Package ‘simglm’

August 9, 2021

Type Package

Version 0.8.8

Date 2021-08-04

License MIT + file LICENSE

Title Simulate Models Based on the Generalized Linear Model

Description Simulates regression models, including both simple regression and generalized linear mixed models with up to three level of nesting. Power simulations that are flexible allowing the specification of missing data, unbalanced designs, and different random error distributions are built into the package.

Depends R (>= 3.6.0)

Imports stats, methods, Matrix, dplyr, purrr, broom, future.apply

Suggests knitr, lme4, nlme, testthat, shiny, e1071, ggplot2, tidyr, geepack, rmarkdown, future, splines, covr

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.1.1

Author Brandon LeBeau [aut, cre]

Maintainer Brandon LeBeau <lebebr01+simglm@gmail.com>

URL https://github.com/lebebr01/simglm

BugReports https://github.com/lebebr01/simglm/issues

NeedsCompilation no

Repository CRAN

Date/Publication 2021-08-09 18:40:02 UTC
compute_statistics

Description

Compute Power, Type I Error, or Precision Statistics

Usage

```r
compute_statistics(
  data,
  sim_args,
  power = TRUE,
  type_1_error = TRUE,
  precision = TRUE
)
```
correlate_variables

Arguments

data A list of model results generated by `replicate_simulation` function.

sim_args A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:

- fixed: This is the fixed portion of the model (i.e. covariates)
- random: This is the random portion of the model (i.e. random effects)
- error: This is the error (i.e. residual term).

power TRUE/FALSE flag indicating whether power should be computed. Defaults to TRUE.

type_1_error TRUE/FALSE flag indicating whether type I error rate should be computed. Defaults to TRUE.

precision TRUE/FALSE flag indicating whether precision should be computed. Defaults to TRUE.

Description

Correlate elements

Usage

`correlate_variables(data, sim_args, ...)`

Arguments

data Data simulated from other functions to pass to this function.

sim_args A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:

- fixed: This is the fixed portion of the model (i.e. covariates)
- random: This is the random portion of the model (i.e. random effects)
- error: This is the error (i.e. residual term).
- correlate: These are the correlations for random effects and/or fixed effects.

... Additional arguments, currently not used.
### desireVar

**Computes mixture normal variance**

**Description**

Input the desired variance, number of distributions, and mean of the distributions, returns a value of the variance of each mixture distribution.

**Usage**

```r
desireVar(desVar, num_dist, means, equalWeight = TRUE)
```

**Arguments**

- `desVar`: Desired overall variance of mixture normal distribution.
- `num_dist`: Number of normal distributions.
- `means`: Vector of means for each normal distribution. Must equal `num_dist`.
- `equalWeight`: Should equal weights be used, only `TRUE` is currently supported.

**Details**

This function can be used to generate the inputs for the `rbimod` variances when a specific variance is desired. Especially useful when attempting to simulate a mixture normal/bimodal distribution.

### extract_coefficients

**Extract Coefficients**

**Description**

Extract Coefficients

**Usage**

```r
extract_coefficients(model, extract_function = NULL)
```

**Arguments**

- `model`: A returned model object from a fitted model.
- `extract_function`: A function that extracts model results. The function must take the model object as the only argument.
generate_missing  
*Tidy Missing Data Function*

**Description**

Tidy Missing Data Function

**Usage**

```r
generate_missing(data, sim_args)
```

**Arguments**

- `data`  
  Data simulated from other functions to pass to this function.

- `sim_args`  
  A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - `fixed`: This is the fixed portion of the model (i.e. covariates)
  - `random`: This is the random portion of the model (i.e. random effects)
  - `error`: This is the error (i.e. residual term).

---

generate_response  
*Simulate response variable*

**Description**

Simulate response variable

**Usage**

```r
generate_response(data, sim_args, keep_intermediate = TRUE, ...)
```

**Arguments**

- `data`  
  Data simulated from other functions to pass to this function.

- `sim_args`  
  A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - `fixed`: This is the fixed portion of the model (i.e. covariates)
  - `random`: This is the random portion of the model (i.e. random effects)
  - `error`: This is the error (i.e. residual term).

- `keep_intermediate`  
  TRUE/FALSE flag indicating whether intermediate steps should be kept. This would include fixed effects times regression weights, random effect summations, etc. Default is TRUE.

- `...`  
  Other arguments to pass to error simulation functions.
Description

Function that inputs simulated data and returns data frame with new response variable that includes missing data. Missing data types incorporated include dropout missing data, missing at random, and random missing data.

Usage

```r
missing_data(
  sim_data,
  resp_var = "sim_data",
  new_outcome = "sim_data2",
  clust_var = NULL,
  within_id = NULL,
  miss_prop = NULL,
  dropout_location = NULL,
  type = c("dropout", "random", "mar"),
  miss_cov,
  mar_prop
)

dropout_missing(
  sim_data,
  resp_var = "sim_data",
  new_outcome = "sim_data2",
  clust_var = "clustID",
  within_id = "withinID",
  miss_prop = NULL,
  dropout_location = NULL
)

random_missing(
  sim_data,
  resp_var = "sim_data",
  new_outcome = "sim_data2",
  miss_prop,
  clust_var = NULL,
  within_id = "withinID"
)

mar_missing(
  sim_data,
  resp_var = "sim_data",
  new_outcome = "sim_data2",
)```
model_fit

    miss_cov,
    mar_prop

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sim_data</td>
<td>Simulated data frame</td>
</tr>
<tr>
<td>resp_var</td>
<td>Character string of response variable with complete data.</td>
</tr>
<tr>
<td>new_outcome</td>
<td>Character string of new outcome variable name that includes the missing data.</td>
</tr>
<tr>
<td>clust_var</td>
<td>Cluster variable used for the grouping, set to NULL by default which means no</td>
</tr>
<tr>
<td></td>
<td>clustering.</td>
</tr>
<tr>
<td>within_id</td>
<td>ID variable within each cluster.</td>
</tr>
<tr>
<td>miss_prop</td>
<td>Proportion of missing data overall</td>
</tr>
<tr>
<td>dropout_location</td>
<td>A vector the same length as the number of clusters representing the number</td>
</tr>
<tr>
<td></td>
<td>of data observations for each individual.</td>
</tr>
<tr>
<td>type</td>
<td>The type of missing data to generate, currently supports dropout, random, or</td>
</tr>
<tr>
<td></td>
<td>missing at random (mar) missing data.</td>
</tr>
<tr>
<td>miss_cov</td>
<td>Covariate that the missing values are based on.</td>
</tr>
<tr>
<td>mar_prop</td>
<td>Proportion of missing data for each unique value specified in the miss_cov</td>
</tr>
<tr>
<td>argument.</td>
<td></td>
</tr>
</tbody>
</table>

model_fit Tidy Model Fitting Function

Description

Tidy Model Fitting Function

Usage

model_fit(data, sim_args, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>A data object, most likely generated from within</td>
</tr>
<tr>
<td>sim_args</td>
<td>simglm</td>
</tr>
<tr>
<td></td>
<td>A named list with special model formula syntax.</td>
</tr>
<tr>
<td></td>
<td>See details and examples for more information.</td>
</tr>
<tr>
<td></td>
<td>The named list may contain the following:</td>
</tr>
<tr>
<td></td>
<td>• fixed: This is the fixed portion of the model</td>
</tr>
<tr>
<td></td>
<td>(i.e. covariates)</td>
</tr>
<tr>
<td></td>
<td>• random: This is the random portion of the model</td>
</tr>
<tr>
<td></td>
<td>(i.e. random effects)</td>
</tr>
<tr>
<td></td>
<td>• error: This is the error (i.e. residual term).</td>
</tr>
<tr>
<td></td>
<td>• model_fit: These are arguments passed to the</td>
</tr>
<tr>
<td></td>
<td>model_fit function.</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>Currently not used.</td>
</tr>
</tbody>
</table>
parse_correlation  
(Parse correlation arguments)

Description

This function is used to parse user specified correlation attributes. The correlation attributes need to be in a dataframe to be processed internally. Within the dataframe, there are expected to be 3 columns, 1) names of variable/attributes, 2) the variable/attribute pair for 1, 3) the correlation.

Usage

parse_correlation(sim_args)

Arguments

sim_args  A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
- fixed: This is the fixed portion of the model (i.e. covariates)
- random: This is the random portion of the model (i.e. random effects)
- error: This is the error (i.e. residual term).
- correlate: These are the correlations for random effects and/or fixed effects.

parse_crossclass  
(Parse Cross-classified Random Effects)

Description

Parse Cross-classified Random Effects

Usage

parse_crossclass(sim_args, random_formula_parsed)

Arguments

sim_args  Simulation arguments
random_formula_parsed  This is the output from parse_randomeffect.
**parse_formula**

Parses tidy formula simulation syntax

---

**Description**

A function that parses the formula simulation syntax in order to simulate data.

**Usage**

```r
parse_formula(sim_args)
```

**Arguments**

- `sim_args` A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - fixed: This is the fixed portion of the model (i.e. covariates)
  - random: This is the random portion of the model (i.e. random effects)
  - error: This is the error (i.e. residual term).

---

**parse_power**

Parse power specifications

---

**Description**

Parse power specifications

**Usage**

```r
parse_power(sim_args, samp_size)
```

**Arguments**

- `sim_args` A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - fixed: This is the fixed portion of the model (i.e. covariates)
  - random: This is the random portion of the model (i.e. random effects)
  - error: This is the error (i.e. residual term).

- `samp_size` The sample size pulled from the simulation arguments or the power model results when `vary_arguments` is used.
parse_varyarguments

Parse varying arguments

Description
Parse varying arguments

Usage
parse_varyarguments(sim_args)

Arguments
sim_args  A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - fixed: This is the fixed portion of the model (i.e. covariates)
  - random: This is the random portion of the model (i.e. random effects)
  - error: This is the error (i.e. residual term).
rbimod  Simulating mixture normal distributions

Description

Input simulation metrics returns mixture normal random variable.

Usage

rbimod(n, mean, var, num_dist)

Arguments

n  Number of random draws. Optionally can be a vector with number in each simulated normal distribution.
mean  Vector of mean values for each normal distribution. Must be the same length as num_dist.
var  Vector of variance values for each normal distribution. Must be the same length as num_dist.
num_dist  Number of normal distributions to use when simulating mixture normal distribution.

Details

Function to simulate mixture normal distributions. The function computes adds the specified number of normal distributions into a single vector.

Use of the function desireVar can be used to generate a mixture normal distribution with a specific global variance.

replicate_simulation  Replicate Simulation

Description

Replicate Simulation

Usage

replicate_simulation(sim_args, return_list = FALSE, future.seed = TRUE, ...)

Arguments

<table>
<thead>
<tr>
<th>sim_args</th>
<th>A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• fixed: This is the fixed portion of the model (i.e. covariates)</td>
<td></td>
</tr>
<tr>
<td>• random: This is the random portion of the model (i.e. random effects)</td>
<td></td>
</tr>
<tr>
<td>• error: This is the error (i.e. residual term).</td>
<td></td>
</tr>
<tr>
<td>return_list</td>
<td>TRUE/FALSE indicating whether a full list output should be returned. If TRUE, the nested list is returned. If FALSE, replications are combined with a replication id appended.</td>
</tr>
<tr>
<td>future.seed</td>
<td>TRUE/FALSE or numeric. Default value is true, see future_replicate.</td>
</tr>
<tr>
<td>...</td>
<td>Currently not used.</td>
</tr>
</tbody>
</table>

---

run_shiny

*Run Shiny Application Demo*

Description

Function runs Shiny Application Demo

Usage

run_shiny()

Details

This function does not take any arguments and will run the Shiny Application. If running from RStudio, will open the application in the viewer, otherwise will use the default internet browser.

---

simglm

*simglm: A package to simulate and perform power by simulation for models based on the generalized linear model.*

Description

The simglm package provides two categories of important functions: simulation functions and power functions. The package follows a tidy framework where functions are designed to be similar, do one thing, and stack on top of each other to build more complex systems. °’

This function is most useful to pass to replicate_simulation. The function attempts to determine automatically which aspects to add to the simulation/power generation based on the elements found in the sim_args argument.

Usage

simglm(sim_args)
**simulate_error**

**Arguments**
- **sim_args**
  A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - fixed: This is the fixed portion of the model (i.e. covariates)
  - random: This is the random portion of the model (i.e. random effects)
  - error: This is the error (i.e. residual term).

---

**simulate_error**  
*Tidy error simulation*

**Description**

Tidy error simulation

**Usage**

`simulate_error(data, sim_args, ...)`

**Arguments**
- **data**
  Data simulated from other functions to pass to this function.
- **sim_args**
  A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - fixed: This is the fixed portion of the model (i.e. covariates)
  - random: This is the random portion of the model (i.e. random effects)
  - error: This is the error (i.e. residual term).
- **...**
  Other arguments to pass to error simulation functions.

---

**simulate_fixed**  
*Tidy fixed effect formula simulation*

**Description**

This function simulates the fixed portion of the model using a formula syntax.

**Usage**

`simulate_fixed(data, sim_args, ...)`
**simulate_heterogeneity**

*Tidy heterogeneity of variance simulation*

**Description**

This function simulates heterogeneity of level one error variance.

**Usage**

```r
simulate_heterogeneity(data, sim_args, ...)
```

**Arguments**

data  
Data simulated from other functions to pass to this function. Can pass NULL if first in simulation string.

sim_args  
A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:

- fixed: This is the fixed portion of the model (i.e. covariates)
- random: This is the random portion of the model (i.e. random effects)
- error: This is the error (i.e. residual term).

...  
Other arguments to pass to error simulation functions.
simulate_knot  

**Simulate knot locations**

**Description**
Function that generates knot locations. An example of usefulness of this function would be with generation of interrupted time series data. Another application may be with simulation of piecewise linear data structures.

**Usage**
simulate_knot(data, sim_args)

**Arguments**
- **data**: Mostly internal argument.
- **sim_args**: A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - fixed: This is the fixed portion of the model (i.e. covariates)
  - random: This is the random portion of the model (i.e. random effects)
  - error: This is the error (i.e. residual term).

simulate_randomeffect  

**Tidy random effect formula simulation**

**Description**
This function simulates the random portion of the model using a formula syntax.

**Usage**
simulate_randomeffect(data, sim_args, ...)

**Arguments**
- **data**: Data simulated from other functions to pass to this function. Can pass NULL if first in simulation string.
- **sim_args**: A named list with special model formula syntax. See details and examples for more information. The named list may contain the following:
  - fixed: This is the fixed portion of the model (i.e. covariates)
  - random: This is the random portion of the model (i.e. random effects)
  - error: This is the error (i.e. residual term).
- **...**: Other arguments to pass to error simulation functions.
sim_continuous2  
*Simulate continuous variables*

**Description**

Function that simulates continuous variables. Any distribution function in R is supported.

**Usage**

```r
sim_continuous2(
  n, 
  dist = "rnorm", 
  var_level = 1, 
  variance = NULL, 
  ther_sim = FALSE, 
  ther_val = NULL, 
  ...
)
```

**Arguments**

- **n**  
  A list of sample sizes.

- **dist**  
  A distribution function. This argument takes a quoted R distribution function (e.g. `rnorm`). Default is `rnorm`.

- **var_level**  
  The level the variable should be simulated at. This can either be 1, 2, or 3 specifying a level 1, level 2, or level 3 variable respectively.

- **variance**  
  The variance for random effect simulation.

- **ther_sim**  
  A TRUE/FALSE flag indicating whether the error simulation function should be simulated, that is should the mean and standard deviation used for standardization be simulated.

- **ther_val**  
  A vector of 2 that should include the theoretical mean and standard deviation of the generating function.

- **...**  
  Additional parameters to pass to the `dist_fun` argument.

---

sim_factor2  
*Simulate categorical, factor, or discrete variables*

**Description**

Function that simulates discrete, factor, or categorical variables. Is essentially a wrapper around the sample function from base R.
Usage
sim_factor2(n, levels, var_level = 1, replace = TRUE, ...)

Arguments
n A list of sample sizes.
levels Scalar indicating the number of levels for categorical, factor, or discrete variable. Can also specify levels as a character vector.
var_level The level the variable should be simulated at. This can either be 1, 2, or 3 specifying a level 1, level 2, or level 3 variable respectively.
replace TRUE/FALSE indicating whether levels should be sampled with replacement. Default is TRUE.
... Additional parameters passed to the sample function.

Description
This function simulates data for the time variable of longitudinal data.

Usage
sim_time(n, time_levels = NULL, ...)

Arguments
n Sample size of the levels.
time_levels The values the time variable should take. If NULL (default), the time values are discrete integers starting at 0 and going to n - 1.
... Currently not used.

transform_outcome Transform response variable

Description
Transform response variable

Usage
transform_outcome(outcome, type, ...)
Arguments

outcome The outcome variable to transform.
type Type of transformation to apply.
... Additional arguments passed to distribution functions.
Index

compute_statistics, 2
correlate_variables, 3

desireVar, 4,11
dropout_missing (missing_data), 6

extract_coefficients, 4

future_replicate, 12

generate_missing, 5
generate_response, 5

mar_missing (missing_data), 6

missing_data, 6
model_fit, 7,7

parse_correlation, 8

parse_crossclass, 8

parse_formula, 9,10

test_power, 9

test_randomeffect, 8,10

test_varyarguments, 10

random_missing (missing_data), 6

rbimod, 4,11

replicate_simulation, 3,11,12

run_shiny, 12

sim_continuous2, 16

sim_factor2, 16

sim_time, 17

simglm, 12

simulate_error, 13

simulate_fixed, 13

simulate_heterogeneity, 14

simulate_knot, 15

simulate_randomeffect, 15

transform_outcome, 17