

Package ‘multiREG’

November 17, 2019

Type Package

Title Multiple Model Estimation using Regularization

Version 0.1.0

Date 2019-11-12

Author Ethan M. McCormick [aut, cre],
Kathleen M. Gates [aut]

Maintainer EM McCormick <emccormick@unc.edu>

Depends R (\geq 3.6.0)

Imports glmnet (\geq 2.0-18), gimme (\geq 0.6-0), qgraph(\geq 1.6.2), igraph(\geq 1.2.4.1), dplyr(\geq 0.8.1), imputeTS(\geq 2.7)

Description Directed connectivity search using regularization.
Utilizes the GIMME heuristic of using group information
to initialize individual model search.

License GPL-2

URL <https://github.com/McCormickNeuro/multiREG/>

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests testthat

R topics documented:

examplesim	2
HRFsim	2
multiREG	2
sitNeuro	6

Index	7
--------------	----------

examplesim	<i>Toy example for documentation</i>
------------	--------------------------------------

Description

This object contains a list of time series data for 3 individuals with 500 time points and 5 variables.

Usage

```
examplesim
```

Format

A list of data frames with 3 individuals, who each have 500 observations on 5 variables.

HRFsim	<i>Small Node example with exogenous variable</i>
--------	---

Description

This object contains a list of time series data for twenty-five individuals with 500 time points and 5 variables.

Usage

```
HRFsim
```

Format

A list of data frames with 25 individuals, who each have 500 observations on 5 variables.

multiREG	<i>Network Model Search using Regularization</i>
----------	--

Description

This function utilizes regression with regularization to build models for individuals consisting of individual and group-level paths.

Usage

```

multiREG(data           = NULL,
          out            = NULL,
          sep            = NULL,
          header         = TRUE,
          ar             = TRUE,
          plot           = TRUE,
          conv_vars      = NULL,
          conv_length    = 16,
          conv_interval  = 1,
          groupcutoff    = .75,
          alpha          = .5,
          model_crit     = 'bic',
          penalties      = NULL,
          test_penalties = FALSE,
          exogenous      = NULL,
          lag_exogenous  = FALSE,
          interactions    = NULL,
          subgroup       = FALSE,
          subgroupcutoff = .5,
          sub_method     = 'Walktrap',
          sub_feature     = 'count',
          confirm_subgroup = NULL,
          heuristic      = 'GIMME',
          verbose        = TRUE)

```

Arguments

data	The path to the directory where individual data files are located, or the name of the list containing individual data. Each file or matrix within the list must contain a single matrix containing the a T (time) by p (number of variables) matrix, where the rows represent time and columns represent individual variables. Individuals may have different numbers of observations (T), but must have the same number of variables (p).
out	(Optional) The path to directory where results will be stored. If specified, a copy of output data will be saved into the directory. If the specified directory does not exist, it will be created.
sep	Spacing scheme for input files. " " indicates space-separated; ',' indicates comma separated; '/t' indicates tab-separated Only necessary when reading in files from physical directory.
header	(Logical) Indicate TRUE if variable names included in input file, FALSE otherwise. Only necessary when reading in files from physical directory.
ar	(Logical) If TRUE, begin model search with all autoregressive pathways estimated with no shrinkage (i.e., penalty = 0).
plot	(Logical) IF TRUE, will create pdf plots of network maps during output.
conv_vars	Vector of variable names to be convolved via smoothed Finite Impulse Response (sFIR). Note, conv_vars are not not automatically considered exogenous variables. To treat conv_vars as exogenous use the exogenous argument. Variables listed in conv_vars must be binary variables. If there is missing data in the endogenous variables their values will be imputed

	for the convolution operation only. Defaults to NULL. ### If there are multiple variables listed in conv_vars they are not used in the convolution of additional conv_vars.## Lagged variables cannot be convolved.
conv_length	Expected response length in seconds. For functional magnetic resonance imagine (fMRI) blood-oxygenation-level-dependent (BOLD) response, 16 seconds (default) is typical for the hemodynamic response function.
conv_interval	Interval between data acquisition. Currently must be a constant. For fMRI studies, this is the repetition time. Defaults to 1.
groupcutoff	Cutoff value for inclusion of a given path at the group-level. For instance, group_cutoff = .75 indicates that a path needs to be estimated for 75 individuals to be included as a group-level path.
alpha	Elastic-net parameter for the regularization approach. Values close to 0 mimic the ridge penalty, which tends to shrink correlated parameters towards one another. Values close to 1 mimic the lasso penalty, which tends to select one parameter and shrink the others. The default value (alpha=.5) balances these two considerations, and tends to select groups of correlated parameters and shrink other groups towards zero.
model_crit	Argument to indicate the model selection criterion to use for model selection. Defaults to 'bic' (select on BIC). Options: 'bic', 'aic', 'aicc', 'hqc', 'cv' (cross-validation). BIC = Bayesian information criterion, AIC = Akaike information criterion, aicc = Akaike information criterion for small samples, hqc = Hannan-Quinn information criterion
penalties	(Optional) A matrix of user-provided penalties to initialize group-model search. Should contain a column for all variables (including lagged versions and interactions) that will be included in the model search. Values of 1 (the default) will initialize a variable to be normally considered in the regularization, values of 0 will initialize a variable to be estimated (i.e., no shrinkage), and values of Inf will exclude variables from the model.
test_penalties	(Optional, Logical) Optional argument to output a sample penalty matrix based on function parameters. Helpful for specifying a matrix to use in the penalties argument. Function will exit gracefully before running anything if test_penalties = TRUE.
exogenous	(Optional) A list of user-specified variables to consider as exogenous (e.g., cannot be predicted) in the model search procedure. If variable names are supplied, variables should be referred to by name. If not, then variables should be referenced by the pattern 'V#', where # represents the column number in the original data file (e.g., 'V5').
lag_exogenous	(Optional, Logical) If TRUE, a lagged version of the exogenous variable will be created. If set to TRUE, but exogenous variables are not indicated in the argument above, the function will not run properly.
interactions	(Optional) A list of user-specified interaction variables to be created automatically by the algorithm. Individual interactions can be specified as: c('V1*V2', 'V3*V5Lag', 'V2*V4Lag*V5'). WARNING: If specifying an N-way interaction where N > 2, make sure to specify the (N-1)-way interactions. These lower-order interactions will NOT be created automatically. For convenience, several shortcuts have been provided. Including 'all' in the list will create all possible 2-way interactions (including V^2 polynomials). Including 'all.cross' in the list will create all possible 2-way

interactions between variables (excluding V^2 polynomials). Including 'all_exogenous' will create all 2-way interactions between exogenous variables (excluding V^2 polynomials). Including 'all_endogenous' will create all 2-way interactions between endogenous variables (excluding V^2 polynomials). Including 'all_endog_by_exog' will create all 2-way interactions between pairs of endogenous and exogenous variables. Duplicated interactions are automatically, but caution when using shortcuts is encouraged. Shortcuts and specific interactions can be specified at the same time: `c('all_endog_by_exog', 'V3*V4Lag')`. However, including the options 'all' or 'all_cross' will cause other user-specified interactions to be ignored.

subgroup	Logical. If TRUE, subgroups are generated based on similarities in model features using the <code>walktrap.community</code> function from the <code>igraph</code> package. When <code>ms_allow=TRUE</code> , subgroup should be set to FALSE. Defaults to FALSE.
subgroupcutoff	Cutoff value for inclusion of a given path at the subgroup-level. For instance, <code>subgroup_cutoff = .5</code> indicates that a path needs to be estimated for 50 individuals within the subgroup to be included as a subgroup-level path.
sub_method	Community detection method used to cluster individuals into subgroups. Options align with those available in the <code>igraph</code> package: "Walktrap" (default), "Infomap", "Louvain", "Edge Betweenness", "Label Prop", "Fast Greedy", "Leading Eigen", and "Spinglass".
sub_feature	Features used to generate similarity matrix for subgrouping individuals if subgroup option invoked. "count" uses the counts of similar paths (default); "PCA" (principal components analysis) reduces the data to those components that explain at least 95 percent of variance and correlates these for each pair of individuals; "correlation" correlates all paths for each given pair of individuals to arrive at elements in the N-individual by N-individual similarity matrix.
confirm_subgroup	Option to specify a priori the subgroup membership. If not NULL, the user should provide a data frame with the first column a string vector of subject names and the second column a vector subgroup assignments.
heuristic	Approach for building individual network maps. The default ('GIMME') proceeds using group- and individual information. The 'individual' option causes the algorithm to ignore group-level information and estimate individuals independently. The 'group' option aggregates across individuals by concatenating all timeseries data; note that no individual-level results will be generated in this case and subgroup search will be disabled.
verbose	Logical. If TRUE, algorithm will print progress to console.

Value

Object containing individual regression matrices as well as plots if desired.

Author(s)

Ethan M. McCormick and Kathleen M. Gates

Examples

```
output = multiREG(data=examplesim, exogenous='V5', plot=FALSE)
```

sitNeuro*Medium Node example with 2 exogenous variables*

Description

This object contains a list of time series data for forty-eight individuals with 910 time points and 11 variables.

Usage

```
sitNeuro
```

Format

A list of data frames with 48 individuals, who each have 910 observations on 11 variables.

Index

- *Topic **datasets**
 - examplesim, [2](#)
 - HRFsim, [2](#)
 - sitNeuro, [6](#)
- *Topic **multiREG**
 - multiREG, [2](#)
- examplesim, [2](#)
- HRFsim, [2](#)
- multiREG, [2](#)
- sitNeuro, [6](#)