

Package: efast (via r-universe)

January 17, 2025

Type Package

Title Exploratory Factor Analysis with Structured Residuals

Version 0.6.3

Author Erik-Jan van Kesteren

Maintainer Erik-Jan van Kesteren <e.vankesteren1@uu.nl>

Description Create and estimate EFA and EFA with structured residuals (EFAST) models using structural equation modeling.

License GPL-3

Encoding UTF-8

LazyData true

Imports stringr, stats, MASS, corrplot

RoxygenNote 7.1.1

Suggests testthat (>= 2.1.0), knitr, rmarkdown

Depends lavaan (>= 0.6.4), R (>= 3.5.0)

VignetteBuilder knitr

Config/pak/sysreqs libicu-dev

Repository <https://vankesteren.r-universe.dev>

RemoteUrl <https://github.com/vankesteren/efast>

RemoteRef HEAD

RemoteSha 4c3bfb819d0131f5655045a2bab514bd5da84d61

Contents

cplot	2
decomposition	3
efast	3
efast_efa	5
efast_hemi	6
efast_loadings	7
is_efast	7

is_efast_efa	8
is_efast_hemi	8
lateralization	8
roi_volume	9
simulate_efast	9

Index	11
--------------	-----------

cplot	<i>Create a clean correlation plot</i>
-------	--

Description

This function is a thin wrapper around `corrplot`

Usage

```
cplot(
  mat,
  tl.pos = "n",
  cl.pos = "n",
  method = "square",
  addgrid.col = NA,
  ...
)
```

Arguments

<code>mat</code>	<matrix> or <list<matrix>> the matrix or list of matrices to be displayed.
<code>tl.pos</code>	<string> text label position, see <code>corrplot::corrplot()</code>
<code>cl.pos</code>	<string> color label position, see <code>corrplot::corrplot()</code>
<code>method</code>	<string> drawing method, see <code>corrplot::corrplot()</code>
<code>addgrid.col</code>	<string> grid colour see <code>corrplot::corrplot()</code>
<code>...</code>	other arguments passed to <code>corrplot::corrplot()</code>

See Also

[corrplot](#)

decomposition	<i>Get correlation matrix decomposition</i>
---------------	---

Description

Get correlation matrix decomposition

Usage

```
decomposition(fit)
```

Arguments

fit an efast model

Value

The decomposition returns four components: the observed correlation matrix, the factor-implied covariance, the residual variance, and the structural covariance matrix. You can plot these matrices using `corrplot`.

efast	<i>Estimate an EFAST model</i>
-------	--------------------------------

Description

This function estimates efa models with residual covariance.

Usage

```
efast(  
  data,  
  M,  
  rstruct,  
  sample.nobs = NULL,  
  auto.fix.first = FALSE,  
  auto.var = TRUE,  
  auto.efa = TRUE,  
  information = "observed",  
  std.ov = TRUE,  
  ...  
)
```

Arguments

data	<data.frame> the dataset or <matrix> covariance matrix
M	<numeric> How many factors, minimum 2
rstruct	<list> residual structure (see details)
sample.nobs	<numeric> sample size (if data = covmat, see lavaan)
auto.fix.first	<bool> see lavaan
auto.var	<bool> see lavaan
auto.efa	<bool> see lavaan
information	<character> see lavaan
std.ov	<bool> see lavaan
...	other arguments passed to lavaan

Details

Residual structure (in the form of residual covariances) can be added to the EFA through a list of pairs of variable names. See the example for more information.

Examples

```
## Not run:
# Use a lavaan test dataset
test_data <- lavaan::HolzingerSwineford1939[,7:15]

# create an EFA model
test_efa <- efast(test_data, 3)

# create a (simple) residual structure
res_struct <- list(
  c("x4", "x7"),
  c("x5", "x9")
)

# create an efast model
test_efast <- efast(test_data, 3, res_struct)

compare the models
lavaan::lavTestLRT(test_efa, test_efast)

## End(Not run)
```

efast_efa *Estimate an EFA model in lavaan*

Description

This function estimates efa models in the same way that efast models are estimated: using lavaan.

Usage

```
efast_efa(  
  data,  
  M,  
  sample.nobs = NULL,  
  auto.fix.first = FALSE,  
  auto.var = TRUE,  
  auto.efa = TRUE,  
  information = "observed",  
  std.ov = TRUE,  
  ...  
)
```

Arguments

data	<data.frame> the dataset or <matrix> covariance matrix
M	<numeric> How many factors, minimum 2
sample.nobs	<numeric> sample size (if data = covmat, see lavaan)
auto.fix.first	<bool> see lavaan
auto.var	<bool> see lavaan
auto.efa	<bool> see lavaan
information	<character> see lavaan
std.ov	<bool> see lavaan
...	other arguments passed to lavaan

Details

The constrained model constrains the residual covariance to be equal across the different ROIs.

Examples

```
## Not run:  
# create a test dataset  
test_data <- simulate_efast()  
fit_efa <- efast_efa(simdat, M = 4)  
summary(fit_efa)  
  
## End(Not run)
```

 efast_hemi

Estimate an EFAST-hemi model

Description

This function estimates efast models with covariance due to hemispheric symmetry.

Usage

```
efast_hemi(
  data,
  M,
  lh_idx,
  rh_idx,
  roi_names,
  constrain = FALSE,
  sample.nobs = NULL,
  auto.fix.first = FALSE,
  auto.var = TRUE,
  auto.efa = TRUE,
  information = "observed",
  std.ov = TRUE,
  ...
)
```

Arguments

data	<data.frame> the dataset or <matrix> covariance matrix
M	<numeric> How many factors, minimum 2
lh_idx	<numeric> column numbers of left hemisphere variables
rh_idx	<numeric> column numbers of right hemisphere variables
roi_names	<character> optional names of rois
constrain	<bool> whether to constrain the symmetry (see details)
sample.nobs	<numeric> sample size (if data = covmat, see lavaan)
auto.fix.first	<bool> see lavaan
auto.var	<bool> see lavaan
auto.efa	<bool> see lavaan
information	<character> see lavaan
std.ov	<bool> see lavaan
...	other arguments passed to lavaan

Details

The constrained model constrains the residual covariance to be equal across the different ROIs.

Examples

```
## Not run:  
# create a test dataset  
test_data <- simulate_efast()  
fit_efast <- efast_hemi(test_data, M = 4, 1:17, 18:34)  
summary(fit_efast)  
  
## End(Not run)
```

efast_loadings	<i>Get loadings</i>
----------------	---------------------

Description

Get loadings

Usage

```
efast_loadings(fit, symmetry = FALSE)
```

Arguments

fit	efa or efast model
symmetry	whether to display the left and right hemisphere loadings side-by-side

is_efast	<i>Check whether an object is an efast model</i>
----------	--

Description

Check whether an object is an efast model

Usage

```
is_efast(x)
```

Arguments

x	any object
---	------------

is_efast_efa *Check whether an object is an efast-efa model*

Description

Check whether an object is an efast-efa model

Usage

is_efast_efa(x)

Arguments

x any object

is_efast_hemi *Check whether an object is an efast-hemi model*

Description

Check whether an object is an efast-hemi model

Usage

is_efast_hemi(x)

Arguments

x any object

lateralization *Get lateralization per ROI*

Description

The lateralization index (LI) is calculated as (sum of the uniquenesses for the contralateral homologues) / (total residual variance after accounting for the exploratory factors). Thus, if there is no symmetry at all, the LI is 1, and if all the residual variance is explained by symmetry, the LI is 0.

Usage

lateralization(fit)

Arguments

fit an efast model

Value

parameter estimates table of lateralization per ROI

roi_volume	<i>Synthesised volume data for DKT-atlas ROIs.</i>
------------	--

Description

This data has been synthesised on the basis of real-world data from the Cam-CAN cohort. It contains 68 grey matter volume measurements (34 LH and 34 RH ROIs) for 647 participants.

Usage

```
data(roi_volume)
```

Format

A data frame with 647 rows and 68 columns

References

cam-can.org

simulate_efast	<i>Simulate data from an efast matrix</i>
----------------	---

Description

Simulate a dataset as in the simulations of Van Kesteren & Kievit (2019). There are 17 regions of interest, measured in both the left and right hemisphere. These ROIs have a predefined amount of correlation over and above that expected by only the underlying factors.

Usage

```
simulate_efast(
  N = 650L,
  lam_lat = 0.595,
  lam_bil = 0.7,
  psi_cov = 0.5,
  cor_uniq = 0.4
)
```

Arguments

N	<int> Sample size
lam_lat	<numeric> factor loading for the lateralised factor
lam_bil	<numeric> factor loading for the bilateral factors
psi_cov	<numeric> covariances of latent variables in (0, 1)
cor_uniq	<numeric> residual correlation

Value

data frame with 17 regions of interest, bilaterally measured with 4 underlying factors and contralateral homology.

References

Van Kesteren, E. J., & Kievit, R. K. (2019) Exploratory factor analysis with structured residuals applied to brain morphology.

Index

* datasets

roi_volume, 9

corrplot, 2

cplot, 2

decomposition, 3

efast, 3

efast_efa, 5

efast_hemi, 6

efast_loadings, 7

is_efast, 7

is_efast_efa, 8

is_efast_hemi, 8

lateralization, 8

roi_volume, 9

simulate_efast, 9