Package 'sweepr'

June 10, 2020

Title Run Parameter Sweeps on SEIR Models

Version 0.11.7

Description Runs SEIR model and performs parameter sweeps on SEIR model output.

License CC BY 4.0

Depends R (>= 2.10)

VignetteBuilder knitr

Encoding UTF-8

Language en

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.0

Imports adaptivetau,
deSolve,
dplyr,
DT,
forcats,
ggplot2,
htmlwidgets,
janitor,
lhs,
magrittr,
openxlsx,
plotly,
readxl,
scales,
tibble,
tidyr,
triangle

Suggests knitr,
rmarkdown,
klippy

R topics documented:

`Assess.covariate.importance` *Assess covariate importance*

Description

Assess covariate importance

Usage

```
Assess.covariate.importance(don, X, Y, method)
```

Arguments

`don`	A data frame
`X`	A numerical vector
`Y`	A numerical vector
`method`	A string

`ever.been.here` *Another SEIR model function*

Description

Another SEIR model function

Usage

```
ever.been.here(SEIR.object)
```

Arguments

`SEIR.object`	A SEIR model object

fetch.package *Fetch a package*

Description

Fetch a package

Usage

```
fetch.package(
  nom_du_package,
  try.this = c("nothing", "load", "install"),
  verbose = FALSE
)
```

Arguments

nom_du_package	A string
try.this	A string
verbose	A logical

from.tbl.to.df *Convert a tibble to a data frame*

Description

Convert a tibble to a data frame

Usage

```
from.tbl.to.df(data.frame.ish, rename = NULL)
```

Arguments

data.frame.ish	A tibble
rename	A vector or data frame

get.silly.model.chunk *Get model chunk*

Description

Get model chunk

Usage

```
get.silly.model.chunk(
  model_flows,
  init.cond,
  init.cond.numeric.vars,
  which.flow,
  NewFrom,
  NewTo
)
```

Arguments

model_flows	A string
init.cond	A string
init.cond.numeric.vars	A vector string
which.flow	A string
NewFrom	A string
NewTo	A string

get_scatter_plot *Render a scatter plot*

Description

Render a scatter plot

Usage

```
get_scatter_plot(
  x,
  y,
  x_label_text = deparse(substitute(x)),
  y_label_text = deparse(substitute(y)),
  geom_point_size = 2,
  element_text_size = 12,
  width = NULL,
  height = NULL
)
```

Arguments

x	A vector
y	A vector
x_label_text	A string
y_label_text	A string
geom_point_size	A numeric
element_text_size	A numeric
width	A numeric
height	A numeric

`get_tornado_plot` *Render a tornado plot*

Description

Render a tornado plot

Usage

```
get_tornado_plot(
  outcome_variable,
  parameters = parms.tried.df,
  outcomes = outcomes.summary.df,
  method = "kendall-partial-correlation-slow",
  bin_width = 0.5,
  element_text_size = 12,
  order_by_absolute_value = FALSE,
  add_label = FALSE,
  width = NULL,
  height = NULL
)
```

Arguments

outcome_variable	A string
parameters	The parms.tried.df data frame
outcomes	The outcomes.summary.df data frame
method	A string
bin_width	A numeric
element_text_size	A numeric
order_by_absolute_value	A logical
add_label	A logical
width	A numeric
height	A numeric

get_tornado_table *Render a tornado table*

Description

Render a tornado table

Usage

```
get_tornado_table(
  outcome_variable,
  parameters = parms.tried.df,
  outcomes = outcomes.summary.df,
  method = "kendall-partial-correlation-slow"
)
```

Arguments

outcome_variable	A string
parameters	The parms.tried.df data frame
outcomes	The outcomes.summary.df data frame
method	A string

overlap.length *Overlap length function*

Description

Overlap length function

Usage

```
overlap.length(L1, U1, L2, U2)
```

Arguments

L1	A vector
U1	A vector
L2	A vector
U2	A vector

SaveModelInExcel *Save SEIR model as Excel file*

Description

Save SEIR model as Excel file

Usage

```
SaveModelInExcel(input.info.list, file_name, map.names)
```

Arguments

input.info.list	A data frame
file_name	A string
map.names	A vector string

SEIR.n.Age.Classes *Main SEIR model function*

Description

Main SEIR model function

Usage

```
SEIR.n.Age.Classes(
  file.name,
  sheets.names,
  differential.eqns.func = NULL,
  just.get.functions = FALSE,
  also.get.flows = NULL,
  post.processing.func = NULL,
  post.processing.companion.kit = NULL,
  agegrp.glue = "",
  CTMC.random.seeds = NULL
)
```

Arguments

file.name	A string
sheets.names	A string vector
differential.eqns.func	A string
just.get.functions	A logical
also.get.flows	A string

`post.processing.func`
: A string

`post.processing.companion.kit`
: A string

`agegrp.glue`
: A string

`CTMC.random.seeds`
: A vector of integers

`smooch.parms.df.into.list` *Put model parameters into a list*

Description

Put model parameters into a list

Usage

```
smooch.parms.df.into.list(df.parms.1d, df.parms.2d, these.are.not.parms)
```

Arguments

`df.parms.1d`
: A data frame

`df.parms.2d`
: A data frame

`these.are.not.parms`
: A data frame

`sweepr_demo` *Prices of 50,000 round cut diamonds.*

Description

A dataset containing the prices and other attributes of almost 54,000 diamonds.

Usage

```
sweepr_demo
```

Format

A data frame with 53940 rows and 10 variables:

price price, in US dollars

carat weight of the diamond, in carats ...

Source

http://www.diamondse.info/

try.various.parms.values *Perform parameter sweeping*

Description

Perform parameter sweeping

Usage

```
try.various.parms.values(
  SEIR.object,
  parm.cloud.grid.specs,
  only.show.parms.to.try = FALSE
)
```

Arguments

SEIR.object	A SEIR model object
parm.cloud.grid.specs	A data frame
only.show.parms.to.try	A logical

verbose.save *Verbose save function*

Description

Verbose save function

Usage

```
verbose.save(
  object.name,
  path.with.trailing.slash = "",
  prefix.suffix = c(prefix = "This file contains an R object called ", suffix =
    ".SavedFromR"),
  time.stamp = gsub(":", "-", Sys.time())
)
```

Arguments

object.name	A string
path.with.trailing.slash	A string
prefix.suffix	A string
time.stamp	A time object

Index