

# Package ‘shortIRT’

July 23, 2025

**Type** Package

**Title** Procedures Based on Item Response Theory Models for the Development of Short Test Forms

**Version** 0.1.3

**Author** Ottavia M. Epifania <ottavia.epifania@unipd.it> [aut, cre, cph]  
Pasquale Anselmi [aut, ctb]  
Egidio Robusto <egidio.robusto@unipd.it> [ctb]

**Maintainer** Ottavia M. Epifania <ottavia.epifania@unipd.it>

**Description** Implement different Item Response Theory (IRT) based procedures for the development of static short test forms (STFs) from a test. Two main procedures are considered, specifically the typical IRT-based procedure for the development of STF, and a recently introduced procedure (Epifania, Anselmi & Robusto, 2022 <[doi:10.1007/978-3-031-27781-8\\_7](https://doi.org/10.1007/978-3-031-27781-8_7)>). The procedures differ in how the most informative items are selected for the inclusion in the STF, either by considering their item information functions without considering any specific level of the latent trait (typical procedure) or by considering their informativeness with respect to specific levels of the latent trait, denoted as theta targets (the newly introduced procedure). Regarding the latter procedure, three methods are implemented for the definition of the theta targets: (i) theta targets are defined by segmenting the latent trait in equal intervals and considering the midpoint of each interval (equal interval procedure, eip), (ii) by clustering the latent trait to obtain unequal intervals and considering the centroids of the clusters as the theta targets (unequal intervals procedure, uip), and (iii) by letting the user set the specific theta targets of interest (user-defined procedure, udp). For further details on the procedure, please refer to Epifania, Anselmi & Robusto (2022) <[doi:10.1007/978-3-031-27781-8\\_7](https://doi.org/10.1007/978-3-031-27781-8_7)>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Imports** stats, TAM, dplyr, ggplot2

**RoxygenNote** 7.2.3

**Suggests** rmarkdown, sirt, testthat (>= 3.0.0), V8

**Config/testthat/edition** 3

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2024-02-16 11:30:04 UTC

## Contents

bp	2
change_names	3
cut_borders	4
diff_theta	4
eip	5
plot_difference	6
plot_tif	7
uip	8
<b>Index</b>	<b>10</b>

---

bp	<i>Benchmark Procedure</i>
----	----------------------------

---

### Description

Create a Short Test Form (STF) using the typical IRT procedure for shortening test (Benchmark Procedure, BP)

### Usage

```
bp(data, item_par = NULL, seed = 999, starting_theta = NULL, num_item = NULL)
```

### Arguments

data	data.frame, subject $\times$ item matrix containing the accuracy responses
item_par	matrix, two-column matrix containing the item parameters. The first column must contain the difficulty parameters $b_i$ , the second column must contain the discrimination parameters $a_i$ .
seed	integer, define the random seed. Default is 999
starting_theta	vector, define the starting $\theta$ of the subjects. If empty, the $\theta$ values will be estimated from the data
num_item	integer, the number of items to include in the short test form

### Value

A list of length 5:

- item\_stf: data.frame, contains the items included in the STF. The number of rows is equal to the number of items included in the STF. The  $\theta$ -targets and the item information functions of the optimal item for each  $\theta$ -target are reported as well
- summary: data.frame, contains the list of items included in the STF and the test information on both the full-length test and the STF
- info\_stf: list, contains the item information functions of the STF
- info\_full: list, contains the item information functions of the full-length test
- theta: data.frame, contains the starting  $\theta$  and the  $\theta$  estimated with the STF

**Examples**

```
# set a seed to replicate the results
set.seed(999)
# Simulate person and item parameters
true_theta <- rnorm(1000)
b <- runif(30, -3, 3)
a <- runif(30, 0.6, 2)
parameters <- data.frame(b, a)
# simulate data
data <- sirt::sim.raschtype(true_theta, b = b, fixed.a = a)
stf <- bp(data, starting_theta = true_theta, item_par = parameters, num_item = 5)
# check the obtained short test form
stf$item_stf
# check the comparison between the short test form and the full-length test
stf$summary
```

---

change_names	<i>Change column names</i>
--------------	----------------------------

---

**Description**

Change the columns names of a data frame and stores the original column names

**Usage**

```
change_names(data)
```

**Arguments**

data                    data.frame, A data frame

**Value**

A list of length two:

1. a data frame with the original column names and the corresponding new names
2. a data frame with the changed column names

**Examples**

```
# original data frame with 5 columns
data <- data.frame(matrix(1:20, nrow = 4, ncol = 5))
change_names(data)
```

---

 cut\_borders

*Cut borders*


---

### Description

Extract the limits of the intervals obtained from sub setting a vector

### Usage

```
cut_borders(x)
```

### Arguments

x                    numeric/integer vector

### Value

A data frame with two columns. The first column contains the lower bounds of each interval. The second column contains the upper bound of each interval

### Examples

```
x <- seq(-3, 3, length = 5)
groups <- cut(x, 5, include.lowest = TRUE)
boundaries <- cut_borders(groups)
```

---

 diff\_theta

*Difference between  $\theta$ s*


---

### Description

Compute the difference between a starting value of  $\theta$  and the  $\theta$  estimated with the STF

### Usage

```
diff_theta(results, starting_theta = NULL)
```

### Arguments

results            The object obtained from the stf-generating functions

starting\_theta    vector, optional vector of length equal to the number of rows in the original data frame with the true  $\theta$  of the respondents. Default is NULL, such that the  $\theta$  estimated with the full-length test will be considered as the starting  $\theta$

**Value**

A data frame with number of rows equal to the number of respondents and 3 columns, one with the starting/true  $\theta$ , one with the  $\theta$  estimated with the STF, and the difference between the estimated  $\theta$  and the starting/true  $\theta$

**Examples**

```
# set a seed to replicate the results
set.seed(999)
# Simulate person and item parameters
true_theta <- rnorm(1000)
b <- runif(30, -3, 3)
a <- runif(30, 0.6, 2)
parameters <- data.frame(b, a)
# simulate data
data <- sirt::sim.raschtype(true_theta, b = b, fixed.a = a)
stf <- uip(data, starting_theta = true_theta, item_par = parameters, num_item = 5)
# without starting theta
my_diff <- diff_theta(stf)
head(my_diff)
```

---

eip

*Equal Interval Procedure*


---

**Description**

Create a Short Test Form (STF) using the  $\theta$ -target procedure based on the equal segmentation of the latent trait (Equal Interval Procedure, EIP)

**Usage**

```
eip(
  data,
  item_par = NULL,
  seed = 999,
  starting_theta = NULL,
  num_item = NULL,
  theta_targets = NULL
)
```

**Arguments**

data	data.frame, subject $\times$ item matrix containing the accuracy responses
item_par	matrix, two-column matrix containing the item parameters. The first column must contain the difficulty parameters $b_i$ , the second column must contain the discrimination parameters $a_i$ .
seed	integer, define the random seed. Default is 999

starting\_theta vector, define the starting  $\theta$  of the subjects. If empty, the  $\theta$  values will be estimated from the data

num\_item integer, the number of items to include in the short test form

theta\_targets vector, define the specific  $\theta$  targets for the user defined procedure. Might also be the same  $\theta$  target repeated for as many times as the number of items to be included in the short test form

### Value

A list of length 5:

- item\_stf: data.frame, contains the items included in the STF. The number of rows is equal to the number of items included in the STF. The  $\theta$ -targets and the item information functions of the optimal item for each  $\theta$ -target are reported as well
- summary: data.frame, contains the list of items included in the STF and the test information on both the full-length test and the STF
- info\_stf: list, contains the item information functions of the STF
- info\_full: list, contains the item information functions of the full-length test
- theta: data.frame, contains the starting  $\theta$  and the  $\theta$  estimated with the STF

### Examples

```
# set a seed to replicate the results
set.seed(999)
# Simulate person and item parameters
true_theta <- rnorm(1000)
b <- runif(30, -3, 3)
a <- runif(30, 0.6, 2)
parameters <- data.frame(b, a)
# simulate data
data <- sirt::sim.raschtype(true_theta, b = b, fixed.a = a)
stf <- eip(data, starting_theta = true_theta, item_par = parameters, num_item = 5)
# check the obtained short test form
stf$item_stf
# check the comparison between the short test form and the full-length test
stf$summary

# Short test form with cut off values
stf_cutoff <- eip(data, starting_theta = true_theta,
  item_par = parameters, theta_targets = rep(2, 5))
stf_cutoff$item_stf
```

---

plot\_difference

*Plot the difference between  $\theta$ s*

---

### Description

Plot the difference or the absolute difference between the starting  $\theta$  and the  $\theta$  estimated with the STF as a function of different levels of the latent trait

**Usage**

```
plot_difference(difference, type = c("diff", "absolute_diff"), levels = 4)
```

**Arguments**

difference	data.frame, data frame obtained with the function [diff_theta()]
type	character, type of difference, either as is ("diff") or absolute ("absolute_diff"). Default is "diff".
levels	integer, number of levels of the starting $\theta$ Default is 4

**Value**

A ggplot object

**Examples**

```
# set a seed to replicate the results
set.seed(999)
# Simulate person and item parameters
true_theta <- rnorm(1000)
b <- runif(30, -3, 3)
a <- runif(30, 0.6, 2)
parameters <- data.frame(b, a)
# simulate data
data <- sirt::sim.raschtype(true_theta, b = b, fixed.a = a)
stf <- uip(data, starting_theta = true_theta, item_par = parameters, num_item = 5)
# compute the difference between starting theta and that estimated with the stf
my_diff <- diff_theta(stf)
# plot the difference with default number of levels
plot_difference(my_diff, type = "diff")
# plot the absolute difference with 10 levels
plot_difference(my_diff, type = "absolute_diff", levels = 10)
```

---

plot\_tif

*Plot Test Information Functions*

---

**Description**

Plot the test information functions of the short test form (default), of the full length test or of both versions

**Usage**

```
plot_tif(results, tif = c("stf", "full", "both"))
```

**Arguments**

results	The object obtained from the stf-generating functions
tif	character, define the TIF to plot, either "stf" (TIF of the STF), "full", (TIF of the full-length test) or "both" (TIF of both STF and full-length test). Default is "stf"

**Value**

A ggplot object

**Examples**

```
# set a seed to replicate the results
set.seed(999)
# Simulate person and item parameters
true_theta <- rnorm(1000)
b <- runif(30, -3, 3)
a <- runif(30, 0.6, 2)
parameters <- data.frame(b, a)
# simulate data
data <- sirt::sim.raschtype(true_theta, b = b, fixed.a = a)
stf <- uip(data, starting_theta = true_theta, item_par = parameters, num_item = 5)
# plot the test information function of the full-length test
plot_tif(stf, tif = "full")
# plot the test information of the full-length test and of the short test form
plot_tif(stf, tif = "both")
```

---

uip

*Unequal interval procedure*


---

**Description**

Create a Short Test Form (STF) using the  $\theta$ -target procedure based on the unequal segmentation of the latent trait (Unequal Interval Procedure, EIP)

**Usage**

```
uip(data, item_par = NULL, seed = 999, starting_theta = NULL, num_item = NULL)
```

**Arguments**

data	data.frame, subject $\times$ item matrix containing the accuracy responses
item_par	matrix, two-column matrix containing the item parameters. The first column must contain the difficulty parameters $b_i$ , the second column must contain the discrimination parameters $a_i$ .
seed	integer, define the random seed. Default is 999
starting_theta	vector, define the starting $\theta$ of the subjects. If empty, the $\theta$ values will be estimated from the data
num_item	integer, the number of items to include in the short test form



**Value**

A list of length 5:

- `item_stf`: data.frame, contains the items included in the STF. The number of rows is equal to the number of items included in the STF. The  $\theta$ -targets and the item information functions of the optimal item for each  $\theta$ -target are reported as well
- `summary`: data.frame, contains the list of items included in the STF and the test information on both the full-length test and the STF
- `info_stf`: list, contains the item information functions of the STF
- `info_full`: list, contains the item information functions of the full-length test
- `theta`: data.frame, contains the starting  $\theta$  and the  $\theta$  estimated with the STF

**Examples**

```
# set a seed to replicate the results
set.seed(999)
# Simulate person and item parameters
true_theta <- rnorm(1000)
b <- runif(30, -3, 3)
a <- runif(30, 0.6, 2)
parameters <- data.frame(b, a)
# simulate data
data <- sirt::sim.raschtype(true_theta, b = b, fixed.a = a)
stf_uip = uip(data, starting_theta = true_theta, item_par = parameters, num_item = 10)
# check the obtained short test form
stf_uip$item_stf
# check the comparison between the short test form and the full-length test
stf_uip$summary
```

# Index

bp, [2](#)

change\_names, [3](#)

cut\_borders, [4](#)

diff\_theta, [4](#)

eip, [5](#)

plot\_difference, [6](#)

plot\_tif, [7](#)

uip, [8](#)