

Package ‘netDFI’

September 15, 2026

Type Package

Title Dynamic Fit Index for Network Models

Version 1.2.0

Description Implement methods to determine the dynamic fit index cutoffs for network models. The package allows users to evaluate model fit based on their own model statement, model type, and sample size. Methods are described in Du and Epskamp (2026) <[doi:10.31234/osf.io/5wj2y_v2](https://doi.org/10.31234/osf.io/5wj2y_v2)>.

License GPL (>= 3)

Encoding UTF-8

URL <https://github.com/xinkaidupsy/netDFI>

BugReports <https://github.com/xinkaidupsy/netDFI/issues>

Imports bootnet, dplyr, future, future.apply, progressr,
psychonetrics, qgraph, ggplot2, patchwork

Suggests psych

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Xinkai Du [aut, cre] (ORCID: <<https://orcid.org/0000-0002-4158-7878>>)

Maintainer Xinkai Du <xinkai.du.xd@gmail.com>

Repository CRAN

Date/Publication 2026-09-15 13:20:08 UTC

Contents

dfi_ggm	2
plot.dfi_ggm	4
Index	5

dfi_ggm

*Dynamic fit index cutoffs for Gaussian Graphical Model***Description**

This function determines the dynamic fit index cutoffs for Gaussian Graphical Models (GGM) following the algorithm developed in Du and Epskamp (2026).

Usage

```
dfi_ggm(
  net,
  specificity = 0.95,
  n_misspec = 3,
  iter = 500,
  n = 500,
  prop_pos = 0.8,
  ordinal = FALSE,
  n_levels = 4,
  skew_factor = 1,
  size_extra = c("manual", "beta_min"),
  manual_size = 0.2,
  type = c("uniform", "random"),
  missing = 0,
  ncores = 1,
  progressbar = TRUE
)
```

Arguments

net	The empirical network to estimate dynamic fit index cutoffs for; Input should be a matrix
specificity	Numeric. The specificity ($1 - \alpha$, the true-negative rate against the correctly-specified model) that the cutoff value aims for. Defaults to 0.95, i.e. cutoffs are placed so that 95% of correctly-specified simulated models pass.
n_misspec	Number of mis-specified model you want in the simulation. Default to 3, meaning there are five mis-specified models with 1, 2, ..., 5 extra edges respectively. Avoid setting too large numbers. Otherwise the simulation might fail or take too long.
iter	Integer. The number of iterations used in your simulation.
n	Integer. Sample size
prop_pos	Numeric data between 0-1. Decide the proportion of positive in the extra edges added to the empirical network to create the mis-specified network
ordinal	Logical; should ordinal data be generated?
n_levels	Integer; number of levels used in ordinal data.

skew_factor	Numeric; How skewed should ordinal data be? 1 indicates uniform data and higher values increase skewedness.
size_extra	Character; How to size the extra edges added to create misspecified models. One of "beta_min" or "manual" (default). "beta_min" uses the minimum detectable partial correlation from LASSO theory (Buhlmann & Van De Geer, 2011), $\sqrt{\log(p) / n} / \Theta_{ii}$, evaluated at the node the edge is added from. "manual" uses a fixed magnitude given by manual_size.
manual_size	Numeric in (0, 1); the absolute edge weight used when size_extra = "manual". Ignored when size_extra = "beta_min".
type	Should thresholds for ordinal data be sampled at random or determined uniformly?
missing	Proportion of data that should be simulated to be missing.
ncores	How many cores you want to use in the simulation. Recommend to leave one core free so that other tasks in the system are not impacted.
progressbar	Logical; if TRUE (default), show progress bars while fitting the misspecified and true models. Set to FALSE to suppress.

Value

An object of class dfi_ggm. Can use summary to view a summary of results

Author(s)

Xinkai Du <xinkai.du.xd@gmail.com>

References

Du, X., & Epskamp, S. (2026). Dynamical fit index cutoffs for Gaussian graphical models. *PsyArXiv*.
doi:10.31234/osf.io/5wj2y_v2

Examples

```
if (requireNamespace("psych", quietly = TRUE)) {
  library(psychonetrics)
  library(dplyr)

  # get the big five inventory data from psych
  data("bfi", package = "psych")

  # estimate ggm
  bfi_mod <- ggm(bfi) %>%
    prune() %>%
    runmodel()

  # obtain the partial correlation matrix
  bfi_net <- getmatrix(bfi_mod, "omega")

  # allow future_apply to use more memory
  options(future.globals.maxSize = 2 * 1024^3)
```

```
# run dfi
dfi_bfi <- dfi_ggm(
  bfi_net,
  ncores = 1,
  specificity = 0.80,
  iter = 200,
  n_misspec = 2
)
dfi_bfi

# plot results
p <- plot(dfi_bfi)
p[[1]]
}
```

plot.dfi_ggm

Plot functions

Description

Plot the fit distribution

Usage

```
## S3 method for class 'dfi_ggm'
plot(x, ...)
```

Arguments

x an object of class dfi_ggm
... further arguments passed from plot(); currently ignored.

Value

Plots

Index

`dfi_ggm`, [2](#)

`plot.dfi_ggm`, [4](#)