Package 'bmemLavaan'

May 7, 2026

Type Package

Title Mediation Analysis with Missing Data and Non-Normal Data

Version 0.7

Date 2026-02-02

Depends R (>= 3.5.0), Amelia, MASS, snowfall, rsem

Imports lavaan, sem

Description Methods for mediation analysis with missing data and non-normal data are implemented. For missing data, four methods are available: Listwise deletion, Pairwise deletion, Multiple imputation, and Two Stage Maximum Likelihood algorithm. For MI and TS-ML, auxiliary variables can be included to handle missing data. For handling non-normal data, bootstrap and two-stage robust methods can be used. Technical details of the methods can be found in Zhang and Wang (2013, <doi:10.1007/s11336-012-9301-5>), Zhang (2014, <doi:10.3758/s13428-013-0424-0>), and Yuan and Zhang (2012, <doi:10.1007/s11336-012-9282-4>).

License GPL-2

LazyLoad yes

LazyData true

URL https://bigdatalab.nd.edu

ZipData no

NeedsCompilation no

Suggests R.rsp

VignetteBuilder R.rsp

Author Shuigen Ming [aut],
Hong Zhang [aut],
Zhiyong Zhang [aut, cre],
Lijuan Wang [aut]

Maintainer Zhiyong Zhang <johnnyzhz@gmail.com>

Repository CRAN

Date/Publication 2026-02-03 23:00:27 UTC

Contents

bmem 2
Math 4
power.bmem 5
power.curve 7
summary.bmem 8
summary.power 9

Index **11**

---

bmem *Mediation analysis based on bootstrap*

Description

Mediation analysis based on bootstrap

Usage

```
bmem(data, model, v, method='list', ci='perc', cl=.95,
     boot=1000, m=10, varphi=.1, st='i', robust=FALSE,
     max_it=500, parallel=FALSE, ncore=1,  ...)
```

Arguments

`data`	A data set
`model`	RAM path for the mediaiton model
`v`	Indices of variables used in the mediation model. If omitted, all variables are used.
`method`	`list`: listwise deletion, `pair`: pairwise deletion, `mi`: multiple imputation, `em`: EM algorithm.
`ci`	`norm`: normal approximation CI, `perc`: percentile CI, `bc`: bias-corrected CI, `bca`: BCa
`cl`	Confidence level. Can be a vector.
`boot`	Number of bootstraps
`m`	Number of imputations
`varphi`	Percent of data to be downweighted in robust method
`st`	Starting values
`robust`	Whether to use roubst method
`max_it`	Maximum number of iterations in EM
`parallel`	Whether to use parallel method to calculate.
`ncore`	Number of cores for parallel method.
`...`	Other options for `sem` function can be used.

Details

The indirect effect can be specified using equations such as `a*b`, `a*b+c`, and `a*b*c+d*e+f`, which can be defined in 'model' parameter.

Value

The on-screen output includes the parameter estimates, bootstrap standard errors, and CIs.

Author(s)

Zhiyong Zhang, Shuigen Ming and Lijuan Wang

References

Zhang, Z., & Wang, L. (2013). Methods for mediation analysis with missing data. Psychometrika, 78(1), 154-184. doi:10.1007/s1133601293015

Yuan, KH., Zhang, Z. Robust Structural Equation Modeling with Missing Data and Auxiliary Variables. Psychometrika 77, 803-826 (2012). doi:10.1007/s1133601292824

Examples

```
data("PoliticalDemocracy")

model_l <- '
ind60 =~ x1 + g*x2 + h*x3
dem60 =~ y1 + d*y2 + e*y3 + f*y4
dem65 =~ y5 + d*y6 + e*y7 + f*y8

dem60 ~ a * ind60
dem65 ~ c * ind60 + b * dem60

y1 ~~ y5
y2 ~~ y4 + y6
y3 ~~ y7
y4 ~~ y8
y6 ~~ y8
ind := a*b
'

fit_l <- bmem(data=PoliticalDemocracy, model = model_l, method='list',
      ci='perc', boot=50, parallel = TRUE, ncore = 8)
summary(fit_l)
```

`Math` *Parents' education levels and adolescent mathematics achievement of 76 families in 1986*

Description

A dataset containing the mothers' education level, children's mathematical achievement and other attributes of 76 families.

Usage

```
data("Math")
```

Format

A data frame with 76 observations on the following 4 variables.

ME mothers' education level

HE home environment

MATH children's mathematical achievement

READ reading recognition ability

Details

Data used in this example are randomly sampled from the National Longitudinal Survey of Youth, the 1979 cohort, which were collected in 1986.

Source

https://www.nlsinfo.org/content/cohorts/nlsy79-children/using-and-understanding-the-data/childyoung-adult-documentation

References

Center for Human Resource Research (2006). NLSY79 child & young adult data users guide: a guide to the 1986-2004 child data (Computer software manual). Columbus.

Examples

```
data(Math)
```

power.bmem *Conducting power analysis*

Description

A comprehensive power analysis function, it can conduct power analysis based on normal, bootstrap and robust Huber-type confidence intervals.

Usage

```
power.bmem(model, method="normal", nobs = 100, nrep = 1000, nboot = 1000,
alpha = 0.95, skewness = NULL, kurtosis = NULL, ovnames = NULL,
ci='perc', boot.type='default',
se = "default", estimator = "default", parallel = FALSE,
ncore = 1, verbose=TRUE, ...)
```

Arguments

model	A model specified using lavaan notation and above. See model.syntax for basic model specification. For the power analysis, the population parameter values should be provided in the following way. For example, the coefficient between math and HE is .39. Then it is specified as start(.39). If the parameter will be referred in the mediation effect, a label should be given as a modifier as b*HE+start(.39)*HE. It also specify the indirect or other composite effects using lavaan notation. model<-' math ~ c*ME+start(0)*ME + b*HE+start(.39)*HE HE ~ a*ME+start(.39)*ME ab := a*b abc := a*b + c '
method	Type of confidence intervals based on. Must be "normal", "boot" or "robust", which correspond to the normal, bootstrap or robust Huber-type confidence interval, respectively.
nobs	Number of observations for power analysis. If it is a vector, multiple group analysis will be conducted.
nrep	Number of replications for Monte Carlo simulation. At least 1,000 is recommended.
nboot	Number of bootstrap replicates. It's only required when bootstrap method is used.
alpha	The alpha level is used to obtain the confidence interval for model parameters.
skewness	A vector to give the skewness for the observed variables.
kurtosis	A vector to give the kurtosis for the observed variables.
ovnames	A vector to give the variable names for the observed variables. This is only needed when the skewness and kurtosis are provided. The skewness, kurtosis and variable names should be in the same order.
se	How to calculate the standard error, for example, robust standard error can be specified using se="robust".

estimator	Estimation methods to be used here.
parallel	Whether to use parallel method to calculate.
ncore	Number of cores to be used in parallel.
ci	Type of bootstrap confidence intervals. By default, the percentile one is used. otherwise get the bias-corrected one. It's only required when bootstrap method is used.
boot.type	Type of bootstrap method. By default, the nonparametric one is used. Changing it to "BS" to use the Bollen-Stine method. It's only required when bootstrap method is used.
verbose	Whether to print power information.
...	Other named arguments for lavaan can be passed here.

Value

power	power for all parameters and required ones in the model
coverage	coverage probability
pop.value	Population parameter values
results	A list to give all intermediate results
data	The last data set generated for checking purpose

Author(s)

Zhiyong Zhang, Shuigen Ming and Lijuan Wang

References

Zhang, Z. Monte Carlo based statistical power analysis for mediation models: methods and software. Behav Res 46, 1184-1198 (2014). doi:10.3758/s1342801304240

Examples

```
ex1model<-'
math ~ c*ME + start(0)*ME + b*HE + start(0.39)*HE
HE ~ a*ME + start(0.39)*ME
ab := a*b
'

N <- 50

system.time(power_normal <- power.bmem(ex1model, method = "normal", nobs = N,
  nrep=100, parallel=TRUE, skewness=c(-.3, -.7, 1.3),
  kurtosis=c(1.5, 0, 5), ovnames=c('ME', 'HE', 'math'), ncore=8))
summary(power_normal)
```

power.curve *Generate a power curve*

Description

Generate a power curve either based on Sobel test or bootstrap

Usage

```
power.curve(model, nobs=seq(100, 2000, 200), method='normal', nrep=1000,
nboot=1000, alpha=.95, skewness=NULL, kurtosis=NULL, ovnames=NULL,
ci='perc', boot.type='default',
se="default", estimator="default", parallel=FALSE,
ncore=1, interactive=TRUE, ...)
```

Arguments

model	A model specified using lavaan notation and above. See model.syntax for basic model specification. For the power analysis, the population parameter values should be provided in the following way. For example, the coefficient between math and HE is .39. Then it is specified as start(.39). If the parameter will be referred in the mediation effect, a label should be given as a modifier as b*HE+start(.39)*HE. It also specify the indirect or other composite effects using lavaan notation. model<-' math ~ c*ME+start(0)*ME + b*HE+start(.39)*HE HE ~ a*ME+start(.39)*ME ab := a*b abc := a*b + c '
method	Type of confidence intervals based on. Must be "normal", "boot" or "robust", which correspond to the normal, bootstrap or robust Huber-type confidence interval, respectively.
nobs	Number of observations for power analysis. It is typically should be a vector for single group analysis. For multiple group analysis, it should be a matrix.
nrep	Number of replications for Monte Carlo simulation. At least 1,000 is recommended.
nboot	Number of bootstraps to conduct.
alpha	The alpha level is used to obtain the confidence interval for model parameters.
skewness	A vector to give the skewness for the observed variables.
kurtosis	A vector to give the kurtosis for the observed variables.
ovnames	A vector to give the variable names for the observed variables. This is only needed when the skewness and kurtosis are provided. The skewness, kurtosis and variable names should be in the same order.
se	How to calculate the standard error, for example, robust standard error can be specified using se="robust".
estimator	Estimation methods to be used here.

parallel	Parallel methods, snow or multicore, can be used here.
ncore	Number of cores to be used in parallel. By default, the maximum number of cores are used.
ci	Type of bootstrap confidence intervals. By default, the percentile one is used. To get the bias-corrected one, use ci='BC'
boot.type	Type of bootstrap method. By default, the nonparametric one is used. Changing it to "BS" to use the Bollen-Stine method.
interactive	Whether to get the figure interactively.
...	Other named arguments for lavaan can be passed here.

Value

Generate the nobs-power curves for all relationships given in the model.

Examples

```
ex1model<-'
math ~ c*ME + start(0)*ME + b*HE + start(0.39)*HE
HE ~ a*ME + start(0.39)*ME
ab := a*b
'

nobs <- seq(50, 200, by=50)

power.curve(model=ex1model, nobs=nobs, method='normal',
nrep = 100, parallel=TRUE, ncore=8)
```

---

summary.bmem *Sumarize the results of function 'bmem'*

Description

Sumarize the results of function 'bmem'

Usage

```
## S3 method for class 'bmem'
summary(object, estimates=TRUE,...)
```

Arguments

object	An output object from the function bmem
estimates	Whether output a more detailed results of parameters and values of statistics
...	other options can be used for the generic summary function.

Details

The other type of confidence intervals can be constructed from the output of the function bmem. Note if the BCa is required, the ci='BCa' should have been specified in the function bmem.

Value

The on-screen output includes the parameter estimates, bootstrap standard errors, and CIs.

Examples

```
data("PoliticalDemocracy")

model_l <- '
ind60 =~ x1 + g*x2 + h*x3
dem60 =~ y1 + d*y2 + e*y3 + f*y4
dem65 =~ y5 + d*y6 + e*y7 + f*y8

dem60 ~ a * ind60
dem65 ~ c * ind60 + b * dem60

y1 ~~ y5
y2 ~~ y4 + y6
y3 ~~ y7
y4 ~~ y8
y6 ~~ y8
ind := a*b'

fit_l <- bmem(data=PoliticalDemocracy, model = model_l, method='list',
    ci='perc', boot=30, parallel = TRUE, ncore = 8)
summary.bmem(fit_l)
```

---

summary.power *Organize the results into a table*

Description

This function is adapted from the lavaan summary function to put the results in a table.

Usage

```
## S3 method for class 'power'
summary(object,...)
```

Arguments

object	Output from the function either power.bmem.
...	Other options

Value

The on-screen output includes the basic information of this power analysis, parameters' true values, parameter estimates, average bootstrap standard error, standard deviation of the parameter estimates, powers, standard error of the estimated powers and empirical coverage probability of the constructed CIs.

Examples

```
ex1model<-'
math ~ c*ME + start(0)*ME + b*HE + start(0.39)*HE
HE ~ a*ME + start(0.39)*ME
ab := a*b
'

N <- 50

system.time(power_robust <- power.bmem(ex1model, method = "robust", nobs = N,
        nrep=100, parallel=TRUE, ncore=8))
summary.power(power_robust)
```

Index

∗ **datasets**
Math, 4

bmem, 2, *8*, *9*

lavaan, *9*

Math, 4
model.syntax, *5*, *7*

power.bmem, 5, *9*
power.curve, 7

sem, *2*
summary *(*summary.bmem*)*, 8
summary.bmem, 8
summary.power, 9