

Package ‘dfConn’

June 13, 2019

Type Package

Title Dynamic Functional Connectivity Analysis

Version 0.2.1

Maintainer Zikai Lin <ziklin@iu.edu>

Description

An implementation of multivariate linear process bootstrap (MLPB) method and sliding window technique to assess the dynamic functional connectivity (dFC) estimate by providing its confidence bands, based on Maria Kudela (2017) <doi: 10.1016/j.neuroimage.2017.01.056>.

It also integrates features to visualize non-zero coverage for selected a-priori regions of interest estimated by the dynamic functional connectivity model (dFCM) and dynamic functional connectivity (dFC) curves for reward-related a-priori regions of interest where the activation-based analysis reported.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 2.10)

Suggests iterators, testthat, itertools, mgcv, latex2exp

Imports doParallel, nlme, parallel, foreach, ggplot2, fields, gplots, splines, stats, stringr, graphics, data.table, gtools, Rcpp (>= 0.12.18)

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 6.1.1

NeedsCompilation yes

Author Zikai Lin [aut, cre],
Maria Kudela [aut],
Jaroslaw Harezlak [aut],
Mario Dziedzic [aut]

Repository CRAN

Date/Publication 2019-06-13 19:30:03 UTC

R topics documented:

coverage.tbl.list	2
data_summary_coverage	3
DynModel_results	4
fMRI_dataList	5
fMRI_dataList_shrunked	5
ImmConn	6
ImmDyn	7
MLPB3	9
MLPB_boot	10
MLPB_output_median	11
nzc_vis	11
options_boot	12
options_lme	13
plot_generate_timeDependence	14
Index	16

coverage.tbl.list	<i>Sample dynamic functional connectivity estimates summary tables</i>
-------------------	--

Description

Contains a list of data frames with comparisons and corresponding significant non-zero estimates coverage.

Usage

```
data(coverage.tbl.list)
```

Format

A list which contains 9 data frames:

coverage.zero.cond.diff Percentage of estimates for condition difference that are within 95% confidence interval with a center of zero.

coverage.zero.cond.1 Percentage of estimates for condition 1 that are within 95% confidence interval with a center of zero.

coverage.zero.cond.2 Percentage of estimates for condition 2 that are below 95% confidence interval with a center of zero.

coverage.below.zero.cond.2 Percentage of estimates for condition 2 that are below 95% confidence interval with a center of zero.

coverage.below.zero.cond.diff Percentage of estimates for condition difference that are below 95% confidence interval with a center of zero.

coverage.below.zero.cond.1 Percentage of estimates for condition 1 that are below 95% confidence interval with a center of zero.

coverage.above.zero.cond.diff Percentage of estimates for condition difference that are above 95% confidence interval with a center of zero.

coverage.above.zero.cond.1 Percentage of estimates for condition 1 that are above 95% confidence interval with a center of zero.

coverage.above.zero.cond.2 Percentage of estimates for condition 1 that are above 95% confidence interval with a center of zero.

data_summary_coverage *Obtain the non-zero coverage for both two conditions and the condition-based difference*

Description

Summarize the data from dynamic functional connectivity linear mixed modeling.

Usage

```
data_summary_coverage(dataList.CI, save_res = FALSE, output_dir = NULL)
```

Arguments

dataList.CI	list, a list of data matrices which include the confidence interval estimates obtained from linear mixed models.
save_res	logical, whether to save the result or not, if true, an output directory should be provided
output_dir	character, directory for output files

Value

A list which contains 9 data frames:

coverage.zero.cond.diff Percentage of estimates for condition difference that are within 95% confidence interval with a center of zero.

coverage.zero.cond.1 Percentage of estimates for condition 1 that are within 95% confidence interval with a center of zero.

coverage.zero.cond.2 Percentage of estimates for condition 2 that are below 95% confidence interval with a center of zero.

coverage.below.zero.cond.2 Percentage of estimates for condition 2 that are below 95% confidence interval with a center of zero.

coverage.below.zero.cond.diff Percentage of estimates for condition difference that are below 95% confidence interval with a center of zero.

coverage.below.zero.cond.1 Percentage of estimates for condition 1 that are below 95% confidence interval with a center of zero.

coverage.above.zero.cond.diff Percentage of estimates for condition difference that are above 95% confidence interval with a center of zero.

coverage.above.zero.cond.1 Percentage of estimates for condition 1 that are above 95% confidence interval with a center of zero.

coverage.above.zero.cond.2 Percentage of estimates for condition 1 that are above 95% confidence interval with a center of zero.

Examples

```
# Summarize the output of linear mixed effect model for dynamic functional connectivity
data(DynModel_results)
data_summary_coverage(DynModel_results$est_CI)
```

DynModel_results	<i>Sample confidence interval output of dynamic functional connectivity (dFC) model</i>
------------------	---

Description

A sample confidence interval estimates (between pairwise regions of interest 1 and 2) coming from dynamic functional connectivity model

Usage

```
data(DynModel_results)
```

Format

A list with 5 objects:

params Store the parameters for the linear mixed model

parallel Store the information about parallel computing environment and parameters

output_by_row Store the model results information for data between each pair of regions

modelDyn_results A combined dataframe with all the model results informations

est_CI A list of matrices, contains information of confidence band estimate, see description below.

In each matrix of est_CI:

row 1 Base line estimate, difference between condition 1 and condition 2

row 2 Dynamic functional connecvity estimates for condition 1

row 3 Dynamic functional connecvity estimates for condition 2

row 4 Lower bound of 95% confidence interval of condition-difference estimates

row 5 Upper bound of 95% confidence interval of condition-difference estimates

row 6 Lower bound of 95% confidence interval of condition-1 estimates

row 7 Upper bound of 95% confidence interval of condition-1 estimates

row 8 Lower bound of 95% confidence interval of condition-2 estimates

row 9 Upper bound of 95% confidence interval of condition-2 estimates

fMRI_dataList	<i>Sample fMRI time series data matrices</i>
---------------	--

Description

Sample functional magnetic resonance imaging (fMRI) data contains a list of data matrices with consisting time series data of two subjects, with 6 scans for each subject in total.

Usage

```
data(fMRI_dataList)
```

Format

A list consisting of two matrices, each with 278 rows and 750 columns:

subject1 sample time series data matrix for subject 1.

subject2 sample time series data matrix for subject 2.

For each data matrix in the list:

row index of regions of interest .

column fMRI time series data at each time point.

fMRI_dataList_shrunked	<i>A shrunked sample subject functional magnetic resonance imaging (fMRI) time series data</i>
------------------------	--

Description

Contains a list of data matrices with consisting time series data of only one subject, and only 1 scan is preserved. This shrunked dataset is useful for user to test the programs and functions.

Usage

```
data(fMRI_dataList_shrunked)
```

Format

A list which contains a matrix with 2 rows and 19 columns:

row index of regions of interest (ROIs).

column fMRI time series data at each time point.

ImmConn

*Semiparametric modelling - Static***Description**

Linear mixed effect modeling for static functional connectivity

Usage

```
ImmConn(dataList, op_lme)
```

Arguments

`dataList` list, list of data matrices .
`op_lme` list, options constructed by function `options_lme`, see `options_lme`.

Details

ImmDyn and ImmConn, we need to summarize the bootstrapping by taking the mean or median of the dFC estimate at each time point.

Value

An object of list containing all the information of the static functional connectivity linear mixed model. It stores the information of model parameters, input data, and model results.

params Store the parameters for the linear mixed model

parallel Store the information about parallel computing environment and parameters

output_by_row Store the model results information for data between each pair of regions

modelDyn_results A combined dataframe with all the model results informations

est_CI A list of matrices, contains information of confidence band estimate, see description below.

In each matrix of `est_CI`:

row 1 Base line estimate, difference between condition 1 and condition 2

row 2 Dynamic functional connecvity estimates for condition 1

row 3 Dynamic functional connecvity estimates for condition 2

row 4 Lower bound of 95% confidence interval of condition-difference estimates

row 5 Upper bound of 95% confidence interval of condition-difference estimates

row 6 Lower bound of 95% confidence interval of condition-1 estimates

row 7 Upper bound of 95% confidence interval of condition-1 estimates

row 8 Lower bound of 95% confidence interval of condition-2 estimates

row 9 Upper bound of 95% confidence interval of condition-2 estimates

Examples

```
# Assuming user has run MLPB_boot() and has summarized the bootstrapping results
# by calculating the median or mean.

## Example with 5 subjects bootstrap-based static functional connectivity estimates data included
## Each subject's data has 731 time points in total, which includes 6 scans and 105 effective
## time point for each scan.
## 3 regions of interest (ROIs) comparison pairs are selected

data(MLPB_output_median)
subjects <- c('subject1', 'subject2', 'subject3', 'subject4', 'subject5')

# In our demo data, each subject has a scan with a total of 750 time points
time.points <- c(1:105, 126:230, 251:355,
                376:480, 501:605, 626:730)

num.scan <- 6 # Each subject has 6 scans
ntps.per.scan <- 105 # Each scan has 105 time points

op <- options_lme(effective_tp = time.points,
                 ntps.per.scan = ntps.per.scan,
                 subjects = subjects,
                 num.scan = num.scan)

resConn <- ImmConn(MLPB_output_median, op)
```

 ImmDyn

Semiparametric modelling - dynamic functional connectivity model

Description

Linear mixed effect model for dynamic functional connectivity.

Usage

```
ImmDyn(dataList, op_lme)
```

Arguments

dataList	list, a list of data matrices.
op_lme	list, options constructed by function options_lme, see options_lme.

Details

The output of MLPB_boot is a complete bootstrapping result for the dynamic functional connectivity (dFC) estimates of each pair of region of interest, however, for further analysis, like ImmDyn and ImmConn, we need to summarize the bootstrapping by taking the mean or median of the dFC estimate at each time point.

Value

An object of list containing all the information of the static functional connectivity linear mixed model. It stores the information of model parameters, input data, and model results.

params Store the parameters for the linear mixed model

parallel Store the information about parallel computing environment and parameters

output_by_row Store the model results information for data between each pair of regions

modelDyn_results A combined dataframe with all the model results informations

est_CI A list of matrices, contains information of confidence band estimate, see description below.

In each matrix of est_CI:

row 1 Base line estimate, difference between condition 1 and condition 2

row 2 Dynamic functional connectivity estimates for condition 1

row 3 Dynamic functional connectivity estimates for condition 2

row 4 Lower bound of 95% confidence interval of condition-difference estimates

row 5 Upper bound of 95% confidence interval of condition-difference estimates

row 6 Lower bound of 95% confidence interval of condition-1 estimates

row 7 Upper bound of 95% confidence interval of condition-1 estimates

row 8 Lower bound of 95% confidence interval of condition-2 estimates

row 9 Upper bound of 95% confidence interval of condition-2 estimates

Examples

```
# Assuming user has run MLPB_boot() and has summarize the bootstrapping result

data("MLPB_output_median")

subjects <- c('subject1', 'subject2', 'subject3', 'subject4', 'subject5')

# In our demo data, each subject has a scan with a total of 750 time points
time.points <- c(1:105, 126:230, 251:355,
                376:480, 501:605, 626:730)

num.scan <- 6 # Each subject has 6 scans
ntps.per.scan <- 105 # Each scan has 105 time points

op <- options_lme(effective_tp = time.points,
                 ntps.per.scan = ntps.per.scan,
                 subjects = subjects,
                 num.scan = num.scan,
                 cores = 5)

resDyn <- ImmDyn(MLPB_output_median, op)
```

MLPB3

Multivariate Linear Bootstrapping - core function

Description

Multivariate Linear Bootstrapping - core function

Usage

```
MLPB3(X, Boot, l = 1, eps = 1, beta = 1, l_automatic = 0,  
      l_automatic_local = 0)
```

Arguments

X	data matrix
Boot	integer, number of bootstrap samples to be generated
l	numeric, the banding parameter, default is 1
eps	numeric the parameters for making Gamma_kappa_matrix positive definite if necessary, default is 1;
beta	numeric the parameters for making Gamma_kappa_matrix positive definite if necessary, default is 1;
l_automatic	numeric the banding parameter, default is 1, data-adaptively;
l_automatic_local	numeric, the banding parameter default is 0

Value

a numeric matrix with $n \cdot \text{boot}$ rows and m columns, where n , m refer to the number of rows and number of columns in input data matrix, and boot refer to number of bootstrap samples to be generated.

Examples

```
# Multivariate linear bootstrapping on a random matrix with 2 rows and 4 columns  
MLPB3(matrix(rnorm(16),2,4), 3)
```

MLPB_boot

Multivariate Linear Process Bootstrap Method

Description

Multivariate Linear Process Bootstrap (MLPB) method to assess the uncertainty in dynamic functional connectivity (dFC) by providing its confidence band.

Usage

```
MLPB_boot(boot_options)
```

Arguments

`boot_options` a list of option being specify by `option_control` function.

Details

The `dataList` parameter is a list of matrices which contains time series data of each region of interest (ROI). Output directory is required here because the results to be generate is massive.

References

Kudela et al. (2017) NeuroImage 10.1016/j.neuroimage.2017.01.056 ([PubMed](#))

Examples

```
# Load sample data

data(fMRI_dataList)
boot_option <- options_boot(input_list = fMRI_dataList, output_dir = tempdir(),
  rois = c(1,2,3), timepoints = 5, window_size = 5)
MLPB_boot(boot_option)
```

MLPB_output_median	<i>Sample median output of Multivariate Linear Process Bootstrap (MLPB)</i>
--------------------	---

Description

A dataset containing summarized by calculating the median for dynamic functional connectivity estimates of pairwise region of interest (ROI) comparisons.

Usage

```
data(MLPB_output_median)
```

Format

A list with 3 matrices, each matrix has 5 rows and 731 columns, where rows represent subjects-level multivariate linear process bootstrapping (MLPB) estimates and columns represent median of MLPB estimates at each window.

median_1_2 matrix with 5 rows and 731 columns, containing median of multivariate linear process bootstrapping results between region 1 and 2 in subject-level.

median_1_3 matrix with 5 rows and 731 columns, containing median of multivariate linear process bootstrapping results between region 1 and 3 in subject-level.

median_2_3 matrix with 5 rows and 731 columns, containing median of multivariate linear process bootstrapping results between region 2 and 3 in subject-level.

nzc_vis	<i>Non-zero coverage correlation matrix visualization</i>
---------	---

Description

Visualize non-zero coverage of dynamic functional connectivity modeling estimates.

Usage

```
nzc_vis(coverage.tbl.list, model_results, roi, ntps.per.scan,
  save_fig = FALSE, output_dir = NULL, alpha = 0.05,
  thres.cond1 = 0.5, thres.cond2 = 0.5, thres.diff = 0.1)
```

Arguments

coverage.tbl.list	A list with a list of zero-coverage table, can be obtained from data_summary_coverage.
model_results	data frame, output of either lmmConn or lmmDyn.
roi	integer vector, specify the list of regions of interests.
ntps.per.scan	integer, number of timepoints per scan.
save_fig	logical, whether to save the figures locally.
output_dir	character string, output directory of figures to save.
alpha	alpha level of proportion-testing, default is 0.05.
thres.cond1	numeric, threshold for cond1 flavor proportion-test, default is 0.5.
thres.cond2	numeric, threshold for cond2 flavor proportion-test, default is 0.5.
thres.diff	numeric, threshold for proportion-test of difference between condition 1 and condition 2 .

Examples

```
# load sample summary of the linear mixed model output
data(coverage.tbl.list)
data(DynModel_results)
nzc_vis(coverage.tbl.list, DynModel_results$modelDyn_results,
        roi=c(54,191,235), DynModel_results$params$ntp_per_scan, save_fig=FALSE)
```

options_boot

Options control - Multivariate linear process bootstrap

Description

Options control for running multivariate linear Multivariate Linear Process Bootstrap (MLPB) method.

Usage

```
options_boot(input_list, output_dir = NULL, rois = NULL,
             subset = NULL, timepoints = NULL, window_size = 20,
             number_of_intervals = 1, boot_rep = 250, n_boot = 1, cores = 1,
             save_file_suffix = "")
```

Arguments

input_list,	list of matrices, subject-specific time series data.
output_dir	character string, output directory for bootstrap samples.
rois	vector of integers, specify the list of regions of interests

subset	vector of character, which subject to run, specify the index in dataList, for example, c(1, 2, ...)
timepoints	integer, number of timepoints in total
window_size	integer, window size for sliding window technique
number_of_intervals	integer, number of intervals in sliding window technique
boot_rep	integer, bootstrapping repetition times
n_boot	integer, number of bootstrap sample to be generated in MLPB
cores	integer, number of cores to register for parallel execution, if set to 1 a sequential job will be run
save_file_suffix	character, suffix of output files, treated as a labels.

Value

A list of options

Examples

```
# Summarize the output of linear mixed effect model for dynamic functional connectivity
data(fMRI_dataList)
op <- options_boot(input_list = fMRI_dataList,
                   output_dir = NULL,
                   rois = c(1,2,3),
                   window_size = 10,
                   save_file_suffix = "",
                   number_of_interval = 1,
                   boot_rep = 250,
                   n_boot = 1,
                   cores = 1)
```

options_lme

Options control - Linear mixed model

Description

Options control for running linear mixed model.

Usage

```
options_lme(effective_tp, num.scan, ntps.per.scan, output_dir = NULL,
            subjects = NULL, ci_level = 0.975, numIntKnots = 40, cores = 1,
            ngrid = 201, seed = 1114)
```

Arguments

effective_tp	integer, effective scan time points.
num.scan	integer, number of scan.
ntps.per.scan	integer, number of timepoints per scan.
output_dir	string, directory for storing output files.
subjects	character vector, names of subjects.
ci_level	numeric, level of confidence interval, with default 0.975.
numIntKnots	integer, number of knots for O'Sullivan penalized splines, default is 40
cores	integer, number of cores to register while running parallel jobs.
ngrid	integer, number of grids.
seed	integer, random seed.

Value

A list of options

Examples

```
data("MLPB_output_median")

subjects <- c('subject1', 'subject2', 'subject3', 'subject4', 'subject5')

# In our demo data, each subject has a scan with a total of 750 time points
time.points <- c(1:105, 126:230, 251:355,
                376:480, 501:605, 626:730)

num.scan <- 6 # Each subject has 6 scans
ntps.per.scan <- 105 # Each scan has 105 time points

op <- options_lme(effective_tp = time.points,
                 ntps.per.scan = ntps.per.scan,
                 subjects = subjects,
                 num.scan = num.scan,
                 cores = 5)
```

plot_generate_timeDependence

95% Coverage time dependence plots for dFCM

Description

Visualize the dFC time dependence modeled in dynamic functional connectivity modelling along with its confidence band.

Usage

```
plot_generate_timeDependence(dataList.CI, scan.length, rois,  
  save_fig = FALSE, output_dir = NULL, cores = 0)
```

Arguments

<code>dataList.CI</code>	The path of confidence interval data
<code>scan.length</code>	integer, usually set as number of total timepoints per scan
<code>rois</code>	Numerical array, regions of interests to generate plots
<code>save_fig</code>	logical, whether or not the plots should be saved as files
<code>output_dir</code>	character, output directory of PNG-format figures
<code>cores</code>	integer, indicate the number of cores to use in parallel execution

Examples

```
data(DynModel_results) # Load sample model result object  
scan.length <- DynModel_results$params$ntp_per_scan  
plot_generate_timeDependence(DynModel_results$est_CI[1], scan.length, rois=c(114,134))
```

Index

*Topic **bootstarring**

MLPB_boot, [10](#)

*Topic **datasets**

coverage.tbl.list, [2](#)

DynModel_results, [4](#)

fMRI_dataList, [5](#)

fMRI_dataList_shrunked, [5](#)

MLPB_output_median, [11](#)

coverage.tbl.list, [2](#)

data_summary_coverage, [3](#)

DynModel_results, [4](#)

fMRI_dataList, [5](#)

fMRI_dataList_shrunked, [5](#)

ImmConn, [6](#)

ImmDyn, [7](#)

MLPB3, [9](#)

MLPB_boot, [10](#)

MLPB_output_median, [11](#)

nzc_vis, [11](#)

options_boot, [12](#)

options_lme, [13](#)

plot_generate_timeDependence, [14](#)