Package ‘eirm’

October 25, 2021

Type  Package
Title  Explanatory Item Response Modeling for Dichotomous and Polytomous Items
Version  0.5
Maintainer  Okan Bulut <bulut@ualberta.ca>
URL  https://github.com/okanbulut/eirm


Depends  lme4, blme, reshape2, optimx
Imports  magrittr, shiny, shinydashboard, shinycssloaders, readxl, ggeffects, ggplot2
Suggests  knitr, rmarkdown
VignetteBuilder  knitr
License  GPL (>= 3)
Encoding  UTF-8
LazyData  true
RoxygenNote  7.1.1
NeedsCompilation  no

Author  Okan Bulut [aut, cre] (<https://orcid.org/0000-0001-5853-1267>)
Repository  CRAN
Date/Publication  2021-10-25 07:20:10 UTC

R topics documented:

- eirm-package ................................................................. 2
- eirm ......................................................... 2
Description

The package eirm provides a set of tools to analyze and visualize dichotomous and polytomous item responses using the explanatory item response modeling (EIRM). The generalized linear mixed modeling (GLMM) framework is used for examining the impact of item-related and person-related variables on responses. The items are repeated measures that are nested underpersons. The model details can be found at de Boeck and Wilson’s (2004) Explanatory Item Response Models - A Generalized Linear and Nonlinear Approach <DOI: 10.1007/978-1-4757-3990-9>.

Author(s)

Okan Bulut <bulut@ualberta.ca>

References


Description

The eirm function estimates explanatory item response models with item-related and person-related covariates. The function requires the data to be in a long format where items are nested within persons. If item responses are polytomous, then the data has to be reformatted using the polyreformat function.
Usage

eirm(
  formula,
  data,
  engine = "lme4",
  na.action = "na.omit",
  weights = NULL,
  mustart = NULL,
  etastart = NULL,
  cov.prior = "wishart",
  fixef.prior = NULL,
  control = glmerControl(optimizer = "optimx", calc.derivs = FALSE, optCtrl =
    list(method = "nlminb", starttests = FALSE, kkt = FALSE))
)

Arguments

  formula  A regression-like formula that defines item responses as a dependent variable
            and explanatory predictors as independent predictors. For example, "response ~
            -1 + predictor1 + predictor2". Use -1 in the formula to avoid the estimation of
            an intercept parameter.
  data     A data frame in a long format where there are multiple rows for each person
            (i.e., nested data). The data should involve a variable that represents item re-
            sponses, a variable that represents persons, and additional predictors to be used
            for 'explaining' item responses.
  engine   Estimation engine with the options of either "lme4" (default) or "blme".
  na.action How missing data should be handled (default: "na.omit").
  weights  Weights to be used in the estimation.
  mustart, etastart  Model specification arguments for glmer. See glmer for details. Not used when
                     engine = "lme4".
  cov.prior A BLME prior or list of priors with the options of "wishart" (default), "in-
             vwishart", "gamma", "invgamma", or NULL to impose a prior over the covari-
             ance of the random effects. Not used when engine = "lme4".
  fixef.prior A BLME prior of family "normal", "t", "horseshoe", or NULL (default) to im-
              pose a prior over the fixed effects. Not used when engine = "lme4".
  control  Control settings for the glmer function in lme4. Note that the optimx package
            is used by default to speed up the estimation. For higher accuracy in the results,
            the default lme4 optimizers can be used.

Value

An eirm-class list that includes the EIRM formula used for the estimation, estimated parameters in
the model, ability estimates for persons and other random effects (if any), and the complete set of
results returned from the glmer function. These results can be used for further analysis and graphics
based on lme4.
Examples

data("VerbAgg")
mod0 <- eirm(formula = "r2 ~ -1 + situ + btype + (1|id)", data = VerbAgg)
print(mod0) # To get easiness parameters
print(mod0, difficulty = TRUE) # To get difficulty parameters
plot(mod0)
mod1 <- eirm(formula = "r2 ~ -1 + situ + btype + mode + (1|id)", data = VerbAgg)
print(mod1) # To get easiness parameters
print(mod1, difficulty = TRUE) # To get difficulty parameters
plot(mod1)

\---

eirmShiny

**Graphical user interface (GUI) of the eirm function**

Description

An interactive Shiny application for running the eirm function. The application allows users to import the (long-format) response data, define the response variable and predictors, and run the estimation, and produce the output, as well as the item-person plot, on the screen.

Usage

eirmShiny()

Examples

```r
## Not run:
eirmShiny()
## End(Not run)
```

\---

marginalplot

**Plots of Estimated Marginal Effects in Explanatory IRT Models**

Description

This function uses \texttt{ggpredict} to calculate marginal effects for explanatory variables in an explanatory IRT model estimated with the \texttt{eirm} function. It returns a plot of estimated probabilities generated by the explanatory IRT model while holding some predictors constant.

Usage

```r
marginalplot(x, predictors, conf.int = 0.95, plot.title = NULL)
```
Arguments

- x: An eirm object returned from the eirm function.
- predictors: Character vector with the names of up to three categorical predictors from the eirm model. The first predictor is plotted on the x-axis; the second predictor is used as a group variable; the third predictor is used as a facet in the plot.
- conf.int: Confidence interval to be used in the plot (default = 0.95 for 95% confidence intervals).
- plot.title: A title to be used in the plot.

Value

A ggplot2 object.

Examples

data("VerbAgg")
mod <- eirm(formula = "r2 ~ -1 + situ + btype + mode + (1|id)", data = VerbAgg)

# Only one predictor
p1 <- marginalplot(mod, predictors = c("situ"))

# Two predictors
p2 <- marginalplot(mod, predictors = c("situ", "btype"))

# All three predictors
p3 <- marginalplot(mod, predictors = c("situ", "btype", "mode"))

Description

This function creates a person-item map for an object returned from the eirm function. The function was modified from plotPImap in package eRm.

Usage

```r
## S3 method for class 'eirm'
plot(
  x,
  difficulty = FALSE,
  sorted = TRUE,
  theta = NULL,
  main = "Person-Item Map",
  latdim = "Latent Dimension",
  pplabel = "Person\nParameter\nDistribution",
  cex.gen = 0.7,
  ...
)
```
Arguments

x An object returned from the eirm function.
difficulty Whether difficulty should be used instead of easiness (default: FALSE).
sorted Whether the parameters should be sorted in the plot (default: TRUE).
theta A vector of estimated theta values. If NULL, then theta values are obtained from the estimated eirm model. It might be better to save the theta values from a baseline model (e.g., Rasch) and use them when creating a person-item map.
main Main title for the person-item map.
latdim Label of the x-axis, i.e., the latent dimension.
pplabel Title for the upper panel displaying the person parameter distribution.
cex.gen A numerical value giving the amount by which plotting text and symbols should be magnified relative to the default. Here cex.gen applies to all text labels. The default is 0.7.
... Other plot-related arguments.

Value

A person-item map.

Examples

data("VerbAgg")
mod0 <- eirm(formula = "r2 ~ -1 + situ + btype + (1|id)", data = VerbAgg)
plot(mod0)
plot(mod0, difficulty = TRUE) # Plot difficulty instead of easiness

Description

This function prepares the data with polytomous item responses for explanatory item response modeling. If the data is already in a long format (i.e., items by person), it only recodes the polytomous responses and creates a new variable to be used for the estimation. If the data is not in the long format, then both reshaping the data into the long format and recoding items can be done simultaneously.

Usage

polyreformat(
  data, 
id.var,
  long.format = FALSE,
  var.name = "item",
  val.name = "resp"
)
Arguments

- **data**: A data frame – either in a wide format where the rows represent persons and columns represent items explanatory variables or in a long format where there are multiple rows for each person (i.e., nested data).
- **id.var**: The variable that represents examinee IDs.
- **long.format**: Whether the data follow a wide format and thus need to be transformed into a long format first (default is FALSE).
- **var.name**: The variable that represents item IDs if the data is already in long format; otherwise this is the of the variable that represents item IDs once the data is transformed into long format.
- **val.name**: The variable that represents item responses if the data is already in long format; otherwise this is the of the variable that represents item responses once the data is transformed into long format.

Value

Reformatted data for explanatory item response modeling.

Examples

```r
data("VerbAgg")
VerbAgg2 <- polyreformat(data=VerbAgg, id.var = "id", long.format = FALSE,
var.name = "item", val.name = "resp")
head(VerbAgg2)
```

**print.eirm**  
*Printing Estimated Parameters for Explanatory IRT Models*

Description

This generic function prints estimated parameters from an eirm object returned from the *eirm* function.

Usage

```r
## S3 method for class 'eirm'
print(x, difficulty = FALSE, ...)
```

Arguments

- **x**: An object returned from the *eirm* function.
- **difficulty**: Whether difficulty should be used instead of easiness (default: FALSE)
- **...**: Other print-related arguments.
Value

Estimated parameters from an eirm object.

Examples

data("VerbAgg")
mod0 <- eirm(formula = "r2 ~ -1 + situ + btype + (1|id)", data = VerbAgg)
print(mod0) # or, just mod0
print(mod0, difficulty = TRUE)

<table>
<thead>
<tr>
<th>testdata</th>
<th>Fabricated Short Quiz with Explanatory Variables</th>
</tr>
</thead>
</table>

Description

This dataset contains examinees’ responses to a short quiz with 10 items. The data set has been fabricated to demonstrate Explanatory Item Response Modeling.

Usage

testdata

Format

A long format data frame containing 1000 examinees’ responses to 10 items and additional variables.

<table>
<thead>
<tr>
<th>person</th>
<th>Examinee ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>item</td>
<td>Item ID</td>
</tr>
<tr>
<td>response</td>
<td>Dichotomous item responses</td>
</tr>
<tr>
<td>gender</td>
<td>Examinees’ gender where F is female and M is male</td>
</tr>
<tr>
<td>itemtype</td>
<td>A variable to define whether the items on the quiz have a visual component</td>
</tr>
</tbody>
</table>

VerbAggWide

<table>
<thead>
<tr>
<th>VerbAggWide</th>
<th>Verbal Aggression Data in a Wide Format</th>
</tr>
</thead>
</table>

Description

The Verbal Aggression data set is a wide-format version of the VerbAgg data set in the lme4 package (Bates, Maechler, Bolker, & Walker, 2015) in R.

Usage

VerbAggWide
VerbAggWide

Format

A data frame with 316 participants and 27 variables.

The original Verbal Aggression data set was in a long format where items are nested within respondents. VerbAggWide is a wide-format version of the original data set. The first three variables are 'id' as the respondent ID, 'Anger' as the respondents' anger scores, and 'Gender' is the respondents' gender (M: male; F: Female). The remaining columns are the respondents' responses to the Verbal aggression items (0: No; 1: Perhaps; 2: Yes). For more information about the data set, see http://bear.soe.berkeley.edu/EIRM/.

Source

Index

* dataset
  testdata, 8
  VerbAggWide, 8

* package
  eirm-package, 2
  eirm, 2, 4–7
  eirm-package, 2
  eirmShiny, 4
  ggpredict, 4
  glmer, 3
  marginalplot, 4
  plot.eirm, 5
  plotPImap, 5
  polyreformat, 2, 6
  print.eirm, 7
  testdata, 8
  VerbAgg, 8
  VerbAggWide, 8