Package 'stabiliser'

June 7, 2022

Title Stabilising Variable Selection

Version 1.0.2

Description
A stable approach to variable selection through stability selection and the use of a permutation-based objective stability threshold. Lima et al (2021) <doi:10.1038/s41598-020-79317-8>, Meinshausen and Buhlmann (2010) <doi:10.1111/j.1467-9868.2010.00740.x>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Config/testthat/edition 3

Depends R (>= 3.0.0)

Suggests rmarkdown, testthat (>= 3.0.0), markdown

Imports glmnet, dplyr, bigstep, rsample, tibble, purrr, tidyr, stringr, ggplot2, broom, caret, ncvreg, knitr, Hmisc, expss, lme4, matrixStats, recipes, lmerTest

VignetteBuilder knitr

NeedsCompilation no

Author Robert Hyde [aut, cre] (<https://orcid.org/0000-0002-8705-9405>), Martin Green [aut], Eliana Lima [aut]

Maintainer Robert Hyde <robert.hyde4@nottingham.ac.uk>

Repository CRAN

Date/Publication 2022-06-07 12:00:04 UTC

R topics documented:

`simulate_data` *simulate_data*

Description

Simulate a dataset. This can optionally include variables with a given associated with the outcome.

Usage

```
simulate_data(nrows, ncols, n_true = 0, amplitude = 0)
```

Arguments

`nrows`	The number of rows to simulate.
`ncols`	The number of columns to simulate.
`n_true`	The number of variables truly associated with the outcome.
`amplitude`	The strength of association between true variables and the outcome.

Value

A simulated dataset

`simulate_data_re` *simulate_data_re*

Description

Simulate a 500x500 dataset with 8 true fixed effects, 492 junk variables and a clustered outcome suitable for a 2 level random effects analysis. The strength of association between true variables and the outcome is governed by the error added at level 1 (defined by parameter sd_level_1) and level 2 (sd_level_2).

Arguments

`sd_level_1`	Standard deviation of level 1 variables
`sd_level_2`	Standard deviation of level 2 variables

Value

A simulated dataset with a clustered outcome sutable for random effects analysis

`simulate_selection_bias` *simulate_selection_bias*

Description

An function to illustrate the risk of selection bias in conventional modelling approaches by simulating a dataset with no information and conducting conventional modelling with prefiltration.

Arguments

`nrows`	A vector of the number of rows to simulate (i.e., c(100, 200)).
`ncols`	A vector of the number of columns to simulate (i.e., c(100, 200)).
`p_thresh`	A vector of the p-value threshold to use in univariate pre-filtration (i.e., c(0.1, 0.2)).

Value

A list including a dataframe of results, a dataframe of the median number of variables selected and a plot illustrating false positive selection.

`stabilise` *stabilise*

Description

Function to calculate stability of variables' association with an outcome for a given model over a number of bootstrap repeats

Arguments

`data`	A dataframe containing an outcome variable to be permuted.
`outcome`	The outcome as a string (i.e. "y").
`boot_reps`	The number of bootstrap samples. Default is "auto" which selects number based on dataframe size.
`permutations`	The number of times to be permuted per repeat. Default is "auto" which selects number based on dataframe size.
`perm_boot_reps`	The number of times to repeat each set of permutations. Default is 20.
`models`	The models to select for stabilising. Default is elastic net (models = c("enet")), other available models include "lasso", "mbic", "mcp".
`type`	The type of model, either "linear" or "logistic"
`quantile`	The quantile of null stabilities to use as a threshold.
`normalise`	Normalise numeric variables (TRUE/FALSE)
`dummy`	Create dummy variables for factors/characters (TRUE/FALSE)
`impute`	Impute missing data (TRUE/FALSE)

Value

A list for each model selected. Each list contains a dataframe of variable stabilities, a numeric permutation threshold, and a dataframe of coefficients for both bootstrap and permutation.

`stabiliser_example` *stabiliser_example*

Description

A simulated dataset

Usage

```
stabiliser_example
```

Format

A data frame with 50 rows and 100 variables.

The stabiliser_example dataset is a simulated example with the following properties: 1 simulated outcome variable: `y` 4 variables simulated to be associated with `y`: `causal1`, `causal2`... 95 variables simulated to have no association with `y`: `junk1`, `junk2`...

`stabilise_re` *stabilise_re*

Description

Function to calculate stability of variables' association with an outcome for a given model over a number of bootstrap repeats using clustered data.

Arguments

`data`	A dataframe containing an outcome variable to be permuted.
`outcome`	The outcome as a string (i.e. "y").
`level_2_id`	The variable name determining level 2 status as a string (i.e., "level_2_column_name").
`n_top_filter`	The number of variables to filter for final model (Default = 50).
`boot_reps`	The number of bootstrap samples. Default is "auto" which selects number based on dataframe size.
`permutations`	The number of times to be permuted per repeat. Default is "auto" which selects number based on dataframe size.
`perm_boot_reps`	The number of times to repeat each set of permutations. Default is 20.
`normalise`	Normalise numeric variables (TRUE/FALSE)
`dummy`	Create dummy variables for factors/characters (TRUE/FALSE)
`impute`	Impute missing data (TRUE/FALSE)

Value

A list containing a table of variable stabilities and a numeric permutation threshold.

`stab_plot` *stab_plot*

Description

Plot from stability object

Arguments

`stabiliser_outcome`
: Outcome from stabilise() or triangulate() function.

Value

A ggplot object.

`triangulate` *triangulate*

Description

Triangulate multiple models using a stability object

Arguments

`object`	An object generated through the stabilise() function.
`quantile`	The quantile of null stabilities to use as a threshold.

Value

A combined list of model results including a dataframe of stability results for variables and a numeric permutation threshold.

Index