Package 'grrrr'

June 6, 2023

Type Package

Title Set of grouped regression methods

Version 1.0

Date 2023-06-02

Author Bartosz Rozek, Jakub Szypula

Description This package allows user to create grouped regression models such as grouped lasso and grouped lars. User also can test them with a given framework which can be extended to handle additionalmethods.

RoxygenNote 7.2.3

Imports ggplot2, purrr, pracma, gglasso

Encoding UTF-8

R topics documented:

+,test_container,test_results-method 2
calculate_cp 3
calculate_me 3
calculate_test 4
calc_group_lars 4
calc_group_lasso 5
categorize_matrix 5
count_factors 6
create_boxplot,test_container,character-method 6
create_model1 7
create_model2 7
create_model3 8
create_model4 8
create_table,test_container-method 9
df_lars 9
df_lasso 10
find_alpha_lars 10
first_up 11
generate_noise 11

get_test,test_container,character-method 12
group_lars-class 12
group_lasso-class 13
norm_L 14
perform_ttest,test_container,character,character-method 14
quad_roots 15
test_container-class 15
test_lars 16
test_lars_group 16
test_lasso_group 17
test_lasso_group_library 17
test_ls 18
test_result-class 18
test_results-class 19
test_step 19
%-% 20

Index **21**

`+,test_container,test_results-method`
Adding test_results to container

Description

Adding test_results to container

Usage

```
## S4 method for signature 'test_container,test_results'
e1 + e2
```

Arguments

`e1`	test_container. Instance of class test_container
`e2`	test_results. Instance of class test_results

Value

instance of class test_container with added new test_results.

calculate_cp *Calculation of Cp value*

Description

Calculation of Cp value

Usage

```
calculate_cp(indexes, group_sizes, betas, betas_ls, X, y, df_function)
```

Arguments

indexes	array with factors chosen in the model
group_sizes	array with sizes of consecutive groups
betas	beta coefficients in the investigated model
betas_ls	beta coefficients in the OLS model build on the same data
X	matrix of regressors
y	target variable
df_function	function that calculates degrees of freedom for specific model

Value

value of Cp statistic

calculate_me *Calculation of model error value*

Description

Calculation of model error value

Usage

```
calculate_me(X, beta_hat, beta)
```

Arguments

X	matrix of regressors
beta_hat	beta coefficients in the investigated model
beta	original beta coefficients used to generate data set

Value

value of model error

calculate_test *Function used to test specific method*

Description

Function used to test specific method

Usage

```
calculate_test(name, test_function, n, create_model, ...)
```

Arguments

name	string name which will be visible in the results
test_function	function with (X, y, true_betas, groups) as input and returning test_result object
n	how many times the test will be run
create_model	function that returns data set (X, y, groups)
...	additional arguments for test_function

Value

test_results object

calc_group_lars *Creates a instance of group lars model*

Description

Creates a instance of group lars model

Usage

```
calc_group_lars(X, y, groups, result_indicator = "cp", true_betas = NULL)
```

Arguments

X	matrix with regressors
y	target variable
groups	list of integers with a length equals to number of columns in X. Indicates to which group given variable belongs to
result_indicator	one of values ("cp", "me"). Indicates which of those two statistic should be used to select the final model. To use "me" also true_betas needs to be supplied.
true_betas	array of true values of betas

Value

object of class group_lars

`calc_group_lasso` *Creates a instance of group lasso model*

Description

Creates a instance of group lasso model

Usage

```
calc_group_lasso(X, y, groups, result_indicator = "cp", true_betas = NULL)
```

Arguments

`X`	matrix with regressors
`y`	target variable
`groups`	list of integers with a length equals to number of columns in X. Indicates to which group given variable belongs to
`result_indicator`	one of values ("cp", "me"). Indicates which of those two statistic should be used to select the final model. To use "me" also true_betas needs to be supplied.
`true_betas`	array of true values of betas

Value

object of class group_lasso

`categorize_matrix` *Trichotomization of values in the matrix*

Description

Trichotomization of values in the matrix

Usage

```
categorize_matrix(Z)
```

Arguments

`Z`	matrix

Value

matrix with trichomizated values

count_factors *Function used to calculate unique factors*

Description

Function used to calculate unique factors

Usage

```
count_factors(betas, betas_names)
```

Arguments

betas	values of coefficients
betas_names	names of the coefficients

Value

number of unique factors

create_boxplot,test_container,character-method *Creates boxplot for test_container.*

Description

Creates boxplot for test_container.

Usage

```
## S4 method for signature 'test_container,character'
create_boxplot(container, column)
```

Arguments

container	test_container. Instance of test_container_class
column	character. Column which values will be presented in the boxplot. One of the ("model_error", "n_factors", "cpu_time")

Value

ggplot2 object with boxplot

create_model1 *Creation of type 1 data set*

Description

Creation of type 1 data set

Usage

```
create_model1(n = 50, p = 15)
```

Arguments

n	number of observations
p	number of variables

Value

list with three elements - X: design matrix, y: target variable, betas: coefficients used to create y

create_model2 *Creation of type 2 data set*

Description

Creation of type 2 data set

Usage

```
create_model2(n = 100, p = 4)
```

Arguments

n	number of observations
p	number of variables

Value

list with three elements - X: design matrix, y: target variable, betas: coefficients used to create y

create_model3 *Creation of type 3 data set*

Description

Creation of type 3 data set

Usage

```
create_model3(n = 100, p = 16)
```

Arguments

n	number of observations
p	number of variables

Value

list with three elements - X: design matrix, y: target variable, betas: coefficients used to create y

create_model4 *Creation of type 4 data set*

Description

Creation of type 4 data set

Usage

```
create_model4(n = 100, p1 = 10, p2 = 10)
```

Arguments

n	number of observations
p1	number of discrete variables

Value

list with three elements - X: design matrix, y: target variable, betas: coefficients used to create y

create_table,test_container-method *Creates table with aggregated results of the tests*

Description

Creates table with aggregated results of the tests

Usage

```
## S4 method for signature 'test_container'
create_table(container)
```

Arguments

container	test_container. Instance of class test_container

Value

data frame with results of all tests. This table's shape is based on the results in the article.

df_lars *Calculates degrees of freedom for group lars model.*

Description

Calculates degrees of freedom for group lars model.

Usage

```
df_lars(indexes, group_sizes, betas, betas_ls)
```

Arguments

indexes	array with factors chosen in the model
group_sizes	array with sizes of consecutive groups
betas	beta coefficients in the investigated model
betas_ls	beta coefficients in the OLS model build on the same data

Value

number of degrees of freedom

df_lasso — *Calculates degrees of freedom for group lasso model.*

Description

Calculates degrees of freedom for group lasso model.

Usage

```
df_lasso(indexes, group_sizes, betas, betas_ls)
```

Arguments

indexes	array with factors chosen in the model
group_sizes	array with sizes of consecutive groups
betas	beta coefficients in the investigated model
betas_ls	beta coefficients in the OLS model build on the same data

Value

number of degrees of freedom

find_alpha_lars — *Finds optimum for quadratic equation needed to find next factor included in the LARS algorithm.*

Description

Finds optimum for quadratic equation needed to find next factor included in the LARS algorithm.

Usage

```
find_alpha_lars(X, r, j, mcs, gamma_)
```

Arguments

X	matrix with regressors
r	current residuals
j	candidate factor
mcs	current "active set"
gamma_	current direction

Value

value of root which is in [0,1] interval

first_up *Makes first letter of string uppercase*

Description

Makes first letter of string uppercase

Usage

```
first_up(x)
```

Arguments

x	string

Value

transformed string

generate_noise *Generating noise for target variable*

Description

Generating noise for target variable

Usage

```
generate_noise(Y, ratio)
```

Arguments

Y	array
ratio	signal-to-noise-ratio

Value

array with noise

get_test,test_container,character-method *Test getter*

Description

Test getter

Usage

```
## S4 method for signature 'test_container,character'
get_test(container, name)
```

Arguments

container	test_container. Instance of test_container class
name	character. Name of the test to be returned

Value

instance of class test_results from the container. If there is no test with such a name method will throw an error.

group_lars-class *Class storing information about group lasso model*

Description

Class storing information about group lasso model

Value

instance of group_lars class

Slots

X matrix. Design matrix

y numeric. Target variable

betas numeric. Final beta coefficients

betas_path list. List of all beta coefficients obtain during calculations

true_betas numericOrNULL. Beta coefficients used in target variable calculations

Cp numeric. Value of Cp

Cp_path list. List of values of Cp obtained during calculations

`model_error` numericOrNULL. Value of model_error for final model. Not null only if true_betas was supplied.

`me_path` listOrNULL. List of values of model_error obtained during calculations. Not null only if true_betas was supplied.

`group_lasso-class` *Class storing information about group lasso model*

Description

Class storing information about group lasso model

Value

instance of group_lasso class

Slots

`X` matrix. Design matrix

`y` numeric. Target variable

`betas` numeric. Final beta coefficients

`betas_path` list. List of all beta coefficients obtain during calculations

`true_betas` numericOrNULL. Beta coefficients used in target variable calculations

`lambda_max` numeric. Maximum value of lambda

`lambda_best` numeric. Value of lambda used for final model

`Cp` numeric. Value of Cp

`Cp_path` list. List of values of Cp obtained during calculations

`model_error` numericOrNULL. Value of model_error for final model. Not null only if true_betas was supplied.

`me_path` listOrNULL. List of values of model_error obtained during calculations. Not null only if true_betas was supplied.

norm_L *Vector norm mentioned in the article*

Description

Vector norm mentioned in the article

Usage

```
norm_L(vector, p)
```

Arguments

vector	array
p	multiplier of identity matrix

Value

vector norm

perform_ttest,test_container,character,character-method *Performs check if results of the models are statistically different*

Description

Performs check if results of the models are statistically different

Usage

```
## S4 method for signature 'test_container,character,character'
perform_ttest(container, tests_rows, tests_cols)
```

Arguments

container	test_container. Instance of test_container class
tests_rows	character. One group of tests (may be an array). Will be presented in the rows
tests_cols	character. Second group of tests (may be an array). Will be presented in the columns

Value

table with p-values of t-test.

quad_roots *Very simple quadratic equation solver*

Description

Very simple quadratic equation solver

Usage

```
quad_roots(a, b, c)
```

Arguments

a	quadratic coefficient
b	linear coefficient
c	constant coefficient

Value

array with two roots

test_container-class *Object that stores instances of tests_results*

Description

Object that stores instances of tests_results

Value

object of class test_container

Slots

`tests` list. List of tests_results instances

test_lars — *Function that test lars method without grouping*

Description

Function that test lars method without grouping

Usage

```
test_lars(X, y, true_betas, groups, ...)
```

Arguments

X	design matrix
y	target variable
true_betas	beta coefficients used in Y calculation
groups	list of integers with a length equals to number of columns in X. Indicates to which group given variable belongs to

Value

instance of test_result class

test_lars_group — *Function that test group lars method*

Description

Function that test group lars method

Usage

```
test_lars_group(X, y, true_betas, groups, ...)
```

Arguments

X	design matrix
y	target variable
true_betas	beta coefficients used in Y calculation
groups	list of integers with a length equals to number of columns in X. Indicates to which group given variable belongs to

Value

instance of test_result class

test_lasso_group *Function that test group lasso method*

Description

Function that test group lasso method

Usage

```
test_lasso_group(X, y, true_betas, groups, ...)
```

Arguments

X	design matrix
y	target variable
true_betas	beta coefficients used in Y calculation
groups	list of integers with a length equals to number of columns in X. Indicates to which group given variable belongs to

Value

instance of test_result class

test_lasso_group_library *Function that test group lasso method from external package*

Description

Function that test group lasso method from external package

Usage

```
test_lasso_group_library(X, y, true_betas, groups)
```

Arguments

X	design matrix
y	target variable
true_betas	beta coefficients used in Y calculation
groups	list of integers with a length equals to number of columns in X. Indicates to which group given variable belongs to

Value

instance of test_result class

test_ls *Function that test OLS method*

Description

Function that test OLS method

Usage

```
test_ls(X, y, true_betas, groups)
```

Arguments

X	design matrix
y	target variable
true_betas	beta coefficients used in Y calculation
groups	added only to keep function's shape

Value

instance of test_result class

test_result-class *Title*

Description

Title

Value

instance of test_result class

Slots

model_error numeric. Model error obtained in the test

n_factors numeric. Number of factors obtained in the test

cpu_time numeric. CPU time obtained in the test

test_results-class *Class containing information from multiple tests runs*

Description

Class containing information from multiple tests runs

Value

instance of test_results class

Slots

`name` character. Name of the model
`model_error` numeric. Mean model error
`model_error_list` numeric. All model errors obtained during testing
`model_error_std` numeric. Standard deviation of model error
`n_factors` numeric. Mean number of factors
`n_factors_list` integer. All numbers of factors obtained during testing
`n_factors_std` numeric. Standard deviation of number of factors
`cpu_time` numeric. Mean CPU time
`cpu_time_list` numeric. All CPU times obtained during testing
`cpu_time_std` numeric. Standard deviation of CPU time

test_step *Function that test stepwise regression method*

Description

Function that test stepwise regression method

Usage

```
test_step(X, y, true_betas, groups)
```

Arguments

`X`	design matrix
`y`	target variable
`true_betas`	beta coefficients used in Y calculation
`groups`	added only to keep function's shape

Value

instance of test_result class

`%-%` *A easier form of setting part of array to zero*

Description

Implementation of b_-j=(b^’_1, ..., b^’_j-1, 0’, b^’_j+1, ..., b^’_J)

Usage

```
vector %-% index
```

Arguments

`vector`	array which is meant to be used
`index`	indexes were zeros will be inserted

Value

array with zeros in selected indexes

Index

+,test_container,test_results-method, 2
%-%, 20

calc_group_lars, 4
calc_group_lasso, 5
calculate_cp, 3
calculate_me, 3
calculate_test, 4
categorize_matrix, 5
count_factors, 6
create_boxplot,test_container,character-method, 6
create_model1, 7
create_model2, 7
create_model3, 8
create_model4, 8
create_table,test_container-method, 9

df_lars, 9
df_lasso, 10

find_alpha_lars, 10
first_up, 11

generate_noise, 11
get_test,test_container,character-method, 12
group_lars-class, 12
group_lasso-class, 13

norm_L, 14

perform_ttest,test_container,character,character-method, 14

quad_roots, 15

test_container-class, 15
test_lars, 16
test_lars_group, 16
test_lasso_group, 17
test_lasso_group_library, 17
test_ls, 18
test_result-class, 18
test_results-class, 19
test_step, 19