Package ‘denim’

June 5, 2024

Type Package

Title Generate and Simulate Deterministic Discrete-Time Compartmental Models

Version 1.0.0

Date 2024-05-28

Description R package to build and simulate deterministic discrete-time compartmental models that can be non-Markov. Length of stay in each compartment can be defined to follow a parametric distribution (d_exponential(), d_gamma(), d_weibull(), d_lognormal()) or a non-parametric distribution (nonparametric()). Other supported types of transition from one compartment to another includes fixed transition (constant()), multinomial (multinomial()), fixed transition probability (transprob()).

License MIT + file LICENSE


BugReports https://github.com/thinhong/denim/issues

Imports Rcpp (>= 1.0.6), viridisLite

Suggests covr, knitr, rmarkdown, testthat (>= 3.0.0), xml2, deSolve, DiagrammeR

LinkingTo Rcpp, testthat

Encoding UTF-8

RoxygenNote 7.3.1

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation yes

Author Thinh Ong [aut, cph] (<https://orcid.org/0000-0001-6772-9291>), Anh Phan [aut, cre], Marc Choisy [aut] (<https://orcid.org/0000-0002-5187-6390>), Niels Lohman [ctb], Bjoern Hoehrmann [ctb], Florian Loitsch [ctb], Ingo Berg [ctb]
Maintainer  Anh Phan <anhptq@oucru.org>
Repository  CRAN
Date/Publication  2024-06-05 19:50:10 UTC

Contents

- denim-package .............................................................. 2
- constant ................................................................. 3
- d_exponential ............................................................ 3
- d_gamma ................................................................. 4
- d_lognormal ............................................................. 4
- d_weibull ............................................................... 5
- mathexpr ............................................................... 6
- multinomial .............................................................. 6
- nonparametric ........................................................... 7
- sim ......................................................................... 7
- transprob ............................................................... 9

Index 10

---

denim-package  denim

---

Description

Simulate deterministic discrete time model

Details

Imports

Author(s)

Maintainer:  Anh Phan <anhptq@oucru.org>

Authors:

- Thinh Ong <thinhop@oucru.org> (ORCID) [copyright holder]
- Marc Choisy <mchoisy@oucru.org> (ORCID)

Other contributors:

- Niels Lohman [contributor]
- Bjoern Hoehrmann <bjoern@hoehrmann.de> [contributor]
- Florian Loitsch [contributor]
- Ingo Berg [contributor]
constant

See Also

Useful links:

- https://drthinhong.com/denim/
- https://github.com/thinhong/denim
- Report bugs at https://github.com/thinhong/denim/issues

---

**constant**  
*Fixed transition*

**Description**

Define a fixed number of individuals of the left compartment transit to the right compartment at every time step

**Usage**

countant(x)

**Arguments**

- **x** 
  number of individuals who move from one compartment to another

**Value**

a Distribution object for simulator

**Examples**

transitions <- list("S->I" = constant(10))

d_exponential

**Description**

Discrete exponential distribution

**Usage**

d_exponential(rate)

**Arguments**

- **rate** 
  rate parameter of an exponential distribution
**d_lognormal**

**Value**

a Distribution object for simulator

**Examples**

```r
transitions <- list("I -> D" = d_exponential(0.3))
```

---

**d_gamma**

*Discrete gamma distribution*

**Description**

Discrete gamma distribution

**Usage**

```r
d_gamma(scale, shape)
```

**Arguments**

- `scale` scale parameter of a gamma distribution
- `shape` shape parameter of a gamma distribution

**Value**

a Distribution object for simulator

**Examples**

```r
transitions <- list("S -> I" = d_gamma(1, 5))
```

---

**d_lognormal**

*Discrete log-normal distribution*

**Description**

Discrete log-normal distribution

**Usage**

```r
d_lognormal(mu, sigma)
```
**d_weibull**

**Arguments**

mu  
location parameter or the ln mean

sigma  
scale parameter or ln standard deviation

**Value**

a Distribution object for simulator

**Examples**

transitions <- list("I -> D" = d_lognormal(3, 0.6))

d_weibull  
*Discrete Weibull distribution*

**Description**

Discrete Weibull distribution

**Usage**

d_weibull(scale, shape)

**Arguments**

scale  
scale parameter of a Weibull distribution

shape  
shape parameter of a Weibull distribution

**Value**

a Distribution object for simulator

**Examples**

transitions <- list("I -> D" = d_weibull(0.6, 2))
**mathexpr**  
*Mathematical expression*

**Description**

Mathematical expression

**Usage**

```r
mathexpr(expr)
```

**Arguments**

- `expr`  
  User defined mathematical expression. The expression will be processed by `muparser` library which offers a wide variety of operators. Visit the `muparser` website (https://beltoforion.de/en/muparser/features.php) to see the full list of available operators.

**Value**

- a `Distribution` object for the simulator

**Examples**

```r
transitions <- list("S->I"=mathexpr("beta*S/N"))
params <- c(N = 1000, beta = 0.3)
```

---

**multinomial**  
*Multinomial*

**Description**

Define a set of probabilities of transition from one compartment to multiple compartments

- "I -> R, D" = `multinomial(0.9, 0.1)`
- "I -> R" = `d_gamma(3, 2)`
- "I -> D" = `d_lognormal(2, 0.5)`

is equal to

- "0.9 * I -> R" = `d_gamma(3, 2)`
- "0.1 * I -> D" = `d_lognormal(2, 0.5)`

**Usage**

```r
multinomial(...)
```
nonparametric

Arguments

... a vector of probabilities, must add up to 1

Value

a Distribution object for simulator

Description

Convert a vector of frequencies, percentages... into a distribution

Usage

nonparametric(...)

Arguments

... a vector of values

Value

a Distribution object for simulator

Examples

transitions <- list("S->I"=nonparametric(0.1, 0.2, 0.5, 0.2))

sim

Simulator for deterministic discrete time model with memory

Description

Simulation function that call the C++ simulator

Usage

sim(
    transitions,
    initialValues,
    parameters = NULL,
    simulationDuration,
    timeStep = 1,
    errorTolerance = 0.001
)
Arguments

transitions a list of transitions follows this format "transition" = distribution()
initialValues a vector contains the initial values of all compartments defined in the transitions, follows this format compartment_name = initial_value
parameters a vector contains values of any parameters that are not compartments, usually parameters used in mathexp() functions
simulationDuration duration of time to be simulate
timeStep set the output time interval. For example, if simulationDuration = 10 means 10 days and timeStep = 0.1, the output will display results for each 0.1 daily interval
errorTolerance set the threshold so that a cumulative distribution function can be rounded to 1. For example, if we want a cumulative probability of 0.999 to be rounded as 1, we set errorTolerance = 0.001 (1 - 0.999 = 0.001). Default is 0.001

Value

a data.frame with class denim that can be plotted with a plot() method

Examples

transitions <- list(
  "S -> I" = "beta * S * I / N",
  "I -> R" = d_gamma(3, 2)
)

initialValues <- c(
  S = 999,
  I = 1,
  R = 0
)

parameters <- c(
  beta = 0.012,
  N = 1000
)

simulationDuration <- 30
timeStep <- 0.01

mod <- sim(transitions = transitions,
            initialValues = initialValues,
            parameters = parameters,
            simulationDuration = simulationDuration,
            timeStep = timeStep)
transprob

Transition probability

Description
A fixed percentage of the left compartment transit to the right compartment at every time step

Usage
transprob(x)

Arguments
x a float number between 0 to 1

Value
a Distribution object for simulator

Examples
transitions <- list("S->I"=transprob(0.8))
Index

constant, 3

`d_exponential`, 3
`d_gamma`, 4
`d_lognormal`, 4
`d_weibull`, 5
`denim` (denim-package), 2
`denim-package`, 2

`mathexpr`, 6
`multinomial`, 6

`nonparametric`, 7

`sim`, 7

`transprob`, 9