Package 'coefficientalpha'

July 2, 2014

Type Package

Title Robust Cronbach's alpha with missing and non-normal data

Version 0.2.6

Date 2013-01-22

Depends MASS

Author Zhiyong Zhang and Ke-Hai Yuan

Maintainer Zhiyong Zhang <johnnyzhz@gmail.com>

Description Cronbach's alpha is one of the most widely used reliability or internal consistency measures in social,behavioral and education sciences. Alpha is reported in nearly every study that involves measuring a construct through multiple test items. The package coefficientalpha calculates Cronbach's coefficient alpha with missing data and non-normal data. Robust standard error and confidence interval are also provided.

License GPL

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2013-04-25 15:44:25

R topics documented:

`coefficientalpha-package` *Robust Cronbach's alpha with missing and non-normal data*

Description

An R package to calculate coefficient alpha with missing data and non-normal data. Robust standard error and confidence interval are also provided.

Details

Package:	coefficientalpha
Type:	Package
Version:	0.1
Date:	2013-01-22
License:	GPL

Author(s)

Zhiyong Zhang and Ke-Hai Yuan Maintainer: Zhiyong Zhang <zzhang4@nd.edu>

References

Zhang, Z. & Yuan, K.-H. (2013). Robust Cronbach's alpha.

`cronbach` *Robust Cronbach's alpha*

Description

Calculate alpha for a given data set.

Usage

```
cronbach(y, varphi = 0.1, se = FALSE, complete = FALSE)
```

Arguments

y	Data
varphi	Downweight rate
se	Whether to estimate standard error. It will take significant more time if se is requested.
complete	Calculate alpha only based on complete data (lisewise deletion)

Author(s)

Zhiyong Zhang and Ke-Hai Yuan

References

Zhang, Z. & Yuan, K.-H. (2013). Robust Cronbach's alpha.

Examples

```
data(holzinger)
spatial<-holzinger[, c('visual', 'cubes', 'paper', 'lozenge', 'paperrev','flagssub')]
verbal<-holzinger[, c('general', 'paragrap', 'sentence', 'wordc', 'wordm')]
speed<-holzinger[, c('add', 'code', 'counting', 'straight')]
memory<-holzinger[, c('wordr', 'numberr', 'figurer', 'object', 'numberf', 'figurew')]

alpha.spatial<-cronbach(spatial, varphi=.01)
```

---

holzinger *An example data set*

Description

The Holzinger and Swine data set.

Usage

```
data(holzinger)
```

plot.alpha *Alpha related plot*

Description

Generate plot related to alpha. Three plot can be generated. (1) The weight plot will plot the weight associated with alpha calculation. (2) Profile plot will plot the cases with smallest weights and the average value. (3) The diagnostic plot plots the alpha according to different values of the tuning parameter.

Usage

```
## S3 method for class 'alpha'
plot(x, type="weight", profile=5, interval=0.01,center=TRUE, scale=FALSE, w1=FALSE, numbered=FALSE, p
```

Arguments

x	Results from the function cronbach.
type	Three types of plots can be generated. type="weight" generates the plot of weight of each case. type="profile" generates a profile plot for the smallest weight. type="weight" generates the diagnostic plot for varphi with an interval of .01.
profile	Number of cases used on the profile plot. At most 10.
interval	The interval used in the diagnostic plot. The default is 0.01.
center	Whether to center the data in the profile plot.
scale	Whether to scale the data using variance parameters in the profile plot.
w1	Whether to plot the weight for means
numbered	Whether to number the profile plot
pos	Position of legend. If pos=NULL, no legend is plotted.
...	Options can be passed to the plot function.

Author(s)

Zhiyong Zhang and Ke-Hai Yuan

References

Zhang, Z. & Yuan, K.-H. (2013). Robust Cronbach's alpha.

Examples

```
data(holzinger)
spatial<-holzinger[, c('visual', 'cubes', 'paper', 'lozenge', 'paperrev','flagssub')]
verbal<-holzinger[, c('general', 'paragrap', 'sentence', 'wordc', 'wordm')]
speed<-holzinger[, c('add', 'code', 'counting', 'straight')]
memory<-holzinger[, c('wordr', 'numberr', 'figurer', 'object', 'numberf', 'figurew')]

alpha.spatial<-cronbach(spatial)
## diagnostic plot
plot(alpha.spatial, type='d')

## alpha with varphi=.01 & standard error
alpha.spatial<-cronbach(spatial, varphi=.01, se=TRUE)
## confidence interval
summary(alpha.spatial)

## weight plot
plot(alpha.spatial)
# or
plot(alpha.spatial, type='w')

## profile plot
plot(alpha.spatial, type='p')


alpha.verbal<-cronbach(verbal, varphi=.02, se=TRUE)
## confidence interval
summary(alpha.verbal)

## weight plot
plot(alpha.verbal)

## profile plot
plot(alpha.verbal, type='p', profile=6)

alpha.speed<-cronbach(speed, varphi=.01, se=TRUE)
## confidence interval
summary(alpha.speed)

## weight plot
plot(alpha.speed)

## profile plot
plot(alpha.speed, type='p', pos='topleft')

alpha.memory<-cronbach(memory, varphi=.04, se=TRUE)
## confidence interval
summary(alpha.memory)

## weight plot
plot(alpha.memory)
```

```
## profile plot
plot(alpha.memory, type='p', profile=10, pos='bottomright')
 plot(alpha.memory, type='p', profile=10, scale=TRUE, pos='bottomright')
```

`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

See Also

`rsem.emmusig`

rsem.DP *Generate a duplication matrix*

Description

Generate a duplication matrix

Usage

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

Arguments

x	A matrix

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
weight	weight used in robust mean and covariance estimation.

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.

See Also

rsem.emmusig

---

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.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.

In addition, a matrix of 0/1 is also used to indicate missing data. 1 means missing and 0 means observed.

Value

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

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.

rsem.vec *Stacking a matrix to a vector*

Description

Stacking a matrix to a vector

Usage

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

Arguments

x	A matrix

Examples

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

`summary.alpha` *Print alpha and its confidence interval.*

Description

Print alpha and its confidence interval.

Usage

```
## S3 method for class 'alpha'
summary(object, prob = 0.95,...)
```

Arguments

`object`	Results from the function `cronbach`.
`prob`	Alpha level for confidence interval. By default, the 95
`...`	Option can be passed to the summary function.

Author(s)

Zhiyong Zhang and Ke-Hai Yuan

References

Zhang, Z. & Yuan, K.-H. (2013). Robust Cronbach's alpha.

Examples

```
data(holzinger)
spatial<-holzinger[, c('visual', 'cubes', 'paper', 'lozenge', 'paperrev','flagssub')]
verbal<-holzinger[, c('general', 'paragrap', 'sentence', 'wordc', 'wordm')]
speed<-holzinger[, c('add', 'code', 'counting', 'straight')]
memory<-holzinger[, c('wordr', 'numberr', 'figurer', 'object', 'numberf', 'figurew')]

alpha.spatial<-cronbach(spatial)
## diagnostic plot
plot(alpha.spatial, type='d')

## alpha with varphi=.01 & standard error
alpha.spatial<-cronbach(spatial, varphi=.01, se=TRUE)
## confidence interval
summary(alpha.spatial)
```

Index