

Package: rsurv (via r-universe)

October 11, 2026

Title Random Generation of Survival Data

Version 0.0.3

Date 2026-01-23

Author Fabio Demarqui [aut, cre, cph]
(<https://orcid.org/0000-0001-9236-1986>)

Maintainer Fabio Demarqui <fndemarqui@est.ufmg.br>

Description Random generation of survival data from a wide range of regression models, including accelerated failure time (AFT), proportional hazards (PH), proportional odds (PO), accelerated hazard (AH), Yang and Prentice (YP), and extended hazard (EH) models. The package 'rsurv' also stands out by its ability to generate survival data from an unlimited number of baseline distributions provided that an implementation of the quantile function of the chosen baseline distribution is available in R. Another nice feature of the package 'rsurv' lies in the fact that linear predictors are specified via a formula-based approach, facilitating the inclusion of categorical variables and interaction terms. The functions implemented in the package 'rsurv' can also be employed to simulate survival data with more complex structures, such as survival data with different types of censoring mechanisms, left-, right-, and double-truncated survival data, survival data with cure fraction, survival data with random effects (frailties), multivariate survival data, and competing risks survival data. Details about the R package 'rsurv' can be found in Demarqui (2024) <[doi:10.48550/arXiv.2406.01750](https://doi.org/10.48550/arXiv.2406.01750)>.

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 3.4.0)

Imports bellreg (>= 0.0.2.2), dplyr, MASS, Rdpack, stabledist

RdMacros Rdpack

Suggests copula, flexsurv, knitr, LambertW, rmarkdown, survstan,
testthat (>= 3.0.0)

VignetteBuilder knitr

URL <https://github.com/fndemarqui/rsurv>,
<https://fndemarqui.github.io/rsurv/>

BugReports <https://github.com/fndemarqui/rsurv/issues>

Config/testthat/edition 3

Config/pak/sysreqs cmake make libicu-dev

Repository <https://fndemarqui.r-universe.dev>

Date/Publication 2026-01-23 12:02:01 UTC

RemoteUrl <https://github.com/fndemarqui/rsurv>

RemoteRef HEAD

RemoteSha 78e54546de01b8c0f95bc0d393ad49e1dc8b26c8

Contents

rsurv-package	3
bernoulli	3
fatigue	4
inv_pgf	5
loglogistic	6
lp	7
negbin	8
raftreg	9
rahreg	10
rayleigh	11
rehreg	13
rfrailty	14
rinterval	15
rphreg	16
rporeg	17
rypreg	18
Index	21

rsurv-package

The 'rsurv' package

Description

Random generation of survival data based on different survival regression models available in the literature, including Accelerated Failure Time (AFT) model, Proportional Hazard (PH) model, Proportional Odds (PO) model and the Yang & Prentice (YP) model.

`_PACKAGE`

References

Demarqui FN, Mayrink VD (2021). “Yang and Prentice model with piecewise exponential baseline distribution for modeling lifetime data with crossing survival curves.” *Brazilian Journal of Probability and Statistics*, **35**(1), 172 – 186. doi:[10.1214/20BJPS471](https://doi.org/10.1214/20BJPS471).

Yang S, Prentice RL (2005). “Semiparametric analysis of short-term and long-term hazard ratios with two-sample survival data.” *Biometrika*, **92**(1), 1-17.

bernoulli

Implemented link functions for the mixture cure rate model

Description

This function is used to specify different link functions for the count component of the mixture cure rate model.

Usage

```
bernoulli(link = "logit")
```

Arguments

link	desired link function; currently implemented links are: logit, probit, cloglog and cauchy.
------	--

Value

A list containing the codes associated with the count distribution assumed for the latent variable N and the chosen link.

fatigue

The Birnbaum-Saunders (Fatigue Life) Distribution

Description

Density, distribution function, quantile function and random generation for the Birnbaum-Saunders distribution with shape parameter alpha, scale parameter gamma, and location parameter mu.

Usage

```
dfatigue(x, alpha, gamma, mu = 0, log = FALSE)

pfatigue(q, alpha, gamma, mu = 0, lower.tail = TRUE, log.p = FALSE)

qfatigue(p, alpha, gamma, mu = 0, lower.tail = TRUE, log.p = FALSE)

rfatigue(n, alpha, gamma, mu = 0)
```

Arguments

x, q	vector of quantiles.
alpha	shape parameter. Must be positive.
gamma	scale parameter. Must be positive.
mu	location parameter (default is 0).
log, log.p	logical; if TRUE, probabilities/densities p are returned as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.
p	vector of probabilities.
n	number of observations. If length(n) > 1, the length is taken to be the number required.

Details

The Birnbaum-Saunders distribution, also known as the fatigue life distribution, is commonly used in reliability and survival analysis. It was originally proposed to model fatigue failure times of materials subjected to cyclic stress.

The probability density function is given by:

$$f(x) = \frac{\sqrt{(x-\mu)/\gamma} + \sqrt{\gamma/(x-\mu)}}{2\alpha(x-\mu)} \phi\left(\frac{\sqrt{(x-\mu)/\gamma} - \sqrt{\gamma/(x-\mu)}}{\alpha}\right)$$

for $x > \mu$, where ϕ is the standard normal density function.

Value

dfatigue gives the density, pfatigue gives the distribution function, qfatigue gives the quantile function, and rfatigue generates random deviates.

The length of the result is determined by n for rfatigue, and is the maximum of the lengths of the numerical arguments for the other functions.

References

Birnbaum, Z. W., & Saunders, S. C. (1969). A new family of life distributions. *Journal of Applied Probability*, 6(2), 319-327.

Examples

```
# Density at x = 2 with alpha = 0.5 and gamma = 1
dfatigue(2, alpha = 0.5, gamma = 1)

# CDF at x = 2
pfatigue(2, alpha = 0.5, gamma = 1)

# Quantile for p = 0.5 (median)
qfatigue(0.5, alpha = 0.5, gamma = 1)

# Generate 10 random values
rfatigue(10, alpha = 0.5, gamma = 1)
```

inv_pgf

Inverse of the probability generating function

Description

This function is used to specify different link functions for the count component of the promotion time cure rate model

Usage

```
inv_pgf(formula, incidence = "bernoulli", kappa = NULL, zeta = NULL, data, ...)
```

Arguments

formula	formula specifying the linear predictor for the incidence sub-model.
incidence	the desired incidence model.
kappa	vector of regression coefficients associated with the incidence sub-model.
zeta	extra negative-binomial parameter.
data	a data.frame containing the explanatory covariates passed to the formula.
...	further arguments passed to other methods.

Value

A vector with the values of the inverse of the desired probability generating function.

loglogistic	<i>The Log-Logistic Distribution</i>
-------------	--------------------------------------

Description

Density, distribution function, quantile function and random generation for the log-logistic distribution with shape and scale parameters.

Usage

```
dloglogistic(x, shape, scale, log = FALSE)

ploglogistic(q, shape, scale, lower.tail = TRUE, log.p = FALSE)

qloglogistic(p, shape, scale, lower.tail = TRUE, log.p = FALSE)

rloglogistic(n, shape, scale)
```

Arguments

x, q	vector of quantiles.
shape	shape parameter. Must be positive.
scale	scale parameter. Must be positive.
log, log.p	logical; if TRUE, probabilities/densities p are returned as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.
p	vector of probabilities.
n	number of observations. If length(n) > 1, the length is taken to be the number required.

Details

The log-logistic distribution is a continuous probability distribution for a non-negative random variable. It is the probability distribution of a random variable whose logarithm has a logistic distribution.

The probability density function is given by:

$$f(x) = \frac{(\alpha/\gamma)(x/\gamma)^{\alpha-1}}{(1 + (x/\gamma)^\alpha)^2}$$

for $x \geq 0$, where α is the shape parameter and γ is the scale parameter.

The cumulative distribution function is:

$$F(x) = \frac{1}{1 + (x/\gamma)^{-\alpha}}$$

Value

dloglogistic gives the density, ploglogistic gives the distribution function, qloglogistic gives the quantile function, and rloglogistic generates random deviates.

The length of the result is determined by n for rloglogistic, and is the maximum of the lengths of the numerical arguments for the other functions.

Examples

```
# Density at x = 1 with shape = 2 and scale = 1
dloglogistic(1, shape = 2, scale = 1)

# CDF at x = 1
ploglogistic(1, shape = 2, scale = 1)

# Quantile for p = 0.5 (median)
qloglogistic(0.5, shape = 2, scale = 1)

# Generate 10 random values
rloglogistic(10, shape = 2, scale = 1)
```

lp	<i>Linear predictors</i>
----	--------------------------

Description

Function to construct linear predictors.

Usage

```
lp(formula, coefs, data, ...)
```

Arguments

formula	formula specifying the linear predictors.
coefs	vector of regression coefficients.
data	data frame containing the covariates used to construct the linear predictors.
...	further arguments passed to other methods.

Value

a vector containing the linear predictors.

Examples

```
library(rsurv)
library(dplyr)

n <- 100
coefs <- c(1, 0.7, 2.3)

simdata <- data.frame(
  age = rnorm(n),
  sex = sample(c("male", "female"), size = n, replace = TRUE)
) %>%
  mutate(
    lp = lp(~age+sex, coefs)
  )
glimpse(simdata)
```

negbin

Implemented link functions for the promotion time cure rate model with negative binomial distribution

Description

This function is used to specify different link functions for the count component of the promotion time cure rate model.

Usage

```
negbin(zeta = stop("'theta' must be specified"), link = "log")
```

Arguments

zeta	The known value of the additional parameter.
link	desired link function; currently implemented links are: log, identity and sqrt.

Value

A list containing the codes associated with the count distribution assumed for the latent variable N and the chosen link.

raftreg*Random generation from accelerated failure time models*

Description

Function to generate a random sample of survival data from accelerated failure time models.

Usage

```
raftreg(  
  u,  
  formula,  
  baseline,  
  beta,  
  dist = NULL,  
  package = NULL,  
  lwr = 0,  
  upr = Inf,  
  data,  
  ...  
)
```

Arguments

<code>u</code>	a numeric vector of quantiles.
<code>formula</code>	formula specifying the linear predictors.
<code>baseline</code>	the name of the baseline survival distribution.
<code>beta</code>	vector of regression coefficients.
<code>dist</code>	an alternative way to specify the baseline survival distribution.
<code>package</code>	the name of the package where the assumed quantile function is implemented.
<code>lwr</code>	left-truncation time (default to 0 in the absence of left-truncation).
<code>upr</code>	right-truncation time (default to Inf in the absence of right-truncation).
<code>data</code>	data frame containing the covariates used to generate the survival times.
<code>...</code>	further arguments passed to other methods.

Value

a numeric vector containing the generated random sample.

Examples

```

library(rsurv)
library(dplyr)
set.seed(123)
n <- 1000
simdata <- data.frame(
  age = rnorm(n),
  sex = sample(c("f", "m"), size = n, replace = TRUE)
) %>%
  mutate(
    t = raftreg(runif(n), ~ age+sex, beta = c(1, 2),
               dist = "weibull", shape = 1.5, scale = 1),
    c = runif(n, 0, 10)
  ) %>%
  rowwise() %>%
  mutate(
    time = min(t, c),
    status = as.numeric(time == t)
  )
glimpse(simdata)

```

rahreg

*Random generation from accelerated hazard models***Description**

Function to generate a random sample of survival data from accelerated hazard models.

Usage

```

rahreg(
  u,
  formula,
  baseline,
  beta,
  dist = NULL,
  package = NULL,
  lwr = 0,
  upr = Inf,
  data,
  ...
)

```

Arguments

u a numeric vector of quantiles.

formula formula specifying the linear predictors.

baseline	the name of the baseline survival distribution.
beta	vector of regression coefficients.
dist	an alternative way to specify the baseline survival distribution.
package	the name of the package where the assumed quantile function is implemented.
lwr	left-truncation time (default to 0 in the absence of left-truncation).
upr	right-truncation time (default to Inf in the absence of right-truncation).
data	data frame containing the covariates used to generate the survival times.
...	further arguments passed to other methods.

Value

a numeric vector containing the generated random sample.

Examples

```
library(rsurv)
library(dplyr)
set.seed(123)
n <- 1000
simdata <- data.frame(
  age = rnorm(n),
  sex = sample(c("f", "m"), size = n, replace = TRUE)
) %>%
  mutate(
    t = rahreg(runif(n), ~ age+sex, beta = c(1, 2),
              dist = "weibull", shape = 1.5, scale = 1),
    c = runif(n, 0, 10)
  ) %>%
  rowwise() %>%
  mutate(
    time = min(t, c),
    status = as.numeric(time == t)
  )
glimpse(simdata)
```

Description

Density, distribution function, quantile function and random generation for the Rayleigh distribution with scale parameter σ .

Usage

```

drayleigh(x, sigma, log = FALSE)

prayleigh(q, sigma, lower.tail = TRUE, log.p = FALSE)

qrayleigh(p, sigma, lower.tail = TRUE, log.p = FALSE)

rrayleigh(n, sigma)

```

Arguments

x, q	vector of quantiles.
sigma	scale parameter. Must be positive.
log, log.p	logical; if TRUE, probabilities/densities p are returned as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.
p	vector of probabilities.
n	number of observations. If $\text{length}(n) > 1$, the length is taken to be the number required.

Details

The Rayleigh distribution is a continuous probability distribution for non-negative random variables. It arises as the distribution of the magnitude of a two-dimensional vector whose components are independent, identically distributed Gaussian random variables with zero mean.

The probability density function is given by:

$$f(x) = \frac{x}{\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

for $x \geq 0$ and $\sigma > 0$.

The cumulative distribution function is:

$$F(x) = 1 - \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

Value

drayleigh gives the density, prayleigh gives the distribution function, qrayleigh gives the quantile function, and rrayleigh generates random deviates.

The length of the result is determined by n for rrayleigh, and is the maximum of the lengths of the numerical arguments for the other functions.

References

Rayleigh, Lord (1880). On the resultant of a large number of vibrations of the same pitch and of arbitrary phase. *Philosophical Magazine*, 10(60), 73-78.

Examples

```
# Density at x = 1 with sigma = 1
drayleigh(1, sigma = 1)

# CDF at x = 1
prayleigh(1, sigma = 1)

# Quantile for p = 0.5 (median)
qrayleigh(0.5, sigma = 1)

# Generate 10 random values
rrayleigh(10, sigma = 1)
```

rehreg

*Random generation from extended hazard models***Description**

Function to generate a random sample of survival data from extended hazard models.

Usage

```
rehreg(
  u,
  formula,
  baseline,
  beta,
  phi,
  dist = NULL,
  package = NULL,
  lwr = 0,
  upr = Inf,
  data,
  ...
)
```

Arguments

u	a numeric vector of quantiles.
formula	formula specifying the linear predictors.
baseline	the name of the baseline survival distribution.
beta	vector of regression coefficients.
phi	vector of regression coefficients.
dist	an alternative way to specify the baseline survival distribution.
package	the name of the package where the assumed quantile function is implemented.

lwr	left-truncation time (default to 0 in the absence of left-truncation).
upr	right-truncation time (default to Inf in the absence of right-truncation).
data	data frame containing the covariates used to generate the survival times.
...	further arguments passed to other methods.

Value

a numeric vector containing the generated random sample.

Examples

```
library(rsurv)
library(dplyr)
set.seed(123)
n <- 1000
simdata <- data.frame(
  age = rnorm(n),
  sex = sample(c("f", "m"), size = n, replace = TRUE)
) %>%
  mutate(
    t = rehref(runif(n), ~ age+sex, beta = c(1, 2), phi = c(-1, 2),
              dist = "weibull", shape = 1.5, scale = 1),
    c = runif(n, 0, 10)
  ) %>%
  rowwise() %>%
  mutate(
    time = min(t, c),
    status = as.numeric(time == t)
  )
glimpse(simdata)
```

rfrailty

Frailties random generation

Description

The frailty function for adding a simple random effects term to the linear predictor of a given survival regression model.

Usage

```
rfrailty(
  cluster,
  frailty = c("gamma", "gaussian", "ps"),
  sigma = 1,
  alpha = NULL,
  ...
)
```

Arguments

cluster	a vector determining the grouping of subjects (always converted to a factor object internally).
frailty	the frailty distribution; current implementation includes the gamma (default), lognormal and positive stable (ps) distributions.
sigma	standard deviation assumed for the frailty distribution; sigma = 1 by default; this value is ignored for positive stable (ps) distribution.
alpha	stability parameter of the positive stable distribution; alpha must lie in (0,1) interval and an NA is return otherwise.
...	further arguments passed to other methods.

Value

a vector with the generated frailties.

rinterval	<i>Random generation of type I and type II interval censored survival data</i>
-----------	--

Description

Function to generate a random sample of type I and type II interval censored survival data.

Usage

```
rinterval(time, tau, type = c("I", "II"), prob)
```

Arguments

time	a numeric vector of survival times.
tau	either a vector of censoring times (for type I interval-censored survival data) or time grid of scheduled visits (for type II interval censored survival data).
type	type of interval-censored survival data (I or II).
prob	= 0.5 attendance probability of scheduled visit; ignored when type = I.

Value

a data.frame containing the generated random sample.

`rphreg`*Random generation from proportional hazards models*

Description

Function to generate a random sample of survival data from proportional hazards models.

Usage

```
rphreg(  
  u,  
  formula,  
  baseline,  
  beta,  
  dist = NULL,  
  package = NULL,  
  lwr = 0,  
  upr = Inf,  
  data,  
  ...  
)
```

Arguments

<code>u</code>	a numeric vector of quantiles.
<code>formula</code>	formula specifying the linear predictors.
<code>baseline</code>	the name of the baseline survival distribution.
<code>beta</code>	vector of regression coefficients.
<code>dist</code>	an alternative way to specify the baseline survival distribution.
<code>package</code>	the name of the package where the assumed quantile function is implemented.
<code>lwr</code>	left-truncation time (default to 0 in the absence of left-truncation).
<code>upr</code>	right-truncation time (default to Inf in the absence of right-truncation).
<code>data</code>	data frame containing the covariates used to generate the survival times.
<code>...</code>	further arguments passed to other methods.

Value

a numeric vector containing the generated random sample.

Examples

```

library(rsurv)
library(dplyr)
set.seed(123)
n <- 1000
simdata <- data.frame(
  age = rnorm(n),
  sex = sample(c("f", "m"), size = n, replace = TRUE)
) %>%
  mutate(
    t = rphreg(runif(n), ~ age+sex, beta = c(1, 2),
              dist = "weibull", shape = 1.5, scale = 1),
    c = runif(n, 0, 10)
  ) %>%
  rowwise() %>%
  mutate(
    time = min(t, c),
    status = as.numeric(time == t)
  )
glimpse(simdata)

```

rporeg

*Random generation from proportional odds models***Description**

Function to generate a random sample of survival data from proportional odds models.

Usage

```

rporeg(
  u,
  formula,
  baseline,
  beta,
  dist = NULL,
  package = NULL,
  lwr = 0,
  upr = Inf,
  data,
  ...
)

```

Arguments

u a numeric vector of quantiles.

formula formula specifying the linear predictors.

baseline	the name of the baseline survival distribution.
beta	vector of regression coefficients.
dist	an alternative way to specify the baseline survival distribution.
package	the name of the package where the assumed quantile function is implemented.
lwr	left-truncation time (default to 0 in the absence of left-truncation).
upr	right-truncation time (default to Inf in the absence of right-truncation).
data	data frame containing the covariates used to generate the survival times.
...	further arguments passed to other methods.

Value

a numeric vector containing the generated random sample.

Examples

```
library(rsurv)
library(dplyr)
set.seed(123)
n <- 1000
simdata <- data.frame(
  age = rnorm(n),
  sex = sample(c("f", "m"), size = n, replace = TRUE)
) %>%
  mutate(
    t = rporeg(runif(n), ~ age+sex, beta = c(1, 2),
              dist = "weibull", shape = 1.5, scale = 1),
    c = runif(n, 0, 10)
  ) %>%
  rowwise() %>%
  mutate(
    time = min(t, c),
    status = as.numeric(time == t)
  )
glimpse(simdata)
```

Description

Function to generate a random sample of survival data from Yang and Prentice models.

Usage

```
rypreg(
  u,
  formula,
  baseline,
  beta,
  phi,
  dist = NULL,
  package = NULL,
  lwr = 0,
  upr = Inf,
  data,
  ...
)
```

Arguments

<code>u</code>	a numeric vector of quantiles.
<code>formula</code>	formula specifying the linear predictors.
<code>baseline</code>	the name of the baseline survival distribution.
<code>beta</code>	vector of short-term regression coefficients.
<code>phi</code>	vector of long-term regression coefficients.
<code>dist</code>	an alternative way to specify the baseline survival distribution.
<code>package</code>	the name of the package where the assumed quantile function is implemented.
<code>lwr</code>	left-truncation time (default to 0 in the absence of left-truncation).
<code>upr</code>	right-truncation time (default to Inf in the absence of right-truncation).
<code>data</code>	data frame containing the covariates used to generate the survival times.
<code>...</code>	further arguments passed to other methods.

Value

a numeric vector containing the generated random sample.

Examples

```
library(rsurv)
library(dplyr)
set.seed(123)
n <- 1000
simdata <- data.frame(
  age = rnorm(n),
  sex = sample(c("f", "m"), size = n, replace = TRUE)
) %>%
  mutate(
    t = rypreg(runif(n), ~ age+sex, beta = c(1, 2), phi = c(-1, 2),
              dist = "weibull", shape = 1.5, scale = 1),
```

```
      c = runif(n, 0, 10)
    ) %>%
  rowwise() %>%
  mutate(
    time = min(t, c),
    status = as.numeric(time == t)
  )
glimpse(simdata)
```

Index

bernoulli, [3](#)

dfatigue (fatigue), [4](#)
dloglogistic (loglogistic), [6](#)
drayleigh (rayleigh), [11](#)

fatigue, [4](#)

inv_pgf, [5](#)

loglogistic, [6](#)
lp, [7](#)

negbin, [8](#)

pfatigue (fatigue), [4](#)
ploglogistic (loglogistic), [6](#)
prayleigh (rayleigh), [11](#)

qfatigue (fatigue), [4](#)
qloglogistic (loglogistic), [6](#)
qrayleigh (rayleigh), [11](#)

raftreg, [9](#)
rahref, [10](#)
rayleigh, [11](#)
rehreg, [13](#)
rfatigue (fatigue), [4](#)
rfrailty, [14](#)
rinterval, [15](#)
rloglogistic (loglogistic), [6](#)
rphreg, [16](#)
rporeg, [17](#)
rrayleigh (rayleigh), [11](#)
rsurv (rsurv-package), [3](#)
rsurv-package, [3](#)
rypreg, [18](#)