Package ‘multilevLCA’

May 16, 2023

Type Package
Version 1.2
Date 2023-05-08
Title Estimates and Plots Single-Level and Multilevel Latent Class Models
Description Efficiently estimates single- and multilevel latent class models with covariates, allowing for output visualization in all specifications. For more technical details, see Lyrvall et al (2023) <arXiv:2305.07276>.
Depends R (>= 3.5.0)
Imports Rcpp (>= 1.0.9), magrittr, MASS, dplyr, tidyR, klaR, foreach, parallel
LinkingTo Rcpp, RcppArmadillo
License GPL (>= 2)
LazyData true
NeedsCompilation yes
Author Roberto Di Mari [aut, cre], Johan Lyrvall [aut], Zsuzsa Bakk [ctb], Jennifer Oser [ctb], Jouni Kuha [ctb]
Maintainer Roberto Di Mari <roberto.dimari@unict.it>
Repository CRAN
Date/Publication 2023-05-16 10:30:10 UTC

R topics documented:

multilevLCA-package .................................................................................. 2
dataIEA ........................................................................................................... 3
dataTOY ......................................................................................................... 4
multiLCA ....................................................................................................... 5
plot.multiLCA .............................................................................................. 13

Index 15
multilevLCA-package  Estimates and Plots Single-Level and Multilevel Latent Class Models

Description

Efficiently estimates single- and multilevel latent class models with covariates, allowing for output visualization in all specifications. For more technical details, see Lyrvall et al (2023) <arXiv:2305.07276>.

Details

To estimate latent class models, see `multiLCA`.
To plot latent class models, see `plot.multiLCA`

Author(s)

Roberto Di Mari and Johan Lyrvall.
Maintainer: Roberto Di Mari <roberto.dimari@unict.it>

References


Examples

```r
data = dataIEA
Y = colnames(dataIEA)[4+1:12]

out = multiLCA(data = data, Y = Y, iT = 2)
out
plot(out, horiz = FALSE)
```
Description

Data set from the International Civic and Citizenship Education Study 2016 (Schulz et al., 2018). As part of a comprehensive evaluation of education systems, the IEA conducted surveys in 1999, 2009 and 2016 in school classes of 14-year olds to investigate civic education with the same scientific rigor as the evaluation of more traditional educational skills of language and mathematics. The present study focuses on the third wave of the survey that was conducted in 2016.

Questions regarding citizenship norms in all three waves asked respondents to explain their understanding of what a good adult citizen is or does. The survey then lists a variety of activities for respondents to rate in terms of how important these activities are in order to be considered a good adult citizen. The twelve items range from obeying the law and voting in elections, to protecting the environment and defending human rights.

Covariates included are customary determinants of citizenship norms from the literature at the individual-level of socio-economic measures and country-level measure of gross domestic product (GDP) per capita.

Usage

data("dataIEA")

Format

A data frame with 90221 observations on the following 28 variables.

- ICCS_year: Year of survey
- COUNTRY: Country
- IDSTUD: Study ID
- TOTWGTs: Study weight
- obey: Always obeying the law
- rights: Taking part in activities promoting human rights
- local: Participating in activities to benefit people in the local community
- work: Working hard
- envir: Taking part in activities to protect the environment
- vote: Voting in every national election
- history: Learning about the country’s history
- respect: Showing respect for government representatives
- news: Following political issues in the newspaper, on the radio, on TV, or on the Internet
- protest: Participating in peaceful protests against laws believed to be unjust
- discuss: Engaging in political discussions
party Joining a political party
female Female
books Number of books at home
edexp Educational expectations
ed_mom Mother education
ed_dad Father education
nonnat_born Non-native born
immigrantfam Immigrant family
nonnat_lang Non-native language level
gdp_constant GDP
log_gdp_constant Log GDP
gdp_currentusd GDP in USD
log_gdp_currentusd Log GDP in USD

References

dataTOY

Description
Artificial multilevel data set.

Usage
data("dataTOY")

Format
A data frame with 3000 observations on the following 13 variables.

id_high High-level id
Y_1 Indicator n.1
Y_2 Indicator n.2
Y_3 Indicator n.3
Y_4 Indicator n.4
Y_5 Indicator n.5
Y_6 Indicator n.6
**multiLCA**

Y_7  Indicator n.7  
Y_8  Indicator n.8  
Y_9  Indicator n.9  
Y_10  Indicator n.10  
Z_low  Continuous low-level covariate  
Z_high  Continuous high-level covariate

**References**


---

**multiLCA**  
*Estimates and plots single- and multilevel latent class models*

**Description**

The `multiLCA` function in the `multilevLCA` package estimates single- and multilevel measurement and structural latent class models. Moreover, the function performs two different strategies for model selection. Methodological details can be found in Bakk et al. (2022), Bakk and Kuha (2018), and Di Mari et al. (2023).

Different output visualization tools are available for all model specifications. See, e.g., `plot.multiLCA`.

**Usage**

```r
multiLCA(data,  
Y,  
iT,  
id_high = NULL,  
iM = NULL,  
Z = NULL,  
Zh = NULL,  
extout = FALSE,  
dataout = FALSE,  
kmea = TRUE,  
sequential = TRUE,  
umFreeCores = 2,  
maxIter = 1e3,  
tol = 1e-8,  
reord = 1,  
fixedpars = 1,  
NRtol = 1e-6,  
NRmaxit = 100,  
verbose = TRUE)
```
Arguments

- **data**: Input matrix or dataframe
- **Y**: Names of data columns with 0-1 coded items
- **iT**: Number of low-level clusters
- **id_high**: Name of data column with high-level id. Default NULL
- **iM**: Number of high-level clusters. Default NULL
- **Z**: Names of data columns with low-level covariates. Default NULL
- **Zh**: Names of data columns with high-level covariates. Default NULL
- **extout**: Whether to output extensive model and estimation statistics. Default FALSE
- **dataout**: Whether to output the cleaned dataset on which estimation was performed. Default FALSE
- **kmea**: Whether to compute starting values for single-level estimation using $K$-means (TRUE), recommended for stability, or $K$-modes (FALSE). Default TRUE
- **sequential**: Whether to perform sequential (TRUE) or parallelized (FALSE) model selection. Default TRUE
- **numFreeCores**: Number of CPU cores to keep free in parallelized model selection. Default 2
- **maxIter**: Maximum number of iterations for EM algorithm in single-level model estimation. Default 1e3
- **tol**: EM tolerance for measurement model estimation. Default 1e-8
- **reord**: Whether to (re)order classes (1) in decreasing order according to probability of scoring yes on all items, or not (0). Default 1
- **fixedpars**: Estimator of multilevel model; one-step estimator (0), two-step estimator (1) or two-stage estimator (2). Default 1
- **NRtol**: Newton-Raphson tolerance for structural model estimation. Default 1e-6
- **NRmaxit**: Maximum number of iterations for Newton-Raphson algorithm in multilevel model estimation. Default 100
- **verbose**: Whether to print model selection results. Default TRUE

Details

To directly estimate a latent class model, iT and (optionally) iM should be specified as a single positive integer. To perform model selection over a set of number of classes, iT and/or (optionally) iM can be specified as a range of consecutive positive integers of the form iT_min:iT_max and iM_min:iM_max, respectively. It is possible to specify iT = iT_min:iT_max and iM = iM_min:iM_max with iT equal to a single positive integer, or iT = iT_min:iT_max with iM = iM_min:iM_max. All model selection procedures print the results and returns the output of the optimal model based on BIC.

In the case where both iT and iM are defined as a range of consecutive positive integers, model selection can be performed using the sequential three-stage approach (Lukociene et al., 2010) or a simultaneous approach. The choice between the two is indicated using the sequential argument, with default TRUE for sequential model selection. The sequential approach involves (first step:) estimating iT_min:iT_max single-level models and identifying the optimal alternative iT_opt1 based
on BIC, (second step:) estimating \( i_{M_{\min}}:i_{M_{\max}}|i_T = i_{T_{\text{opt1}}} \) multilevel models and identifying the optimal alternative \( i_{M_{\text{opt2}}} \) based on high-level BIC, and finally (third step:) estimating \( i_{T_{\min}}:i_{T_{\max}}|i_M = i_{M_{\text{opt2}}} \) multilevel models and identifying the optimal alternative \( i_{T_{\text{opt3}}} \) based on low-level BIC. The simultaneous approach involves devoting multiple CPU cores on the local machine to estimate all combinations in \( i_T = i_{T_{\min}}:i_{T_{\max}}, i_M = i_{M_{\min}}:i_{M_{\max}} \) and identifying the optimal alternative based on low-level BIC. The number of CPU cores to keep free during estimation is indicated using the \( \text{numFreeCores} \) argument with default 2.

**Value**

Single-level measurement model estimation returns (if \( \text{extout} = \text{TRUE} \), a subset):

- \( \text{vPg} \) Class proportions
- \( \text{mPhi} \) Conditional response probabilities
- \( \text{alphas} \) Intercept parameters in logistic models for class proportions
- \( \text{gammas} \) Intercept parameters in logistic models for response probabilities
- \( \text{parvec} \) Vector of all model parameters
- \( \text{Infomat} \) Expected information matrix
- \( \text{Varmat} \) Variance-covariance matrix
- \( \text{SEs} \) Standard errors
- \( \text{eps} \) Epsilon: difference between the last two elements of the EM log-likelihood sequence
- \( \text{mU} \) Posterior class membership probabilities
- \( \text{mU\_modal} \) Modal class assignments as matrix
- \( \text{vU\_modal} \) Modal class assignments as vector
- \( \text{mClassErr} \) Expected number of classification errors
- \( \text{mClassErrProb} \) Expected proportion of classification errors
- \( \text{AvgClassErrProb} \) Average expected proportion of classification errors
- \( \text{R2entr} \) Entropy \( R^2 \)
- \( \text{AIC} \) Akaike Information Criterion
- \( \text{BIC} \) Bayesian Information Criterion
- \( \text{iter} \) Number of iterations of EM algorithm
- \( \text{LLKSeries} \) Full log-likelihood series of EM algorithm
- \( \text{mScore} \) Individual contributions to log-likelihood score
- \( \text{spec} \) Specification

Single-level structural model estimation returns (if \( \text{extout} = \text{TRUE} \), a subset):

- \( \text{mPg} \) Class proportions conditional on covariates
- \( \text{vPg\_avg} \) Conditional class proportions averaged over units
- \( \text{mPhi} \) Conditional response probabilities
<table>
<thead>
<tr>
<th>term</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gammas</td>
<td>Intercept parameters in logistic models for response probabilities</td>
</tr>
<tr>
<td>betas</td>
<td>Intercept and slope parameters in logistic models for conditional class membership</td>
</tr>
<tr>
<td>parvec</td>
<td>Vector of all model parameters</td>
</tr>
<tr>
<td>mV2</td>
<td>Inverse of the information matrix from the regression stage</td>
</tr>
<tr>
<td>mQ</td>
<td>Matrix of cross-derivatives for the asymptotic standard error correction in two-step estimation (see Bakk &amp; Kuha, 2018; Di Mari et al., 2023)</td>
</tr>
<tr>
<td>Varmat_unc</td>
<td>Uncorrected variance-covariance matrix</td>
</tr>
<tr>
<td>Varmat_cor</td>
<td>Corrected variance-covariance matrix</td>
</tr>
<tr>
<td>SEs_unc</td>
<td>Uncorrected standard errors</td>
</tr>
<tr>
<td>SEs_cor</td>
<td>Corrected standard errors</td>
</tr>
<tr>
<td>SEs_cor_beta</td>
<td>Corrected standard errors for beta</td>
</tr>
<tr>
<td>eps</td>
<td>Epsilon: difference between the last two elements of the EM log-likelihood sequence</td>
</tr>
<tr>
<td>mU</td>
<td>Posterior class membership probabilities</td>
</tr>
<tr>
<td>mClassErr</td>
<td>Expected number of classification errors</td>
</tr>
<tr>
<td>mClassErrProb</td>
<td>Expected proportion of classification errors</td>
</tr>
<tr>
<td>AvgClassErrProb</td>
<td>Average expected proportion of classification errors</td>
</tr>
<tr>
<td>R2entr</td>
<td>Entropy $R^2$</td>
</tr>
<tr>
<td>AIC</td>
<td>Akaike Information Criterion</td>
</tr>
<tr>
<td>BIC</td>
<td>Bayesian Information Criterion</td>
</tr>
<tr>
<td>iter</td>
<td>Number of iterations of EM algorithm</td>
</tr>
<tr>
<td>LLKSeries</td>
<td>Full log-likelihood series of EM algorithm</td>
</tr>
<tr>
<td>spec</td>
<td>Specification</td>
</tr>
</tbody>
</table>

Multilevel measurement model estimation returns (if extout = TRUE, a subset):

<table>
<thead>
<tr>
<th>term</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vOmega</td>
<td>High-level class proportions</td>
</tr>
<tr>
<td>mPi</td>
<td>Conditional low-level class proportions</td>
</tr>
<tr>
<td>mPhi</td>
<td>Conditional response probabilities</td>
</tr>
<tr>
<td>vDelta</td>
<td>Intercept parameters in logistic models for high-level class proportions</td>
</tr>
<tr>
<td>mGamma</td>
<td>Intercept parameters in logistic models for conditional low-level class proportions</td>
</tr>
<tr>
<td>mBeta</td>
<td>Intercept parameters in logistic models for response probabilities</td>
</tr>
<tr>
<td>parvec</td>
<td>Vector of all model parameters</td>
</tr>
<tr>
<td>Infomat</td>
<td>Expected information matrix</td>
</tr>
<tr>
<td>Varmat</td>
<td>Variance-covariance matrix</td>
</tr>
<tr>
<td>SEs</td>
<td>Standard errors</td>
</tr>
</tbody>
</table>
eps
Epsilon: difference between the last two elements of the EM log-likelihood sequence

cPMX
Joint posterior class membership probabilities

cLogPMX
Log of joint posterior class membership probabilities

cPX
Conditional posterior low-level class membership probabilities

cLogPX
Log of conditional posterior low-level class membership probabilities

mSumPX
Posterior high-level class membership probabilities for low-level units after marginalizing over low-level classes

mPW
Posterior high-level class membership probabilities for high-level units

mlogPW
Log of posterior high-level class membership probabilities for high-level units

mPW_N
Posterior high-level class membership probabilities for low-level units

mPMsumX
Posterior low-level class membership probabilities for low-level units after marginalizing over high-level classes

R2entr_low
Low-level entropy $R^2$

R2entr_high
High-level entropy $R^2$

AIC
Akaike Information Criterion

BIClow
Low-level Bayesian Information Criterion

BIChigh
High-level Bayesian Information Criterion

ICL_BIClow
Low-level BIC-type approximation the integrated complete likelihood

ICL_BIChigh
High-level BIC-type approximation the integrated complete likelihood

iter
Number of iterations of EM algorithm

LLKSeries
Full log-likelihood series of EM algorithm

vLLK
Current log-likelihood for high-level units

mScore
Individual contributions to log-likelihood score

spec
Specification

Multilevel structural model with low-level covariates estimation returns (if extout = TRUE, a subset):

vOmega
High-level class proportions

mPi
Low-level class proportions conditional on high-level class membership and covariates

mPi_avg
Conditional low-level class proportions averaged over low-level units

mPhi
Conditional response probabilities

vDelta
Intercept parameters in logistic models for high-level class proportions

cGamma
Intercept and slope parameters in logistic models for conditional low-level class membership

mBeta
Intercept parameters in logistic models for response probabilities

parvec
Vector of all model parameters

Infomat
Expected information matrix
cGamma_Info: Expected information matrix for gamma
mV2: Inverse of the information matrix from the regression stage
mQ: Matrix of cross-derivatives for the asymptotic standard error correction in two-step estimation (see Bakk & Kuha, 2018; Di Mari et al., 2023)
Varmat_unc: Uncorrected variance-covariance matrix
Varmat_cor: Corrected variance-covariance matrix
SEs_unc: Uncorrected standard errors
SEs_cor: Corrected standard errors
SEs_cor_gamma: Corrected standard errors for gamma
eps: Epsilon: difference between the last two elements of the EM log-likelihood sequence
cPMX: Joint posterior class membership probabilities
cLogPMX: Log of joint posterior class membership probabilities
cPX: Conditional posterior low-level class membership probabilities
cLogPX: Log of conditional posterior low-level class membership probabilities
mSumPX: Posterior high-level class membership probabilities for low-level units after marginalizing over low-level classes
mPW: Posterior high-level class membership probabilities for high-level units
mlogPW: Log of posterior high-level class membership probabilities for high-level units
mPW_N: Posterior high-level class membership probabilities for low-level units
mPMsumX: Posterior low-level class membership probabilities for low-level units after marginalizing over high-level classes
R2entr_low: Low-level entropy $R^2$
R2entr_high: High-level entropy $R^2$
AIC: Akaike Information Criterion
BIClow: Low-level Bayesian Information Criterion
BIChigh: High-level Bayesian Information Criterion
ICL_BIClow: Low-level BIC-type approximation the integrated complete likelihood
ICL_BIChigh: High-level BIC-type approximation the integrated complete likelihood
iter: Number of iterations of EM algorithm
LLKSeries: Full log-likelihood series of EM algorithm
vLLK: Current log-likelihood for high-level units
mScore: Individual contributions to log-likelihood score
mGamma_Score: Individual contributions to log-likelihood score for gamma
spec: Specification
Multilevel structural model with low- and high-level covariates estimation returns (if extout = TRUE, a subset):
mOmega: High-level class proportions conditional on covariates
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vOmega_avg</td>
<td>High-level class proportions averaged over high-level units</td>
</tr>
<tr>
<td>mPi</td>
<td>Low-level class proportions conditional on high-level class membership and co-variants</td>
</tr>
<tr>
<td>mPi_avg</td>
<td>Conditional low-level class proportions averaged over low-level units</td>
</tr>
<tr>
<td>mPhi</td>
<td>Conditional response probabilities</td>
</tr>
<tr>
<td>mDelta</td>
<td>Intercept and slope parameters in logistic models for conditional high-level class membership</td>
</tr>
<tr>
<td>cGamma</td>
<td>Intercept and slope parameters in logistic models for conditional low-level class membership</td>
</tr>
<tr>
<td>mBeta</td>
<td>Intercept parameters in logistic models for response probabilities</td>
</tr>
<tr>
<td>parvec</td>
<td>Vector of all model parameters</td>
</tr>
<tr>
<td>Infomat</td>
<td>Expected information matrix</td>
</tr>
<tr>
<td>cDelta_Info</td>
<td>Expected information matrix for delta</td>
</tr>
<tr>
<td>cGamma_Info</td>
<td>Expected information matrix for gamma</td>
</tr>
<tr>
<td>mV2</td>
<td>Inverse of the information matrix from the regression stage</td>
</tr>
<tr>
<td>mQ</td>
<td>Matrix of cross-derivatives for the asymptotic standard error correction in two-step estimation (see Bakk &amp; Kuha, 2018; Di Mari et al., 2023)</td>
</tr>
<tr>
<td>Varmat_unc</td>
<td>Uncorrected variance-covariance matrix</td>
</tr>
<tr>
<td>Varmat_cor</td>
<td>Corrected variance-covariance matrix</td>
</tr>
<tr>
<td>SEs_unc</td>
<td>Uncorrected standard errors</td>
</tr>
<tr>
<td>SEs_cor</td>
<td>Corrected standard errors</td>
</tr>
<tr>
<td>SEs_cor_delta</td>
<td>Corrected standard errors for delta</td>
</tr>
<tr>
<td>SEs_cor_gamma</td>
<td>Corrected standard errors for gamma</td>
</tr>
<tr>
<td>eps</td>
<td>Epsilon: difference between the last two elements of the EM log-likelihood sequence</td>
</tr>
<tr>
<td>cPMX</td>
<td>Joint posterior class membership probabilities</td>
</tr>
<tr>
<td>cLogPMX</td>
<td>Log of joint posterior class membership probabilities</td>
</tr>
<tr>
<td>cPX</td>
<td>Conditional posterior low-level class membership probabilities</td>
</tr>
<tr>
<td>cLogPX</td>
<td>Log of conditional posterior low-level class membership probabilities</td>
</tr>
<tr>
<td>mSumPX</td>
<td>Posterior high-level class membership probabilities for low-level units after marginalizing over low-level classes</td>
</tr>
<tr>
<td>mPW</td>
<td>Posterior high-level class membership probabilities for high-level units</td>
</tr>
<tr>
<td>mlogPW</td>
<td>Log of posterior high-level class membership probabilities for high-level units</td>
</tr>
<tr>
<td>mPW_N</td>
<td>Posterior high-level class membership probabilities for low-level units</td>
</tr>
<tr>
<td>mPMsumX</td>
<td>Posterior low-level class membership probabilities for low-level units after marginalizing over high-level classes</td>
</tr>
<tr>
<td>R2entr_low</td>
<td>Low-level entropy R^2</td>
</tr>
<tr>
<td>R2entr_high</td>
<td>High-level entropy R^2</td>
</tr>
<tr>
<td>AIC</td>
<td>Akaike Information Criterion</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BIClow</td>
<td>Low-level Bayesian Information Criterion</td>
</tr>
<tr>
<td>BIChigh</td>
<td>High-level Bayesian Information Criterion</td>
</tr>
<tr>
<td>ICL_BIClow</td>
<td>Low-level BIC-type approximation the integrated complete likelihood</td>
</tr>
<tr>
<td>ICL_BIChigh</td>
<td>High-level BIC-type approximation the integrated complete likelihood</td>
</tr>
<tr>
<td>iter</td>
<td>Number of iterations of EM algorithm</td>
</tr>
<tr>
<td>LLKSeries</td>
<td>Full log-likelihood series of EM algorithm</td>
</tr>
<tr>
<td>vLLK</td>
<td>Current log-likelihood for high-level units</td>
</tr>
<tr>
<td>mScore</td>
<td>Individual contributions to log-likelihood score</td>
</tr>
<tr>
<td>mDelta_Score</td>
<td>Individual contributions to log-likelihood score for delta</td>
</tr>
<tr>
<td>mGamma_Score</td>
<td>Individual contributions to log-likelihood score for gamma</td>
</tr>
<tr>
<td>spec</td>
<td>Specification</td>
</tr>
</tbody>
</table>

References


Examples

```R
# Use the artificial data set
data = dataTOY

# Define vector with names of columns with items
Y = colnames(data)[1+1:10]

# Define name of column with high-level id
id_high = "id_high"

# Define vector with names of columns with low-level covariates
Z = c("Z_low")

# Define vector with names of columns with high-level covariates
Zh = c("Z_high")

# Single-level 3-class LC model with covariates
out = multiLCA(data, Y, 3, Z = Z, verbose = FALSE)
out
```
# Multilevel LC model
out = multiLCA(data, Y, 3, id_high, 2, verbose = FALSE)
out

# Multilevel LC model low-level covariates
out = multiLCA(data, Y, 3, id_high, 2, Z, verbose = FALSE)
out

# Multilevel LC model low- and high-level covariates
out = multiLCA(data, Y, 3, id_high, 2, Z, Zh, verbose = FALSE)
out

# Model selection over single-level models with 1-3 classes
out = multiLCA(data, Y, iT = 1:3, verbose = FALSE)
out

# Model selection over multilevel models with 1-3 low-level classes and
# 2 high-level classes
out = multiLCA(data, Y, iT = 1:3, id_high, 2, verbose = FALSE)
out

# Model selection over multilevel models with 3 low-level classes and
# 1-2 high-level classes
out = multiLCA(data, Y, 3, id_high, 1:2, verbose = FALSE)
out

# Model selection over multilevel models with 1-3 low-level classes and
# 1-2 high-level classes using the default sequential approach
out = multiLCA(data, Y, 1:3, id_high, 1:2, verbose = FALSE)
out

---

**plot.multiLCA**

*Plots conditional response probabilities*

**Description**

Visualises conditional response probabilities estimated by the `multiLCA` function. The method works for both single- and multilevel models.

Let `out` denote the list object returned by the `multiLCA` function. Then, executing `plot(out)` visualises the conditional response probabilities given by the `mPhi` matrix in `out`.

**Usage**

```r
## S3 method for class 'multiLCA'
plot(x, horiz = TRUE, clab = NULL, ...)
```
Arguments

- **x**: The object returned by the `multiLCA` function.
- **horiz**: Whether item labels should be oriented horizontally (`TRUE`) or vertically (`FALSE`). Default `TRUE`.
- **clab**: A character vector with user-specified class labels, if available, in the order "Class 1", "Class 2", ..., under the default settings, i.e. top-to-bottom. Default `NULL`.
- ...: Additional plotting arguments.

Value

- No return value.

Examples

```r
# Use IEA data
data = dataIEA

# Define vector with names of columns with items
Y = colnames(data)[4+1:12]

# Define number of (low-level) classes
iT = 3

# Estimate single-level measurement model
out = multiLCA(data = data, Y = Y, iT = iT)
out

# Plot conditional response probabilities with default settings
plot(out)

# Plot with vertical item labels and custom class labels
plot(out, horiz = FALSE, clab = c("Maximal", "Engaged", "Subject"))
```
Index

* datasets
  dataIEA, 3
  dataTOY, 4
* package
  multilevLCA-package, 2

dataIEA, 3
dataTOY, 4

multiLCA, 2, 5, 13, 14
multilevLCA-package, 2

plot.multiLCA, 2, 5, 13