

# Package ‘TwoArmSurvSim’

October 12, 2022

**Title** Simulate Survival Data for Randomized Clinical Trials

**Version** 0.2

**Description** A system to simulate clinical trials with time to event endpoints. Event simulation is based on Cox models allowing for covariates in addition to the treatment or group factor. Specific drop-out rates (separate from administrative censoring) can be controlled in the simulation. Other features include stratified randomization, non-proportional hazards, different accrual patterns, and event projection (timing to reach the target event) based on interim data.

**License** GPL ( $\geq 2$ )

**Depends** R ( $\geq 4.0.0$ ),blockrand,dplyr,survival,simsurv

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Bo Zhang [cre, aut],  
Yi Zhong [aut, ctb],  
Ginny Peng [aut, ctb],  
Bin Yao [aut, ctb]

**Maintainer** Bo Zhang <bzhang@pumabiotechnology.com>

**Repository** CRAN

**Date/Publication** 2021-02-26 08:50:06 UTC

## R topics documented:

censordata . . . . .	2
sensor_surv . . . . .	3
column_freq . . . . .	3
cov_simu . . . . .	4
dummy_convert . . . . .	4
LambdaCensor . . . . .	5
LambdaCensor_betapw . . . . .	6
linear_accrual . . . . .	6
objfunction . . . . .	7

objfunction_betpw . . . . .	8
projection . . . . .	8
projection_simulation . . . . .	9
randomize_trt . . . . .	10
randomize_trt2 . . . . .	11
run_simulation . . . . .	11
run_simulation_simsurv . . . . .	13
self_blockrand . . . . .	15
step_accrual . . . . .	16
surv_data_simulation . . . . .	16
trial_data_simulation . . . . .	17
weibullsim . . . . .	18
weibullsim_betapw . . . . .	19
weibullsim_pw . . . . .	19

<b>Index</b>	<b>20</b>
--------------	-----------

---

censordata	<i>Censor Events Given a Fixed Dropout Rate</i>
------------	---

---

## Description

Censor events given a fixed dropout rate

## Usage

```
censordata(simdata,lambda,gamma, dropoutrate,ebx=1,gammac=1,groupfreq=1,
censordist='exponential',timeinterval=NULL,HRPW=FALSE)
```

## Arguments

simdata	Simulated event dataset
lambda	lambda for event hazard function
gamma	gamma for event hazard function
dropoutrate	Patient dropout rate with range [0,1). If dropoutrate contains only one number. The program will control the dropout rate at population level(treatment + control). If dropoutrate contains two numbers (ie. c(0.2,0.1)), the program will control the dropout rate of control and treatment arm seperately, with the first dropout rate number for control and the second number for treatment. Default value is "0" (no dropout)
ebx	exp(beta*x), if there's no covariates, ebx=1.
groupfreq	frequence of each level of ebx value
timeinterval	time intervals for piecewise baseline hazard function
gammac	gamma for censor hazard function. Default is 1 (exponential)

censordist      censor hazard distribution. Default is exponential  
 HRPW             Indicator of piecewise hazard ratios. TURE for piecewise. FALSE for non piecewise

censor\_surv                       *censor survival event*

**Description**

    censor event time

**Usage**

    censor\_surv(eventtime, censortime, x)

**Arguments**

eventtime        vector of event times  
 censortime        vector of censor time  
 x                    covariates matrix

column\_freq                       *Calculate the Combination Frequency of Several Columns within a Data Frame*

**Description**

    Calculate the combination frequency of several columns in a data frame.

**Usage**

    column\_freq(x, namelist, keepID=FALSE)

**Arguments**

x                    Input data as a data frame.  
 namelist            A list of column names that need to be counted.  
 keepID              If TRUE, the output will keep the unique ID for the column combination.

---

cov_simu	<i>Simulate Covariates Matrix Based on User Provided Factor Information</i>
----------	---

---

**Description**

simulate covariates matrix based on user provided factor information

**Usage**

```
cov_simu(sample_size = sample_size, factors=factors)
```

**Arguments**

sample_size	Total number of patients
factors	A list contains basic information about the covariate factors. Each element should have factor's name, number of levels and their frequency, as well as the hazard ratio to the reference group. Factors should be categorical data.

**Examples**

```
f1<-list(name='Region', N_level=3, prevalence=c(0.1,0.2,0.7), HR=c(1,1,1), strata=TRUE)
f2<-list(name='Gender', N_level=2, prevalence=c(0.5,0.5), HR=c(1,0.9), strata=TRUE)
factors<-list(f1,f2)
cov_simu(sample_size=300, factors=factors)
```

---

dummy_convert	<i>Convert Categorical Data to Dummy Variables</i>
---------------	--

---

**Description**

Convert Categorical Data to Dummy Variables

**Usage**

```
dummy_convert(dataset, column_names)
```

**Arguments**

dataset	Data frame that contains the categorical columns
column_names	A list of column names that need to be converted to dummy variables.

**Examples**

```
x<-data.frame(trt=as.factor(rbinom(100,1,0.5)), Gender=as.factor(rbinom(100,1,0.5)))
dummy_convert(x,c("trt","Gender"))
```

---

LambdaCensor	<i>Find the Censor Hazard Function Parameter for Proportional Hazard Model Given a Fixed Dropout Rate</i>
--------------	---

---

**Description**

calculate censor hazard function parameter for a given dropout rate.

**Usage**

```
LambdaCensor(lambda=lambda,gamma=gamma,theta=theta,ebx=1,gammac=1,
  groupfreq=1,censordist='exponential',timeinterval=NULL)
```

**Arguments**

lambda	lambda for event hazard function
gamma	gamma for event hazard function
gammac	gamma for censor hazard function. This is required if the censoring hazard function is weibull
theta	Dropout rate
ebx	$\exp(\beta \cdot x)$ , if there's no covariates, ebx=1.
groupfreq	frequency of each level of ebx value
censordist	censor hazard function distribution
timeinterval	time intervals for piecewise baseline hazard function

**References**

Wan F. (2017) *Simulating survival data with predefined censoring rates for proportional hazards models*. *Statist. Med.* 2017; 36(5): 838-854

Martinez EZ, Achcar JA, de Oliveira Peres MV, de Queiroz JAM (2016) *A brief note on the simulation of survival data with a desired percentage of right-censored data*. *Journal of Data Science* . 2016, Vol. 14 Issue 4, p701-712. 12p

**Examples**

```
lambdac<-LambdaCensor(lambda=0.03,gamma=1,theta=0.2)
```

---

LambdaCensor\_betapw     *Find the Censor Hazard Function Parameter for Nonproportional Hazard Model Given a Fixed Dropout Rate*

---

### Description

calculate censor hazard function parameter for a given dropout rate. (for piecewise hazard ratios)

### Usage

```
LambdaCensor_betapw(lambda=lambda,gamma=gamma,theta=theta,ebx=1,
  gammac=1,groupfreq=1,censordist='exponential',timeinterval=NULL)
```

### Arguments

lambda	lambda for event hazard function
gamma	gamma for event hazard function
gammac	gamma for censor hazard function. This is required if the censoring hazard function is weibull
theta	Dropout rate
ebx	$\exp(\beta x)$ , if there's no covariates, $\text{ebx}=1$ .
groupfreq	frequency of each level of ebx value
censordist	censor hazard function distribution
timeinterval	time intervals for piecewise baseline hazard function

### References

Wan F. (2017) *Simulating survival data with predefined censoring rates for proportional hazards models*. *Statist. Med.* 2017; 36(5): 838-854

Martinez EZ, Achcar JA, de Oliveira Peres MV, de Queiroz JAM (2016) *A brief note on the simulation of survival data with a desired percentage of right-censored data*. *Journal of Data Science* . 2016, Vol. 14 Issue 4, p701-712. 12p

---

linear\_accrual     *Simulate Accrual Time Where the Accrual Rate is Linearly Increased*

---

### Description

simulate accrual time where the accrual rate is linearly increased

**Usage**

```
linear_accrual(np,rampupt,acceleration)
```

**Arguments**

np	Total number of patients
rampupt	The length of the ramp up period.
acceleration	The acceleration of the accrual rate (increase of each time unit).

**Examples**

```
linear_accrual(np=200,rampupt=10,acceleration=5)
```

---

objfunction	<i>Objective Function for the Finding of Censor Hazard Function Parameter for Proportional Hazard Model</i>
-------------	---

---

**Description**

objective function for the finding of censor hazard function parameter.

**Usage**

```
objfunction(x,lambda,gamma,gammac,theta,ebx,groupfreq,censordist,timeinterval)
```

**Arguments**

x	Censor hazard function parameter. For exponential or weibull censor hazard, x is lambda, for uniform hazard, x is the maximum time of the censoring window.
lambda	lamda for event hazard function
gamma	gamma for event hazard function
gammac	gamma for censor hazard function. This is required if the censoring hazard function is weibull
theta	Dropout rate
ebx	$\exp(\beta x)$ , if there's no covariates, ebx=1.
groupfreq	frequence of each level of ebx value
censordist	censor hazard function distribution
timeinterval	time intervals for piecewise baseline hazard function

---

objfunction_betpw	<i>Objective Function for The Finding of Censor Hazard Function Parameter for Nonproportional Hazard Model</i>
-------------------	--

---

**Description**

objective function for the finding of censor hazard function parameter.

**Usage**

```
objfunction_betpw(x, lambda, gamma, gammac, theta, ebx, groupfreq, censordist, timeinterval)
```

**Arguments**

x	Censor hazard function parameter. For exponential or weibull censor hazard, x is lambda, for uniform hazard, x is the maximum time of the censoring window.
lambda	lamda for event hazard function
gamma	gamma for event hazard function
gammac	gamma for censor hazard function. THis is required if the censoring hazard function is weibull
theta	Dropout rate
ebx	exp(beta*x), if there's no covariates, ebx=1.
groupfreq	frequence of each level of ebx value
censordist	censor hazard function distribution
timeinterval	time intervals for piecewise baseline hazard function

---

projection	<i>Event and Trial Projection</i>
------------	-----------------------------------

---

**Description**

Trial projection given current snapshot dataset.

**Usage**

```
projection(snapshot_data, enroll_continue=FALSE, samplesize=0, rand_ratio=c(1,1),
blocksize=1, accrual_interval=NULL, accrual_rate=NULL, lambda=NULL,
trtTHR=NULL, dropoutrate=NULL, eventtarget=0, maxlpfollowup=NULL)
```



**Arguments**

snapshot_data	Snapshot dataset in data frame. Data frame must include column "time", "status" and "accrual". Optional column "onstudy" is indicator of patients are still on study.
rand_ratio	randomization ratio: control vs treatment
enroll_continue	Indicate whether trial is still enrolling new patients. Default is FALSE. if TRUE, user needs provide "samplesize", "rand_ratio", "blocksize", "accrual_interval", "accrual_rate".
samplesize	Total sample size of the trial. only needed if "enroll_continue=TRUE"
blocksize	Randomization blocksize, only needed if "enroll_continue=TRUE"
accrual_interval	accrual time windows. only needed if "enroll_continue=TRUE"
accrual_rate	accrual rate for each time window. only needed if "enroll_continue=TRUE"
lambda	Baseline hazard function parameter for exponential distribution
trtHR	hazard ratio between treatment and control
dropoutrate	Dropout rate
eventtarget	Total target event number
maxlpfollowup	Maximum followup time for last enrolled patient

---

projection\_simulation *Project Final Event Numbers or Trial Stop Time Based on Interim Analysis Data*

---

**Description**

Project final event numbers or trial stop time based on interim analysis data

**Usage**

```
projection_simulation(snapshot_data, rand_ratio=c(1,1), enroll_continue=FALSE,
samplesize=0, blocksize=1,accrual_interval=NULL, accrual_rate=NULL, lambda=NULL,
trtHR=NULL, dropoutrate=NULL,eventtarget=NULL,maxlpfollowup=NULL,N_simulation=1)
```

**Arguments**

snapshot_data	Snapshot dataset or interim analysis dataset
rand_ratio	Randomization ratio between control and treatment
blocksize	The value of this parameter is used to define the size of the randomization blocks. The actual blocksize is number of treatment levels multiplied by this parameter. Please refer to "blockrand" package for detailed usage.

enroll_continue	True if trial is still in the enrollment period
samplesize	if enroll_continue=TRUE, please provide the total sample size of the trial.
trtHR	Hazard ratio between treatment groups (treatment vs control)
accrual_interval	Time windows for accrual
accrual_rate	accrual rate for each accrual time window
lambda	lambda for event hazard function (exponential)
dropoutrate	Patient dropout rate with range [0,1). If dropoutrate contains only one number. The program will control the dropout rate at population level(treatment + control). If dropoutrate contains two numbers (ie. c(0.2,0.1)), the program will control the dropout rate of control and treatment arm seperately, with the first dropout rate number for control and the second number for treatment. Default value is "0" (no dropout)
eventtarget	Number of target events
maxlpfollowup	maximum follow up time for the last enrolled patient
N_simulation	number of simulations to run

---

randomize_trt	<i>Generate Block Randomized Treatment Label Based on Covariates Matrix</i>
---------------	---

---

### Description

Generate block randomized treatment label based on covariates matrix

### Usage

```
randomize_trt(cov_mat=cov_mat,blocksize=blocksize,trtHR=trtHR,rand_ratio=c(1,1))
```

### Arguments

cov_mat	Covariates matrix.
blocksize	Randomization block size
trtHR	Hazard ratio between treatment arms.
rand_ratio	Randomization ratio between control and treatment

---

randomize_trt2	<i>Generate Block Randomized Treatment Label Based on Covariates Matrix for Two Arm Trial</i>
----------------	---

---

**Description**

Generate block randomized treatment label based on covariates matrix for two arm trial.

**Usage**

```
randomize_trt2(cov_mat=cov_mat,blocksize=blocksize,rand_ratio=c(1,1))
```

**Arguments**

cov_mat	Covariates matrix.
blocksize	Randomization block size
rand_ratio	Randomization ratio between control and treatment

---

run_simulation	<i>Run Clinical Trial Simulations Based on User Defined Trial Settings</i>
----------------	--

---

**Description**

Runs single or multiple clinical trial (Time to event endpoint) simulations based on the clinical trial settings. Trial data summary will be provided for each simulation. Cox model will be fitted after trial simulation. If stratification factors were provided, stratified cox model results will also be provided. If "N\_simulation" is set to 1, one simulation dataset will be generated.

**Usage**

```
run_simulation(samplesize, rand_ratio=c(1,1), blocksize, factors=NULL, trtHR=trtHR,
trt_timeinterval=NULL, accrual_interval=NULL, accrual_rate=NULL, rampuptime=NULL,
acceleration=NULL, lambda, gamma, timeinterval=NULL, dropoutrate=0, gammac=1,
censordist='exponential', eventtarget=NULL,maxlpfollowup=NULL, N_simulation=1,
alpha=0.05)
```

**Arguments**

samplesize	Total number of patients in the simulated clinical trial
rand_ratio	Randomization ratio between control and treatment
blocksize	The value of this parameter is used to define the size of the randomization blocks. The actual blocksize is number of treatment levels multiplied by this parameter. Please refer to "blockrand" package for detailed usage.
factors	stratification factors. Default is NULL
trtHR	Hazard ratio between treatment groups (treatment vs control)
trt_timeinterval	Time windows for trtHR when trtHR is piecewise. Always start with time 0. Example: c(0,10,30)
accrual_interval	Time windows for accrual
accrual_rate	accrual rate for each accrual time window
rampuptime	rampup time for linear increased accrual
acceleration	acceleration rate for linear increased accrual
lambda	lambda for event hazard function
gamma	gamma for event hazard function
timeinterval	time intervals for piecewise baseline hazard function
dropoutrate	Patient dropout rate with range [0,1). If dropoutrate contains only one number. The program will control the dropout rate at population level(treatment + control). If dropoutrate contains two numbers (ie. c(0.2,0.1)), the program will control the dropout rate of control and treatment arm separately, with the first dropout rate number for control and the second number for treatment. Default value is "0" (no dropout)
gammac	gamma for censor hazard function. Default is 1 (exponential)
censordist	censor hazard distribution. Can be "weibull", "exponential" or "uniform". Default is exponential
eventtarget	Number of target events
maxlpfollowup	maximum follow up time for the last enrolled patient
N_simulation	number of simulations to run
alpha	Two sided alpha for testing power calculation

**Value**

TrilInfo	Summary of the simulated trial data
ModelResult	Cox model results comparing treatment vs control
StraModelResult	Stratified Cox model results comparing treatment vs control
Data	simulated dataset only if "N_simulateion" is set to 1

## Examples

```
f1<-list(name='Region', N_level=3, prevalence=c(0.1,0.2,0.7), HR=c(1,0.7,0.9), strata=TRUE)
f2<-list(name='Gender', N_level=2, prevalence=c(0.5,0.5), HR=c(1,0.9), strata=TRUE)
f3<-list(name='Stage', N_level=4, prevalence=c(0.2,0.25,0.3,0.25), HR=c(1,1.05,1.3,1.5),
strata=TRUE)

factors<-list(f1,f2,f3)

samplesize<-400
blocksize<-2
accrual_interval<-c(0,5,10)
accrual_rate<-c(5,10,20)
trtHR<-0.7
lambda<-0.03
gamma<-1.2
dropoutrate<-0.2
eventtarget<-240
N_simulation<-10

out<-run_simulation(samplesize=samplesize,blocksize=blocksize,factors=factors,
accrual_interval=accrual_interval,accrual_rate=accrual_rate, trtHR=trtHR, lambda=lambda,
gamma=gamma,dropoutrate=dropoutrate,eventtarget=eventtarget,N_simulation=N_simulation)
```

---

```
run_simulation_simsurv
```

*Run Clinical Trial Simulations Based on survival data generated by  
simsurv package*

---

## Description

Runs single or mutiple clinical trial (Time to event endpoint) simulations based survival time generated by simsurv package. Trial data summary will be provided for each simulation. Cox model will be fitted afther trial simulation. If stratification factors were provided, stratified cox model results will also be provided. If "N\_simulation" is set to 1, one simulation dataset will be generated.

## Usage

```
run_simulation_simsurv(samplesize, rand_ratio=c(1,1), blocksize, factors=NULL,
accrual_interval=NULL,accrual_rate=NULL, eventtarget=NULL,maxlpfollowup=NULL,
N_simulation=1,alpha=0.05,simsurv1=NULL, simsurv2=NULL)
```

**Arguments**

samplesize	Total number of patients in the simulated clinical trial
rand_ratio	Randomization ratio between control and treatment
blocksize	The value of this parameter is used to define the size of the randomization blocks. The actual blocksize is number of treatment levels multiplied by this parameter. Please refer to "blockrand" package for detailed usage.
factors	stratification factors. Default is NULL
accrual_interval	Time windows for accrual
accrual_rate	accrual rate for each accrual time window
eventtarget	Number of target events
maxlpfollowup	maximum follow up time for the last enrolled patient
N_simulation	number of simulations to run
alpha	Two sided alpha for testing power calculation
simsurv1	simsurv command to generate survival time. Design matrix should set to "x". Please refer to examples.
simsurv2	simsurv command to generate dropout time.

**Value**

TrilInfo	Summary of the simulated trial data
ModelResult	Cox model results comparing treatment vs control
StraModelResult	Stratified Cox model results comparing treatment vs control
Data	simulated dataset only if "N_simulation" is set to 1

**Examples**

```
# Example 1, compare simsurv and TwoArmSurvSim, trtHR=0.7 eventtarget=247, power should be 0.8

f1<-list(name='Gender', N_level=2, prevalence=c(0.5,0.5), HR=c(1,0.9), strata=TRUE)
factors=list(f1)

samplesize = 400
blocksize = 2
accrual_interval = c(0,5,10)
accrual_rate = c(5,10,20)
eventtarget = 247
N_simulation = 1

# Simsurv
```

```

simsurv1 <- "simsurv(lambdas = 0.03, gammas = 1,
betas = c(trt = log(0.7),Gender.1=log(0.9)),x = x)"
simsurv2 <-NULL

out<-run_simulation_simsurv(samplesize=samplesize,blocksize=blocksize,factors=factors,
accrual_interval=accrual_interval,accrual_rate=accrual_rate, eventtarget=eventtarget,
N_simulation=N_simulation,simsurv1=simsurv1,simsurv2=simsurv2)

# example 2, Time dependent treatment effect.
# h(t)=h0(t)*exp(beta0*x+beta1*x*log(t)). beta0=log(0.7), beta1=0.15

simsurv1 <- "simsurv( lambdas = 0.1, gammas = 1.5,betas = c(trt = log(0.7)),
x = x, tde = c(trt = 0.15),tdefunction = \"log\")"
simsurv2 <-NULL

```

---

self_blockrand	<i>Block Randomization</i>
----------------	----------------------------

---

### Description

Block randomization

### Usage

```
self_blockrand(N=N,trt_levels=trt_levels,blocksize=blocksize,rand_ratio=rand_ratio)
```

### Arguments

N	Total number of patients
trt_levels	treatment levels in vector. for example, c(0,1) for control vs treatment trial.
blocksize	Randomization block size
rand_ratio	Randomization ratio between control and treatment

### Examples

```
self_blockrand(N=100, trt_levels=c(0,1),blocksize=2,rand_ratio=c(1,1))
```

---

step\_accrual                      *Simulate Accrual Time*

---

**Description**

Simulate accrual time

**Usage**

```
step_accrual(np, tt, arate)
```

**Arguments**

np	Total number of patients
tt	Time windows
arate	accrual rate for each time windows

**Examples**

```
timeinterval<-c(0,5,10)
accrualrate<-c(10,15,20)
N<-200
accrual<-step_accrual(N,timeinterval,accrualrate)
```

---

surv\_data\_simulation      *Simulate Survival Data*

---

**Description**

Simulate survival data given design matrix and covariates (betas).

**Usage**

```
surv_data_simulation(lambda,gamma,x,betas=NULL, dropoutrate=0,
gammac=1,censordist='exponential',timeinterval=NULL,trt_timeinterval=NULL)
```



**Arguments**

lambda	lambda for event hazard function
gamma	gamma for event hazard function
x	design matrix
betas	coefficients for the covariates. The length of betas should be the same of number of columns of x
dropoutrate	Patient dropout rate with range [0,1). If dropoutrate contains only one number. The program will control the dropout rate at population level(treatment + control). If dropoutrate contains two numbers (ie. c(0.2,0.1)), the program will control the dropout rate of control and treatment arm separately, with the first dropout rate number for control and the second number for treatment. Default value is "0" (no dropout)
gammac	gamma for censor hazard function. Default is 1 (exponential)
censordist	censor hazard distribution. Default is exponential
timeinterval	time intervals if the baseline hazard function is piecewise.
trt_timeinterval	Time windows for piecewise hazard ratios

**Examples**

```

N<-400
x<-data.frame(arm=rbinom(N,1,0.5), factor1=rbinom(N,1,0.7), factor2=rbinom(N,1,0.8))
betas<-c(arm=-0.35667, factor1=0.3, factor2=-0.1)

data<-surv_data_simulation(lambda=0.2, gamma=2, x=x, betas=betas, dropoutrate=0.2)

```

---

trial\_data\_simulation *Simulate Clinical Trial with Accrual Time and Trial Stop Rules*

---

**Description**

Simulate clinical trial by adding accrual time to the simulated survival data. And cut the trial at the target event time or at the maximum follow up time

**Usage**

```
trial_data_simulation(simdata, accrual, eventtarget=NULL, maxlpfollowup=NULL)
```

**Arguments**

simdata	A data frame contains simulated survival time and censor status.
accrual	Simulated accrual time.
eventtarget	Target number of events of the trial
maxlpfollowup	The maximum follow up time for the last enrolled patient.

---

weibullsim	<i>Simulate Event Time with weibull Hazard Function</i>
------------	---

---

**Description**

Simulate event time with weibull hazard function

**Usage**

```
weibullsim(N=NULL,lambda,gamma,x=NULL,betas=NULL)
```

**Arguments**

N	Total number of patients
lambda	lambda for event hazard function
gamma	gamma for event hazard function
x	design matrix
betas	covariates for design matrix

**Examples**

```
data<-weibullsim(N=300,lambda=0.03,gamma=0.9)
```

---

weibullsim_betapw	<i>Simulate Event Time with weibull Hazard Function and Piecewise Hazard Ratios</i>
-------------------	---

---

**Description**

Simulate event time with weibull hazard function for piecewise hazard ratios

**Usage**

```
weibullsim_betapw(lambda, gamma, t, x, betas)
```

**Arguments**

lambda	lambda for event hazard function
gamma	gamma for event hazard function
x	design matrix
betas	covariates for design matrix
t	Time windows for piecewise hazard ratios

---

weibullsim_pw	<i>Simulate event time with weibull hazard function. The parameters can be piecewise.</i>
---------------	---

---

**Description**

Simulate event time with weibull hazard function given the piecewise parameters.

**Usage**

```
weibullsim_pw(lambda, gamma, t, x, betas=NULL)
```

**Arguments**

lambda	lambda for event hazard function. It will be a vector for piecewise hazard function.
gamma	gamma for event hazard function. It will be a vector for piecewise hazard function.
x	design matrix
betas	covariates for desing matrix
t	time intervals for piecewise weibull parameters.

# Index

[censor\\_surv](#), [3](#)  
  [censordata](#), [2](#)  
  [column\\_freq](#), [3](#)  
  [cov\\_simu](#), [4](#)  
  
  [dummy\\_convert](#), [4](#)  
  
  [LambdaCensor](#), [5](#)  
  [LambdaCensor\\_betapw](#), [6](#)  
  [linear\\_accrual](#), [6](#)  
  
  [objfunction](#), [7](#)  
  [objfunction\\_betapw](#) ([objfunction\\_betapw](#)),  
    [8](#)  
  [objfunction\\_betpw](#), [8](#)  
  
  [projection](#), [8](#)  
  [projection\\_simulation](#), [9](#)  
  
  [randomize\\_trt](#), [10](#)  
  [randomize\\_trt2](#), [11](#)  
  [run\\_simulation](#), [11](#)  
  [run\\_simulation\\_simsurv](#), [13](#)  
  
  [self\\_blockrand](#), [15](#)  
  [step\\_accrual](#), [16](#)  
  [surv\\_data\\_simulation](#), [16](#)  
  
  [trial\\_data\\_simulation](#), [17](#)  
  
  [weibullsim](#), [18](#)  
  [weibullsim\\_betapw](#), [19](#)  
  [weibullsim\\_pw](#), [19](#)