Package ‘pompom’

July 13, 2018

Type Package

Title Person-Oriented Method and Perturbation on the Model

Version 0.2.0

Maintainer Xiao Yang <xfy5031@psu.edu>

Description An implementation of a hybrid method of person-oriented method and perturbation on the model. Pompom is the initials of the two methods. The hybrid method will provide a multivariate intraindividual variability metric (iRAM). The person-oriented method used in this package refers to uSEM (unified structural equation modeling, see Kim et al., 2007, Gates et al., 2010 and Gates et al., 2012 for details). Perturbation on the model was conducted according to impulse response analysis introduced in Lutkepohl (2007).


License GPL-2

Encodings UTF-8

LazyData true

RoxygenNote 6.0.1

Depends R (>= 3.0.0)

Imports lavaan (>= 0.5.23.1097), ggplot2 (>= 2.2.1), reshape2 (>= 1.4.2), qgraph, utils

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation no

Author Xiao Yang [cre, aut], Nilam Ram [aut], Peter Molenaar [aut]

Repository CRAN

Date/Publication 2018-07-13 20:10:03 UTC
R topics documented:

- **bootstrap_iRAM_2node**
- **bootstrap_iRAM_3node**
- **iRAM**
- **iRAM_equilibrium**
- **model_summary**
- **parse_beta**
- **plot_integrated_time_profile**
- **plot_iRAM_dist**
- **plot_network_graph**
- **plot_time_profile**
- **simts_2node**
- **simts_3node**
- **true_beta_2node**
- **true_beta_3node**
- **uSEM**
- **usemmodelfit**

Index

<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>bootstrap_iRAM_2node</td>
<td>2</td>
</tr>
<tr>
<td>bootstrap_iRAM_3node</td>
<td>3</td>
</tr>
<tr>
<td>iRAM</td>
<td>4</td>
</tr>
<tr>
<td>iRAM_equilibrium</td>
<td>5</td>
</tr>
<tr>
<td>model_summary</td>
<td>6</td>
</tr>
<tr>
<td>parse_beta</td>
<td>7</td>
</tr>
<tr>
<td>plot_integrated_time_profile</td>
<td>8</td>
</tr>
<tr>
<td>plot_iRAM_dist</td>
<td>8</td>
</tr>
<tr>
<td>plot_network_graph</td>
<td>9</td>
</tr>
<tr>
<td>plot_time_profile</td>
<td>9</td>
</tr>
<tr>
<td>simts_2node</td>
<td>10</td>
</tr>
<tr>
<td>simts_3node</td>
<td>11</td>
</tr>
<tr>
<td>true_beta_2node</td>
<td>11</td>
</tr>
<tr>
<td>true_beta_3node</td>
<td>12</td>
</tr>
<tr>
<td>uSEM</td>
<td>12</td>
</tr>
<tr>
<td>usemmodelfit</td>
<td>14</td>
</tr>
</tbody>
</table>

### Description

Bootstrapped iRAM (including replications of iRAM and corresponding time profiles) for the bivariate time-series (simts2node)

### Usage

```r
bootstrap_iRAM_2node
```

### Format

An object of class `list` of length 5.

### Details

Data bootstrapped from the estimated three-node network structure with 200 replications.
Examples

```
bootstrap_IRAM_3node$mean # mean of bootstrapped iRAM
bootstrap_IRAM_3node$upper # upper bound of confidence interval of bootstrapped iRAM
bootstrap_IRAM_3node$lower # lower bound of confidence interval of bootstrapped iRAM
bootstrap_IRAM_3node$time.profile.data # time profiles generated from the bootstrapped beta matrices
bootstrap_IRAM_3node$recovery.time.reps # iRAMs generated from the bootstrapped beta matrices
```

Description

Bootstrapped iRAM (including replications of iRAM and corresponding time profiles) for the 3-variate time-series (simts)

Usage

```
bootstrap_IRAM_3node
```

Format

An object of class list of length 5.

Details

Data bootstrapped from the estimated three-node network structure with 200 replications.

Examples

```
bootstrap_IRAM_3node$mean # mean of bootstrapped iRAM
bootstrap_IRAM_3node$upper # upper bound of confidence interval of bootstrapped iRAM
bootstrap_IRAM_3node$lower # lower bound of confidence interval of bootstrapped iRAM
bootstrap_IRAM_3node$time.profile.data # time profiles generated from the bootstrapped beta matrices
bootstrap_IRAM_3node$recovery.time.reps # iRAMs generated from the bootstrapped beta matrices
```
iRAM

Generate iRAM (impulse response anlaysis metric) from model fit.

Description

Generate iRAM (impulse response anlaysis metric) from model fit.

Usage

iRAM(model.fit, beta, var.number, lag.order = 1, threshold = 0.01,
     boot = FALSE, replication = 200, steps = 100)

Arguments

- **model.fit**: model fit object generated by lavaan
- **beta**: beta matrix for a point estimate
- **var.number**: number of variables in the time series
- **lag.order**: lag order of the model to be fit
- **threshold**: threshold of calculation of recovery time (duration of perturbation), default value is 0.01
- **boot**: to bootstrap, default value is FALSE
- **replication**: number of replication of bootstrap, default value is 200
- **steps**: number of steps of impulse response analysis, default value is 100

Value

iRAM matrix. Rows represent where the orthogonal impulse was given, and columns represent the response. Dimension is var.number by var.number.

References


Examples

```r
boot.iRAM <- iRAM(model.fit = useemmodelfit,
                  beta = NULL,
                  var.number = 3,
                  lag.order = 1,
                  threshold = 0.01,
                  boot = TRUE,
                  replication = 200,
                  steps = 100
)
```
Generate iRAM (impulse response analysis metric) in the equilibrium form.

Usage

irm_equilibrium(beta_matrix, var_number, lag_order)

Arguments

- beta_matrix: beta matrix for a point estimate
- var_number: number of variables in the time series
- lag_order: lag order of the model to be fit

Value

A list of equilibria. First numeric number in the variable name indicates where the impulse was given, and the second numeric number indicates the response, e.g., e12 indicates equilibrium of node 2 when node 1 is given an impulse.

Examples

```r
irm_eval <- irm_equilibrium(beta.matrix = true_beta_3node,
                              var.number = 3,
                              lag.order = 1)
irm_eval
```
Provide model summary.

Usage
model_summary(model.fit, var.number, lag.order)

Arguments
model.fit       model fit object generated by lavaan
var.number     number of variables in the time-series
lag.order      lag order of model

Details
Model fit criteria: 3 out of 4 rule, meaning 3 out of 4 criteria should be satisfied, including CFI and TLI should be greater than 0.95, RMSEA and SRMR should be less than 0.08.

Value
beta matrix estimates
matrix of standard error of beta
matrix of psi estimates
fit statistics CFI
fit statistics TLI
fit statistics RMSEA
fit statistics SRMR

Examples
mdl <- model_summary(model.fit = usemmodelfit,
                      var.number = 3,
                      lag.order = 1)
mdl$beta
mdl$beta.se
mdl$psi
mdl$cfi
mdl$tli
mdl$rmsea
mdl$srmr
parse_beta  Parse the beta from model fit object

Description
Parse the beta from model fit object

Usage
parse_beta(var.number, model.fit, lag.order, matrix = F)

Arguments
- var.number: number of variables in the time series
- model.fit: model fit object generated by lavaan
- lag.order: lag order of the model to be fit
- matrix: output beta in matrix format or estimates format, default value is FALSE (as estimates)

Value
beta

Examples

data(usemmmodelfit)
beta.matrix <- parse_beta(var.number = 3,
                          model.fit = usemmmodelfit,
                          lag.order = 1,
                          matrix = TRUE)
beta.matrix
plot_integrated_time_profile

*Plot the time profiles in the integrated form*

**Description**

Plot the time profiles in the integrated form

**Usage**

```r
plot_integrated_time_profile(beta.matrix, var.number, lag.order = 1)
```

**Arguments**

- `beta.matrix`: matrix of temporal relations, containing both lag-1 and contemporaneous
- `var.number`: number of variables in the time series
- `lag.order`: lag order of the model to be fit

**Examples**

```r
plot_integrated_time_profile(beta.matrix = true_beta_3node,
                            var.number = 3,
                            lag.order = 1)
```

---

plot_iRAM_dist

*Plot distribution of recovery time based on bootstrapped version of iRAM*

**Description**

Plot distribution of recovery time based on bootstrapped version of iRAM

**Usage**

```r
plot_iRAM_dist(recovery.time.reps)
```

**Arguments**

- `recovery.time.reps`: bootstrapped version of recovery time
plot_network_graph

Examples

plot_iRAM_dist(bootstrap_iRAM_3node$recovery.time.reps)

plot_network_graph

Plot the network graph

Description

Plot the network graph

Usage

plot_network_graph(beta, var.number)

Arguments

beta matrix of temporal relations, containing both lag-1 and contemporaneous
var.number number of variables in the time series

Examples

plot_network_graph(beta = true_beta_3node,
                   var.number = 3)

plot_time_profile

Plot time profiles given a time-series generated by impulse response analysis

Description

Plot time profiles given a time-series generated by impulse response analysis

Usage

plot_time_profile(time.series.data, var.number, threshold = 0.01,
xupper = 20)
Arguments

- `time.series.data`: data of impulse response in long format
- `var.number`: number of variables in the time-series
- `threshold`: threshold of asymptote of equilibrium
- `xupper`: upper limit of x-axis

Examples

```r
plot_time_profile(time.series.data = bootstrap_iRAM_2node$time.profile.data,
                  var.number = 2,
                  threshold = .01,
                  xupper = 20)
```

---

### simts_2node

**Simulated bivariate time-series data**

**Description**

Simulated bivariate time-series data

**Usage**

`simts_2node`

**Format**

An object of class `data.frame` with 200 rows and 2 columns.

**Details**

Data simulated from a given three-node network structure with 200 measurements. Network structure is shown in the dataset `true.beta`. Process noise has mean of 0 and SD .1.

**Examples**

```r
data(simts_2node)
```
**Description**

Simulated 3-variate time-series data

**Usage**

simts_3node

**Format**

An object of class `data.frame` with 100 rows and 3 columns.

**Details**

Data simulated from a given three-node network structure with 200 measurements. Network structure is shown in the dataset `true.beta`. Process noise has mean of 0 and SD .1.

**Examples**

```r
data(simts_3node)
```

---

**true_beta_2node**

*The true beta matrix (4 by 4) used in simulation.*

**Description**

The true beta matrix (4 by 4) used in simulation.

**Usage**

true_beta_2node

**Format**

An object of class `matrix` with 4 rows and 4 columns.

**Details**

```r
ture_beta_2node <- matrix(c(0,0,0,0, 0,0,0,0, 0.2,-.4,0,-0.25, 0,0.3,-0.2,0), nrow = 4, ncol = 4, by-row = TRUE)
```
Examples

true_beta_2node

true_beta_3node  The true beta matrix (6 by 6) used in simulation.

Description
The true beta matrix (6 by 6) used in simulation.

Usage
true_beta_3node

Format
An object of class matrix with 6 rows and 6 columns.

Details
ture_beta_3node <- matrix(c(0,0,0,0,0,0, 0,0,0,0,0,0, 0,0,0,0,0,0, 0.2,0,0.25,0,0,0.6, 0,0.3,0,-0.2,0,-0.6, 0,-0.2,0.3,0,0,0), nrow = 6, ncol = 6, byrow = TRUE)

Examples

true_beta_3node

uSEM  Fit a multivariate time series with uSEM (unified Structural Equation Model).

Description
Fit a multivariate time series with uSEM (unified Structural Equation Model).
uSEM

Usage

```
uSEM(var.number, 
    data, 
    lag.order = 1, 
    verbose = FALSE, 
    trim = FALSE)
```

Arguments

- `var.number`: number of variables in the time series
- `data`: time series data, must be in long format
- `lag.order`: lag order of the model to be fit, default value is 1. Note: Higher order (greater than 1) might not run.
- `verbose`: print intermediate model fit (iterations), default value is FALSE
- `trim`: to trim the insignificant betas (just one step, not iterative), default value is FALSE

Details

The purpose of uSEM is to quantify the temporal relations (both contemporaneous and lag-1) between variables. Model specification and estimation can be found in the references.

Value

model fit object generated by lavaan

References


Examples

```
model.fit <- uSEM(var.number = 3, 
                   data = simts_3node, 
                   lag.order = 1, 
                   verbose = FALSE, 
                   trim = FALSE)

model.fit
```
usemmodelfit

Model fit based on simulated time-series by uSEM.

Description
Model fit based on simulated time-series by uSEM.

Usage
usemmodelfit

Format
An object of class lavaan of length 1.

Examples

data(usemmodelfit)
Index

*Topic datasets
  - bootstrap_iRAM_2node, 2
  - bootstrap_iRAM_3node, 3
  - simts_2node, 10
  - simts_3node, 11
  - true_beta_2node, 11
  - true_beta_3node, 12
  - usemmodelfit, 14

bootstrap_iRAM_2node, 2
bootstrap_iRAM_3node, 3

iRAM, 4
iRAM_equilibrium, 5

model_summary, 6

parse_beta, 7
plot_integrated_time_profile, 8
plot_iRAM_dist, 8
plot_network_graph, 9
plot_time_profile, 9

simts_2node, 10
simts_3node, 11

true_beta_2node, 11
true_beta_3node, 12

uSEM, 12
usemmodelfit, 14