Package 'irtbem2pl'

August 26, 2026

Title Marginalized Bayesian Item Parameter Estimation, 2pl Model IRT

Version 1.0.1

Maintainer Juan Luis Legorreta Torres <jlegorreta2002@yahoo.com.mx>

Description Estimates item parameters of the two-parameter logistic (2PL) model in Item Response Theory (IRT) using the marginal Bayesian modal estimation via the Expectation-Maximization (EM) algorithm. The package calibrates item discrimination and difficulty parameters, yielding results comparable to software like 'BILOG-MG'.

License GPL (>= 2)

URL https://github.com/juanluislegorretatorres/irtbem2pl

Depends R (>= 3.5.0)

Imports IRTBEMM, statmod

NeedsCompilation no

Encoding UTF-8

Language en-US

RoxygenNote 7.3.2

Author Juan Luis Legorreta Torres [aut, cre]

Repository CRAN

Date/Publication 2026-08-26 19:20:02 UTC

Contents

dat01 *Dichotomous Response Dataset for IRT Calibration*

Description

A sample dataset containing binary examination responses used to demonstrate 2PL model parameter estimation.

Usage

```
data(dat01)
```

Format

A data frame with rows representing examinees and 20 columns representing items (ITEM001 to ITEM023). Responses are dichotomous where 1 indicates a correct response and 0 indicates an incorrect response.

irt2pl *Estimation of the 2PL Model via Marginal Bayesian Modal Estimation*

Description

Estimates item discrimination and difficulty parameters for the two-parameter logistic (2PL) model using an Expectation-Maximization (EM) algorithm with Bayesian priors.

Usage

```
irt2pl(
  Model = Model,
  data = data,
  data.simple = data.simple,
  CountNum = CountNum,
  n.class = n.class,
  Prior = Prior,
  Par.est0 = Par.est0,
  Par.SE0 = Par.SE0,
  D = D,
  np,
  Tol = Tol,
  max.ECycle = max.ECycle,
  max.MCycle = max.MCycle,
  n.Quadpts = n.Quadpts,
  n.decimal = n.decimal,
  Theta.lim = Theta.lim,
```

```
  Missing = Missing,
  ParConstraint = ParConstraint,
  BiasSE = BiasSE,
  I = I,
  J = J,
  Time.Begin = Time.Begin
)
```

Arguments

`Model`	An object specifying the IRT model configuration.
`data`	A data frame or matrix containing the raw response data.
`data.simple`	A simplified or matrix version of the response data.
`CountNum`	A numeric vector indicating the frequency of each response pattern.
`n.class`	An integer specifying the number of latent classes or groups.
`Prior`	A list containing the prior distributions for parameters A and B.
`Par.est0`	A list with the initial parameter estimates for the items.
`Par.SE0`	A list with the initial standard error estimates.
`D`	A numeric scaling constant (typically 1.7 or 1.0).
`np`	An integer representing the number of items or parameters.
`Tol`	A numeric value specifying the convergence tolerance limit.
`max.ECycle`	An integer indicating the maximum number of E-steps.
`max.MCycle`	An integer indicating the maximum number of M-steps.
`n.Quadpts`	An integer specifying the number of quadrature points.
`n.decimal`	An integer for decimal formatting in output objects.
`Theta.lim`	A numeric vector of length 2 defining the boundaries of theta.
`Missing`	A value or matrix specifying how missing data is handled.
`ParConstraint`	A logical value indicating if parameter constraints are applied.
`BiasSE`	A logical value specifying if bias-corrected standard errors are computed.
`I`	An integer representing the total number of examinees.
`J`	An integer representing the total number of items.
`Time.Begin`	A POSIXct timestamp marking the start of the execution.

Value

A list containing the calibrated item parameters, information matrices, and convergence metadata.

`irtbem2pl` *Calibrate 2PL Model via Marginal Bayesian Modal Estimation*

Description

' Estimates item parameters of the two-parameter logistic (2PL) model using marginal Bayesian modal estimation via the Expectation-Maximization (EM) algorithm. The results are similar to those of "BILOG-MG", from the book Item Response Theory Parameter Estimation Techniques.

Usage

```
irtbem2pl(
  data,
  PriorA = c(0, 0.25),
  PriorB = c(0, 4),
  InitialA = 1,
  InitialB = 0,
  Tol = 0.01,
  max.ECycle = 50L,
  max.MCycle = 10L,
  n.decimal = 5L,
  n.Quadpts = 50L,
  Theta.lim = c(-4, 4),
  Missing = -9,
  ParConstraint = FALSE,
  BiasSE = FALSE,
  D = 1
)
```

Arguments

`data`	A matrix or data.frame consisting of dichotomous data (1 for correct and 0 for wrong response).
`PriorA`	A numeric vector with two hyperparameters: the mean and variance of the log-normal distribution for all a parameters. Defaults to c(0, 0.25).
`PriorB`	A numeric vector with two hyperparameters: the mean and variance of the normal distribution prior for item difficulty (b) parameters. Defaults to c(0, 4).
`InitialA`	A single numeric value or vector specifying initial values for the item discrimination parameters (a). Defaults to 1. It is recommended to use the point-biserial correlation from Classical Test Theory.
`InitialB`	A single numeric value or vector specifying initial values for the difficulty parameters (b). Defaults to 0. It is recommended to use the item difficulty index from Classical Test Theory.
`Tol`	A single numeric value specifying the convergence threshold for E-step cycles. Defaults to 0.01.

max.ECycle	A single integer specifying the maximum number of E-step cycles. Defaults to 50L.
max.MCycle	A single integer specifying the maximum number of M-step cycles. Defaults to 10L.
n.decimal	A single integer specifying the number of decimal places for the output results. Defaults to 5L.
n.Quadpts	A single integer specifying the number of quadrature points per dimension (must be larger than 5). Defaults to 50L.
Theta.lim	A numeric vector of length 2 defining the integration grid range for each dimension. Defaults to c(-4, 4).
Missing	A single numeric value indicating missing elements. Defaults to -9. Cannot be 0 or 1.
ParConstraint	A logical value indicating whether to restrict estimates within a reasonable range. Defaults to FALSE.
BiasSE	A logical value determining whether to estimate SEs directly from the inverted Hessian matrix. Defaults to FALSE.
D	A numeric scale constant (1 or 1.702). Defaults to 1.

Details

The two-parameter logistic (2PL) model proposed by Birnbaum (1968) is defined as:

$$P(x = 1|\Theta, a, b) = \frac{1}{1 + exp(-D \cdot a \cdot (\Theta - b))}$$

where x=1 is the correct response, Theta is the examinee's ability, a is the item discrimination, and b is the difficulty parameter.

Value

A list containing the following components:

- Est.ItemPars A data frame with estimates of parameters a and b, along with their standard errors.
- Est.Theta A data frame with EAP ability estimates and their standard errors.
- Loglikelihood The final log-likelihood value.
- Iteration The number of iterations until convergence.
- EM.Map The parameter estimation history across iterations.
- fits.test Model fit statistics including G2, AIC, BIC, and RMSEA.
- Elapsed.time Total execution time of the program.
- InitialValues The initial values used for item calibration.

Author(s)

Juan Luis Legorreta Torres <jlegorreta2002@yahoo.com.mx>

References

Baker, F. B., & Kim, S.-H. (2004). *Item Response Theory: Parameter Estimation Techniques* (2nd ed.). Marcel Dekker.

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. M. Lord & M. R. Novick (Eds.), *Statistical theories of mental test scores* (pp. 395-479). Addison-Wesley.

Examples

```
data(dat01)

# It is recommended to use:
# InitialA = point-biserial correlation (Classical Test Theory)
# InitialB = item difficulty index (Classical Test Theory)

mod_2PL <- irtbem2pl(data = dat01, PriorA = c(0, 0.25), PriorB = c(0, 4),
                     InitialA = 1, InitialB = 0, Tol = 0.01, max.ECycle = 50L,
                     max.MCycle = 10L, n.decimal = 5L, n.Quadpts = 50L,
                     Theta.lim = c(-4, 4), Missing = -9, ParConstraint = FALSE,
                     BiasSE = FALSE, D = 1)

mod_2PL$Est.ItemPars       # show item estimates
mod_2PL$Est.Theta          # show ability estimates
```

---

irtchek2pl *Checking user specifiied input variables*

Description

Verifies whether the input variables provided by the user are correct based on the 2PL model specifications. If acceptable, the function formats and returns them as a combined list; otherwise, it throws an error

Usage

```
irtchek2pl(
  Model = "2PL",
  data,
  PriorA = c(0, 0.25),
  PriorB = c(0, 4),
  InitialA = 1,
  InitialB = 0,
  Tol = 0.01,
  max.ECycle = 50L,
  max.MCycle = 10L,
  n.decimal = 5L,
  n.Quadpts = 50L,
```

```
  Theta.lim = c(-4, 4),
  Missing = -9,
  ParConstraint = FALSE,
  BiasSE = FALSE,
  D = 1
)
```

Arguments

Model	A character string declaring the type of item model (e.g., "2PL").
data	A matrix or data.frame consisting of dichotomous data (1 for correct and 0 for wrong response), with missing data coded as in Missing.
PriorA	A numeric vector or matrix specifying the log-normal prior distribution hyperparameters (mean and variance) for item discrimination. Defaults to c(0, 0.25).
PriorB	A numeric vector or matrix specifying the normal prior distribution hyperparameters (mean and variance) for item difficulty. Defaults to c(0, 4).
InitialA	A single numeric value or vector specifying initial values for item discrimination parameters. Defaults to 1.
InitialB	A single numeric value or vector specifying initial values for item difficulty parameters. Defaults to 0.
Tol	A single numeric value specifying the convergence threshold for E-step cycles. Defaults to 0.01.
max.ECycle	A single integer specifying the maximum number of E-step cycles. Defaults to 50L.
max.MCycle	A single integer specifying the maximum number of M-step cycles. Defaults to 10L.
n.decimal	A single integer specifying the number of decimal places for output formatting. Defaults to 5L.
n.Quadpts	A single integer specifying the number of quadrature points per dimension. Defaults to 50L.
Theta.lim	A numeric vector of length 2 defining the integration grid boundaries. Defaults to c(-4, 4).
Missing	A single numeric value indicating missing elements. Defaults to -9. Cannot be 0 or 1.
ParConstraint	A logical value indicating whether parameters are constrained within a reasonable range. Defaults to FALSE.
BiasSE	A logical value determining whether to estimate SEs directly from the inverted Hessian matrix. Defaults to FALSE.
D	A numeric scaling constant (1 or 1.702). Defaults to 1.

Value

A combined list containing formatted data arrays, sample sizes, prior specifications, initial parameter values, and convergence controls.

Author(s)

Juan Luis Legorreta Torres <jlegorreta2002@yahoo.com.mx>

References

Item Response Theory Parameter Estimation Techniques Second Edition, Revised and Expanded Frank B. Baker University ofWisconsin Madison, Wisconsin, U.S.A. Seock-Ho Kim The University ofGeorgia Athens, Georgia, u.S.A.

Examples

```
data(dat01)
library(irtbem2pl)
 Checking_2PL<-irtchek2pl(Model = "2PL", dat01, D = 1)
```

`LikelihoodInfo2pl` *Compute Likelihood Information for the 2PL IRT Model*

Description

' Calculates the log-likelihood, deviance, and expected frequencies (f, fz, rz) required for the E-step of the Expectation-Maximization (EM) algorithm under a two-parameter logistic (2PL) model.

Usage

```
LikelihoodInfo2pl(
  data.simple,
  CountNum,
  Model,
  Par.est0,
  n.Quadpts,
  node.Quadpts,
  weight.Quadpts,
  D
)
```

Arguments

`data.simple`	A numeric matrix representing the simplified unique response patterns.
`CountNum`	A numeric vector containing the frequencies of each unique response pattern.
`Model`	A character string declaring the type of IRT model to be evaluated (e.g., "2PL").
`Par.est0`	A list containing current item parameter estimates (A and B).
`n.Quadpts`	A single integer indicating the number of quadrature points.
`node.Quadpts`	A numeric vector containing the nodes of the quadrature grid.
`weight.Quadpts`	A numeric vector containing the weights of the quadrature grid.
`D`	A numeric scaling constant (typically 1 or 1.702).

Value

A list containing the following statistical components:

- `LH` The log-likelihood value of the model given the data.
- `Deviance` The model deviance statistic calculated as -2 * LH.
- `f` A numeric matrix of expected frequencies per quadrature point.
- `fz` A numeric matrix of expected correct response frequencies based on P*.
- `rz` A numeric matrix of expected correct joint frequencies.

`Prob.model` *Compute Item Response Probabilities for Various IRT Models*

Description

Calculates the probability of a correct response for a given latent trait level (Theta) under specified Item Response Theory (IRT) models.

Usage

```
Prob.model(X, Model, Par.est0, D = 1)
```

Arguments

`X`	A numeric value or vector representing the latent trait level (Theta) or quadrature node.
`Model`	A character string specifying the IRT model type. Options include "Rasch", "2PL", "3PL", "4PL", "1PLAG", or "1PLG".
`Par.est0`	A list containing current item parameter estimates (A, B, C, S, Beta, Alpha, Gamma, depending on the model).
`D`	A numeric scaling constant (typically 1 or 1.702). Defaults to 1.

Value

A numeric vector or matrix containing the calculated correct response probabilities, bounded between 1e-12 and (1 - 1e-12) for numerical stability.

Index