

Package: MAT (via r-universe)

October 13, 2026

Type Package

Title Multidimensional Adaptive Testing

Version 2.3.2

Date 2024-06-1

Author Seung W. Choi and David R. King

Maintainer Seung W. Choi <schoi@austin.utexas.edu>

Description Simulates Multidimensional Adaptive Testing using the multidimensional three-parameter logistic model as described in Segall (1996) <doi:10.1007/BF02294343>, van der Linden (1999) <doi:10.3102/10769986024004398>, Reckase (2009) <doi:10.1007/978-0-387-89976-3>, and Mulder & van der Linden (2009) <doi:10.1007/s11336-008-9097-5>.

License GPL (>= 2.10)

Imports Rcpp (>= 1.0.0), methods

LinkingTo Rcpp, RcppArmadillo

LazyLoad yes

NeedsCompilation yes

Repository https://choi-phd.r-universe.dev

Date/Publication 2024-06-02 02:43:24 UTC

RemoteUrl https://github.com/cran/MAT

RemoteRef HEAD

RemoteSha e868d2e1cb96fdf1b6855507d91eb959db717592

Contents

MAT-package	2
MAT	3
sample.ipar	6
simM3PL	7

Index	9
--------------	----------

MAT-package

Multidimensional Adaptive Testing (MAT)

Description

MAT is a package to simulate Multidimensional Adaptive Testing (MAT) for the Multidimensional 3-Parameter Logistic (M3PL) Model as described in Segall (1996), Reckase (2009), and Mulder & van der Linden (2009).

Author(s)

Seung W. Choi and David R. King

Maintainer: Seung W. Choi <s-choi@northwestern.edu>

References

1. Choi, S. W., & King, D. R. (2015). R Package MAT: Simulation of multidimensional adaptive testing for dichotomous IRT models. *Applied Psychological Measurement*, 39(3), 239-240.
2. Segall, D. O. (1996). Multidimensional adaptive testing, *Psychometrika*, 61(2), 331-354
3. van der Linden, W. J. (1999). Multidimensional adaptive testing with a minimum error-variance criterion, *Journal of Educational and Behavioral Statistics*, 24(4), 398-412.
4. Mulder, J., & van der Linden, W. J. (2009). Multidimensional adaptive testing with optimal design criteria for item selection, *Psychometrika*, 74(2), 273-296.
5. Reckase, M. D. (2009). *Multidimensional Item Response Theory*. New York: Springer.

Examples

```
#load sample item parameters containing 180 items measuring three dimensions
data(sample.ipar)
#create a variance-covariance (correlation) matrix
vcv1<-diag(3); vcv1[lower.tri(vcv1,diag=FALSE)]<-c(.5,.6,.7)
#simulate item responses
resp1<-simM3PL(sample.ipar, vcv1, 3, n.simulee = 100)$resp
#specify target content distributions
target.content.dist1<-c(1/3,1/3,1/3)
#content category designations for items
content.cat1<-rep(1:3,rep(60,3))
#simulate multidimensional adaptive testing
MCAT.1<-MAT(sample.ipar,
             resp1,
             vcv1,
             target.content.dist=target.content.dist1,
             content.cat=content.cat1,
             ncc=3,
             p=3,
             selectionMethod="A",
             topN=1,
```

```

selectionType="FISHER",
stoppingCriterion="CONJUNCTIVE",
minNI=10,
maxNI=30)

```

MAT

*Multidimensional Adaptive Testing (MAT)***Description**

MAT is a package to simulate multidimensional adaptive testing for the Multidimensional 3-Parameter Logistic (M3PL) model.

Usage

```

MAT(ipar, resp, cors,
    target.content.dist = NULL, content.cat = NULL, ncc = 1,
    content.order = NULL, p = stop("p is required"),
    selectionMethod = c("D", "A", "C", "R"),
    selectionType = c("FISHER", "BAYESIAN"), c.weights = NA,
    stoppingCriterion = c("CONJUNCTIVE", "COMPENSATORY"),
    topN = 1, minNI = 10, maxNI = 30, minSE = 0.3, D = 1,
    maxIter = 30, conv = 0.001, minTheta = -4, maxTheta = 4,
    plot.audit.trail = TRUE, theta.labels = NULL, easiness = TRUE)

```

Arguments

<code>ipar</code>	a data frame containing M3PL item parameters, specifically $a_1, a_2, \dots, d$, and c
<code>resp</code>	a data frame (that will be converted to a numeric matrix) of item responses, e.g., $R_1, R_2, \dots, R_{180}$
<code>cors</code>	a square matrix of the lower diagonal elements of a variance-covariance (VCV) matrix, including 1's in the main diagonal
<code>target.content.dist</code>	an optional vector of target content distributions summed to 1.0, e.g., $c(0.25, 0.5, 0.25)$
<code>content.cat</code>	an optional vector specifying content designations
<code>ncc</code>	the number of content categories (default=1, i.e., no content balancing)
<code>content.order</code>	an optional vector specifying administration order of content categories, e.g., $c(3, 1, 2)$
<code>p</code>	the number of latent dimensions
<code>selectionMethod</code>	item selection criterion: "D"=D-optimality, "A"=A-optimality, "C"=C-optimality, "R"=Random (default="D")
<code>selectionType</code>	item selection method type: "FISHER"=Fisher information, "BAYESIAN"=adds inverse prior VCV

<code>c.weights</code>	an optional vector of weights of length <code>p</code> when <code>selectionMethod="C"</code>
<code>stoppingCriterion</code>	stopping criterion: "CONJUNCTIVE"=SEs for all dimensions must be met, "COMPENSATORY"=the generalized variance or SEs weighted by <code>c.weights</code> must be met
<code>topN</code>	Randomesque exposure control: selects an item randomly from the top <code>N</code> most informative items (default=1, no exposure control)
<code>minNI</code>	minimum number of items to administer (default=10)
<code>maxNI</code>	maximum number of items to administer (default=30)
<code>minSE</code>	minimum SE for stopping (default=0.3)
<code>D</code>	scaling constant: 1.7 or 1.0 (default=1.0)
<code>maxIter</code>	maximum number of Fisher scoring (default=30)
<code>conv</code>	convergence criterion for Fisher scoring (default=0.001)
<code>minTheta</code>	minimum theta value for plotting (default=-4)
<code>maxTheta</code>	maximum theta value for plotting (default=4)
<code>plot.audit.trail</code>	show CAT audit trail: T or F (default=T)
<code>theta.labels</code>	theta labels for plotting (default=c("Theta 1", "Theta 2", ...))
<code>easiness</code>	logical, T if <code>d</code> is related to the <i>easiness</i> of items per Reckase, F otherwise

Details

The purpose of this function is to simulate multidimensional adaptive testing based on the Multidimensional 3-Parameter Logistic (M3PL) model (Reckase, 2009):

$$P_i(\theta) \equiv P(U_i = 1 | \theta, \mathbf{a}_i, d_i, c_i) \equiv c_i + \frac{1 - c_i}{1 + \exp[-D(\mathbf{a}_i \cdot \theta + d_i)]}$$

where $\mathbf{a}_i$ is a vector of discrimination parameters of item i , θ is a vector of abilities, c_i is a scalar representing the guessing parameter of item i , d_i is a scalar representing the easiness of item i . Thetas are estimated using the Bayesian maximum a posteriori (MAP) estimator and the Fisher scoring method. Three item selection criteria are available: D-optimality, A-optimality, and C-optimality (Segall, 1996; van der Linden, 1999; Mulder & van der Linden, 2009). An option is provided to add the inverse of a prior variance-covariance matrix to the multivariate information matrix (selectionType="BAYESIAN"). The stopping condition can be specified as a conjunctive criterion or a compensatory criterion. Content balancing can be imposed by specifying target content distributions. An exposure control option is provided via the randomesque technique.

Value

Returns a list of class "MAT" with the following components:

<code>call</code>	function call stack
<code>items.used</code>	a matrix of items administered
<code>selected.item.resp</code>	a matrix containing item responses for selected items

<code>ni.administered</code>	a vector of the number of items administered
<code>theta.CAT</code>	a matrix of theta estimates from CAT
<code>se.CAT</code>	a matrix of SE estimates from CAT
<code>theta.history</code>	a matrix of theta history from CAT
<code>se.history</code>	a matrix of SE history from CAT
<code>theta.Full</code>	a matrix of theta estimates based on the full bank
<code>se.Full</code>	a matrix of SE estimates based on the full bank
<code>ipar</code>	a matrix of item parameters
<code>p</code>	the number of latent dimensions

Note

1. The MAT function performs a number of checks to determine if the arguments for content balancing and content ordering have been specified correctly. If the arguments have not been specified correctly, content balancing and/or content ordering will not be used for the simulation. Additionally, a warning message will be printed to the console detailing the misspecification.
2. Content ordering is only available for fixed-length CAT. Namely, to invoke a particular content order, the user must set the minimum number of items equal to the maximum number of items (e.g., `minNI=30` & `maxNI=30`).

Note

requires **MASS**

Author(s)

Seung W. Choi and David R. King

References

1. Segall, D. O. (1996). Multidimensional adaptive testing, *Psychometrika*, 61(2), 331-354
2. van der Linden, W. J. (1999). Multidimensional adaptive testing with a minimum error-variance criterion, *Journal of Educational and Behavioral Statistics*, 24(4), 398-412.
3. Mulder, J., & van der Linden, W. J. (2009). Multidimensional adaptive testing with optimal design criteria for item selection, *Psychometrika*, 74(2), 273-296.
4. Reckase, M. D. (2009). *Multidimensional Item Response Theory*. New York: Springer.

Examples

```
## Not run: MCAT.1<-MAT(ipar1,
                        resp1,
                        vcv1,
                        target.content.dist=target.content.dist1,
                        content.cat=content.cat1,
                        ncc=3,
```

```

p=3,
selectionMethod="A",
topN=1,
selectionType="FISHER",
stoppingCriterion="CONJUNCTIVE",
minNI=10,
maxNI=30)

## End(Not run)

```

sample.ipar

*Sample item parameters***Description**

A sample item parameter file containing 180 Multidimensional 3-PL (M3PL) model.

Usage

```
data(sample.ipar)
```

Format

A data frame with item parameters for 180 items.

a1 the discrimination parameter for theta 1

a2 the discrimination parameter for theta 2

a3 the discrimination parameter for theta 3

d the easiness parameter, $d=-a*b$

c the guessing parameter

Details

First 60 items are primarily loaded on theta 1, second 60 on theta 2, and last 60 on theta 3.

Examples

```
data(sample.ipar)
```

simM3PL*Simulate M3PL item responses*

Description

Simulates item responses according to the Multidimensional 3-Parameter Logistic (M3PL) model

Usage

```
simM3PL(ipar, cors, p, n.simulee = 100, D = 1, easiness = T, seed = NULL)
```

Arguments

ipar	a data frame containing M3PL item parameters, specifically a1, a2, ... , d, and c
cors	a square matrix of the lower diagonal elements of a variance-covariance (VCV) matrix, including 1's in the main diagonal
p	the number of latent dimensions
n.simulee	the number of simulees to generate
D	scaling constant: 1.7 or 1.0 (default=1.0)
easiness	logical, T if d is related to the <i>easiness</i> of items per Reckase, F otherwise
seed	random number seed

Details

This function simulates item responses according to the Multidimensional 3-Parameter Logistic (M3PL) model using the item parameters input to the function. Thetas are drawn from the multivariate standard normal distribution with the population variance-covariance (correlation) matrix input to the function.

Value

call	function call stack
theta	a <i>n.simulee</i> by <i>p</i> matrix of true theta values
resp	a data frame of simulated item responses named "R1", "R2", ...

Author(s)

Seung W. Choi

References

Reckase, M. D. (2009). Multidimensional Item Response Theory. New York: Springer.

Examples

```
data(sample.ipar)
vcv1<-diag(3)
vcv1[lower.tri(vcv1,diag=FALSE)]<-c(.5,.6,.7)
resp1<-simM3PL(sample.ipar, vcv1, 3, n.simulee = 100, seed = 1234)$resp
```

Index

- * **Computerized Adaptive Testing**

- MAT, [3](#)

- * **IRT**

- simM3PL, [7](#)

- * **Psychometrics**

- MAT, [3](#)

- simM3PL, [7](#)

- * **datasets**

- sample.ipar, [6](#)

- * **package**

- MAT-package, [2](#)

MAT, [3](#)

MAT-package, [2](#)

sample.ipar, [6](#)

simM3PL, [7](#)