

Package ‘tmt’

April 1, 2019

Type Package

Title Estimation of the Rasch Model for Multistage Tests

Version 0.1.9-3

Date 2019-03-31

Author Jan Steinfeld [cre, aut] (<<https://orcid.org/0000-0001-9853-8260>>),
Alexander Robitzsch [aut] (<<https://orcid.org/0000-0002-8226-3132>>)

Maintainer Jan Steinfeld <jan.d.steinfeld@gmail.com>

URL <https://github.com/jansteinfeld/tmt>

BugReports <https://github.com/jansteinfeld/tmt/issues>

Description Provides conditional maximum likelihood (CML) estimation of item parameters in multi-stage designs (Zwitser & Maris, 2013, <[doi:10.1007/s11336-013-9369-6](https://doi.org/10.1007/s11336-013-9369-6)>) and CML estimation for conventional designs. Additional features are the likelihood ratio test (Andersen, 1973, <[doi:10.1007/BF02291180](https://doi.org/10.1007/BF02291180)>) and simulation of multistage designs.

License GPL-3

LazyLoad yes

LazyData true

VignetteBuilder knitr

Depends R (>= 3.0)

Encoding UTF-8

NeedsCompilation yes

Suggests roxygen2, eRm, knitr, prettydoc, psychotools, testthat,
rmarkdown, dexterMST

Imports parallel, ggplot2, Rcpp (>= 0.12.0), stats

LinkingTo Rcpp

RoxygenNote 6.1.1

Repository CRAN

Date/Publication 2019-04-01 09:10:03 UTC

R topics documented:

tmt-package	2
tmt_gmc	3
tmt_lrtest	5
tmt_mstdesign	6
tmt_rm	8
tmt_sim	12

Index	15
--------------	-----------

tmt-package	<i>tmt: Estimation of the Rasch Model for Multistage Tests</i>
-------------	----------------------------------------------------------------

Description

Provides conditional maximum likelihood (CML) estimation of item parameters in multistage designs (Zwitser & Maris, 2013, <doi:10.1007/s11336-013-9369-6>) and CML estimation for conventional designs. Additional features are the likelihood ratio test (Andersen, 1973, <doi:10.1007/BF02291180>) and simulation of multistage designs.

Details

In multistage tests different groups of items (blocks) are presented to persons depending on their response behavior to previous item groups. Multistage testing is thus a simple form of adaptive testing. If data is collected on the basis of such a multistage design and the items are estimated using the Conditional Maximum Likelihood (CML) method, Glas (1989) <doi:10.3102/10769986013001045> has shown, that the item parameters are biased. Zwitser and Maris (2013) <doi:10.1007/s11336-013-9369-6> showed in their work, that taking the applied multistage design in consideration and including it in the estimation of the item parameters, the estimation of item parameters is not biased using the CML method. Their proposed solution is implemented in our package.

An application example can be found in the vignette by using the following command in the R console `vignette("introduction_to_tmt")`

logo



Author(s)

Maintainer: Jan Steinfeld <jan.d.steinfeld@gmail.com> (0000-0001-9853-8260)

Authors:

- Alexander Robitzsch <robitzsch@ipn.uni-kiel.de> (0000-0002-8226-3132)

References

- Andersen, E. B. (1973). A goodness of fit test for the Rasch model. *Psychometrika*, 38(1), 123-140.
- Baker, F. B., & Harwell, M. R. (1996). Computing elementary symmetric functions and their derivatives: A didactic. *Applied Psychological Measurement*, 20(2), 169-192. Chicago
- Baker, F. B., & Kim, S. H. (2004). *Item response theory: Parameter estimation techniques*. CRC Press.
- Fischer, G. H., & Molenaar, I. W. (Eds.). (2012). *Rasch models: Foundations, recent developments, and applications*. Springer Science & Business Media.
- Formann, A. K. (1986). A note on the computation of the second-order derivatives of the elementary symmetric functions in the Rasch model. *Psychometrika*, 51(2), 335-339.
- Glas, C.A.W. (1988). The Rasch model and multistage testing. *Journal of Educational Statistics*, 13(1), 45-52.
- Glas, C.A.W. (2016). Maximum-Likelihood Estimation. In van der Linden, W.J. (Ed.), *Handbook of Item Response Theory: Volume two: Statistical tools*. (pp. 197 - 236). New York: CRC Press.
- Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests*. Danmarks pædagogiske institut.
- Verhelst, N.D., Glas, C.A.W. und van der Sluis, A. (1984). Estimation Problems in the Rasch-Model: The Basic Symmetric Functions. *Computational Statistics Quarterly*, 1(3), 245-262.
- Zwitser, R. J., & Maris, G. (2015). Conditional statistical inference with multistage testing designs. *Psychometrika*, 80(1), 65-84.

See Also

Useful links:

- <https://github.com/jansteinfeld/tmt>
- Report bugs at <https://github.com/jansteinfeld/tmt/issues>

Examples

```
tmt:::tmt_ascii()
```

tmt_gmc

Function for the Graphical Model Check

Description

This function is a wrapper that processes the data of the likelihood ratio test for ggplot2. Items can be color coded and also excluded.

Usage

```
tmt_gmc(object, title = "graphical model check", xaxis = NULL,
        yaxis = NULL, lim = NULL, ellipse = FALSE, drop = NULL,
        alpha = 0.05, legendtitle = "split criteria", info = NULL)
```

Arguments

object	object of the function tmt_lrtest
title	of the plot
xaxis	description of the x-axis
yaxis	description of the y-axis
lim	of the plot
ellipse	should confidence-ellipse be plotted
drop	which items should be excluded from the plot
alpha	which alpha should be used for the ellipse
legendtitle	Title of the Legend
info	vector with further information for the Plot with names of submitted items

Author(s)

Jan Steinfeld

Examples

```
#####
# Example of Graphical Model Check
#####
items <- seq(-3,3,length.out = 16)
names(items) <- paste0("i",1:16)
persons = 500
mean = 0
sd = 1
dat <- tmt::sim.rm(theta = persons, b = items, seed = 1234)
dat.rm <- tmt_rm(dat)
dat.lrt <- tmt_lrtest(dat.rm, split = "median")

info <- rep(c("group_a", "group_b"),each = 8)
names(info) <- paste0("i",1:16)

drop <- c("i1", "i18")

#library(ggplot2)
plot <- tmt_gmc(object = dat.lrt,
               ellipse = TRUE,
               info = info,
               drop = drop,
               title = "graphical model check",
               alpha = 0.05,
```

```
legendtitle = "split criteria")
```

tmt_lrtest

Computation of Andersen's Likelihood-Ratio Test

Description

This function applies the Likelihood Ratio Test of Andersen. Note that all persons with raw score equal to "median" are assigned to lower group. It is also allowed to split after "mean" or submit any dichotomous vector.

Usage

```
tmt_lrtest(object, split = "median", cores = NULL, se = TRUE, ...)
```

Arguments

object	it is necessary to submit an object of the function mst or nmst
split	default is the split criteria "median" of the raw score, optional are "mean" or any dichotomous vector
cores	submit integer of cores you would like to apply
se	logical: if true, the standard error is estimated
...	further arguments for the tmt_rm function

Value

List with following entries

data_orig	Submitted data frame with item responses
betapars_subgroup	List of item parameters (difficulty) for each subgroup
se.beta_subgroup	List of standard errors of the estimated item parameters
model	Used model ((mst) for Rasch model with multistage design)
LRvalue	LR-value
df	Degrees of freedoms for the test statistic
pvalue	P-value of the likelihood ratio test
loglik_subgroup	Log-likelihoods for the subgroups
split_subgroup	List of split vector for each subgroup
call	Submitted arguments for the function (matched call)
fitobj	List of objects from subgroup estimation

Author(s)

Jan Steinfeld

References

- Andersen, E. B. (1973). A goodness of fit test for the Rasch model. *Psychometrika*, 38(1), 123-140.
- Fischer, G. H., & Molenaar, I. W. (Eds.). (2012). *Rasch models: Foundations, recent developments, and applications*. Springer Science & Business Media.

See Also[tmt_rm](#)**Examples**

```
# example for tmt_lrtest
#####
# Example Rasch model and Likelihood Ratio Test
#####
dat <- tmt::sim.rm(theta = 100, b = 10, seed = 1111)
dat.rm <- tmt_rm(dat = dat)
dat.lrt <- tmt_lrtest(dat.rm)
summary(dat.lrt)
```

tmt_mstdesign

Function to Translate the mstdesign Syntax

Description

This function translates the multistage design for different purposes.

Usage

```
tmt_mstdesign(mstdesign, options = c("design", "simulation", "blocks",
  "items"))
```

Arguments

mstdesign	definition of desired multistage design
options	vector of required output. 'blocks' = Matrix with the classification of Block and Items. 'simulation' = list of all stages. 'design' = matrix of all branches. 'items' = vector of all Items.

Value

List with following entries

blocks	Matrix which contains each block with its corresponding items
simulation	List of the multistage design. Each element within the list contains a matrix for each stage
design	Matrix of all possible branches
items	Vector of item names
start	Items of the starting block(s)

Author(s)

Jan Steinfeld

Examples

```
# example for tmt_mstdesign
## Not run:
#####
# Example-2
#####
mstdesign <- "
  B1 =~ c(i1, i2, i3, i4, i5)
  B2 =~ c(i6, i7, i8, i9, i10)
  B3 =~ c(i11, i12, i13, i14, i15)
  B4 =~ c(i16, i17, i18, i19, i20)
  B5 =~ c(i21, i22, i23, i24, i25)
  B6 =~ c(i26, i27, i28, i29, i30)

  # define starting Block
  Start == B4

  # define branches
  b1 := Start(0,2) + B2(0,2) + B1(0,5)
  b2 := Start(0,2) + B2(3,5) + B3(0,5)
  b3 := Start(3,5) + B5(0,2) + B3(0,5)
  b4 := Start(3,5) + B5(3,5) + B6(0,5)
"
# -----
# for simulation purposes
tmt_mstdesign(mstdesign, options = "simulation")$simulation

# -----
# summary of the submitted design
tmt_mstdesign(mstdesign, options = "design")$tmtdesign

# -----
# matrix of all blocks with the containing items
tmt_mstdesign(mstdesign, options = "blocks")$tmtblocks
```

```

# -----
# vector of all items
tmt_mstdesign(mstdesign, options = "items")$items

# -----
# list of all four elements
tmt_mstdesign(mstdesign, options = c("design", "simulation", "blocks", "items"))

## End(Not run)

#####
# Example-2
#####
mstdesign <- "
  B1 =~ paste0('i',1:5)
  B2 =~ paste0('i',6:10)
  B3 =~ paste0('i',11:15)
  B4 =~ paste0('i',16:20)
  B5 =~ paste0('i',21:25)
  B6 =~ paste0('i',26:30)

  # define starting Block
  Start == B4

  # define branches
  b1 := Start(0,2) + B2(0,2) + B1
  b2 := Start(0,2) + B2(3,5) + B3
  b3 := Start(3,5) + B5(0,2) + B3
  b4 := Start(3,5) + B5(3,5) + B6
"
designelements <- tmt_mstdesign(mstdesign,
  options = c("design", "simulation", "blocks", "items"))

```

tmt_rm

Estimation (CML) of the Rasch model with or without Multistage-Designs.

Description

The `tmt_rm` function estimates the Rasch model if the data is collected within a multistage design (see Zwitser and Maris, 2015). If no `mstdesign` is submitted to the function, than a simple Rasch model will be estimated.

Usage

```

tmt_rm(dat, mstdesign = NULL, weights = NULL, start = NULL,
  sum0 = TRUE, se = TRUE, optimization = "nlminb", ...)

```


Arguments

dat	a matrix of dichotomous (0/1) data or a list of the function tmt_designsim
mstdesign	Model for the multistage design, if CML should be applied to multistage. If not, leave the default value
weights	is optional for the weights of cases
start	Vector of start values. If no vector is provided, the start values will be automatic generated
sum0	logical: If the item parameters should be normed to 'sum = 0' as recommended by Glas (2016, p. 208). Otherwise sum0=FALSE
se	logical: should the standard error should be estimated?
optimization	character: Per default 'nlnmb' is used but 'optim' is also supported.
...	optional further arguments for optim and nlnmb use control = list() with arguments.

Details

According to Glas (1988) <doi:10.3102/10769986013001045> CML estimation of item parameters is biased if the data is collected in multistage designs and this design is not considered. Zwitser and Maris (2015) <doi:10.1007/s11336-013-9369-6> propose to use an additional design matrix to fragment the elementary-symmetric-function. Their approach is implemented in this package.

Value

List with following entries

betapar	Estimated item difficulty parameters (if sum0=FALSE, than the first item is set to 0)
se.beta	Standard errors of the estimated item parameters
loglik	Conditional log-likelihood of the model
df	Number of estimated parameters
N	Number of Persons
I	Number of items
data_orig	Submitted data frame with item responses
data	Used data frame with item responses
desmat	Design matrix
convergence	Convergence criterion
iterations	Number of iterations
hessian	Hessian-Matrix
model	Used model ((mst) for Rasch model with multistage design)
call	Submitted arguments for the function (matched call)
designelements	If the multistage version is requested, the preprocessed design is returned, otherwise NULL
mstdesign	If the multistage version is requested, the submitted design is returned, otherwise NULL

Author(s)

Jan Steinfeld

References

- Baker, F. B., & Harwell, M. R. (1996). Computing elementary symmetric functions and their derivatives: A didactic. *Applied Psychological Measurement*, 20(2), 169-192. Chicago
- Baker, F. B., & Kim, S. H. (2004). *Item response theory: Parameter estimation techniques*. CRC Press.
- Fischer, G. H., & Molenaar, I. W. (Eds.). (2012). *Rasch models: Foundations, recent developments, and applications*. Springer Science & Business Media.
- Formann, A. K. (1986). A note on the computation of the second-order derivatives of the elementary symmetric functions in the Rasch model. *Psychometrika*, 51(2), 335-339.
- Glas, C.A.W. (1988). The Rasch model and multistage testing. *Journal of Educational Statistics*, 13(1), 45-52.
- Glas, C.A.W. (2016). Maximum-Likelihood Estimation. In van der Linden, W.J. (Ed.), *Handbook of Item Response Theory: Volume two: Statistical tools*. (pp. 197 - 236). New York: CRC Press.
- Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests*. Danmarks pædagogiske institut.
- Verhelst, N.D., Glas, C.A.W. und van der Sluis, A. (1984). Estimation Problems in the Rasch-Model: The Basic Symmetric Functions. *Computational Statistics Quarterly*, 1(3), 245-262.
- Zwitser, R. J., & Maris, G. (2015). Conditional statistical inference with multistage testing designs. *Psychometrika*, 80(1), 65-84.

See Also[tmt_lrtest](#)**Examples**

```
# example for tmt_rm
#####
# Example-1 simple Rasch model
#####
dat <- tmt::sim.rm(theta = 100, b = 10, seed = 1111)
dat.rm <- tmt_rm(dat = dat)
summary(dat.rm)

#####
# Example-1 for multistage-design
#####
mstdesign <- "
  B1 =~ c(i1, i2, i3, i4, i5)
  B2 =~ c(i6, i7, i8, i9, i10)
  B3 =~ c(i11, i12, i13, i14, i15)

# define starting Block
```

```

Start == B2

# define branches
b1 := Start(0,2) + B1
b2 := Start(3,5) + B3
"

items <- seq(-1,1,length.out = 15)
names(items) <- paste0("i",1:15)
persons = 1000
mean = 0
sd = 1
dat <- tmt_sim(mstdesign = mstdesign,
  items = items, persons = persons, mean = mean, sd = sd)
dat.rm <- tmt_rm(dat = dat, mstdesign = mstdesign)
summary(dat.rm)

## Not run:
#####
# Example-2 simple Rasch model
#####
dat <- tmt::sim.rm(theta = 100, b = 10, seed = 1111)
dat.rm <- tmt_rm(dat = dat)
summary(dat.rm)

#####
# Example-2 for multistage-design
#####
# also using 'paste' is possible
mstdesign <- "
  B1 =~ paste0('i',1:5)
  B2 =~ paste0('i',6:10)
  B3 =~ paste0('i',11:15)
  B4 =~ paste0('i',16:20)
  B5 =~ paste0('i',21:25)
  B6 =~ paste0('i',26:30)
# define starting Block
Start == B4
# define branches
b1 := Start(0,2) + B2(0,2) + B1
b2 := Start(0,2) + B2(3,5) + B3
b3 := Start(3,5) + B5(0,2) + B3
b4 := Start(3,5) + B5(3,5) + B6
"

items <- seq(-1,1,length.out = 30)
names(items) <- paste0("i",1:30)
persons = 1000
mean = 0
sd = 1
dat <- tmt_sim(mstdesign = mstdesign,
  items = items, persons = persons, mean = mean, sd = sd)
dat.rm <- tmt_rm(dat = dat, mstdesign = mstdesign)
summary(dat.rm)

```

```
## End(Not run)
```

```
tmt_sim
```

```
Function for the Simulation of Multistage-Designs
```

Description

This function simulates data after a multistage design. The subjects are drawn from a normal distribution with specified mean and standard deviation (default $N(0,1)$). As an additional argument, a seed can also be passed.

Usage

```
tmt_sim(mstdesign = NULL, items = NULL, persons = NULL, mean = 0,
        sd = 1, ...)
```

Arguments

mstdesign	definition of desired multistage design
items	vector of difficulty parameters for each items
persons	amount of persons per starting block
mean	optional mean for person parameter; default = 0
sd	optional sd for person parameter; default = 1
...	further optional arguments like <code>set.seed</code>

Value

List with following entries

data	Matrix with item responses
data_mst	Data frame with item responses and additional a vector of used blocks per person
persons	Generated and used person parameters
mstdesign	Submitted multistage design

Author(s)

Jan Steinfeld

Examples

```
#####
# translate multistage model 1
#####
mstdesign <- "
B1 =~ c(i1, i2, i3, i4, i5)
B2 =~ c(i6, i7, i8, i9, i10)
B3 =~ c(i11, i12, i13, i14, i15)

# define starting Block
Start == B2

# define branches
b1 := Start(0,2) + B1
b2 := Start(3,5) + B3
"

items <- seq(-3,3,length.out = 15)
names(items) <- paste0("i",1:15)

persons = 500
set.seed(1111)
data_1 <- tmt_sim(mstdesign = mstdesign,
  items = items,
  persons = persons,
  mean = 0,
  sd = 1)

#####
# translate multistage model 2
#####
mstdesign <- "
B1 =~ c(i1, i2, i3, i4, i5)
B2 =~ c(i6, i7, i8, i9, i10)
B3 =~ c(i11, i12, i13, i14, i15)
B4 =~ c(i16, i17, i18, i19, i20)
B5 =~ c(i21, i22, i23, i24, i25)
B6 =~ c(i26, i27, i28, i29, i30)

# define starting Block
Start == B4

# define branches
b1 := Start(0,2) + B2(0,2) + B1
b2 := Start(0,2) + B2(3,5) + B3
b3 := Start(3,5) + B5(0,2) + B3
b4 := Start(3,5) + B5(3,5) + B6
"

items <- seq(-3,3,length.out = 30)
names(items) <- paste0("i",1:30)
persons = 500
set.seed(1111)
```

```
data_2 <- tmt_sim(mstdesign = mstdesign,  
  items = items,  
  persons = persons,  
  mean = 0,  
  sd = 1)
```

Index

tmt (tmt-package), 2
tmt-package, 2
tmt_gmc, 3
tmt_lrtest, 4, 5, 10
tmt_mstdesign, 6
tmt_rm, 6, 8
tmt_sim, 12