

Package ‘sims’

September 20, 2021

Title Simulate Data from R or 'JAGS' Code

Version 0.0.3

Description Generates data from R or 'JAGS' code for use in simulation studies. The data are returned as an 'nlist::nlists' object and/or saved to file as individual '.rds' files. Parallelization is implemented using the 'future' package. Progress is reported using the 'progressr' package.

License MIT + file LICENSE

Depends R (>= 3.6)

Imports chk, future.apply, nlist, parallel, stats, yesno

Suggests covr, future, knitr, progressr, rjags, rmarkdown, testthat (>= 3.0.0), withr

VignetteBuilder knitr

Encoding UTF-8

Language en-US

RoxygenNote 7.1.2

Config/testthat/edition 3

NeedsCompilation no

Author Audrey Beliveau [aut],
Joe Thorley [aut, cre] (<<https://orcid.org/0000-0002-7683-4592>>)

Maintainer Joe Thorley <joe@poissonconsulting.ca>

Repository CRAN

Date/Publication 2021-09-20 18:50:02 UTC

R topics documented:

sims_add	2
sims_check	2
sims_copy	3
sims_data	4

sims_data_files	4
sims_info	5
sims_rdist	5
sims_simulate	6

Index	8
--------------	----------

sims_add	<i>Add Simulated Datasets</i>
----------	-------------------------------

Description

Add Simulated Datasets

Usage

```
sims_add(path = ".", nsims = 1)
```

Arguments

path	A string specifying the path to the directory to add the data sets to.
nsims	A count of the number of additional datasets to generate.

Value

A character vector of the names of the files created.

sims_check	<i>Check Simulated Data</i>
------------	-----------------------------

Description

Checks the simulated data argument values in the '.sims.rds' file.

Usage

```
sims_check(path = ".")
```

Arguments

path	A string of the path to the directory with the simulated data.
------	--

Details

The checks include whether number and names of the data files in the directory are consistent with the number of simulations.

Value

An informative error or invisible list of the argument values.

Examples

```
set.seed(10)
sims_simulate("a <- runif(1)",
  save = TRUE, path = tempdir(), exists = NA,
  ask = FALSE
)
(sims_check(tempdir()))
```

sims_copy

Copy Simulated Datasets

Description

Copy Simulated Datasets

Usage

```
sims_copy(
  path_from = ".",
  path_to = paste0(path_from, "_copy"),
  exists = FALSE,
  ask = getOption("sims.ask", TRUE),
  silent = FALSE
)
```

Arguments

path_from	A string of the path to the directory containing the simulated datasets.
path_to	A string of the path to the directory to copy the simulated dataset to.
exists	A flag specifying whether path_to should already exist. If exists = NA it doesn't matter. If the directory already exists sims compatible files are deleted if exists = TRUE or exists = NA otherwise an error is thrown.
ask	A flag specifying whether to ask before deleting files.
silent	A flag specifying whether to suppress warnings.

Value

A character vector of the names of the files copied.

`sims_data`*Simulated Datasets*

Description

Gets the simulated datasets as an `nlist::nlists_object()`. There is no guarantee that all the datasets will fit in memory.

Usage

```
sims_data(path = ".")
```

Arguments

`path` A string of the path to the directory with the simulated data.

Value

An `nlist::nlists_object()` of the simulated datasets.

Examples

```
set.seed(10)
sims_simulate("a <- runif(1)",
  nsims = 10L, path = tempdir(),
  exists = NA, ask = FALSE
)
library(nlist)
sims_data(tempdir())
```

`sims_data_files`*Simulated Data Files*

Description

Gets the names of the simulated data files.

Usage

```
sims_data_files(path = ".")
```

Arguments

`path` A string of the path to the directory with the simulated data.

Value

A character vector of the names of the simulated data files.

Examples

```
set.seed(10)
sims_simulate("a <- runif(1)",
  nsims = 10L, path = tempdir(),
  exists = NA, ask = FALSE
)
sims_data_files(tempdir())
```

sims_info*Simulated Data Argument Values*

Description

Gets the simulated data argument values in the '.sims.rds' file.

Usage

```
sims_info(path = ".")
```

Arguments

path A string of the path to the directory with the simulated data.

Value

A named list of the values in `file.path(path, '.sims.rds')`.

Examples

```
set.seed(10)
sims_simulate("a <- runif(1)", path = tempdir(), exists = NA, ask = FALSE)
sims_info(tempdir())
```

sims_rdists*Sims Random R Distributions*

Description

Gets the names of the R random variate generating functions listed in [Distributions\(\)](#).

Usage

```
sims_rdist
```

Value

A character vector.

Examples

```
sims_rdistts()
```

```
sims_simulate
```

```
Simulate Datasets
```

Description

Simulates datasets using JAGS or R code. By default returns the datasets as an `nlist::nlists_object()`. If path is provided then the datasets are written to the directory as individual `.rds` files.

Usage

```
sims_simulate(
  code,
  constants = nlist::nlist(),
  parameters = nlist::nlist(),
  monitor = ".*",
  stochastic = NA,
  latent = NA,
  nsims = 1,
  save = FALSE,
  path = ".",
  exists = FALSE,
  rdistts = sims_rdistts(),
  ask = getOption("sims.ask", TRUE),
  silent = FALSE
)
```

Arguments

<code>code</code>	A string of the JAGS or R code to generate the data. The JAGS code must not be in a data or model block.
<code>constants</code>	An nlist object (or list that can be coerced to nlist) specifying the values of nodes in code. The values are included in the output dataset.
<code>parameters</code>	An nlist object (or list that can be coerced to nlist) specifying the values of nodes in code. The values are not included in the output dataset.
<code>monitor</code>	A character vector (or regular expression if a string) specifying the names of the nodes in code to include in the dataset. By default all nodes are included.
<code>stochastic</code>	A logical scalar specifying whether to monitor deterministic and stochastic (NA), only deterministic (FALSE) or only stochastic nodes (TRUE).
<code>latent</code>	A logical scalar specifying whether to monitor observed and latent (NA), only latent (TRUE) or only observed nodes (FALSE).
<code>nsims</code>	A whole number between 1 and 1,000,000 specifying the number of data sets to simulate. By default 1 data set is simulated.

save	A flag specifying whether to return the data sets as an <code>nlists</code> object or save in path. If <code>save = NA</code> the datasets are returned as an <code>nlists</code> object and saved in path.
path	A string specifying the path to the directory to save the data sets in.
exists	A flag specifying whether the path directory should already exist (if <code>exists = NA</code> it doesn't matter).
rdists	A character vector specifying the R functions to recognize as stochastic.
ask	A flag specifying whether to ask before deleting sims compatible files.
silent	A flag specifying whether to suppress warnings.

Details

JAGS code is identified by the presence of `'~'` indicating a stochastic variable node. Otherwise code is assumed to be R code and stochastic variable nodes are those where assignment is immediately succeeded by a call to one of the functions named in `rdists`.

Both constants and parameters must be `[nlist::nlist_object]s` (or lists that can be coerced to such) . The only difference between constants and parameters is that the values in constants are appended to the output data while the values in parameters are not. Neither constants or parameters can include missing values nor can they have elements with the same name. Elements which are not in code are dropped with a warning (unless `silent = TRUE` in which case the warning is suppressed).

Each set of simulated data set is written as a separate `.rds` file. The files are labelled `data0000001.rds`, `data0000002.rds`, `data0000003.rds` etc. The argument values are saved in the hidden file `.sims.rds`.

sims compatible files are those matching the regular expression `^(data\\\\d\\{7,7\\})\\.sims\\.rds$`.

Parallelization is implemented using the `future` package.

Value

By default an `nlist::nlists_object()` of the simulated data. Otherwise if `path` is defined saves the datasets as individual `.rds` files and returns `TRUE`.

See Also

[sims_rdistts\(\)](#)

Examples

```
set.seed(101)
sims_simulate("a <- runif(1)", path = tempdir(), exists = NA, ask = FALSE)
```

Index

Distributions(), 5

nlist::nlists_object(), 4, 6, 7

sims_add, 2

sims_check, 2

sims_copy, 3

sims_data, 4

sims_data_files, 4

sims_info, 5

sims_rdists, 5

sims_rdists(), 7

sims_simulate, 6