Package ‘MultiLCIRT’

Type Package

Title Multidimensional latent class Item Response Theory models

Version 2.3

Date 2013-03-02

Author Francesco Bartolucci, Silvia Bacci, Michela Gnaldi - University of Perugia (IT)

Maintainer Francesco Bartolucci <bart@stat.unipg.it>

Description This package provides a flexible framework for the Item Response Theory analysis of dichotomous and ordinal polytomous outcomes under the assumption of multidimensionality and discreteness of the latent traits. The fitting algorithms allow for missing responses and for different item parameterizations and are based on the Expectation-Maximization paradigm. Individual covariates affecting the class weights may be included in the new version (since 2.1).

License GPL (>= 2)

Depends MASS

NeedsCompilation yes

Repository CRAN

Date/Publication 2013-03-03 16:53:52

R topics documented:

MultiLCIRT-package .................................................. 2
aggr_data ............................................................. 3
class_item ........................................................... 4
est_multilogit ......................................................... 6
est_multi_glob ......................................................... 7
est_multi_poly ....................................................... 8
hads ................................................................. 11
Description

This package provides a flexible framework for the Item Response Theory (IRT) analysis of dichotomous and ordinal polytomous outcomes under the assumption of multidimensionality and discreteness of latent traits (abilities). Every level of the abilities identify a latent class of subjects. The fitting algorithms are based on the Expectation-Maximization (EM) paradigm and allow for missing responses and for different item parameterizations. The package also allows for the inclusion individual covariates affecting the class weights.

Details

Package: MultiLCIRT
Type: Package
Version: 2.3
Date: 2013-03-02
License: GPL (>= 2)

Function est_multi_poly performs the parameter estimation of the following IRT models, allowing for one or more latent traits:

- Binary responses: Rasch model, 2-Parameter Logistic (2PL) model;
- Ordinal polytomous responses: Samejima’s Graded Response Model (GRM) and constrained versions with fixed discrimination parameters and/or additive decomposition of difficulty parameters (rating scale parameterization); Muraki’s Generalized Partial Credit Model and constrained versions with fixed discrimination parameters and/or additive decomposition of difficulty parameters, such as Partial Credit Model and Rating Scale Model.

The basic input arguments for est_multi_poly are the person-item matrix of available response configurations and the corresponding frequencies, the number of latent classes, the type of link function, the specification of constraints on the discriminating and difficulty item parameters, and the allocation of items to the latent traits. Missing responses are coded with 999, and units and items without responses are automatically removed.

Function test_dim performs a likelihood ratio test to choose the optimal number of latent traits.
(or dimensions) by comparing nested models that differ in the number of latent traits, being all the other elements let equal (i.e., number of latent classes, type of link function, constraints on item parameters). The basic input arguments for `test_dim` are similar as those for `est_multi_poly`.

Function `class_item` performs a hierarchical clustering of items based on a specified LC IRT model. The basic input arguments are given by the number of latent classes, the type of model, and the constraints on the item parameters (only for polytomous responses). An allocation of items to the different latent traits is obtained depending on the cut-point of the resulting dendrogram.

**Author(s)**

Francesco Bartolucci, Silvia Bacci, Michela Gnaldi - University of Perugia (IT)

Maintainer: Francesco Bartolucci <bart@stat.unipg.it>

**References**


**Examples**

```r
## Estimation of different Multidimensional LC IRT models with binary responses
# Aggregate data
out = aggr_data(X)
S = out$data_dis
yv = out$freq
# Define matrix to allocate each item on one dimension
multi1 = rbind(c(1,2,9,1/zero.noslash),c(3,5,8,11),c(4,6,7,12))
# Three-dimensional LC Rasch model with 4 latent classes
out1 = est_multi_poly(S,yv,k=4,start=/zero.noslash^-6)
```

---

**Description**

Given a matrix of configurations (covariates and responses) unit-by-unit, this function finds the corresponding matrix of distinct configurations and the corresponding vector of frequencies (it does not work properly with missing data).

**Usage**

```r
aggr_data(data, disp=FALSE)
```
class_item

Arguments

data matrix of covariate and unit-by-unit response configurations
disp to display partial results

Value
data_dis matrix of distinct configurations
dfreq vector of corresponding frequencies
dlabel the index of each provided response configuration among the distinct ones

Author(s)

Francesco Bartolucci - University of Perugia (IT)

Examples

# draw a matrix of random responses and find distinct responses
X = matrix(sample(5,100,replace=TRUE),50,2)
out = aggr_data(X)

# find the distinct responses and the corresponding vector of frequencies for naep data
data(naep)
X = as.matrix(naep)
out = aggr_data(X)
length(out$freq)

Description

It performs a hierarchical classification of a set of test items on the basis of the responses provided by a sample of subjects. The classification is based on a sequence of likelihood ratio tests between pairs of multidimensional models suitably formulated.

Usage

class_item(S, yv, k, link = 1, disc = 0, difl = 0, fort = FALSE, disp = FALSE)

Arguments

S matrix of all response sequences observed at least once in the sample and listed row-by-row (use 999 for missing response)
yv vector of the frequencies of every response configuration in S
k number of ability levels (or latent classes)
### Examples

```r
## Not run:
## Model-based hierarchical classification of items from simulated data
# Setup
r = 6  # number of items	n = 1000  # sample size
bev = rep(0, r)
k = r/2
multi = rbind(1:(r/2),(r/2+1):r)
L = chol(matrix(c(1, 0.6, 0.6, 1), 2, 2))
data = matrix(0, n, r)
model = 1
# Create data
```
est_multilogit

Estimate multilogit model

Description

The function performs maximum likelihood estimation of the multilogit model (internal function).

Usage

est_multilogit(Y, Xdis, label = 1:n, be = NULL, Pdis = NULL, dis = F, fort = F, ex = F)

Arguments

Y matrix of all responses
Xdis matrix of all covariates (two- or three-dimensional)
est_multi_glob

label label associated to every covariate configuration
be initial value of the parameter vector
Pdis probability configurations as initial values
dis to display partial results
fort to use fortran routines when possible
ex TRUE if the function exits without running the estimation algorithm providing the score vector and the information matrix

Value

be estimated vector regression coefficients
P estimated matrix of probabilities
Pdis estimated matrix of distinct probabilities
sc score vector
Fi information matrix

Author(s)

Francesco Bartolucci - University of Perugia (IT)

---

est_multi_glob Fit marginal regression models for categorical responses

Description

It estimates marginal regression models to datasets consisting of a categorical response and one or more covariates by a Fisher-scoring algorithm; this is an internal function.

Usage

est_multi_glob(yv, X, model, ind = rep(1, length(yv)), w = rep(1, length(yv)), be = NULL)

Arguments

yv vector of responses
X matrix of all distinct covariate configurations
model type of logit (g = global, l = local, m = multinomial)
ind vector to link responses to covariates
w vector of weights
be initial vector of regression coefficients
Value

be estimated vector of regression coefficients
lk log-likelihood at convergence
P matrix of the probabilities for each covariate configuration

Author(s)

Francesco Bartolucci - University of Perugia (IT)

References


---

**est_multi_poly** Estimate multidimensional LC IRT model for dichotomous and polytomous responses

**Description**

The function performs maximum likelihood estimation of the parameters of the IRT models assuming a discrete distribution for the ability. Every ability level corresponds to a latent class of subjects in the reference population. Maximum likelihood estimation is based on Expectation-Maximization algorithm.

**Usage**

```r
est_multi_poly(S, yv = rep(1,ns), k, X = NULL, start = /zero.noslash, link = /zero.noslash, disc = /zero.noslash, difl = /zero.noslash, multi = 1:J, piv = NULL, Phi = NULL, gac = NULL, De = NULL, fort = FALSE, tol = 1/zero.noslash^-1/zero.noslash, disp = FALSE, output = FALSE)
```

**Arguments**

- `S` matrix of all response sequences observed at least once in the sample and listed row-by-row (use NA for missing response)
- `yv` vector of the frequencies of every response configuration in `S`
- `k` number of ability levels (or latent classes)
- `X` matrix of covariates that affects the weights
- `start` method of initialization of the algorithm (0 = deterministic, 1 = random, 2 = arguments given as input)
**est_multi_poly**

- **link**: type of link function (0 = no link function, 1 = global logits, 2 = local logits); with no link function the Latent Class model results; with global logits the Graded Response model results; with local logits the Partial Credit results (with dichotomous responses, global logits is the same as using local logits resulting in the Rasch or the 2PL model depending on the value assigned to disc)

- **disc**: indicator of constraints on the discriminating indices (0 = all equal to one, 1 = free)

- **difl**: indicator of constraints on the difficulty levels (0 = free, 1 = rating scale parameterization)

- **multi**: matrix with a number of rows equal to the number of dimensions and elements in each row equal to the indices of the items measuring the dimension corresponding to that row

- **piv**: initial value of the vector of weights of the latent classes (if start=2)

- **Phi**: initial value of the matrix of the conditional response probabilities (if start=2)

- **gac**: initial value of the complete vector of discriminating indices (if start=2)

- **De**: initial value of regression coefficients for the covariates (if start=2)

- **fort**: to use fortran routines when possible

- **tol**: tolerance level for checking convergence of the algorithm as relative difference between consecutive log-likelihoods

- **disp**: to display the likelihood evolution step by step

- **output**: to return additional outputs (Phi,Pp,Piv)

**Value**

- **piv**: estimated vector of weights of the latent classes (average of the weights in case of model with covariates)

- **Th**: estimated matrix of ability levels for each dimension and latent class

- **Bec**: estimated vector of difficulty levels for every item (split in two vectors if difl=1)

- **gac**: estimated vector of discriminating indices for every item (with all elements equal to 1 with Rasch parametrization)

- **fv**: vector indicating the reference item chosen for each latent dimension

- **Phi**: array of the conditional response probabilities for every item and latent class

- **De**: matrix of regression coefficients for the multinomial logit model on the class weights

- **Piv**: matrix of the weights for every response configuration

- **Pp**: matrix of the posterior probabilities for each response configuration and latent class

- **lk**: log-likelihood at convergence of the EM algorithm

- **np**: number of free parameters

- **aic**: Akaike Information Criterion index

- **bic**: Bayesian Information Criterion index
Examples

## Estimation of different Multidimensional LC IRT models with binary responses

### Aggregate data

data(naep)
X = as.matrix(naep)
out = aggr_data(X)
S = out$data_dis
yv = out$freq

### Define matrix to allocate each item to one dimension
multi1 = rbind(c(1,2,9,10),c(3,5,8,11),c(4,6,7,12))

### Three-dimensional Rasch model with 3 latent classes
# the tolerance level has been rise to increase the speed (to be reported to a smaller value)
out1 = est_multi_poly(S,yv,k=3,start=0,link=1,multi=multi1,tol=10^-6)

## Not run:

### Three-dimensional 2PL model with 3 latent classes
out2 = est_multi_poly(S,yv,k=3,start=0,link=1,disc=1,multi=multi1)

## End(Not run)

## Estimation of different Multidimensional LC IRT models with ordinal responses

### Aggregate data

data(hads)
X = as.matrix(hads)
out = aggr_data(X)
S = out$data_dis
yv = out$freq

### Define matrix to allocate each item to one dimension
multi1 = rbind(c(2,6,7,8,10,11,12),c(1,3,4,5,9,13,14))

### Bidimensional LC Graded Response Model with 3 latent classes
# (free discriminating and free difficulty parameters)
out1 = est_multi_poly(S,yv,k=3,start=0,link=1,disc=1,multi=multi1)

### Bidimensional LC Partial Credit Model with 3 latent classes
# (constrained discrimination and free difficulty parameters)
out2 = est_multi_poly(S,yv,k=3,start=0,link=2,multi=multi1)

### Bidimensional LC Rating Scale Model with 3 latent classes
# (constrained discrimination and constrained difficulty parameters)
out3 = est_multi_poly(S,yv,k=3,start=0,link=2,difl=1,multi=multi1)

## End(Not run)
```
## Not run:
## Estimation of LC model with covariates
# generate covariates
be = c(0,1,-1)
X = matrix(rnorm(2000),1000,2)
U = cbind(1,X)
p = exp(u)/(1+exp(u))
c = 1+(runif(1000)<p)
Y = matrix(0,1000,5)
a = c(0.3,0.7)
for(i in 1:1000) Y[i,] = runif(5)<la[c[i]]
# fit the model with k=2 and k=3 classes
out1 = est_multi_poly(Y,k=2,X=X)
out2 = est_multi_poly(Y,k=3,X=X)
# fit model with k=2 and k=3 classes in fortran
out3 = est_multi_poly(Y,k=2,X=X,fort=TRUE)
out4 = est_multi_poly(Y,k=3,X=X,fort=TRUE)
## End(Not run)
```

**hads**

Dataset about measurement of anxiety and depression in oncological patients

**Description**

This data set contains the responses of 201 oncological patients to 14 ordinal polytomous items that measure anxiety (7 items) and depression (7 items), according to the Hospital Anxiety and Depression Scale questionnaire.

**Usage**

```
data(hads)
```

**Format**

A data frame with 201 observations on 14 items:

- `item1` measure of depression
- `item2` measure of anxiety
- `item3` measure of depression
- `item4` measure of depression
- `item5` measure of depression
- `item6` measure of anxiety
- `item7` measure of anxiety
- `item8` measure of anxiety
inv_glob

item9 measure of depression
item10 measure of anxiety
item11 measure of anxiety
item12 measure of anxiety
item13 measure of depression
item14 measure of depression

Details

All items have 4 response categories: the minimum value 0 corresponds to a low level of anxiety or depression, whereas the maximum value 3 corresponds to a high level of anxiety or depression.

References


Examples

data(hads)
## maybe str(hads)
str(hads)

---

inv_glob Invert marginal logits

Description

Function used within est_multi_glob to invert marginal logits and fit the marginal regression model; this is an internal function.

Usage

inv_glob(eta, type = "g", der = F)

Arguments

eta vector of logits
type type of logit (l = local-logits, g = global-logits)
der indicator that the derivative of the canonical parameters with respect to the vector of marginal logits is required (F = not required, T = required)

Value

p vector of probabilities
D derivative of the canonical parameters with respect to the vector of marginal logits (if der = T)
Matrices to compute generalized logits

Description

It provides the matrices used to compute a vector of generalized logits on the basis of a vector of probabilities according to the formula $C_0 \log(M_0 p)$; this is an internal function.

Usage

matr_glob(l, type = "g")

Arguments

- l: number of response categories
- type: type of logit (l = local-logits, g = global-logits)

Value

- Co: matrix of contrasts
- Ma: marginalization matrix

Author(s)

Francesco Bartolucci - University of Perugia (IT)

References


**Description**

This dataset contains the responses of a sample of 1510 examinees to 12 binary items on Mathematics. It has been extrapolated from a larger dataset collected in 1996 by the Educational Testing Service within the National Assessment of Educational Progress (NAEP) project.

**Usage**

```r
data(naep)
```

**Format**

A data frame with 1510 observations on the following 12 items:

- **Item1**: round to thousand place
- **Item2**: write fraction that represents shaded region
- **Item3**: multiply two negative integers
- **Item4**: reason about sample space (number correct)
- **Item5**: find amount of restaurant tip
- **Item6**: identify representative sample
- **Item7**: read dials on a meter
- **Item8**: find \((x, y)\) solution of linear equation
- **Item9**: translate words to symbols
- **Item10**: find number of diagonals in polygon from a vertex
- **Item11**: find perimeter (quadrilateral)
- **Item12**: reason about betweenness

**References**


**Examples**

```r
data(naep)
## maybe str(naep)
str(naep)
```
prob_multilogit  

Compute multinomial probabilities

Description
The function compute multinomial probabilities (internal function).

Usage
prob_multilogit(Xdis, be, label, fort=F)

Arguments
- Xdis: matrix of all covariates (two- or three-dimensional)
- be: initial value of the parameter vector
- label: label associated to every covariate configuration
- fort: to use fortran routines when possible

Value
- P: estimated matrix of probabilities
- Pdis: estimated matrix of distinct probabilities

Author(s)
Francesco Bartolucci - University of Perugia (IT)

search.model  

Search for the global maximum of the log-likelihood

Description
It search for the global maximum of the log-likelihood given a vector of possible number of classes to try for.

Usage
search.model(S, yv = rep(1, ns), kv, X = NULL, link = 0, disc = 0, difl = 0, multi = 1:J, fort = FALSE, tol = 10^-10)
Arguments

**S**
- matrix of all response sequences observed at least once in the sample and listed row-by-row (use 999 for missing response)

**yv**
- vector of the frequencies of every response configuration in S

**kv**
- vector of the possible numbers of latent classes

**X**
- matrix of covariates that affects the weights

**link**
- type of link function (1 = global logits, 2 = local logits); with global logits the Graded Response model results; with local logits the Partial Credit results (with dichotomous responses, global logits is the same as using local logits resulting in the Rasch or the 2PL model depending on the value assigned to disc)

**disc**
- indicator of constraints on the discriminating indices (0 = all equal to one, 1 = free)

**difl**
- indicator of constraints on the difficulty levels (0 = free, 1 = rating scale parametrization)

**multi**
- matrix with a number of rows equal to the number of dimensions and elements in each row equal to the indices of the items measuring the dimension corresponding to that row

**fort**
- to use fortran routines when possible

**tol**
- tolerance level for checking convergence of the algorithm as relative difference between consecutive log-likelihoods

Value

**out.single**
- output of each single model (as from est_multi_poly) for each k in kv

**bicv**
- value of BIC index for each k in kv

**lkv**
- value of log-likelihood for each k in kv

Author(s)

Francesco Bartolucci, Silvia Bacci, Michela Gnaldi - University of Perugia (IT)

References


Examples

```r
## Not run:
## Search models (in terms of number of items) of different Multidimensional LC IRT models with binary responses

# Aggregate data
data(naep)
X = as.matrix(naep)
out = aggr_data(X)
```
standard.matrix

standard.matrix

Description

Given a matrix of support points X and a corresponding vector of probabilities piv it computes the mean for each dimension, the variance covariance matrix, the correlation matrix, Spearman correlation matrix, and the standarized matrix Y.

Usage

standard.matrix(X, piv)

Arguments

X matrix of support points for the distribution included row by row
piv vector of probabilities with the same number of elements as the rows of X

Value

mu vector of the means
V variance-covariance matrix
si2 vector of the variances
si vector of standard deviations
Cor Braives-Pearson correlation matrix
Sper Spearman correlation matrix
Y matrix of standardized support points

Author(s)

Francesco Bartolucci, Silvia Bacci, Michela Gnaldi - University of Perugia (IT)

Examples

## Example of standardization of a distribution with randomly generated support points and probabilities
X = matrix(rnorm(100), 20, 5)
piv = runif(20); piv = piv/sum(piv)
out = standard.matrix(X, piv)
### Description

The function tests a certain multidimensional model (restricted model) against a larger multidimensional model based on a higher number of dimensions. A typical example is testing a unidimensional model (and then the hypothesis of unidimensionality) against a bidimensional model. Both models are estimated by \texttt{est_multi_poly}.

### Usage

```r
test_dim(S, yv, k, link = 1, disc = 0, difl = 0, multi0 = 1:J, multi1, tol = 1e-10, disp = FALSE)
```

### Arguments

- **S**: matrix of all response sequences observed at least once in the sample and listed row-by-row (use 999 for missing response)
- **yv**: vector of the frequencies of every response configuration in S
- **k**: number of ability levels (or latent classes)
- **link**: type of link function (1 = global logits, 2 = local logits); with global logits the Graded Response model results; with local logits the Partial Credit results (with dichotomous responses, global logits is the same as using local logits resulting in the Rasch or the 2PL model depending on the value assigned to disc)
- **disc**: indicator of constraints on the discriminating indices (0 = all equal to one, 1 = free)
- **difl**: indicator of constraints on the difficulty levels (0 = free, 1 = rating scale parametrization)
- **multi0**: matrix specifying the multidimensional structure of the restricted model
- **multi1**: matrix specifying the multidimensional structure of the larger model
- **tol**: tolerance level for checking convergence of the algorithm as relative difference between consecutive log-likelihoods
- **disp**: to display intermediate output

### Value

- **out0**: output for the restricted model obtained from \texttt{est_multi_poly}
- **out1**: output for the larger model obtained from \texttt{est_multi_poly}
- **dev**: likelihood ratio statistic
- **df**: number of degrees of freedom of the test
- **pv**: \( p \)-value for the test
- **table**: summary of all the results
test_dim

Author(s)
Francesco Bartolucci, Silvia Bacci, Michela Gnaldi - University of Perugia (IT)

References


Examples

```r
## Computation of the LR statistic testing unidimensionality on HADS data
# Aggregate data
data(hads)
X = as.matrix(hads)
out = aggr_data(X)
S = out$data_dis
yv = out$freq
# Define matrix to allocate each item on one dimension
multi1 = rbind(c(2,6,7,8,1/zero.noslash,11,12),c(1,3,4,5,9,13,14))
# Compare unidimensional vs bidimensional Graded Response models with free discrimination
# and free difficulty parameters
# with less severe tollerance level (to be increased)
out = test_dim(S,yv,k=3,link=1,disc=1,multi1=multi1,tol=5*10^-4)
```

Index

*Topic **Expectation-Maximization algorithm**
  est_multi_poly, 8
*Topic **Fisher-scoring algorithm**
  est_multi_glob, 7
*Topic **classification**
  class_item, 4
*Topic **datasets**
  hads, 11
  naep, 14
*Topic **education**
  naep, 14
*Topic **likelihood ratio testing**
  test_dim, 18
*Topic **maximum likelihood estimation**
  est_multi_glob, 7
  est_multi_poly, 8
*Topic **model selection**
  search.model, 15
*Topic **multivariate statistics**
  standard.matrix, 17
*Topic **package**
  MultiLCIRT-package, 2
*Topic **psychology**
  hads, 11
  aggr_data, 3
  class_item, 4
  est_multi_glob, 7
  est_multi_poly, 8
  est_multilogit, 6
  hads, 11
  inv_glob, 12
  matr_glob, 13
  MultiLCIRT (MultiLCIRT-package), 2

MultLCIRT-package, 2
naep, 14
prob_multilogit, 15
search.model, 15
standard.matrix, 17
test_dim, 18