Package 'RealSurvSim'

December 9, 2025

Title Simulate Survival Data

Version 1.0.0

Date 2025-11-05

Description Provides tools for simulating synthetic survival data using a variety of methods, including kernel density estimation, parametric distribution fitting, and bootstrap resampling techniques for a desired sample size.

Encoding UTF-8

RoxygenNote 7.3.2

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Imports fitdistrplus, kdensity, survival, univariateML, actuar, flexsurv, stats, FAdist

Depends R (>= 3.5)

LazyData true

License MIT + file LICENSE

NeedsCompilation no

Author Maria Thurow [aut, cre] (ORCID: <https://orcid.org/0000-0002-8710-6857>), Manasi Butee [aut], Ina Dormuth [ctb], Christina Sauer [ctb], Marc Ditzhaus [ctb], Markus Pauly [ctb]

Maintainer Maria Thurow <maria.thurow@tu-dortmund.de>

Repository CRAN

Date/Publication 2025-12-09 07:50:18 UTC

Contents

`data_simul_Bootstr` *Simulate Data Using Bootstrap Methods*

Description

Simulates event and censoring times from an original dataset using specified bootstrap methodologies. This function supports conditional and case resampling bootstrap methods, allowing for flexible data simulation scenarios tailored to survival analysis.

Usage

```
data_simul_Bootstr(dat, n = NULL, type = "cond")
```

Arguments

`dat`	A dataframe containing the original dataset, expected to include columns for event times (V1), censoring indicators (V2), and group indicators (optional).
`n`	Integer specifying the number of observations to simulate. If `NULL`, the function simulates the same number of observations as in the original dataset. Defaults to `NULL`.
`type`	Character string specifying the type of bootstrap method to be used. Supported types include "cond" for conditional and "case" for case resampling. Defaults to "cond".

Value

A dataframe or a numeric vector of simulated values depending on the chosen bootstrap method. - For "case" bootstrap - For "cond" bootstrap, the arbitary n function does not work

Examples

```
dat <- data.frame(
  V1 = rexp(100, rate = 0.1), # Time-to-event data
  V2 = sample(0:1, 100, replace = TRUE),
  V3 = sample(0:1, 100, replace = TRUE)# Event indicator (0 = censored, 1 = event)
)
simulated_case <- data_simul_Bootstr(dat = dat, n = 100, type = "case")
simulated_cond <- data_simul_Bootstr(dat = dat, type = "cond")
```

data_simul_Estim *Simulate Data Based on Parametric Distribution Estimates*

Description

This function simulates data based on parameter estimates from a specified parametric distribution. It fits a chosen distribution to the original dataset and samples new values from this fitted distribution. Supported distributions include "inverse_gamma", "llogis" (log-logistic), "gumbel". "log-normal", "gamma", "exp", "cauchy".

Usage

```
data_simul_Estim(orig_vals, n = NULL, distrib = "exp")
```

Arguments

orig_vals	Numeric vector of values from the original dataset.
n	Integer specifying the number of observations to simulate. If NULL, the function simulates the same number of observations as in the original dataset. Defaults to NULL.
distrib	Character; one of "inverse_gamma", "llogis", "gumbel", "exp", "gamma", "normal", or "cauchy".

Value

Numeric vector of n simulated values based on the fitted parametric distribution.

Examples

```
original_data <- rnorm(100, mean = 50, sd = 10)
simulated_data <- data_simul_Estim(orig_vals = original_data, n = 100, distrib = "inverse_gamma")
```

data_simul_KDE *Kernel Density Estimation-based Data Simulation*

Description

Simulates data based on the kernel density estimation (KDE) of given data. KDE is a non-parametric way to estimate the probability density function of a random variable. This function applies the accept-reject method to generate values that follow the estimated density of the original dataset.

Usage

```
data_simul_KDE(orig_vals, n = NULL, kernel = "gaussian")
```

Arguments

orig_vals	Numeric vector of values from the original dataset.
n	Integer, number of observations to simulate. If NULL, the function simulates the same number of observations as in the original dataset. Defaults to NULL.
kernel	Character, specifying the kernel to be used for KDE. Defaults to "gaussian".

Value

Numeric vector of n simulated values.

Examples

```
original_data <- c(rnorm(100, mean = 50, sd = 10))
simulated_data <- data_simul_KDE(original_data, n = 100)
```

---

dats *Collection of Survival Datasets (dats)*

Description

dats is a collection of seven survival datasets used for testing and simulation of survival data. These datasets were reconstructed from published Kaplan-Meier survival curves using the widely applied algorithm by Guyot et al. (2012). The datasets were originally sourced from various clinical studies and digitized using WebPlotDigitizer. They are used as benchmarks for synthetic survival data methods, including kernel density estimation, parametric distribution fitting, and bootstrap resampling.

Usage

```
data(dats)
```

Format

A list containing 7 data frames. Each data frame includes:

V1 Time to event (numeric).

V2 Event indicator (0 = censored, 1 = event; numeric).

V3 Group identifier (numeric or categorical).

The datasets in dats are:

- **Liang**: Derived from Liang et al. (2019).
- **Spigel**: Derived from Spigel et al. (2022).
- **Wu**: Derived from Wu et al. (2015).
- **Wei**: Derived from Wei et al. (2020).
- **Lima**: Derived from Lima et al. (2018).
- **Yoshioka**: Derived from Yoshioka et al. (2019).
- **Seto**: Derived from Seto et al. (2020).

Source

- Maria Thurow et al. (2024). "How to Simulate Realistic Survival Data? A Simulation Study to Compare Realistic Simulation Models" *arXiv preprint*, https://arxiv.org/abs/2308.07842.
- Original datasets from respective publications (see dataset documentation for details).
- Data reconstructed using the algorithm by Guyot et al. (2012), *BMC Medical Research Methodology*, doi:10.1186/14712288129.
- Data digitized using WebPlotDigitizer (Rohatgi, A.), https://automeris.io/WebPlotDigitizer.

Examples

```
data(dats)
names(dats)
head(dats$Liang)
```

`RealSurvSim` *Simulate Datasets Using Various Simulation Models*

Description

Simulates survival datasets(Time-to-event data) based on original or reconstructed data using four different simulation models: Kernel Density Estimation (KDE), parametric distributions, conditional bootstrap, and Case Resampling. This function is designed to support comprehensive survival analysis simulations.

Usage

```
RealSurvSim(
  dat,
  col_time,
  col_status,
  col_group,
  reps = 10000,
  random_seed = 123,
  n = NULL,
  simul_type = c("cond", "case", "distr", "KDE"),
  distribs = c("exp", "exp", "exp", "exp")
)
```

Arguments

`dat`	A data.frame representing the original or reconstructed dataset for simulation. The dataset must include three columns: for event times, for censoring status, and for group identifiers.
`col_time`	The name or index of the column in `dat` representing time to event.

`col_status` The name or index of the column in `dat` representing the event status (1 for event occurred, 0 for censored).

`col_group` The name or index of the column in `dat` representing group assignments.

`reps` The number of iterations, equivalent to the number of datasets simulated for each simulation model. Defaults to 10000.

`random_seed` Seed for random number generation to ensure reproducibility. Defaults to 123.

`n` An optional numeric vector specifying the number of observations to simulate for each group. If `NULL`, the function uses the original dataset's group sizes for simulation. For all simulation types except "conditional bootstrap," `n` can be set to arbitrary values, such as `c(50, 60)`, where each element specifies the number of observations for a group. Defaults to `NULL`.

`simul_type` A vector of characters specifying the types of simulation to perform. It includes "cond" (conditional bootstrap), "case" (case resampling), "distr" (parametric distributions), and "KDE" (kernel density estimation, supports all kernels available in the `kdensity` function. Refer to 'kdensity'). Note: Only one simulation type can be used at a time.

`distribs` Character vector of length 4, one distribution per stratum. Must be one of:

- `"inverse_gamma"`
- `"llogis"`
- `"gumbel"`
- `"exp"`
- `"gamma"`
- `"normal"`
- `"cauchy"`

Defaults to `c("exp", "exp", "exp", "exp")`.

Value

A list containing the simulated datasets for each specified simulation model. The structure of the output list is as follows:

```
- {datasets}: A list of data frames, where each data frame represents a simulated dataset.
  - Each data frame contains:
      - {V1}: A numeric vector representing the simulated time-to-event data.
      - {V2}: A numeric or integer vector indicating the status, representing
        whether the event of interest has occurred (1) or is censored (0).
      - {V3}: An integer vector representing group.

- The number of data frames within {datasets} corresponds to the number of repetitions specified
  by the {reps} parameter.
```

Examples

```
# liang should have columns: V1 (time), V2 (status), V3 (group)

        # Simulate data using parametric distribution fitting
```

```
liang<- dats$Liang
liang_distr <- RealSurvSim(
 dat = liang,
 col_time = "V1",
 col_status = "V2",
 col_group = "V3",
 reps = 10,
 simul_type = "distr",
 distribs = c("exp", "exp", "exp", "exp")
)
```

Index