Package 'ShinyItemAnalysis'

February 25, 2019

Type Package

Title Test and Item Analysis via Shiny

Version 1.3.0

Date 2019-02-25

Author Patricia Martinkova [aut, cre],
Adela Drabinova [aut],
Ondrej Leder [aut],
Jakub Houdek [aut],
Lubomir Stepanek [ctb],
Tomas Jurica [ctb],
Jana Vorlickova [ctb]

Maintainer Patricia Martinkova <martinkova@cs.cas.cz>

Depends R (>= 3.1)

Imports corrplot, cowplot, CTT, data.table, deltaPlotR, DT, difNLR (>= 1.2.2), difR (>= 5.0), ggdendro, ggplot2 (>= 2.2.1), gridExtra, knitr, latticeExtra, ltm, mirt (>= 1.24), moments, msm, nnet, plotly, psych, psychometric, reshape2, rmarkdown, shiny (>= 1.0.3), shinyBS, shinydashboard, shinyjs (>= 0.9), stringr, xtable

Description Interactive shiny application for analysis of educational tests and their items.

License GPL-3

LazyData TRUE

RoxygenNote 6.1.0

BugReports https://github.com/patriciamar/ShinyItemAnalysis/issues

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2019-02-25 16:40:02 UTC

R topics documented:

`dataMedical` *Dichotomous Dataset of Admission Test to Medical School*

Description

The `dataMedical` dataset consists of the responses of 2,392 subjects (750 males, 1,633 females and 9 subjects without gender specification) to admission test to a medical school. It contains 100 items. A correct answer is coded as 1 and incorrect answer as 0. Missing answers were evaluated as incorrect, i.e. 0.

Usage

```
data(dataMedical)
```

Format

A `dataMedical` is a `data.frame` consisting of 2,392 observations on the following 102 variables. The first 100 columns represent dichotomously scored items of the test. The 101st column is vector of gender membership; values 0 and 1 refer to males and females. The 102nd columns in criterion variable; value 1 means that student studies standardly, 0 otherwise (e.g. leaving or interrupting studies).

Author(s)

Cestmir Stuka
First Faculty of Medicine, Charles University in Prague

Martin Vejrazka
First Faculty of Medicine, Charles University in Prague

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

Jakub Houdek
Institute of Computer Science, The Czech Academy of Sciences
The Faculty of Informatics and Statistics, University of Economics, Prague
houdek.james@gmail.com

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

References

Stuka, C. Vejrazka, M., Martinkova, P. Komenda, M. & Stepanek, L. (2016). The Use of Test and Item Analisis for Improvement of Tests. Workshop held at conference MEFANET, 2016, Brno, Czech Republic.

See Also

dataMedicaltest, dataMedicalkey, dataMedicalgraded

---

dataMedicalgraded *Graded Dataset of Admission Test to Medical School*

Description

The dataMedicalgraded dataset consists of the responses of 2,392 subjects (750 males, 1,633 females and 9 subjects without gender specification) to multiple-choice admission test to a medical school. It contains 100 items. Each item is graded with 0 to 4 points. Maximum of 4 points were set if all correct answers and none of incorrect answers were selected.

Usage

```
data(dataMedicalgraded)
```

Format

A `dataMedicalgraded` is a `data.frame` consisting of 2,392 observations on the following 102 variables. The first 100 columns represent graded answers of subject to items of the test. The 101st column is vector of gender membership; values 0 and 1 refer to males and females. The 102nd columns in criterion variable; value 1 means that student study standardly, 0 otherwise (e.g. leaving or interrupting studies).

Author(s)

Cestmir Stuka
First Faculty of Medicine, Charles University in Prague

Martin Vejrazka
First Faculty of Medicine, Charles University in Prague

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

Jakub Houdek
Institute of Computer Science, The Czech Academy of Sciences
The Faculty of Informatics and Statistics, University of Economics, Prague
houdek.james@gmail.com

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

References

Stuka, C. Vejrazka, M., Martinkova, P. Komenda, M. & Stepanek, L. (2016). The Use of Test and Item Analisis for Improvement of Tests. Workshop held at conference MEFANET, 2016, Brno, Czech Republic.

See Also

`dataMedical`, `dataMedicaltest`, `dataMedicalkey`

dataMedicalkey *Key of Correct Answers for* dataMedicaltest *Dataset*

Description

The dataMedicalkey is a vector of factors representing correct answers of dataMedicaltest data set.

Usage

```
data(dataMedicalkey)
```

Format

A data.frame with 100 values representing correct answers to items of dataMedicaltest dataset. For more details see dataMedicaltest.

Author(s)

Cestmir Stuka
First Faculty of Medicine, Charles University in Prague

Martin Vejrazka
First Faculty of Medicine, Charles University in Prague

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

Jakub Houdek
Institute of Computer Science, The Czech Academy of Sciences
The Faculty of Informatics and Statistics, University of Economics, Prague
houdek.james@gmail.com

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

References

Stuka, C. Vejrazka, M., Martinkova, P. Komenda, M. & Stepanek, L. (2016). The Use of Test and Item Analisis for Improvement of Tests. Workshop held at conference MEFANET, 2016, Brno, Czech Republic.

See Also

dataMedical, dataMedicaltest, dataMedicalgraded

---

dataMedicaltest *Dataset of Admission Test to Medical School*

Description

The dataMedicaltest dataset consists of the responses of 2,392 subjects (750 males, 1,633 females and 9 subjects without gender specification) to multiple-choice admission test to a medical school. It contains 100 items, possible answers were A, B, C, D, while any combination of these can be correct.

Usage

```
data(dataMedicaltest)
```

Format

dataMedicaltest is a data.frame consisting of 2,392 observations on the following 102 variables. The first 100 columns represent answers of subject to items of the test. The 101st column is vector of gender membership; values 0 and 1 refer to males and females. The 102nd columns in criterion variable; value 1 means that student studies standardly, 0 otherwise (e.g. leaving or interrupting studies).

Author(s)

Cestmir Stuka
First Faculty of Medicine, Charles University in Prague

Martin Vejrazka
First Faculty of Medicine, Charles University in Prague

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

Jakub Houdek
Institute of Computer Science, The Czech Academy of Sciences
The Faculty of Informatics and Statistics, University of Economics, Prague
houdek.james@gmail.com

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

References

Stuka, C. Vejrazka, M., Martinkova, P. Komenda, M. & Stepanek, L. (2016). The Use of Test and Item Analisis for Improvement of Tests. Workshop held at conference MEFANET, 2016, Brno, Czech Republic.

See Also

dataMedical, dataMedicalkey, dataMedicalgraded

`DDplot` *Graphical representation of difficulty and (generalized) discrimination in item analysis*

Description

Plots difficulty and (generalized) discrimination for items ordered by difficulty.

Usage

```
DDplot(data, item.names, k = 3, l = 1, u = 3,
discrim = "ULI", maxscore, minscore, bin = FALSE, cutscore)
```

Arguments

`data`	numeric: binary or ordinal data matrix or data frame. See **Details**.
`item.names`	character: the names of items.
`k`	numeric: number of groups to which may be data.frame x divided by the total score. Default value is 3. See **Details**.
`l`	numeric: lower group. Default value is 1. See **Details**.
`u`	numeric: upper group. Default value is 3. See **Details**.
`discrim`	character: type of discrimination index to be calculated. Deafult value is "ULI". See **Details**.
`maxscore`	vector or numeric: maximal scores of items. If numeric, the same maximal score is used for all items. If missing, vector of achieved maximal scores is calculated and used in calculations.
`minscore`	vector or numeric: minimal scores of items. If numeric, the same minimal score is used for all items. If missing, vector of achieved minimal scores is calculated and used in calculations.
`bin`	logical: should the ordinal data be binarized. Deafult value is FALSE. See **Details**.
`cutscore`	vector or numeric: cutscore used to binarize the data.set. If numeric, the same cutscore is used for all items. If missing, vector of maximal scores is used in calculations.

Details

The `data` is a matrix or data frame whose rows represents examinee answers (`1` correct, `0` incorrect, or ordinal item scores) and columns correspond to the items.

The `item.names` argument stands for names of items. If not specified, the names of dataset columns are used. Difficulty and discrimination indices are plotted for each item, items are ordered by their difficulty.

Discrimination is calculated using method specified in `discrim`. Default option "ULI" calculates difference in ratio of correct answers in upper and lower third of students. "RIT" index caluclates correlation between item score and test total score. "RIR" index caclulates correlation between item score and total score for the rest of the items. With option "none", only difficulty is displayed.

"ULI" index can be generalized using arguments `k`, `l` and `u`. Generalized ULI discrimination is then computed as follows: The function takes data on individuals, computes their total test score and then divides individuals into `k` groups. The lower and upper group are determined by `l` and `u` parameters, i.e. l-th and u-th group where the ordering is defined by increasing total score.

For ordinal data, difficulty is defined as relative score (achieved - minimal)/(maximal - minimal). Minimal score can be specified by `minscore`, maximal score can be specified by `maxscore`.

Binarization of data is allowed in `bin`, for this purpose `cutscore` is used.

Note

Generalized discrimination is calculated by `gDiscrim` function.

Author(s)

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

Lubos Stepanek
First Faculty of Medicine, Charles University
lubomir.stepanek@lf1.cuni.cz

Jana Vorlickova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University

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

References

Martinkova, P., Stepanek, L., Drabinova, A., Houdek, J., Vejrazka, M., & Stuka, C. (2017). Semi-real-time analyses of item characteristics for medical school admission tests. In: Proceedings of the 2017 Federated Conference on Computer Science and Information Systems.

See Also

gDiscrim, discrim

Examples

```
## Not run:
# loading 100-item medical admission test data sets
data(dataMedical, dataMedicalgraded)
# binary data set
dataBin <- dataMedical[, 1:100]
# ordinal data set
dataOrd <- dataMedicalgraded[, 1:100]

# DDplot of binary data set
DDplot(dataBin)
# compared to DDplot using ordinal data set and 'bin = TRUE'
DDplot(dataOrd, bin = TRUE)
# compared to binarized data set using bin = TRUE and cutscore equal to 3
DDplot(dataOrd, bin = TRUE, cutscore = 3)

# DDplot of binary data using generalized ULI
# discrimination based on 5 groups, comparing 4th and 5th
DDplot(dataBin, k = 5, l = 4, u = 5)

# DDplot of ordinal data set using ULI
DDplot(dataOrd)
# DDplot of ordinal data set using generalized ULI
# discrimination based on 5 groups, comparing 4th and 5th
DDplot(dataOrd, k = 5, l = 4, u = 5)
# DDplot of ordinal data set using RIT
DDplot(dataOrd, discrim = "RIT")
# DDplot of ordinal data set using RIR
DDplot(dataOrd, discrim = "RIR")
# DDplot of ordinal data set disaplaying only difficulty
DDplot(dataBin, discrim = "none")

## End(Not run)
```

DistractorAnalysis *Function for item distractor analysis*

Description

Performs distractor analysis for each item and optional number of groups.

Usage

```
DistractorAnalysis(data, key, p.table = FALSE, num.groups = 3, matching = NULL)
```

Arguments

data	character: data matrix or data frame. See **Details**.
key	character: answer key for the items.
p.table	logical: should the function return the proportions. If FALSE (default) the counts are returned.
num.groups	numeric: number of groups to that should be respondents splitted.
matching	numeric: numeric vector. If not provided, total score is calculated and distractor analysis is performed based on it.

Details

This function is adapted version of distractor.analysis function from CTT package. The scores are calculatede using the item data and key. The respondents are then splitted into the num.groups-quantiles and the number (or proportion) of respondents in each quantile is reported with respect to their answers.

The data is a matrix or data frame whose rows represents unscored item response from a multiple-choice test and columns correspond to the items.

The key must be a vector of the same length as ncol(data).

Author(s)

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

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

See Also

distractor.analysis

Examples

```
## Not run:
# loading 100-item medical admission test data
data(dataMedicaltest, dataMedicalkey)
data <- dataMedicaltest[, 1:100]
key <- unlist(dataMedicalkey)

# distractor analysis for dataMedicaltest data set
DistractorAnalysis(data, key)

# distractor analysis for dataMedicaltest data set with proportions
DistractorAnalysis(data, key, p.table = T)
```

```
# distractor analysis for dataMedicaltest data set for 6 groups
DistractorAnalysis(data, key, num.group = 6)

## End(Not run)
```

---

gDiscrim *Generalized Item Discrimination*

Description

gDiscrim function computes various generalizations of discrimination index ULI. It enumerates the ablitity of item to distinguish between individuals from upper (U) vs. lower (L) ability groups, i.e. between respondents with high vs. low overall score on the test. Number of groups, as well as upper and lower groups can be specified by user. Maximal and minimal score in ordinal data sets can be specified by user.

Usage

```
gDiscrim(x, k = 3, l = 1, u = 3, maxscore, minscore)
```

Arguments

x	matrix or data.frame of items to be examined. Rows represent persons, columns reperesent items.
k	numeric: number of groups to which may be data.frame x divided by the total score. Default value is 3. See **Details**.
l	numeric: lower group. Default value is 1. See **Details**.
u	numeric: upper group. Default value is 3. See **Details**.
maxscore	numeric: maximal score in ordinal items. If missing, vector of obtained maximal scores is imputed. See **Details**.
minscore	numeric: minimal score in ordinal items. If missing, vector of obtained minimal scores is imputed. See **Details**.

Details

The function computes total test scores for all respondents and then divides the respondents into k groups. The lower and upper groups are determined by l and u parameters, i.e. l-th and u-th group where the ordering is defined by increasing total score.

In ordinal items, difficulty is calculated as difference of average score divided by range (maximal possible score maxscore minus minimal possible score minscore for given item).

Discrimination is calculated as difference in difficulty between upper and lower group.

Note

gDiscrim is used by DDplot function.

Author(s)

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

Lubos Stepanek
First Faculty of Medicine, Charles University
lubomir.stepanek@lf1.cuni.cz

Jana Vorlickova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University

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

References

Martinkova, P., Stepanek, L., Drabinova, A., Houdek, J., Vejrazka, M., & Stuka, C. (2017). Semi-real-time analyses of item characteristics for medical school admission tests. In: Proceedings of the 2017 Federated Conference on Computer Science and Information Systems. http://dx.doi.org/10.15439/2017F380

See Also

DDplot

Examples

```
## Not run:
# loading 100-item medical admission test data sets
data(dataMedical, dataMedicalgraded)
# binary data set
dataBin <- dataMedical[, 1:100]
# ordinal data set
dataOrd <- dataMedicalgraded[, 1:100]

# ULI for first 5 items for binary data set
# compare to psychometric::discrim(x)
gDiscrim(dataBin)[1:5]
# generalized ULI using 5 groups, compare 4th and 5th for binary data set
gDiscrim(dataBin, k = 5, l = 4, u = 5)[1:5]
```

```
# ULI for first 5 items for ordinal data set
gDiscrim(dataOrd)[1:5]
# generalized ULI using 5 groups, compare 4th and 5th for binary data set
gDiscrim(dataOrd, k = 5, l = 4, u = 5)[1:5]
# maximum (4) and minimum (0) score are same for all items
gDiscrim(dataOrd,  k = 5, l = 4, u = 5, maxscore = 4, minscore = 0)[1:5]

## End(Not run)
```

---

ggWrightMap *Wright Map using ggplot*

Description

This function allows to generate Wright Map (also called item-person map) using `ggplot` function from package `ggplot2` and `plot_grid` function from `cowplot`. Wright Map is used to display histogram of factor scores and the item difficulty parameters estimated by the Rasch IRT model.

Usage

```
ggWrightMap(theta, b, binwidth = 0.5, color = "blue", size = 15)
```

Arguments

`theta`	numeric: vector of ability estimates.
`b`	numeric: vector of difficulty estimates.
`binwidth`	numeric: the width of the bins of histogram.
`color`	character: color of histogram.
`size`	text size in pts.

Author(s)

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

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

References

Wright, B. D., & Stone, M. H. (1979). Best test design.

See Also

wrightMap

Examples

```
## Not run:
library(mirt)

# loading 100-item medical admission test data sets
data(dataMedical)
# binary data set
dataBin <- dataMedical[, 1:100]

# fit Rasch model with mirt package
fit <- mirt(dataBin, model = 1, itemtype = "Rasch")
# factor scores
theta <- as.vector(fscores(fit))
# difficulty estimates
b <- coef(fit, simplify = T)$items[, "d"]

ggWrightMap(theta, b)

## End(Not run)
```

HCI *Homeostasis Concept Inventory Dichotomous Dataset*

Description

(HCI) dataset consists of the dichotomously scored responses of 651 students (405 males, 246 females) to Homeostasis Concept Inventory multiple-choice test. It containts 20 items, vector of gender membership and identificator whether students plan to major.

Usage

```
data(HCI)
```

Format

HCI is a data.frame consisting of 651 observations on the 22 variables. First 20 variables represent dichotomously scored responses to multiple-choice items (1 means correct, 0 is incorrect). 21st column is a vector of gender membership; values 0 and 1 refer to males and females. 22nd column is a identificator whether students planning to major in the life sciences.

Author(s)

Jenny L. McFarland
Biology Department, Edmonds Community College

References

McFarland, J. L., Price, R. M., Wenderoth, M. P., Martinkova, P., Cliff, W., Michael, J., ... & Wright, A. (2017). Development and validation of the homeostasis concept inventory. CBE-Life Sciences Education, 16(2), ar35.

See Also

HCItest, HCIkey

HCIkey *Key of Correct Answers for Homeostasis Concept Inventory Dataset*

Description

The HCIkey is a vector of factors representing correct answers of HCItest dataset.

Usage

```
data(HCIkey)
```

Format

A data.frame with 20 values representing correct answers to items of HCItest dataset. For more details see HCItest.

Author(s)

Jenny L. McFarland
Biology Department, Edmonds Community College

References

McFarland, J. L., Price, R. M., Wenderoth, M. P., Martinkova, P., Cliff, W., Michael, J., ... & Wright, A. (2017). Development and validation of the homeostasis concept inventory. CBE-Life Sciences Education, 16(2), ar35.

See Also

HCI, HCItest

HCItest *Homeostasis Concept Inventory Dataset*

Description

(HCItest) dataset consists of the responses of 651 students (405 males, 246 females) to Homeostasis Concept Inventory multiple-choice test. It containts 20 items, vector of gender membership and identificator whether students plan to major.

Usage

```
data(HCItest)
```

Format

HCItest is a data.frame consisting of 651 observations on the 22 variables. First 20 variables represent responses to multiple-choice items. 21st column is a vector of gender membership; values 0 and 1 refer to males and females. 22nd column is a identificator whether students planning to major in the life sciences.

Author(s)

Jenny L. McFarland
Biology Department, Edmonds Community College

References

McFarland, J. L., Price, R. M., Wenderoth, M. P., Martinkova, P., Cliff, W., Michael, J., ... & Wright, A. (2017). Development and validation of the homeostasis concept inventory. CBE-Life Sciences Education, 16(2), ar35.

See Also

HCI, HCIkey

ItemAnalysis *Item Analysis*

Description

ItemAnalysis function computes various traditional item analysis indices including difficulty, discrimination and item validity. For ordinal items the difficulty and discrimination indices take into account minimal item score as well as range.

Usage

```
ItemAnalysis(data, y = NULL, k = 3, l = 1, u = 3,
maxscore, minscore, cutscore, add.bin = FALSE)
```

Arguments

data	matrix or data.frame of items to be examined. Rows represent respondents, columns reperesent items.
y	vector of criterion values.
k	numeric: number of groups to which may be data.frame x divided by the total score. Default value is 3. See **Details**.
l	numeric: lower group. Default value is 1. See **Details**.
u	numeric: upper group. Default value is 3. See **Details**.
maxscore	numeric or vector: maximal score in ordinal items. If missing, vector of obtained maximal scores is imputed. See **Details**.
minscore	numeric or vector: minimal score in ordinal items. If missing, vector of obtained minimal scores is imputed. See **Details**.
cutscore	numeric or vector: cut score used for binarization of ordinal data. If missing, vector of maximal scores is imputed. See **Details**.
add.bin	logical: If TRUE, indices are printed also for binarized data. See **Details**.

Details

For ordinal items the difficulty and discrimination indices take into account minimal item score as well as range.

For calculation of discimination ULI index, it is possible to specify the number of groups k, and which two groups l and u are to be compared.

In ordinal items, difficulty is calculated as difference of average score divided by range (maximal possible score maxscore minus minimal possible score minscore).

If add.bin is set to TRUE, item analysis of binarized data is included in the output table. In such a case, cutscore is used for binarization. When binarizing the data, values greater or equal to cut-score are set to 1, other values are set to 0.

Value

ItemAnalysis function computes various traditional item analysis indices. Output is a data.frame with following columns:

Difficulty	item difficulty based on ratio of correct answers
Scaled score	
Sample SD	standard deviation of the item
Correct answers	proportion of correct answers
Min score	minimal score specified in minscore; if not provided, observed minimal score

`Max score`	maximal score specified in `maxscore`; if not provided, observed maximal score
`Obtained min`	observed minimal score
`Obtained max`	observed maximal score
`Cut score`	cut-score specified in `cutscore`
`ULI`	generalized ULI
`ULI default`	dscrimination with ULI
`RIT`	correlation between item score and overall test score
`RIR`	correlation between item score and overall test score
`Item criterion`	correlation of item score with criterion
`Item reliability`	item reliability index
`Item reliability woi`	item reliability index (scored without item)
`Item validity`	item validity index
`Item criterion`	correlation between item and criterion `y`
`Alpha drop`	Cronbach's alpha without given item

With `add.bin == TRUE`, indices based on binarized data set are also provided and marked with `bin`.

Author(s)

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

Jana Vorlickova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

References

Martinkova, P., Stepanek, L., Drabinova, A., Houdek, J., Vejrazka, M., & Stuka, C. (2017). Semi-real-time analyses of item characteristics for medical school admission tests. In: Proceedings of the 2017 Federated Conference on Computer Science and Information Systems. http://dx.doi.org/10.15439/2017F380

Allen, M. J. & Yen, W. M. (1979). Introduction to measurement theory. Monterey, CA: Brooks/Cole.

See Also

`DDplot`, `gDiscrim`

Examples

```
## Not run:
# loading 100-item medical admission test data sets
data(dataMedical, dataMedicalgraded)
# binary data set
dataBin <- dataMedical[, 1:100]
# ordinal data set
dataOrd <- dataMedicalgraded[, 1:100]
# study success is the same for both data sets
StudySuccess <- dataMedical[, 102]

# item analysis for binary data
head(ItemAnalysis(dataBin))
# item analysis for binary data using also study success
head(ItemAnalysis(dataBin, y = StudySuccess))

# item analysis for binary data
head(ItemAnalysis(dataOrd))
# item analysis for binary data using also study success
head(ItemAnalysis(dataOrd, y = StudySuccess))
# including also item analysis for binarized data
head(ItemAnalysis(dataOrd, y = StudySuccess, k = 5, l = 4, u = 5,
maxscore = 4, minscore = 0, cutscore = 4, add.bin = TRUE) )

## End(Not run)
```

---

plotDIFirt *Function for characteristic curve of DIF IRT model*

Description

Plots characteristic curve of IRT model.

Usage

```
plotDIFirt(parameters, test = "Lord", item = "all", item.name, same.scale = F)
```

Arguments

parameters	numeric: data matrix or data frame. See **Details**.
test	character: type of statistic to be shown. See **Details**.
item	either character ("all"), or numeric vector, or single number corresponding to column indicators. See **Details**.
item.name	character: the name of item.
same.scale	logical: are the item parameters on the same scale? (default is "FALSE"). See **Details**.

Details

This function plots characteristic curve of DIF IRT model.

The `parameters` matrix has a number of rows equal to twice the number of items in the data set. The first J rows refer to the item parameter estimates in the reference group, while the last J ones correspond to the same items in the focal group. The number of columns depends on the selected IRT model: 2 for the 1PL model, 5 for the 2PL model, 6 for the constrained 3PL model and 9 for the unconstrained 3PL model. The columns of irtParam have to follow the same structure as the output of `itemParEst`, `difLord` or `difRaju` command from `difR` package.

Two possible type of `test` statistics can be visualized - `"Lord"` gives only characteristic curves, `"Raju"` also highlights area between these curves.

For default option `"all"`, all characteristic curves are plotted.

Author(s)

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

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

See Also

`itemParEst`

`difLord`

`difRaju`

Examples

```
## Not run:
# loading libraries
library(difR)
library(ltm)

 # loading data based on GMAT2
data(GMAT2, package = "difNLR")

# Estimation of 2PL IRT model and Lord's statistic
# by difR package
fitLord <- difLord(GMAT2, group = 21, focal.name = 1, model = "2PL")
# plot of item 1 and Lord's statistic
plotDIFirt(fitLord$itemParInit, item = 1)

# Estimation of 2PL IRT model and Raju's statistic
# by difR package
```

```
fitRaju <- difRaju(GMAT2, group = 21, focal.name = 1, model = "2PL")
# plot of item 1 and Lord's statistic
plotDIFirt(fitRaju$itemParInit, test = "Raju", item = 1)

## End(Not run)
```

`plotDIFLogistic` *Function for characteristic curve of 2PL logistic DIF model*

Description

Plots characteristic curve of 2PL logistic DIF model

Usage

```
plotDIFLogistic(data, group, type = "both", item, item.name,
IRT = F, p.adjust.method = "none", purify = F, match = "score")
```

Arguments

`data`	numeric: the data matrix. See Details.
`group`	numeric: the vector of group membership. See Details.
`type`	character: a character string specifying which DIF effects must be tested. Possible values are "both" (default), "udif" and "nudif". See Details.
`item`	numeric: number of item to be plotted
`item.name`	character: the name of item.
`IRT`	logical: if IRT parameterization (`TRUE`, default) or classic logistic parameterization (`FALSE`) may be applied.
`p.adjust.method`	character: the acronym of the method for p-value adjustment for multiple comparisons. See Details.
`purify`	logical: if item purification may be applied.
`match`	specifies the type of matching criterion. Can be either `"score"` (default) to compute the test score, or any continuous or discrete variable with the same length as the number of rows of `Data`.

Details

This function plots characteristic curve of 2PL logistic DIF model.

Author(s)

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

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

Examples

```
## Not run:
# loading libraries
library(difNLR, difR)

 # loading data based on GMAT
data(GMAT, package = "difNLR")
data  <- GMAT[, 1:20]
group <- GMAT[, 21]

# Characteristic curve by logistic regression model
plotDIFLogistic(data, group, item = 1)

# Characteristic curve by logistic regression model using scaled score
plotDIFLogistic(data, group, item = 1, IRT = T)

## End(Not run)
```

---

plotDistractorAnalysis

Function for graphical representation of item distractor analysis

---

Description

Plots graphical representation of item distractor analysis with proportions and optional number of groups.

Usage

```
plotDistractorAnalysis(data, key, num.groups = 3, item = 1, item.name,
multiple.answers = TRUE, matching = NULL)
```

Arguments

`data`	character: data matrix or data frame. See **Details**.
`key`	character: answer key for the items.
`num.groups`	numeric: number of groups to that should be respondents splitted.
`item`	numeric: the number of item to be plotted.
`item.name`	character: the name of item.
`multiple.answers`	logical: should be all combinations plotted (default) or should be answers splitted into distractors. See **Details**.
`matching`	numeric: numeric vector. If not provided, total score is calculated and distractor analysis is performed based on it.

Details

This function is graphical representation of `DistractorAnalysis` function. The scores are calculatede using the item data and key. The respondents are then splitted into the `num.groups`-quantiles and the proportion of respondents in each quantile is reported with respect to their answers, using all reported combinations (default) or distractors. These proportions are plotted.

The `data` is a matrix or data frame whose rows represents unscored item response from a multiple-choice test and columns correspond to the items.

The `key` must be a vector of the same length as `ncol(data)`.

If `multiple.answers = TRUE` (default) all reported combinations of answers are plotted. If `multiple.answers = FALSE` all combinations are splitted into distractors and only these are then plotted with correct combination.

Author(s)

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

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

#' @seealso `DistractorAnalysis` #' @seealso `distractor.analysis`

Examples

```
## Not run:
# loading 100-item medical admission test data
data(dataMedicaltest, dataMedicalkey)
dataBin <- dataMedical[, 1:100]
data <- dataMedicaltest[, 1:100]
key <- unlist(dataMedicalkey)
```

```
# Difficulty/Discriminaton plot for medical admission test
DDplot(dataBin)
# item 48 is very hard, thus does not discriminate well
# item 57 discriminates well
# item 32 does not discriminate well

plotDistractorAnalysis(data, key, item = 48, multiple.answers = F)
# correct answer B does not function well
plotDistractorAnalysis(data, key, item = 57, multiple.answers = F)
# all options function well, thus the whole item discriminates well
plotDistractorAnalysis(data, key, item = 32, multiple.answers = F)
# functions well, thus the whole item discriminates well

# distractor analysis plot for item 48, 57 and 32, all combinations
plotDistractorAnalysis(data, key, item = 48)
plotDistractorAnalysis(data, key, item = 57)
plotDistractorAnalysis(data, key, item = 32)

# distractor analysis plot for item 57, all combinations and 6 groups
plotDistractorAnalysis(data, key, num.group = 6, item = 57)

## End(Not run)
```

`startShinyItemAnalysis` *This function will start ShinyItemAnalysis application.*

Description

An interactive shiny application for running test and item analysis.

Usage

```
startShinyItemAnalysis()
```

Author(s)

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

Adela Drabinova
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
drabinova@cs.cas.cz

Ondrej Leder
Institute of Computer Science, The Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University
ondraleder@gmail.com

Jakub Houdek
Institute of Computer Science, The Czech Academy of Sciences
The Faculty of Informatics and Statistics, University of Economics, Prague
houdek.james@gmail.com

Examples

```
## Not run:
rm(list = ls())
startShinyItemAnalysis()

## End(Not run)
```

`theme_app` *Complete theme for* `ShinyItemAnalysis` *graphics*

Description

This complete theme is based on `theme_bw` and it was modified for purposes of `ShinyItemAnalysis`.

Usage

```
theme_app(base_size = 15, base_family = "")
```

Arguments

`base_size`	base font size
`base_family`	base font family

See Also

ggtheme

Examples

```
## Not run:
data(GMAT)
data <- GMAT[, 1:20]
# total score calculation
df <- data.frame(score = apply(data, 1, sum))
# histogram
g <- ggplot(df, aes(score)) +
```

```
geom_histogram(binwidth = 1) +
xlab("Total score") +
ylab("Number of respondents")

g
g + theme_app()

## End(Not run)
```

Index