Package ‘lvnet’

October 13, 2022

Type Package
Title Latent Variable Network Modeling
Version 0.3.5
Author Sacha Epskamp
Maintainer Sacha Epskamp <mail@sachaepskamp.com>
Description Estimate, fit and compare Structural Equation Models (SEM) and network models (Gaussian Graphical Models; GGM) using OpenMx. Allows for two possible generalizations to include GGMs in SEM: GGMs can be used between latent variables (latent network modeling; LNM) or between residuals (residual network modeling; RNM). For details, see Epskamp, Rhemtulla and Borsboom (2017) <doi:10.1007/s11336-017-9557-x>.
License GPL-2
Imports glasso, qgraph, Matrix, psych, mvtnorm, parallel, corpcor, dplyr, methods, lavaan, semPlot
Depends R (>= 3.2.0), OpenMx
NeedsCompilation no
Repository CRAN
Date/Publication 2019-06-21 07:50:03 UTC

R topics documented:

EBIClvglasso ................................................. 2
lassoSelect .................................................. 3
lav2lvnet .................................................... 4
lvglasso ....................................................... 5
lvnet .......................................................... 6
lvnetCompare ............................................... 9
lvnetLasso ................................................... 10
lvnetRefit .................................................... 11
lvnetSearch .................................................. 12
plot.lvnet .................................................... 13
summary.lvnet ............................................... 14

Index 16
EBIClvglasso

Description

This function minimizes the Extended Bayesian Information Criterion (EBIC; Chen and Chen, 2008) to choose the lvglasso tuning parameter. See lvglasso

Usage

EBIClvglasso(S, n, nLatents, gamma = 0.5, nRho = 100, lambda, ...)

Arguments

S Sample variance-covariance matrix
n Sample Size
nLatents Number of latent variables
gamma EBIC hyper-parameter
nRho Number of tuning parameters to test
lambda The lambda argument containing factor loadings, only used for starting values!
... Arguments sent to lvglasso

Value

The optimal result of lvglasso, with two more elements:

rho The selected tuning parameter
ebic The optimal EBIC

Author(s)

Sacha Epskamp <mail@sachaepskamp.com>

References


See Also

lvglasso
Description

This function can be used to select a model using any fit index

Usage

lassoSelect(object, select, minimize = TRUE, refit = TRUE, lassoTol = 1e-04)

Arguments

object An lvnetLasso object
select A raw R expression using names used in the object$fitMeasures part of the output of lvnet
minimize Logical. Minimize or maximize?
refit Logical. Should the new best model be refitted.
lassoTol Tolerance for absolute values to be treated as zero in counting parameters.

Author(s)

Sacha Epskamp <mail@sachaepskamp.com>

Examples

## Not run:
# Load dataset:
library("lavaan")
data(HolzingerSwineford1939)
Data <- HolzingerSwineford1939[,7:15]

# Measurement model:
Lambda <- matrix(0, 9, 3)
Lambda[1:3,1] <- NA
Lambda[4:6,2] <- NA
Lambda[7:9,3] <- NA

# Search best fitting omega_theta:
res <- lvnetLasso(Data, "omega_theta", lambda = Lambda)
res$best
summary(res)

# Update to use EBIC:
resEBIC <- lassoSelect(res, ebic)
summary(resEBIC)

# Update to use minimal fitting model with RMSEA < 0.05:
resMinimal <- lassoSelect(res, df * (rmsea < 0.05), minimize = FALSE)
summary(resMinimal)

## End(Not run)

lav2lvnet  

Convert lavaan model to lvnet model matrices

Description
This function can be used to easily generate input matrices for lvnet based on a lavaan model.

Usage
lav2lvnet(model, data, std.lv = TRUE, lavaanifyOps = list(auto = TRUE, std.lv = std.lv))

Arguments
- **model**: Lavaan model syntax
- **data**: The dataset. Only used to extract order of variables names from the column-names.
- **std.lv**: Should the model be identified by constraining latent variable variance to 1. Defaults to TRUE unlike lavaan! This is because the starting values work better for this identification.
- **lavaanifyOps**: A list with other options sent to lavaanify

Value
A list with the model matrices for lambda, psi, theta and beta

Author(s)
Sacha Epskamp <mail@sachaepskamp.com>

Examples

## Not run:
library("lavaan")

# Load dataset:
data(HolzingerSwinford1939)
Data <- HolzingerSwinford1939[,7:15]

# lavaan model
HS.model <- '
visual  =~ x1 + x2 + x3
textual =~ x4 + x5 + x6
speed   =~ x7 + x8 + x9 '

# fit via lavaan:
lavFit <- cfa(HS.model, HolzingerSwineford1939[7:15], std.lv=TRUE)

# Fit via lvnet:
mod <- lav2lvnet(HS.model, HolzingerSwineford1939[7:15])
lvnetFit <- lvnet(Data, lambda = mod$lambda, psi = mod$psi)

# Compare:
Compare <- data.frame(lvnet = round(unlist(lvnetFit$fitMeasures)[c("npar","df","chisq","fmin","aic","bic", "rmsea","cfi","tli","nfi","logl")],3),
lavaan = round(fitMeasures(lavFit)[c("npar","df","chisq","fmin","aic","bic","rmsea", "cfi","tli","nfi","logl")],3))

Compare

## End(Not run)

---

### lvglasso

**Latent variable graphical LASSO**

#### Description

The lvglasso algorithm to estimate network structures containing latent variables, as proposed by Yuan (2012). Uses the glasso package (Friedman, Hastie and Tibshirani, 2014) and mimics input and output of the *glasso* function.

#### Usage

lvglasso(S, nLatents, rho = 0, thr = 1e-04, maxit = 10000, lambda)

#### Arguments

- **S**
  - Sample variance-covariance matrix
- **nLatents**
  - Number of latent variables.
- **rho**
  - The LASSO tuning parameter
- **thr**
  - The threshold to use for convergence
- **maxit**
  - Maximum number of iterations
- **lambda**
  - The lambda argument containing factor loadings, only used for starting values!

#### Value

A list of class lvglasso containing the following elements:

- **w**
  - The estimated variance-covariance matrix of both observed and latent variables
- **wi**
  - The estimated inverse variance-covariance matrix of both observed and latent variables
pcor  Estimated partial correlation matrix of both observed and latent variables
observed Logical vector indicating which elements of w, wi and pcor are observed
niter  The number of iterations used
lambda The estimated lambda matrix, when result is transformed to EFA model
theta  The estimated theta matrix
omega_theta The estimated omega_theta matrix
psi    The estimated psi matrix

Author(s)

Sacha Epskamp <mail@sachaepskamp.com>

References


Description

This function utilizes OpenMx (Boker et al., 2011, 2014) to confirmatory test latent variable network models between P manifests and M latents. See the details section for information about the modeling framework used. All the input matrices can be assigned R matrices with numbers indicating fixed values and NA indicating a value is free to estimate.

Usage

lvnet(data, lambda, beta, omega_theta, delta_theta, omega_psi, delta_psi, psi, theta, sampleSize, fitInd, fitSat, startValues = list(), scale = FALSE, nLatents, lasso = 0, lassoMatrix, lassoTol = 1e-4, ebicTuning = 0.5, mimic = c("lavaan","lvnet"), fitFunction = c("penalizedML","ML"), exogenous)

Arguments

data An N (sample size) x P matrix or data frame containing the raw data, or a P x P variance-covariance matrix.
lambda A P x M matrix indicating factor loadings. Defaults to a full NA P x M matrix if psi or omega_psi is not missing, or a P x 0 dummy matrix.
beta An M x M matrix indicating linear effects between latent variables. Defaults to an M x M matrix containing only zeroes.
omega_theta  A P x P matrix encoding the residual network structure. By default, theta is modeled instead.

delta_theta  A P x P diagonal scaling matrix. Defaults to NA on all diagonal elements. Only used if omega_theta is modeled.

omega_psi  An M x M matrix containing the latent network structure. By default, psi is modeled instead.

delta_psi  A diagonal M x M scaling matrix. Defaults to an identity matrix. Only used if omega_psi is modeled.

psi  An M x M variance-covariance matrix between latents and latent residuals. Defaults to a full NA matrix.

theta  A P x P variance-covariance matrix of residuals of the observed variables. Defaults to a diagonal matrix containing NAs.

sampleSize  The sample size, only used if data is assigned a variance-covariance matrix.

fitInd  The fit of the independence model. Used to speed up estimation fitting multiple models.

fitSat  The fit of the saturated model. Used to speed up estimation fitting multiple models.

startValues  An optional named list containing starting values of each model. e.g., list(lambda = matrix(1,9,3)) would set the starting values of a 10 x 3 lambda matrix to ones.

scale  Logical, should data be standardized before running lvnet?

nLatents  The number of latents. Allows for quick specification when lambda is missing. Not needed if lambda is assigned.

lasso  The LASSO tuning parameter.

lassoMatrix  Character vector indicating the names of matrices to apply LASSO regularization on. e.g., "omega_psi" or "omega_theta".

lassoTol  Tolerance for absolute values to be treated as zero in counting parameters.

ebicTuning  Tuning parameter used in extended Bayesian Information Criterion.

mimic  If set to "lavaan" (default), covariance matrix is rescaled and N is used rather than N - 1 in likelihood computation.

fitFunction  The fit function to be used. penalizedML will fit the penalized fit function and ML the maximum likelihood function.

exogenous  Numeric vector indicating which variables are exogenous.

Details

The modeling framework follows the all-y LISREL framework for Structural Equation Models (SEM; Hayduk, 1987) to model relationships between P observed variables and M latent variables:

\[
\Sigma = \lambda \cdot (I - B)^{-1} \psi (I - B)^{-1T} \cdot \lambda^T + \Theta
\]

Where Sigma is the P x P model-implied covariance matrix, \( \lambda \) a P x M matrix of factor loadings, \( B \) an M x M matrix containing regression effects between latent variables, \( \Psi \) a M x M covariance matrix of the latent variables/residuals and \( \Theta \) a P x P covariance matrix of residuals of the observed indicators.
The lvnet function allows for two extensions of this modeling framework. First, psi can be chosen to be modeled as follows:

\[ \psi = \delta_{\psi} (I - \omega_{\psi})^{(-1)} \delta_{\psi} \]

In which \( \delta_{\psi} \) is a \( M \times M \) diagonal scaling matrix and \( \omega_{\psi} \) a \( M \times M \) matrix containing zeroes on the diagonal and partial correlation coefficients on the offdiagonal values of two latent variables conditioned on all other latent variables. \( \omega_{\psi} \) therefore corresponds to a Gaussian Graphical Model, or a network structure.

Similarly, theta can be chosen to be modeled as follows:

\[ \theta = \delta_{\theta} (I - \omega_{\theta})^{(-1)} \delta_{\theta} \]

In which \( \delta_{\theta} \) is a \( P \times P \) diagonal scaling matrix and \( \omega_{\theta} \) a \( P \times P \) matrix containing zeroes on the diagonal and partial correlation coefficients on the offdiagonal values of two residuals conditioned on all other residuals.

Modeling \( \omega_{\psi} \) is termed Latent Network Modeling (LNM) and modeling \( \omega_{\theta} \) is termed Residual Network Modeling (RNM). lvnet automatically chooses the appropriate modeling framework based on the input.

Value

An lvnet object, which is a list containing the following elements:

- matrices: A list containing the estimated model matrices
- sampleStats: A list containing the covariance matrix (covMat) and sample size sampleSize
- mxResults: The OpenMx object of the fitted model
- fitMeasures: A named list containing the fit measures of the fitted model

Author(s)

Sacha Epskamp <mail@sachaepskamp.com>

References


See Also

lvnetSearch
Examples

# Load dataset:
library("lavaan")
data(HolzingerSwineford1939)
Data <- HolzingerSwineford1939[,7:15]

# Measurement model:
Lambda <- matrix(0, 9, 3)
Lambda[1:3,1] <- NA
Lambda[4:6,2] <- NA
Lambda[7:9,3] <- NA

# Fit CFA model:
CFA <- lvnet(Data, lambda = Lambda)

# Latent network:
Omega_psi <- matrix(c(0,NA,NA,
                      NA,0,0,
                      NA,0,0
                      ),3,3,byrow=TRUE)

# Fit model:
LNM <- lvnet(Data, lambda = Lambda, omega_psi=Omega_psi)

# Compare fit:
lvnetCompare(cfa=CFA,lnm=LNM)

# Summary:
summary(LNM)

# Plot latents:
plot(LNM, "factorStructure")

---

**lvnetCompare**

*Compare lvnet objects*

**Description**

Compares several results of lvnet

**Usage**

```r
lvnetCompare(...)
```

## S3 method for class 'lvnet'
anova(object, ...)

Arguments

object An lvnet object
... Any number of lvnet objects. Arguments can be named to make the resulting table named.

Author(s)

Sacha Epskamp <mail@sachaepskamp.com>

See Also

lvnet

Description

This function runs lvnet for a number of different tuning parameters, selects the best model based on some criterion and refits that model to obtain accurate parameter estimates. The lassoSelect function can afterwards be used to select a different model.

Usage

lvnetLasso(data, lassoMatrix, lassoTol = 1e-04, nTuning = 20, tuning.min = 0.01, tuning.max = 0.5, criterion = c("bic", "aic", "ebic"), verbose = TRUE, refitFinal = TRUE, refitAll = FALSE, nCores = 1, ...)

Arguments

data The data argument as used in lvnet
lassoMatrix Vector indicating the matrix or matrices to use in LASSO optimization
lassoTol Tolerance for absolute values to be treated as zero in counting parameters.
nTuning Number of tuning parameters to estimate.
tuning.min Minimal tuning parameter
tuning.max Maximal tuning parameter
criterion Criterion to use in model selection
verbose Should progress be printed to the console?
refitFinal Logical, should the best fitting model be refitted without LASSO regularization?
refitAll Logical, should *all* models be refitted without LASSO regularization (but with zeroes constrained) before evaluating fit criterium?
nCores Number of cores to use in parallel computing.
... Arguments sent to lvnet
Author(s)
Sacha Epskamp <mail@sachaepskamp.com>

Examples

# Load dataset:
library("lavaan")
data(HolzingerSwineford1939)
Data <- HolzingerSwineford1939[,7:15]

# Measurement model:
Lambda <- matrix(0, 9, 3)
Lambda[1:3,1] <- NA
Lambda[4:6,2] <- NA
Lambda[7:9,3] <- NA

# Search best fitting omega_theta:
## Not run:
res <- lvnetLasso(Data, "omega_theta", lambda = Lambda)
res$best
summary(res)

## End(Not run)

lvnetRefit

Refit lvnet model to new data

Description
Obtain fit indices from the estimated model parameters on a new dataset.

Usage
lvnetRefit(lvnetObject, data, sampleSize)

Arguments

lvnetObject Output of lvnet.
data New dataset or variance-covariance matrix.
sampleSize Sample size (if data is a variance-covariance matrix).

Author(s)
Sacha Epskamp <mail@sachaepskamp.com>
lvnetSearch

Step-wise exploratory search for optimal fitting model

Description

Performs stepwise search to optimize the structure of omega_theta, omega_psi, theta or psi. Starts at empty or full structure and iteratively adds or removes edges to optimize the criterion.

Usage

```r
lvnetSearch(data, matrix = c("omega_theta", "omega_psi", "theta", "psi"),
criterion = c("bic", "ebic","chisq","aic"),
start = c("default","empty","full"), alpha = 0.05, lambda, sampleSize,
maxIter, nCores = 1, maxChange = 1, ..., verbose = TRUE, file,
startValues = list())
```

Arguments

data The data argument as used in `lvnet`

matrix Character string indicating the matrix to be optimized. Can be "omega_theta", "omega_psi", "theta" and "psi".

criterion Character string indicating the criterion to be used. "AIC" and "BIC" optimize the AIC or BIC respectively, and "chisq" performs chi-square tests to see if adding an edge significantly improves model fit or removing an edges does not significantly reduce model fit.

start A character string indicating the structure of the matrix at the start of the algorithm. "empty" starts with a matrix with only zeroes and "full" starts with a matrix in which all elements are free to estimate. "lvglasso" employs the lvglasso algorithm (EBIC1vglasso) to find a starting structure for omega_theta and "glasso" employs the glasso algorithm to find a starting point for omega_psi (EBICglasso). "default" will lead to a full matrix if omega_psi or psi is optimized, and an empty matrix if omega_theta or theta is optimized.

alpha The alpha level for chi-square significance testing.

lambda The lambda argument as used in `lvnet`

sampleSize The sample size, only used if `data` is a covariance matrix.

maxIter The maximum number of edges to test. Defaults to M(M-1)/2

nCores Number of cores to use in parallel estimation.

maxChange Set to higher than one to change multiple edges in each run. Each iteration, maxChange is reset to max(number of changed edges - 1, 1). Can result in unstable results when searching "omega_theta".

... Arguments sent to `lvnet`

verbose Logical if progress should be printed to the console.

file An optional character string containing a file name to store temporary results in.

startValues A list containing start values as used in `lvnet`
**Value**

An object of class `lvnetSearch`, which is a list containing:

- **best**: The `lvnet` object of the best fitting model
- **modList**: A list containing the chain of fitted models
- **niter**: The number of iterations used

**Author(s)**

Sacha Epskamp <mail@sachaepskamp.com>

**See Also**

`lvnet`

**Examples**

```r
# Load dataset:
library("lavaan")
data(HolzingerSwineford1939)
Data <- HolzingerSwineford1939[,7:15]

# Measurement model:
Lambda <- matrix(0, 9, 3)
Lambda[1:3,1] <- NA
Lambda[4:6,2] <- NA
Lambda[7:9,3] <- NA

# Search best fitting omega_psi:
## Not run:
res <- lvnetSearch(Data, "omega_psi", lambda = Lambda)
res$best
## End(Not run)
```

---

**plot.lvnet**

Plot model matrices

**Description**

Plot method for `lvnet`. For `lvnetSearch` and `lvnetLasso` objects this is simply defined as `plot(object$best, ...)`


## Usage

```r
## S3 method for class 'lvnet'
plot(x, what = c("factorStructure", "residual", "latent"), partial,
     layout = "circle", ...)
## S3 method for class 'lvnetLasso'
plot(x, ...)
## S3 method for class 'lvnetSearch'
plot(x, ...)
```

### Arguments

- `x`: An `lvnet` object.
- `what`: What to plot? "factorStructure" plots the factor loadings and latent correlations or network. "residual" the residual correlations or network and "latent" the latent correlations or network.
- `partial`: Plot partial correlations instead of correlations? Defaults to `TRUE` if `omega_psi` or `omega_theta` is estimated.
- `layout`: The layout argument as used in `qgraph`
- `...`: Arguments sent to `qgraph`

### Author(s)

Sacha Epskamp <mail@sachaepskamp.com>

---

**summary.lvnet**

**Summary method for lvnet**

### Description

Plot method for `lvnet`. For `lvnetSearch` and `lvnetLasso` objects this is simply defined as `summary(object$best, ...)`

### Usage

```r
## S3 method for class 'lvnet'
summary(object, include = c("input", "chisq", "infcrit", "fitindices",
                           "rmsea", "parests"), digits = 3, ...)
## S3 method for class 'lvnetLasso'
summary(object, ...)
## S3 method for class 'lvnetSearch'
summary(object, ...)
```
Arguments

object    An lvnet object
include   Vector indicating what to include? "input" for the input used, "chisq" for the chi-square fit, "infcrit" for information criteria, "fitindices" for fit indices, "rmsea" for the RMSEA, ans "parests" for parameter estimates.
digits   Number of digits to round to.
...     Not used.

Author(s)

Sacha Epskamp <mail@sachaepskamp.com>
Index

anova.lvnet(lvnetCompare), 9

EBICglasso, 12
EBIClvglasso, 2, 12

glasso, 5

lassoSelect, 3, 10
lav2lvnet, 4
lavaanify, 4
lvglasso, 2, 5
lvnet, 6, 9–13
lvnetCompare, 9
lvnetLasso, 10
lvnetRefit, 11
lvnetSearch, 8, 12

plot.lvnet, 13
plot.lvnetLasso(plot.lvnet), 13
plot.lvnetSearch(plot.lvnet), 13

qgraph, 14

summary.lvnet, 14
summary.lvnetLasso(summary.lvnet), 14
summary.lvnetSearch(summary.lvnet), 14