

Package ‘mnet’

July 23, 2025

Type Package

Title Modeling Group Differences and Moderation Effects in Statistical Network Models

Version 0.1.4

Maintainer Jonas Haslbeck <jonashaslbeck@protonmail.com>

Description A toolbox for modeling manifest and latent group differences and moderation effects in various statistical network models.

BugReports <https://github.com/jmbh/mnet/issues>

Encoding UTF-8

LazyData true

License GPL-2

Imports mlVAR, doParallel, parallel, foreach

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Jonas Haslbeck [aut, cre]

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2025-02-13 14:20:02 UTC

Contents

mlVAR_GC	2
VAR_GC	5
Index	8

Description

Does inference on group differences in mIVAR models using a permutation test. The function is a wrapper around the function `mIVAR()` from the `mIVAR` package.

Usage

```
mIVAR_GC(data, vars, idvar, dayvar = NULL,
          beepvar = NULL, groups, test = "permutation",
          partest = "Welch", paired = FALSE,
          estimator, contemporaneous, temporal,
          scale, nCores = 1, nP = 500,
          saveModels = FALSE, saveEmpModels = FALSE,
          verbose = FALSE, pbar = TRUE)
```

Arguments

<code>data</code>	The data.frame containing the data (rows=cases, columns=variables) including the variables to be modeled, a unique id per subject, and potentially additional variables about the measurement procedure (see <code>dayvar</code> and <code>dayvar</code> below).
<code>vars</code>	A character vector with the column names of the variables that should be included in the mIVAR models.
<code>idvar</code>	The name of the column containing the unique subject id.
<code>dayvar</code>	Optional: The name of the column indicating assessment day. Adding this argument makes sure that the first measurement of a day is not regressed on the last measurement of the previous day. IMPORTANT: only add this if the data has multiple observations per day. See also <code>mIVAR()</code> .
<code>beepvar</code>	Optional: The name of the column indicating measurement occasion per day. Adding this argument will cause non-consecutive beeps to be treated as missing! See also <code>mIVAR()</code> .
<code>groups</code>	The name of the column indicating group membership.
<code>test</code>	The type of test used. Defaults to <code>test = "permutation"</code> , which performs a permutation test. Alternatively, <code>test = "parametric"</code> performs a parametric test based on confidence intervals.
<code>partest</code>	The type of parametric test used if <code>test = "parametric"</code> is selected. Defaults to <code>partest = "Welch"</code> for Welch's t-test, which does not assume equal population variances. The alternative is <code>partest = "Student"</code> for Student's t-test.
<code>paired</code>	Indicates whether samples are paired or not (independent). If <code>paired = TRUE</code> indicates dependent samples. This requires that <code>idvar</code> needs to be specified such that each subject is contained in both groups, which implies that both groups have the same size. Also, the function assumes that the subjects are ordered in

	the same way in both groups. In this case, the permutation is computed based on within-person permutationst. The parametric test is not possible in this case since we are not able to obtain the required standard error from mIVAR models. If paired = FALSE the permutations are unrestricted. Defaults to paired = FALSE.
estimator	The estimator to be used. "lmer" for sequential univariate multi-level estimation, "Mplus" for multivariate Bayesian estimation (requires Mplus), and "lm" for fixed effects estimation. See also mIVAR().
contemporaneous	How should the contemporaneous networks be estimated? These networks are always estimated post-hoc by investigating the residuals of the temporal models. "correlated" and "orthogonal" run second multi-level models in which the networks are estimated using node-wise estimation. "fixed" and "unique" simply correlate the residuals, either by computing one network for all subjects (fixed) or a single network per per subject. See also mIVAR().
temporal	How should the temporal effects be estimated? "correlated" estimates correlated random effects, "orthogonal" estimates non-correlated random effects and "fixed" estimates a model in which only the intercept is random. Defaults to "correlated" when the number of variables is less than 6 and "orthogonal" otherwise. "unique" uses lm to estimate an unique model for each subject. See also mIVAR().
scale	Argument passed to mIVAR. If scale = TRUE, each variable is standardized before estimation.
nCores	Number of cores to use in computation. See also mIVAR().
nP	Number of permutations used to create the sampling distribution under the null hypothesis for the permutation test. Defaults to nP = 500.
saveModels	If true, the pair of mIVAR objects for each permutations are saved in the output. Defaults to saveModels = FALSE since the resulting output object can become very large.
saveEmpModels	If true, the pair of mIVAR objects estimated on the two groups in the empirical data are returned in a list of two. The first entry contains the mIVAR model object for the first group. Defaults to saveEmpModels = FALSE to save space.
verbose	If true, the the verbose option in foreach, which is used in side mIVAR_GC(), is switched on. Defaults to verbose = FALSE.
pbar	If true, a progress bar is shown. Defaults to pbar = TRUE.

Details

The output consists of a list with the following entries:

Value

Returns a list with the following entries:

EmpDiffs	A list containing five matrices with the empirically observed differences (Group 1 - Group 2) in the between-person network, the fixed and random effects variances of the temporal network, and the fixed and random effects variances of the contemporaneous network.
----------	---

Pval	Same structure as TrueDiffs but provides p-values for the corresponding observed group differences. Note that for the parametric test, p-values are only available for fixed lagged effects, fixed contemporaneous (residual) effects, and the between network effects. This is because mlVAR only provides standard errors for those types of parameters.
SampDist	A list of arrays, containing the sampling distributions for between-person network, the fixed and random effects variances of the temporal network, and the fixed and random effects variances of the contemporaneous network.
Models	If saveModels = TRUE, this contains a list with all pairs of mlVAR models for each permutation.
Runtime	The runtime of the function in minutes.

Author(s)

Jonas Haslbeck <jonashaslbeck@protonmail.com>

References

Epskamp, S., Waldorp, L. J., Mottus, R., & Borsboom, D. (2018). The Gaussian graphical model in cross-sectional and time-series data. *Multivariate behavioral research*, 53(4), 453-480.

Examples

```
# Use simulated example data loaded with package
head(ExampleData)

# Call Permutation test
out <- mlVAR_GC(data = ExampleData,
                vars = c("V1", "V2", "V3"),
                idvar = "id",
                groups = "group",
                nCores = 2, # choose max cores possible on your machine
                nP = 2) # Should be much more in practice, see paper!

# P-values for the five parameter types:
out$Pval
# e.g., we see that there is a small p-value for
out$Pval$Phi_mean[1,2]
# which is a parameter with a true group difference
# (which we know since this is simulated data)

# The observed group differences (i.e., the test statistics)
# Can be found in:
out$EmpDiffs

# Specifically, the difference is: group 1 - group 2
# For example:
out$EmpDiffs$Phi_mean[1,2,]
# The true group difference was -0.4
```

VAR_GC	<i>Function to test whether parameters of two VAR(1) models are different</i>
--------	---

Description

Function to test whether parameters of two VAR(1) models are different

Usage

```
VAR_GC(data, vars, dayvar, beepvar,
        groups, test = "parametric",
        nP = 1000)
```

Arguments

data	A n x p data matrix.
vars	An integer vector indicating the column numbers of the variables that should be modeled in the pair of VAR models.
dayvar	Integer indicating the column number of the variable specifying the day of the measurement point. This is designed for EMA studies. If not applicable, this argument can be left unspecified.
beepvar	Integer indicating the column number of the variable specifying the number of the measurement occasion during a specific day. The full set of measurement occasions has to be consecutive and increasing sequence of integers (e.g., 1, 2,..., 5). This is designed for EMA studies. If not applicable, this argument can be left unspecified.
groups	Integer indicating the column number of the group variable. The values of the group variable have to be 1 and 2.
test	If test = "parametric" p-values are computed using the t-distribution, using the Welch-Satterthwaite equation to approximate degrees of freedom (df). This is the default. Alternatively, test = "permutation" uses a permutation to sample from a distribution in which the null hypothesis that no group differences exist is true. The permutation uses nP permuted datasets.
nP	The number of permuted datasets used if test = "permutation".

Value

Call	Returns the call of the function
phi_diff	A p x p matrix of differences in phi coefficients (Group 1 - Group 2) in the empirical data. The test-statistics.

`phi_pval` A $p \times p$ matrix with pvalues corresponding to `phi_diff`.
`int_diff` A p numeric vector of differences in intercepts (Group 1 - Group 2).
`int_pvar` A p numeric vector of pvalues corresponding to `int_diff`

Author(s)

Jonas Haslbeck <jonashaslbeck@protonmail.com>

Examples

```

library(mlVAR) # for simulateVAR() function

# Specify Model
p <- 4
A1 <- diag(p) * 0.8
A2 <- diag(p) * 0.8
A2[2,1] <- 0.7

# Simulate datasets
Nt <- 500
set.seed(13) # for reproducibility
data1_x <- simulateVAR(A1, means=rep(0, p), Nt = Nt, residuals=.1)
data2_x <- simulateVAR(A2, means=rep(0, p), Nt = Nt, residuals=.1)

# Add beep and day vars
dayvar1 <- dayvar2 <- rep(1:(Nt/5), each=5)
beepvar1 <- beepvar2 <- rep(1:5, Nt/5)

# Add grouping var
groups1 <- rep(1, Nt)
groups2 <- rep(2, Nt)

# Combine
data1 <- data.frame(cbind(dayvar1, beepvar1, groups1, data1_x))
data2 <- data.frame(cbind(dayvar2, beepvar2, groups2, data2_x))
colnames(data1) <- colnames(data2) <- c("dayvar", "beepvar", "groups", paste0("V", 1:4))
data <- rbind(data1, data2)

# Call
out <- VAR_GC(data = data,
               vars = 4:7,
               dayvar = 1,
               beepvar = 2,
               groups = 3)

round(out$phi_pval, 2)
round(out$phi_pval[2,1], 2) # worked!

```


Index

m1VAR_GC, [2](#)

VAR_GC, [5](#)