Package ‘DFIT’

August 17, 2021

Title  Differential Functioning of Items and Tests
Version  1.1
Date  2021-08-16
Author  Victor H. Cervantes <vhcervantesb@unal.edu.co>
Maintainer  Victor H. Cervantes <vhcervantesb@unal.edu.co>
Imports  msm, mirt, simex, mvtnorm, ggplot2
Description  A set of functions to perform Raju, van der Linden and Fleer's
and Tests (DFIT) analyses. It includes functions to use the Monte Carlo Item
Parameter Replication approach (Oshima, Raju, & Nanda, 2006, <doi:10.1111/j.1745-
3984.2006.00001.x>)
for obtaining the associated statistical significance
tests cut-off points. They may also be used for a priori and post-hoc power
License  GPL (>= 2)
LazyData  true
NeedsCompilation  no
Encoding  UTF-8
RoxygenNote  7.1.1
Repository  CRAN
Date/Publication  2021-08-17 05:20:02 UTC

R topics documented:

Ase1pl . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
Ase2pl . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
Ase3pl . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
AseIrt . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
Bound3PlIpr . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
Bound4PlIpr . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
Calculate1plProb . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
Ase1pl

Calculates the asymptotic variance for difficulty parameter estimates under the 1pl model

Description

Calculates the asymptotic variance for difficulty parameter estimates under the 1pl model
Usage

Ase1pl(
  itemParameters,
  distribution = "norm",
  distributionParameters = list(mean = 0, sd = 1),
  logistic = TRUE,
  sampleSize = 1,
  subdivisions = 5000
)

Arguments

itemParameters A matrix or vector containing the item difficulties.
distribution A string character indicating the generic name for the assumed distribution.
distributionParameters A list of extra parameters for the distribution function.
logistic A logical indicating whether the logistic or the normal metric should be used.
sampleSize A value indicating the sample size.
subdivisions A numeric value stating the maximum number of subdivisions for adaptive quadrature.

Value

ase A list containing the asymptotic variances for each item

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Ase2pl

Calculates the asymptotic covariance matrix of item parameter estimates under the 2pl model

Description

Calculates the asymptotic covariance matrix of item parameter estimates under the 2pl model
Usage

Ase2pl(
  itemParameters,
  distribution = "norm",
  distributionParameters = list(mean = 0, sd = 1),
  logistic = TRUE,
  sampleSize = 1,
  subdivisions = 5000
)

Arguments

- **itemParameters**: A matrix or vector containing the item difficulties.
- **distribution**: A string character indicating the generic name for the assumed distribution.
- **distributionParameters**: A list of extra parameters for the distribution function.
- **logistic**: A logical indicating whether the logistic or the normal metric should be used.
- **sampleSize**: A value indicating the sample size.
- **subdivisions**: A numeric value stating the maximum number of subdivisions for adaptive quadrature.

Value

ase A list containing the asymptotic matrices for each item

Author(s)

Victor H. Cervantes <vhcervantesb@unal.edu.co>

References


Description

Calculates the asymptotic covariance matrix of item parameter estimates under the 3pl model
Usage

Ase3pl(
    itemParameters,
    distribution = "norm",
    distributionParameters = list(mean = 0, sd = 1),
    logistic = TRUE,
    sampleSize = 1,
    subdivisions = 5000
)

Arguments

itemParameters  A matrix or vector containing the item difficulties.
distribution    A string character indicating the generic name for the assumed distribution.
distributionParameters  A list of extra parameters for the distribution function.
logistic        A logical indicating whether the logistic or the normal metric should be used.
sampleSize      A value indicating the sample size.
subdivisions    A numeric value stating the maximum number of subdivisions for adaptive quadrature.

Value

ase A list containing the asymptotic matrices for each item

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


AseIrt    Calculates the asymptotic covariance matrices for item parameters according with the IRT model.

Description

Calculates the asymptotic covariance matrices for item parameters according with the IRT model.
Usage

AseIrt(
  itemParameters,
  distribution = "norm",
  distributionParameters = list(mean = 0, sd = 1),
  logistic = TRUE,
  sampleSize = 1,
  irtModel = "3pl",
  subdivisions = 5000
)

Arguments

  itemParameters  A matrix or vector containing the item difficulties.
  distribution    A string character indicating the generic name for the assumed distribution. Defaults to 'norm' for normal distribution.
  distributionParameters  A list of extra parameters for the distribution function.
  logistic         A logical indicating whether the logistic or the normal metric should be used.
  sampleSize       A value indicating the sample size.
  irtModel         A string stating the IRT model for all items.
  subdivisions      A numeric value stating the maximum number of subdivisions for adaptive quadrature.

Value

  ase A list containing the asymptotic matrices for each item

Author(s)

  Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Examples

  # # Not run
  # #
  # # data(dichotomousItemParameters)
  # # threePlParameters <- dichotomousItemParameters
  # # isNot3Pl <- ((dichotomousItemParameters[[‘focal’]][, 3] == 0) |
  # #   (dichotomousItemParameters[[‘reference’]][, 3] == 0))
  # #
  # # threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
## Description

Makes all simulated guessing values from a 3PL model that are outside the $[0, 1]$ interval to be 0 or 1.

## Usage

```
Bound3PlIpr(itemParameterList)
```

## Arguments

- **itemParameterList**
  
  A list where each element is a list containing "focal" and "reference" item parameters from a 3PL model. Item parameters are assumed to be on the same scale. Item parameters for each group should be a matrix with nrow equal to the number of items.

## Value

- **itemParameterList**
  
  A list where each element is a list containing "focal" and "reference" item parameters where guessing parameters outside the $[0, 1]$ interval are changed by 0 and 1.

## Author(s)

Victor H. Cervantes <vhcervantesb@unal.edu.co>
Examples

```
# Not run
#
# data(dichotomousItemParameters)
# threePlParameters <- dichotomousItemParameters
# isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
# (dichotomousItemParameters[['reference']][, 3] == 0))
#
# threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
# threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
# threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
# threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
# threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
# threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
# threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]
#
# threePlAse <- list()
# threePlAse[['focal']] <- AseIrt(itemParameters = threePlParameters[['focal']],
# logistic = TRUE,
# sampleSize = 10000,
# irtModel = "3pl")
# threePlAse[['reference']] <- AseIrt(itemParameters = threePlParameters[['reference']],
# logistic = TRUE,
# sampleSize = 15000,
# irtModel = "3pl")
#
# set.seed(41568)
# threePlIpr <- Ipr(itemParameters = threePlParameters, itemCovariances = threePlAse,
# nReplicates = 100)
# threePlIpr <- Bound3PlIpr(threePlIpr)
```

---

**Bound4PlIpr**

Takes item parameters from Ipr and forces guessing to lie between 0 and 1

**Description**

Makes all simulated guessing values from a 3PL model that are outside the [0, 1] interval to be 0 or 1.

**Usage**

`Bound4PlIpr(itemParameterList)`
Arguments

    itemParameterList
    A list where each element is a list containing "focal" and "reference" item parameters from a 3PL model. Item parameters are assumed to be on the same scale. Item parameters for each group should be a matrix with nrow equal to the number of items.

Value

    itemParameterList A list where each element is a list containing "focal" and "reference" item parameters where guessing parameters outside the [0, 1] interval are changed by 0 and 1.

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

Examples

    # Not run
    #
    # data(dichotomousItemParameters)
    # threePlParameters <- dichotomousItemParameters
    # isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
    #             (dichotomousItemParameters[['reference']][, 3] == 0))
    #
    # threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
    # threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
    #
    # threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
    # threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
    # threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
    #
    # threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
    # threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]
    #
    # threePlAse <- list()
    # threePlAse[['focal']] <- AseIrt(itemParameters = threePlParameters[['focal']],
    #                             logistic = TRUE,
    #                             sampleSize = 10000,
    #                             irtModel = "3pl")
    # threePlAse[['reference']] <- AseIrt(itemParameters = threePlParameters[['reference']],
                                          logistic = TRUE,
                                          sampleSize = 15000,
                                          irtModel = "3pl")
    #
    # set.seed(41568)
    # threePlIpr <- Ipr(itemParameters = threePlParameters, itemCovariances = threePlAse,
    #                  nReplicates = 100)
    # threePlIpr <- Bound3PlIpr(threePlIpr)
Calculate1plProb  \textit{Calculates the item success probability under the 1PL model.}

\textbf{Description}
Calculates the item success probability under the 1PL model.

\textbf{Usage}
\texttt{Calculate1plProb(thetaValue, itemParameters, logistic = TRUE)}

\textbf{Arguments}
- \texttt{thetaValue} A numeric value or array for the theta (ability) value(s) where the difference will be calculated
- \texttt{itemParameters} A vector or column matrix containing the numeric values of item difficulties
- \texttt{logistic} A logical value stating if the IRT model will use the logistic or the normal metric.

\textbf{Value}
probabilities A numeric matrix with the probabilities on each thetaValue for each item.

\textbf{Author(s)}
Victor H. Cervantes \texttt{<vhcervantesb at unal.edu.co>}

\textbf{References}

Calculate2plProb  \textit{Calculates the item success probability under the 2PL model.}

\textbf{Description}
Calculates the item success probability under the 2PL model.

\textbf{Usage}
\texttt{Calculate2plProb(thetaValue, itemParameters, logistic = TRUE)}
Calculate3plProb

Arguments

thetaValue A numeric value or array for the theta (ability) value(s) where the difference will be calculated

itemParameters A matrix containing the numeric values of item discriminations on the first column and item difficulties on the second

logistic A logical value stating if the IRT model will use the logistic or the normal metric.

Value

probabilities A numeric matrix with the probabilities on each thetaValue for each item.

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Calculate3plProb  Calculates the item success probability under the 3PL model.

Description

Calculates the item success probability under the 3PL model.

Usage

Calculate3plProb(thetaValue, itemParameters, logistic = TRUE)

Arguments

thetaValue A numeric value or array for the theta (ability) value(s) where the difference will be calculated

itemParameters A matrix containing the numeric values of item discriminations on the first column, item difficulties on the second and item guessing parameters on the third

logistic A logical value stating if the IRT model will use the logistic or the normal metric.

Value

probabilities A numeric matrix with the probabilities on each thetaValue for each item.

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>
Calculate4plProb

Calculates the item success probability under the 4PL model.

Description

Calculates the item success probability under the 4PL model.

Usage

Calculate4plProb(thetaValue, itemParameters, logistic = TRUE)

Arguments

thetaValue A numeric value or array for the theta (ability) value(s) where the difference will be calculated
itemParameters A matrix containing the numeric values of item discriminations on the first column, item difficulties on the second, item guessing parameters on the third, and item upper asymptote on the fourth
logistic A logical value stating if the IRT model will use the logistic or the normal metric.

Value

probabilities A numeric matrix with the probabilities on each thetaValue for each item.

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References

**CalculateGrmExp**

*Calculates the expected item score under the GRM model.*

**Description**

Calculates the expected item score under the GRM model.

**Usage**

`CalculateGrmExp(thetaValue, itemParameters, logistic = TRUE)`

**Arguments**

- `thetaValue`: A numeric value or array for the theta (ability) value(s) where the difference will be calculated.
- `itemParameters`: A matrix containing the numeric values of item discriminations on the first column and category thresholds on the rest columns where the (column position - 1) indicates the category score or weight.
- `logistic`: A logical value stating if the IRT model will use the logistic or the normal metric.

**Value**

- `expectedScore`: A numeric matrix with the expected score on each thetaValue for each item.

**Author(s)**

Victor H. Cervantes <vhcervantesb at unal.edu.co>

**References**


---

**CalculateItemDifferences**

*Calculates the differences between two item option characteristic curves for all options (minus one).*

**Description**

Calculates the differences between two item option characteristic curves for all options (minus one).
Usage

CalculateItemDifferences(
  thetaValue,
  itemParameters,
  irtModel = "2pl",
  logistic = TRUE
)

Arguments

thetaValue  A numeric value or array for the theta (ability) value(s) for which the difference will be calculated

itemParameters A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should me a matrix with nrow equal to the number of items.

irtModel A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".

logistic A logical value stating if the IRT model will use the logistic or the normal metric.

Value

difference A numeric matrix with the differences on probabilities or on expected score for each item between focal and reference groups.

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

CalculatePcmExp

Calculates the expected item score under the (G)PCM model.

Description

Calculates the expected item score under the (G)PCM model.

Usage

CalculatePcmExp(thetaValue, itemParameters, logistic = TRUE)

Arguments

thetaValue  A numeric value or array for the theta (ability) value(s) where the difference will be calculated

itemParameters A matrix containing the numeric values of item discriminations on the first column and category thresholds on the rest columns where the (column position - 1) indicates the category score or weight.

logistic A logical value stating if the IRT model will use the logistic or the normal metric.
Value

expectedScore A numeric matrix with the expected score on each thetaValue for each item.

Author(s)

Victor H. Cervantes <vhcervantesb@unal.edu.co>

References


Cdif

Calculates CDIF index for an item with given item parameters of focal and reference groups.

Description

Calculates CDIF index for an item with given item parameters of focal and reference groups.

Usage

Cdif(
  itemParameters,
  irtModel = "2pl",
  focalAbilities = NULL,
  focalDistribution = "norm",
  subdivisions = 5000,
  logistic = TRUE,
  focalDistrExtra = list(mean = 0, sd = 1)
)

Arguments

itemParameters A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should me a matrix with nrow equal to the number of items.

irtModel A string stating the irmModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".

focalAbilities If NULL, NCDIF is calculated by numerical integration of focal distribution. If not NULL, it must be a numerical vector containing the abilities for the individuals in the focal group.
focalDistribution
A string stating the distribution name to be used for integrating. Only used if focalAbilities is NULL.

subdivisions
A numeric value indicating the number of subdivisions for numerical integration. Only used if focalAbilities is NULL.

logistic
A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.

focalDistrExtra
Extra parameters for the focal group distribution function if needed.

Value
cdif Numeric vector with the CDIF index value for each item.

Author(s)
Victor H. Cervantes <vhervantesb at unal.edu.co>

References

Examples

```r
# Not run
#
# data(dichotomousItemParameters)
#
# threePLParameters <- dichotomousItemParameters
# isNot3PL <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
# (dichotomousItemParameters[['reference']][, 3] == 0))
#
# threePLParameters[['focal']][, 3] <- threePLParameters[['focal']][, 3] + 0.1
# threePLParameters[['reference']][, 3] <- threePLParameters[['reference']][, 3] + 0.1
# threePLParameters[['focal']][, 2] <- threePLParameters[['focal']][, 2] + 1.5
# threePLParameters[['reference']][, 2] <- threePLParameters[['reference']][, 2] + 1.5
# threePLParameters[['focal']][, -c(12, 16, 28), ] <- threePLParameters[['focal']][, -c(12, 16, 28), ]
#
# threePLCdif <- Cdif(itemParameters = dichotomousItemParameters, irtModel = '3pl',
# focalAbilities = NULL, focalDistribution = "norm",
# subdivisions = 5000, logistic = TRUE)
```
CheckDiscriminations

Identifies items with nonpositive discrimination

Description
Identifies items with nonpositive discrimination

Usage
CheckDiscriminations(itemParameters)

Arguments
itemParameters  A vector or column matrix containing the numeric values of item difficulties

Value
message A character string used to signal items with nonpositive discriminations

Author(s)
Victor H. Cervantes <vhcervantesb at unal.edu.co>

CheckGuessings

Identifies items with guessing outside \([0, 1]\)

Description
Identifies items with guessing outside \([0, 1]\)

Usage
CheckGuessings(itemParameters)

Arguments
itemParameters  A vector or column matrix containing the numeric values of item difficulties

Value
message A character string used to signal items iadmissible guessing parameters

Author(s)
Victor H. Cervantes <vhcervantesb at unal.edu.co>
CheckUpper

*Identifies items with upper asymptote outside [0, 1]*

**Description**

Identifies items with upper asymptote outside [0, 1]

**Usage**

CheckUpper(itemParameters)

**Arguments**

- **itemParameters**: A vector or column matrix containing the numeric values of item difficulties

**Value**

- **message**: A character string used to signal items admissible guessing parameters

**Author(s)**

Victor H. Cervantes <vhcervantesb at unal.edu.co>

CrossedProbabilities

*Calculates the crossed probabilities associated with the numerator and denominator of the odds-ratio under dichotomous IRT models*

**Description**

Calculates the crossed probabilities associated with the numerator and denominator of the odds-ratio under dichotomous IRT models

**Usage**

CrossedProbabilities(thetaValue, itemParameters, logistic, irtModel = "3pl")

**Arguments**

- **thetaValue**: A numeric value or array for the theta (ability) value(s) for which the odds will be calculated
- **itemParameters**: A list containing the "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale.
- **logistic**: A logical indicating whether the logistic or the normal metric should be used.
- **irtModel**: A string stating the IrtModel used. May be one of "1pl", "2pl", or "3pl".
CutoffIpr

Value

A list containing the crossed products for the 'num' the numerator, 'den' the denominator for the odds-ratio, and 'or' the odds-ratio

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Description

Calculates a given quantile cut-off point for each item on the IPR estimated items statistics. This function may produce the cut-off points for the NCDIF index, Signed and Unsigned Area Measures and the Mantel-Haenszel statistic based on the Monte Carlo Item parameter replication approach. The quantiles may be calculated directly on the output from the IprNcdif, IprSam, IprUam, and IprMh functions; the may be calculated by obtaining the corresponding statistics for the item parameters simulated under the IPR approach; or by obtaining both the simulated item parameters and the statistics based on the item parameter values and their corresponding covariance matrices for the parameter estimates. In the latter case, the user may choose to obtain the IPR simulated item parameters based only on the focal group’s covariance matrix as proposed by Oshima et al. (2006), or both focal and reference groups’ matrices as proposed by Cervantes (2012).

Usage

CutoffIpr(iprStatistics = NULL, quantiles, statistic = "ncdif", itemParameterList = NULL, irtModel = "2pl", focalAbilities = NULL, focalDistribution = "norm", focalDistrExtra = list(mean = 0, sd = 1), referenceDistribution = "norm", referenceDistrExtra = list(mean = 0, sd = 1), groupRatio = 1, subdivisions = 5000, logistic = TRUE, itemParameters = NULL,
CutoffIpr

itemCovariances = NULL,
nullGroup = NULL,
focalSampleSize = NULL,
referenceSampleSize = NULL,
nReplicates = 5000
)

Arguments

iprStatistics  A numeric matrix with the statistics obtained for the simulated IPR item parameters or a list containing all the elements of the output of this function. If not NULL they will be used for calculating the cut-off points.
quantiles  A numeric vector with the quantiles to be calculated.
statistic  A character indicating which statistic will the cut-off point will be obtained for. If iprStatistics are provided, it is up to the user to correctly specify this string for it will only be informative; otherwise, it will be used to identify the statistic to be calculated. Should be one of "ncdif", "sam", "uam" or "mh".
itemParameterList  A list where each element is a list containing "focal" and "reference" item Parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should be a matrix with nrow equal to the number of items. Not used if iprStatistics are not NULL. If itemParameterList is not NULL, the statistic indicated with the argument "statistic" will be obtained for the set of itemParameterList, the corresponding arguments may be provided.
irtModel  A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm". Not used if iprStatistics are not NULL.
focalAbilities  Only used if statistic is "ncdif". If NULL, NCDIF is calculated by numerical integration of focal distribution. If not NULL, must be a numerical vector containing the abilities for the individuals in the focal group.
focalDistribution  A string stating the distribution assumed for the focal group. Not used if iprStatistics are not NULL.
focalDistrExtra  A list stating the extra parameters needed by the focal distribution function. Not used if iprStatistics are not NULL.
referenceDistribution  A string stating the distribution assumed for the reference group. Not used if iprStatistics are not NULL.
referenceDistrExtra  A list stating the extra parameters needed by the reference distribution function. Not used if iprStatistics are not NULL.
groupRatio  A positive value indicating how many members of the reference group are expected for each member of the focal group. Only used if iprStatistics are NULL and statistic is "mh".
subdivisions  A numeric value indicating the number of subdivisions for numerical integration. Only used if focalAbilities and iprStatistics are NULL.
CutoffIpr

**logistic**
A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.

**itemParameters**
A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should me a matrix with nrow equal to the number of items. Only used if both iprStatistics and itemParameterList are NULL. If used an itemParameterList from applying the IPR procedure will be simulated and the "statistic" will be calculated.

**itemCovariances**
Either a list containing "focal" and "reference" lists of matrices of covariance for item estimates or the string "asymptotic". Defaults to NULL. Only used if iprStatistics and itemParameterList are NULL, in all other cases the itemCovariances element of the returned list is equal to what is provided as value for these arguments.

**nullGroup**
If different from NULL and itemParameterList is NULL, a string equal to 'focal' or 'reference' to indicate which set of item parameters from itemParameters should be taken for the null hypothesis. If equal to NULL, itemParameterList will be generated using the given itemParameters for both groups.

**focalSampleSize**
A positive integer indicating the size of the focal group. Only used if itemCovariances is 'asymptotic'. Defaults to NULL.

**referenceSampleSize**
A positive integer indicating the size of the reference group. Only used if itemCovariances is 'asymptotic'. Defaults to NULL.

**nReplicates**
A numeric value indicating the number of replications to perform. Only used if iprStatistics and itemParameterList are NULL.

**Value**

cutoff A list containing: 'itemParameters', NULL if not provided as argument, 'itemCovariances', NULL if not provided as argument, 'itemParameterList', NULL unless calculated from 'itemParameters' or provided as argument, 'iprStatistics' the matrix of 'statistics' provided as argument or calculated from 'itemParameterList', 'statistic' for which the IPR approach is used according to the provided argument, 'quantiles' the vector or matrix of calculated quantiles for each item

**Author(s)**
Victor H. Cervantes <vhcervantesb at unal.edu.co>

**References**


Examples
## Not run
##
## data(dichotomousItemParameters)
## threePlParameters <- dichotomousItemParameters
## isNot3Pl <- ((dichotomousItemParameters[['focal']][[3]] == 0) |
##             (dichotomousItemParameters[['reference']][[3]] == 0))
##
## threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
## threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
## threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
## threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
## threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
## threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
## threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]
##
## threePlAse <- list()
## threePlAse[['focal']] <- AseIrt(itemParameters = threePlParameters[['focal']],
##                               logistic = TRUE,
##                               sampleSize = 10000,
##                               irtModel = "3pl")
## threePlAse[['reference']] <- AseIrt(itemParameters = threePlParameters[['reference']],
##                                  logistic = TRUE,
##                                  sampleSize = 15000,
##                                  irtModel = "3pl")
##
## set.seed(41568)
##
## threePlIrprCutoff <- CutoffIrpr(itemParameters = threePlParameters,
##                                itemCovariances = threePlAse, nullGroup = 'focal',
##                                nReplicates = 1000, statistic = 'ncdif', irtModel = '3pl')

DeltaMhIrt

Obtains the ETS Delta measure for Mantel-Haenszel DIF statistic effect size.
**Description**

Obtains the ETS Delta measure for Mantel-Haneszel DIF statistic effect size.

**Usage**

```r
DeltaMhIrt(mh, logistic = FALSE)
```

**Arguments**

- `mh` A numeric vector containing the MH statistic values
- `logistic` A logical indicating whether the logistic or the normal metric should be used.

**Value**

- `delta` A numeric vector containing the delta values

**Author(s)**

Victor H. Cervantes <vhcervantesb at unal.edu.co>

**References**


**Examples**

```r
data(dichotomousItemParameters)
threePlParameters <- dichotomousItemParameters
isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
(dichotomousItemParameters[['reference']][, 3] == 0))

threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
threePlParameters[['focal']] <- threePlParameters[['focal']][[-c(12, 16, 28), ]
threePlParameters[['reference']] <- threePlParameters[['reference']][[-c(12, 16, 28), ]

threePlMh <- IrtMh(itemParameters = threePlParameters, irtModel = "3pl",
  focalDistribution = "norm", referenceDistribution = "norm",
  focalDistrExtra = list(mean = 0, sd = 1),
  referenceDistrExtra = list(mean = 0, sd = 1), groupRatio = 1,
  logistic = FALSE)

delta3pl <- DeltaMhIrt(threePlMh)
```
DFIT provides functions for calculating the differential item and test functioning proposed by Raju et al. (1995).

Details

DFIT provides a set of functions to calculate the noncompensatory (NCDIF), compensatory (CDIF) and test level (DTF) differential functioning indices for items and tests under Raju's (Raju, et al. 1995) DFIT framework. It also provides functions for obtaining cut-off points for identifying differential functioning for these indices following the Monte Carlo Item Parameter Replication approach proposed by Oshima et al. (2006).

This package also improves upon available DFIT software by allowing the covariance matrices for both focal and reference groups to be used. This improves the obtained cut-off points, which result in type I error rates at the nominal level, and increased power, when compared to the cut-off points obtained when using only the focal group item parameter estimates and their estimate covariances (Cervantes, 2012). Furthermore, this package includes functions for obtaining the asymptotic covariance matrices of item parameter estimates (currently only for dichotomous IRT models) and for calculating the DFIT indices base on the focal group distribution as well as ability estimates for a sample from the focal population are included; these enable ad hoc and a priori power calculations for given item parameters and sample sizes to be possible with this package.

References


dichotomousItemParameters

Sets of focal and reference item parameters from Wright (2011).

Description


Usage

data(dichotomousItemParameters)

Format

A list with ‘focal’ and ‘reference’ elements. Each is a matrix 1 row per item by 3 columns: item discrimination, difficulty and guessing parameters.

Source

This data set contains the item parameters based on those found in Wright, K. (2011).

References

Dtf

Calculates DTF index for a set of items with given item parameters of focal and reference groups.

Description

Calculates DTF index for a set of items with given item parameters of focal and reference groups.

Usage

Dtf(
  cdif = NULL,
  itemParameters = NULL,
  irtModel = "2pl",
  focalAbilities = NULL,
  focalDistribution = "norm",
  subdivisions = 5000,
  logistic = TRUE,
  focalDistrExtra = list(mean = 0, sd = 1)
)

Arguments

cdif
A numeric vector of CDIF values for the test items. If NULL it is calculated using itemParameters and the other arguments.

itemParameters
A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Only used if cdif is NULL. Item parameters for each group should me a matrix with nrow equal to the number of items.

irtModel
A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm". Only used if cdif is NULL.

focalAbilities
If NULL, CDIF is calculated by numerical integration of focal distribution. If not NULL, it must be a numerical vector containing the abilities for the individuals in the focal group. Only used if cdif is NULL.

focalDistribution
A string stating the distribution name to be used for integrating. Only used if focalAbilities and cdif are NULL.

subdivisions
A numeric value indicating the number of subdivisions for numerical integration. Only used if focalAbilities and cdif are NULL.

logistic
A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used. Only used if cdif is NULL.

focalDistrExtra
Extra parameters for the focal group distribution function if needed.
Value
dtf Numeric vector with the CDIF index value for each item.

Author(s)
Victor H. Cervantes <vhcervantesb at unal.edu.co>

References

Examples

```
# # Not run
# #
# # data(dichotomousItemParameters)
# #
# # threePlParameters <- dichotomousItemParameters  
# # isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |  
# # (dichotomousItemParameters[['reference']][, 3] == 0))
# #
# # threePlParameters[['focal']] <- threePlParameters[['focal']][isNot3Pl, ]  
# # threePlParameters[['reference']] <- threePlParameters[['reference']][isNot3Pl, ]  
# # threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1  
# # threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1  
# # threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5  
# # threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]  
# # threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]  
# #
# # threePlCdif <- Cdif(itemParameters = threePlParameters, irtModel = '3pl',  
# # focalAbilities = NULL, focalDistribution = "norm",  
# # subdivisions = 5000, logistic = TRUE)  
# # threePlDtf <- Dtf(cdif = threePlCdif)
```

---

**Extract2PLMirt**

Extract item discrimination and difficulties and estimate covariance estimates for 2PL items from a fitted mirt object for one or two groups

**Description**

Extract item discrimination and difficulties and estimate covariance estimates for 2PL items from a fitted mirt object for one or two groups

**Usage**

Extract2PLMirt(mod, focal = NULL, reference = NULL)
Arguments

mod A mirt object containing the fit of unidimensional model.

focal Character. Required if mod is MultipleGroupClass, focal should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.

reference Character. Required if mod is MultipleGroupClass, reference should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.

Value

If mod contains any itemtype == "2PL", a list with the item parameters and the estimate covariances (if available). If mod is SingleGroupClass, the list contains the item parameters as a matrix and the covariances as a list. If mod is MultipleGroupClass, the list contains the item parameters and covariances for the focal and reference groups only.

Examples

```r
library(mirt)
data <- expand.table(LSAT7)
(mod1 <- mirt(data, model = 1, itemtype = "2PL", SE = TRUE))
(DFIT:::Extract2PLMirt(mod1))
```

Extract3PLMirt

Extract item discrimination, difficulties, and guessing parameters and estimate covariance estimates for 3PL items from a fitted mirt object for one or two groups

Usage

Extract3PLMirt(mod, focal = NULL, reference = NULL)

Arguments

mod A mirt object containing the fit of unidimensional model.

focal Character. Required if mod is MultipleGroupClass, focal should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.

reference Character. Required if mod is MultipleGroupClass, reference should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.
Value

If mod contains any itemtype == "3PL", a list with the item parameters and the estimate covariances (if available). If mod is SingleGroupClass, the list contains the item parameters as a matrix and the covariances as a list. If mod is MultipleGroupClass, the list contains the item parameters and covariances for the focal and reference groups only.

Examples

library(mirt)
data <- expand.table(LSAT7)
(mod1 <- mirt(data, model = 1, itemtype = "3PL", SE = TRUE))
(DFIT::Extract3PLMirt(mod1))

Extract4PLMirt

Extract item discrimination, difficulties, guessing, and upper asymptote parameters and estimate covariance estimates for 4PL items from a fitted mirt object for one or two groups

Description

Extract item discrimination, difficulties, guessing, and upper asymptote parameters and estimate covariance estimates for 4PL items from a fitted mirt object for one or two groups

Usage

Extract4PLMirt(mod, focal = NULL, reference = NULL)

Arguments

mod A mirt object containing the fit of unidimensional model.
focal Character. Required if mod is MultipleGroupClass, focal should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.
reference Character. Required if mod is MultipleGroupClass, reference should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.

Value

If mod contains any itemtype == "4PL", a list with the item parameters and the estimate covariances (if available). If mod is SingleGroupClass, the list contains the item parameters as a matrix and the covariances as a list. If mod is MultipleGroupClass, the list contains the item parameters and covariances for the focal and reference groups only.
Examples

library(mirt)
data <- expand.table(LSAT7)
(mod1 <- mirt(data, model = 1, itemtype = "4PL", SE = TRUE))
(DFIT:::Extract4PLMirt(mod1))

ExtractGPCMMirt

Extract item discrimination and difficulties and estimate covariance estimates for GPCM items from a fitted mirt object for one or two groups

Description

Extract item discrimination and difficulties and estimate covariance estimates for GPCM items from a fitted mirt object for one or two groups

Usage

ExtractGPCMMirt(mod, focal = NULL, reference = NULL)

Arguments

mod A mirt object containing the fit of unidimensional model.
focal Character. Required if mod is MultipleGroupClass, focal should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.
reference Character. Required if mod is MultipleGroupClass, reference should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.

Value

If mod contains any itemtype == "gpcm", a list with the item parameters and the estimate covariances (if available). If mod is SingleGroupClass, the list contains the item parameters as a matrix and the covariances as a list. If mod is MultipleGroupClass, the list contains the item parameters and covariances for the focal and reference groups only.

Examples

library(mirt)
(mod1 <- mirt(Science, model = 1, itemtype = "gpcm", SE = TRUE))
(DFIT:::ExtractGPCMMirt(mod1))
ExtractGRMMirt

Extract item discrimination and difficulties and estimate covariance estimates for GRM items from a fitted mirt object for one or two groups

Description

Extract item discrimination and difficulties and estimate covariance estimates for GRM items from a fitted mirt object for one or two groups

Usage

ExtractGRMMirt(mod, focal = NULL, reference = NULL)

Arguments

- `mod`: A mirt object containing the fit of unidimensional model.
- `focal`: Character. Required if `mod` is `MultipleGroupClass`, focal should coincide with the label for the focal group. If `mod` is `SingleGroupClass`, it is ignored.
- `reference`: Character. Required if `mod` is `MultipleGroupClass`, reference should coincide with the label for the focal group. If `mod` is `SingleGroupClass`, it is ignored.

Value

If `mod` contains any `itemtype` == "graded", a list with the item parameters and the estimate covariances (if available). If `mod` is `SingleGroupClass`, the list contains the item parameters as a matrix and the covariances as a list. If `mod` is `MultipleGroupClass`, the list contains the item parameters and covariances for the focal and reference groups only.

Examples

```r
library(mirt)
(mod1 <- mirt(Science, model = 1, itemtype = "graded", SE = TRUE))
(DFIT::ExtractGRMMirt(mod1))
```

ExtractMirtPars

Extracts the item parameters from a unidimensional mirt model

Description

Extracts the item parameters from a unidimensional mirt model

Usage

ExtractMirtPars(mod, focal = NULL, reference = NULL)
Arguments

mod A mirt object containing the fit of unidimensional model.
focal Character. Required if mod is MultipleGroupClass, focal should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.
reference Character. Required if mod is MultipleGroupClass, reference should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.

Value

If mod contains any itemtype == "gpcm", a list with the item parameters and the estimate covariances (if available). If mod is SingleGroupClass, the list contains the item parameters as a matrix and the covariances as a list. If mod is MultipleGroupClass, the list contains the item parameters and covariances for the focal and reference groups only.

Examples

library(mirt)
(mod1 <- mirt(Science, model = 1, itemtype = c("graded", "graded", "gpcm", "gpcm"), SE = TRUE))
(ExtractRaschMirtPars(mod1))

ExtractRaschMirt

Description

Extract item difficulties and item difficulty variance estimates for Rasch items from a fitted mirt object for one or two groups

Usage

ExtractRaschMirt(mod, focal = NULL, reference = NULL)

Arguments

mod A mirt object containing the fit of unidimensional model.
focal Character. Required if mod is MultipleGroupClass, focal should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.
reference Character. Required if mod is MultipleGroupClass, reference should coincide with the label for the focal group. If mod is SingleGroupClass, it is ignored.

Value

If mod contains any itemtype == "Rasch", a list with the item parameters and the estimate covariances (if available). If mod is SingleGroupClass, the list contains the item parameters as a matrix and the covariances as a list. If mod is MultipleGroupClass, the list contains the item parameters and covariances for the focal and reference groups only.
Examples

```r
library(mirt)
data <- expand.table(LSAT7)
(mod1 <- mirt(data, model = 1, itemtype = "Rasch", SE = TRUE))
(DFIT:::ExtractRaschMirt(mod1))
```

---

**Ipr**

*Item parameter replication*

---

**Description**

Generates a sample of item parameters assuming multivariate normality of estimates

**Usage**

```r
Ipr(itemParameters, itemCovariances, nReplicates = 5000)
```

**Arguments**

- `itemParameters`: A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should me a matrix with nrow equal to the number of items.
- `itemCovariances`: A list containing "focal" and "reference" matrices of covariance for item estimates. Each (focal and reference) may be either a list of covariance matrices for each item or a single matrix of covariance of all parameters.
- `nReplicates`: A numeric value indicating the number of replications to perform

**Value**

- `itemParameters`: A list with item parameters for focal and reference groups

**Author(s)**

Victor H. Cervantes <vhcervantesb at unal.edu.co>

**References**

Examples

```r
# Not run
#
# data(dichotomousItemParameters)
# threePlParameters <- dichotomousItemParameters
# isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
#               (dichotomousItemParameters[['reference']][, 3] == 0))
#
# threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
# threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
# threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
# threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
# threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
# threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
# threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]
#
# threePlAse <- list()
# threePlAse[['focal']] <- AseIrt(itemParameters = threePlParameters[['focal']],
#                               logistic = TRUE,
#                               sampleSize = 10000,
#
# threePlAse[['reference']] <- AseIrt(itemParameters = threePlParameters[['reference']],
#                               logistic = TRUE,
#                               sampleSize = 15000,
#
# set.seed(41568)
# threePlIpr <- Ipr(itemParameters = threePlParameters, itemCovariances = threePlAse,
#                   nReplicates = 100)
```

---

**IprMh**

*Mantel Haenszel for Item parameter replication*

**Description**

Calculates the Mantel-Haenszel theoretical parameter under IRT assumptions on a list of item parameters such as those produced by the Ipr function.

**Usage**

```r
IprMh(
  itemParameterList,
  irtModel = "2pl",
  focalDistribution = "norm",
  focalDistrExtra = list(mean = 0, sd = 1),
  referenceDistribution = "norm",
  referenceDistrExtra = list(mean = 0, sd = 1),
)```

IprMh

```r
  groupRatio = 1,
  subdivisions = 5000,
  logistic = TRUE
```

Arguments

- `itemParameterList`: A list where each element is a list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should be a matrix with nrow equal to the number of items.
- `irtModel`: A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".
- `focalDistribution`: A string stating the distribution assumed for the focal group.
- `focalDistrExtra`: A list stating the extra parameters needed by the focal distribution function.
- `referenceDistribution`: A string stating the distribution assumed for the reference group.
- `referenceDistrExtra`: A list stating the extra parameters needed by the reference distribution function.
- `groupRatio`: A positive value indicating how many members of the reference group are expected for each member of the focal group.
- `subdivisions`: A numeric value indicating the number of subdivisions for numerical integration.
- `logistic`: A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.

Value

- `mh`: A numeric matrix with the Mantel Haenszel values for all the item parameter in each set of `itemParameterList`

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Examples

```r
# Not run
#
# data(dichotomousItemParameters)
# threePlParameters <- dichotomousItemParameters
# isNot3Pl <- ((dichotomousItemParameters[['focal']][[, 3]] == 0) |
# (dichotomousItemParameters[['reference']][[, 3]] == 0))
#
# threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
# threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
# threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
# threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
# threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
# threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
# threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]
#
# threePlAse <- list()
# threePlAse[["focal"]]<- AseIrt(itemParameters = threePlParameters[["focal"]],
# logic = TRUE,
# sampleSize = 10000,
# irtModel = "3pl")
# threePlAse[["reference"]]<- AseIrt(itemParameters = threePlParameters[["reference"]],
# logic = TRUE,
# sampleSize = 10000,
# irtModel = "3pl")
#
# set.seed(41568)
# threePlIpr <- Ipr(itemParameters = threePlParameters, itemCovariances = threePlAse,
# nReplicates = 100)
# threePlMhIpr <- IprMh(itemParameterList = threePlIpr, irtModel = '3pl', logistic = TRUE)
```

IprNcdif

**NCDIF for Item parameter replication**

Description

Calculates the NCDIF index on a list of item parameters such as those produced by the Ipr function.

Usage

```r
IprNcdif(  
  itemParameterList,  
  irtModel = "2pl",  
  focalAbilities = NULL,  
  focalDistribution = "norm",  
  subdivisions = 5000,  
  ...)
```
logistic = TRUE,
focalDistrExtra = list(mean = 0, sd = 1)
)

Arguments

itemParameterList
   A list where each element is a list containing "focal" and "reference" item Parameters. Item parameters for each group should be a matrix with nrow equal to the number of items.

irtModel
   A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".

focalAbilities
   If NULL, NCDIF is calculated by numerical integration of focal distribution. If not NULL, must be a numerical vector containing the abilities for the individuals in the focal group.

focalDistribution
   A string stating the distribution name to be used for integrating. Only used if focalAbilities is NULL.

subdivisions
   A numeric value indicating the number of subdivisions for numerical integration. Only used if focalAbilities is NULL.

logistic
   A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.

focalDistrExtra
   A list stating the extra parameters needed by the focal distribution function.

Value

ncdif A numeric matrix with the NCDIF values for all the item parameter in each set of itemParameterList

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Examples

# # Not run
# # data(dichotomousItemParameters)
# # threePlParameters <- dichotomousItemParameters
# # isNot3Pl <- (dichotomousItemParameters[['focal']][, 3] == 0) |
# # (dichotomousItemParameters[['reference']][, 3] == 0)}
IprSam

Signed Area Measure for Item parameter replication

Description

Calculates Raju's Signed Area Measure index on a list of item parameters such as those produced by the Ipr function.

Usage

IprSam(
  itemParameterList,
  irtModel = "2pl",
  subdivisions = 5000,
  logistic = TRUE
)

Arguments

itemParameterList

A list where each element is a list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should be a matrix with nrow equal to the number of items.
irtModel A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".

subdivisions A numeric value indicating the number of subdivisions for numerical integration.

logistic A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.

Value

sam A numeric matrix with the Signed Area Measure values for all the item parameter in each set of itemParameterList

Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Examples

```r
# # Not run
# #
# # data(dichotomousItemParameters)
# # threePlParameters <- dichotomousItemParameters
# # isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
# # (dichotomousItemParameters[['reference']][, 3] == 0))
# #
# # threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl,]
# # threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl,]
# # threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
# # threePlParameters[['focal']][, c(12, 16, 28)] <- threePlParameters[['focal']][, c(12, 16, 28)]
# # threePlParameters[['reference']][, c(12, 16, 28)] <- threePlParameters[['reference']][, c(12, 16, 28)]
# #
# # threePlAse <- list()
# # threePlAse[['focal']] <- AseIrt(itemParameters = threePlParameters[['focal']],
# # logistic = TRUE,
# # sampleSize = 10000,
# # irtModel = "3pl"
# # threePlAse[['reference']] <- AseIrt(itemParameters = threePlParameters[['reference']],
# # logistic = TRUE,
```
IprUam

Unsigned Area Measure for Item parameter replication

Description

Calculates Raju’s Unsigned Area Measure index on a list of item parameters such as those produced by the Ipr function.

Usage

IprUam(
  itemParameterList, 
  irtModel = "2pl", 
  subdivisions = 5000, 
  logistic = TRUE
)

Arguments

itemParameterList A list where each element is a list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should be a matrix with nrow equal to the number of items.

irtModel A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".

subdivisions A numeric value indicating the number of subdivisions for numerical integration.

logistic A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.

Value

uam A numeric matrix with the Unsigned Area Measure values for all the item parameter in each set of itemParameterList.
Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Examples

```r
# Not run
#
# # data(dichotomousItemParameters)
# threePlParameters <- dichotomousItemParameters
# isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
# # (dichotomousItemParameters[['reference']][, 3] == 0))
# # threePlParameters[['focal']] <- threePlParameters[['focal']]![isNot3Pl, ]
# threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
# threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
# threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
# threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
# threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]
# threePlAse <- list()
# threePlAse[["focal"]]<- AseIrt(itemParameters = threePlParameters["focal"],
# logistic = TRUE,
# sampleSize = 10000,
# irtModel = "3pl")
# threePlAse[["reference"]]<- AseIrt(itemParameters = threePlParameters["reference"],
# logistic = TRUE,
# sampleSize = 10000,
# irtModel = "3pl")
# set.seed(41568)
# threePlIpr <- Ipr(itemParameters = threePlParameters, itemCovariances = threePlAse,
# nReplicates = 100)
# threePlUamIpr <- IprUam(itemParameterList = threePlIpr, irtModel = '3pl', logistic = TRUE)
```
IrtMh

Calculates the Mantel-Haenszel theoretical parameter when a dichotomous IRT model holds

Description

Calculates the Mantel-Haenszel theoretical parameter when a dichotomous IRT model holds

Usage

IrtMh(
  itemParameters,
  irtModel = "2pl",
  focalDistribution = "norm",
  referenceDistribution = "norm",
  focalDistrExtra = list(mean = 0, sd = 1),
  referenceDistrExtra = list(mean = 0, sd = 1),
  groupRatio = 1,
  logistic = TRUE,
  subdivisions = 5000
)

Arguments

itemParameters A list containing the "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale.
irtModel A string stating the irtModel used. May be one of "1pl", "2pl", or "3pl".
focalDistribution A string stating the distribution assumed for the focal group.
referenceDistribution A string stating the distribution assumed for the reference group.
focalDistrExtra A list of extra parameters for the focal distribution function.
referenceDistrExtra A list of extra parameters for the reference distribution function.
groupRatio A positive value indicating how many members of the reference group are expected for each member of the focal group.
logistic A logical indicating whether the logistic or the normal metric should be used.
subdivisions A numeric value stating the maximum number of subdivisions for adaptive quadrature.

Value

mh A list containing the asymptotic matrices for each item
Author(s)

Victor H. Cervantes <vhcervantesb at unal.edu.co>

References


Examples

data(dichotomousItemParameters)
threePlParameters <- dichotomousItemParameters
isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
(dichotomousItemParameters[['reference']][, 3] == 0))

threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]

threePlMh <- IrtMh(itemParameters = threePlParameters, irtModel = "3pl",
focalDistrExtra = list(mean = 0, sd = 1),
referenceDistrExtra = list(mean = 0, sd = 1), groupRatio = 1,
logistic = FALSE)

Ncdif

Calculates NCDIF index for an item with given item parameters of focal and reference groups.

Description

Calculates NCDIF index for an item with given item parameters of focal and reference groups.

Usage

Ncdif(
  itemParameters,
  irtModel = "2pl",
  focalAbilities = NULL,
  focalDistribution = "norm",
  ...)
ncdif = 5000,
logistic = TRUE,
focalDistrExtra = list(mean = 0, sd = 1)
)

**Arguments**

- **itemParameters** A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should me a matrix with nrow equal to the number of items.
- **irtModel** A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".
- **focalAbilities** If NULL, NCDIF is calculated by numerical integration of focal distribution. If not NULL, it must be a numerical vector containing the abilities for the individuals in the focal group.
- **focalDistribution** A string stating the distribution name to be used for integrating. Only used if focalAbilities is NULL.
- **subdivisions** A numeric value indicating the number of subdivisions for numerical integration. Only used if focalAbilities is NULL.
- **logistic** A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.
- **focalDistrExtra** Extra parameters for the focal group distribution function if needed.

**Value**

- **ncdif** Numeric vector with the NCDIF index value for each item.

**Author(s)**

Victor H. Cervantes <vhcervantesb at unal.edu.co>

**References**


**Examples**

data(dichotomousItemParameters)

threePlParameters <- dichotomousItemParameters

isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) | (dichotomousItemParameters[['reference']][, 3] == 0))

threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
threePlParameters[['reference']] <- threePlParameters[['reference']][[isNot3Pl], ]
threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
threePlParameters[['reference']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
threePlParameters[['reference']] <- threePlParameters[['focal']][-c(12, 16, 28), ]

threePlNcdif <- Ncdif(itemParameters = threePlParameters, irtModel = '3pl',
                       focalAbilities = NULL, focalDistribution = 'norm',
                       subdivisions = 5000, logistic = TRUE)

Description

Plot the item characteristic (expected score) curve for focal and reference groups for the iiItem along with a representation of the focal group density.

Usage

PlotNcdif(
  iiItem,
  itemParameters,
  irtModel = "2pl",
  logistic = TRUE, plotDensity = FALSE,
  focalAbilities = NULL, focalDistribution = "norm",
  focalDistrExtra = list(mean = 0, sd = 1),
  from = -5,
  to = 5,
  thetaInt = 0.01, colour = TRUE,
  highColour = "blue",
  main = "", xlab = "Ability",
  ylab = "Probability",
  iccText = "Group ICCs",
  focalIccText = "Focal group ICC",
  referenceIccText = "Reference group ICC",
  focalDensityText = "Focal group density" )
Arguments

**item**  
Item (row) number for the item in each of the itemParameter matrices to plot.

**itemParameters**  
A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should me a matrix with nrow equal to the number of items.

**irtModel**  
A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".

**logistic**  
A logical value stating if the IRT model will use the logistic or the normal metric.

**plotDensity**  
logical indicating if the focal distribution density should be plotted as a density curve (TRUE) or if it should be represented as an area gradient (FALSE). Defaults to gradient.

**focalAbilities**  
If NULL, density is calculated theoretically from focal distribution. If not NULL, it must be a numerical vector containing the abilities for the individuals in the focal group.

**focalDistribution**  
A string stating the distribution name to be used for density calculation. Only used if focalAbilities is NULL.

**focalDistrExtra**  
Extra parameters for the focal group distribution function if needed.

**from**  
value on the x-axis to serve as minimum for the plot

**to**  
value on the x-axis to serve as maximum for the plot

**thetaInt**  
value for the x-axis step for probabilities and density evaluation. Only used if focalAbilities is NULL.

**colour**  
logical value indicating if the area gradient should be presented in colour when plotDensity is FALSE, or if the different lines should be presented in colour when plotDensity is TRUE.

**highColour**  
character indicating the colour text name that should be used for high density regions.

**main**  
text for plot main title.

**xlab**  
text for x-axis label.

**ylab**  
text for y-axis label.

**iccText**  
text for legend title related to ICC curves.

**focalIccText**  
legend for focal group ICC curve.

**referenceIccText**  
legend for reference group ICC curve.

**focalDensityText**  
legend for focal group density curve when plotDensity is TRUE. Text for legend title related to the colour gradient when plotDensity is FALSE.

Value

plotNCDIF A ggplot object for the plot
polytomousItemParameters

Author(s)
Victor H. Cervantes <vhcervantesb@unal.edu.co>

Examples

data(dichotomousItemParameters)
threePlParameters <- dichotomousItemParameters
isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
              (dichotomousItemParameters[['reference']][, 3] == 0))
threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]

# # Non Uniform - != guess DIF item
PlotNcdif(iiItem = 22, itemParameters = threePlParameters, irtModel = "3pl",
          plotDensity = FALSE, main = "Item 22 Non uniform and different guessing DIF. 3PL")

# # Uniform - != guess DIF item
PlotNcdif(iiItem = 15, itemParameters = threePlParameters, irtModel = "3pl",
          plotDensity = FALSE, main = "Item 15 Uniform and different guessing DIF. 3PL")

Description
This data set contains the item parameters found in Raju, N., Fortmann-Johnson, K., Kim, W.,
differential functioning in the polytomous DFIT framework. Applied psychological measurement,
33(2), 133–147.

Usage

data(polytomousItemParameters)

Format
a list with ‘focal’ and ‘reference’ elements. Each is a matrix 1 per item by 5 columns: item
discrimination, four item step parameters.
Source
This data set contains the item parameters based on those found in Raju et al. (2009).

References

ProductProbabilities

Calculates the product of item response probabilities for dichotomous IRT models

Description
Calculates the product of item response probabilities for dichotomous IRT models

Usage
ProductProbabilities(thetaValue, itemParameters, logistic, irtModel = "3pl")

Arguments

thetaValue A numeric value or array for the theta (ability) value(s) for which the product will be calculated
itemParameters A matrix containing the item parameters.
logistic A logical indicating whether the logistic or the normal metric should be used.
irtModel A string stating the irtModel used. May be one of "1pl", "2pl", or "3pl".

Value
pq A numeric matrix containing the crossed product on each thetaValue for each item.

Author(s)
Victor H. Cervantes <vhcervantesb at unal.edu.co>
**SignedArea**

*Calculates Raju’s Signed Area Measure index for an item with given item parameters of focal and reference groups.*

**Description**

Calculates Raju’s Signed Area Measure index for an item with given item parameters of focal and reference groups.

**Usage**

```r
SignedArea(
  itemParameters,
  irtModel = "2pl",
  subdivisions = 5000,
  logistic = TRUE
)
```

**Arguments**

- **itemParameters**: A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should me a matrix with nrow equal to the number of items.
- **irtModel**: A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".
- **subdivisions**: A numeric value indicating the number of subdivisions for numerical integration.
- **logistic**: A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.

**Value**

- **sam**: A numeric matrix with the Signed Area Measure values for all the item parameter in each set of itemParameterList

**Author(s)**

Victor H. Cervantes <vhcervantesb at unal.edu.co>

**References**


Examples

data(dichotomousItemParameters)

threePlParameters <- dichotomousItemParameters

isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) |
             (dichotomousItemParameters[['reference']][, 3] == 0))

threePlParameters[['focal']]    <- threePlParameters[['focal']][!isNot3Pl,]
threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl,]
threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
threePlParameters[['focal']]    <- threePlParameters[['focal']][-c(12, 16, 28),]
threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28),]

sam3pl <- SignedArea(itemParameters = threePlParameters, irtModel = "3pl",
                             subdivisions = 5000, logistic = TRUE)

Usage

UnsignedArea(itemParameters, irtModel = "2pl",
             subdivisions = 5000, logistic = TRUE)

Arguments

itemParameters A list containing "focal" and "reference" item parameters. Item parameters are assumed to be on the same scale. Item parameters for each group should me a matrix with nrow equal to the number of items.

irtModel A string stating the irtModel to be used. Should be one of "1pl", "2pl", "3pl", "grm" or "pcm".

subdivisions A numeric value indicating the number of subdivisions for numerical integration.
**logistic**

A logical value stating if the IRT model will use the logistic or the normal metric. Defaults to using the logistic metric by fixing the D constant to 1. If FALSE the constant is set to 1.702 so that the normal metric is used.

**Value**

uam A numeric matrix with the Unsigned Area Measure values for all the item parameter in each set of itemParameterList

**Author(s)**

Victor H. Cervantes <vhcervantesb at unal.edu.co>

**References**


**Examples**

data(dichotomousItemParameters)

threePlParameters <- dichotomousItemParameters
isNot3Pl <- ((dichotomousItemParameters[['focal']][, 3] == 0) | (dichotomousItemParameters[['reference']][, 3] == 0))

threePlParameters[['focal']] <- threePlParameters[['focal']][!isNot3Pl, ]
threePlParameters[['reference']] <- threePlParameters[['reference']][!isNot3Pl, ]
threePlParameters[['focal']][, 3] <- threePlParameters[['focal']][, 3] + 0.1
threePlParameters[['reference']][, 3] <- threePlParameters[['reference']][, 3] + 0.1
threePlParameters[['focal']][, 2] <- threePlParameters[['focal']][, 2] + 1.5
threePlParameters[['focal']] <- threePlParameters[['focal']][-c(12, 16, 28), ]
threePlParameters[['reference']] <- threePlParameters[['reference']][-c(12, 16, 28), ]

uam3pl <- UnsignedArea(itemParameters = threePlParameters, irtModel = "3pl",
                        subdivisions = 5000, logistic = TRUE)
# Index

<table>
<thead>
<tr>
<th>Ase1pl</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ase2pl</td>
<td>3</td>
</tr>
<tr>
<td>Ase3pl</td>
<td>4</td>
</tr>
<tr>
<td>AseIrt</td>
<td>5</td>
</tr>
<tr>
<td>Bound3PlIpr</td>
<td>7</td>
</tr>
<tr>
<td>Bound4PlIpr</td>
<td>8</td>
</tr>
<tr>
<td>Calculate1plProb</td>
<td>10</td>
</tr>
<tr>
<td>Calculate2plProb</td>
<td>10</td>
</tr>
<tr>
<td>Calculate3plProb</td>
<td>11</td>
</tr>
<tr>
<td>Calculate4plProb</td>
<td>12</td>
</tr>
<tr>
<td>CalculateGrmExp</td>
<td>13</td>
</tr>
<tr>
<td>CalculateItemDifferences</td>
<td>13</td>
</tr>
<tr>
<td>CalculatePcmExp</td>
<td>14</td>
</tr>
<tr>
<td>Cdif</td>
<td>15</td>
</tr>
<tr>
<td>CheckDiscriminations</td>
<td>17</td>
</tr>
<tr>
<td>CheckGuessings</td>
<td>17</td>
</tr>
<tr>
<td>CheckUpper</td>
<td>18</td>
</tr>
<tr>
<td>CrossedProbabilities</td>
<td>18</td>
</tr>
<tr>
<td>CutoffIpr</td>
<td>19</td>
</tr>
<tr>
<td>DeltaMhIrt</td>
<td>22</td>
</tr>
<tr>
<td>DFIT</td>
<td>24</td>
</tr>
<tr>
<td>dichotomousItemParameters</td>
<td>25</td>
</tr>
<tr>
<td>Dtf</td>
<td>26</td>
</tr>
<tr>
<td>Extract2PLMirt</td>
<td>27</td>
</tr>
<tr>
<td>Extract3PLMirt</td>
<td>28</td>
</tr>
<tr>
<td>Extract4PLMirt</td>
<td>29</td>
</tr>
<tr>
<td>ExtractGPCMMirt</td>
<td>30</td>
</tr>
<tr>
<td>ExtractGRMMirt</td>
<td>31</td>
</tr>
<tr>
<td>ExtractMirtPars</td>
<td>31</td>
</tr>
<tr>
<td>ExtractRaschMirt</td>
<td>32</td>
</tr>
<tr>
<td>Ipr</td>
<td>33</td>
</tr>
<tr>
<td>IprMh</td>
<td>34</td>
</tr>
<tr>
<td>IprNcdif</td>
<td>36</td>
</tr>
<tr>
<td>IprSam</td>
<td>38</td>
</tr>
<tr>
<td>IprUam</td>
<td>40</td>
</tr>
<tr>
<td>IrtMh</td>
<td>42</td>
</tr>
<tr>
<td>Ncdif</td>
<td>43</td>
</tr>
<tr>
<td>PlotNcdif</td>
<td>45</td>
</tr>
<tr>
<td>polytomousItemParameters</td>
<td>47</td>
</tr>
<tr>
<td>ProductProbabilities</td>
<td>48</td>
</tr>
<tr>
<td>SignedArea</td>
<td>49</td>
</tr>
<tr>
<td>UnsignedArea</td>
<td>50</td>
</tr>
</tbody>
</table>