Package 'rsem'

August 26, 2023

Type Package

Title Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

Version 0.5.1

Date 2023-08-26

Author Ke-Hai Yuan and Zhiyong Zhang

Maintainer Zhiyong Zhang <zhiyongzhang@nd.edu>

Depends R (>= 2.7), MASS, lavaan

Description

A robust procedure is implemented to estimate means and covariance matrix of multiple variables with missing data using Huber weight and then to estimate a structural equation model.

License GPL-2

URL https://bigdatalab.nd.edu

ZipData no

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2023-08-26 20:40:02 UTC

R topics documented:

`rsem-package` *Robust Structural Equation Modeling with Missing Data and Auxiliary Variables*

Description

A robust procedure is implemented to estimate means and covariance matrix of multiple variables with missing data using Huber weight and then to estimate a structural equation model.

Details

Package:	rsem
Type:	Package
License:	GPL-2
LazyLoad:	yes

Author(s)

Ke-Hai Yuan and Zhiyong Zhang Maintainer: Zhiyong Zhang <zhiyongzhang@nd.edu>

References

Yuan, K.-H., & Zhang, Z. (2012). Robust structural equation modeling with missing data and auxiliary variables. Psychometrika, 77(4), 803-826. https://doi.org/10.1007/s11336-012-9282-4

mardiamv25 *Simulated data*

Description

mardiamv25: Original data

mardiamv25_contaminated: Contaminated data with outliers

Usage

```
data(mardiamv25)
data(mardiamv25_contaminated)
```

rsem *The main function for robust SEM analysis*

Description

This is the function to carry out all analysis.

Usage

```
rsem(dset, select, EQSmodel, moment=TRUE, varphi=.1, st='i', max.it=1000,
eqsdata='data.txt', eqsweight='weight.txt', EQSpgm="C:/Progra~1/EQS61/WINEQS.EXE",
serial="1234")
```

Arguments

dset	A data matrix or a data frame
select	Variables to be seleted for SEM analysis. If omitted, all variables in the data set will be used.
moment	With mean structure. For covariance only, set moment=FALSE.
EQSmodel	The input file for EQS. If omitted, only the first-stage analysis will be conducted.
varphi	Proportion of data to be down-weighted. Default is 0.1.
max.it	Maximum number of iterations for EM. Default is 1000
st	Starting values for EM algorithm. The default is 0 for mean and I for covariance. Alternative, the starting values can be estimated according to MCD.
eqsdata	Data file name used in EQS
eqsweight	File name for weight matrix
EQSpgm	The path to the installed EQS program
serial	The serial no of EQS

Details

This function will run the robust analysis and output results.

Value

If `EQSmodel` is not supplied

`sem`	Information for SEM analysis including estimated means, covariance matrix and their sandwich type covariance matrix in the order of mean first and then covariance matrix.
`misinfo`	Information related to missing data pattern
`em`	Results from expectation robust algorithm
`ascov`	Covariance matrix

If `EQSmodel` is supplied,

`sem`	Information for SEM analysis including estimated means, covariance matrix and their sandwich type covariance matrix according to the requirement of EQS.

In addition, the following model parameters are from EQS

`fit.stat`	Fit indices and associated p-values
`para`	Parameter estimates
`eqs`	All information from REQS

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

See Also

`rsem.pattern`, `rsem.emmusig`, `rsem.Ascov`

Examples

```
## Not run:
## an example
 ## to use eqs, first load the package semdiag
 library(semdiag)
 data(mardiamv25)
 analysis<-rsem(mardiamv25, c(1,2,4,5), 'eqsinput.eqs')

## End(Not run)
```

`rsem.Ascov` *Sandwich-type covariance matrix*

Description

Returns the sandwich type covariance matrix. This function is not intended to use seperately from the `rsem.emmusig` function.

Usage

```
rsem.Ascov(xpattern, musig, varphi=.1)
```

Arguments

`xpattern`	Missing data pattern output from `rsem.pattern`.
`musig`	Robust mean and covariance matrix from `rsem.emmusig`
`varphi`	Proportion of data to be down-weighted. Default is 0.1.

Details

Data should be a matrix. To change a data frame to a matrix, using data.matrix(x).

Value

`Abeta`	A matrix
`Bbeta`	B matrix
`Gamma`	Sandwich type covariance matrix

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

See Also

`rsem.emmusig`

Examples

```
#dset<-read.table('MardiaMV25.dat.txt', na.string='-99')
#dset<-data.matrix(dset)
#n<-dim(dset)[1]
#p<-dim(dset)[2]
#miss_pattern<-rsem.pattern(n,p,dset)
#misinfo<-miss_pattern$misinfo
#V_forana<-c(1,2,4,5)
#em_results<-rsem.emmusig(dset,misinfo)
#hmu1<-em_results$mu
#hsigma1<-em_results$sigma
#rsem.Ascov(x, hmu1, hsigma1)
```

---

rsem.DP *Generate a duplication matrix*

Description

Generate a duplication matrix

Usage

```
rsem.DP(x)
```

Arguments

x	A matrix

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

Examples

```
x<-array(1:6, c(2,3))
rsem.DP(x)
```

`rsem.emmusig` *Robust mean and covariance matrix using Huber-type weight*

Description

Robust mean and covariance matrix using Huber-type weight.

Usage

```
rsem.emmusig(xpattern, varphi=.1, max.it=1000, st='i')
```

Arguments

`xpattern`	Missing data pattern output from `rsem.pattern`.
`varphi`	Proportion of data to be down-weighted. Default is 0.1.
`max.it`	Maximum number of iterations for EM. Default is 1000
`st`	Starting values for EM algorithm. The default is 0 for mean and I for covariance. Alternative, the starting values can be estimated according to MCD.

Details

Estimate mean and covariance matrix using the expectation robust (ER) algorithm.

Value

`err`	Error code. 0: good. 1: maximum iterations are exceeded.
`mu`	Mean vector
`sigma`	Covariance matrix

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

See Also

`rsem.emmusig`

Examples

```
#dset<-read.table('MardiaMV25.dat.txt', na.string='-99')
#dset<-data.matrix(dset)
#n<-dim(dset)[1]
#p<-dim(dset)[2]
#miss_pattern<-rsem.pattern(n,p,dset)
#misinfo<-miss_pattern$misinfo
#V_forana<-c(1,2,4,5)
#em_results<-rsem.emmusig(dset,misinfo)
#em_results
```

`rsem.fit` *Calculate robust test statistics*

Description

Calculate robust test statistics

Usage

```
rsem.fit(object, gamma, musig)
```

Arguments

`object`	Output from lavaan analysis, such as growth, factor, sem functions.
`gamma`	Robust covariance matrix for saturated mean and covariances
`musig`	Robust saturated mean and covariances

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

Examples

```
x<-array(1:6, c(2,3))
rsem.vec(x)
```

rsem.gname *Internal function*

Description

Internal function

Usage

```
rsem.gname(name)
```

Arguments

name	Variable names.

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

rsem.index *rsem.index function*

Description

To be added

Usage

```
rsem.index(p, oj)
```

Arguments

p	number of variables
oj	observed variables

rsem.indexv *rsem.indexv function*

Description

Internal function.

Usage

```
rsem.indexv(p, select)
```

Arguments

p	number of variables
select	variables to be used

rsem.indexvc *rsem.indexvc function*

Description

Internal function.

Usage

```
rsem.indexvc(p, select)
```

Arguments

p	number of variables
select	variables to be used

rsem.lavaan *Conduct robust SEM analysis using lavaan*

Description

Conduct robust SEM analysis using lavaan

Usage

```
rsem.lavaan(dset, model, select,  varphi=.1, max.it=1000)
```

Arguments

dset	A data matrix or a data frame
select	Variables to be seleted for SEM analysis. If omitted, all variables in the data set will be used.
model	The model using lavaan syntax
varphi	Proportion of data to be down-weighted. Default is 0.1.
max.it	Maximum number of iterations for EM. Default is 1000

Details

This function will run the robust analysis and output results.

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Yuan, K.-H., & Zhang, Z. (2012). Robust Structural Equation Modeling with Missing Data and Auxiliary Variables. Psychometrika, 77(4), 803-826.

See Also

rsem.pattern, rsem.emmusig, rsem.Ascov

Examples

```
 data(mardiamv25)
 names(mardiamv25)<-paste('V', 1:5, sep='')

 fa.model<-'f1 =~ V1 + V2
f2 =~ V4 + V5
f1 ~ 1
f2 ~ 1
```

```
V1 ~0*1
V2 ~0*1
V4 ~0*1
V5 ~0*1'

 analysis<-rsem.lavaan(mardiamv25, fa.model, c(1,2,4,5))
```

`rsem.pattern` *Obtaining missing data patterns*

Description

This function obtains the missing data patterns and the number of cases in each patterns. It also tells the number of observed variables and their indices for each pattern.

Usage

```
rsem.pattern(x, print=FALSE)
```

Arguments

`x`	A matrix as data
`print`	Whether to print the missing data pattern. The default is FALSE.

Details

The missing data pattern matrix has 2+p columns. The first column is the number cases in that pattern. The second column is the number of observed variables. The last p columns are a matrix with 1 denoting observed data and 0 denoting missing data.

Value

`x`	Data ordered according to missing data pattern
`misinfo`	Missing data pattern matrix
`mispat`	Missing data pattern in better readable form.

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

Examples

```
#dset<-read.table('MardiaMV25.dat.txt', na.string='-99')
#dset<-data.matrix(dset)
#n<-dim(dset)[1]
#p<-dim(dset)[2]
#miss_pattern<-rsem.pattern(n,p,dset)
#miss_pattern
```

`rsem.print` *Organize the output for Lavaan with robust s.e. and test statistics*

Description

Organize the output for Lavaan with robust s.e. and test statistics. Modified from the print function of Lavaan.

Usage

```
rsem.print(object, robust.se, robust.fit, estimates=TRUE, fit.measures=FALSE,
standardized=FALSE, rsquare=FALSE, std.nox=FALSE, modindices=FALSE)
```

Arguments

`object`	Output from lavaan analysis, such as growth, factor, sem functions.
`robust.se`	Robust standard error from the function rsem.se
`robust.fit`	Robust fit statistics from the function rsem.fit
`estimates`	Show parameter estimates
`fit.measures`	Show fit statistics of lavaan (no need for it)
`standardized`	standardized coefficients
`rsquare`	R square for dependent variables.
`std.nox`	to add
`modindices`	Modification indices

Details

This function will run the robust analysis and output results.

Value

If `EQSmodel` is not supplied

`sem`	Information for SEM analysis including estimated means, covariance matrix and their sandwich type covariance matrix in the order of mean first and then covariance matrix.
`misinfo`	Information related to missing data pattern

em	Results from expectation robust algorithm
ascov	Covariance matrix

If EQSmodel is supplied,

sem	Information for SEM analysis including estimated means, covariance matrix and their sandwich type covariance matrix according to the requirement of EQS.

In addition, the following model parameters are from EQS

fit.stat	Fit indices and associated p-values
para	Parameter estimates
eqs	All information from REQS

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

See Also

rsem.pattern, rsem.emmusig, rsem.Ascov

Examples

```
##\dontrun{
## an example
 data(mardiamv25)
 names(mardiamv25)<-paste('V', 1:5, sep='')

 fa.model<-'f1 =~ V1 + V2
f2 =~ V4 + V5
f1 ~ 1
f2 ~ 1
V1 ~0*1
V2 ~0*1
V4 ~0*1
V5 ~0*1'

pat<-rsem.pattern(mardiamv25)

phi<-0.1
musig<-rsem.emmusig(pat, varphi=phi)

res.lavaan<-sem(fa.model, sample.cov=musig$sigma, sample.mean=musig$mu, sample.nobs=88,mimic='EQS')

ascov<-rsem.Ascov(pat, musig, varphi=phi)
```

```
robust.se<-rsem.se(res.lavaan, ascov$Gamma)

robust.fit <- rsem.fit(res.lavaan, ascov$Gamma, musig)

rsem.print(res.lavaan, robust.se, robust.fit)
## }
```

`rsem.se` *Calculate robust standard errors*

Description

Calculate robust standard errors

Usage

```
rsem.se(object, gamma)
```

Arguments

`object`	Output from lavaan analysis, such as growth, factor, sem functions.
`gamma`	Robust covariance matrix for saturated mean and covariances

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

Examples

```
x<-array(1:6, c(2,3))
rsem.vec(x)
```

rsem.ssq *Calculate the squared sum of a matrix*

Description

Calculate the squared sum of a matrix

Usage

```
rsem.ssq(x)
```

Arguments

x	A matrix

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

Examples

```
x<-array(1:6, c(2,3))
rsem.ssq(x)
```

rsem.switch *swith function*

Description

swith function

Usage

```
rsem.switch(p)
```

Arguments

p	number of variables

rsem.switch.gamma *Internal function*

Description

Internal function

Usage

```
rsem.switch.gamma(gamma, ov.names)
```

Arguments

gamma	Robust covariance matrix for saturated mean and covariances
ov.names	Observed variable names.

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

rsem.vec *Stacking a matrix to a vector*

Description

Stacking a matrix to a vector

Usage

```
rsem.vec(x)
```

Arguments

x	A matrix

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

Examples

```
x<-array(1:6, c(2,3))
rsem.vec(x)
```

`rsem.vech` *Stacking lower triange of a matrix to a vector*

Description

Stacking lower triange of a matrix to a vector

Usage

```
rsem.vech(x)
```

Arguments

`x`	A matrix

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

References

Ke-Hai Yuan and Zhiyong Zhang (2011) Robust Structural Equation Modeling with Missing Data and Auxiliary Variables

Examples

```
x<-array(1:9, c(3,3))
rsem.vec(x)
```

rsem.weight *Calculate weight for each subject*

Description

Calculate weight for each subject in estimating the mean and covariance matrix.

Usage

```
rsem.weight(x, varphi, mu0, sig0)
```

Arguments

x	Data
varphi	Downweight rate.
mu0	Robust mean
sig0	Robust covariance matrix.

Value

w1	Weight for robust mean estimates
w2	Weight for robust covariance estimates

Author(s)

Zhiyong Zhang and Ke-Hai Yuan

References

Yuan, K.-H., & Zhang, Z. (2012). Robust Structural Equation Modeling with Missing Data and Auxiliary Variables. Psychometrika, 77(4), 803-826.

semdiag.combinations *Enumerate the Combinations of the Elements of a Vector*

Description

Enumerate the Combinations of the Elements of a Vector

Usage

```
semdiag.combinations(n, r)
```

Arguments

n	Size of the source vector
r	Size of the target vectors

`semdiag.read.eqs` *Import of EQS outputs into R*

Description

This function reads EQS output files (.ets, .CBK and .ETP) into R and stores the results as objects.

Usage

```
semdiag.read.eqs(file)
```

Arguments

`file`	The name (string) of the .ets file or the full path which the data are to be read from. If it does not contain an absolute path, the file name is relative to the current working directory, 'getwd()'. A .CBK and .ETP file have to be of the same name and in the same directory.

Details

The value list below provides objects for the full EQS output. If in EQS some objects are not computed, the corresponding values in R are `NA`.

Value

Returns a list with the following objects:

`model.info`	General model information
`pval`	p-values for various test statistics
`fit.indices`	Variuos fit indices
`model.desc`	Descriptive measures
`Phi`	Phi matrix
`Gamma`	Gamma matrix
`Beta`	Beta matrix
`par.table`	Parameter table (with standard errors)
`sample.cov`	Sample covariance matrix
`sigma.hat`	Model covariance matrix
`inv.infmat`	Inverse information matrix
`rinv.infmat`	Robust inverse information matrix
`cinv.infmat`	Corrected inverse information matrix
`derivatives`	First derivatives
`moment4`	Matrix with 4th moments
`ssolution`	Standardized elements

Rsquared	R-squared measures
fac.means	Factor means
var.desc	Descriptive measures for the variables (univariate statistics)
indstd	Independent variable standardization vector
depstd	Dependent variable standardization vector

Author(s)

Patrick Mair, Eric Wu

References

Bentler, P. M. (2008). EQS Program Manual. Encino, CA: Multivariate Software Inc.

See Also

semdiag.call.eqs, semdiag.run.eqs

---

semdiag.run.eqs *Run EQS from R*

Description

Calls an EQS script file from R, executes EQS, and imports the results into R. Basically it is a wrapper function of call.eqs and the subsequent read.eqs.

Usage

```
semdiag.run.eqs(EQSpgm, EQSmodel, serial, Rmatrix = NA, datname = NA, LEN = 2000000)
semdiag.call.eqs(EQSpgm, EQSmodel, serial, Rmatrix = NA, datname = NA, LEN = 2000000)
```

Arguments

EQSpgm	String containing path where EQS is located (see details)
EQSmodel	String containing path where .eqs script file is located (see details)
serial	EQS serial number as integer value
Rmatrix	Optional matrix argument if data or covariances are stored in R
datname	If data is specified, a filename (string) must be provided for saving the data in text format (blank separated; see details)
LEN	Integer containing number of working array units. By default, it is 2000000 8 bytes units

Details

If the path in EQSpgm and EQSmodel contains a blank, single quotes and double quotes are required in argument. See EQSpgm argument in examples. The last statement in the EQSpgm argument refers to the name of the executable program file. Under Windows it is "../WINEQS" (referring to WINEQS.exe), under Mac "../MACEQS" and under Linux "../EQS". When specifying the path, use slash instead of backslash.

The .ETS, .CBK and .ETP files are written in the directory where the .eqs file is located. Note that these 3 files must be in the same directory than the .eqs file.

The argument datname must match with the input data specified in the corresponding .eqs file. This option can be used for simulations: Generate data in R, run.eqs() on with the corresponding data argument, pick out the relevant return values.

The value list below provides objects for the full EQS output. If in EQS some objects are not computed, the corresponding values in R are NA.

Value

Returns a list with the following objects:

success	TRUE if estimation was successful, FALSE otherwise
model.info	General model information
pval	p-values for various test statistics
fit.indices	Variuos fit indices
model.desc	Descriptive measures
Phi	Phi matrix
Gamma	Gamma matrix
Beta	Beta matrix
par.table	Parameter table (with standard errors)
sample.cov	Sample covariance matrix
sigma.hat	Model covariance matrix
inv.infmat	Inverse information matrix
rinv.infmat	Robust inverse information matrix
cinv.infmat	Corrected inverse information matrix
derivatives	First derivatives
moment4	Matrix with 4th moments
ssolution	Standardized elements
Rsquared	R-squared measures
fac.means	Factor means
var.desc	Descriptive measures for the variables (univariate statistics)
indstd	Independent variable standardization vector
depstd	Dependent variable standardization vector

Author(s)

Patrick Mair, Eric Wu

References

Bentler, P. M. (1995). EQS Program Manual. Encino, CA: Multivariate Software Inc.

See Also

semdiag.read.eqs, semdiag.call.eqs

Index