# Package 'simevent'

April 24, 2026

**Title** Simulation and Analysis of Event History Data

**Version** 0.1.1

**Description** Simulate event history data from a framework where treatment decisions and disease progression are represented as counting process. The user can specify number of events and parameters of intensities thereby creating a flexible simulation framework.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**URL** https://github.com/miclukacova/simevent

**BugReports** https://github.com/miclukacova/simevent/issues

**Imports** stats, data.table, ggplot2, survival, Rcpp

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**LinkingTo** Rcpp

**VignetteBuilder** knitr

**NeedsCompilation** yes

**Author** Michaela Lukacova [aut, cre]

**Maintainer** Michaela Lukacova <michaela.lukacova@sund.ku.dk>

**Repository** CRAN

**Date/Publication** 2026-04-24 08:30:02 UTC

## Contents

---

`alphaSim` *Simulation and Estimation with Modified Shape Parameter*

---

## Description

This function simulates event history data from either the Disease, Treatment or Drop In setting. See simDisease, simTreatment and simDropIn. The shape parameter $\eta$ of respectively the disease process, the Drop In process and the Treatment process is multiplied by `alpha`. The function either

- returns the proportion of individuals who experience death and the proportion of individuals who experience disease/drop in/treatment by a specified time $\tau$ (in group `A0` = `a0` for drop in and disease).
- returns number of years lost before $\tau$ of death and disease/drop in/treatment
- returns simulated data. One can specify all the same parameters as in the functions `simDisease`, `simTreatment` and `simDropIn`.

## Usage

```
alphaSim(
  N = 10000,
  eta = rep(0.1, 4),
  nu = rep(1.1, 4),
  alpha = 0.5,
  tau = 5,
  a0 = 1,
  years_lost = FALSE,
  setting = "Disease",
  return_data = FALSE,
  cens = 0,
  ...
)
```

## Arguments

| | |
|---|---|
| N | Integer. Number of individuals to simulate. Default is 10,000. |
| eta | Numeric vector. Shape parameters for Weibull hazards. Length of the vector should match number of events. For the Disease and Drop In setting this is 4. For the Treatment setting, this is 3. (default `rep(0.1, 4)`). |
| nu | Numeric vector. Scale parameters for Weibull hazards. Length of the vector should match number of events. For the Disease and Drop In setting this is 4. For the Treatment setting, this is 3. (default `rep(1.1, 4)`). |
| alpha | Numeric scalar. Multiplicative factor applied to the disease process shape parameter $\eta$. |
| tau | Numeric scalar. Time horizon at which proportions are computed. |
| a0 | Binary (0/1). Specifies the group for comparison in setting Drop In and Disease. |
| years_lost | Logical. If `TRUE`, computes years lost instead of proportions. |
| setting | Character string. Must be either "Disease", "Drop In" or "Treatment". Depending on the simulation setting. |
| return_data | Logical. If `TRUE` the simulated data is returned. |
| cens | Binary scalar. Indicates whether individuals are at risk of censoring (default `0`). |
| ... | Additional arguments passed to respectively simDisease, simTreatment and simDropIn. |

## Value

A list with two components:

`effect_L` Proportion (or years lost) of individuals diagnosed with disease by time $\tau$, under intervention.

`effect_death` Proportion (or years lost) of individuals who died by time $\tau$ under intervention.

Or the simulated data.

## Examples

```
alphaSim(N = 100, eta = rep(0.1,3), nu = rep(1.1,3), alpha = 0.5, setting = "Disease")
alphaSim(N = 100, setting = "Drop In", beta_A0_Z = 1)
```

---

`intEffectAlpha` *Estimate Effect of Intervention: Modifying Eta Parameter of Process*

---

## Description

This function simulates data from the disease setting in two scenarios. Under intervention on the shape parameter $\eta$ of the disease process is multiplied by `alpha`, and a baseline (non-intervened) scenario. It computes the proportion of individuals who experience death or disease by a specified time $\tau$ in the group `A0 = a0`, optionally returning years_lost. The function can also plot a sample of the event data for each scenario for comparison.

## Usage

```
intEffectAlpha(
  N = 10000,
  setting = "Disease",
  eta = rep(0.1, 4),
  nu = rep(1.1, 4),
  alpha = 0.5,
  tau = 5,
  a0 = 1,
  years_lost = FALSE,
  plot = TRUE,
  lower = 10^(-30),
  upper = 200,
  cens = 0,
  ...
)
```

## Arguments

| | |
|---|---|
| N | Integer. Number of individuals to simulate. Default is 10,000. |
| setting | Character string. Must be either "Disease", "Drop In" or "Treatment". Depending on the simulation setting. |
| eta | Numeric vector of length 3. Shape parameters for Weibull hazards (default rep(0.1, 4)). |
| nu | Numeric vector of length 3. Scale parameters for Weibull hazards (default rep(1.1, 4)). |
| alpha | Numeric scalar. Multiplicative factor applied to the disease process shape parameter $\eta$. |
| tau | Numeric scalar. Time horizon at which proportions are computed. |
| a0 | Binary (0/1). Specifies the group for comparison.Only relevant in setting "Drop In" and "Disease". |
| years_lost | Logical. If TRUE, computes years lost instead of proportions. |
| plot | Logical. If TRUE, plots timelines for sample of intervention and non intervention data. |
| lower | Numeric scalar. Lower bound for root-finding in hazard inversion (default 1e-30). |
| upper | Numeric scalar. Upper bound for root-finding in hazard inversion (default 200). |
| cens | Binary scalar. Indicates whether individuals are at risk of censoring (default 0). |
| ... | Additional arguments passed to respectively simDisease, simTreatment and simDropIn. |

## Value

A list with two components:

`effect_L` Proportion (or years lost) of individuals diagnosed with disease by time $\tau$ in group `A0 = a0`, under intervention.

`effect_death` Proportion (or years lost) of individuals who died by time $\tau$ in group `A0 = a0`, under intervention.

## Examples

```
intEffectAlpha(N = 1000, alpha = 0.7, tau = 5, years_lost = FALSE, a0 = 1, setting = "Drop In")
```

---

`IntFormatData` *Transform Event Data into Interval Format for Classical Inference*

---

## Description

Converts simulated event history data with time-dependent covariates into an interval (start-stop) format, suitable for classical survival analysis functions like `coxph`. Adds interval start and stop times (`tstart`, `tstop`) and a counting variable `k` indexing events. Optionally, the function can split intervals at a specified time point to accomadate estimation of time-varying effects.

## Usage

```
IntFormatData(data, N_cols = 6:9, timeVar = FALSE, t_prime = NULL)
```

## Arguments

| | |
|---|---|
| `data` | A `data.frame` or `data.table` containing event data with columns `ID`, `Time`, `Delta`, and counting process columns specified by `N_cols`. |
| `N_cols` | Integer vector. Column indices of `data` that correspond to counting process variables. Defaults to `6:9`. |
| `timeVar` | Logical. If `TRUE`, the intervals are split at `t_prime` to allow time-varying covariate effects. Default is `FALSE`. |
| `t_prime` | Numeric scalar. Time point at which to split intervals if `timeVar = TRUE`. |

## Value

A `data.table` with columns `tstart`, `tstop`, `k`, and other original variables, formatted for survival analysis.

## Examples

```
data <- simEventData(10)
IntFormatData(data)
```

`inverseScHaz` *Wrapper for inverse cumulative hazard*

## Description

A wrapper around the Rcpp function `inverseScHaz`, used to find the inverse by numeric methods in case of no simple analytical solution.

## Usage

```
inverseScHaz(
  p,
  t,
  lower,
  upper,
  eta,
  nu,
  phi,
  at_risk,
  tol = 1e-09,
  max_iter = 100
)
```

## Arguments

| | |
|---|---|
| `p` | The random variable (typically `-log(U)`). |
| `t` | Current time. |
| `lower` | Lower bound for root finding. |
| `upper` | Upper bound for root finding. |
| `eta` | Numeric vector of shape parameters. |
| `nu` | Numeric vector of scale parameters. |
| `phi` | Numeric vector of multiplicative effects. |
| `at_risk` | Numeric vector indicating at-risk indicators for each event type. |
| `tol` | Numeric tolerance for root-finding. Default is 1e-9. |
| `max_iter` | Maximum iterations. Default is 100. |

## Value

A numeric scalar, the root `u`.

## Examples

```
eta <- c(0.1, 0.1)
nu <- c(1.1, 1.1)
phi <- c(1, 1)
at_risk <- c(1, 1)
inverseScHaz(0.5, t = 0, lower = 1e-15, upper = 200, eta, nu, phi, at_risk)
```

---

`inverseScHazTV` *Wrapper for inverse cumulative hazard*

---

## Description

A wrapper around the Rcpp function `inverseScHazCppTV`, used to find the inverse of the summed cumulative hazard.

## Usage

```
inverseScHazTV(
  p,
  t,
  lower = 1e-15,
  upper = 200,
  t_prime,
  eta,
  nu,
  phi,
  phi_prime,
  at_risk,
  tol = 1e-09,
  max_iter = 100
)
```

## Arguments

| | |
|---|---|
| `p` | The random variable (typically `-log(U)`). |
| `t` | The time of the previous event |
| `lower` | Lower bound for root finding. |
| `upper` | Upper bound for root finding. |
| `t_prime` | The time where the time varying effects change |
| `eta` | Numeric vector of shape parameters. |
| `nu` | Numeric vector of scale parameters. |
| `phi` | Numeric vector of multiplicative effect bedfore time t_prime |
| `phi_prime` | Numeric vector of multiplicative effects after time t_prime |
| `at_risk` | Numeric vector indicating at-risk indicators for each event type. |
| `tol` | Numeric tolerance for root-finding. Default is 1e-9. |
| `max_iter` | Maximum iterations. Default is 100. |

## Value

A numeric scalar, the root u.

## Examples

```
eta <- c(0.1, 0.1)
nu <- c(1.1, 1.1)
phi <- c(1, 1)
at_risk <- c(1, 1)
phi_prime <- c(2, 2)
inverseScHazTV(p = 0.5, t= 1, t_prime = 2, eta = eta, nu = nu,
                       phi = phi, phi_prime = phi_prime, at_risk = at_risk)
```

---

# plotEventData *Plot Simulated Event History Data*

---

## Description

Visualizes event history data by plotting individual event times colored and shaped by event type. Each individual's timeline is displayed horizontally with events marked along it.

## Usage

```
plotEventData(data, title = "Event Data")
```

## Arguments

| | |
|---|---|
| data | A data.frame or data.table containing at least the columns ID, Time, and Delta. |
| title | Character string specifying the plot title. Defaults to "Event Data". |

## Value

A ggplot object representing the event data visualization.

## Examples

```
data <- simEventData(10)
plotEventData(data)
```

## sampleEvents *Sample event types from matrix of probabilities*

### Description

Sample event types from matrix of probabilities

### Usage

```
sampleEvents(probs)
```

### Arguments

| | |
|---|---|
| probs | A matrix where each column is a probability vector |

### Value

A vector of sampled event types (0-indexed)

## simCRdata *Simulate Competing Risks Data*

### Description

Simulates competing risks data for $N$ individuals who are at risk of mutually exclusive event types. Three event types are simulated, where one can be interpreted as censoring.

### Usage

```
simCRdata(N, beta = NULL, eta = rep(0.1, 3), nu = rep(1.1, 3), cens = 1, ...)
```

### Arguments

| | |
|---|---|
| N | Integer. Number of individuals to simulate. |
| beta | Numeric matrix of dimension 2x3. Covariate effects of L0 and A0 on the three competing processes (columns correspond to processes). Defaults to zero matrix if NULL. |
| eta | Numeric vector of length 3. Shape parameters for Weibull hazards, parameterized as $\eta\nu t^{\nu-1}$. Defaults to rep(0.1, 3). |
| nu | Numeric vector of length 3. Scale parameters for Weibull hazards. Defaults to rep(1.1, 3). |
| cens | Binary (0 or 1). Indicates if a censoring process is included. Default is 1. |
| ... | Additional arguments passed to simEventData, including add_cov for extra covariates. |

## Details

The event intensities follow Weibull hazard models parameterized by shape and scale parameters $\eta$ and $\nu$. Covariate effects on the hazard are specified by the `beta` matrix, which models the effects of baseline covariates `L0` and `A0` on each event type.

## Value

A `data.frame` with simulated competing risk data including:

- `ID` - Individual identifier.
- `Time` - Event time.
- `Delta` - Event type (0, 1, or 2).
- `L0` - Baseline covariate.
- `A0` - Baseline treatment indicator.

## Examples

```
simCRdata(10)
```

---

`simDisease` *Simulate Data in a Disease Setting*

---

## Description

This function simulates event data representing three event types: Censoring (0), Death (1), and Change in Covariate Process (2). Death and Censoring are terminal events, while Change in Covariate Process can occur only once.

## Usage

```
simDisease(
  N,
  eta = rep(0.1, 3),
  nu = rep(1.1, 3),
  cens = 1,
  beta_L0_D = 1,
  beta_L0_L = 1,
  beta_L_D = 1,
  beta_A0_D = 0,
  beta_A0_L = 0,
  beta_L0_C = 0,
  beta_A0_C = 0,
  beta_L_C = 0,
  followup = Inf,
  lower = 10^(-15),
  upper = 200,
```

```
  beta_L_D_t_prime = NULL,
  t_prime = NULL,
  gen_A0 = NULL,
  at_risk_cov = NULL,
  ...
)
```

## Arguments

| | |
|---|---|
| N | Numeric scalar. Number of individuals to simulate. |
| eta | Numeric vector of length 3. Shape parameters for Weibull intensities with parameterization $\eta\nu t^{\nu-1}$. Defaults to `rep(0.1, 3)`. |
| nu | Numeric vector of length 3. Scale parameters for the Weibull hazards. Defaults to `rep(1.1, 3)`. |
| cens | Binary scalar. Indicates whether individuals are at risk of censoring (default 1). |
| beta_L0_D | Numeric scalar. Effect of baseline covariate L0 on death risk (default 1). |
| beta_L0_L | Numeric scalar. Effect of baseline covariate L0 on covariate change risk (default 1). |
| beta_L_D | Numeric scalar. Effect of covariate change (L = 1) on death risk (default 1). |
| beta_A0_D | Numeric scalar. Effect of baseline treatment (A0 = 1) on death risk (default 0). |
| beta_A0_L | Numeric scalar. Effect of baseline treatment (A0 = 1) on covariate change risk (default 0). |
| beta_L0_C | Numeric scalar. Effect of baseline covariate L0 on censoring probability (default 0). |
| beta_A0_C | Numeric scalar. Effect of baseline treatment A0 on censoring probability (default 0). |
| beta_L_C | Numeric scalar. Effect of covariate change (L = 1) on censoring probability (default 0). |
| followup | Numeric scalar. Maximum follow-up (censoring) time. Defaults to `Inf`. |
| lower | Numeric scalar. Lower bound for root-finding (inverse cumulative hazard) (default `1e-15`). |
| upper | Numeric scalar. Upper bound for root-finding (default 200). |
| beta_L_D_t_prime | Numeric scalar or NULL. Additional effect of covariate change on death risk after time `t_prime` (optional). |
| t_prime | Numeric scalar or NULL. Time point where effects change (optional). |
| gen_A0 | Function. Function to generate the baseline treatment covariate A0. Takes N and L0 as inputs. Default is a Bernoulli(0.5) random variable. |
| at_risk_cov | Function. Function determining if an individual is at risk for each event type, given their covariates. Takes a numeric vector covariates and returns a binary vector. Default returns 1 for all events. |
| ... | Additional arguments passed to `simEventData` or `simEventTV` |

## Details

Event intensities depend on previous events and predefined parameters $\nu$ and $\eta$.

The arguments `beta_X_Y` control how the X affects Y. A positive value means that a higher value of X increases the intensity of Y, while a negative value decreases the intensity.

The simulation uses an event history framework with terminal events (death, censoring) and a single recurrent covariate change. The event intensities depend on covariates and previous events according to user-specified parameters. Time-varying effects can be included via `beta_L_D_t_prime` and `t_prime`.

## Value

A data frame containing the simulated data with columns:

| | |
|---|---|
| `ID` | Individual identifier |
| `Time` | Time of the event |
| `Delta` | Event type (0 = censoring, 1 = death, 2 = covariate change) |
| `L0` | Baseline covariate |
| `L` | Covariate indicating change in covariate process |

## Examples

```
simDisease(10)
```

---

`simDropIn` `simDropIn`

---

## Description

Simulate Event Data from a "Drop In" Setting

## Usage

```
simDropIn(
  N,
  eta = c(0.5, 0.5, 0.1, 0.25),
  nu = c(1.1, 1.1, 1.1, 1.1),
  adherence = FALSE,
  followup = Inf,
  cens = 1,
  generate.A0 = function(N, L0) stats::rbinom(N, 1, 0.5),
  lower = 1e-200,
  upper = 1e+10,
  t_prime = NULL,
  at_risk_cov = NULL,
```

```
  beta_L_A = 1,
  beta_L_Z = 2,
  beta_L_D = 1.5,
  beta_L_C = 0,
  beta_A_L = -0.5,
  beta_A_Z = -0.5,
  beta_A_D = -1,
  beta_A_C = 0,
  beta_Z_L = -1,
  beta_Z_A = 0,
  beta_Z_D = -1,
  beta_Z_C = 0,
  beta_L0_L = 1,
  beta_L0_A = 1,
  beta_L0_Z = 1,
  beta_L0_D = 1,
  beta_L0_C = 0,
  beta_A0_L = -1.5,
  beta_A0_A = 0,
  beta_A0_Z = 0,
  beta_A0_D = -2,
  beta_A0_C = 0,
  beta_L_A_prime = 0,
  beta_L_Z_prime = 0,
  beta_L_D_prime = 0,
  beta_L_C_prime = 0,
  beta_A_L_prime = 0,
  beta_A_Z_prime = 0,
  beta_A_D_prime = 0,
  beta_A_C_prime = 0,
  beta_Z_L_prime = 0,
  beta_Z_A_prime = 0,
  beta_Z_D_prime = 0,
  beta_Z_C_prime = 0,
  beta_L0_L_prime = 0,
  beta_L0_A_prime = 0,
  beta_L0_Z_prime = 0,
  beta_L0_D_prime = 0,
  beta_L0_C_prime = 0,
  beta_A0_L_prime = 0,
  beta_A0_A_prime = 0,
  beta_A0_Z_prime = 0,
  beta_A0_D_prime = 0,
  beta_A0_C_prime = 0,
  ...
)
```

## Arguments

| | |
|---|---|
| N | Integer. Number of individuals to simulate. |
| eta | Numeric vector of length 4 (or 5). Shape parameters of the Weibull baseline intensity for each event type. $$\eta\nu t^{\nu-1}$$ . |
| nu | Numeric vector of length 4 (or 5). Scale parameters for the Weibull hazard. |
| adherence | Logical. Indicator of whether a Treatment process should be simulated. |
| followup | Numeric. Maximum censoring time. Events occurring after this time are censored. Default is Inf (no censoring). |
| cens | Logical. Indicator of whether there should be a censoring process. |
| generate.A0 | Function. Function to generate the baseline treatment covariate A0. Takes N and L0 as inputs. Default is a Bernoulli(0.5) random variable. |
| lower | Numeric. Lower bound for root-finding in inverse cumulative hazard calculations. Default is $10^{-15}$. |
| upper | Numeric. Upper bound for root-finding in inverse cumulative hazard calculations. Default is 200. |
| t_prime | Numeric scalar or NULL. Time point where effects change (optional). |
| at_risk_cov | Function. Function determining if an individual is at risk for each event type, given their covariates. Takes a numeric vector covariates and returns a binary vector. Default returns 1 for all events. |
| beta_L_A | Numeric. Specifies how L affects A. |
| beta_L_Z | Numeric. Specifies how L affects Z. |
| beta_L_D | Numeric. Specifies how L affects D. |
| beta_L_C | Numeric. Specifies how L affects C. |
| beta_A_L | Numeric. Specifies how L affects A. |
| beta_A_Z | Numeric. Specifies how L affects Z. |
| beta_A_D | Numeric. Specifies how L affects D. |
| beta_A_C | Numeric. Specifies how L affects C. |
| beta_Z_L | Numeric. Specifies how L affects A. |
| beta_Z_A | Numeric. Specifies how L affects Z. |
| beta_Z_D | Numeric. Specifies how L affects D. |
| beta_Z_C | Numeric. Specifies how L affects C. |
| beta_L0_L | Numeric. Specifies how L affects A. |
| beta_L0_A | Numeric. Specifies how L affects Z. |
| beta_L0_Z | Numeric. Specifies how L affects Z. |
| beta_L0_D | Numeric. Specifies how L affects D. |
| beta_L0_C | Numeric. Specifies how L affects C. |
| beta_A0_L | Numeric. Specifies how L affects A. |

beta_A0_A Numeric. Specifies how L affects Z.

beta_A0_Z Numeric. Specifies how L affects Z.

beta_A0_D Numeric. Specifies how L affects D.

beta_A0_C Numeric. Specifies how L affects C.

beta_L_A_prime Numeric. Specifies how L additionally affects A after time t_prime.

beta_L_Z_prime Numeric. Specifies how L additionally affects Z after time t_prime.

beta_L_D_prime Numeric. Specifies how L additionally affects D after time t_prime.

beta_L_C_prime Numeric. Specifies how L additionally affects C after time t_prime.

beta_A_L_prime Numeric. Specifies how L additionally affects A after time t_prime.

beta_A_Z_prime Numeric. Specifies how L additionally affects Z after time t_prime.

beta_A_D_prime Numeric. Specifies how L additionally affects D after time t_prime.

beta_A_C_prime Numeric. Specifies how L additionally affects C after time t_prime.

beta_Z_L_prime Numeric. Specifies how L additionally affects A after time t_prime.

beta_Z_A_prime Numeric. Specifies how L additionally affects Z after time t_prime.

beta_Z_D_prime Numeric. Specifies how L additionally affects D after time t_prime.

beta_Z_C_prime Numeric. Specifies how L additionally affects C after time t_prime.

beta_L0_L_prime
Numeric. Specifies how L additionally affects after time A.

beta_L0_A_prime
Numeric. Specifies how L additionally affects after time Z.

beta_L0_Z_prime
Numeric. Specifies how L additionally affects after time Z.

beta_L0_D_prime
Numeric. Specifies how L additionally affects after time D.

beta_L0_C_prime
Numeric. Specifies how L additionally affects after time C.

beta_A0_L_prime
Numeric. Specifies how L additionally affects after time A.

beta_A0_A_prime
Numeric. Specifies how L additionally affects after time Z.

beta_A0_Z_prime
Numeric. Specifies how L additionally affects after time Z.

beta_A0_D_prime
Numeric. Specifies how L additionally affects after time D.

beta_A0_C_prime
Numeric. Specifies how L additionally affects after time C.

... Additional arguments passed to simEventData or simEventTV

## Details

simDropIn is a function that simulates data corresponding to N individuals that are at risk for 4 or 5 events. Censoring (C), Death (D), Drop In Initiation (Z), Change in Covariate Process (L) and optionally Treatment (A).

## Value

Data frame containing the simulated event history data

## Examples

```
simDropIn(10)
```

---

`simEventCox` *Simulate Event History Data Based on Cox Models*

---

## Description

Simulates recurrent and terminal event data for a cohort of individuals based on a list of fitted Cox proportional hazards models. Each event type is governed by its own model, and simulation proceeds by iteratively sampling event times until a terminal event occurs.

## Usage

```
simEventCox(
  N,
  cox_fits,
  list_old_vars = NULL,
  n_event_max = c(1, 1),
  term_events = 1,
  intervention1 = NULL,
  intervention2 = NULL
)
```

## Arguments

| | |
|---|---|
| `N` | Integer. The number of individuals to simulate. |
| `cox_fits` | A named list of fitted Cox proportional hazards models (`coxph` objects), one for each event type. The names are used as event type labels. |
| `list_old_vars` | A named list containing the old covariates. New covariates will be simulated by drawing from the old covariates with replacement. |
| `n_event_max` | Integer vector. Maximum number of times each event type can occur per individual. |
| `term_events` | Integer or integer vector. Indices of event types that are terminal, i.e., events that stop further simulation for an individual. |
| `intervention1` | Optional function. Takes arguments `(j, sim_matrix)` and returns an updated simulation matrix. Used to modify covariates dynamically at each event iteration. |
| `intervention2` | Optional function. Takes arguments `(j, H_j)` and returns a modified baseline cumulative hazard vector for event type `j`. Allows dynamic hazard modification. The function `intervention2 <- function(j, basehaz) if(j ==2) 1.15 * basehaz else basehaz` performs an intervention where the baseline hazard of process 2 is multiplied by 1.15. |

### Details

The function simulates individual event histories by:

1. Sampling initial baseline covariates by resampling observed values.
2. Extracting baseline cumulative hazard functions from the Cox models.
3. Iteratively sampling event times.
4. Updating covariate histories and event counts.
5. Stopping simulation per individual after a terminal event or maximum events reached.

### Value

A `data.table` with one row per event per individual containing:

- `ID` — Individual identifier.
- `Time` — Event time.
- `Delta` — Event type indicator.
- Baseline covariates
- Columns for each event type indicating cumulative event counts.

### Examples

```
# The observed data
data_obs <- simDisease(N = 1000)
data_obs <- IntFormatData(data_obs, N_cols = 6)

# Fit Cox models
cox_death <- survival::coxph(survival::Surv(tstart, tstop, Delta == 1)
~ L0 + A0 + L, data = data_obs)
cox_Disease <- survival::coxph(survival::Surv(tstart, tstop, Delta == 2)
~ L0 + A0, data = data_obs[L == 0])

# Then simulate new data:
cox_fits <- list("D" = cox_death, "L" = cox_Disease)
list_old_vars <- list("L0" = data_obs$L0, "A0" = data_obs$A0)
new_data <- simEventCox(100, cox_fits = cox_fits, list_old_vars = list_old_vars)
```

---

## simEventData *Simulate Event Data with Multiple Event Types and Covariates*

---

### Description

Simulate Continuous Time-to-Event Data with Multiple Event Types

## Usage

```
simEventData(
  N,
  beta = NULL,
  eta = NULL,
  nu = NULL,
  at_risk = NULL,
  term_deltas = c(0, 1),
  max_cens = Inf,
  add_cov = NULL,
  override_beta = NULL,
  max_events = 10,
  lower = 10^(-15),
  upper = 200,
  gen_A0 = NULL,
  gen_L0 = NULL,
  at_risk_cov = NULL,
  ...
)
```

## Arguments

| | |
|---|---|
| N | Integer. Number of individuals to simulate. |
| beta | Numeric matrix. Regression coefficients matrix where columns correspond to event types (N0, N1, ...) and rows correspond to covariates (L0, A0, L1, L2, ...) and event counts (N0, N1, ...). Default is a zero matrix. |
| eta | Numeric vector. Shape parameters of the Weibull baseline intensity for each event type. Default is 0.1 for all events. |
| nu | Numeric vector. Scale parameters of the Weibull baseline intensity for each event type. Default is 1.1 for all events. |
| at_risk | Function. Function determining if an individual is at risk for each event type, given their current event counts. Takes a numeric vector events and returns a binary vector. Default returns 1 for all events. |
| term_deltas | Integer vector. Event types considered terminal (after which no further events occur). Default is c(0, 1). |
| max_cens | Numeric. Maximum censoring time. Events occurring after this time are censored. Default is Inf (no maximal censoring). |
| add_cov | Named list of functions. Functions generating additional baseline covariates. Each function takes integer N and returns a numeric vector of length N. Default is NULL. |
| override_beta | Named list. Used to specify entries of the beta matrix to override defaults. For example, `list("L0" = c("N1" = 2))` sets the effect of L0 on N1 to 2. |
| max_events | Integer. Maximum number of events to simulate per individual. Default is 10. |
| lower | Numeric. Lower bound for root-finding in inverse cumulative hazard calculations. Default is $10^{-15}$. |

| | |
|---|---|
| upper | Numeric. Upper bound for root-finding in inverse cumulative hazard calculations. Default is 200. |
| gen_A0 | Function. Function to generate the baseline treatment covariate A0. Takes N and L0 as inputs. Default is a Bernoulli(0.5) random variable. |
| gen_L0 | Function. Function to generate the baseline covariate L0. Takes N as inputs. Default is a N(0,1) random variable. |
| at_risk_cov | Function. Function determining if an individual is at risk for each event type, given their covariates. Takes a numeric vector covariates and returns a binary vector. Default returns 1 for all events. |
| ... | Additional technical arguments |

## Details

simEventData simulates event times and types for a cohort of individuals in a counting process framework. It supports multiple event types (by default 4), including terminal events, with intensities influenced by baseline covariates and previous event history.

The event intensities for event type $x$ at time $t$ are given by

$$\lambda^x(t) = \lambda_0^x(t) \exp(\beta_x^T L),$$

where the baseline intensity follows a Weibull hazard function:

$$\lambda_0^x(t) = \eta^x \nu^x t^{\nu^x - 1}.$$

Here, $L$ is the vector of covariates and event counts, and $\beta^x$ is the a vector of coefficients representing the effect of covariates and previous events on the intensity.

## Value

A data.table with columns:

| | |
|---|---|
| ID | Individual identifier |
| Time | Time of event |
| Delta | Event type at time |
| L0 | Baseline covariate |
| A0 | Baseline treatment |
| L1, L2, ... | Additional baseline covariates if specified |
| N0, N1, ... | Event counts up to the current event |

## Examples

```
# Simulate data for 10 individuals with default settings
sim_data <- simEventData(N = 10)
head(sim_data)
```

`simEventObj` *Simulate Survival and Competing Risk Data Based on a General Model*

## Description

The `simEventObj` function simulates survival or competing risk data for a cohort of individuals based on a general model with a `predict2` method. The function is useful for simulating additional data under the same distribution as an original data set. The procedure consists of fitting a model, such as a random forest or Cox Proportional Hazards model on an original data set. Next the model is equipped with a `predict2` method, and passed as an argument to the `simEventObj` function, which simulates new data using the `predict2` method. The method should output the cumulative hazard array and the jump times of the cumulative hazard. Simulation proceeds by sampling from the uniform distribution and obtaining event times using the inverse of the cumulative hazard function(s).

## Usage

```
simEventObj(N, obj, event_names = NULL, list_old_vars = NULL)
```

## Arguments

| | |
|---|---|
| `N` | Integer. The number of individuals to simulate. |
| `obj` | An object of class `simevent`. The object should have a predict2 method. The method should return a list containing `chf` and `time`. `chf` should be an array of dimension Individuals x Times x Events. The array should contain the cumulative hazard values for each individual, at each time for each event. `time` should be a vector of times where the cumulative hazard function jumps. |
| `event_names` | A character vector. Containing the names of the various processes. The argument is optional. By default events will be named `N1`, `N2`, .... |
| `list_old_vars` | A named list containing the old covariates. New covariates will be simulated by drawing from the old covariates with replacement. |

## Details

The function simulates individual event histories by:

1. Sampling initial baseline covariates by resampling observed values.
2. Extracting cumulative hazard functions from the object.
3. Iteratively sampling event times.
4. Updating covariate histories and event counts.
5. Stopping simulation per individual after a terminal event or maximum events reached.

## Value

A `data.table` with one row per event per individual containing:

- `ID` — Individual identifier.
- `Time` — Event time.
- `Delta` — Event type indicator.
- Baseline covariates.
- Columns for each event type indicating cumulative event counts.

---

`simEventTV` *simEventTV*

---

## Description

Simulate Event Data with Time-Varying Effects

## Usage

```
simEventTV(
  N,
  beta = NULL,
  tv_eff = NULL,
  t_prime = Inf,
  eta = NULL,
  nu = NULL,
  at_risk = NULL,
  term_deltas = c(0, 1),
  max_cens = Inf,
  add_cov = NULL,
  override_beta = NULL,
  max_events = 10,
  lower = 10^(-15),
  upper = 200,
  gen_A0 = NULL,
  at_risk_cov = NULL
)
```

## Arguments

| | |
|---|---|
| `N` | Integer. Number of individuals to simulate. |
| `beta` | Matrix. Coefficients for covariates and processes. Columns correspond to events (`N0`, `N1`, ...), rows correspond to covariates (`L0`, `A0`, ..., and past event counts). |
| `tv_eff` | Matrix. Time-varying changes to `beta`, applied at time `t_prime`. Must have same dimensions as `beta`. |
| `t_prime` | Numeric. Time at which `tv_eff` is added to `beta`. |

| | |
|---|---|
| eta | Numeric vector. Shape parameters of Weibull intensities for each event. |
| nu | Numeric vector. Scale parameters of Weibull intensities for each event. |
| at_risk | Function. Determines which events an individual is at risk for, based on event history. |
| term_deltas | Integer vector. Event types considered terminal (e.g., death). |
| max_cens | Numeric. Maximum censoring time. Defaults to Inf. |
| add_cov | Named list of functions for generating additional baseline covariates. Each function takes one argument N and returns a vector of length N. |
| override_beta | Named list to override elements of beta. Format: list("covariate" = c("event" = value)). |
| max_events | Integer. Maximum number of events allowed per individual. |
| lower | Numeric. Lower bound for the root-finding algorithm used in inverse cumulative hazard computation. |
| upper | Numeric. Upper bound for the root-finding algorithm used in inverse cumulative hazard computation. |
| gen_A0 | Function. Generates baseline treatment assignment. Takes arguments N and L0. |
| at_risk_cov | Function. Function determining if an individual is at risk for each event type, given their covariates. Takes a numeric vector covariates and returns a binary vector. Default returns 1 for all events. |

## Details

simEventTV is a function that simulates event data, with the option of adding time varying effects. The function is build up in the same way as simEventData, with the additional arguments tv_eff and t_prime, which specify the change of the beta matrix at time t_prime.

## Value

A data.table with columns:

| | |
|---|---|
| ID: | Individual identifier |
| Time: | Time of event |
| Delta: | Type of event |
| L0: | Baseline covariate |
| A0: | Baseline treatment |
| N0, N1, ...: | Cumulative event counts |
| L1, L2, ...: | Additional covariates (if specified) |

## Examples

```
eta <- rep(0.1, 2)
simEventTV(N = 100, t_prime = 1, eta = eta, term_deltas = c(0, 1))
```

simStatinData *Simulate Data in a Statin Setting*

## Description

Simulate Data in a Statin Setting

## Usage

```
simStatinData(
  N,
  eta = rep(0.1, 6),
  nu = rep(1.1, 6),
  beta = NULL,
  cens = 1,
  followup = 5,
  lower = 10^(-15),
  upper = 200,
  gen_A0 = NULL,
  gen_L0 = NULL,
  ...
)
```

## Arguments

| | |
|---|---|
| N | Numeric scalar. Number of individuals to simulate. |
| eta | Numeric vector of length 6. Shape parameters for Weibull intensities with parameterization $\eta\nu t^{\nu-1}$. Defaults to `rep(0.1, 6)`. |
| nu | Numeric vector of length 6. Scale parameters for the Weibull hazards. Defaults to `rep(1.1, 6)`. |
| beta | Numeric matrix. Of dimension p times 6. Regression coefficients matrix where columns correspond to event types (N0, ..., N5) and rows correspond to covariates (L0, A0, L1, L2, ...) followed by event counts (N0, ..., N5). Default is a zero matrix. |
| cens | Binary scalar. Indicates whether individuals are at risk of censoring (default 1). |
| followup | Numeric scalar. Maximum follow-up (censoring) time. Defaults to `Inf`. |
| lower | Numeric scalar. Lower bound for root-finding (inverse cumulative hazard) (default `1e-15`). |
| upper | Numeric scalar. Upper bound for root-finding (default 200). |
| gen_A0 | Function. Function to generate the baseline treatment covariate A0.Takes N and L0 as inputs. Default is a Bernoulli(0.5) random variable. |
| gen_L0 | Function. Function to generate the baseline covariate L0. Takes N as input. Default is a N(0,1) random variable. |
| ... | Additional arguments passed to `simEventData` |

## Value

A data frame containing the simulated data with columns:

| | |
|---|---|
| `ID` | Individual identifier |
| `Time` | Time of the event |
| `Delta` | Event type (0,...,5) |
| `L0` | Baseline covariate |
| `A0` | Baseline covariate |
| `L1, ...Lp` | Additional baseline covariates |

## Examples

```
simDisease(10)
```

---

`simSurvData` *Simulate Survival Data*

---

## Description

Simulate Survival Data with Censoring and Event Times

## Usage

```
simSurvData(N, beta = NULL, eta = rep(0.1, 2), nu = rep(1.1, 2), cens = 1, ...)
```

## Arguments

| | |
|---|---|
| `N` | Numeric scalar. Number of individuals to simulate. |
| `beta` | Numeric 2x2 matrix specifying effects of baseline covariates `L0` and `A0` on censoring and event hazards.<br>• Rows correspond to covariates `L0` and `A0`.<br>• Columns correspond to censoring (1st column) and event (2nd column). Defaults to zero matrix if `NULL`. |
| `eta` | Numeric vector of length 2. Shape parameters for Weibull hazard with parameterization $\eta\nu t^{\nu-1}$. Defaults to `rep(0.1, 2)`. |
| `nu` | Numeric vector of length 2. Scale parameters for the Weibull hazard. Defaults to `rep(1.1, 2)`. |
| `cens` | Numeric binary indicator (0 or 1) specifying if censoring is included (default 1). |
| `...` | Additional arguments passed to `simEventData`, including the argument `add_cov` to specify extra covariates. |

## Details

Simulates survival data for $N$ individuals who are at risk for censoring (0) and an event (1). The hazard functions for censoring and event times follow Weibull distributions parameterized by shape parameters $\eta$ and scale parameters $\nu$. Covariate effects on censoring and event hazards are specified via a matrix `beta`.

## Value

Data frame containing the simulated survival data

## Examples

```
simSurvData(10)
```

---

`simTreatment` *Simulate Event History Data with Treatment and Time-Dependent Covariate*

---

## Description

Simulates event history data with four types of events representing censoring (0), death (1), treatment (2), and covariate change (3). Death and censoring are terminal events; treatment and covariate events can occur only once.

## Usage

```
simTreatment(
  N,
  eta = rep(0.1, 4),
  nu = rep(1.1, 4),
  beta_L_A = 1,
  beta_L_D = 1,
  beta_A_D = -1,
  beta_A_L = -0.5,
  beta_L0_A = 1,
  beta_L0_L = 1,
  beta_L0_D = 1,
  beta_L0_C = 0,
  beta_L_C = 0,
  beta_A_C = 0,
  beta_L_A_prime = 0,
  beta_L_D_prime = 0,
  beta_A_D_prime = 0,
  beta_A_L_prime = 0,
  beta_L0_A_prime = 0,
  beta_L0_L_prime = 0,
```

```
  beta_L0_D_prime = 0,
  beta_L0_C_prime = 0,
  beta_L_C_prime = 0,
  beta_A_C_prime = 0,
  t_prime = NULL,
  at_risk_cov = NULL,
  cens = 1,
  op = 1,
  lower = 10^(-15),
  upper = 200,
  followup = Inf,
  ...
)
```

## Arguments

| | |
|---|---|
| N | Integer. Number of individuals to simulate. |
| eta | Numeric vector of length 4. Shape parameters for Weibull intensities, parameterized as $\eta\nu t^{\nu-1}$. Default is `rep(0.1, 4)`. |
| nu | Numeric vector of length 4. Scale parameters for Weibull hazards. Default is `rep(1.1, 4)`. |
| beta_L_A | Numeric. Effect of covariate `L = 1` on treatment hazard. Default 1. |
| beta_L_D | Numeric. Effect of covariate `L = 1` on death hazard. Default 1. |
| beta_A_D | Numeric. Effect of treatment `A = 1` on death hazard. Default -1. |
| beta_A_L | Numeric. Effect of treatment `A = 1` on covariate hazard. Default -0.5. |
| beta_L0_A | Numeric. Effect of baseline covariate `L0` on treatment hazard. Default 1. |
| beta_L0_L | Numeric. Effect of baseline covariate `L0` on covariate hazard. Default 1. |
| beta_L0_D | Numeric. Effect of baseline covariate `L0` on death hazard. Default 1. |
| beta_L0_C | Numeric. Effect of baseline covariate `L0` on censoring hazard. Default 0. |
| beta_L_C | Numeric. Effect of covariate `L = 1` on censoring hazard. Default 0. |
| beta_A_C | Numeric. Effect of treatment `A = 1` on censoring hazard. Default 0. |
| beta_L_A_prime | Numeric. Additional effect of covariate `L = 1` on treatment hazard. Default 0. |
| beta_L_D_prime | Numeric. Additionalffect of covariate `L = 1` on death hazard. Default 0. |
| beta_A_D_prime | Numeric. Effect of treatment `A = 1` on death hazard. Default 0. |
| beta_A_L_prime | Numeric. Effect of treatment `A = 1` on covariate hazard. Default 0. |
| beta_L0_A_prime | Numeric. Effect of baseline covariate `L0` on treatment hazard. Default 0. |
| beta_L0_L_prime | Numeric. Effect of baseline covariate `L0` on covariate hazard. Default 0. |
| beta_L0_D_prime | Numeric. Effect of baseline covariate `L0` on death hazard. Default 0. |
| beta_L0_C_prime | Numeric. Effect of baseline covariate `L0` on censoring hazard. Default 0. |

| | |
|---|---|
| beta_L_C_prime | Numeric. Effect of covariate L = 1 on censoring hazard. Default 0. |
| beta_A_C_prime | Numeric. Effect of treatment A = 1 on censoring hazard. Default 0. |
| t_prime | Numeric scalar or NULL. Time point where effects change (optional). |
| at_risk_cov | Function. Function determining if an individual is at risk for each event type, given their covariates. Takes a numeric vector covariates and returns a binary vector. Default returns 1 for all events. |
| cens | Integer (0 or 1). Indicates if censoring is possible. Default 1. |
| op | Integer (0 or 1). Indicates if treatment (operation) is possible. Default 1. |
| lower | Numeric. Lower bound for root finding (inverse cumulative hazard). Default 1e-15. |
| upper | Numeric. Upper bound for root finding (inverse cumulative hazard). Default 200. |
| followup | Numeric. Maximum censoring time. Defaults to Inf (no censoring). |
| ... | Additional arguments passed to simEventData or simEventTV |

## Details

Event intensities are modeled using Weibull hazards with parameters $\nu$ (scale) and $\eta$ (shape), and covariate effects controlled by specified beta parameters. For example, beta_L_A quantifies the effect of covariate L = 1 on the hazard of treatment.

## Value

A data.frame with columns:

- ID - Individual identifier.
- Time - Event time.
- Delta - Event type (0=censoring, 1=death, 2=treatment, 3=covariate change).
- L0 - Baseline covariate.
- L - Time-dependent covariate.
- A - Treatment status.

## Examples

```
simTreatment(10)
```

# Index