

Package ‘easybgm’

March 13, 2024

Type Package

Title Extracting and Visualizing Bayesian Graphical Models

Version 0.1.2

Maintainer Karoline Huth <k.huth@uva.nl>

Description

Fit and visualize the results of a Bayesian analysis of networks commonly found in psychology. The package supports fitting cross-sectional network models fitted using the packages 'BDgraph', 'bgms' and 'BGGM'. The package provides the parameter estimates, posterior inclusion probabilities, inclusion Bayes factor, and the posterior density of the parameters. In addition, for 'BDgraph' and 'bgms' it allows to assess the posterior structure space. Furthermore, the package comes with an extensive suite for visualizing results.

License GPL (>= 2)

URL <https://github.com/KarolineHuth/easybgm>

BugReports <https://github.com/KarolineHuth/easybgm/issues>

Encoding UTF-8

RoxygenNote 7.2.3

Imports BDgraph, BGGM, bgms (>= 0.1.3), dplyr, ggplot2, HDInterval, igraph, qgraph

Suggests testthat (>= 3.0.0), vdiff

Config/testthat/edition 3

NeedsCompilation no

Author Karoline Huth [aut, cre] (<<https://orcid.org/0000-0002-0662-1591>>), Sara Keetelaar [ctb]

Repository CRAN

Date/Publication 2024-03-13 13:40:02 UTC

R topics documented:

bgm_extract	2
bgm_fit	2
centrality	3
complexity_probs	4
easybgm	4
edgevidence	7
HDI	9
network	10
print.easybgm	11
prior_sensitivity	11
structure	12
structure_probs	13
summary.easybgm	14

Index	16
--------------	-----------

bgm_extract	<i>Extract the results of a Bayesian analysis of networks</i>
-------------	---

Description

Extract the results of a Bayesian analysis of networks

Usage

```
bgm_extract(fit, ...)
```

Arguments

fit	Fit object with a particular class that will dispatch to the respective package functions
...	Additional arguments to be passed onto the respective fitting functions

bgm_fit	<i>Fit a Bayesian analysis of networks</i>
---------	--

Description

Fit a Bayesian analysis of networks

Usage

```
bgm_fit(fit, ...)
```

Arguments

<code>fit</code>	Object with a particular class that will dispatch to the respective package functions
<code>...</code>	Additional arguments to be passed onto the respective fitting functions

<code>centrality</code>	<i>Plot strength centralities and 95% highest density interval</i>
-------------------------	--

Description

Visualize the strength centralities and their uncertainties. The centrality estimate can be obtained for each sample of the posterior distribution of the association parameters to obtain an estimate of the uncertainty of the strength centrality estimate.

Usage

```
plot_centrality(output, ...)
```

Arguments

<code>output</code>	Output object from the <code>easybgm</code> function. Supports also objects from the <code>bgm</code> function of the <code>bgms</code> package.
<code>...</code>	Additional arguments passed onto <code>ggplot2</code>

Value

Returns a plot

Examples

```
library(easybgm)
library(bgms)

data <- na.omit(Wenchuan)
subdata <- data[1:50, 1:3] #for demonstration choosing a subset to increase computation speed
fit <- easybgm(subdata, type = "ordinal",
              iter = 1000, # for demonstration only (> 5e4 recommended)
              save = TRUE, centrality = TRUE)

plot_centrality(fit)
```

complexity_probs	<i>Plot posterior complexity probabilities</i>
------------------	--

Description

Plots the posterior complexity probabilities of all visited structures, where complexity comprises the network density.

Usage

```
plot_complexity_probabilities(output, ...)
```

Arguments

output	Output object from the easybgm function. Supports also objects from the bgm function of the bgms package.
...	Additional arguments passed onto ggplot2

Value

Returns a plot

Examples

```
library(easybgm)
library(bgms)

data <- na.omit(Wenchuan)
subdata <- data[1:50, 1:3] #for demonstration choosing a subset to increase computation speed
fit <- easybgm(subdata, type = "ordinal", save = TRUE, edge_selection = TRUE,
              iter = 1000 # for demonstration only (> 5e4 recommended)
              )

plot_complexity_probabilities(fit)
```

easybgm	<i>Fit a Bayesian analysis of networks</i>
---------	--

Description

Easy estimation of a Bayesian analysis of networks to obtain conditional (in)dependence relations between variables in a network.

Usage

```

easybgm(
  data,
  type,
  package = NULL,
  not_cont = NULL,
  iter = 10000,
  save = FALSE,
  centrality = FALSE,
  progress = TRUE,
  posterior_method = "model-averaged",
  ...
)

```

Arguments

<code>data</code>	An $n \times p$ matrix or dataframe containing the variables for n independent observations on p variables.
<code>type</code>	What is the data type? Options: continuous, mixed, ordinal, binary
<code>package</code>	The R-package that should be used for fitting the network model; supports BGGM, BDgraph, and bgms. Optional argument; default values are specified depending on the datatype.
<code>not_cont</code>	If data-type is mixed, a vector of length p , specifying the not-continuous variables (1 = not continuous, 0 = continuous).
<code>iter</code>	number of iterations for the sampler.
<code>save</code>	Logical. Should the posterior samples be obtained (default = FALSE)?
<code>centrality</code>	Logical. Should the centrality measures be extracted (default = FALSE)? Note, that it will significantly increase the computation time.
<code>progress</code>	Logical. Should a progress bar be shown (default = TRUE)?
<code>posterior_method</code>	Determines how the posterior samples of the edge weight parameters are obtained for models fit with BDgraph. The argument can be either MAP for the maximum-a-posteriori or model-averaged. If MAP, samples are obtained for the edge weights only for the most likely structure. If model-averaged, samples are obtained for all plausible structures weighted by their posterior probability. Default is model-averaged.
<code>...</code>	Additional arguments that are handed to the fitting functions of the packages, e.g., informed prior specifications.

Details

Users may oftentimes wish to deviate from the default, usually uninformative, prior specifications of the packages to informed priors. This can be done by simply adding additional arguments to the `easybgm` function. Depending on the package that is running the underlying network estimation, researcher can specify different prior arguments. We give an overview of the prior arguments per package below.

bgms:

- `interaction_prior` prior distribution of the interaction parameters, can be either "UnitInfo" for the Unit Information prior, or "Cauchy" for the Cauchy distribution. The default is set to "UnitInfo".
- `edge_prior` prior on the graph structure, which can be either "Bernoulli" or "Beta-Bernoulli". The default is "Bernoulli".
- `inclusion_prior` prior edge inclusion probability for the "Bernoulli" distribution. The default is 0.5.
- `beta_bernoulli_alpha` and `beta_bernoulli_alpha` the parameters of the "Beta-Bernoulli" distribution. The default is 1 for both.
- `threshold_alpha` and `threshold_beta` the parameters of the beta-prime distribution for the threshold parameters. The defaults are both set to 1.

BDgraph:

- `df.prior` prior on the parameters (i.e., inverse covariance matrix), degrees of freedom of the prior G-Wishart distribution. The default is set to 2.5.
- `g.prior` prior probability of edge inclusion. This can be either a scalar, if it is the same for all edges, or a matrix, if it should be different among the edges. The default is set to 0.5.

BGGM:

- `prior_sd` the standard deviation of the prior distribution of the interaction parameters, approximately the scale of a beta distribution. The default is 0.25.

We would always encourage researcher to conduct prior robustness checks.

Value

The returned object of `easybgm` contains several elements:

- `parameters` A $p \times p$ matrix containing partial associations.
- `inc_probs` A $p \times p$ matrix containing the posterior inclusion probabilities.
- `BF` A $p \times p$ matrix containing the posterior inclusion Bayes factors.
- `structure` Adjacency matrix of the median probability model (i.e., edges with a posterior probability larger 0.5).

In addition, for `BDgraph` and `bgms`, the function returns:

- `structure_probabilities` A vector containing the posterior probabilities of all visited structures, between 0 and 1.
- `graph_weights` A vector containing the number of times a particular structure was visited.
- `sample_graphs` A vector containing the indexes of a particular structure.

For all packages, when setting `save = TRUE` and `centrality = TRUE`, the function will return the following objects respectively:

- `samples_posterior` A $k \times \text{iter}$ matrix containing the posterior samples for each parameter (i.e., $k = (p/(p-1))/2$) at each iteration (i.e., `iter`) of the sampler.
- `centrality` A $p \times \text{iter}$ matrix containing the centrality of a node at each iteration of the sampler.

Examples

```
library(easybgm)
library(bgms)

data <- na.omit(Wenchuan)

# Fitting the Wenchuan PTSD data

fit <- easybgm(data, type = "continuous",
               iter = 1000 # for demonstration only (> 5e4 recommended)
               )

summary(fit)

# To extract the posterior parameter distribution
# and centrality measures

fit <- easybgm(data, type = "continuous",
               iter = 1000, # for demonstrative purposes, generally, 1e5 iterations are recommended
               save = TRUE,
               centrality = TRUE)
```

edgevidence

Edge evidence plot

Description

The edge evidence plot colors edges according to their hypothesis testing results: blue for included, red for excluded, and gray for inconclusive. This plot can be used to visualize the hypothesis testing results whether edge presence or absence. The edge evidence plot can aid researchers in deciding which edges provide robust inferential conclusions

Usage

```
plot_edgevidence(
  output,
  evidence_thresh = 10,
  split = FALSE,
  show = "all",
```

```
    ...
  )
```

Arguments

output	Output object from the easybgm function. Supports also objects from the bgm function of the bgms package.
evidence_thresh	Bayes Factor which will be considered sufficient evidence for in-/exclusion, default is 10.
split	if TRUE, plot is split in included and excluded edges. Note that by default separate plots are shown and appear after each other in the plot window. To show the plots side-by-side specify <code>par(mfrow = c(1, 2))</code> .
show	specifies which edges should be shown, indicated by "all", "included", "inconclusive", "excluded".
...	Additional arguments passed onto <code>qgraph</code> .

Value

Returns a plot

Examples

```
library(easybgm)
library(bgms)

data <- na.omit(Wenchuan)
fit <- easybgm(data, type = "continuous",
              iter = 1000 # for demonstration only (> 5e4 recommended)
              )

plot_edgeevidence(fit)

oldpar <- par(mfrow = c(1,1))

par(mfrow = c(1, 2))
plot_edgeevidence(fit, split = TRUE)

#' par(mfrow = c(1, 3))
plot_edgeevidence(fit, show = "included")
plot_edgeevidence(fit, show = "inconclusive")
plot_edgeevidence(fit, show = "excluded")

par(oldpar)
```


Description

Plots the 95% highest density interval of the posterior distribution of the parameter estimates. The plot can be used to visualize the uncertainty of the partial association estimates. The x-axis indicates the strength of the partial association. The y-axis indicates the edge between nodes i and j . The farther the posterior estimates (i.e., the points in the plot) are from zero, the stronger the partial association of the edge. The wider the highest density intervals (i.e., the error bar around the point), the less certain we are about the strength of the association.

Usage

```
plot_parameterHDI(output, ...)
```

Arguments

output	Output object from the <code>easybgm</code> function. Supports also objects from the <code>bgm</code> function of the <code>bgms</code> package.
...	Additional arguments passed onto <code>ggplot2</code>

Value

Returns a plot

Examples

```
library(easybgm)
library(bgms)

data <- na.omit(Wenchuan)
subdata <- data[1:50, 1:3] #for demonstration choosing a subset to increase computation speed
fit <- easybgm(subdata, type = "ordinal",
              iter = 1000, # for demonstration only (> 5e4 recommended)
              save = TRUE)
plot_parameterHDI(fit)
```

network

*Network plot***Description**

The network plot visualizes the strength of interactions between two nodes, the partial associations. Solely edges with a posterior inclusion probability larger than the `exc_prob` argument (default = 0.5) are shown. Edge thickness and saturation represent the strength of the association; the thicker the edge, the stronger the association. Red edges indicate negative relations and blue edges indicate positive associations.

Usage

```
plot_network(output, exc_prob = 0.5, evidence_thresh = 10, dashed = FALSE, ...)
```

Arguments

<code>output</code>	Output object from the <code>easybgm</code> function. Supports also objects from the <code>bgm</code> function of the <code>bgms</code> package.
<code>exc_prob</code>	The threshold for excluding edges. All edges with a lower inclusion probability will not be shown. The default is set to 0.5 in line with the median probability plot.
<code>evidence_thresh</code>	If <code>dashed = TRUE</code> , users can specify the threshold for sufficient evidence for inclusion. All edges with evidence lower than <code>evidence_tresh</code> are dashed.
<code>dashed</code>	A binary parameter indicating whether edges with inconclusive evidence should be dashed. Default is <code>FALSE</code>
<code>...</code>	Additional arguments passed onto <code>qgraph</code> .

Value

Returns a plot

Examples

```
library(easybgm)
library(bgms)

data <- na.omit(Wenchuan)
fit <- easybgm(data, type = "continuous",
               iter = 1000 # for demonstration only (> 5e4 recommended)
               )

plot_network(fit)
```

```
# Shows all edges with an inclusion probability larger than 0.1
plot_network(fit, exc_prob = 0.1)

# Indicate which edges have insufficient evidence for inclusion through a dashed line
plot_network(fit, dashed = TRUE, evidence_thresh = 10)
```

```
print.easybgm      Print method for easybgm objects
```

Description

Used to print easybgm results. The nicest overview is created by first feeding it to summary()

Usage

```
## S3 method for class 'easybgm'
print(x, ...)
```

Arguments

x	easybgm object
...	unused argument

Value

Prints the output of a Bayesian cross-sectional network model fitted with 'easybgm'

```
prior_sensitivity  Plot sensitivity to edge inclusion prior setting
```

Description

For a given list of easybgm outputs with different prior edge inclusion probabilities, the function plots the percentage of edges that are included, excluded, and inconclusive.

Usage

```
plot_prior_sensitivity(output, ...)
```

Arguments

output	A list of easybgm outputs with different prior edge inclusion probabilities
...	Additional arguments passed onto ggplot2.

Details

Prior sensitivity plot

Value

Returns a plot

Examples

```
#not run due to computation speed

library(easybgm)
library(bgms)

#data <- na.omit(Wenchuan)
#fit1 <- easybgm(data, type = "ordinal",
#               iter = 1000 # for demonstration only (> 5e4 recommended),
#               inclusion_probability = .1
#               )
#fit2 <- easybgm(data, type = "ordinal",
#               iter = 1000,
#               inclusion_probability = .5
#               )
#fit3 <- easybgm(data, type = "ordinal",
#               iter = 1000, inclusion_probability = .9)

#plot_prior_sensitivity(list(fit1, fit2, fit3))
```

structure

Structure plot

Description

The plot shows the resulting graph structure, i.e. all edges with some evidence of inclusion (i.e., inclusion Bayes factor greater than 1).

Usage

```
plot_structure(output, ...)
```

Arguments

output	Output object from the easybgm function. Supports also objects from the bgm function of the bgms package.
...	Additional arguments passed onto qgraph

Value

Returns a plot

Examples

```
library(easybgm)
library(bgms)

data <- na.omit(Wenchuan)
fit <- easybgm(data, type = "continuous",
               iter = 1000 # for demonstration only (> 5e4 recommended)
               )

plot_structure(fit)
```

structure_probs

Plot Posterior Structure Probabilities

Description

Plots the posterior structure probabilities of all visited structures, sorted from the most to the least probable.

Usage

```
plot_structure_probabilities(output, as_BF = FALSE, ...)
```

Arguments

output	Output object from the easybgm function. Supports also objects from the bgm function of the bgms package.
as_BF	If TRUE plots the y-axis as Bayes factors instead of posterior structure probability. Default is FALSE.
...	Additional arguments passed onto ggplot2

Value

Returns a plot

Examples

```

library(easybgm)
library(bgms)

data <- na.omit(Wenchuan)
subdata <- data[1:50, 1:3] #for demonstration choosing a subset to increase computation speed
fit <- easybgm(subdata, type = "ordinal", save = TRUE, edge_selection = TRUE,
              iter = 1000 # for demonstration only (> 5e4 recommended)
              )

plot_structure_probabilities(fit)

```

summary.easybgm

Summary method for easybgm objects

Description

Used to create a object of easybgm results and in turn print it

Usage

```

## S3 method for class 'easybgm'
summary(object, evidence_thresh = 10, ...)

```

Arguments

object	easybgm object
evidence_thresh	Bayes Factor which will be considered sufficient evidence for in-/exclusion, default is 10.
...	unused argument

Value

Creates and prints the output of a Bayesian cross-sectional network analysis. The summary output has four parts. The first part lists the package used, the number of variables, and the data type. The second part is a matrix of edge-specific information. Each edge is listed in a row. This row contains the posterior parameter estimate, the posterior inclusion probability, the inclusion Bayes factor, and the categorization of the edge. The category encodes whether an edge is included, excluded, or inconclusive based on the inclusion Bayes factor. Users can set the threshold for the Bayes factor classification with the evidence threshold. By default, the threshold is set to \$10\$. The third part of the summary provides aggregated edge information. It lists the number of included, excluded, and inconclusive edges in the network, as well as the number of possible edges. This gives the user a quick overview of the robustness and density of the network. The higher the number of conclusive

edges (i.e., classified as either included or excluded), the more robust the network. Conversely, if the network has a high percentage of inconclusive edges, the network is not robust. Researchers should refrain from making strong inferential conclusions. The final output section is a description of the structure uncertainty. It shows the number of structures visited, the number of possible structures, and the highest posterior structure probability. This last section can only be obtained for networks fitted with 'BDgraph' and 'bgms'.

Index

`bgm_extract`, 2

`bgm_fit`, 2

`centrality`, 3

`complexity_probs`, 4

`easybgm`, 4

`edgeevidence`, 7

`HDI`, 9

`network`, 10

`plot_centrality` (`centrality`), 3

`plot_complexity_probabilities`

(`complexity_probs`), 4

`plot_edgeevidence` (`edgeevidence`), 7

`plot_network` (`network`), 10

`plot_parameterHDI` (`HDI`), 9

`plot_prior_sensitivity`

(`prior_sensitivity`), 11

`plot_structure` (`structure`), 12

`plot_structure_probabilities`

(`structure_probs`), 13

`print.easybgm`, 11

`prior_sensitivity`, 11

`structure`, 12

`structure_probs`, 13

`summary.easybgm`, 14