Package 'compositeReliabilityInNestedDesigns'

July 22, 2025

Title Optimizing the Composite Reliability in Multivariate Nested Designs

Version 1.0.4

Description The reliability of assessment tools is a crucial aspect of monitoring student performance in various educational settings. It ensures that the assessment outcomes accurately reflect a student's true level of performance. However, when assessments are combined, determining composite reliability can be challenging, especially for naturalistic and unbalanced datasets in nested design as is often the case for Workplace-Based Assessments. This package is designed to estimate composite reliability in nested designs using multivariate generalizability theory and enhance the analysis of assessment data. The package allows for the inclusion of weight per assessment type and produces extensive G- and D-study results with graphical interpretations, and options to find the set of weights that maximizes the composite reliability or minimizes the standard error of measurement (SEM).

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.3

Imports dplyr, ggplot2, lme4, magrittr, plyr, psych, reshape2, tidyr, Rsolnp

Depends R (>= 2.10)

LazyData true

URL https://github.com/jmoonen/compositeReliabilityInNestedDesigns

BugReports https://github.com/jmoonen/compositeReliabilityInNestedDesigns/issues

NeedsCompilation no

Author Joyce Moonen - van Loon [aut, cre] (ORCID: <https://orcid.org/0000-0002-8883-8822>)

Maintainer Joyce Moonen - van Loon <j.moonen@maastrichtuniversity.nl>

Repository CRAN

Date/Publication 2024-09-21 17:40:01 UTC

Contents

calculateReliability . . . 2
calculateVarCov . . . 3
checkDatasets . . . 3
computeCompositeReliability . . . 4
computeMaxCompositeReliability . . . 5
DStudy . . . 6
GStudy . . . 6
GStudyPerType . . . 7
mydata . . . 8

Index **9**

`calculateReliability` *calculateReliability: determine the reliability and SEM per Type*

Description

calculateReliability: determine the reliability and SEM per Type

Usage

```
calculateReliability(mydata, n)
```

Arguments

`mydata`	A dataframe containing columns ID, Type, Score (numeric)
`n`	A vector containing for each Type the number of scores or assessments, e.g. averages, requirements.

Value

A list containing 2 vectors; one vector with the reliability coefficient of each Type, the other vector with the SEM values for each Type

Examples

```
rel <- calculateReliability(mydata, n=c("A"=3,"B"=3,C="2"))
```

calculateVarCov

calculateVarCov: Estimate variance and covariance components of assessee p S_p and mean assessment scores i nested in assessees S_iINp, and determine the error scores S_delta

Description

calculateVarCov: Estimate variance and covariance components of assessee p S_p and mean assessment scores i nested in assessees S_iINp, and determine the error scores S_delta

Usage

```
calculateVarCov(mydata, n)
```

Arguments

`mydata` A dataframe containing columns ID, Type, Score (numeric)

`n` A vector containing for each Type the number of scores or assessments, e.g. averages, requirements.

Value

A list containing the observed variances, covariances and errors scores

Examples

```
varcov <- calculateVarCov(mydata, c(”A”=3, ”B”=3, ”C”=2))
varcov$S_p
varcov$S_iINp
varcov$S_delta
```

checkDatasets

checkDatasets: assert that the given datasets adhere to the assumptions and requirements of this package i.e. the data set ’mydata’ is a dataframe with 3 columns, named "ID", "Type" and "Score", column "Score" contains numeric data, and each combination of "ID" and "Type" exists at least once, data set n contains a numerical value for each "Type", and data set weights contains a numerical value for each "Type" and the sum of all values is equal to 1.

Description

checkDatasets: assert that the given datasets adhere to the assumptions and requirements of this package i.e. the data set ’mydata’ is a dataframe with 3 columns, named "ID", "Type" and "Score", column "Score" contains numeric data, and each combination of "ID" and "Type" exists at least once, data set n contains a numerical value for each "Type", and data set weights contains a numerical value for each "Type" and the sum of all values is equal to 1.

Usage

```
checkDatasets(mydata, n = NULL, weights = NULL)
```

Arguments

`mydata` A dataframe containing columns ID, Type, Score (numeric)

`n` A vector containing for each Type the number of scores or assessments, e.g. averages, requirements.

`weights` A vector containing for each Type the weight assigned to it. The sum of weights should be equal to 1.

Value

A list with the number of Assessments per ID per Type

Examples

```
checkDatasets(mydata, n=c("A"=10, "B"=5, "C"=2), weights=c("A"=1/3,"B"=1/3, "C"=1/3))
```

---

computeCompositeReliability

computeCompositeReliability: multivariate generalizability theory approach to estimate the composite reliability of student performance across different types of assessments.

Description

computeCompositeReliability: multivariate generalizability theory approach to estimate the composite reliability of student performance across different types of assessments.

Usage

```
computeCompositeReliability(mydata, n, weights, optimizeSEM)
```

Arguments

`mydata` A dataframe containing columns ID, Type, Score (numeric)

`n` A vector containing for each Type the number of scores or assessments, e.g. averages, requirements.

`weights` A vector containing for each Type the weight assigned to it. The sum of weights should be equal to 1.

`optimizeSEM` Boolean, if TRUE, the weights are adjusted in order to minimize the Standard Error of Measurement (SEM)

Value

A list containing the composite reliability coefficient, the SEM and the distribution of weights. If 'optimizeSEM' is set to TRUE, the vector of weights minimizes the SEM.

Examples

```
compRel <- computeCompositeReliability(mydata, n=c("A"=10, "B"=5, "C"=2),
                              weights=c("A"=1/3,"B"=1/3, "C"=1/3), optimizeSEM=TRUE)
compRel$reliability
compRel$SEM
compRel$weights
```

computeMaxCompositeReliability

computeMaxCompositeReliability: multivariate generalizability theory approach to estimate the maximum composite reliability of student performance across different types of assessments.

Description

computeMaxCompositeReliability: multivariate generalizability theory approach to estimate the maximum composite reliability of student performance across different types of assessments.

Usage

```
computeMaxCompositeReliability(mydata, n)
```

Arguments

mydata	A dataframe containing columns ID, Type, Score (numeric)
n	A vector containing for each Type the number of scores or assessments, e.g. averages, requirements.

Value

A list containing the composite reliability coefficient, the SEM and the distribution of weights.

Examples

```
compMaxRel <- computeMaxCompositeReliability(mydata, n=c("A"=3, "B"=2, "C"=1))
compMaxRel$reliability
compMaxRel$SEM
compMaxRel$weights
```

DStudy *DStudy: the program presents the reliability coefficient and the SEM for different numbers of assessments per type. Both the reliability coefficient and the SEM are presented in graphs for differing numbers of assessments, given insight in the impact on the reliability if more or less assessments per type were required or advised.*

Description

DStudy: the program presents the reliability coefficient and the SEM for different numbers of assessments per type. Both the reliability coefficient and the SEM are presented in graphs for differing numbers of assessments, given insight in the impact on the reliability if more or less assessments per type were required or advised.

Usage

```
DStudy(mydata, maxNrAssessments = 60)
```

Arguments

mydata	A dataframe containing columns ID, Type, Score (numeric)
maxNrAssessments	The maximum (Int) number of assessments per type on with the D study is executed

Value

A list containing 2 plots: reliability (plotRel) and Standard Error of Measurement SEM (plotSEM)

Examples

```
plots <- DStudy(mydata, maxNrAssessments = 10)
```

GStudy *GStudy for a dataset in which every student p has a potentially differing number of scores i on each assessment type m. i.e. model i: (p x m). The output gives descriptive statistics, reliability coefficient and SEM for each assessment type.*

Description

GStudy for a dataset in which every student p has a potentially differing number of scores i on each assessment type m. i.e. model i: (p x m). The output gives descriptive statistics, reliability coefficient and SEM for each assessment type.

Usage

```
GStudy(mydata, nrDigitsOutput = 4)
```

Arguments

`mydata`	A dataframe containing columns ID, Type, Score (numeric)
`nrDigitsOutput`	Integer, number of digits in the output

Value

Matrix with descriptive statistics for each Type of assessment

Examples

```
GStudy(mydata,nrDigitsOutput=4)
```

`GStudyPerType` *GStudyPerType: This function is mainly used within calculateVarCov.R, but can be executed on its own to determine the reliability coefficient and SEM for a dataset with a single type of assessment.*

Description

GStudyPerType: This function is mainly used within calculateVarCov.R, but can be executed on its own to determine the reliability coefficient and SEM for a dataset with a single type of assessment.

Usage

```
GStudyPerType(dataPerAssessmentType)
```

Arguments

`dataPerAssessmentType`
A dataframe containing columns ID, Type, Score (numeric), with only one value in column Type

Value

A matrix presenting the observerd varianced and residual, number of ID's and the percentage of the total variance for each group

mydata *mydata*

Description

A dataset that can be used as example in package compositeReliabilityInNestedDesigns.

Usage

```
mydata
```

Format

`mydata`**:**

A data frame with 7,240 rows and 60 columns:

ID ID of the student

Type The type of assessment

Score The obtained score by this student on this occasion, using the type of assessment ...

Index

∗ **datasets**
    mydata, 8

calculateReliability, 2
calculateVarCov, 3
checkDatasets, 3
computeCompositeReliability, 4
computeMaxCompositeReliability, 5

DStudy, 6

GStudy, 6
GStudyPerType, 7

mydata, 8