time of the ‘i‘th individual.

ssij A logical array of dimension 100×25×3. ‘ssij[i,j,s]‘ specifies if state ‘s‘ can be occupied at the ‘j‘th visit for the ‘i‘th individual.

nobss An integer vector of length 1000. ‘nobss[i]‘ is the number of visits of the ‘i‘th individual.

zzi A numeric matrix of dimension 1000×3. Three covariates in the multistate model.

F_zidx A 3×3 matrix of integer vectors. ‘F_zidx[[i,j]]‘ contains the indices of covariates used in the model for transitions from state ‘i‘ to state ‘j‘.

possible_transition A 3×3 logical matrix. ‘possible_transition[i,j]==TRUE‘ if and only if transitions from state ‘i‘ to state ‘j‘ are possible.

Source

Simulated by the authors

example_single_event *Example interval-censored single-event dataset*

Description

A simulated dataset used to illustrate interval-censored single-event models.

Usage

```
example_single_event
```

Format

A data.frame:

left Left endpoint of the censoring interval.

right Right endpoint of the censoring interval.

event Event indicator. The event indicator equals 1 or TRUE if an event is observed; the event indicator equals 0 or FALSE if the event is right-censored.

X1 Covariate 1

X2 Covariate 2

X3 Covariate 3

Source

Simulated by the authors

`fit_competing_risks` *Fit a proportional hazards model for interval-censored competing risks data*

Description

Estimates regression coefficients in a proportional hazards framework for interval-censored competing risks data. Each individual is observed within an interval `(left, right]`, with a recorded failure cause if an event occurs. Cause-specific hazards are modeled using transition-specific covariate effects. Computation is performed by the C++ backend.

Usage

```
fit_competing_risks(
  left,
  right,
  cause,
  zzi,
  F_zidx,
  timepoints,
  niter = 1000,
  tol = 1e-04,
  factor = 0.75
)
```

Arguments

`left`	Numeric vector of left endpoints of observation intervals. For the *i*-th individual, `left[i]` is the lower bound of the interval in which the event time lies. In cases of right-censoring, right-censoring), `left[i]` is the last follow-up time.
`right`	Numeric vector of right endpoints of observation intervals. For the *i*-th individual, `right[i]` is the upper bound of the interval in which the event time lies. In cases of right-censoring, `right[i]` = `Inf`.
`cause`	Integer vector indicating the event type (cause). `cause[i]` is the cause for the *i*-th individual. A value of 0 indicates right-censoring, while positive integers (1, 2, ..., K) indicate the cause of failure.
`zzi`	Numeric matrix of covariates. The *i*-th row corresponds to the *i*-th individual and each column to a predictor variable.
`F_zidx`	A list of integer vectors specifying covariate indices for each cause. `F_zidx[[k]]` (or the k-th element) contains the indices of covariates used to model the cause-specific hazard for cause `k`.
`timepoints`	Timepoints for discrete approximation. `max(timepoints)` should be greater than or equal to `max(right)`.
`niter`	Maximum number of iterations for the optimization algorithm. Default is `1000`.
`tol`	Convergence tolerance for the iterative estimation procedure. Default is `1e-4`.
`factor`	Step-size adjustment factor used in the optimization routine. Must be between 0 and 1. Default is `0.75`.

Details

This function fits a cause-specific proportional hazards model under interval censoring. For each failure cause, a separate hazard function is modeled with a log-linear dependence on selected covariates.

The R interface calls an RcppArmadillo implementation of the core numerical optimization.

Value

A list containing:

- `coef`: Estimated regression coefficients for each cause.
- `converged`: Logical indicator of convergence.
- `niter`: Number of iterations performed.

Examples

```
# A simpler example
library(IntervalCensoredMultistateR2)
data("examplesimple_competing_risks")
model_fit_competing_risks<-
  IntervalCensoredMultistateR2::fit_competing_risks(
    examplesimple_competing_risks$left,
    examplesimple_competing_risks$right,
```

```
    examplesimple_competing_risks$cause,
    examplesimple_competing_risks$zzi,
    examplesimple_competing_risks$F_zidx,
    seq(0.05,1,0.05))

# A more complicated example
library(IntervalCensoredMultistateR2)
data("example_competing_risks")

model_fit_competing_risks<-
  IntervalCensoredMultistateR2::fit_competing_risks(
    example_competing_risks$left,
    example_competing_risks$right,
    example_competing_risks$cause,
    example_competing_risks$zzi,
    example_competing_risks$F_zidx,
    seq(0.01,1,0.01))
```

`fit_multistate` *Fit a proportional hazards model for interval-censored multistate data*

Description

Estimates regression coefficients in a proportional hazards framework for interval-censored multistate data. Individuals start in state 1 at time 0 and are followed over time. At each observation time, the exact state may be uniquely observed or only partially observed. Computation is performed by the C++ backend.

Usage

```
fit_multistate(
  TTij,
  ssij,
  nobss,
  zzi,
  F_zidx,
  possible_transition,
  timepoints,
  niter = 1000,
  tol = 1e-04,
  factor = 0.75
)
```

Arguments

`TTij`	Numeric matrix of observation times. `TTij[i, j]` is the observation time for the *j*-th observation of the *i*-th individual. Only `TTij[i, 1:nobss[i]]` are used.

ssij	Logical 3-dimensional array indicating possible state occupation. ssij[i, j, s] = TRUE if the *i*-th individual may occupy state s at the *j*-th observation time.
nobss	Integer vector giving the number of observation times for each individual. nobss[i] is the number of observations for the *i*-th individual.
zzi	Numeric matrix of covariates. The *i*-th row corresponds to the *i*-th individual and each column to a predictor variable.
F_zidx	A matrix of lists with integer vectors specifying transition-specific covariate indices. The (s1, s2) element contains the indices of covariates used to model the transition from state s1 to state s2.
possible_transition	Logical matrix indicating allowable state transitions. possible_transition[s1, s2] = TRUE if and only if the transition from state s1 to state s2 is allowed.
timepoints	Integer scalar specifying the total number of discrete time points defining the observation window. It should be greater than or equal to the maximum effective entry of TTij.
niter	Maximum number of iterations for the optimization algorithm. Default is 1000.
tol	Convergence tolerance for the iterative estimation procedure. Default is 1e-4.
factor	Step-size adjustment factor used in the optimization routine. Must be between 0 and 1. Default is 0.75.

Details

We consider *N* individuals (N = nrow(TTij)) who are observed at possibly unevenly spaced time points. For the *i*-th individual, the observation times are:

TTij[i, 1], ..., TTij[i, nobss[i]].

There are *ns* possible states (ns = nrow(possible_transition)). The logical matrix possible_transition encodes allowable transitions:

possible_transition[s1, s2] = TRUE

if and only if a transition from state s1 to state s2 is allowed.

At each observation time TTij[i, j], the individual's state may be uniquely determined or partially observed. If the state is uniquely known to be s, then ssij[i, j, s] = TRUE and all other components of ssij[i, j,] are FALSE. If the individual may occupy multiple states (e.g., s1 or s2), then ssij[i, j, s1] and ssij[i, j, s2] are TRUE, representing the set of possible states at that time.

There are *nz* predictor variables (nz = ncol(zzi)). Transition-specific covariate effects are specified through F_zidx, where:

F_zidx[[s1]][[s2]]

(or equivalently the (s1, s2) entry) contains the indices of covariates used to model the transition from state s1 to state s2.

Value

A list containing:

Examples

```
# A simpler example
library(IntervalCensoredMultistateR2)
data("examplesimple_multistate")
model_fit_multistate<-
  IntervalCensoredMultistateR2::fit_multistate(
    examplesimple_multistate$TTij,
    examplesimple_multistate$ssij,
    examplesimple_multistate$nobss,
    examplesimple_multistate$zzi,
    examplesimple_multistate$F_zidx,
    examplesimple_multistate$possible_transition,
    timepoints=seq(0.05,1,0.05))

# A more complicated example
library(IntervalCensoredMultistateR2)
data("example_multistate")

model_fit_multistate<-
  IntervalCensoredMultistateR2::fit_multistate(
    example_multistate$TTij,
    example_multistate$ssij,
    example_multistate$nobss,
    example_multistate$zzi,
    example_multistate$F_zidx,
    example_multistate$possible_transition,
    timepoints=seq(0.01,1,0.01))
```

---

fit_single_event *Fit a proportional hazards model for interval-censored single-event data*

Description

Estimates regression coefficients in a proportional hazards model for interval-censored time-to-event data. Each individual is observed within an interval `(left, right]`, with an indicator specifying whether the event occurred. Computation is performed by the C++ backend.

Usage

```
fit_single_event(
  left,
  right,
  event,
  zzi,
  timepoints,
  niter = 1000,
  tol = 1e-04,
```

```
  factor = 0.75
)
```

Arguments

`left`	Numeric vector of left endpoints of observation intervals. For the *i*-th individual, `left[i]` is the lower bound of the interval in which the event time lies. In cases of right-censoring, right-censoring), `left[i]` is the last follow-up time.
`right`	Numeric vector of right endpoints of observation intervals. For the *i*-th individual, `right[i]` is the upper bound of the interval in which the event time lies. In cases of right-censoring, `right[i] = Inf`.
`event`	Integer or logical vector indicating event status. Typically, `event[i] = 1` (or `TRUE`) indicates the event occurred within `(left[i], right[i]]`, and `0` (or `FALSE`) indicates right-censoring.
`zzi`	Numeric matrix of covariates. The *i*-th row corresponds to the *i*-th individual and each column to a predictor variable.
`timepoints`	Timepoints for discrete approximation. `max(timepoints)` should be greater than or equal to `max(right)`.
`niter`	Maximum number of iterations for the optimization algorithm. Default is `1000`.
`tol`	Convergence tolerance for the iterative estimation procedure. Default is `1e-4`.
`factor`	Step-size adjustment factor used in the optimization routine. Must be between 0 and 1. Default is `0.75`.

Details

This function fits a proportional hazards model for interval-censored single-event data. The model assumes that the hazard function depends multiplicatively on covariates through a log-linear form.

The R interface calls an RcppArmadillo implementation of the core numerical optimization.

Value

A list of summary:

- `coef`: Estimated regression coefficients.
- `converged`: Logical indicator of convergence.
- `niter`: Number of iterations performed.

Examples

```
library(IntervalCensoredMultistateR2)
data("examplesimple_single_event")
model_fit_single_event<-
  IntervalCensoredMultistateR2::fit_single_event(
    examplesimple_single_event$left,
    examplesimple_single_event$right,
    examplesimple_single_event$event,
    cbind(
```

```
      examplesimple_single_event$X1,
      examplesimple_single_event$X2,
      examplesimple_single_event$X3),
    seq(0.05,1,0.05),
    niter=100)

# A more complicated example
library(IntervalCensoredMultistateR2)
data("example_single_event")

model_fit_single_event<-
  IntervalCensoredMultistateR2::fit_single_event(
    example_single_event$left,
    example_single_event$right,
    example_single_event$event,
    cbind(
      example_single_event$X1,
      example_single_event$X2,
      example_single_event$X3),
    seq(0.01,1,0.01),
    niter=100)
```

Index