

Package ‘semdiag’

February 20, 2015

Type Package

Title Structural equation modeling diagnostics

Version 0.1.2

Date 2010-02-07

Author Zhiyong Zhang and Ke-Hai Yuan

Maintainer Zhiyong Zhang <zhiyongzhang@nd.edu>

Depends R (>= 2.10), sem

Description Outlier and leverage diagnostics for SEM.

License GPL-2

LazyLoad yes

Repository CRAN

Date/Publication 2012-01-30 14:01:44

NeedsCompilation no

R topics documented:

semdiag-package	2
N100	2
semdiag	3
semdiag.combinations	6
semdiag.cpp	7
semdiag.DP	7
semdiag.eq	8
semdiag.eq.lisrel	8
semdiag.input.model	9
semdiag.lisrel	9
semdiag.mdist.f	10
semdiag.musig	11
semdiag.orthog	11
semdiag.parse	12
semdiag.plot	12
semdiag.read.eq	13

semdiag.robfit	14
semdiag.run.eqs	15
semdiag.vech	17
semdiag.write.eqs	17

Index	18
--------------	-----------

semdiag-package	<i>Structural equation modeling diagnostic</i>
-----------------	--

Description

Structural equation modeling diagnostic

Details

Package:	semdiag
Type:	Package
Version:	1.0
Date:	2011-02-07
License:	GPL-2
LazyLoad:	yes

To use the package, see

Author(s)

Zhiyong Zhang and Ke-Hai Yuan

Maintainer: Zhiyong Zhang <zhiyongzhang@nd.edu>

References

To be added

N100	<i>Simulated data</i>
------	-----------------------

Description

N100: normal data

N85: data with outliers

Usage

data(N100)

data(N85)

semdiag *The semdiag function*

Description

A function to calculate distance and parameter estimates for SEM diagnostics

Usage

```
semdiag(x, EQSmodel, varphi=0.1, EQSdata='data.txt', D='E', delete=integer(0), max_it=1000, EQSprog=
```

Arguments

x	Data frame or data matrix
ram.path	Ram path for R sem package
software	Software to be used. sem for R sem package and EQS for the use of EQS
varphi	Percentage of data to be down-weighted
EQSmodel	EQS input file name
EQSdata	Data file name used in EQS input file
D	How to treat prediction error. E: errors; F: factors
delete	A vector of data to be deleted. For example c(99,100) delete the 99th and 100th cases.
max_it	The maximum number of iterations
EQSprog	The path to where EQS program is installed.
serial	Serial no. for EQS. This is a string with spaces. Currently, it does not need to be supplied.

Value

d_f	Distance for f
d_r	Distance for r
mu	Mean
p	Number of observed variables
q	Number of factors used in calculating f and r
res	Model fit and paramter estimates. nml: normal ML; tsr: two-stage robust; dr: direct robust
eqs	Full EQS output for the above three models
x	Data

Author(s)

Zhiyong Zhang and Ke-Hai Yuan

Maintainer: Zhiyong Zhang <zhiyongzhang@nd.edu>

References

Yuan, K.-H. and Zhang, Z. (2011). Structural Equation Modeling Diagnostics Using R Package semdiag and EQS. *Structural Equation Modeling: An Interdisciplinary Journal*.

Yuan, K.-H., and Hayashi, K. (2010). Fitting data to model: Structural equation modeling diagnosis using two scatter plots. *Psychological Methods*, 15, 335–351.

Examples

```
## Not run:
## Examples based on EQS
library(semdiag)

## Example 1. Normally distributed data
data(N100)

## Not run
## The EQS input file is semplot.eqs
## Model 1: treating prediction errors as factors
N100out.1<-semdiag(N100, 'semplot.eqs', D='F')

## Diagnostics plot
semdiag.plot(N100out.1)

## Summary output
semdiag.summary(N100out.1)

## Model 0: treating prediction errors the same as measurement errors
N100out.0<-semdiag(N100, 'semplot.eqs')

## Diagnostics plot
semdiag.plot(N100out.0)

## Summary output
semdiag.summary(N100out.0)

## Example 2. Contaminated data
data(N85)

## The EQS input file is semplot.eqs
## Model 1: treating prediction errors as factors
N85out.1<-semdiag(N85, 'semplot.eqs', D='F')

## Diagnostics plot
semdiag.plot(N85out.1)

## Summary output
semdiag.summary(N85out.1)

## Model 0: treating prediction errors the same as measurement errors
N85out.0<-semdiag(N85, 'semplot.eqs', D='E')
```

```
## Diagnostics plot
semdiag.plot(N85out.0)

## Summary output
semdiag.summary(N85out.0)

## Case profile plot
semdiag.cpp(N85out.0, cases=c(86, 90, 98:100))

## Delete the 99th and 100th observations
N85out.1.del<-semdiag(N85, 'semplot.eqs', D='F', delete=c(99,100))

## Examples based on the sem package
library(sem)
data(N100)

## path diagram for the model
sem1<-specify.model()
f1 -> y1, NA, 1
f1 -> y2, 11, NA
f1 -> y3, 12, NA
f2 -> y4, NA, 1
f2 -> y5, 13, NA
f2 -> y6, 14, NA
f3 -> y7, NA, 1
f3 -> y8, 15, NA
f3 -> y9, 16, NA
f1 -> f2, g1, NA
f1 -> f3, g2, NA
f2 -> f3, g3, NA
y1 <-> y1, e1, NA
y2 <-> y2, e2, NA
y3 <-> y3, e3, NA
y4 <-> y4, e4, NA
y5 <-> y5, e5, NA
y6 <-> y6, e6, NA
y7 <-> y7, e7, NA
y8 <-> y8, e8, NA
y9 <-> y9, e9, NA
f1 <-> f1, e10, NA
f2 <-> f2, e11, NA
f3 <-> f3, e12, NA

## Model 1: treating prediction errors as factors
N100out.1<-semdiag(N100, ram.path=sem1, software='sem', D='F')

## Diagnostics plot
semdiag.plot(N100out.1)

## Summary output
semdiag.summary(N100out.1)

## Model 0: treating prediction errors the same as measurement errors
```

```
N100out.0<-semdiag(N100, ram.path=sem1, software='sem')

## Diagnostics plot
semdiag.plot(N100out.0)

## Summary output
semdiag.summary(N100out.0)

## Example 2. Contaminated data
data(N85)

## Model 1: treating prediction errors as factors
N85out.1<-semdiag(N85, ram.path=sem1, software='sem', D='F')

## Diagnostics plot
semdiag.plot(N85out.1)

## Summary output
semdiag.summary(N85out.1)

## Model 0: treating prediction errors the same as measurement errors
N85out.0<-semdiag(N85, ram.path=sem1, software='sem', D='E')

## Diagnostics plot
semdiag.plot(N85out.0)

## Summary output
semdiag.summary(N85out.0)

## Case profile plot
semdiag.cpp(N85out.0, cases=c(86, 90, 98:100))

## Delete the 99th and 100th observations
N85out.1.del<-semdiag(N85, ram.path=sem1, software='sem', D='F', delete=c(99,100))

## End(Not run)
```

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.cpp	<i>Case profile plot</i>
-------------	--------------------------

Description

Generate profile plots for selected cases

Usage

```
semdiag.cpp(d, cases)
```

Arguments

d	Object from semdiag
cases	Cases to be plotted

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

semdiag.DP	<i>Generate a duplication matrix</i>
------------	--------------------------------------

Description

Generate a duplication matrix

Usage

```
semdiag.DP(x)
```

Arguments

x	A matrix
---	----------

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

semdiag.eqs

Orgnize results from EQS

Description

Orgnize results from EQS

Usage

semdiag.eqs(res)

Arguments

res EQS output

Value

Return model chi-square, p-value and parameter estimates

semdiag.eqs.lisrel

Format EQS output into LISREL format

Description

Format EQS output into LISREL format

Usage

semdiag.eqs.lisrel(model)

Arguments

model An EQS output from the function semdiag.run.eqs

Value

Return LISREL notation matrices

semdiag.input.model *Input model in R*

Description

Input model in R

Usage

```
semdiag.input.model(file = "")
```

Arguments

file A file name or blank

Value

EQS model

semdiag.lisrel *Generate the lisrel notation matrices based on an sem object.*

Description

Generate the lisrel notation matrices based on an sem object.

Usage

```
semdiag.lisrel(model)
```

```
semdiag.start(ram.path, coeff)
```

Arguments

model An sem model
ram.path An ram path object
coeff Coefficients to be used as starting values.

Author(s)

Ke-Hai Yuan and Zhiyong Zhang

semdiag.mdist.f *Calculate the M-distance for factors and residuals*

Description

Function to calculate M-distance for factors and residuals

Usage

```
semdiag.mdist.f(x, mu, lisrel)
semdiag.mdist.f1(x, mu, lisrel)
semdiag.mdist.f.r(x, mu, lisrel)
semdiag.mdist.f1.r(x, mu, lisrel)
semdiag.mdist.r(x, mu, lisrel)
semdiag.mdist.r1(x, mu, lisrel)
semdiag.mdist.r.r(x, mu, lisrel)
semdiag.mdist.r1.r(x, mu, lisrel)
```

Arguments

x	Data
mu	Mean
lisrel	Lisrel notation matrices

Value

semdiag.mdist.f: M-distance for factors based on Bartlett-factor score using EQS
semdiag.mdist.f1: M-distance for factors based on Bartlett-factor score based on all latent factors (regarding D as factors) using EQS
semdiag.mdist.f.r: M-distance for factors based on Bartlett-factor score using R sem package
semdiag.mdist.f1.r: M-distance for factors based on Bartlett-factor score based on all latent factors (regarding D as factors) using R sem package
semdiag.mdist.r: M-distance for residuals using EQS
semdiag.mdist.r1: M-distance for residuals based on all latent factors (regarding D as factors) using EQS
semdiag.mdist.r.r: M-distance for residuals using R sem package
semdiag.mdist.r1.r: M-distance for residuals based on all latent factors (regarding D as factors) using R sem package

semdiag.musig	<i>Robust covariance estimation</i>
---------------	-------------------------------------

Description

Robust covariance estimation

Usage

```
semdiag.musig(x, varphi, max_it = 1000)
```

Arguments

x	Data
varphi	Percentage of data to be down-weighted
max_it	The maximum number of iterations

Value

Estimated mean and covariance matrix

semdiag.orthog	<i>semdiag.orthog</i>
----------------	-----------------------

Description

semdiag.orthog

Usage

```
semdiag.orthog(A)
```

Arguments

A	A matrix
---	----------

Value

Return

semdiag.parse	<i>Parse EQS input file</i>
---------------	-----------------------------

Description

Parse EQS input file

Usage

```
semdiag.parse(eqs)
```

Arguments

eqs	EQS input file
-----	----------------

Value

RAM notation for the model in EQS input file

semdiag.plot	<i>Plot the diagnostics graphs</i>
--------------	------------------------------------

Description

Plot the diagnostics graphs

Usage

```
semdiag.plot(d, alpha=.01, label=0, cex=1)
```

```
semdiag.summary(d, alpha=.01, digits=2)
```

Arguments

d	Distance from semdiag function
label	Label automatically if 0. 1, label manually
cex	size of the labels
alpha	Critical value, default 0.01
digits	Digits of results to be kept

Value

Print outliers and leverage cases.

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.robfit	<i>Robust method for calculating d_r</i>
----------------	--

Description

Robust method for calculating d_r

Usage

```
semdiag.robfit(lisrel0, x, q, varphi, EQSmodel, EQSdata, max_it=1000, EQSprog='C:/Progra~1/EQS61/WINE
```

```
semdiag.robfit1(lisrel0, x, q, varphi, EQSmodel, EQSdata, max_it=1000, EQSprog='C:/Progra~1/EQS61/WINE
```

```
semdiag.robfit.r(lisrel0, x, q, varphi, ram.path, max_it=1000)
```

```
semdiag.robfit1.r(lisrel0, x, q, varphi, ram.path, max_it=1000)
```

Arguments

lisrel0	Lisrel notation matrices
x	Data frame or data matrix
varphi	Percentage of data to be down-weighted
EQSmodel	EQS input file name
EQSdata	Data file name used in EQS input file
max_it	The maximum number of iterations
EQSprog	The path to where EQS program is installed.
serial	Serial no. for EQS
q	number of factors to be counted
ram.path	Ram path for the sem package

Value

Distance d_r

 semdiag.run.eq *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.eq` and the subsequent `read.eq`.

Usage

```
semdiag.run.eq(EQSpgm, EQSmodel, serial, Rmatrix = NA, datname = NA, LEN = 2000000)
semdiag.call.eq(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.eq()` 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.eq](#), [semdiag.call.eq](#)

semdiag.vech	<i>Stacking lower triange of a matrix to a vector</i>
--------------	---

Description

Stacking lower triange of a matrix to a vector

Usage

```
semdiag.vech(x)
```

Arguments

x	A matrix
---	----------

semdiag.write.eq	<i>Generate an EQS input file</i>
------------------	-----------------------------------

Description

Generate an EQS input file. This is an internal function.

Usage

```
semdiag.write.eq(eqs, par, N, P)
```

Arguments

eqs	EQS input file
par	Parameter estimates
N	Sample size
P	Number of variables

Value

A filed called eqsonce.eq.

Index

N100, [2](#)
N85 (N100), [2](#)

semdiag, [3](#)
semdiag-package, [2](#)
semdiag.call.eqs, [14](#), [16](#)
semdiag.call.eqs (semdiag.run.eqs), [15](#)
semdiag.combinations, [6](#)
semdiag.cpp, [7](#)
semdiag.DP, [7](#)
semdiag.eqs, [8](#)
semdiag.eqs.lisrel, [8](#)
semdiag.input.model, [9](#)
semdiag.lisrel, [9](#)
semdiag.mdist.f, [10](#)
semdiag.mdist.f1 (semdiag.mdist.f), [10](#)
semdiag.mdist.r (semdiag.mdist.f), [10](#)
semdiag.mdist.r1 (semdiag.mdist.f), [10](#)
semdiag.musig, [11](#)
semdiag.orthog, [11](#)
semdiag.parse, [12](#)
semdiag.plot, [12](#)
semdiag.read.eqs, [13](#), [16](#)
semdiag.robfit, [14](#)
semdiag.robfit1 (semdiag.robfit), [14](#)
semdiag.run.eqs, [14](#), [15](#)
semdiag.start (semdiag.lisrel), [9](#)
semdiag.summary (semdiag.plot), [12](#)
semdiag.vch, [17](#)
semdiag.write.eqs, [17](#)