Package ‘subscore’

October 14, 2022

Title Computing Subscores in Classical Test Theory and Item Response Theory

Version 3.3

Author Shenghai Dai [aut, cre],
Xiaolin Wang [aut],
Dubravka Svetina [aut]

Maintainer Shenghai Dai <s.dai@wsu.edu>

Description Functions for computing test subscores using different methods in both classical test theory (CTT) and item response theory (IRT). This package enables three types of subscoring methods within the framework of CTT and IRT, including (1) Wainer's augmentation method (Wainer et. al., 2001) <doi:10.4324/9781410604729>, (2) Haberman's subscoring methods (Haberman, 2008) <doi:10.3102/1076998607302636>, and (3) Yen's objective performance index (OPI; Yen, 1987) <https://www.ets.org/research/policy_research_reports/publications/paper/1987/hrap>. It also includes functions to compute Proportional Reduction of Mean Squared Errors (PRMSEs) in Haberman's methods which are used to examine whether test subscores are of added value. In addition, the package includes a function to assess the local independence assumption of IRT with Yen's Q3 statistic (Yen, 1984 <doi:10.1177/014662168400800201>; Yen, 1993 <doi:10.1111/j.1745-3984.1993.tb00423.x>).

Depends R (>= 3.4.0), CTT, stats, irtoys, sirt, ltm

Imports cocor, boot

NeedsCompilation no

LazyData true

Encoding UTF-8

License GPL (>= 2)

RoxygenNote 7.2.0

Repository CRAN

Date/Publication 2022-05-24 11:00:07 UTC

R topics documented:

CTTsub . . . 2
data.prep . . . 4
scored.data . . . 5
subscore.corr . . . 7
subscore.s . . . 8
subscore.sx . . . 9
subscore.Wainer . . . 10
subscore.x . . . 11
test.data . . . 12
TIMSS11G8M.data . . . 12
Yen.OPI . . . 14
Yen.Q3 . . . 15

Index 16

`CTTsub` *This main function estimates true subscores using different methods based on original CTT scores.*

Description

This function estimates true subscores using methods introduced in studies of Haberman (2008) <doi:10.3102/1076998607302636> and Wainer et al. (2001) <doi:10.4324/9781410604729>. Hypothesis tests (i.e., Olkin' Z,Williams's t, and Hedges-Olkin's Z) are used to determine whether a subscore or an augmented subscore has added value. Codes for the hypothesis tests are from Sinharay (2019) <doi: 10.3102/1076998618788862>.

Usage

```
CTTsub(test.data, method = "Haberman")
```

Arguments

`test.data` A list that contains item responses of all subtests and the entire test, which can be obtained using function 'data.prep'.

`method` Subscore estimation methods. method="Haberman" (by default) represents the three methods proposed by Haberman (2008) <doi:10.3102/1076998607302636>. method="Wainer" represents Wainer's augmented method.

Value

`summary` Summary of estimated subscores (e.g., mean, sd).

`PRMSE` (a) PRMSE values of estimated subscores (for Haberman's methods only).(b) Decisions on whether subscores have added value - added.value.s (or added.value.sx) = 1 means subscore.s (or subscore.sx) has added value, and added.value.s (or added.value.sx) = 0 vice versa.

`PRMSE.test` All information in PRMSE plus results of hypopthesis testing based on Sinharay (2019) <doi:10.3102/1076998618788862>.

`subscore.original`
Original subscores and total score.

`estimated.subscores`
Subscores computed using selected method. Three sets of subscores will be returned if method = "Haberman".

References

Haberman, S. J. (2008). "When can subscores have value?." Journal of Educational and Behavioral Statistics, 33(2), 204-229. doi:10.3102/1076998607302636.

Sinharay, S. (2019). "Added Value of Subscores and Hypothesis Testing." Journal of Educational and Behavioral Statistics, 44(1), 25-44. doi:10.3102/1076998618788862.

Wainer, H., Vevea, J., Camacho, F., Reeve, R., Rosa, K., Nelson, L., Swygert, K., & Thissen, D. (2001). "Augmented scores - "Borrowing strength" to compute scores based on small numbers of items." In Thissen, D. & Wainer, H. (Eds.), Test scoring (pp.343 - 387). Mahwah, NJ: Lawrence Erlbaum Associates, Inc. doi:10.4324/9781410604729.

Examples

```
# Transferring original scored data to a list format
# that can be used in other functions.
test.data<-data.prep(scored.data,c(3,15,15,20),
                     c("Algebra","Geometry","Measurement", "Math"))
#-----------------------------------------------
# Estimating subscores using Haberman's methods
CTTsub(test.data,method="Haberman") # Estimating subscores using Haberman's methods

# Obtaining original correlation for the three methods
CTTsub(test.data,method="Haberman")$Correlation

# Obtaining disattenuated correlation for the three methods
CTTsub(test.data,method="Haberman")$Disattenuated.correlation

# Obtaining PRMSEs for the three methods
CTTsub(test.data,method="Haberman")$PRMSE

# Obtaining descriptive statistics summary for estimated subscores
CTTsub(test.data,method="Haberman")$summary

# Obtaining raw subscores
CTTsub(test.data,method="Haberman")$subscore.original

# Obtaining subscores that are estimated as a function of the observed subscores
CTTsub(test.data,method="Haberman")$subscore.s

# Obtaining subscores that are estimated as a function of the observed total score
CTTsub(test.data,method="Haberman")$subscore.x
```

```
# Obtaining subscores that are estimated as a function of
# both the observed subscores and the observed total score.
CTTsub(test.data,method="Haberman")$subscore.sx

#-----------------------------------------
# Estimating subscores using Wainer's method
CTTsub(test.data,method="Wainer")

# Obtaining descriptive statistics summary for subscores
CTTsub(test.data,method="Wainer")$summary

# Obtaining original subscores
CTTsub(test.data,method="Wainer")$subscore.original

# Obtaining subscores that are estimated using Wainer's augmentation method
CTTsub(test.data,method="Wainer")$subscore.augmented
```

`data.prep` *This function prepares data into a required list format*

Description

This function generates a list of data sets using the scored original data set, which can be used as objects in subscore computing functions.

Usage

```
data.prep(scored.data, subtest.infor, subtest.names = NULL)
```

Arguments

`scored.data`	Original scored data set with rows as individuals and columns as items.
`subtest.infor`	A numerical vector. The first number indicates the number of subtests, followed by numbers of items on each subscale.
`subtest.names`	Names of the subscales AND the entire test. The default is NULL. If not provided, names of "subtest.1", "subtest.2",..., will be assigned.

Value

A list that contains item responses of all subtests and the entire test. The list is then used by other functions (e.g., CTTsub) in the package to obtain subscores.

Examples

```
        subtest.infor<-c(3,15,15,20)
        subtest.names<-c("Algebra","Geometry","Measurement", "Math")
        # This math test consists of 3 subtests, which have 15 algebra
        # items, 15 geometry items, and 20 measurement items.
        test.data<-data.prep(scored.data, subtest.infor, subtest.names)
```

`scored.data` *Sample scored data*

Description

This dataset contains responses of 150 examinees to three subscales. These subscales consist of 15, 15, and 20 items respectively.

Usage

```
data("scored.data")
```

Format

A data frame with 150 observations on the following 50 variables.

`V1` Item 1

`V2` Item 2

`V3` Item 3

`V4` Item 4

`V5` Item 5

`V6` Item 6

`V7` Item 7

`V8` Item 8

`V9` Item 9

`V10` Item 10

`V11` Item 11

`V12` Item 12

`V13` Item 13

`V14` Item 14

`V15` Item 15

`V16` Item 16

`V17` Item 17

`V18` Item 18

`V19` Item 19

`V20` Item 20

`V21` Item 21

`V22` Item 22

`V23` Item 23

`V24` Item 24

`V25`	Item 25
`V26`	Item 26
`V27`	Item 27
`V28`	Item 28
`V29`	Item 29
`V30`	Item 30
`V31`	Item 31
`V32`	Item 32
`V33`	Item 33
`V34`	Item 34
`V35`	Item 35
`V36`	Item 36
`V37`	Item 37
`V38`	Item 38
`V39`	Item 39
`V40`	Item 40
`V41`	Item 41
`V42`	Item 42
`V43`	Item 43
`V44`	Item 44
`V45`	Item 45
`V46`	Item 46
`V47`	Item 47
`V48`	Item 48
`V49`	Item 49
`V50`	Item 50

Details

A dataset containing responses of 150 examinees to a total number of 50 items on three subscales (15, 15, and 20 items respectively).

Examples

```
data(scored.data)
# maybe str(scored.data); plot(scored.data) ...
```

`subscore.corr` *Computing correlation indices for subscores and the total score.*

Description

This function computes Cronbach’s Alpha and Stratified Alpha (Cronbach et al., 1965) <doi: 10.1177/001316446502500201> Disattenuated correlations are also provided.

Usage

```
subscore.corr(test.data)
```

Arguments

`test.data` A list that contains item responses of all subtests and the entire test, which can be obtained using function ’data.prep’.

Value

`summary` Summary of obtained subscores (e.g., mean, sd).

`correlation` Correlation indices as indicated above.

References

Cronbach, L., Schonenman, P., & McKie, D. (1965). "Alpha coefficients for stratified-parallel tests." Educational and Psychological Measurement, 25, 291-282. doi: 10.1177/001316446502500201.

Examples

```
# Transferring scored response data to the required list format
test.data<-data.prep(scored.data,c(3,15,15,20),
                     c("Algebra","Geometry","Measurement", "Math"))

#Estimate true subscores using Haberman's method based on observed subscores
subscore.corr(test.data)

subscore.s(test.data)$summary
subscore.s(test.data)$correlation
```

subscore.s *Computing subscores using Haberman's method based on observed subscores.*

Description

This function estimate true subscores based on observed subscores, using the method introduced by Haberman (2008) <doi:10.3102/1076998607302636>.

Usage

```
subscore.s(test.data)
```

Arguments

`test.data`	A list that contains item responses of all subtests and the entire test, which can be obtained using function 'data.prep'.

Value

`summary`	Summary of obtained subscores (e.g., mean, sd).
`PRMSE`	PRMSEs of obtained subscores (for Haberman's methods only).
`subscore.original`	Original subscores and total score.
`subscore.s`	Subscores that are estimated based on the observed subscore.

References

Haberman, S. J. (2008). "When can subscores have value?." Journal of Educational and Behavioral Statistics, 33(2), 204-229. doi:10.3102/1076998607302636.

Examples

```
# Transferring scored response data to the required list format
test.data<-data.prep(scored.data,c(3,15,15,20),
                     c("Algebra","Geometry","Measurement", "Math"))

# Estimate true subscores using Haberman's method based on observed subscores
subscore.s(test.data)

subscore.s(test.data)$summary
subscore.s(test.data)$Correlation
subscore.s(test.data)$Disattenuated.correlation
subscore.s(test.data)$PRMSE
subscore.s(test.data)$subscore.s
```

`subscore.sx` *Computing subscores using Haberman's method based on both observed total scores and observed subscores.*

Description

This function estimate true subscores based on both observed total scores and observed subscores using the method introduced by Haberman (2008) <doi:10.3102/1076998607302636>.

Usage

```
subscore.sx(test.data)
```

Arguments

`test.data`	A list that contains item responses of all subtests and the entire test, which can be obtained using function 'data.prep'.

Value

`summary`	Summary of obtained subscores (e.g., mean, sd).
`PRMSE`	PRMSEs of obtained subscores (for Haberman's methods only).
`subscore.original`	Original observed subscores and total score.
`subscore.sx`	Subscores that are estimated based on both the observed total score and observed subscore.

References

Haberman, S. J. (2008). "When can subscores have value?." Journal of Educational and Behavioral Statistics, 33(2), 204-229. doi:10.3102/1076998607302636.

Examples

```
test.data<-data.prep(scored.data,c(3,15,15,20),
                     c("Algebra","Geometry","Measurement", "Math"))

subscore.sx(test.data)
subscore.s(test.data)$Correlation
subscore.s(test.data)$Disattenuated.correlation
subscore.sx(test.data)$summary
subscore.sx(test.data)$PRMSE
subscore.sx(test.data)$subscore.sx
```

`subscore.Wainer` *Estimating true subscores using Wainer's augmentation method*

Description

This function estimates subscores using Wainer's augmentation method (Wainer et. al., 2001) <doi:10.4324/9781410604729>. The central idea of this procedure is that, the estimation of subscores will be improved by shrinking the individual observed subscores towards some aggregate values (i.e., group mean subscores). The extent of the shrinkage depends on the closeness of the subscale being estimated with other subscales as well as reliabilities of all the subscales. Wainer's augmentation is a multivariate version of Kelly's formula (Kelly, 1947) <https://www.hup.harvard.edu/catalog.php?isbn=978067
For details of Wainer's augmentation subscoring method, please refer to Wainer et al. (2001) <doi:10.4324/9781410604729>.

Usage

```
subscore.Wainer(test.data)
```

Arguments

`test.data` A list that contains item responses of all subtests and the entire test, which can be obtained using function 'data.prep'.

Value

`summary` It contains statistical summary of the augmented subscores (mean, sd, and reliability).

`Augmented.subscores`
It contains augmented subscores that are obtained using Wainer's method.

References

Wainer, H., Vevea, J., Camacho, F., Reeve, R., Rosa, K., Nelson, L., Swygert, K., & Thissen, D. (2001). "Augmented scores - "Borrowing strength" to compute scores based on small numbers of items" In Thissen, D. & Wainer, H. (Eds.), Test scoring (pp.343 - 387). Mahwah, NJ: Lawrence Erlbaum Associates, Inc. doi:10.4324/9781410604729.

Kelley, T. L. (1947). Fundamentals of statistics. Harvard University Press. https://www.hup.harvard.edu/catalog.php?isbn=97

Examples

```
        test.data<-data.prep(scored.data,c(3,15,15,20),
                             c("Algebra","Geometry","Measurement", "Math"))

         subscore.Wainer(test.data)

         subscore.Wainer(test.data)$summary
         subscore.Wainer(test.data)$subscore.augmented
```

subscore.x *Computing subscores using Haberman's method based on observed total scores.*

Description

This function estimates true subscores based on observed total scores using the method introduced by Haberman (2008) <doi:10.3102/1076998607302636>.

Usage

```
subscore.x(test.data)
```

Arguments

`test.data`	A list that contains item responses of all subtests and the entire test, which can be obtained using function 'data.prep'.

Value

`summary`	Summary of obtained subscores (e.g., mean, sd).
`PRMSE`	PRMSEs of obtained subscores (for Haberman's methods only).
`subscore.original`	Original observed subscores and total score.
`subscore.x`	Subscores that are estimated based on the observed total score.

References

Haberman, S. J. (2008). "When can subscores have value?." Journal of Educational and Behavioral Statistics, 33(2), 204-229.doi:10.3102/1076998607302636

Examples

```
test.data<-data.prep(scored.data,c(3,15,15,20),
                     c("Algebra","Geometry","Measurement", "Math"))

subscore.x(test.data)

subscore.x(test.data)$summary
subscore.x(test.data)$PRMSE
subscore.x(test.data)$Correlation
subscore.x(test.data)$Disattenuated.correlation
subscore.x(test.data)$subscore.x
```

`test.data` *A list of objects that include both test information and subscores.*

Description

This list consists of four objects. The first three objects are item responses on the three subscales (algebra, geometry, and measurement). The fourth object is the response data on the total test.

Usage

```
data("test.data")
```

Format

The format is: A list with 4 objects:

$ Algebra :'data.frame': 150 obs. of 15 variables:

$ Geometry :'data.frame': 150 obs. of 15 variables:

$ Measurement:'data.frame': 150 obs. of 20 variables:

$ Math :'data.frame': 150 obs. of 50 variables:

Details

Algebra: Responses of 150 participants to 15 items; Geometry: Responses of 150 participants to 15 items. Measurement: Responses of 150 participants to 20 items; Math: Responses of 150 participants to 20 items.

Examples

```
data(test.data)
# maybe str(test.data); plot(test.data) ...
```

`TIMSS11G8M.data` *The 2011 TIMSS Grade 8 Mathematics Assessment Dataset*

Description

The TIMSS dataset used in Dai, Svetina, and Wang (2017) (doi:10.3102/1076998617716462). It contained responses from 765 students to 32 items with 6 to 9 items on each of the subscales of (1) number (Q1 to Q9), (2) algebra (Q10 to Q18), (3) geometry (Q19 to Q24), and (4) data and chance (Q25 to Q30). Omitted responses were treated as incorrect.

Usage

```
data("TIMSS11G8M.data")
```

Format

A data frame with 765 observations on the following 32 variables.

`Q1` a numeric vector

`Q2` a numeric vector

`Q3` a numeric vector

`Q4` a numeric vector

`Q5` a numeric vector

`Q6` a numeric vector

`Q7` a numeric vector

`Q8` a numeric vector

`Q9` a numeric vector

`Q10` a numeric vector

`Q11` a numeric vector

`Q12` a numeric vector

`Q13` a numeric vector

`Q14` a numeric vector

`Q15` a numeric vector

`Q16` a numeric vector

`Q17` a numeric vector

`Q18` a numeric vector

`Q19` a numeric vector

`Q20` a numeric vector

`Q21` a numeric vector

`Q22` a numeric vector

`Q23` a numeric vector

`Q24` a numeric vector

`Q25` a numeric vector

`Q26` a numeric vector

`Q27` a numeric vector

`Q28` a numeric vector

`Q29` a numeric vector

`Q30` a numeric vector

`Q31` a numeric vector

`Q32` a numeric vector

Source

Dai, S., Svetina, D., & Wang, X. (2017). "Reporting subscores using R: A software review." Journal of Educational and Behavioral Statistics. 42(2), 617-638. doi: 10.3102/1076998617716462.

Examples

```
data(TIMSS11G8M.data)
# maybe str(TIMSS11G8M.data); plot(TIMSS11G8M.data) ...
```

`Yen.OPI` *Estimating true subscores using Yen's OPI*

Description

This function estimates subscores using Yen's Objective Performance Index (OPI; Yen, 1987) <https://www.ets.org/research/policy_research_reports/publications/paper/1987/hrap>. Yen's OPI (Yen, 1987) is a procedure combining Bayesian method and item response theory (IRT; Embretson & Reise, 2000 <https://psycnet.apa.org/record/2000-03918-000>; Reckase, 1997 <doi: 10.1177/0146621697211002>). This method pulls an examinee's performance on a certain objective (i.e., subscale) towards his/her total test performance in order to get a more stable and precise objective subscore estimate.

Usage

```
Yen.OPI(test.data)
```

Arguments

`test.data` A list that contains item responses of all subtests and the entire test, which can be obtained using function 'data.prep'.

Value

`summary` It contains statistical summary of OPI (mean & sd).

`OPI` Estimated OPI values

References

Embretson, S. E., & Reise, S. P. (2013). "Item response theory". Mahwah, NJ: Lawrence Erlbaum Associates, Inc. https://psycnet.apa.org/record/2000-03918-000.

Reckase, M. D. (1997). "The past and future of multidimensional item response theory". Applied Psychological Measurement, 21(1), 25-36. doi: 10.1177/0146621697211002.

Yen, W. M. (1987, June). "A Bayesian/IRT index of objective performance". Paper presented at annual meeting of the Psychometric Society, Montreal, Quebec, Canada. https://www.ets.org/research/policy_research_reports/

Examples

```
    test.data<-data.prep(scored.data,c(3,15,15,20),
                         c("Algebra","Geometry","Measurement", "Math"))

    Yen.OPI(test.data)
```

`Yen.Q3` *Computing Yen's Q3 statistic for unidimensional Rasch, 1-, 2-, and 3-PL logistic IRT models*

Description

This function calculates Yen's Q3 statistics as introduced in Yen (1984) <doi: 10.1177/014662168400800201> and Yen (1993) <doi: 10.1111/j.1745-3984.1993.tb00423.x> for unidimensional Rasch, 1-, 2-, and 3-PL logistic IRT models to assess the local independence assumption.

Usage

```
Yen.Q3(scored.data, IRT.model = "2pl")
```

Arguments

`scored.data` Item response data with rows as individuals and columns as items.

`IRT.model` IRT model ('Rasch', '1pl', '2pl', or '3pl') to be used.The default option is 2pl.

Value

`Q3` A matrix of Q3 statistics

`Q3.weighted` A matrix of Q3 statistics as obtained by weighting the residual values to reflect the number of examinees with each response pattern.

References

Yen, W. M. (1984). "Effects of local item dependence on the fit and equating performance of the three-parameter logistic model." Applied Psychological Measurement, 8(2), 125-145. doi: 10.1177/014662168400800201.

Yen, W. M. (1993). "Scaling performance assessments: Strategies for managing local item dependence. " ournal of educational measurement, 30(3), 187-213. doi: 10.1111/j.1745-3984.1993.tb00423.x.

Examples

```
Yen.Q3(scored.data,IRT.model="2pl")

Yen.Q3(scored.data)$Q3
Yen.Q3(scored.data)$Q3.weighted
```

Index

∗ **datasets**
    scored.data, 5
    test.data, 12
    TIMSS11G8M.data, 12

CTTsub, 2

data.prep, 4

scored.data, 5
subscore.corr, 7
subscore.s, 8
subscore.sx, 9
subscore.Wainer, 10
subscore.x, 11

test.data, 12
TIMSS11G8M.data, 12

Yen.OPI, 14
Yen.Q3, 15