Package ‘flirt’

February 25, 2015

Type Package

Title Flexible Item Response Theory Modeling

Version 1.15

Date 2015-2-19

Author Minjeong Jeon, Frank Rijmen,and Sophia Rabe-Hesketh

Maintainer Minjeong Jeon <jeon.117@osu.edu>

Depends R (>= 2.13.0), stats

SystemRequirements Matlab Runtime Installer (MCR) for Matlab 2014a Windows 64bit

OS_type windows

URL http://www.mathworks.com/products/compiler/mcr/

Description Estimation of uni- and multi-dimensional explanatory IRT models

License GPL (>= 2)

Repository http://faculty.psy.ohio-state.edu/jeon/lab/flirt.php

LazyLoad yes

R topics documented:

`flirt-package` *Flexible item response theory modeling with efficient maximum likelihood (ML) estimation*

Description

This package provides a flexible framework for uni- and multi- dimensional explanatory item response theory modeling for binary and polytomous item responses. The flexibility stems from specifying IRT models as generalized linear and nonlinear mixed models (Rijmen, Tuerlinckx, De Boeck, & Kuppens, 2003).

For estimation, the package `flirt` uses an efficient modified EM algorithm based on the graphical model framework. The modified EM algorithm is much faster than the traditional EM algorithm. For more details on the modified EM algorithm, refer to e.g., Rijmen, F., Vansteelandt, K., & De Boeck, P. (2008).

Currently, uni- and multi- dimensional Rasch models, two-parameter logistic (2PL) IRT models, and bifactor models are available with extensions to multiple groups, item covariates, person covariates, and differential item functioning analyses.

The package `flirt` is based on the Matlab code BNLflirt (Rijmen and Jeon, 2013) for estimation that employs sub-functions from the Matlab toolbox BNL (Bayesian Networks with Logistic Regression Nodes; Rijmen, 2006).

flirt requires the Matlab Compiler Runtime (MCR) for Matlab 2014a, Windows. Having the correct version of the MCR is critical. If you have a different version of Matlab on your computer, please make sure to remove the MCR that you have and download/install the correct version in the following link:

`http://www.mathworks.com/products/compiler/mcr/`

To cite `flirt`,

Jeon, M., Rijmen, F. & Rabe-Hesketh, S. (2014). Flexible item response theory modeling with `flirt`. Applied Psychological Methods, 38, 404-405

Details

Package:	flirt
Type:	Package
Version:	1.15
Date:	2015-2-19
License:	GPL

The R script containing sample analyses using `flirt` is available by contacting the first author.

Author(s)

Minjeong Jeon, Frank Rijmen, and Sophia Rabe-Hesketh

Maintainer: Minjeong Jeon<jeon.117@osu.edu>

References

Jeon, M. and Rijmen, F. (2014). A modular approach for item response theory modeling with the R package flirt. Under revision.

Jeon, M. and De Boeck, P. (2015). A generalized item response tree model for psychological assessments. Under revision.

Jeon, M., Rijmen, F. & Rabe-Hesketh, S. (2014). Flexible item response theory modeling with `flirt`. Applied Psychological Methods, 38, 404-405

Jeon, M., Rijmen, F., and Rabe-Hesketh, R. (2013). Modeling differential item functioning using a generalization of the multiple-group bifactor model. Journal of Behavioral and Educational Statistics, 38, 32-60.

Rijmen, F. (2006). BNL: A Matlab toolbox for Bayesian networks with logistic regression. Technical Report. Vrije Universiteit Medical Center, Amsterdam.

Rijmen, F., Tuerlinckx, F., De Boeck, P., & Kuppens, P. (2003). A nonlinear mixed model framework for item response theory. Psychological Methods, 8, 185-205.

Rijmen, F., Vansteelandt, K., & De Boeck, P. (2008). Latent class models for diary method data: Parameter estimation by local computations. Psychometrika, 73, 167-182.

Rijmen, F. and Jeon, M. (2013). BNLflirt: Flexible item response theory modeling with BNL. Matlab file exchange.

`anova` *Anova method for fitted nested models*

Description

A generic function to conduct a likelihood ratio test between two nested models.

Usage

```
## S3 method for class flirt
anova(object, object2, ...)
```

Arguments

`object`	an object inheriting from class `flirt` for the model under the null hypothesis.
`object2`	an object inheriting from class `flirt` for the model under the alternative hypothesis.
`...`	additional arguments; currently none is used.

Value

An object of either class `anova.flirt` with components,

`name0`	the name of `object`.
`L0`	the log-likelihood under the null hypothesis (`object`).
`aic0`	the AIC value for the model given by `object`.
`bic0`	the BIC value for the model given by `object`.
`name1`	the name of `object2`.
`L1`	the log-likelihood under the alternative hypothesis (`object2`).
`aic1`	the AIC value for the model given by `object2`.
`bic1`	the BIC value for the model given by `object2`.
`LRT`	the value of the Likelihood Ratio Test statistic.
`df`	the degrees of freedom for the test (i.e., the difference in the number of parameters).
`p.value`	the p-value of the test.

Warning

The code does not check if the models are nested. The user is responsible to supply nested models in order for the LRT to be valid.

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

`flirt`, `logLik`

Examples

```
library(flirt)

## verbal aggression data

data(verb2)

# 2-dimensional 2PL model for binary data: a(th+b) parameterization
model1 <- flirt(data=verb2, select=2:25, loading=list(on=TRUE, inside=TRUE),
            mul=list(on=TRUE, dim_info=list(dim1=1:12, dim2=13:24))  )

#  2-dimensional Rasch model
model2 <- flirt(data=verb2, select=2:25,
            mul=list(on=TRUE, dim_info=list(dim1=1:12, dim2=13:24))  )

# LR test
anova(model2, model1) # model1 is nested within model2
```

`charity` *charity item responses*

Description

This data come from the 38th round of the State Survey conducted by Michigan State University's Institute for Public Policy and Social Research (2005). The survey was administered to 949 Michigan citizens from May 28 to July 18, 2005, by telephone. The focus of the survey included charitable giving and volunteer activities of Michigan households. Five questions measured the public's faith and trust in charity organizations.

Format

A data frame with 949 observations on the following five items:

ta1 Charitable organizations are more effective now in providing services than they were 5 years ago.

ta2 I place a low degree of trust in charitable organizations.

ta3 Most charitable organizations are honest and ethical in their use of donated funds.

ta4 Generally, charitable organizations play a major role in making our communities better places to live.

ta5 On the whole, charitable organizations do not do a very good job in helping those who need help.

The questions had four response categories corresponding to 'strongly agree', 'somewhat agree', 'somewhat disagree, and 'strongly disagree. In this dataset, the responses are coded from 0 to 3, with larger scores indicating less favorable views of charities.

References

Institute for Public Policy and Michigan State University Social Research. 2005. State of the state survey-38. Spring 2005. http://www.ippsr.msu.edu/SOSS.

Examples

```
str(charity)
```

`coef` *Extract estimated coefficients*

Description

A generic function to extract parameter estimates from `flirt` objects.

Usage

```
## S3 method for class flirt
coef(object, ...)
```

Arguments

object	an object inheriting from class flirt.
...	additional arguments; currently none is used.

Value

a matrix of the estimated parameters for the fitted model.

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

flirt

Examples

```
library(flirt)


## verbal aggression data

data(verb2)

# 2-dimensional 2PL model  for binary data:  a(th+b) parameterization
model1 <- flirt(data=verb2, select=2:25, loading=list(on=TRUE, inside=TRUE),
            mul=list(on=TRUE, dim_info=list(dim1=1:12, dim2=13:24))  )

coef(model1)
```

---

dendrify2 — *Expandsion of item responses based on an item resonse tree*

Description

Expands a wide-form matrix of item responses based on an item response tree structure

Usage

```
## S3 method for class flirt
dendrify2(mat, cmx, missing.omit=FALSE, wide=FALSE)
```

Arguments

mat	an integer matrix of IRT responses (columns represent items, rows represent respondents)
cmx	a mapping matrix with as many rows as there are response options for the items
missing	for a long-form data, the data lines with a missing item response are removed
wide	If TRUE, expanded data are provided in a wide form

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

References

De Boeck, P. and Partchev, I. (2012). IRTrees: Tree-Based Item Response Models of the GLMM Family, Journal of Statistical Software 48, 1-8.

Jeon, M. and De Boeck, P. (2015). A generalized item response tree model for psychological assessments. Under revision.

See Also

flirt

Examples

```
library(flirt)

## verbal aggression data

data(verb2)

# expand the data based on a tree structure (mapping matrix)
mapping <- cbind(c(0, 1, 1), c(NA, 0, 1))
wide <- dendrify2(verb2, mapping, wide=TRUE)
```

---

flirt *Uni- and Multi- dimensional Item Response Theory Analysis*

Description

Maximum likelihood estimation of item response theory (IRT) models for binary and polytomous data

Usage

```
flirt(data, select=NULL, subset=NULL, loading=list(on=FALSE, inside=FALSE),
    mul=list(on=FALSE, dim_info=NULL, cov_info=NULL),
    bifac=list(on=FALSE, dim_info=NULL, cov_info=NULL),
    second=list(on=FALSE, dim_info=NULL, cov_info=NULL),
    guess=list(on=FALSE),
    person_cov=list(on=FALSE, person_matrix=NULL, main=NULL),
    item_cov=list(on=FALSE, item_matrix_beta=NULL, item_matrix_alpha=NULL),
    dif=list(on=FALSE, dif_beta=NULL, dif_alpha=NULL),
    mg=list(on=FALSE, group_matrix=NULL), mixture =list(on=FALSE, num=NULL),
    weight=list(on=FALSE, weight_matrix=NULL), post=FALSE,
    start = list(on=FALSE, npar=NULL, start_info=NULL, start_value=NULL),
    constraint = list(on=FALSE, npar=NULL, cons_info=NULL, cons_value=NULL),
    evaluate = list(on=FALSE, eval_value=NULL),
    control=list(minpercent= NULL, max_it=NULL, nq=NULL, conv=NULL,
            link=NULL, adapt=NULL, se_num=NULL, se_emp=NULL, alp_bounds_up=NULL,
            verbose=NULL, show=NULL) )
```

Arguments

`data` matrix or data.frame in wide form - persons in rows and items in columns. Group membership, person covariates and sampling weights may be included as additional columns

`select` vector of item numbers or item names. If data include other than item responses or the user wants to analyze only subset of items, `select` should be specified.

`subset` vector of names or numbers of cases (rows) of choice.

`loading` list.

- **on**: logical. if `TRUE`, a 2PL model family, if `FALSE`, a 1PL model family (default) is used.
- **inside**: logical. When on==`TRUE`, if inside==`TRUE`, $\alpha_i(\theta_p + \beta_i)$ parameterization, if `FALSE`, $\alpha_i\theta_p + \beta_i$ parameterization (default) is used.

`mul` list.

- **on**: if `TRUE`, a multidimensional model, if `FALSE`, a unidimensional model (default) is used.
- **dim_info**: list. If on==`TRUE`, a list of item numbers or columns for each dimension should be provided.
- **cov_info**: matrix or data.frame. If there are person covariates, cov_info should be provided, which indicates which covariates are used in each dimension.

`bifac` list.

- **on**: logical. If `TRUE`, bifactor model, if `FALSE`, unidimensional model (default).
- **dim_info**: list. If on==`TRUE`, list of item numbers or columns for each dimension should be provided.
- **cov_info**: matrix or data.frame. If there are person covariates, cov_info should be provided, which indicates which covariates are used in each dimension.

`second` list.

- **on**: logical. If TRUE, second-order (testlet) model, if FALSE, unidimensional model (default).
- **dim_info**: list. If on==TRUE, list of item numbers or columns for each dimension should be provided.
- **cov_info**: matrix or data.frame. If there are person covariates, cov_info should be provided, which indicates which covariates are used in each dimension.

guess
: list.
- **on**: logical. If TRUE, a 3PL model is estimated (α_i, β_i, and c_i parameters), if FALSE, only β_i parameters are estimated (default is 1PL models).

person_cov
: list.
- **on**: logical. If TRUE, person covariates columns (or names) or the data matrix should be provided in person_matrix.
- **person_matrix**: column numbers or variable names if part of data, otherwise, matrix or data.frame for the person covariates. Number and order of rows should be the same as those in data. If subset is used, the number of rows should be adjusted accordingly.
- **main**: logical. If main==TRUE, person covariates are treated as main effects in the linear predictor instead of as predictors of a latent regression model. Default is main==FALSE.

item_cov
: list.
- **on**: logical. If TRUE, data matrix for item covariates should be provided either for β_i or α_i.
- **item_matrix_beta**: matrix or data.frame for item covariates that are regressed on β_i. Number of rows should be the same as the number of items. If select is used, the number of rows should be adjusted accordingly.
- **item_matrix_alpha**: matrix or data.frame for item covariates that are regressed on α_i. Number of rows should be the same as the number of items. If this option is used, standard errors are not estimated. If select is used, the number of rows should be adjusted accordingly.

dif
: list.
- **on**: logical. If TRUE, DIF analysis is performed. FALSE is default.
- **dif_beta**: vector. Item numbers or variable names for DIF for β_i.
- **dif_alpha**: vector. Item numbers or variable names for DIF for α_i.

mg
: list.
- **on**: logical. If TRUE, multiple-group analysis is performed. FALSE is default.
- **group_matrix**: column number or variable name for person membership if part of data, otherwise data.matrix should be provided in group_matrix. Number and order of rows should be the same as those in data. If subset is used, the number of rows should be adjusted accordingly. Persons should be ordered by groups.

mixture
: list.
- **on**: logical. If TRUE, mixture model is specified (currently, not available).
- **num**: logical. If on==TRUE, number of latent classes should be provided.

weight
: list.
- **on**: logical. If TRUE, sampling or frequency weights are included. FALSE is default.

- **weight_matrix**: if on==TRUE, column number or variable name if part of data, otherwise data.matrix for weights should be provided. Number and order of rows should be the same as those in data. If subset is used, the number of rows should be adjusted accordingly.

post
: logical. if TRUE, expected a posteriori (EAP) (and its variance and covariance), expected sum-scores, and IRT reliability are provided. FALSE is default.

start
: - **on**: logical. If TRUE, use-specified starting values are used; otherwise, random starting values are used (default).
  - **npar**: integer of the total number of parameters.
  - **start_info**: vector of the parameter numbers for starting values to be used.
  - **start_value**: vector of the starting values for the specified parameters in start_info.

constraint
: - **on**: logical. If TRUE, users can specify parameter constraints. FALSE is default.
  - **npar**: integer of the total number of parameters.
  - **cons_info**: vector of the parameter numbers to be constrained.
  - **cons_value**: vector of the specific values for the specified parameters to be fixed at.

evaluate
: - **on**: logical. If TRUE, log-likelihood is evaluated at given parameter values. FALSE is default.
  - **eval_value**: vector of parameter values where the log-likelihood is evaluated at.

control
: list of control options with components:
  - **minpercent**: positive real value. Minimum percentage for a response category for polytomous items. If the relative frequency for a category is lower than minpercent, that category is collapsed into the lower adjacent category. Default is 0. To see how the category is collapsed, put the control option show=TRUE.
  - **max_it**: positive integer. Maximum number of iterations. Default is 10,000.
  - **nq**: positive integer. Numbers of quadrature points. Default is 20. For multidimensional and bifactor models, a vector of quadrature points for each dimension in order (dimension 1, dimension 2, and so on for multidimensional models; general dimension, specific dimension 1, specific dimension 2, and so on for bifactor models). If a scalar is specified, the specified number is used for all dimensions.
  - **conv**: positive real value. Convergence criterion. The iteration between EM estimations stops when the maximum absolute difference in the parameter estimates becomes equal or smaller than the criterion between two subsequent iterations. Default is 0.0001.
  - **link**: positive integer 1, 2, or 3. or "multinomial", "cumulative", "adjacent". Default is multinomial or 1 for binary data (that leads to logit link) and "adjacent" for polytomous data.
  - **adapt**: logical. If TRUE, adaptive quadrature is used. if FALSE, Gauss-Hermite quadrature is used (default).
  - **se_num**: logical. If TRUE, standard error estimates are computed using the Hessian matrix obtained by numerical differentiation.
  - **se_emp**: logical. If TRUE, standard error estimates are computed using an empirical information matrix.(default) If **se_num** is TRUE, standard errors are computed using this method are contained as extra attributes. If **se_emp** is FALSE, these standard errors are used as default standard errors.

- **alp_bounds_up**: positive integer. Upper boundary value (positive) for α_i. Default is 15. Cannot be used when item covariates are regressed on α_i.
- **alp_bounds_low**: positive integer. Lower boundary value for α. Default is -1. Cannot be used when item covariates are regressed on α_i.
- **verbose**: logical. If `TRUE`, the iteration number, the maximum absolute difference in the parameter estimates (par_diff) and difference in the log-likelihood (lik_diff) between adjacent iterations are printed.
- **show**: logical. If `TRUE`, print Matlab output (error messages) on R console.

Details

- **Parameterization**: For a linear predictor `flirt` uses $\theta_p + \beta_i$ for 1PL models, $\alpha_i(\theta_p + \beta_i)$ or $\alpha_i\theta_p + \beta_i$ for 2PL models, and $\alpha_{ig}\theta_{pg} + \alpha_{is}\theta_{ps} + \beta_i$ for bifactor and second-order models, where θ_p is the ability of a person p, β_i is the item easiness (or intercept), α_i is the loading (or slope), and θ_{pg} and θ_{ps} are abilities for the general g and specific dimension s with α_{ig} and α_{is}, respectively. For 3PL models, the guessing parameter c_i is incorporated in the probability, $P_i = c_i + \frac{(1-c_i)}{1+exp(-(\alpha_i\theta_p+\beta_i))}$. For second-order models, α_{ig}/α_{is} is the second-order loading for the sth first-order factor on the second-order factor θ_{pg}, which is constant for the items within sth first-order factor.
- **Polytomous item responses**: Minimum category should be 0.
- **Multiple group analysis**: Group membership should start from 0.
- **Standard errors**: For variance-covariance parameters, standard errors are not provided and NA is returned. For parameters that are constrained, standard errors are not calculated and NA is returned.
- **Starting values, evaluation**: Parameter values (to be fixed at) for variance-covariance parameters should be Cholesky elements of lower triangular matrix L ($Cov = LL'$)
- **Boundary values**: The maximum boundary values for parameters are set to ± 15. The upper and lower boundary values for α_i can be modified using the control options `alp_bounds_up` and `alp_bounds_low`. If a parameter estimate crosses the boundary value during iterations, the parameter estimate is automatically restricted to the boundary value and no parameter and standard error estimates are provided with a warning message.
- **Missing values**: Missing values should be specified as NA. Missing values in item responses and item design matrices are treated as ignorable, but for missing values in person design, person group, and weight matrices, listwise deletion is used.

Value

An object of class `flirt`, with the following slots that can be extracted using object@

`pars`	matrix of parameters estimates and standard errors
`parms`	original parameter estimates from BNLflirt (including cholesky estimates for multidimensional models)
`info_num`	information matrix (numerical) evaluated at the maximum likelihood estimates
`se_emp`	standard errors using empirical information matrix (if used)
`info_emp`	information matrix (empirical) evaluated at the maximum likelihood estimates (if used)
`loglik`	log-likelihood value at convergence
`AIC`	Akaike information criterion: $-2 * loglik + 2 * npar$, where $npar$ is number of parameters

`BIC` Bayesian information criterion: $-2 * loglik + npar * log(nobs)$, where $nobs$ is number of cases (persons)

`npar` number of parameters

`post` post analysis with a list of
- eap: expected a posteriori (EAP) ability estimate(s)
- eap_var: variances of expected a posteriori (EAP) ability estimate(s)
- eap_cov: covariance matrix of expected a posteriori (EAP) ability estimate(s)
- exp_s: expected scores
- rel: IRT reliability using empirical priors

`data_inf` data information with a list of
- nobs: number of people
- ngroup: number of groups
- nitem: number of items
- MS_item:maximum category of items (scalar)
- S_item:categories of all items (vector)

`dim_inf` dimension information with a list of
- within: within-item model:1, otherwise:0
- ndim: number of dimensions
- n_it_dim: number of items in each dimension

`inside` inside:1, otherwise:0 (2PL parameterization)

`model` 1:unidimensional 1PL model family, 2:unidimensional 2PL model family, 3: multidimensional 1PL model family, 4: multidimensional 2PL model family, 5: bifactor model family

`dif_beta` items that are under investigation for DIF for β_i

`dif_alpha` items that are under investigation for DIF for α_i

`est_inf` estimation information with a list of
- adapt: adaptive quadrature
- nqr: number of quadrature points
- conv: convergence criterion
- max_it: total number of iterations
- link: link function
- verbose: verbose
- alp_bounds_up: upper boundary value for α_i
- alp_bounds_low: lower boundary value for α_i

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

`summary`, `coef`, `logLik`, `anova`

Examples

```
# set directory where the .tar is located.
install.packages("flirt", type="source", repos=NULL)
library(flirt)

# show built-in datasets
data(package="flirt")

# with adjacent link function (partial credit model)
result1 <- flirt(data=charity, subset=1:100,
        control=list(minpercent=0.05, nq=5, link="adjacent", show=TRUE))

## verbal aggression data
data(verb2)

# 2-dimensional 2PL model for binary data
# with a(th+b) parameterization
result2 <- flirt(data=verb2, select=2:25,
            mul=list(on=TRUE, dim_info=list(dim1=1:12, dim2=13:24))  )

# output
result2
summary(result2)
coef(result2)
logLik(result2)


# 1PL graded response model for polytomous responses
# with cumulative link function (graded response model)
data(charity)
result3 <- flirt(data=charity, subset=1:100,
        control=list(minpercent=0.05, nq=5, link="cumulative", show=TRUE))
```

IRF *Plot item response curves*

Description

A function to plot item response curves as a function of latent ability for unidimensional 1PL, 2PL, and 3PL models.

Usage

```
## S3 method for class flirt
IRF(alpha=NULL, beta, guess=NULL , inside=NULL, ylim=NULL)
```

Arguments

alpha	α_i parameter estimates. If NULL, $\alpha_i = 1$
beta	β_i parameter estimates
guess	c_i (guessing) parameter estimates. If NULL, $c_i = 0$
inside	If TRUE, $P_i = c_i + \frac{(1-c_i)}{1+exp(-\alpha_i(\theta_p+\beta_i))}$ is estimated
ylim	the y limits of the plot.

Details

For single and multiple items.

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

flirt

Examples

```
library(flirt)

## verbal aggression data

data(verb2)

# 2-dimensional 2PL model  for binary data:  a(th+b) parameterization
result2 <- flirt(data=verb2, loading=list(on=T, inside=T), control=list(nq=5) )
alpha <- result2@pars[1:24,2]
beta <- result2@pars[25:48,1]

# item resonse curve for four items
IRF(beta=beta[1:4], alpha=alpha[1:4], inside=T)
```

IRF_pol *Plot category response curves for polytomous items*

Description

A function to plot category response curves as a function of a unidimensional ability for unidimensional graded response models (with a cumulative logit link)

Usage

```
## S3 method for class flirt
IRF_pol(alpha=NULL, beta, guess=NULL , inside=NULL, ylim=NULL)
```

Arguments

alpha	α_i parameter estimates. If NULL, $\alpha_i = 1$
beta	β_{ir} parameter estimates for category r to item i
guess	c_i (guessing) parameter estimates. If NULL, $c_i = 0$
inside	If TRUE, $P_i = c_i + \frac{(1-c_i)}{1+exp(-\alpha_i(\theta_p+\beta_{ir}))}$ is estimated
ylim	the y limits of the plot.

Details

For a single item with three to six response categories.

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

flirt

Examples

```
library(flirt)

## charity data

result4 <- flirt(data=charity, subset=1:100, control=list(minpercent=0.05, nq=5, link="cumulative"))

beta <- result4@pars[2:14,1]

# item response function
IRF_pol(beta=beta[1:2])  # item 1 with three categories
IRF_pol(beta=beta[3:5])  # item 2 with four categories
```

---

item_design_bin *Verbal Aggression Data: item covariates*

Description

Data frame for item design matrix with binary responses for the verbal aggression data. There are five columns for for intercept, four item covariates (dowant, otherself, blame, express).

Format

A data frame in size 24 by 5.

intercept: 1

dowant: do:1 vs. want:0

otherself: other-to-blame:0, self-to-blame:1

blame: curse:0.5, scold:0.5 vs. shout:-1

express: scold:-1 vs. curse:0.5, shout:0.5

References

De Boeck, P. and Wilson, M. (2004), Explanatory Item Response Models, Springer.

Examples

```
str(item_design_bin)
```

---

item_design_pol *Verbal Aggression Data: item covariates*

Description

Data frame for item design matrix for polytomous responses for the verbal aggression data. There are six columns for intercept, four item covariates (dowant, otherself, blame, express), and one variable for category 2 (compared to 1) (same for 24 items). Two adjacent lines represent an item, one for category 1 and next for category 2 (category 0 is reference). This item design matrix assumes that a difference between category 1 to 2 is the same for all items. For unequal category effects, 24 extra columns (one for each item) are needed instead of one column (item_design_pol_full).

Format

A data frame in size 48 by 6.

intercept: 1

dowant: do:1 vs. want:0

otherself: other-to-blame:0, self-to-blame:1

blame: curse:0.5, scold:0.5 vs. shout:-1

express: scold:-1 vs. curse:0.5, shout:0.5

category: category 1 (perhaps):0, category 2 (yes):1

References

De Boeck, P. and Wilson, M. (2004), Explanatory Item Response Models, Springer.

Examples

```
str(item_design_pol)
```

`item_design_pol_full` *Verbal Aggression Data: item covariates*

Description

Data frame for item design matrix for polytomous responses for the verbal aggression data. There are 29 columns for intercept, four item covariates (blame, express, dowant, otherself), and 24 variables for category 2 (compared to 1). Two adjacent lines represent an item, one for category 1 and next for category 2 (category 0 is reference). This item design matrix assumes that a difference between category 1 to 2 is different across items. For equal category effects, only one extra column is needed instead of 24 columns (item_design_pol).

Format

A data frame in size 48 by 29.

intercept: 1

dowant: do:1 vs. want:0

otherself: other-to-blame:0, self-to-blame:1

blame: curse:0.5, scold:0.5 vs. shout:-1

express: scold:-1 vs. curse:0.5, shout:0.5

cat_i1 to cat_i24: category 1 (perhaps):0, category 2 (yes):1 for item 1 to 24

References

De Boeck, P. and Wilson, M. (2004), Explanatory Item Response Models, Springer.

Examples

```
str(item_design_pol_full)
```

`item_design_rating` *Verbal Aggression Data*

Description

Data frame for item design matrix for for polytomous responses for the verbal aggression data. There are 25 columns for 24 items and one variable for category 2 (compared to 1) (same for 24 items). Two adjacent lines represent an item, one for category 1 and next for category 2 (category 0 is reference). This item design matrix assumes that a difference between category 1 to 2 is the same for all items (as in a rating scale model).

Format

A data frame in size 48 by 25.

i1 to i24 indicator variable for item 1 to 24

category: category 1 (perhaps):0, category 2 (yes):1

References

De Boeck, P. and Wilson, M. (2004), Explanatory Item Response Models, Springer.

Examples

```
str(item_design_rating)
```

Item_info *Plot item information function*

Description

A function to plot item information functions for unidimensional 1PL, 2PL, and 3PL models

Usage

```
## S3 method for class flirt
Item_info(alpha=NULL, beta, guess=NULL , inside=NULL, ylim=NULL)
```

Arguments

alpha	α_i parameter estimates. If NULL, $\alpha_i = 1$
beta	β_i parameter estimates
guess	c_i (guessing) parameter estimates. If NULL, $c_i = 0$
inside	If TRUE, $I_i(\theta) = \alpha_i^2 \times \frac{(P_i - c_i)}{(1-c_i)^2} \times \frac{(1-P_i)}{P_i}$ is computed
ylim	the y limits of the plot.

Details

For single and multiple items.

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

flirt

Examples

```
library(flirt)

## verbal aggression data

data(verb2)

# 2-dimensional 2PL model  for binary data:  a(th+b) parameterization
result2 <- flirt(data=verb2, loading=list(on=T, inside=T), control=list(nq=5) )
alpha <- result2@pars[1:24,2]
beta <- result2@pars[25:48,1]

# item information function for first four items
Item_info(beta= beta[1:4])
```

`linking` *Data example for IRT linking*

Description

Sample data for cases with two test forms. The data include three common items, two items from test form A, and one item from test form B.

Format

A data frame in size 30 by 6.

c1 to c3: three common items

f1.1 to f2.1: two items from test form A

f1.2: one item from test form B

Examples

```
str(linking)
```

`logLik` *Extract log-likelihood*

Description

A generic function to extract the log-likelihood from `flirt` objects.

Usage

```
## S3 method for class flirt
logLik(object, ...)
```

Arguments

`object`	an object inheriting from class `flirt`.
`...`	additional arguments; currently none is used.

Value

Returns an object of class `logLik` giving the log-likelihood value and `npar` (number of freely estimated parameters)

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

`flirt`

Examples

```
library(flirt)

## verbal aggression data

data(verb2)

# 2-dimensional 2PL model  for binary data: a(th+b) parameterization
model1 <- flirt(data=verb2, select=2:25, loading=list(on=TRUE, inside=TRUE),
           mul=list(on=TRUE, dim_info=list(dim1=1:12, dim2=13:24))  )

logLik(model1)
```

`person_design` *Verbal Aggression Data: Person design matrix*

Description

Design matrix for person covariates

Format

A data.frame for 316 cases with the following variable:

male subject's gender. male:1 and female:0

References

De Boeck, P. and Wilson, M. (2004), Explanatory Item Response Models, Springer.

Examples

```
str(person_design)
```

`science` *science item responses*

Description

This data come from the British sample of the 1992 Eurobarometer Survey (Reif and Melich, 1992) on perceptions of science and technology provided by Bartholomew et al. (2008). There are seven statements (such as 'Science and technology are making our lives healthier, easier, and more comfortable') and the response categories are 'strongly disagree', 'disagree to some extent', 'agree to some extent', and 'strongly agree'.

Bartholomew et al. (2008) considered a between-item two-dimensional generalized partial credit model by using two item clusters, 1) items 1, 3, 4, and 7, and 2) items 2, 5, and 6.

Format

A data frame with 392 observations with 7 items. Each item has five response categories from 'strongly disagree' to 'strongly agree'.

References

Bartholomew, D. J., Steele, F., Moustaki, I., and Galbraith, J. I. (2008). Analysis of Multivariate Social Science Data (Second Edition). Chapman & Hall/CRC, Boca Raton, FL.

Reif, K. andMelich, A. (1992). Euro-barometer 38.1: Consumer protection and perceptions of science and technology. Technical report, INRA (Europe), Brussels.

Examples

```
str(science)
```

`spelling` *spelling item responses*

Description

This data come from Thissen et al. (1993, p.71) that examined performance of under- graduates on four spelling items 'infidelity', 'panoramic', 'succumb' and 'girder'. Each spelling item was scored either as correct or incorrect. The sample includes 285 male and 374 female undergraduate students from the University of Kansas.

Format

A data frame with 659 observations on the four spelling items. There is one more variable, `male` that takes value 1 for male and 0 for female.

References

Thissen, D., L. Steinberg, and H. Wainer. 1993. Detection of differential item functioning using the parameters of item response models. In Differential Item Functioning, ed. P. Holland and H. Wainer, 67-114. Hillsdale, NJ: Lawrence Erlbaum.

Examples

```
str(spelling)
```

---

spelling_w *spelling item responses with frequency weights*

Description

This data come from Thissen et al. (1993, p.71) that examined performance of under- graduates on four spelling items ‘infidelity’, ‘panoramic’, ‘succumb’ and ‘girder’. Each spelling item was scored either as correct or incorrect. The sample includes 285 male and 374 female undergraduate students from the University of Kansas.

Format

A data frame with 30 observations on the four spelling items. There is one covariate, `male` that takes value 1 for male and 0 for female. There is a frequency weight, `wt2` that indicates the number of people who share the same item response patterns and gender.

References

Thissen, D., L. Steinberg, and H. Wainer. 1993. Detection of differential item functioning using the parameters of item response models. In Differential Item Functioning, ed. P. Holland and H. Wainer, 67-114. Hillsdale, NJ: Lawrence Erlbaum.

Examples

```
str(spelling_w)
```

---

std_coef *Standardized factor loading parameter estimates and correlation matrix*

Description

`std_coef` compute standardized factor loading parameter estimates and correlation matrix.

`std_cov` scales a covariance matrix into the corresponding correlation matrix.

Usage

```
## S3 method for class flirt
std_coef(est, dim_info, cov_matrix)
## S3 method for class flirt
std_cov(dim_info, cov_matrix)
```

Arguments

`est`	a vector of loading parameter estimates from `flirt`
`dim_info`	a list of items for each dimension
`cov_matrix`	a square numeric variance_covariance matrix

Value

`std_coef` returns a list of standardized loading parameter estimates and correlation matrix. `std_cov` returns a correlation matrix.

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

`flirt`

Examples

```
library(flirt)


## verbal aggression data: polytmous item responses

data(verb2)

# 2-dimensional 2PL model for binary data
# a*th + b parameterization
model1 <- flirt(data=resp, loading=list(on=TRUE, inside=FALSE),
    mul=list(on=TRUE, dim_info=list(dim1=1:12, dim2=13:24)) , control=list(nq=5) )

## function for standardized loadings and covariance matrix
# length 24, unstandardized loading parameter estimates
est <- coef(model1)[1:24,1]

# length 3 (sd1, cov12, sd2) unstandardized sd and cov estimates vector
cov_el <- coef(model1)[51:53,1]

# covariance matrix
cov_matrix <- matrix(c(cov_el[1]^2,cov_el[3], cov_el[3],cov_el[2]^2),2,2, byrow=FALSE)

# list
dim_info <- list(dim1=1:12, dim2=13:24)

test0 <- std_coef(est=est, dim_info=dim_info, cov_matrix= cov_matrix)
test1 <- std_cov(dim_info=dim_info, cov_matrix= cov_matrix)
```

summary — *Summary method for fitted models*

Description

A generic function to produce result summaries from `flirt` objects.

Usage

```
## S3 method for class flirt
summary(object, ...)
```

Arguments

`object`	an object inheriting from class `flirt`.
`...`	additional arguments; currently none is used.

Value

An object of class `summary.flirt` with components,

`Data`	nobs: number of observations (nobs), nitem: number of items, maxcat: maximum response category, and ngroup: number of groups
`Model fit`	npar: number of parameters, AIC, BIC, and loglik
`Parameterization`	for 2PL models, $\alpha\theta + \beta$ or $\alpha(\theta + \beta)$
`Type`	between-item or within-item multidimensional models
`Dimension`	number of dimensions and number of items in each dimension for multidimensional and bifactor models
`Parameter estimates`	parameter estimates and standard errors.

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

`flirt`

Examples

```
library(flirt)

## verbal aggression data
data(verb2)

# 2-dimensional 2PL model  for binary data: a(th+b) parameterization
model1 <- flirt(data=verb2, select=2:25, loading=list(on=TRUE, inside=TRUE),
            mul=list(on=TRUE, dim_info=list(dim1=1:12, dim2=13:24))  )

summary(model1)
```

`Test_info` *Plot test information function*

Description

A function to plot test information function for unidimensional 1PL, 2PL, and 3PL models

Usage

```
## S3 method for class flirt
Test_info(alpha=NULL, beta, guess=NULL , inside=NULL, ylim=NULL)
```

Arguments

alpha	α_i parameter estimates. If NULL, $\alpha_i = 1$
beta	β_i parameter estimates
guess	c_i (guessing) parameter estimates. If NULL, $c_i = 0$
inside	If TRUE, $I(\theta) = \sum_i^I [\alpha_i^2 \times \frac{(P_i - c_i)}{(1-c_i)^2} \times \frac{(1-P_i)}{P_i}]$ is computed
ylim	the y limits of the plot.

Author(s)

Minjeong Jeon <jeon.117@osu.edu>

See Also

flirt

Examples

```
library(flirt)

## verbal aggression data

data(verb2)
```

```
# 2-dimensional 2PL model  for binary data:  a(th+b) parameterization
result2 <- flirt(data=verb2, loading=list(on=T, inside=T), control=list(nq=5) )
alpha <- result2@pars[1:24,2]
beta <- result2@pars[25:48,1]

# test information function
Test_info(beta= beta)
```

---

verb2 *Verbal Aggression Data: binary responses*

Description

Data frame in wide form for the verbal aggression data.

Format

A data frame for 316 cases with the following variables:

y1-y24 binary item responses to item 1 to item 24. The order of items are the same as that in VerbAgg (long form). No:0, Perhaps and Yes:1

References

De Boeck, P. and Wilson, M. (2004), Explanatory Item Response Models, Springer.

Examples

```
str(verb2)
```

---

verb3 *Verbal Aggression Data: polytomous responses*

Description

Data frame in wide form for the verbal aggression data.

Format

A data frame for 316 cases with the following variables.

y1-y24 original polytomous item responses to item 1 to item 24. The order of items are the same as that in VerbAgg (long form). No:0, Perhaps:1, and Yes:2

References

De Boeck, P. and Wilson, M. (2004), Explanatory Item Response Models, Springer.

Examples

```
str(verb3)
```

`VerbAgg` *Verbal Aggression item responses*

Description

These are the item responses to a questionnaire on verbal aggression. These data are used throughout De Boeck, P. and Wilson, M. (2004) to illustrate various forms of item response models.

Format

A data frame with 7,584 observations on the following 13 variables.

Anger the subject's Trait Anger score as measured on the State-Trait Anger Expression Inventory (STAXI)

Gender the subject's gender - a factor with levels M and F

item the item on the questionnaire, as a factor

resp the subject's response to the item - an ordered factor with levels no < perhaps < yes

id the subject identifier, as a factor

btype behavior type - a factor with levels curse, scold and shout

situ situation type - a factor with levels other and self indicating other-to-blame and self-to blame

mode behavior mode - a factor with levels want and do

r2 dichotomous version of the response - a factor with levels N and Y

Source

http://bear.soe.berkeley.edu/EIRM

References

De Boeck, P. and Wilson, M. (2004), Explanatory Item Response Models, Springer.

Examples

```
str(VerbAgg)
```

Index