

Package ‘difNLR’

August 20, 2019

Type Package

Title DIF and DDF Detection by Non-Linear Regression Models

Version 1.3.0

Date 2019-08-19

Author Adela Hladka [aut, cre],
Patricia Martinkova [aut],
Karel Zvara [ctb]

Maintainer Adela Hladka <hladka@cs.cas.cz>

Depends R (>= 3.1)

Imports CTT, ggplot2 (>= 2.2.1), grDevices, methods, msm, nnet, plyr,
reshape2, stats, VGAM

Description Detection of differential item functioning (DIF) among dichotomously scored items and differential distractor functioning (DDF) among un-scored items with non-linear regression procedures based on generalized logistic regression models (Drabínová and Martinková, 2017, doi:10.1111/jedm.12158).

License GPL-3

LazyData yes

RoxygenNote 6.1.1

BugReports <https://github.com/adelahladka/difNLR/issues>

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2019-08-20 12:10:10 UTC

R topics documented:

difNLR-package	2
checkInterval	4
ddfMLR	5
ddfORD	9

difNLR	14
estimNLR	20
formulaNLR	22
genNLR	24
GMAT	27
GMAT2	28
GMAT2key	29
GMAT2test	30
GMATkey	31
GMATtest	32
MLR	33
MSATB	36
MSATBkey	37
MSATBtest	38
NLR	39
ORD	43
plot.difNLR	45
predict.difNLR	47
startNLR	49
Index	52

 difNLR-package

DIF and DDF Detection by Non-Linear Regression Models.

Description

The difNLR package contains method for detection of differential item functioning (DIF) based on non-linear regression. Both uniform and non-uniform DIF effects can be detected when considering one focal group. The method also allows to test the difference in guessing or inattention parameters between reference and focal group. DIF detection method is based either on likelihood-ratio test, or on F-test of submodel. Package also offers method for detection of differential distractor functioning (DDF) based on multinomial log-linear regression model and newly for ordinal data via adjacent and cumulative logistic regression models.

Details

Package: difNLR

Type: Package

Version: 1.3.0

Date: 2019-08-19

Depends: R (>= 3.1) Imports: CTT, ggplot2 (>= 2.2.1), grDevices, methods, msm, nnet, plyr, reshape2, stats, VGAM

License: GPL-3

BugReports: <https://github.com/adelahladka/difNLR/issues> Encoding: UTF-8

Note

This package was supported by grant funded by Czech Science foundation under number GJ15-15856Y.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

- Agresti, A. (2010). Analysis of ordinal categorical data. Second edition. John Wiley & Sons.
- Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.
- Kingston, N., Leary, L., & Wightman, L. (1985). An Exploratory Study of the Applicability of Item Response Theory Methods to the Graduate Management Admission Test. *ETS Research Report Series*, 1985(2) : 1-64.
- Martinkova, P., Drabinova, A., Liaw, Y. L., Sanders, E. A., McFarland, J. L., & Price, R. M. (2017). Checking equity: Why Differential Item Functioning Analysis Should Be a Routine Part of Developing Conceptual Assessments. *CBE-Life Sciences Education*, 16(2), <https://doi.org/10.1187/cbe.16-10-0307>.
- Swaminathan, H. & Rogers, H. J. (1990). Detecting Differential Item Functioning Using Logistic Regression Procedures. *Journal of Educational Measurement*, 27, 361-370.
- Vlckova, K. (2014). Test and Item Fairness (Unpublished master's thesis).

See Also

Useful links:

- Report bugs at <https://github.com/adelahladka/difNLR/issues>

checkInterval	<i>Checks interval bounds.</i>
---------------	--------------------------------

Description

Checks whether x lies in interval defined by bounds in `vec`. If it does, it returns value of x . In case that value of x is lower than lower bound specified in `vec`, it returns its value. In case that value of x is greater than upper bound specified in `vec`, it returns its value.

Usage

```
checkInterval(x, vec)
```

Arguments

<code>x</code>	numeric.
<code>vec</code>	numeric: increasingly sorted bounds of interval.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

Examples

```
## Not run:  
checkInterval(x = 0.5, vec = c(0, 1))  
checkInterval(x = 1.5, vec = c(0, 1))  
checkInterval(x = -0.5, vec = c(0, 1))  
  
## End(Not run)
```

 ddfMLR *DDF detection for nominal data.*

Description

Performs DDF detection procedure for nominal data based on multinomial log-linear regression model and likelihood ratio test of submodel.

Usage

```
ddfMLR(Data, group, focal.name, key, type = "both", match = "zscore", anchor = NULL,
purify = FALSE, nrIter = 10, p.adjust.method = "none", parametrization = "irt",
alpha = 0.05)
```

```
## S3 method for class 'ddfMLR'
print(x, ...)
```

```
## S3 method for class 'ddfMLR'
plot(x, item = "all", title, group.names, ...)
```

```
## S3 method for class 'ddfMLR'
coef(object, SE = FALSE, simplify = FALSE, ...)
```

```
## S3 method for class 'ddfMLR'
logLik(object, item = "all", ...)
```

```
## S3 method for class 'ddfMLR'
AIC(object, item = "all", ...)
```

```
## S3 method for class 'ddfMLR'
BIC(object, item = "all", ...)
```

Arguments

Data	character: either the unscored data matrix only, or the unscored data matrix plus the vector of group. See Details .
group	numeric or character: either the binary vector of group membership or the column indicator of group membership. See Details .
focal.name	numeric or character: indicates the level of group which corresponds to focal group
key	character: the answer key. See Details .
type	character: type of DDF to be tested (either "both" (default), "udif", or "nudif"). See Details .
match	specifies matching criterion. Can be either "zscore" (default, standardized total score), "score" (total test score), or vector of the same length as number of observations in Data. See Details .

anchor	Either NULL (default) or a vector of item names or item identifiers specifying which items are currently considered as anchor (DDF free) items. Argument is ignored if match is not "zscore" or "score".
purify	logical: should the item purification be applied? (default is FALSE). See Details .
nrIter	numeric: the maximal number of iterations in the item purification (default is 10).
p.adjust.method	character: method for multiple comparison correction. See Details .
parametrization	character: parametrization of regression coefficients. Possible options are "irt" (default) and "classic". See Details .
alpha	numeric: significance level (default is 0.05).
x	an object of 'ddfMLR' class
...	other generic parameters for print or plot functions.
item	numeric or character: either the vector of column indicator (number or column name) or 'all' (default) for all items. See Details .
title	string: title of plot.
group.names	character: names of reference and focal group.
object	an object of 'ddfMLR' class
SE	logical: should be standard errors also returned? (default is FALSE).
simplify	logical: should the result be simplified to a matrix? (default is FALSE).

Details

Performs DDF detection procedure for nominal data based on multinomial log-linear regression model and likelihood ratio test of submodel.

The `Data` is a matrix which rows represents examinee unscored answers and columns correspond to the items. The `group` must be either a vector of the same length as `nrow(Data)` or column indicator of `Data`. The `key` must be a vector of correct answers corresponding to columns of `Data`.

The `type` corresponds to type of DDF to be tested. Possible values are "both" to detect any DDF (uniform and/or non-uniform), "udif" to detect only uniform DDF or "nudif" to detect only non-uniform DDF.

Argument `match` represents the matching criterion. It can be either the standardized test score (default, "zscore"), total test score ("score"), or any other continuous or discrete variable of the same length as number of observations in `Data`.

A set of anchor items (DDF free) can be specified through the `anchor` argument. It need to be a vector of either item names (as specified in column names of `Data`) or item identifiers (integers specifying the column number). In case anchor items are provided, only these items are used to compute matching criterion match. If the `match` argument is not either "zscore" or "score", `anchor` argument is ignored. When anchor items are provided, purification is not applied.

The `p.adjust.method` is a character for `p.adjust` function from the `stats` package. Possible values are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none". See also [p.adjust](#) for more information.

Argument `parametrization` is a character which specifies parametrization of regression parameters. Default option is "irt" which returns IRT parametrization (difficulty-discrimination). Option "classic" returns intercept-slope parametrization with effect of group membership and interaction with matching criterion.

The output of the `ddfMLR` function is displayed by the `print.ddfMLR` function.

The characteristic curve for item specified in `item` option can be plotted. `item` can be column indicator (numeric or character - column name) or "all" (default). For option "all", characteristic curves of all converged items are plotted. The drawn curves represent best model.

Missing values are allowed but discarded for item estimation. They must be coded as NA for both, Data and group parameters.

Value

A list of class 'ddfMLR' with the following arguments:

`Sval` the values of likelihood ratio test statistics.

`m1rPAR` the estimates of final model.

`m1rSE` standard errors of the estimates of final model.

`parM0` the estimates of null model.

`parM1` the estimates of alternative model.

`alpha` numeric: significance level.

`DDFitems` either the column indicators of the items which were detected as DDF, or "No DDF item detected".

`type` character: type of DDF that was tested.

`purification` purify value.

`nrPur` number of iterations in item purification process. Returned only if `purify` is TRUE.

`ddfPur` a binary matrix with one row per iteration of item purification and one column per item. "1" in *i*-th row and *j*-th column means that *j*-th item was identified as DDF in *i*-1-th iteration. Returned only if `purify` is TRUE.

`conv.puri` logical indicating whether item purification process converged before the maximal number `nrIter` of iterations. Returned only if `purify` is TRUE.

`p.adjust.method` character: method for multiple comparison correction which was applied.

`pval` the p-values by likelihood ratio test.

`adj.pval` the adjusted p-values by likelihood ratio test using `p.adjust.method`.

`df` the degrees of freedom of likelihood ratio test.

`group` the vector of group membership.

`Data` the data matrix.

`match` matching criterion.

`group.names` levels of grouping variable.

`l1M0` log-likelihood of null model.

`l1M1` log-likelihood of alternative model.

AICM0 AIC of null model.
 AICM1 AIC of alternative model.
 BICM0 BIC of null model.
 BICM1 BIC of alternative model.

Author(s)

Adela Hladka (nee Drabinova)
 Institute of Computer Science, The Czech Academy of Sciences
 Faculty of Mathematics and Physics, Charles University
 hladka@cs.cas.cz

Patricia Martinkova
 Institute of Computer Science, The Czech Academy of Sciences

See Also

[p.adjust.multinom](#)

Examples

```
## Not run:
# loading data based on GMAT
data(GMATtest, GMATkey)

Data <- GMATtest[, 1:20]
group <- GMATtest[, "group"]
key <- GMATkey

# Testing both DDF effects
(x <- ddfMLR(Data, group, focal.name = 1, key))

# Testing both DDF effects with Benjamini-Hochberg adjustment method
ddfMLR(Data, group, focal.name = 1, key, p.adjust.method = "BH")

# Testing both DDF effects with item purification
ddfMLR(Data, group, focal.name = 1, key, purify = T)

# Testing uniform DDF effects
ddfMLR(Data, group, focal.name = 1, key, type = "udif")
# Testing non-uniform DDF effects
ddfMLR(Data, group, focal.name = 1, key, type = "nudif")

# Testing both DDF effects with total score as matching criterion
ddfMLR(Data, group, focal.name = 1, key, match = "score")

# Testing both DDF effects using classic parametrization
ddfMLR(Data, group, focal.name = 1, key, parametrization = "classic")
```



```
# Graphical devices
plot(x, item = 1)
plot(x, item = 1, group.names = c("Group 1", "Group 2"))
plot(x, item = x$DDFitems)
plot(x, item = "all")

# AIC, BIC, log-likelihood
AIC(x); BIC(x); logLik(x)
# AIC, BIC, log-likelihood for the first item
AIC(x, item = 1); BIC(x, item = 1); logLik(x, item = 1)

# estimates
coef(x)
coef(x, SE = TRUE)
coef(x, SE = TRUE, simplify = TRUE)

## End(Not run)
```

ddfORD

DDF detection for ordinal data.

Description

Performs DDF detection procedure for ordinal data based either on adjacent category logit model or on cumulative logit model and likelihood ratio test of submodel.

Usage

```
ddfORD(Data, group, focal.name, model = "adjacent", type = "both", match = "zscore",
anchor = NULL, purify = FALSE, nrIter = 10, p.adjust.method = "none",
parametrization = "irt", alpha = 0.05)
```

```
## S3 method for class 'ddfORD'
print(x, ...)
```

```
## S3 method for class 'ddfORD'
coef(object, SE = FALSE, simplify = FALSE, ...)
```

```
## S3 method for class 'ddfORD'
logLik(object, item = "all", ...)
```

```
## S3 method for class 'ddfORD'
AIC(object, item = "all", ...)
```

```
## S3 method for class 'ddfORD'
BIC(object, item = "all", ...)
```

```
## S3 method for class 'ddfORD'
plot(x, item = "all", title, plot.type, group.names,
     ...)
```

Arguments

Data	matrix or data.frame: the ordinarily scored data matrix only or the ordinarily scored data matrix plus the vector of group. See Details .
group	numeric or character: either the binary vector of group membership or the column indicator of group membership. See Details .
focal.name	numeric or character: indicates the level of group which corresponds to focal group
model	character: logistic regression model for ordinal data (either "adjacent" (default) or "cumulative"). See Details .
type	character: type of DDF to be tested (either "both" (default), "udif", or "nudif"). See Details .
match	specifies matching criterion. Can be either "zscore" (default, standardized total score), "score" (total test score), or vector of the same length as number of observations in Data. See Details .
anchor	Either NULL (default) or a vector of item names or item identifiers specifying which items are currently considered as anchor (DDF free) items. Argument is ignored if match is not "zscore" or "score".
purify	logical: should the item purification be applied? (default is FALSE). See Details .
nrIter	numeric: the maximal number of iterations in the item purification (default is 10).
p.adjust.method	character: method for multiple comparison correction. See Details .
parametrization	character: parametrization of regression coefficients. Possible options are "irt" (default) and "classic". See Details .
alpha	numeric: significance level (default is 0.05).
x	an object of 'ddfORD' class
...	other generic parameters for print function.
object	an object of 'ddfORD' class
SE	logical: should be standard errors also returned? (default is FALSE).
simplify	logical: should the result be simplified to a matrix? (default is FALSE).
item	numeric or character: either the vector of column indicator (number or column name) or 'all' (default) for all items.
title	string: title of plot.
plot.type	character: which plot should be displayed for cumulative logistic regression model. Either "category" for category probabilities or "cumulative" for cumulative probabilities.
group.names	character: names of reference and focal group.

Details

DDF detection procedure for ordinal data based either on adjacent category logit model or on cumulative logit model.

The `Data` is a matrix or `data.frame` which rows represents examinee ordinarily scored answers and columns correspond to the items. The `group` must be either a vector of the same length as `nrow(Data)` or column indicator of `Data`.

The `model` corresponds to model to be used for DDF detection. Options are "adjacent" for adjacent category logit model or "cumulative" for cumulative logit model.

The `type` corresponds to type of DDF to be tested. Possible values are "both" to detect any DDF (uniform and/or non-uniform), "uniform" to detect only uniform DDF or "nonuniform" to detect only non-uniform DDF.

Argument `match` represents the matching criterion. It can be either the standardized test score (default, "zscore"), total test score ("score"), or any other continuous or discrete variable of the same length as number of observations in `Data`.

A set of anchor items (DDF free) can be specified through the `anchor` argument. It need to be a vector of either item names (as specified in column names of `Data`) or item identifiers (integers specifying the column number). In case anchor items are provided, only these items are used to compute matching criterion match. If the `match` argument is not either "zscore" or "score", `anchor` argument is ignored. When anchor items are provided, purification is not applied.

The `p.adjust.method` is a character for `p.adjust` function from the `stats` package. Possible values are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", and "none". See also [p.adjust](#) for more information.

Argument `parametrization` is a character which specifies parametrization of regression parameters. Default option is "irt" which returns IRT parametrization (difficulty-discrimination). Option "classic" returns intercept-slope parametrization with effect of group membership and interaction with matching criterion.

The output of the `ddfORD()` function is displayed by the `print.ddfORD` function.

The characteristic curve for item specified in `item` option can be plotted. For default option "all" of item, characteristic curves of all converged items are plotted. The drawn curves represent best model.

Missing values are allowed but discarded for item estimation. They must be coded as NA for both, `Data` and group parameters.

Value

A list of class 'ddfORD' with the following arguments:

`Sval` the values of likelihood ratio test statistics.

`ordPAR` the estimates of the final model.

`ordSE` standard errors of the estimates of the final model.

`parM0` the estimates of null model.

`parM1` the estimates of alternative model.

`model` model used for DDF detection.

`alpha` numeric: significance level.

DDFitems either the column indicators of the items which were detected as DDF, or "No DDF item detected".

type character: type of DDF that was tested.

purification purify value.

nrPur number of iterations in item purification process. Returned only if purify is TRUE.

ddfPur a binary matrix with one row per iteration of item purification and one column per item. "1" in i-th row and j-th column means that j-th item was identified as DDF in i-1-th iteration. Returned only if purify is TRUE.

conv.puri logical indicating whether item purification process converged before the maximal number nrIter of iterations. Returned only if purify is TRUE.

p.adjust.method character: method for multiple comparison correction which was applied.

pval the p-values by likelihood ratio test.

adj.pval the adjusted p-values by likelihood ratio test using p.adjust.method.

df the degrees of freedom of likelihood ratio test.

group the vector of group membership.

Data the data matrix.

match matching criterion.

group.names levels of grouping variable.

l1M0 log-likelihood of null model.

l1M1 log-likelihood of alternative model.

AICM0 AIC of null model.

AICM1 AIC of alternative model.

BICM0 BIC of null model.

BICM1 BIC of alternative model.

parametrization Parameters' parametrization.

Author(s)

Adela Hladka (nee Drabinova)
 Institute of Computer Science, The Czech Academy of Sciences
 Faculty of Mathematics and Physics, Charles University
 hladka@cs.cas.cz

Patricia Martinkova
 Institute of Computer Science, The Czech Academy of Sciences
 martinkova@cs.cas.cz

References

Agresti, A. (2010). Analysis of ordinal categorical data. Second edition. John Wiley & Sons.

See Also

[p.adjust vglm](#)

Examples

```
## Not run:
# loading data
data(dataMedicalgraded, package = "ShinyItemAnalysis")
Data <- dataMedicalgraded[, 1:5]
group <- dataMedicalgraded[, 101]

# Testing both DDF effects with adjacent category logit model
(x <- ddfORD(Data, group, focal.name = 1, model = "adjacent"))

# graphical devices
plot(x, item = 3)
plot(x, item = "X2003")
plot(x, item = "X2003", group.names = c("Group 1", "Group 2"))

# estimated parameters
coef(x)
coef(x, SE = T) # with SE
coef(x, SE = T, simplify = T) # with SE, simplified

# AIC, BIC, log-likelihood
AIC(x); BIC(x); logLik(x)
# AIC, BIC, log-likelihood for the first item
AIC(x, item = 1); BIC(x, item = 1); logLik(x, item = 1)

# Testing both DDF effects with Benjamini-Hochberg adjustment method
ddfORD(Data, group, focal.name = 1, model = "adjacent", p.adjust.method = "BH")

# Testing both DDF effects with item purification
ddfORD(Data, group, focal.name = 1, model = "adjacent", purify = T)

# Testing uniform DDF effects
ddfORD(Data, group, focal.name = 1, model = "adjacent", type = "udif")
# Testing non-uniform DDF effects
ddfORD(Data, group, focal.name = 1, model = "adjacent", type = "nudif")

# Testing both DDF effects with total score as matching criterion
ddfORD(Data, group, focal.name = 1, model = "adjacent", match = "score")

# Testing both DDF effects with cumulative logit model
# using IRT parametrization
(x <- ddfORD(Data, group, focal.name = 1, model = "cumulative", parametrization = "irt"))

# graphical devices
plot(x, item = 3, plot.type = "cumulative")
plot(x, item = 3, plot.type = "category")

# estimated parameters in IRT parametrization
```

```
coef(x, simplify = T)

## End(Not run)
```

difNLR

DIF detection using non-linear regression method.

Description

Performs DIF detection procedure for dichotomous data based on non-linear regression model (generalized logistic regression) and either likelihood-ratio or F test of submodel.

Usage

```
difNLR(Data, group, focal.name, model, constraints, type = "both", method = "nls",
match = "zscore", anchor = NULL, purify = FALSE, nrIter = 10, test = "LR", alpha = 0.05,
p.adjust.method = "none", start, initboot = T, nrBo = 20)
```

```
## S3 method for class 'difNLR'
print(x, ...)
```

```
## S3 method for class 'difNLR'
fitted(object, item = "all", ...)
```

```
## S3 method for class 'difNLR'
coef(object, SE = FALSE, simplify = FALSE, ...)
```

```
## S3 method for class 'difNLR'
logLik(object, item = "all", ...)
```

```
## S3 method for class 'difNLR'
AIC(object, item = "all", ...)
```

```
## S3 method for class 'difNLR'
BIC(object, item = "all", ...)
```

```
## S3 method for class 'difNLR'
residuals(object, item = "all", ...)
```

Arguments

Data	numeric: either the scored data matrix only, or the scored data matrix plus the vector of group. See Details .
group	numeric or character: either the binary vector of group membership or the column indicator (in Data) of group membership. See Details .

<code>focal.name</code>	numeric or character: indicates the level of group which corresponds to focal group
<code>model</code>	character: generalized logistic regression model to be fitted. See Details .
<code>constraints</code>	character: which parameters should be the same for both groups. See Details .
<code>type</code>	character: type of DIF to be tested. Possible values are "both" (default), "udif", "nudif", "all", or combination of parameters "a", "b", "c" and "d". See Details .
<code>method</code>	character: method used to estimate parameters. The options are "nls" for non-linear least squares (default) and "likelihood" for maximum likelihood method.
<code>match</code>	character or numeric: specifies matching criterion. Can be either "zscore" (default, standardized total score), "score" (total test score), or numeric vector of the same length as number of observations in Data. See Details .
<code>anchor</code>	Either NULL (default) or a vector of item names or item identifiers specifying which items are currently considered as anchor (DIF free) items. Argument is ignored if match is not "zscore" or "score". See Details .
<code>purify</code>	logical: should the item purification be applied? (default is FALSE). See Details .
<code>nrIter</code>	numeric: the maximal number of iterations in the item purification (default is 10).
<code>test</code>	character: test to be performed for DIF detection. Can be either "LR" (default), or "F". See Details .
<code>alpha</code>	numeric: significance level (default is 0.05).
<code>p.adjust.method</code>	character: method for multiple comparison correction. See Details .
<code>start</code>	numeric: list with as many elements as number of items. Each element is a named numeric vector with values representing initial values for parameter estimation. See Details .
<code>initboot</code>	logical: in case of convergence issues, should be starting values recalculated based on bootstrapped samples? (default is TRUE). See Details .
<code>nrBo</code>	numeric: the maximal number of iterations for calculation of starting values using bootstrapped samples (default is 20).
<code>x</code>	an object of "difNLR" class
<code>...</code>	other generic parameters for S3 methods.
<code>object</code>	an object of 'difNLR' class
<code>item</code>	either character ("all"), or numeric vector, or single number corresponding to column indicators.
<code>SE</code>	logical: should be standard errors also returned? (default is FALSE).
<code>simplify</code>	logical: should the result be simplified to a matrix? (default is FALSE).

Details

DIF detection procedure based on non-linear regression is the extension of logistic regression procedure (Swaminathan and Rogers, 1990).

The `Data` is a matrix which rows represents scored examinee answers ("1" correct, "0" incorrect) and columns correspond to the items. In addition, `Data` can hold the vector of group membership. If so, `group` is a column indicator of `Data`. Otherwise, `group` must be a dichotomous vector of the same length as `nrow(Data)`.

The unconstrained form of 4PL generalized logistic regression model for probability of correct answer (i.e., $y = 1$) is

$$P(y = 1) = (c + cDif * g) + (d + dDif * g - c - cDif * g) / (1 + \exp(-(a + aDif * g) * (x - b - bDif * g))),$$

where x is by default standardized total score (also called Z-score) and g is group membership. Parameters a , b , c and d are discrimination, difficulty, guessing and inattention. Terms $aDif$, $bDif$, $cDif$ and $dDif$ then represent differences between two groups in relevant parameters.

This 4PL model can be further constrained by `model` and `constraints` arguments. The arguments `model` and `constraints` can be also combined.

The `model` argument offers several predefined models. The options are as follows: `Rasch` for 1PL model with discrimination parameter fixed on value 1 for both groups, `1PL` for 1PL model with discrimination parameter fixed for both groups, `2PL` for logistic regression model, `3PLcg` for 3PL model with fixed guessing for both groups, `3PLdg` for 3PL model with fixed inattention for both groups, `3PLc` (alternatively also `3PL`) for 3PL regression model with guessing parameter, `3PLd` for 3PL model with inattention parameter, `4PLcgdg` for 4PL model with fixed guessing and inattention parameter for both groups, `4PLcgd` (alternatively also `4PLd`) for 4PL model with fixed guessing for both groups, `4PLcdg` (alternatively also `4PLc`) for 4PL model with fixed inattention for both groups, or `4PL` for 4PL model.

The `model` can be specified in more detail with `constraints` argument which specifies what parameters should be fixed for both groups. For example, choice "`ad`" means that discrimination (a) and inattention (d) are fixed for both groups and other parameters (b and c) are not. The arguments `model` and `constraints` can be also item specific if they take a form of vector, where each element correspond to one item. The `NA` value for `constraints` means no constraints.

The `type` corresponds to type of DIF to be tested. Possible values are "`both`" to detect any DIF caused by difference in difficulty or discrimination (i.e., uniform and/or non-uniform), "`udif`" to detect only uniform DIF (i.e., difference in difficulty b), "`nudif`" to detect only non-uniform DIF (i.e., difference in discrimination a), or "`all`" to detect DIF caused by difference caused by any parameter that can differed between groups. The type of DIF can be also specified in more detail by using combination of parameters a , b , c and d . For example, with an option "`c`" for 4PL model only the difference in parameter c is tested. The `type` argument is also item specific.

Argument `match` represents the matching criterion. It can be either the standardized test score (default, "`zscore`"), total test score ("`score`"), or any other continuous or discrete variable of the same length as number of observations in `Data`.

A set of anchor items (DIF free) can be specified through the `anchor` argument. It need to be a vector of either item names (as specified in column names of `Data`) or item identifiers (integers specifying the column number). In case anchor items are provided, only these items are used to compute matching criterion match. If the `match` argument is not either "`zscore`" or "`score`", `anchor` argument is ignored. When anchor items are provided, purification is not applied.

The `p.adjust.method` is a character for `p.adjust` function from the `stats` package. Possible values are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none".

The `start` is a list with as many elements as number of items. Each element is a named numeric vector of length 8 representing initial values for parameter estimation. Specifically, parameters `a`, `b`, `c`, and `d` are initial values for discrimination, difficulty, guessing and inattention for reference group. Parameters `aDif`, `bDif`, `cDif` and `dDif` are then differences in these parameters between reference and focal group. If not specified, starting values are calculated with `startNLR` function.

Missing values are allowed but discarded for item estimation. They must be coded as NA for both, data and group parameters.

In case of convergence issues, with an option `ini.tboot = TRUE`, the starting values are re-calculated based on bootstrapped samples. Newly calculated initial values are applied only to items/models with convergence issues.

In case that model considers difference in guessing or inattention parameter, the different parameterization is used and parameters with standard errors are recalculated by delta method. However, covariance matrices stick with alternative parameterization.

Value

The `diFNLR()` function returns an object of class "diFNLR". The output is displayed by the `print()` method.

Item characteristic curves and graphical representation of DIF statistics can be displayed with `plot()` method. For more details see `plot.diFNLR`. Estimated parameters can be displayed with `coef()` method.

Fitted values can be extracted by the `fitted()` method for converged item(s) specified in `item` argument.

Predicted values are produced by the `predict()` method for converged item(s) specified in `item` argument. New data can be introduced with `match` and `group` arguments. For more details see `predict.diFNLR`.

Residuals are extracted with the `residuals()` method for converged item(s) specified in `item` argument.

Log-likelihood, Akaike's information criterion and Schwarz's Bayesian criterion can be extracted with methods `logLik()`, `AIC()`, `BIC()` for converged item(s) specified in `item` argument.

Object of class "diFNLR" is a list with the following components:

`Sval` the values of test statistics.

`nlrPAR` the estimates of final model.

`nlrSE` the standard errors of estimates of final model.

`parM0` the estimates of null model.

`seM0` the standard errors of estimates of null model.

`covM0` the covariance matrices of estimates of null model.

`parM1` the estimates of alternative model.

`seM1` the standard errors of estimates of alternative model.

`covM1` the covariance matrices of estimates of alternative model.

`alpha` numeric: significance level.
`DIFitems` either the column indicators of the items which were detected as DIF, or "No DIF item detected".
`match` matching criterion.
`model` fitted model.
`type` character: type of DIF that was tested. If parameters were specified, the value is "other".
`types` character: the parameters (specified by user, type has value "other") which were tested for difference.
`p.adjust.method` character: method for multiple comparison correction which was applied.
`pval` the p-values by likelihood ratio test.
`adj.pval` the adjusted p-values by likelihood ratio test using `p.adjust.method`.
`df` the degrees of freedom of likelihood ratio test.
`test` used test.
`purification` purify value.
`nrPur` number of iterations in item purification process. Returned only if `purify` is TRUE.
`difPur` a binary matrix with one row per iteration of item purification and one column per item. "1" in *i*-th row and *j*-th column means that *j*-th item was identified as DIF in *i*-1-th iteration. Returned only if `purify` is TRUE.
`conv.puri` logical: indicating whether item purification process converged before the maximal number `nrIter` of iterations. Returned only if `purify` is TRUE.
`group` the vector of group membership.
`Data` the data matrix.
`method` used estimation method.
`conv.fail` numeric: number of convergence issues.
`conv.fail.which` the indicators of the items which did not converge.
`llM0` log-likelihood of null model.
`llM1` log-likelihood of alternative model.

Author(s)

Adela Hladka (nee Drabinova)
 Institute of Computer Science, The Czech Academy of Sciences
 Faculty of Mathematics and Physics, Charles University
 hladka@cs.cas.cz

Patricia Martinkova
 Institute of Computer Science, The Czech Academy of Sciences
 martinkova@cs.cas.cz

Karel Zvara
 Faculty of Mathematics and Physics, Charles University

References

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

Swaminathan, H. & Rogers, H. J. (1990). Detecting Differential Item Functioning Using Logistic Regression Procedures. *Journal of Educational Measurement*, 27, 361-370.

See Also

[nls p.adjust plot.difNLR startNLR](#)

Examples

```
## Not run:
# loading data based on GMAT
data(GMAT)

Data <- GMAT[, 1:20]
group <- GMAT[, "group"]

# Testing both DIF effects using likelihood-ratio test and
# 3PL model with fixed guessing for groups
(x <- difNLR(Data, group, focal.name = 1, model = "3PLcg"))

# Testing both DIF effects using F test and
# 3PL model with fixed guessing for groups
difNLR(Data, group, focal.name = 1, model = "3PLcg", test = "F")

# Testing both DIF effects using LR test,
# 3PL model with fixed guessing for groups
# and Benjamini-Hochberg correction
difNLR(Data, group, focal.name = 1, model = "3PLcg", p.adjust.method = "BH")

# Testing both DIF effects using LR test,
# 3PL model with fixed guessing for groups
# and item purification
difNLR(Data, group, focal.name = 1, model = "3PLcg", purify = T)

# Testing both DIF effects using 3PL model with fixed guessing for groups
# and total score as matching criterion
difNLR(Data, group, focal.name = 1, model = "3PLcg", match = "score")

# Testing uniform DIF effects using 4PL model with the same
# guessing and inattention
difNLR(Data, group, focal.name = 1, model = "4PLcgdg", type = "udif")

# Testing non-uniform DIF effects using 2PL model
difNLR(Data, group, focal.name = 1, model = "2PL", type = "nudif")

# Testing difference in parameter b using 4PL model with fixed
# a and c parameters
```

```

difNLR(Data, group, focal.name = 1, model = "4PL", constraints = "ac", type = "b")

# Testing both DIF effects using LR test,
# 3PL model with fixed guessing for groups
# with maximum likelihood estimation method
difNLR(Data, group, focal.name = 1, model = "3PLcg", method = "likelihood")

# Graphical devices
plot(x)
plot(x, item = x$DIFitems)
plot(x, item = 1, group.names = c("Group 1", "Group 2"))
plot(x, plot.type = "stat")

# Coefficients
coef(x)
coef(x, SE = T)
coef(x, SE = T, simplify = T)

# Fitted values
fitted(x)
fitted(x, item = 1)

# Residuals
residuals(x)
residuals(x, item = 1)

# Predicted values
predict(x)
predict(x, item = 1)

# Predicted values for new subjects
predict(x, item = 1, match = 0, group = 1)
predict(x, item = 1, match = 0, group = 0)

# AIC, BIC, log-likelihood
AIC(x); BIC(x); logLik(x)
# AIC, BIC, log-likelihood for the first item
AIC(x, item = 1); BIC(x, item = 1); logLik(x, item = 1)

## End(Not run)

```

estimNLR

Non-Linear Regression DIF models estimation.

Description

Estimates parameters of non-linear regression models for DIF detection using either non-linear least squares or maximum likelihood method.

Usage

```
estimNLR(y, match, group, formula, method, lower, upper, start)
```

Arguments

y	numeric: binary vector. See Details .
match	numeric: matching criterion. See Details .
group	numeric: binary vector of group membership. "0" for reference group, "1" for focal group.
formula	formula: specification of the model. See Details .
method	character: method used to estimate parameters. The options are "nls" for non-linear least squares (default) and "likelihood" for maximum likelihood method.
lower	numeric: lower bounds for parameters of model specified in formula.
upper	numeric: upper bounds for parameters of model specified in formula.
start	numeric: initial parameters. See Details .

Author(s)

Adela Hladka (nee Drabinova)
 Institute of Computer Science, The Czech Academy of Sciences
 Faculty of Mathematics and Physics, Charles University
 hladka@cs.cas.cz

Patricia Martinkova
 Institute of Computer Science, The Czech Academy of Sciences
 martinkova@cs.cas.cz

Examples

```
## Not run:
data(GMAT)

# item 1
y <- GMAT[, 1]
match <- scale(apply(GMAT[, 1:20], 1, sum))
group <- GMAT[, "group"]

# formula for 3PL model with the same guessing
M <- formulaNLR(model = "3PLcg", type = "both")

# starting values for 3PL model with the same guessing for item 1
start <- startNLR(GMAT[, 1:20], group, model = "3PLcg", parameterization = "classic")
start <- start[[1]][M$M0$parameters]

# Non-linear least squares
fitNLSM0 <- estimNLR(y = y, match = match, group = group,
```

```

        formula = M$M0$formula, method = "nls",
        lower = M$M0$lower, upper = M$M0$upper, start = start)

fitNLSM0

coef(fitNLSM0)
logLik(fitNLSM0)
vcov(fitNLSM0)
fitted(fitNLSM0)
residuals(fitNLSM0)

# Maximum likelihood
fitLKLM0 <- estimNLR(y = y, match = match, group = group,
                    formula = M$M0$formula, method = "likelihood",
                    lower = M$M0$lower, upper = M$M0$upper, start = start)

fitLKLM0

coef(fitLKLM0)
logLik(fitLKLM0)
vcov(fitLKLM0)
fitted(fitLKLM0)
residuals(fitLKLM0)

## End(Not run)

```

formulaNLR

Formula for Non-Linear Regression DIF model.

Description

Function returns the formula of the non-linear models based on model specification and DIF type to be tested.

Usage

```
formulaNLR(model, constraints = NULL, type = "both", parameterization = "classic",
outcome)
```

Arguments

model	character: generalized logistic regression model to be fitted. See Details .
constraints	character: which parameters should be the same for both groups. Default value is NULL. See Details .
type	character: type of DIF to be tested. Possible values are "both" (default), "udif", "nudif", "all", or combination of parameters 'a', 'b', 'c' and 'd'. See Details .
parameterization	character: which parameterization should be used. Possible values are "classic" (default) and "alternative". See Details .
outcome	character: name of outcome to be printed in formula. If not specified 'y' is used.

Details

The unconstrained form of 4PL generalized logistic regression model for probability of correct answer (i.e., $y = 1$) is $P(y = 1) = (c + cDif*g) + (d + dDif*g - c - cDif*g)/(1 + \exp(-(a + aDif*g)*(x - b - bDif*g)))$, where x is standardized total score (also called Z-score) and g is group membership. Parameters a , b , c and d are discrimination, difficulty, guessing and inattention. Parameters $aDif$, $bDif$, $cDif$ and $dDif$ then represent differences between two groups in discrimination, difficulty, guessing and inattention.

This 4PL model can be further constrained by `model` and `constraints` arguments. The arguments `model` and `constraints` can be also combined.

The `model` argument offers several predefined models. The options are as follows: `Rasch` for 1PL model with discrimination parameter fixed on value 1 for both groups, `1PL` for 1PL model with discrimination parameter fixed for both groups, `2PL` for logistic regression model, `3PLcg` for 3PL model with fixed guessing for both groups, `3PLdg` for 3PL model with fixed inattention for both groups, `3PLc` (alternatively also `3PL`) for 3PL regression model with guessing parameter, `3PLd` for 3PL model with inattention parameter, `4PLcgdg` for 4PL model with fixed guessing and inattention parameter for both groups, `4PLcgd` (alternatively also `4PLd`) for 4PL model with fixed guessing for both groups, `4PLcdg` (alternatively also `4PLc`) for 4PL model with fixed inattention for both groups, or `4PL` for 4PL model.

The `model` can be specified in more detail with `constraints` argument which specifies what arguments should be fixed for both groups. For example, choice `'ad'` means that discrimination (a) and inattention (d) are fixed for both groups and other parameters (b and c) are not.

The type corresponds to type of DIF to be tested. Possible values are `"both"` to detect any DIF caused by difference in difficulty or discrimination (i.e., uniform and/or non-uniform), `"udif"` to detect only uniform DIF (i.e., difference in difficulty b), `"nudif"` to detect only non-uniform DIF (i.e., difference in discrimination a), or `"all"` to detect DIF caused by difference caused by any parameter that can differ between groups. The type of DIF can be also specified in more detail by using combination of parameters a , b , c and d . For example, with an option `'c'` for 4PL model only the difference in parameter c is tested.

For an option `"alternative"` in `parameterization` argument, all models with the different guessing or/and inattention parameters are reparameterized as follows: $P(y = 1) = (cR*(1-g) + cF*g) + (dR*(1-g) + dF*g - cR*(1-g) - cF*g)/(1 + \exp(-(a + aDif*g)*(x - b - bDif*g)))$.

Value

A list of two models. Both includes formula, parameters to be estimated and their lower and upper constraints.

Author(s)

Adela Hladka (nee Drabinova)
 Institute of Computer Science, The Czech Academy of Sciences
 Faculty of Mathematics and Physics, Charles University
 hladka@cs.cas.cz

Patricia Martinkova
 Institute of Computer Science, The Czech Academy of Sciences

martinkova@cs.cas.cz

See Also

[difNLR](#)

Examples

```
## Not run:
# 3PL model with the same guessing for both groups
# to test both types of DIF
formulaNLR(model = "3PLcg", type = "both")

# 4PL model with the same guessing and inattention
# to test uniform DIF
formulaNLR(model = "4PLcgdg", type = "udif")

# 2PL model to test non-uniform DIF
formulaNLR(model = "2PL", type = "nudif")

# 4PL model to test all possible DIF
# with alternative parameterization
formulaNLR(model = "4PL", type = "all", parameterization = "alternative")

# 4PL model with fixed a and c parameter
# to test difference in b with alternative parameterization
formulaNLR(model = "4PL", constraints = "ac", type = "b", parameterization = "alternative")

## End(Not run)
```

genNLR

Generates data set based on Non-Linear Regression DIF a DDF models.

Description

Generates dichotomous and nominal data set based on non-linear regression models for DIF and DDF detection.

Usage

```
genNLR(N = 1000, ratio = 1, itemtype = "dich", a, b, c, d, mu = 0, sigma = 1)
```


Arguments

N	numeric: number of rows representing respondents.
ratio	numeric: ratio of respondents number in reference and focal group.
itemtype	character: type of items to be generated. Options are "dich" for dichotomous item (default), "nominal" for nominal items, and "ordinal" for ordinal data. See Details .
a	numeric: matrix representing discriminations with m rows (where m is number of items). Need to be provided. See Details .
b	numeric: numeric: matrix representing difficulties with m rows (where m is number of items). Need to be provided. See Details .
c	numeric: matrix representing guessings (lower asymptotes) with m rows (where m is number of items). Default is NULL. See Details .
d	numeric: matrix representing inattentions (upper asymptotes) with m rows (where m is number of items). Default is NULL. See Details .
mu	numeric: a mean vector of the underlying distribution. The first value corresponds to reference group, the second to focal group. Default is 0 value for both groups. See Details .
sigma	numeric: a standard deviation vector of the underlying distribution. The first value corresponds to reference group, the second to focal group. Default is 1 value for both groups. See Details .

Details

The `itemtype` argument specify what type of item should be generated. In case `itemtype = "dich"`, dichotomous items are generated with non-linear regression model (generalized logistic regression model) for DIF detection specified in [difNLR](#). In case `itemtype = "nominal"`, nominal items are generated with multinomial model specified in [ddfMLR](#). For option `itemtype = "ordinal"`, ordinal items are generated with adjacent logit model specified in [ddfORD](#) with argument `model = "adjacent"`.

The `a`, `b`, `c` and `d` are numeric matrices with `m` rows (where `m` is number of items) representing parameters of regression models for DIF and DDF detection.

For option `itemtype = "dich"`, matrices should have two columns. The first column represents parameters of the reference group and the second of the focal group. In case that only one column is provided, parameters are set to be the same for both groups.

For options `itemtype = "nominal"` and `itemtype = "ordinal"`, matrices `c` and `d` are ignored. Matrices `a` and `b` contain parameters for distractors. For example, when item with 4 different choices is supposed to be generated, user provide matrices with 6 columns. First 3 columns correspond to distractors parameters for reference group and last three columns for focal group. The number of choices can differ for items. Matrices `a` and `b` need to consist of as many columns as is the maximum number of distractors. Items with less choices can contain NAs.

Single value for `mu` means that reference and focal group have underlying distribution with the same mean. Single value for `sigma` means that reference and focal group have underlying distribution with the same standard deviation. In case that `mu` or `sigma` are vectors of length greater than two, only first two values are taken.

Value

A data frame containing N rows representing respondents and $m + 1$ columns representing m items. Last column is group membership variable with coding 0 for reference group and 1 for focal group.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

See Also

[difNLR](#), [ddfMLR](#), [ddfORD](#)

Examples

```
## Not run:
# seed
set.seed(123)
# generating parameters for dichotomous data with DIF, 5 items
a <- matrix(runif(10, 0.8, 2), ncol = 2)
b <- matrix(runif(10, -2, 2), ncol = 2)
c <- matrix(runif(10, 0, 0.25), ncol = 2)
d <- matrix(runif(10, 0.8, 1), ncol = 2)
# generating dichotomous data set with 300 observations (150 each group)
genNLR(N = 300, a = a, b = b, c = c, d = d)
# generating dichotomous data set with 300 observations (150 each group)
# and different mean and standard deviation for underlying distribution
genNLR(N = 300, a = a, b = b, c = c, d = d, mu = c(1, 0), sigma = c(1, 2))
# generating dichotomous data set with 300 observations (250 reference group, 50 focal)
genNLR(N = 300, ratio = 5, a = a, b = b, c = c, d = d)

# generating parameters for nominal data with DDF, 5 items,
# each item 3 choices
a <- matrix(runif(20, 0.8, 2), ncol = 4)
b <- matrix(runif(20, -2, 2), ncol = 4)
# generating nominal data set with 300 observations (150 each group)
genNLR(N = 300, itemtype = "nominal", a = a, b = b)
```

```

# generating nominal data set with 300 observations (250 reference group, 50 focal)
genNLR(N = 300, itemtype = "nominal", ratio = 5, a = a, b = b)

# generating parameters for nominal data with DDF, 5 items,
# items 1 and 2 have 2 choices, items 3, 4 and 5 have 3 choices
a <- matrix(runif(20, 0.8, 2), ncol = 4)
a[1:2, c(2, 4)] <- NA
b <- matrix(runif(20, -2, 2), ncol = 4)
b[1:2, c(2, 4)] <- NA
# generating nominal data set with 300 observations (150 each group)
genNLR(N = 300, itemtype = "nominal", a = a, b = b)
# generating nominal data set with 300 observations (250 reference group, 50 focal)
genNLR(N = 300, itemtype = "nominal", ratio = 5, a = a, b = b)

## End(Not run)

```

GMAT

Dichotomous Data Set Based on Graduate Management Admission Test with the Same Total Score Distribution for Groups

Description

The GMAT data set is generated data set based on parameters from Graduate Management Admission Test (GMAT) data set (Kingston et al., 1985). First two items were considered to function differently in uniform and non-uniform way respectively. The data set represents responses of 2,000 subjects to multiple-choice test of 20 items. A correct answer is coded as 1 and incorrect answer as 0. The column group represents group membership, where 0 represents reference group and 1 represents focal group. Groups are the same size (i.e. 1,000 per group). The distributions of total scores (sum of correct answers) are the same for both reference and focal group (Martinkova et al., 2016). The column criterion represents generated continuous variable which is intended to be predicted by test.

Usage

```
data(GMAT)
```

Format

A GMAT data frame consists of 2,000 observations on the following 22 variables. The first 20 columns represent dichotomously scored items of the test. The 21st column is vector of group membership; values 0 and 1 refer to reference and focal group. The 22nd column is vector representing variable which is intended to be predicted by test.

Author(s)

Adela Hladka (nee Drabinova)
 Institute of Computer Science, The Czech Academy of Sciences
 Faculty of Mathematics and Physics, Charles University

hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Kingston, N., Leary, L., & Wightman, L. (1985). An Exploratory Study of the Applicability of Item Response Theory Methods to the Graduate Management Admission Test. ETS Research Report Series, 1985(2) : 1-64.

Martinkova, P., Drabinova, A., Liaw, Y. L., Sanders, E. A., McFarland, J. L., & Price, R. M. (2017). Checking equity: Why Differential Item Functioning Analysis Should Be a Routine Part of Developing Conceptual Assessments. CBE-Life Sciences Education, 16(2), <https://doi.org/10.1187/cbe.16-10-0307>.

See Also

[GMATtest](#), [GMATkey](#)

GMAT2

Dichotomous Data Set Based on Graduate Management Admission Test

Description

The GMAT2 data set is generated data set based on parameters from Graduate Management Admission Test (GMAT) data set (Kingston et al., 1985). First two items were considered to function differently in uniform and non-uniform way respectively. The data set represents responses of 1,000 subjects to multiple-choice test of 20 items. A correct answer is coded as 1 and incorrect answer as 0. The column group represents group membership, where 0 represents reference group and 1 represents focal group. Groups are the same size (i.e. 500 per group).

Usage

```
data(GMAT2)
```

Format

A GMAT2 data frame consists of 1,000 observations on the following 21 variables. The first 20 columns represent dichotomously scored items of the test. The 21st column is vector of group membership; values 0 and 1 refer to reference and focal group.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Kingston, N., Leary, L., & Wightman, L. (1985). An Exploratory Study of the Applicability of Item Response Theory Methods to the Graduate Management Admission Test. ETS Research Report Series, 1985(2) : 1-64.

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. Journal of Educational Measurement, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

See Also

[GMAT2test](#), [GMAT2key](#)

GMAT2key

Key of Correct Answers for GMAT2test Data Set

Description

The GMAT2key is a vector of factors representing correct answers of generated GMAT2test data set based on Graduate Management Admission Test (GMAT) data set (Kingston et al., 1985).

Usage

```
data(GMAT2key)
```

Format

A data frame with 20 values representing correct answers to items of GMAT2test data set. For more details see [GMAT2test](#).

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Kingston, N., Leary, L., & Wightman, L. (1985). An Exploratory Study of the Applicability of Item Response Theory Methods to the Graduate Management Admission Test. ETS Research Report Series, 1985(2) : 1-64.

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. Journal of Educational Measurement, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

See Also

[GMAT2](#), [GMAT2test](#)

GMAT2test

Data Set Based on Graduate Management Admission Test

Description

The GMAT2test data set is generated data set based on parameters from Graduate Management Admission Test (GMAT) data set (Kingston et al., 1985). First two items were considered to function differently in uniform and non-uniform way respectively. The data set represents responses of 1,000 subjects to multiple-choice test of 20 items. Additionally, 4 possible answers on all items were generated, coded A, B, C and D. The column group represents group membership, where 0 represents reference group and 1 represent focal group. Groups are the same size (i.e. 500 per group).

Usage

`data(GMAT2test)`

Format

A GMAT2 data frame consists of 1,000 observations on the following 21 variables. The first 20 columns represents answers of subject to an items of the test. The 21st column is vector of group membership; values 0 and 1 refer to reference and focal group. Correct answers are presented in [GMAT2key](#) data set.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Kingston, N., Leary, L., & Wightman, L. (1985). An Exploratory Study of the Applicability of Item Response Theory Methods to the Graduate Management Admission Test. ETS Research Report Series, 1985(2) : 1-64.

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. Journal of Educational Measurement, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

See Also

[GMAT2](#), [GMAT2key](#)

GMATkey

Key of Correct Answers for GMATtest Data Set

Description

The GMATkey is a vector of factors representing correct answers of generated GMATtest data set based on Graduate Management Admission Test (GMAT) data set (Kingston et al., 1985).

Usage

```
data(GMATkey)
```

Format

A data frame with 20 values representing correct answers to items of GMATtest data set. For more details see [GMATtest](#).

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Kingston, N., Leary, L., & Wightman, L. (1985). An Exploratory Study of the Applicability of Item Response Theory Methods to the Graduate Management Admission Test. ETS Research Report Series, 1985(2) : 1-64.

Martinkova, P., Drabinova, A., Liaw, Y. L., Sanders, E. A., McFarland, J. L., & Price, R. M. (2017). Checking equity: Why Differential Item Functioning Analysis Should Be a Routine Part of Developing Conceptual Assessments. CBE-Life Sciences Education, 16(2), <https://doi.org/10.1187/cbe.16-10-0307>.

See Also

[GMAT](#), [GMATtest](#)

GMATtest

Data Set Based on Graduate Management Admission Test with the Same Total Score Distribution for Groups

Description

The GMATtest data set is generated data set based on parameters from Graduate Management Admission Test (GMAT) data set (Kingston et al., 1985). First two items were considered to function differently in uniform and non-uniform way respectively. The data set represents responses of 2,000 subjects to multiple-choice test of 20 items. Additionally, 4 possible answers on all items were generated, coded A, B, C and D. The column group represents group membership, where 0 represents reference group and 1 represent focal group. Groups are the same size (i.e. 1,000 per group). The distributions of total scores (sum of correct answers) are the same for both reference and focal group (Martinkova et al., 2016). The column criterion represents generated continuous variable which is intended to be predicted by test.

Usage

data(GMATtest)

Format

A GMAT data frame consists of 2,000 observations on the following 21 variables. The first 20 columns represents answers of subject to an items of the test. The 21st column is vector of group membership; values 0 and 1 refer to reference and focal group. The 22nd column is vector representing variable which is intended to be predicted by test. Correct answers are presented in [GMATkey](#) data set.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Kingston, N., Leary, L., & Wightman, L. (1985). An Exploratory Study of the Applicability of Item Response Theory Methods to the Graduate Management Admission Test. ETS Research Report Series, 1985(2) : 1-64.

Martinkova, P., Drabinova, A., Liaw, Y. L., Sanders, E. A., McFarland, J. L., & Price, R. M. (2017). Checking equity: Why Differential Item Functioning Analysis Should Be a Routine Part of Developing Conceptual Assessments. CBE-Life Sciences Education, 16(2), <https://doi.org/10.1187/cbe.16-10-0307>.

See Also

[GMAT](#), [GMATkey](#)

MLR

DDF likelihood ratio statistics based on multinomial log-linear regression model.

Description

Calculates DDF likelihood ratio statistics for nominal data based on multinomial log-linear model.

Usage

```
MLR(Data, group, key, type = "both", match = "zscore", anchor = 1:ncol(Data),  
p.adjust.method = "none", parametrization = "irt", alpha = 0.05)
```

Arguments

<code>Data</code>	character: the unscored data matrix.
<code>group</code>	numeric or character: the binary vector of group membership
<code>key</code>	character: the answer key.
<code>type</code>	character: type of DDF to be tested (either "both" (default), "udif", or "nudif"). See Details .
<code>match</code>	specifies matching criterion. Can be either "zscore" (default, standardized total score), "score" (total test score), or vector of the same length as number of observations in <code>Data</code> . See Details .
<code>anchor</code>	a vector of integers specifying which items are currently considered as anchor (DDF free) items. By default, all items are considered as anchors. Argument is ignored if <code>match</code> is not "zscore" or "score". See Details .
<code>p.adjust.method</code>	character: method for multiple comparison correction. See Details .
<code>parametrization</code>	character: parametrization of regression coefficients. Possible options are "irt" (default) and "classic". See Details .
<code>alpha</code>	numeric: significance level (default is 0.05).

Details

Calculates DDF likelihood ratio statistics based on multinomial log-linear model.

The `Data` is a matrix which rows represents examinee unscored answers and columns correspond to the items. The `group` must be a vector of the same length as `nrow(Data)`. The `key` must be a vector of correct answers corresponding to columns of `Data`.

The `type` corresponds to type of DDF to be tested. Possible values are "both" to detect any DDF (uniform and/or non-uniform), "udif" to detect only uniform DDF or "nudif" to detect only non-uniform DDF.

Argument `match` represents the matching criterion. It can be either the standardized test score (default, "zscore"), total test score ("score"), or any other continuous or discrete variable of the same length as number of observations in `Data`. Matching criterion is used in NLR function as a covariate in non-linear regression model.

The `p.adjust.method` is a character for `p.adjust` function from the `stats` package. Possible values are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none". See also [p.adjust](#) for more information.

Argument `parametrization` is a character which specifies parametrization of regression parameters. Default option is "irt" which returns IRT parametrization (difficulty-discrimination). Option "classic" returns intercept-slope parametrization with effect of group membership and interaction with matching criterion.

Value

A list with the following arguments:

`Sval` the values of likelihood ratio test statistics.

`pval` the p-values by likelihood ratio test.
`adj.pval` the adjusted p-values by likelihood ratio test using `p.adjust.method`.
`df` the degrees of freedom of likelihood ratio test.
`par.m0` the estimates of null model.
`par.m1` the estimates of alternative model.
`se.m0` standard errors of parameters in null model.
`se.m1` standard errors of parameters in alternative model.
`ll.m0` log-likelihood of m0 model.
`ll.m1` log-likelihood of m1 model.
`AIC.m0` AIC of m0 model.
`AIC.m1` AIC of m1 model.
`BIC.m0` BIC of m0 model.
`BIC.m1` BIC of m1 model.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

See Also

[p.adjust](#)

Examples

```
## Not run:  
# loading data based on GMAT  
data(GMATtest, GMATkey)  
  
Data <- GMATtest[, 1:20]  
group <- GMATtest[, "group"]  
key <- GMATkey  
  
# Testing both DDF effects  
MLR(Data, group, key, type = "both")  
  
# Testing uniform DDF effects  
MLR(Data, group, key, type = "udif")
```

```
# Testing non-uniform DDF effects
MLR(Data, group, key, type = "nudif")

## End(Not run)
```

MSATB

Dichotomous Data Set of Medical School Admission Test in Biology

Description

The MSATB data set consists of the responses of 1,407 subjects (484 males, 923 females) to admission test to medical school in Czech republic. It contains 20 selected items from original test while first item was previously detected as differently functioning (Vlckova, 2014). A correct answer is coded as 1 and incorrect answer as 0. The column gender represents gender of students, where 0 represents males (reference group) and 1 represents females (focal group).

Usage

```
data(MSATB)
```

Format

A MSATB data frame consists of 1,407 observations on the following 21 variables. The first 20 columns represent dichotomously scored items of the test. The 21st column is vector of group membership; values 0 and 1 refer to males (reference group) and females (focal group).

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

- Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.
- Vlckova, K. (2014). Test and Item Fairness (Unpublished master's thesis).

See Also

[MSATBtest](#), [MSATBkey](#)

MSATBkey

Key of Correct Answers for MSATBtest Data Set

Description

The MSATBkey is a vector of factors representing correct answers of MSATBtest data set.

Usage

```
data(MSATBkey)
```

Format

A data frame with 20 values representing correct answers to items of MSATBtest data set. For more details see [MSATBtest](#).

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

Vlckova, K. (2014). Test and Item Fairness (Unpublished master's thesis).

See Also

[MSATB](#), [MSATBtest](#)

MSATBtest

Data Set of School Admission Test in Biology

Description

The MSATBtest data set consists of the responses of 1,407 subjects (484 males, 923 females) to multiple-choice admission test to medical school in Czech republic. It contains 20 selected items from original test while first item was previously detected as differently functioning (Vlckova, 2014). Possible answers were A, B, C, D, while any combination of these can be correct. The column gender represents gender of students, where 0 represents males (reference group) and 1 represents females (focal group).

Usage

```
data(MSATBtest)
```

Format

A MSATBtest data frame consists of 1,407 observations on the following 21 variables. The first 20 columns represent answers of subject to an items of the test. The 21st column is vector subjects' gender; values 0 and 1 refer to males (reference group) and females (focal group).

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

Vlckova, K. (2014). Test and Item Fairness (Unpublished master's thesis).

See Also

[MSATB](#), [MSATBkey](#)

Description

Calculates either DIF likelihood ratio or F statistics for dichotomous data based on non-linear regression model (generalized logistic regression model).

Usage

```
NLR(Data, group, model, constraints = NULL, type = "both", method = "nls",
match = "zscore", anchor = 1:ncol(Data), start, p.adjust.method = "none", test = "LR",
alpha = 0.05, initboot = T, nrBo = 20)
```

Arguments

Data	numeric: either binary data matrix only, or the binary data matrix plus the vector of group . See Details .
group	numeric: binary vector of group membership. "0" for reference group, "1" for focal group.
model	character: generalized logistic regression model to be fitted. See Details .
constraints	character: which parameters should be the same for both groups. Default value is NULL. See Details .
type	character: type of DIF to be tested. Possible values are "both" (default), "udif", "nudif", "all", or combination of parameters "a", "b", "c" and "d". See Details .
method	character: what method should be used for estimation of parameters in model. The options are "nls" for non-linear least squares (default) and "likelihood" for maximum likelihood method.
match	specifies matching criterion. Can be either "zscore" (default, standardized total score), "score" (total test score), or vector of the same length as number of observations in Data. See Details .
anchor	a vector of integers specifying which items are currently considered as anchor (DIF free) items. By default, all items are considered as anchors. Argument is ignored if match is not "zscore" or "score". See Details .
start	numeric: matrix with n rows (where n is the number of items) and 8 columns containing initial item parameters estimates. See Details .
p.adjust.method	character: method for multiple comparison correction. See Details .
test	character: test to be performed for DIF detection (either "LR" (default), or "F"). See Details .
alpha	numeric: significance level (default is 0.05).
initboot	logical: in case of convergence issues, should be starting values recalculated based on bootstrapped samples? (default is TRUE). See Details .

`nrBo` numeric: the maximal number of iterations for calculation of starting values using bootstrapped samples (default is 20).

Details

DIF detection procedure based on Non-Linear Regression is the extension of Logistic Regression procedure (Swaminathan and Rogers, 1990).

The `Data` is a matrix which rows represents examinee scored answers (1 - correct, 0 - incorrect) and columns correspond to the items. The `group` must be a vector of the same length as `nrow(Data)`.

The unconstrained form of 4PL generalized logistic regression model for probability of correct answer (i.e., $y = 1$) is

$$P(y = 1) = (c + cDif * g) + (d + dDif * g - c - cDif * g) / (1 + \exp(-(a + aDif * g) * (x - b - bDif * g))),$$

where x is standardized total score (also called Z-score) and g is group membership. Parameters a , b , c and d are discrimination, difficulty, guessing and inattention. Parameters $aDif$, $bDif$, $cDif$ and $dDif$ then represent differences between two groups in discrimination, difficulty, guessing and inattention.

This 4PL model can be further constrained by `model` and `constraints` arguments. The arguments `model` and `constraints` can be also combined.

The `model` argument offers several predefined models. The options are as follows: `Rasch` for 1PL model with discrimination parameter fixed on value 1 for both groups, `1PL` for 1PL model with discrimination parameter fixed for both groups, `2PL` for logistic regression model, `3PLcg` for 3PL model with fixed guessing for both groups, `3PLdg` for 3PL model with fixed inattention for both groups, `3PLc` (alternatively also `3PL`) for 3PL regression model with guessing parameter, `3PLd` for 3PL model with inattention parameter, `4PLcgdg` for 4PL model with fixed guessing and inattention parameter for both groups, `4PLcgd` (alternatively also `4PLd`) for 4PL model with fixed guessing for both groups, `4PLcdg` (alternatively also `4PLc`) for 4PL model with fixed inattention for both groups, or `4PL` for 4PL model.

The `model` can be specified in more detail with `constraints` argument which specifies what arguments should be fixed for both groups. For example, choice `"ad"` means that discrimination (a) and inattention (d) are fixed for both groups and other parameters (b and c) are not.

The `type` corresponds to type of DIF to be tested. Possible values are `"both"` to detect any DIF caused by difference in difficulty or discrimination (i.e., uniform and/or non-uniform), `"udif"` to detect only uniform DIF (i.e., difference in difficulty b), `"nudif"` to detect only non-uniform DIF (i.e., difference in discrimination a), or `"all"` to detect DIF caused by difference caused by any parameter that can differed between groups. The type of DIF can be also specified in more detail by using combination of parameters a , b , c and d . For example, with an option `"c"` for 4PL model only the difference in parameter c is tested.

Argument `match` represents the matching criterion. It can be either the standardized test score (default, `"zscore"`), total test score (`"score"`), or any other continuous or discrete variable of the same length as number of observations in `Data`. Matching criterion is used in `NLR` function as a covariate in non-linear regression model.

The `start` is a list with as many elements as number of items. Each element is a named numeric vector representing initial values for parameter estimation. Specifically, parameters a , b , c , and d are initial values for discrimination, difficulty, guessing and inattention for reference group. Parameters $aDif$, $bDif$, $cDif$ and $dDif$ are then differences in these parameters between reference and focal group. If not specified, starting values are calculated with `startNLR` function.

The `p.adjust.method` is a character for `p.adjust` function from the `stats` package. Possible values are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none".

In case of convergence issues, with an option `initboot = TRUE`, the starting values are re-calculated based on bootstrapped samples. Newly calculated initial values are applied only to items/models with convergence issues.

In case that model considers difference in guessing or inattention parameter, the different parameterization is used and parameters with standard errors are recalculated by delta method. However, covariance matrices stick with alternative parameterization.

Value

A list with the following arguments:

`Sval` the values of test statistics.

`pval` the p-values by test.

`adjusted.pval` adjusted p-values by `p.adjust.method`.

`df` the degrees of freedom of test.

`test` used test.

`par.m0` the matrix of estimated item parameters for `m0` model.

`se.m0` the matrix of standard errors of item parameters for `m0` model.

`cov.m0` list of covariance matrices of item parameters for `m0` model.

`par.m1` the matrix of estimated item parameters for `m1` model.

`se.m1` the matrix of standard errors of item parameters for `m1` model.

`cov.m1` list of covariance matrices of item parameters for `m1` model.

`conv.fail` numeric: number of convergence issues.

`conv.fail.which` the indicators of the items which did not converge.

`ll.m0` log-likelihood of `m0` model.

`ll.m1` log-likelihood of `m1` model.

`startBo0` the binary matrix. Columns represents iterations of initial values recalculations, rows represents items. The value of 0 means no convergence issue in `m0` model, 1 means convergence issue in `m0` model.

`startBo1` the binary matrix. Columns represents iterations of initial values recalculations, rows represents items. The value of 0 means no convergence issue in `m1` model, 1 means convergence issue in `m1` model.

Author(s)

Adela Hladka (nee Drabinova)

Institute of Computer Science, The Czech Academy of Sciences

Faculty of Mathematics and Physics, Charles University

hladka@cs.cas.cz

Patricia Martinkova

Institute of Computer Science, The Czech Academy of Sciences

`martinkova@cs.cas.cz`

Karel Zvara
Faculty of Mathematics and Physics, Charles University

References

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

Swaminathan, H. & Rogers, H. J. (1990). Detecting Differential Item Functioning Using Logistic Regression Procedures. *Journal of Educational Measurement*, 27, 361-370.

See Also

[p.adjust](#)

Examples

```
## Not run:
# loading data based on GMAT
data(GMAT)

Data <- GMAT[, 1:20]
group <- GMAT[, "group"]

# Testing both DIF effects using LR test (default)
# and model with fixed guessing for both groups
NLR(Data, group, model = "3PLcg")

# Using F test
NLR(Data, group, model = "3PLcg", test = "F")

# Testing both DIF effects with Benjamini-Hochberg correction
NLR(Data, group, model = "3PLcg", p.adjust.method = "BH")

# 4PL model with the same guessing and inattention
# to test uniform DIF
NLR(Data, group, model = "4PLcgdg", type = "udif")

# 2PL model to test non-uniform DIF
NLR(Data, group, model = "2PL", type = "nudif")

# 4PL model with fixed a and c parameter
# to test difference in b
NLR(Data, group, model = "4PL", constraints = "ac", type = "b")

# using maximum likelihood estimation method
NLR(Data, group, model = "3PLcg", method = "likelihood")
```

```
## End(Not run)
```

 ORD

DDF likelihood ratio statistics for ordinal data.

Description

Calculates DDF likelihood ratio statistics for ordinal data based either on adjacent logistic regression model or on cumulative logistic regression model.

Usage

```
ORD(Data, group, model = "adjacent", type = "both", match = "zscore",
     anchor = 1:ncol(Data), p.adjust.method = "none", parametrization = "irt",
     alpha = 0.05)
```

Arguments

Data	matrix or data.frame: the ordinarily scored data matrix.
group	numeric or character: the binary vector of group membership
model	character: logistic regression model for ordinal data (either "adjacent" (default) or "cumulative"). See Details .
type	character: type of DDF to be tested (either "both" (default), "udif", or "nudif"). See Details .
match	specifies matching criterion. Can be either "zscore" (default, standardized total score), "score" (total test score), or vector of the same length as number of observations in Data. See Details .
anchor	a vector of integers specifying which items are currently considered as anchor (DDF free) items. By default, all items are considered as anchors. Argument is ignored if match is not "zscore" or "score". See Details .
p.adjust.method	character: method for multiple comparison correction. See Details .
parametrization	character: parametrization of regression coefficients. Possible options are "irt" (default) and "classic". See Details .
alpha	numeric: significance level (default is 0.05).

Details

Calculates DDF likelihood ratio statistics for ordinal data based either on adjacent logistic regression model or on cumulative logistic regression model

The Data is a matrix or data.frame which rows represents examinee ordinarily scored answers and columns correspond to the items. The group must be a vector of the same length as nrow(Data).

The `model` corresponds to model to be used for DDF detection. Options are "adjacent" for adjacent logistic regression model or "cumulative" for cumulative logistic regression model.

The `type` corresponds to type of DDF to be tested. Possible values are "both" to detect any DDF (uniform and/or non-uniform), "udif" to detect only uniform DDF or "nudif" to detect only non-uniform DDF.

Argument `match` represents the matching criterion. It can be either the standardized test score (default, "zscore"), total test score ("score"), or any other continuous or discrete variable of the same length as number of observations in `Data`.

The `p.adjust.method` is a character for `p.adjust` function from the `stats` package. Possible values are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", and "none". See also [p.adjust](#) for more information.

Argument `parametrization` is a character which specifies parametrization of regression parameters. Default option is "irt" which returns IRT parametrization (difficulty-discrimination). Option "classic" returns intercept-slope parametrization with effect of group membership and interaction with matching criterion.

Value

A list with the following arguments:

`Sval` the values of likelihood ratio test statistics.

`pval` the p-values by likelihood ratio test.

`adj.pval` the adjusted p-values by likelihood ratio test using `p.adjust.method`.

`df` the degrees of freedom of likelihood ratio test.

`par.m0` the estimates of null model.

`par.m1` the estimates of alternative model.

`se.m0` standard errors of parameters in null model.

`se.m1` standard errors of parameters in alternative model.

`ll.m0` log-likelihood of null model.

`ll.m1` log-likelihood of alternative model.

`AIC.m0` AIC of null model.

`AIC.m1` AIC of alternative model.

`BIC.m0` BIC of null model.

`BIC.m1` BIC of alternative model.

Author(s)

Adela Hladka (nee Drabinova)

Institute of Computer Science, The Czech Academy of Sciences

Faculty of Mathematics and Physics, Charles University

hladka@cs.cas.cz

Patricia Martinkova

Institute of Computer Science, The Czech Academy of Sciences

martinkova@cs.cas.cz

See Also

[p.adjust vglm](#)

Examples

```
## Not run:
# loading data
data(dataMedicalgraded, package = "ShinyItemAnalysis")
df <- dataMedicalgraded[, c(1:5, 101)]
df <- df[complete.cases(df), ]

Data <- df[, 1:5]
group <- df[, 6]

# Testing both DDF effects
ORD(Data, group, type = "both")

# Testing uniform DDF effects
ORD(Data, group, type = "udif")

# Testing non-uniform DDF effects
ORD(Data, group, type = "nudif")

# Testing DDF using cumulative logit model
ORD(Data, group, model = "cumulative")

## End(Not run)
```

plot.difNLR

ICC and test statistics plots for difNLR object

Description

Two types of plots are available. The first one is obtained by setting `plot.type = "cc"` (default). The characteristic curve for item specified in `item` option is plotted. For default option "all" of item, characteristic curves of all converged items are plotted. The drawn curves represent best model.

The second plot is obtained by setting `plot.type = "stat"`. The test statistics (either LR-test, or F-test, depends on argument `test`) are displayed on the Y axis, for each converged item. The detection threshold is displayed by a horizontal line and items detected as DIF are printed with the red color. Only parameters `size` and `title` are used.

Usage

```
## S3 method for class 'difNLR'
plot(x, plot.type = "cc", item = "all",
     col = c("dodgerblue2", "goldenrod2"), shape = 21, size = 0.8,
     linetype = c(2, 1), title, group.names, ...)
```

Arguments

<code>x</code>	an object of "difNLR" class
<code>plot.type</code>	character: type of plot to be plotted (either "cc" for characteristic curve (default), or "stat" for test statistics). See Details .
<code>item</code>	either character ("all"), or numeric vector, or single number corresponding to column indicators. See Details .
<code>col</code>	character: single value, or vector of two values representing colors for plot.
<code>shape</code>	integer: shape parameter for plot.
<code>size</code>	numeric: single number, or vector of two numbers representing line width in plot.
<code>linetype</code>	numeric: single number, or vector of two numbers representing line type in plot for reference and focal group.
<code>title</code>	string: title of plot.
<code>group.names</code>	character: names of reference and focal group.
<code>...</code>	other generic parameters for <code>plot()</code> function.

Author(s)

Adela Hladka (nee Drabinova)
 Institute of Computer Science, The Czech Academy of Sciences
 Faculty of Mathematics and Physics, Charles University
 hladka@cs.cas.cz

Patricia Martinkova
 Institute of Computer Science, The Czech Academy of Sciences
 martinkova@cs.cas.cz

Karel Zvara
 Faculty of Mathematics and Physics, Charles University

References

- Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517.
- Swaminathan, H. & Rogers, H. J. (1990). Detecting Differential Item Functioning Using Logistic Regression Procedures. *Journal of Educational Measurement*, 27, 361-370.

See Also

[difNLR](#)

Examples

```
## Not run:
# loading data based on GMAT
data(GMAT)

Data <- GMAT[, 1:20]
group <- GMAT[, "group"]

# testing both DIF effects using likelihood-ratio test and
# 3PL model with fixed guessing for groups
(x <- difNLR(Data, group, focal.name = 1, model = "3PLcg"))

# graphical devices
plot(x)
plot(x, item = x$DIFitems)
plot(x, plot.type = "stat")

## End(Not run)
```

predict.difNLR	<i>Predicted values for difNLR object</i>
----------------	---

Description

Predicted values based on "difNLR" object.

Usage

```
## S3 method for class 'difNLR'
predict(object, item = "all", match, group, ...)
```

Arguments

object	an object of "difNLR" class
item	either character ("all"), or numeric vector, or single number corresponding to column indicators.
match	numeric: specifies matching criterion for new observation. See Details .
group	numeric: specifies group membership for new observation. See Details .
...	other generic parameters for predict() function.

Details

Arguments `match` and `group` represent matching criterion and group membership of new observations and need to have the same length.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

Karel Zvara
Faculty of Mathematics and Physics, Charles University

References

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517.

Swaminathan, H. & Rogers, H. J. (1990). Detecting Differential Item Functioning Using Logistic Regression Procedures. *Journal of Educational Measurement*, 27, 361-370.

See Also

[difNLR](#)

Examples

```
## Not run:  
# loading data based on GMAT  
data(GMAT)  
  
Data <- GMAT[, 1:20]  
group <- GMAT[, "group"]  
  
# Testing both DIF effects using likelihood-ratio test and  
# 3PL model with fixed guessing for groups  
(x <- difNLR(Data, group, focal.name = 1, model = "3PLcg"))  
  
# Predicted values  
predict(x)  
predict(x, item = 1)  
  
# Predicted values for new observations  
predict(x, item = 1, match = 0, group = 1)  
predict(x, item = 1, match = 0, group = 0)  
  
## End(Not run)
```

startNLR	<i>Calculates starting values for Non-Linear Regression DIF models.</i>
----------	---

Description

Calculates starting values for difNLR function based on linear approximation.

Usage

```
startNLR(Data, group, model, match = "zscore", parameterization = "alternative",
simplify = F)
```

Arguments

Data	numeric: a matrix or data.frame of binary data.
group	numeric: binary vector of group membership. "0" for reference group, "1" for focal group.
model	character: generalized logistic regression model to be fitted. See Details .
match	vector of matching criterion. Its length need to be the same as number of observations in Data.
parameterization	character: parameterization of regression coefficients. Possible options are "classic" (IRT parameterization), "alternative" (default) and "logistic" (logistic regression). See Details .
simplify	logical: whether initial values should be simplified into the table. This is only applicable when parameterization is the same for all items.

Details

The model argument offers several predefined models. The options are as follows: Rasch for 1PL model with discrimination parameter fixed on value 1 for both groups, 1PL for 1PL model with discrimination parameter fixed for both groups, 2PL for logistic regression model, 3PLcg for 3PL model with fixed guessing for both groups, 3PLdg for 3PL model with fixed inattention for both groups, 3PLc (alternatively also 3PL) for 3PL regression model with guessing parameter, 3PLd for 3PL model with inattention parameter, 4PLcgd for 4PL model with fixed guessing and inattention parameter for both groups, 4PLcgd (alternatively also 4PLd) for 4PL model with fixed guessing for both groups, 4PLcdg (alternatively also 4PLc) for 4PL model with fixed inattention for both groups, or 4PL for 4PL model.

Three possible parameterization can be specified in "parameterization" argument: "classic" returns IRT parameters of reference group and differences in these parameters between reference and focal group. "alternative" returns IRT parameters of reference group, the differences in parameters a and b between two groups and parameters c and d for focal group. "logistic" returns parameters in logistic regression parameterization.

Value

A list containing elements representing items. Each element is a named numeric vector of length 8 with initial values for difNLR model.

Author(s)

Adela Hladka (nee Drabinova)
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
hladka@cs.cas.cz

Patricia Martinkova
Institute of Computer Science, The Czech Academy of Sciences
martinkova@cs.cas.cz

References

Drabinova, A. & Martinkova P. (2017). Detection of Differential Item Functioning with NonLinear Regression: Non-IRT Approach Accounting for Guessing. *Journal of Educational Measurement*, 54(4), 498-517, <https://doi.org/10.1111/jedm.12158>.

See Also

[difNLR](#)

Examples

```
## Not run:
# loading data based on GMAT
data(GMAT)

Data <- GMAT[, 1:20]
group <- GMAT[, "group"]

# starting values for 3PL model
startNLR(Data, group, model = "3PL")

# starting values for 3PL model
# simplified into single table
startNLR(Data, group, model = "3PL", simplify = T)

# starting values for 3PL model
# with score as matching criterion
startNLR(Data, group, model = "3PL", match = "score")

# starting values for model specified for each item
startNLR(Data, group,
  model = c(rep("1PL", 5), rep("2PL", 5),
    rep("3PL", 5), rep("4PL", 5)))
```

startNLR

51

End(Not run)

Index

- *Topic **DDF**
 - ddfMLR, 5
 - ddfORD, 9
 - MLR, 33
 - ORD, 43
- *Topic **DIF**
 - difNLR, 14
 - estimNLR, 20
 - NLR, 39
- *Topic **datasets**
 - GMAT, 27
 - GMAT2, 28
 - GMAT2key, 29
 - GMAT2test, 30
 - GMATkey, 31
 - GMATtest, 32
 - MSATB, 36
 - MSATBkey, 37
 - MSATBtest, 38
- _PACKAGE (difNLR-package), 2
- AIC.ddfMLR (ddfMLR), 5
- AIC.ddfORD (ddfORD), 9
- AIC.difNLR (difNLR), 14
- BIC.ddfMLR (ddfMLR), 5
- BIC.ddfORD (ddfORD), 9
- BIC.difNLR (difNLR), 14
- checkInterval, 4
- coef.ddfMLR (ddfMLR), 5
- coef.ddfORD (ddfORD), 9
- coef.difNLR (difNLR), 14
- ddfMLR, 5, 25, 26
- ddfORD, 9, 25, 26
- difNLR, 14, 24–26, 46, 48, 50
- difNLR-package, 2
- estimNLR, 20
- fitted.difNLR (difNLR), 14
- formulaNLR, 22
- genNLR, 24
- GMAT, 27, 32, 33
- GMAT2, 28, 30, 31
- GMAT2key, 29, 29, 30, 31
- GMAT2test, 29, 30, 30
- GMATkey, 28, 31, 33
- GMATtest, 28, 31, 32, 32
- logLik.ddfMLR (ddfMLR), 5
- logLik.ddfORD (ddfORD), 9
- logLik.difNLR (difNLR), 14
- MLR, 33
- MSATB, 36, 37, 38
- MSATBkey, 36, 37, 38
- MSATBtest, 36, 37, 38
- multinom, 8
- NLR, 39
- nls, 19
- ORD, 43
- p.adjust, 6, 8, 11, 13, 17, 19, 34, 35, 42, 44, 45
- plot.ddfMLR (ddfMLR), 5
- plot.ddfORD (ddfORD), 9
- plot.difNLR, 17, 19, 45
- predict.difNLR, 17, 47
- print.ddfMLR (ddfMLR), 5
- print.ddfORD (ddfORD), 9
- print.difNLR (difNLR), 14
- resid.difNLR (difNLR), 14
- residuals.difNLR (difNLR), 14
- startNLR, 17, 19, 40, 49
- vglm, 13, 45