

Package ‘nutriNetwork’

May 4, 2026

Type Package

Title Structure Learning with Copula Graphical Model

Version 0.2.1

Depends R (>= 3.1.0)

Imports Matrix, igraph, parallel, methods, glasso, huge, tmvtnorm

Description Statistical tool for learning the structure of direct associations among variables for continuous data, discrete data and mixed discrete-continuous data. The package is based on the copula graphical model in Behrouzi and Wit (2017) <[doi:10.1111/rssc.12287](https://doi.org/10.1111/rssc.12287)>.

License GPL-3

Date 2026-05-04

NeedsCompilation no

Repository CRAN

Date/Publication 2026-05-04 18:30:09 UTC

Author Pariya Behrouzi [aut, cre] (ORCID:
<<https://orcid.org/0000-0001-6762-5433>>)

Maintainer Pariya Behrouzi <pariya.behrouzi@gmail.com>

Contents

nutriNetwork-package	2
nutriNetwork	2
plot.nutriNetwork	4
plot.select	5
print.nutriNetwork	8
print.select	9
selectnet	10
vfit	12

Index	13
--------------	-----------

nutriNetwork-package *Undirected Network for nutrition multivariate data*

Description

Statistical tool for learning the structure of direct associations among variables for continuous data, discrete data and mixed discrete-continuous data. The package is based on the copula graphical model in Behrouzi and Wit (2017) <doi:10.1111/rssc.12287>.

Author(s)

Pariya Behrouzi
Maintainers: Pariya Behrouzi <pariya.behrouzi@gmail.com>

References

1. Behrouzi, P., and Wit, E. C. (2019). Detecting epistatic selection with partially observed genotype data by using copula graphical models. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 68(1), 141-160.
3. Behrouzi, P., and Wit, E. C. (2017c). netgwas: An R Package for Network-Based Genome-Wide Association Studies. arXiv preprint, arXiv:1710.01236.

nutriNetwork *Reconstructs conditional (in)dependence networks among variables*

Description

This is the main functions of the **nutriNetwork** package. This function infers the direct associations between variables. In another words, it measures pairwise associations among variables while correcting the effect of remaining variables. Three methods are available to reconstruct networks, namely (i) Gibbs sampling, (ii) approximation method, and (iii) nonparanormal approach within the copula graphical model. The first two methods are able to deal with missing genotypes. The last one is computationally faster.

Usage

```
nutriNetwork(data, method = "gibbs", rho = NULL, n.rho = NULL, rho.ratio = NULL,  
ncores = 1, em.iter = 5, em.tol=.001, verbose = TRUE)
```

Arguments

data	An ($n \times p$) matrix or a data.frame corresponding to the data matrix, where n is sample size and p is the number of variables. Input data can contain missing values.
method	Reconstructing undirected graph using the three methods: "gibbs", "approx", and "npn". For a medium (~500) and a large number of variables we recommend to choose "gibbs" and "approx", respectively. Choosing "npn" for a very large number of variables (> 2000) is computationally efficient. The default method is "gibbs".
rho	Optional. A decreasing sequence of non-negative numbers that control the sparsity level. Leaving the input as rho = NULL, the program automatically computes a sequence of rho based on n.rho and rho.ratio. Users can also supply a decreasing sequence values to override this.
n.rho	Optional. The number of regularization parameters. The default value is 10.
rho.ratio	Optional. Determines distance between the elements of rho sequence. A small value of rho.ratio results in a large distance between the elements of rho sequence. And a large value of rho.ratio results into a small distance between elements of rho. Optional. The default value is 0.3.
ncores	Optional. The number of cores to use for the calculations. Using ncores = "all" automatically detects number of available cores and runs the computations in parallel on (available cores - 1).
em.iter	Optional. The number of EM iterations. The default value is 5.
em.tol	Optional. A criteria to stop the EM iterations. The default value is .001.
verbose	Optional. Providing a detail message for tracing output. The default value is TRUE.

Details

This function estimates a graph path . To select an optimal graph please refer to [selectnet](#).

Value

An object with S3 class "nutriNetwork" is returned:

Theta	A list of estimated p by p precision matrices that show the conditional independence relationships patterns among measured items.
path	A list of estimated p by p adjacency matrices. This is the graph path corresponding to Theta.
ES	A list of estimated p by p conditional expectation corresponding to rho.
Z	A list of n by p transformed data based on Gaussian copula.
rho	A n .rho dimensional vector containing the penalty terms.
loglik	A n .rho dimensional vector containing the maximized log-likelihood values along the graph path.
data	The n by p input data matrix. The n by p transformed data in case of using "npn".

Author(s)

Pariya Behrouzi
 Maintainers: Pariya Behrouzi <pariya.behrouzi@gmail.com>

See Also

[selectnet](#)

Examples

```
##### toy example
data(vfit)
test_dat <- vfit[1:10, c("sex", "ani.pro", "veg.pro", "B6",
  "B12", "B9", "SPPB.total", "HandGrip" )]
out_test <- nutriNetwork(test_dat, method = "gibbs")
#####

out <- nutriNetwork(vfit, method = "gibbs")
sel <- selectnet(out)

cl <- c(rep("gray70", 7), rep("green3",17), rep("red3",5))
plot(sel, vis= "parcor.network", sign.edg = TRUE,
  vertex.color = cl, curve = TRUE, layout.tree= TRUE,
  root.node= c(26, 29), pos.legend= "bottomleft",
  cex.legend=1)
#diffeent visualization
plot(sel, vis= "parcor.network", sign.edg = TRUE, layout = NULL,
  vertex.color = cl, curve = TRUE, pos.legend= "topleft",
  cex.legend=1 )
```

plot.nutriNetwork *plot for S3 class "nutriNetwork"*

Description

Plot the graph path which is the output of the [nutriNetwork](#).

Usage

```
## S3 method for class 'nutriNetwork'
plot( x, n.memberships=NULL , ... )
```

Arguments

<code>x</code>	An object from "nutriNetwork" class.
<code>n.memberships</code>	A vector containing number of variables in each group. For example, the <code>vfit</code> dataset that is provided in the package contains 3 different groups, where the first 7 variables are general covariates (e.g. age, sex, BMI, and etc.), the next 17 variables belong to nutrient (e.g. vitamins B6, B12, C, D, and etc.), and the last 5 variables belong to physical performance and muscle strength. Thus, <code>n.memberships = c(7, 17, 5)</code> . If <code>n.memberships = NULL</code> , in the graph visualization all markers are represented same colour.
<code>...</code>	System reserved (No specific usage)

Author(s)

Pariya Behrouzi
 Maintainer: Pariya Behrouzi <pariya.behrouzi@gmail.com>

References

Behrouzi, P., and Wit, E. C. (2017c). `netgwas`: An R Package for Network-Based Genome-Wide Association Studies. arXiv preprint, arXiv:1710.01236.

Examples

```
data(vfit)
out <- nutriNetwork(vfit, method = "gibbs")
plot(out)
```

plot.select

Plot function for S3 class "select"

Description

Plot the optimal graph by model selection

Usage

```
## S3 method for class 'select'
plot(x, vis= NULL, xlab= NULL, ylab= NULL, n.mem= NULL,
      vertex.label= FALSE, ..., layout= NULL, label.vertex= "all",
      vertex.size= NULL, vertex.color= NULL, edge.color= "gray29", sel.nod.label= NULL,
      label.size = NULL, w.btw= 800, w.within = 10, sign.edg= TRUE, edge.width= NULL,
      edge.label= NULL, max.degree= NULL, layout.tree= NULL, root.node= NULL,
      degree.node= NULL, curve= FALSE, pos.legend= "bottomleft", cex.legend= 0.8,
      iter1 = NULL, temp = NULL, tk.width = NULL, tk.height= NULL)
```

Arguments

<code>x</code>	An object with S3 class "select"
<code>vis</code>	Visualizing the results as a graph (network) or as a matrix. There are 4 options to visualize the selected graph: (i) "CI": plotting conditional independence (CI) relationships between variables, (ii) "interactive": plotting the conditional independence network, where opens a new windows with interactive graph drawing facility, and (iii) "parcor.network": plots the estimated graph based on partial correlation values. (iv) "parcor.interactive": plots the estimated graph based on partial correlation matrix with an interactive graph drawing facility. Default is "CI". Also, there are 3 options to visualize the selected graph as a matrix: (i) <code>vis="image.parcorMatrix"</code> plots the image of partial correlation matrix, (ii) <code>vis="image.adj"</code> draws the adjacency matrix (only presence and absence of links), (iii) <code>vis="image.precision"</code> plots the selected precision matrix.
<code>xlab</code>	ONLY applicable when <code>vis="CI"</code> , <code>"image.parcorMatrix"</code> , <code>"image.adj"</code> , or <code>"image.precision"</code> .
<code>ylab</code>	ONLY applicable when <code>vis="CI"</code> , <code>"image.parcorMatrix"</code> , <code>"image.adj"</code> , or <code>"image.precision"</code> .
<code>n.mem</code>	A vector of memberships. For example, the <code>vfit</code> dataset that is provided in the package contains 3 different groups, where the first 7 variables are general covariates (e.g. age, sex, BMI, and etc.), the next 17 variables belong to nutrient (e.g. vitamins B6, B12, C, D, and etc.), and the last 5 variables belong to physical performance and muscle strength. Thus, <code>n.mem = c(7, 17, 5)</code> . If <code>n.mem = NULL</code> and <code>vis="CI"</code> all vertices are coloured the same.
<code>vertex.label</code>	ONLY applicable when <code>vis="CI"</code> . Assign names to the vertices. Default is FALSE.
<code>...</code>	ONLY applicable when <code>vis="CI"</code> . System reserved (No specific usage)
<code>layout</code>	ONLY applicable when <code>vis="interactive"</code> or <code>"parcor.network"</code> . The layout specification. Some graph layouts examples: <code>layout_with_fr</code> , <code>layout_in_circle</code> , <code>layout_as_tree</code> , and <code>layout.fruchterman.reingold</code> . The default layout is <code>layout_with_fr</code> .
<code>label.vertex</code>	ONLY applicable when <code>vis="interactive"</code> . Assign names to the vertices. There are three options: "none", "some", "all". Specify "none" to omit vertex labels in the graph; using <code>label.vertex="some"</code> you must provide a vector of vertex IDs or a single vertex ID to the <code>sel.label</code> argument, which you would like to be shown in the graph. Specify "all" to include all vertex labels in the graph. Default is "all".
<code>vertex.size</code>	Optional. The size of vertices in the graph visualization. The default value is 7.
<code>vertex.color</code>	ONLY applicable when <code>vis="interactive"</code> or <code>"parcor.network"</code> . Optional vector (or a color name) giving the colors of the vertices. The default is "red"
<code>edge.color</code>	ONLY applicable when <code>vis="interactive"</code> . Optional. The default is "gray".
<code>sel.nod.label</code>	ONLY applicable when <code>vis="interactive"</code> or <code>"parcor.network"</code> . A vector of vertex IDs or a single vertex ID, which you would like to be shown in the graph. ONLY applicable when <code>label.vertex="some"</code> .
<code>label.size</code>	ONLY applicable for <code>vis="interactive"</code> or <code>vis="parcor.network"</code> . The font size of the vertex labels.

w.btw	Distance between nodes from different memberships of n.mem in layout.
w.within	Distance of nodes within one membership of n.mem in layout.
sign.edg	Optional. ONLY applicable when vis= "parcor.network". If TRUE then edges are colored as red and blue, where red stands for positive and blue negative partial correlation values. If FALSE all edges are colored as gray. Default is TRUE.
edge.width	Optional. ONLY applicable when vis= "parcor.network". Based on the strength of partial correlation values, edges will shown with different line type. Default is FALSE.
edge.label	Optional. ONLY applicable when vis= "parcor.network". If TRUE then the partial correlation values will be shown on top of each edge. Default is FALSE.
max.degree	Optional. ONLY applicable when vis= "parcor.network". A number showing degree of a node. This can be used to print those vertex labels that the correspondence vertex have at least e.g. 1 degree.
layout.tree	Optional. ONLY applicable when vis= "parcor.network". If TRUE then it uses layout_as_tree from igraph package. Default is FALSE.
root.node	Optional. ONLY applicable when vis= "parcor.network". The index of the root vertex or root vertices. If this is a non-empty vector then the supplied vertex ids are used as the roots of the trees . If it is an empty vector, then the root vertices are automatically calculated based on topological sorting, performed with the opposite mode than the mode argument. After the vertices have been sorted, one is selected from each component.
degree.node	Optional. ONLY applicable when vis= "parcor.network". It is related to the vertex label degree. It controls the position of the labels with respect to the vertices. Value are for example $-\pi/2$, 0, $\pi/2$, π sets above, to the right, below, to the left of a node, respectively.
curve	Optional. ONLY applicable when vis= "parcor.network". Edge curvature, range between 0 and 1 (FALSE sets it to 0, TRUE to 0.5). Default is FALSE.
pos.legend	Applicable when vis= "parcor.network" or vis= "CI". The x and y co-ordinates to be used to position the legend. They can be specified by keywords like "topright", "topleft", and etc. Default is "bottomleft".
cex.legend	Applicable when vis= "parcor.network" or vis= "CI".
iter1	Optional. ONLY applicable when vis= "parcor.interactive". integer scalar, the number of iterations to perform for layout_with_fr layout.
temp	Optional. ONLY applicable when vis= "parcor.interactive". Real scalar, the start temperature for layout_with_fr layout.
tk.width	Optional. The size of the drawing area of interactive plot.
tk.height	Optional. The size of the drawing area of interactive plot.

Value

An object with S3 class "select" is returned:

network	Plot of a selected graph, when vis= "CI".
adjacency	Conditional independence (CI) relationships between variables, when vis= "CI"
network	Interactive plot of a selected graph with .eps format, when vis= "interactive"

Author(s)

Pariya Behrouzi
Maintainer: Pariya Behrouzi <pariya.behrouzi@gmail.com>

References

Behrouzi, P., and Wit, E. C. (2017c). netgwas: An R Package for Network-Based Genome-Wide Association Studies. arXiv preprint, arXiv:1710.01236.

Examples

```
data(vfit)
out <- nutriNetwork(vfit)
sel <- selectnet(out)
plot(sel, vis= "image.parcorMatrix")
```

`print.nutriNetwork` *Print function for S3 class "nutriNetwork"*

Description

Print a summary of results from function [nutriNetwork](#).

Usage

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

Arguments

<code>x</code>	An object with S3 class "nutriNetwork"
<code>...</code>	System reserved (No specific usage)

Author(s)

Pariya Behrouzi and Ernst C. Wit
Maintainer: Pariya Behrouzi <pariya.behrouzi@gmail.com>

See Also

[nutriNetwork](#)

Examples

```
data(vfit)
out <- nutriNetwork(vfit, method = "nbn"); out
```

print.select *Print function for S3 class "select"*

Description

Print function for [selectnet](#).

Usage

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

Arguments

x	An object with S3 class "select"
...	System reserved (No specific usage)

Author(s)

Pariya Behrouzi
Maintainer: Pariya Behrouzi <pariya.behrouzi@gmail.com>

References

Behrouzi, P., and Wit, E. C. (2017c). netgwas: An R Package for Network-Based Genome-Wide Association Studies. arXiv preprint, arXiv:1710.01236.

See Also

[selectnet](#)

Examples

```
data(vfit)
out <- nutriNetwork(vfit, method = "nbn")
sel <- selectnet(out)
#A p x p adjacency matrix
sel$opt.adj
```

 selectnet

Model selection for optimal graph estimation

Description

Estimate the optimal graph based on different information criteria .

Usage

```
selectnet(nutriNetwork.obj, opt.index= NULL, criteria= NULL, ebic.gamma=0.5,
          ncores= NULL, verbose= TRUE)
```

Arguments

nutriNetwork.obj	An object with S3 class "nutriNetwork"
opt.index	The program internally determines an optimal graph using opt.index= NULL. Otherwise, to manually choose an optimal graph from the graph path.
criteria	Model selection criteria. "ebic" and "aic" are available. BIC model selection can be calculated by fixing ebic.gamma = 0. Applicable only if opt.index= NULL.
ebic.gamma	The tuning parameter for ebic. Theebic.gamma = 0 results in bic model selection. The default value is 0.5. Applicable only opt.index= NULL.
ncores	The number of cores to use for the calculations. Using ncores = "all" automatically detects number of available cores and runs the computations in parallel.
verbose	If verbose = FALSE, printing information is disabled. The default value is TRUE. Applicable only opt.index= NULL.

Value

An obj with S3 class "selectnet" is returned:

opt.adj	The optimal graph selected from the graph path
opt.theta	The optimal precision matrix from the graph path
opt.sigma	The optimal covariance matrix from the graph path
ebic.scores	Extended BIC scores for regularization parameter selection at the EM convergence. Applicable if opt.index = NULL.
opt.index	The index of optimal regularization parameter.
opt.rho	The selected regularization parameter.
par.cor	A partial correlation matrix.

and anything else that is included in the input nutriNetwork.obj.

Author(s)

Pariya Behrouzi
 Maintainer: Pariya Behrouzi <pariya.behrouzi@gmail.com>

References

1. Behrouzi, P., and Wit, E. C. (2019). Detecting epistatic selection with partially observed genotype data by using copula graphical models. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 68(1), 141-160.
2. Behrouzi, P., and Wit, E. C. (2017c). netgwas: An R Package for Network-Based Genome-Wide Association Studies. arXiv preprint, arXiv:1710.01236.
3. Ibrahim, Joseph G., Hongtu Zhu, and Niansheng Tang. (2012). Model selection criteria for missing-data problems using the EM algorithm. *Journal of the American Statistical Association*.
4. D. Witten and J. Friedman. (2011). New insights and faster computations for the graphical lasso. *Journal of Computational and Graphical Statistics*, to appear.
5. J. Friedman, T. Hastie and R. Tibshirani. (2007). Sparse inverse covariance estimation with the lasso, *Biostatistics*.
6. Foygel, R. and M. Drton. (2010). Extended bayesian information criteria for Gaussian graphical models. In *Advances in Neural Information Processing Systems*, pp. 604-612.

Examples

```
##### toy example
data(vfit)
test_dat <- vfit[1:10, c("sex", "ani.pro", "veg.pro", "B6",
  "B12", "B9", "SPPB.total", "HandGrip" )]
out_test <- nutriNetwork(test_dat, method = "gibbs")
sel_test <- selectnet(out_test)
#####

out <- nutriNetwork(vfit, method = "gibbs")
sel <- selectnet(out)

cl <- c(rep("gray70", 7), rep("green3",17), rep("red3",5))
plot(sel, vis= "parcor.network", sign.edg = TRUE,
  vertex.color = cl, curve = TRUE, layout.tree= TRUE,
  root.node= c(26, 29), pos.legend= "bottomleft",
  cex.legend=1)
#diffeent visualization
plot(sel, vis= "parcor.network", sign.edg = TRUE, layout = NULL,
  vertex.color = cl, curve = TRUE, pos.legend= "topleft",
  cex.legend=1 )
```

`vfit`*Baseline data from VFIT study*

Description

A dietary study that includes dietary intake, physical performance, and muscle strength-related variables for 207 Dutch elderly people.

Usage

```
data(vfit)
```

Format

The format is a matrix containing 29 variables for 207 participants.

Details

Participants of the V-Fit trial were recruited via personal letters sent to senior residencies, home care organisations, general practitioners and local advertisements. Eligible participants were aged 70 y and older, used care services, did not regularly exercise, had a BMI of less than 25.

Source

Paw, M. J. C. A., de Jong, N., Schouten, E. G., Hiddink, G. J., & Kok, F. J. (2001). Physical exercise and/or enriched foods for functional improvement in frail, independently living elderly: a randomized controlled trial. *Archives of physical medicine and rehabilitation*, 82(6), 811-817.

Examples

```
data(vfit)
head(vfit, n=3)
```

Index

* **datasets**

vfit, [12](#)

* **package**

nutriNetwork-package, [2](#)

nutriNetwork, [2](#), [4](#), [8](#)

nutriNetwork-package, [2](#)

plot.nutriNetwork, [4](#)

plot.select, [5](#)

print.nutriNetwork, [8](#)

print.select, [9](#)

selectnet, [3](#), [4](#), [9](#), [10](#)

vfit, [12](#)