Package ‘mixRasch’

February 20, 2015

Type Package
Title Mixture Rasch Models with JMLE
Version 1.1
Date 2014-02-26
Author John T. Willse
Maintainer John T. Willse <willse@uncg.edu>
Description Estimates Rasch models and mixture Rasch models, including the dichotomous Rasch model, the rating scale model, and the partial credit model.
License GPL (>= 2)
NeedsCompilation no
Repository CRAN
Date/Publication 2014-03-06 19:31:30

R topics documented:
mixRasch-package ................................................. 2
exRasch ............................................................... 2
getEstDetails ..................................................... 3
getItemDetails .................................................... 4
itemFitPlot ........................................................ 6
mixRasch .......................................................... 7
personItemPlot ................................................... 9
rICC ................................................................. 10
SimMix .............................................................. 12

Index 13
mixRasch-package  

Contains a function for estimating and evaluating mixture Rasch models using JMLE.

Description

The included function will estimate a mixture Rasch model using joint maximum likelihood estimation (JMLE). The estimation is based on a mixture partial credit model. Step parameters can be constrained to estimate a mixture rating scale model. Estimating a model with only one latent class accomplishes a standard Rasch analysis with JMLE.

Details

Package: mixRasch  
Type: Package  
Version: 1.1  
Date: 2014-02-26  
License: GPL version 2 or newer

This is an early version of the package. It currently contains only the mixRasch function. Please contact the author if you encounter any bugs or if you have questions or suggestions.

Author(s)

John T. Willse <jtwillse@uncg.edu>

References


Example 1-class data

Description

A small example data set for single-class (standard) Rasch model analysis.

Usage

data(exRasch)
**Format**

A data frame with 100 observations on the following 10 variables.

i1  a numeric vector  
\[\text{i2}  \quad \text{a numeric vector}  \]
\[\text{i3}  \quad \text{a numeric vector}  \]
\[\text{i4}  \quad \text{a numeric vector}  \]
\[\text{i5}  \quad \text{a numeric vector}  \]
\[\text{i6}  \quad \text{a numeric vector}  \]
\[\text{i7}  \quad \text{a numeric vector}  \]
\[\text{i8}  \quad \text{a numeric vector}  \]
\[\text{i9}  \quad \text{a numeric vector}  \]
\[\text{i10}  \quad \text{a numeric vector}  \]

**Source**

Simulated to follow a 1-class (standard) Rasch model.

**See Also**

mixRasch

**Examples**

```r
data(exRasch)
```

---

**Description**

This function extracts information about the estimation procedure conducted using the mixRasch procedure.

**Usage**

```r
getEstDetails(raschResult, camelCase=TRUE)
```

**Arguments**

- `raschResult`  The result of a mixRasch analysis.
- `camelCase`  If TRUE, the variables returned are renamed to conform to camelCase style naming conventions.
Details

The function only requires a mixRasch result. The camelCase option is available so results can be exported to other software that might not support some types of R names (e.g., names with "."s in them).

Value

- model: Reports which model was estimated.
- n.c: Number of latent classes.
- iter: The number of completed iterations.
- max.change: The largest model parameter change in each class in the last iteration of the estimation.
- converge.flag: Indicates if convergence was reached.
- run.time: Reports time elapsed during estimation.

Author(s)

John T. Willse

Examples

```r
# Example data included with mixRasch
data(SimMix)

test1 <- mixRasch(SimMix,1,50, conv.crit=.0001, n.c=1)
getEstDetails(test1)
```

---

**getItemDetails**

Convenience function returning information about an individual item.

Description

This function extracts information about an individual item estimated using the mixRasch procedure.

Usage

```r
getItemDetails(raschResult, item, class=1, camelCase=TRUE)
```
**getItemDetails**

### Arguments

- **raschResult**: The result of a mixRasch analysis.
- **item**: Either an integer representing the item position within the analysis or the item name (the column name from the dataframe used in the mixRasch analysis).
- **class**: The class in a mixture Rasch analysis for which you want the item stats. In a standard Rasch analysis, class will always be 1.
- **camelCase**: If TRUE, the variables returned are renamed to conform to camelCase style naming conventions.

### Details

The function only requires a mixRasch result and an item number or name. If the analysis is a mixture analysis, the class from which the result is to be taken must be provided. The camelCase option is available so results can be exported to other software that might not support some types of R names (e.g., names with "."s in them).

### Value

- **item.name**: The name of the item.
- **n.cat**: The number categories (e.g., 2 for dichotomous).
- **delta.i**: Average step difficulty.
- **SE.delta.i**: Standard error for delta.i.
- **tau**: Step difficulty as a deviation from delta.i.
- **SE.tau**: Standard error for tau.
- **infit**: The infit statistic for the item
- **in.Z**: The standardized infit statistic for the item
- **outfit**: The outfit statistic for the item
- **out.Z**: The standardized outfit statistic for the item
- **itemMean**: The item mean, based on modal class.
- **pBis**: The item-theta correlation, based on modal class.
- **bis**: The item-theta polyserial correlation, based on modal class.

### Author(s)

John T. Willse

### Examples

```r
# Example data included with mixRasch
data(SimMix)
test1 <- mixRasch(SimMix, 1, 50, conv.crit=.0001, n.c=1)
getItemDetails(test1, 2)
```
itemFitPlot  

Function for producing item difficulty by fit statistic plots.

Description

This function produces item difficulty and fit statistic plots for Rasch calibrated items estimated using the mixRasch procedure.

Usage

```r
itemFitPlot(raschResult, itemSet, useItemNames = TRUE, 
            fitStat = c("infit", "outfit", "in.Z", "out.Z"), 
            plotTitle = "Item Fit Plot", 
            xlab, ylab, xlim, ylim, refLines, 
            col = c("black", "white"), colTheme, 
            gDevice, file)
```

Arguments

- **raschResult**: A mixRasch object, where n.c = 1 (i.e., no mixture models).
- **itemSet**: An optional vector of item names or item positions for plotting subsets of items.
- **useItemNames**: If TRUE, then item names are used. If false item numbers are used.
- **fitStat**: The item fit statistic ("infit", "outfit", "in.Z", "out.Z") used in the plot.
- **plotTitle**: Controls the main plot title.
- **xlab**: The label for the x axis.
- **ylab**: The label for the y axis.
- **xlim**: A vector overriding default limits for the x axis.
- **ylim**: A vector overriding default limits for the y axis.
- **refLines**: A vector overriding the default lower and upper reference lines that define the region of rejection.
- **col**: A vector of the colors to be used in the plot. The first color will be used for item labels. The second color will be used for shading the area of rejection.
- **colTheme**: Four color themes ("cavaliers", "dukes", "spartans", "greys") are provided. If you provide a color theme, it will override the col parameter.
- **gDevice**: Controls graphics device. Options are "screen" (default), "jpg", or "png".
- **file**: The name of the output file if a device other than "screen" is chosen.

Details

The function produces an item difficulty by item fit plot. Items are plotted using either their names or their test position (see useItemNames). Lower and upper reference line are drawn (see refLines) that help identify misfitting items. For infit and outfit, default reference lines are drawn at .7 and 1.3. For in.Z and out.Z, default reference lines are drawn at -2 and +2. Using col or colTheme, the region beyond the reference line may be shaded.
mixRasch

Author(s)

John T. Willse

Examples

# Example data included with mixRasch
data(exRasch)

rasch1 <- mixRasch(exRasch, 1, 50, conv.crit = .0001, n.c = 1)

# Default Fit Plot
itemFitPlot(rasch1)

# Using Outfit and a color theme
itemFitPlot(rasch1, fitStat = "outfit", colTheme = "spartans")

Description

This function will estimate a mixture Rasch model using joint maximum likelihood estimation (JMLE). The estimation is based on a mixture partial credit model. Step parameters can be constrained to estimate a mixture rating scale model.

Usage

mixRasch(data, steps, max.iter = 50, conv.crit = 0.001, model = "RSM",
          n.c = 1, class, metric, info.fit = TRUE, treat.extreme = 0.3,
          maxchange = 1.5, maxrange = c(-4, 4), as.LCA = FALSE)

Arguments

data A rectangular data set (matrix or data frame) to be analyzed.

steps The maximum number of item thresholds to be estimated. Some items may have
        less than the maximum.

max.iter Maximum number EM iterations

conv.crit Estimation stops when the largest model parameter change is smaller than this
          criterion.

model "RSM" (the default) constrains all step parameters to be equal (i.e., estimates a
        rating scale model). Assumes all items have the same number of steps. "PCM"
        allows step parameters to differ across items (i.e., estimates a partial credit
        model). The number of steps can differ across items.
n.c  Number of latent classes.
class  Optional matrix of starting values for latent class membership.
metric  Not implemented. Will be an optional argument for setting the final scale of the Rasch results.
info.fit  If "True" the information based criteria of fit (AIC, BIC) are estimated.
treat.extreme  Adjustment to perfect response vectors to allow estimation of person parameters. Perfect vectors are not used during item parameter estimation.
maxchange  Limits the change to model parameters in a single iteration. Helps keep estimates reasonable in the first few iterations.
maxrange  Admissible range of item difficulties.
as.LCA  If TRUE, all person parameters are constrained to equal zero. That analysis accomplishes a latent class analysis rather than a mixture Rasch model.

Details
The analyzed data should be a rectangular data file. Missing data are permitted and handled in the usual JMLE manner. Large amounts of missing data will slow down estimation. Basic results on item parameters are returned to the screen. When more than one class is specified (n.c greater than 1) Rasch model parameters and measures of fit are returned in a list, with each class's results returned in a separate element of that list. When n.c = 1 a standard Rasch model analysis is performed.

Value
LatentClass  One element for each latent class. Each LatentClass element is itself a list containing within class results: item statistics (i.stat), person parameters and measures of person fit (person.par), item parameters and measures of item fit (item.par).
max.change  The largest model parameter change in each class in the last iteration of the estimation.
class  A rectangular matrix of the order number of people by number of classes. Values represent the probability that a person (row) belongs to a particular class (column).
iter  The number of completed iterations.
converge.flag  Indicates if convergence was reached.
info.fit  Provides (if requested) AIC, BIC, CAIC, log likelihood, number of estimated parameters, and number of persons used in the estimation.
model  Reports which model was estimated.
removed.items  Provides a vector indicating which, if any, items had to be removed from estimation for having perfect vectors.
run.time  Reports time elapsed during estimation.

Note
Be aware that this function is an early implementation of the procedure. Please contact the author if you encounter any bugs or if you have questions or suggestions.
personItemPlot

Author(s)
John T. Willse

References

Examples

```r
# Example data included with mixRasch
data(SimMix)
test1 <- mixRasch(SimMix, 1, 50, conv.crit = .0001, n.c = 1)
test2 <- mixRasch(SimMix, 1, 500, conv.crit = .0001, n.c = 2)

# Notice that the AIC and BIC are lowest for the 2 class solution
rbind(test1$info.fit, test2$info.fit)

# Notice that the two "difficulty" columns are ordered differently
# The results reflect that the two groups in the data set have a reversed
# scale from one another.
test2
```

personItemPlot Function for producing back-to-back histograms of item and person distributions.

Description
This function will enable you to make back-to-back histograms of the item and person distributions estimated using the mixRasch procedure.

Usage

```r
personItemPlot(raschResult, nBreaks=15, plotTitle="Person Item Histogram", xlab="Relative Frequency", ylab="Ability", col = c("darkgrey", "lightgrey"), colTheme, makeLegend=TRUE, legendLabels=c("items", "people"), legendLoc="bottomleft", gDevice, file)
```

Arguments

- `raschResult`: A mixRasch object, where n.c = 1 (i.e., no mixture models).
- `nBreaks`: The number of bins to be used in creating the histograms.
- `plotTitle`: The title to be used on the plot.
xlab The label for the x axis.

ylab The label for the y axis.

col A vector of the colors to be used in the plot. The first color will be used for items. The second color will be used for people.

colTheme Four color themes ("cavaliers", "dukes", "spartans", "greys") are provided. If you provide a color theme, it will override the col parameter.

makeLegend Controls whether a legend for the is colors used in the plot.

legendLabels A vector that allows for customized legend labels.

legendLoc Allows placement of the legend in different location. Uses the same descriptions as the standard legend function.

gDevice Controls graphics device. Options are "screen" (default), "jpg", or "png".

file The name of the output file if a device other than "screen" is chosen.

Details

The function produces a standard person and item back-to-back histogram. These plots are only appropriate for standard Rasch models (i.e., not multiple-class mixture models). When used with polytomous model results, the item difficulty is based on the overall (average) step difficulty for each polytomous item.

Author(s)

John T. Willse

Examples

# Example data included with mixRasch
data(SimMix)

rasch1 <- mixRasch(SimMix,1,50, conv.crit=.0001, n.c=1)

personItemPlot(rasch1)

rICC Function for producing theoretical and empirical item characteristic curves.

Description

This function produces item characteristic curves for the family of Rasch models.
Usage

rICC(delt, theta, itemVector, xlim, ylim, plotTitle, xlab, ylab,
col = c("black","white"), colTheme, expectedScore=FALSE,
empICC=FALSE, empOnly=FALSE, gDevice, file, ...)

Arguments

delt The threshold difficulties from a dichotomous or polytomous Rasch model.
theta The estimated theta values associated with the analysis and paired with itemVector, only required when requesting an empirical ICC.
itemVector The observed item responses, only required when requesting an empirical ICC.
xlim A vector overriding default limits for the x axis.
ylim A vector overriding default limits for the y axis.
plotTitle Controls the main plot title.
xlab The label for the x axis.
ylab The label for the y axis.
col A vector of the colors to be used in the plot. The first color will be used for item labels. The second color will be used for shading the area of rejection.
colTheme Four color themes ("cavaliers", "dukes", "spartans", "greys") are provided. If you provide a color theme, it will override the col parameter.
expectedScore Flag indicating whether polytomous items are presented as category probabilities (FALSE) or expected item scores (TRUE). Will be set to TRUE when requesting empirical ICC.
empICC Flag indicating whether an empirical ICC should also be produced.
empOnly A flag indicating if ONLY the empirical ICC should be produced.
gDevice Controls graphics device. Options are "screen" (default), "jpg", or "png".
file The name of the output file if a device other than "screen" is chosen.
... Additional parameters passed to the plot command.

Details

The function produces an item characteristic curve plots. Both empirical and theoretical ICCs can be produced.

Author(s)

John T. Willse

Examples

library(mixRasch)
# Example data included with mixRasch
data(exRasch)
raschl <- mixRasch(exRasch, 1, 50, conv.crit=.0001, n.c=1)

# ICC for item 1
rICC(raschl$item.par$delta[,1], raschl$person.par$theta, exRasch[,1], empICC=TRUE, colTheme="cavaliers")

Description
This simulated data contains two groups. The item difficulties for the two groups are reversed. The analysis should make a clear example for those people less familiar with mixture models.

Usage
data(SimMix)

Format
A data frame with 1000 observations on the following 10 variables.

i1 a numeric vector
i2 a numeric vector
i3 a numeric vector
i4 a numeric vector
i5 a numeric vector
i6 a numeric vector
i7 a numeric vector
i8 a numeric vector
i9 a numeric vector
i10 a numeric vector

Source
Simulated to follow a 2-class mixture Rasch model.

See Also
mixRasch

Examples
data(SimMix)
Index

*Topic datasets
   exRasch, 2
   SimMix, 12
*Topic misc
   getEstDetails, 3
   getItemDetails, 4
   itemFitPlot, 6
   mixRasch, 7
   personItemPlot, 9
   rICC, 10
*Topic package
   mixRasch-package, 2

exRasch, 2
getEstDetails, 3
getItemDetails, 4
itemFitPlot, 6
mixRasch, 7
mixRasch-package, 2
personItemPlot, 9
rICC, 10
SimMix, 12