International Journal of Research Studies in Psychology
2019 Volume 8 Number 2, 81-96

cpm.4.CSE/IRT$^{N\text{=small}}$: A companion to cpm.4.CSE/IRT for N = small

Zendler, Andreas
University of Education Ludwigsburg, Germany (zendler@ph-lzudwigsburg.de)

Received: 2 July 2019 **Revised**: 1 August 2019 **Accepted**: 5 October 2019
Available Online: 24 December 2019 **DOI**: 10.5861/ijrsp.2019.4018

ISSN: 2243-7681
Online ISSN: 2243-769X

OPEN ACCESS

Abstract

cpm.4.CSE/IRT$^{N=small}$ (*c*ompact *p*rocess *m*odel for *C*ompetence *S*cience *E*ducation based on *IRT* for small samples) is a process model for competence measurement based on IRT models that is optimized for small sample sizes. *cpm.4.CSE/IRT/*$^{N=small}$ is a supplementary to *cpm.4.CSE/IRT* and consists of the four sub processes *B1 determine items, B2 test items, B3*$^{N=small}$ *analyze items according to Rasch model*, and *B4 interpret items by criteria*. It is also modeled in IDEF0 and implemented in R. The difference between *cpm.4.CSE/IRT* and *cpm.4.CSE/IRT*$^{N=small}$ is that the processes of *B3*$^{N=small}$ are rearranged, and sub process *b*3.1 *test model assumptions statistically by non-parametric tests* replaces *B3.3 test model assumptions statistically.* With *cpm.4.CSE/IRT*$^{N=small}$ it is possible to develop measuring instruments for computer science education even with small samples (N = small).

Keywords: computer science education; educational process model; item response theory; IRT model; competence-based education; small samples

cpm.4.CSE/IRT$^{N\text{=small}}$: A companion to cpm.4.CSE/IRT for *N* = small

1. Introduction

There is a long history of research in psychology employing designs with *N=small* (Robinson & Foster, 1979). Smith and Little (2018) deal with the merits of *N=small* designs. For IRT models, the sample size has a major impact on the accuracy of (1) estimating parameters and of (2) assessing model fit. Guidelines for sample sizes with respect to the accuracy of estimating parameters are provided by Linacre (1994), Jones, Smith and Talley (2006) as well as De Ayala (2009), emphasizing the importance of the concrete application. For item selection, they recommend about *N* = 100 for the Rasch model (1PL model) (see De Ayala 2009, Chapter 3, Eid & Schmidt, 2014, chapter 4). According to Linacre (1994), 64 to 144 persons should suffice for estimating parameters within 95% confidence intervals with a logit width of 1. Significantly larger samples are required for 2PL and 3PL models, about *N* = 500 for 2PL models, and at least *N* = 1000 for 3PL models (see De Ayala 2009, chapters 5 and 6, respectively), assuming 20 items for 2PL and 3PL models.

Ponocny (2001) as well as Koller and Hatzinger (2013) address the problem of too small sample sizes when assessing model fit accuracy by parametric tests. This is because, the estimated parameters are included to the calculations of the test statistics, that is, inaccuracies in parameter estimation propagate to testing model assumptions. An alternative that reduces the problem of too small sample and that avoids error propagation is available by quasi-exact non-parametric tests. They get along with small sample sizes and do not use estimated parameters in model validation.

For the Rasch model, Ponocny (2002) presents a family of quasi-exact non-parametric tests as an alternative to parametric tests (e.g. Andersen's likelihood ratio test, Martin-Löf test, Wald test) by using Monte Carlo simulations. The principle of such tests is to compare the observed data matrix (data set with the responses of persons to items) with a sample of simulated data matrices in order to calculate statistics to test the most important assumptions of the Rasch model (monotonically increasing item characteristics, local stochastic independence, one-dimensionality, specific objectivity) (see Verhelst, 2008; Koller & Hatzinger, 2013; Verhelst, Hatzinger, & Mair, 2007).

cpm.4.CSE/IRT (*c*ompact *p*rocess *m*odel for *C*ompetence *S*cience *E*ducation based on *IRT* models) is a process model for competence measurement based on IRT models (Zendler, 2018). It allows the efficient development of measuring instruments for computer science education. *cpm.4.CSE/IRT* consists of four sub processes: *B1 determine items, B2 test items, B3 analyze items according to Rasch model*, and *B4 interpret items by criteria*. *cpm.4.CSE/IRT* is modeled in IDEF0, a process modeling language that is standardized and widely used.

cpm.4.CSE/IRT$^{N=small}$ is a supplementary to *cpm.4.CSE/IRT* and consists of the same four sub processes. It is also implemented in R, an open-source software optimized for statistical calculations and graphics that allows users to interact using the web application framework Shiny. The difference between *cpm.4.CSE/IRT* and *cpm.4.CSE/IRT*$^{N=small}$ is that *B3* is rearranged and sub process *B*3.3*test model assumptions statistically* is replaced by sub process *b3.1 test model assumptions statistically by non-parametric tests.* With *b*3.1*, cpm.4.CSE/IRT*$^{N=small}$ can be used to develop measuring instruments for computer science education with small samples more efficiently.

2. Method

For developing *cpm.4.CSE/IRT*$^{N=small}$ the same methods are used as for *cpm.4.CSE/IRT,* especially IDEFO and R. IDEFO (*Integrated Definition for Function Modeling*) is used to visualize the processes of *cpm.4.CSE/IRT*$^{N=small}$ with Functions, Inputs, Outputs, Controls, and Mechanisms (Menzel & Mayer, 2005; see

Figure 1). In more detail IDEFO is described with *cpm.4.CSE/IRT* (see Zendler, 2018).

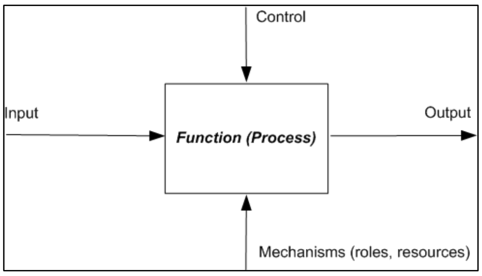


Figure 1. Concepts of IDEF0

R (CRAN, 2019) is used for the implementation of *cpm.4.CRT/IRT*$^{N=small}$. R is an open-source software optimized for statistical calculations and graphics. R is increasingly regarded as the standard language for statistical problems. The standard library of R consists of 29 packages (program libraries), which bundle functions on similar topics. In the subject area of *Psychometrics*, packages are categorized with respect to the theory and method of psychological measurement. The following seven categories are available: (1) *Item Response Theory* (IRT), (2) *Correspondence Analysis* (CA), (3) *Structural Equation Models (*PCA), (4) *Multidimensional Scaling* (MDS), (5) *Classical Test Theory* (CTT), (6) *Knowledge Structure Analysis,* and (7) *Other Related Packages*.

References to non-parametric tests for the Rasch model been have been researched by a three-step procedure: (1) First, the two major international research journals with Rasch model publications were identified: Psychometrika and Rasch Measurement Transactions. (2) In a second step, the identified journals from the year of publication 2000 onwards were researched online with the search terms *non-parametric+Rasch*. (3) Relevant articles were finally searched for relevant secondary literature in a third step.

3. Process model

The process model cpm.4.CSE/IRT$^{N\text{=small}}$ consists of the four sub processes (see Figure 2) that are in principle the same as in cpm.4.CSE/IRT: B1 construct items, B2 test items, B3$^{N\text{=small}}$ analyze items according to Rasch model, and B4 interpret items by criteria. The processes can partially be iterated. For each of the sub processes, the necessary inputs, outputs, conditions (controls), roles and resources are determined. The IDEF0 concept control provides results from the literature that should be used to carry out the processes.

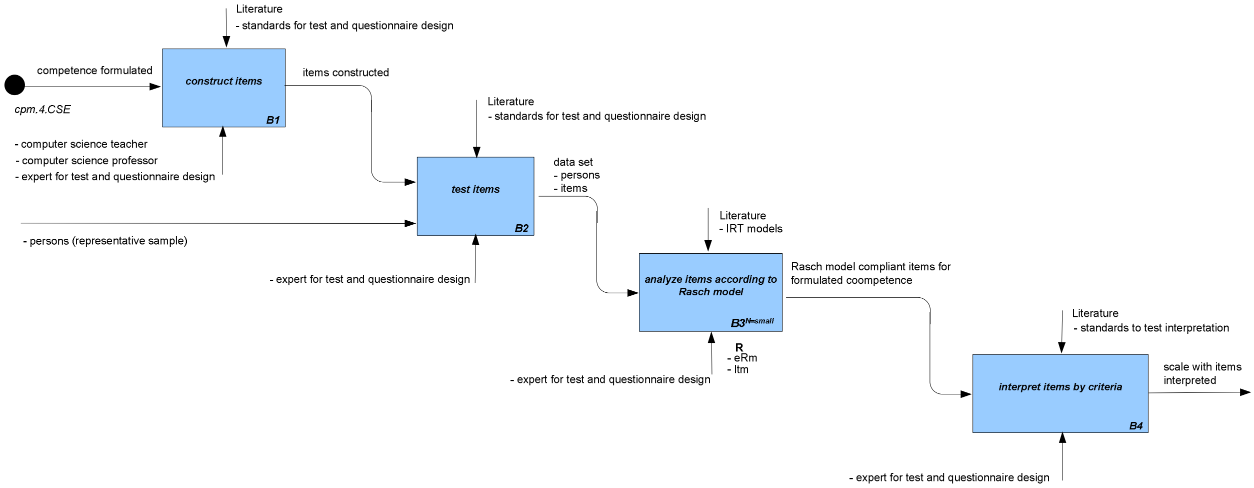

Literature
- standards for test and questionnaire design
competence formulated
construct items
B1
cpm.4.CSE
items constructed
- computer science teacher
- computer science professor
- expert for test and questionnaire design
Literature
- standards for test and questionnaire design
test items
B2
data set
- persons
- items
- persons (representative sample)
- expert for test and questionnaire design
Literature
- IRT models
analyze items according to Rasch model
B3N=small
Rasch model compliant items for formulated coompetence
R
- eRm
- ltm
- expert for test and questionnaire design
Literature
- standards to test interpretation
interpret items by criteria
B4
scale with items interpreted
- expert for test and questionnaire design

Figure 2. Sub processes of *cpm.4.CSE/IRT*$^{N=small}$

3.1 Sub process B1 construct items

The first sub process *B1 construct items* (see Figure 2) deals with the construction of items. From *B1* (see

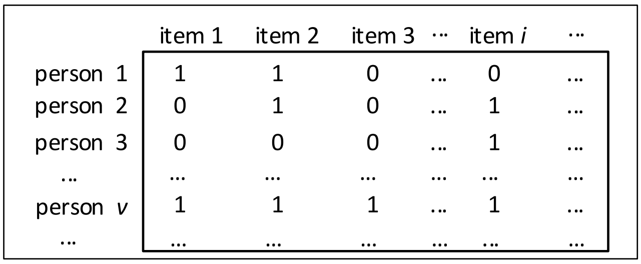
Figure 2) the following 12 items have been constructed for the competence area of *Modeling* with respect to the competence of *Diagram types* (Zendler, Seitz, & Klaudt, 2016):

item1: „Learners justify different model concepts, in particular they qualify the class diagrams for computer science“
Item2: „Learners adequately apply the concepts of sequence modeling“
Item3: „Learners determine the use of class and sequence diagrams in software engineering“
Item4: „Learners demonstrate the use of models in software engineering“
Item5: „Learners use diagram types for requirements modeling“
Item6: „Learners analyze requirements and apply use case diagrams“
Item7: „Learners analyze requirements and apply sequence diagrams“
Item8: „Learners analyze requirements and apply activity diagrams“
Item9: „Learners analyze requirements and apply state diagrams“
Item10: „Learners analyze requirements and apply class diagrams“
Item11: „Learners are convinced of modeling as a key activity in software engineering“
Item12: „Learners have an overview of various modeling languages such as Unified Modeling Language (UML), Event Driven Process Chains (EPK), Petri Nets, IDEF (Integrated DEFinition), SDL (Specification and Description Language), ERM (Entity-Relationship Model)“

The sub process *B1* is described in detail by *cpm.4.CSE/IRT* (Zendler, 2018).

3.2 Sub process B2 test items

The second sub process *B2 test items* (see Figure 2) is about testing the items. Output of *B2* is a data set in the form of a data matrix with scores from persons to items (see in Figure 3). The cells of the data matrix contain codings 1 (person *v* has solved item *i*) and 0 (person *v* has not solved item *i*).



	item 1	item 2	item 3	···	item *i*	···
person 1	1	1	0	...	0	...
person 2	0	1	0	...	1	...
person 3	0	0	0	...	1	...
...	...	...	...	...	...	...
person *v*	1	1	1	...	1	...
...	...	...	...	...	...	...

Figure 3. Data matrix

3.3 Sub process $B3^{N=small}$ analyze items according to Rasch model

As for *cpm.4.CSE/IRT,* the third sub process *$B3^{N=small}$ analyze items according to Rasch model* is the core of *$cpm.4.CSE/IRT^{N=small}$* and includes four subordinated processes (see Figure 4) that can be iterated for data analysis using the dichotomous Rasch logistic model. The steps are: *b3.1 test model assumptions statistically by non-parametric test, b3.2 estimate model parameters, b3.3 assess model fit,* and *b3.4 provide model information graphically.*

It is important to see from Figures 4 that *$B3^{N=small}$* executes *b3.1* without first estimating model parameters. In *cpm.4.CSE/IRT*, first the model parameters had to be estimated, which then influenced the statistical tests confronting the model assumptions.

Input. The input for sub process *$B3^{N=small}$* is a data set which is to be analyzed according to the Rasch model.

Output. The output of sub process *$B3^{N=small}$* are Rasch model compliant items for a formulated competence.

Control. The conditions for sub process *$B3^{N=small}$* are in particular model assumptions underlying the Rasch model; they will be further specified in the subordinated processes of *$B3^{N=small}$*.

*Mechanisms***.** The roles and resources involved in process *$B3^{N=small}$* are experts in test and questionnaire design, computer science teachers and professors (see Figure 2).

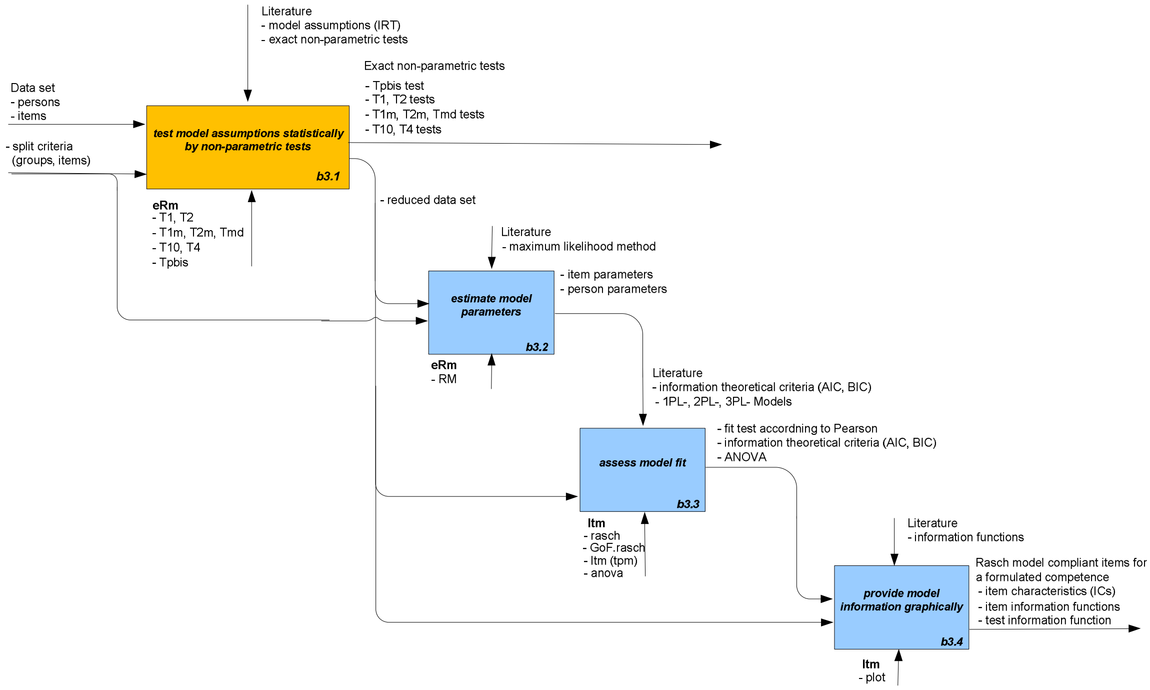

Literature
- model assumptions (IRT)
- exact non-parametric tests
Exact non-parametric tests
- Tpbis test
- T1, T2 tests
- T1m, T2m, Tmd tests
- T10, T4 tests
Data set
- persons
- items
- split criteria
(groups, items)
test model assumptions statistically by non-parametric tests
b3.1
eRm
- T1, T2
- T1m, T2m, Tmd
- T10, T4
- Tpbis
- reduced data set
Literature
- maximum likelihood method
estimate model parameters
b3.2
- item parameters
- person parameters
eRm
- RM
Literature
- information theoretical criteria (AIC, BIC)
- 1PL-, 2PL-, 3PL- Models
assess model fit
b3.3
- fit test accordning to Pearson
- information theoretical criteria (AIC, BIC)
- ANOVA
ltm
- rasch
- GoF.rasch
- ltm (tpm)
- anova
Literature
- information functions
provide model information graphically
b3.4
Rasch model compliant items for a formulated competence
- item characteristics (ICs)
- item information functions
- test information function
ltm
- plot

Figure 4. Sub process B3$^{\text{N=small}}$ analyze items according to Rasch model

Before R can be used to begin the analysis according to the Rasch model, the packages eRm (Mair & Hatzinger, 2007) and ltm Rizopoulos (2006) must be installed and loaded. In addition, the directory for the data set to be loaded must be set:

```
> chooseCRANmirror()
> install.packages("eRm")
> install.packages("ltm")
> library(eRm)
> library(ltm)
> setwd("…/…")
```

Input. Input for sub process *b3.1* is the data set comINF, which is loaded in R:

```
> load(file = "comINF.rda")
```

An overview of the loaded data set, which contains a total of N = 100 persons and k = 12 items, can be retrieved by

```
> head(dat)
> dat[1:7, 1:12]
```

Output. The output shows the data of the first seven persons p1 to p7 and the items item1 to item12

```
item1 item2 item3 item4 item5 item6 item7 item8 item9 item10 item11 item12
p1    1    1    0    0    1    0    0    1    0    1    0    1
p2    0    1    0    1    0    1    0    0    1    0    1    0
p3    0    0    0    1    0    1    1    1    0    1    1    0
p4    0    1    1    0    1    0    0    0    1    1    0    1
p5    0    0    1    1    0    1    0    0    1    1    0    0
p6    0    0    0    0    1    1    0    0    1    0    0    0
p7    1    1    0    0    0    1    0    1    1    1    0    0
```

With respect to *cpm.4.CSE/IRT*$^{N\text{=small}}$ a small sample of N = 30 persons was used. The sample has been generated by

```
> dat=dat[sample(1:nrow(dat),30,replace=FALSE), ]
```

Having prepared the data set, sub process *B3*$^{N\text{=small}}$ can start.

Sub process *b3.1* test model assumptions statistically by non-parametric tests - The sub process *b3.1 test model assumptions statistically by non-parametric tests* (see Figure 5) has the task to confront the data set with the assumptions of the Rasch model by applying quasi-exact non-parametric tests, especially it has the task to identify items that are not Rasch model compliant.

It is important to see from Figures 5 that *b3.1* starts without first estimating model parameters, i.e. the statistical tests are not influenced by estimated parameters, as is the case in *B3.1* through *B3.5* of cpm.4.CSE/IRT (see Zendler, 2018).

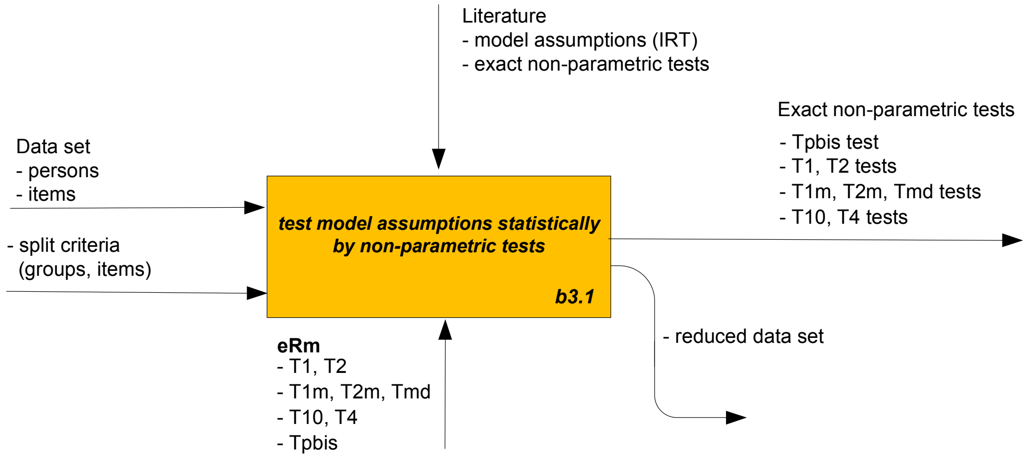


Figure 5. Sub process *b3.1* test model assumptions statistically by non-parametric tests

Sub process *b3.1* tests the following model assumptions:

A) Strictly monotonically increasing item-characteristic functions (via item discrimination).
B) Local stochastic independence,
C) One-dimensionality,
D) Specific objectivity.

A) Strictly monotonically increasing item-characteristic functions (via item discrimination)

To test whether items have monotonically increasing item characteristics, the test statistic T_{pbis} (Koller & Hatzinger, 2013, p. 10) is suitable, which compares the discriminations of items.

Input. To test an item, T_{pbis} is used in comparison against all other items. For example, to test item 1 T_{pbis} is executed by:

```
> tpbis = NPtest(dat, n=1000, method="Tpbis",idxt = 1, idxs=c(2:12))
> print(tpbis)
```

Output. The output of T_{pbis} shows no model violation (one-sided p-value (rpbis too low): 0.98) for item 1 in comparison with the other items (Subscale - Items: 2 3 4 5 6 7 8 9 10 11 12).

```
Nonparametric RM model test: Tpbis (discrimination)(pointbiserial correlation of test item vs. subscale)
Number of sampled matrices: 1000
Test Item: 1
Subscale - Items: 2 3 4 5 6 7 8 9 10 11 12
one-sided p-value (rpbis too low): 0.98
```

Table 1 contains the one-sided p-value for all items with respect to T_{pbis}. Items 4 and 6 show model deviation to strictly monotonically increasing item-characteristic functions.

Table 1

p-value for each item (T_{pbis})

item	1	2	3	4	5	6	7	8	9	10	11	12
p-value	0.98	0.46	0.84	0.0	0.68	0.0	0.99	0.77	0.99	0.86	0.74	0.85

B) Local stochastic independence

To test local stochastic independence, the test statistics T_1 and T_2 are suitable.

1. T_1 - T_1 (Koller & Hatzinger, 2013, p. 6) tests whether there are too many response patterns of type {00} or {11}.

Input. T_1 is started by

```
> t1=NPtest(dat, n=1000, method="T1")
> print(t1, alpha=.05)
```

Output. The output of T_1 shows that 4 item pairs leading to model deviation (Item-Pairs with one-sided $p < 0.05$).

```
Nonparametric RM model test: T1 (local dependence – increased inter-item correlations) (counting cases with equal responses
on both items)
Number of sampled matrices: 1000
Number of Item-Pairs tested: 66
Item-Pairs with one-sided p < 0.05
(3,9) (3,10) (4,6) (8,10)
0.02    0.02 0.01    0.01
```

The output to the item pairs tested shows that items 3, 4, 6, 8, 9, and 10 show increased inter-item correlations, they probably lead to model violation concerning local stochastic independence.

2. T_2 - T_2 (Koller & Hatzinger, 2013, p. 7) tests whether the variance (stat= "var") of person scores is too high for a subscale. For T_2, those items are grouped into a subscale, which are similar with respect to for the competence area of *Modeling*, namely items 3, 5, 8, and 10.

Input. To compare scores for items 3, 5, 8, 10, T_2 is executed by

```
> t2=NPtest(dat, n=1000, method="T2", idx=c(3,5,8,10))
> print(t2)
```

Output. The output of T_2 shows model deviating subscales for Items in subscale: 3 5 8 10 (one-sided p-value: 0.001).

```
Nonparametric RM model test: T2 (local dependence - model deviating subscales) (increased dispersion of subscale person raw
scores)
Number of sampled matrices: 1000
Items in subscale: 3 5 8 10
Statistic: variance
one-sided p-value: 0.001
```

C) One-dimensionality

To test one-dimensionality the test statistics T_{1m}, T_{2m}, and T_{md} (m = “multidimensional“) are appropriate.

1. T_{1m} - T_{1m} (Koller & Hatzinger, 2013, p. 7) tests whether there are too many response patterns of type {00} or {11}.

Input. T_{1m} is executed by

```
> t1m=NPtest(dat, n=1000, method="T1m")
> print(t1m, alpha=.05)
```

Output. The output of T_{1m} shows model violation for 7 item pairs: Item-Pairs with one-sided $p < 0.05$.

```
Nonparametric RM model test: T1m (multidimensionality – reduced inter-item correlations) (counting cases with equal
responses on both items)
Number of sampled matrices: 1000
Number of Item-Pairs tested: 66
Item-Pairs with one-sided p < 0.05
(4,5) (4,8) (4,10) (5,6) (6,8) (6,10)
0.02    0.02 0.02    0.0    0.0    0.01
```

The output to the item pairs tested shows that items 4, 5, 6, 8, and 10 show reduced inter-item correlations, they probably lead to model violation concerning one-dimensionality.

2. T_{2m} - T_{2m} (Koller & Hatzinger, 2013, p. 8) tests whether the variance of person scores is too low for a subscale.

Input. To compare items 3, 5, 8, 10, T_{2m} is executed by

```
> t2m=NPtest(dat, n=1000, method="T2m",idx=c(3,5,8,10), stat= "var")
> print(t2m)
```

Output. The output of T_{2m} shows no model deviating subscales for Items in subscale: 3 5 8 10 (one-sided *p*-value: 1).

```
Nonparametric RM model test: T2m (multidimensionality - model deviating subscales) (decreased dispersion of subscale
person rawscores)
Number of sampled matrices: 1000
Items in subscale: 3 5 8 10
Statistic: variance
one-sided p-value: 1
```

3. T_{md} - T_{md} (Koller & Hatzinger, 2013, p. 9) tests – for two sub scales – the homogeneity of items by using the correlation of person scores. T_{md} is the non-parametric analogue of the Martin-Löf test (see *cpm.4.CSE/IRT)*.

Input. To compare the easiest items 3, 6, 8, 9, 10, 11 from subscale 1 with the most difficult items 1, 2, 4, 5, 7, 12 from subscale 2, T_{md} is executed by

```
> tmd=NPtest(dat,n=1000, method="Tmd",idx1=c(3,6,8,9,10,11), idx2= c(1,2,4,5,7,12))
> print(tmd)
```

Output. The output of T_{md} shows no model violation for Subscale 1 - Items: 3 6 8 9 10 11 and Subscale 2 - Items: 1 2 4 5 7 10 12 due to Observed correlation: -0.04 (one-sided *p*-value: 0.43).

```
Nonparametric RM model test: Tmd (Multidimensionality) (correlation of subscale person scores)
Number of sampled matrices: 1000
Subscale 1 - Items: 3 6 8 9 10 11
Subscale 2 - Items: 1 2 4 5 7 12
Observed correlation: -0.04
one-sided p-value: 0.43
```

D) Specific objectivity

The test statistics T_{10} and T_4 are suitable to test the specific objectivity. While T_{10} tests globally, T_4 tests at the item level.

1. T_{10} - T_{10} (Koller & Hatzinger, 2013, p. 8), which is comparable to Andersen's likelihood ratio, tests globally whether the number of solved / non-solved items is different for two groups of persons. Typically, the persons are divided into two groups. One group contains persons who solved many items, the other group contains persons who solved few items.

Input. In order to compare items from two groups with division criterion "mean", T10 is executed by

```
> t10=NPtest(dat, n=1000, method="T10", splitcr="mean")
> print(t10)
```

Output. The output of T_{10} shows global model violation (one-sided p-value: 0.083) for the two groups (Group 1: n = 15 Group 2: n = 16).

```
Nonparametric RM model test: T10 (global test - subgroup-invariance)
Number of sampled matrices: 1000
Split: mean
Group 1: n = 14    Group 2: n = 16
one-sided p-value: 0.083
```

2. T_4 - T_4 (Koller & Hatzinger, 2013, p. 9) tests two groups of persons whether the number of solved / non-solved items is different for a specific item.

Input. To test an item, T_4 is used with respect to the group of person with solved items > median. For example to test item1, T_4 is executed by

```
> score=rowSums(dat)
> split=ifelse(score > median(score),1,0)
> t4=NPtest(dat, n=1000, method="T4",idx=1, group=split==1)
> print(t4)
```

Output. The output to T_4 shows no model violation (one-sided p-value: 0.90) for item 1 with respect to the specified group.

```
Nonparametric RM model test: T4 (Group anomalies - DIF) (counting high raw scores on item(s) for specified group)
Number of sampled matrices: 1000
Items in Subscale: 1
Group: split == 1    n = 16
one-sided p-value: 0.90
```

Table 2 contains the one-sided p-value for all items with respect to T_4. Item 6 show model deviation to specific objectivity.

Table 2

p-value for each item (T_4)

item	1	2	3	4	5	6	7	8	9	10	11	12
p-value	0.90	0.64	0.57	0.09	0.57	0.02	0.97	0.84	1.000	0.58	0.65	.71

Control. The conditions of sub process *b3.3* include a set of test statistics described in the literature with respect to quasi-exact non-parametric tests for the Rasch model (Ponocny, 2002; Verhelst, Hatzinger, & Mair, 2007; Christensen & Kreiner. 2010; Koller, Alexandrowicz, & Hatzinger, 2012; Koller & Hatzinger, 2013).

*Mechanisms***.** The roles involved in sub process *b3.3* are experts in test and questionnaire design. From the eRm package, the NPtest function has to be used, which contains the exact non-parametric tests. The NPtest function can be optionally equipped with parameters for the so-called *burn-in* period and the step size to ensure that simulated data matrices are independent of each other (see Koller, Alexandrowicz, & Hatzinger, 2012, pp. 108–110).

Summary of non-parametric statistics - Table 3 summarizes the results from the non-parametric tests. It becomes clear that items 4 and 6 are not Rasch-compliant. They are statistically significant in relation to almost all assumptions of the Rasch model. For items 8 and 10, the assumptions of local stochastic independence and of one-dimensionality are violated (tests T_1, T_2 were significant).

Against the background of the statistical tests it is decided to eliminate items 4, 6, 8, and 10. All other items remain in the item set. Items 3, 5, and 9 have been *re*-tested. They show no local dependence having items 8 and 10 eliminated.

Table 3

Summary of tests

A) Strictly increasing item-characteristic funtions		
	T_{bis}	item 4, item 6
B) Local stochastic dependence	T_1	items 3, 4, 6, 8, 9, 10
	T_2	items 3, 5, 8, 10
C) One-dimensionality	T_{1m}	items 4, 5, 6, 8, 10
	T_{2m}	*no* items suspicious
	T_{md}	*no* items suspicious
D) Specific objectivity	T_{10}	*no global deviation*
	T_4	item 6

Sub process b3.2 estimate model parameters - Having eliminated items 4, 6, 8, and 10, sub process *b3.3 estimate model parameters* is executed (*cpm.4.CSE/IRT*, Zendler, 2018). The output of *b3.3* for Item and person parameters is shown below.

A) Item parameters

```
Coefficients:
Dffclt.item1  Dffclt.item2  Dffclt.item3  Dffclt.item5  Dffclt.item7
     1.870          1.195          -0.148          -0.471          1.398
Dffclt.item9  Dffclt.item11  Dffclt.item12
     0.328         -1.392          1.620
```

B) Person parameters

```
Person Parameters:
Raw Score     Estimate Std.Error
        2 -1.760258722 0.7960789
        3 -1.215434189 0.6918836
        4 -0.775274297 0.6404566
        5 -0.382784990 0.6158991
        6 -0.009064153 0.6094216
        7 0.366349101 0.6186335
        8 0.763804152 0.6455400
```

Sub process b3.3 assess model fit - In analogy to *cpm.4.CSE/IRT*, sub process *b3.3 assess model fit* is executed (*cpm.4.CSE/IRT*, Zendler, 2018). The output of *b3.3* contains information to:

A) Global χ^2-fit test according to Pearson
B) Rasch model (1PL model) and Birnbaum model (2PL model),
C) Comparison of models

A) Global χ^2-fit test according to Pearson

The output shows the value for the χ^2-fit (Tobs: 589.05) and the *p*-value (*p*-value: > 0.05).

```
Bootstrap Goodness-of-Fit using Pearson chi-squared

Call:
rasch(data = datReduced, constraint = cbind(ncol(datReduced) + 1, 1), start.val = "random")

Tobs: 589.05
# data-sets: 101
p-value: > 0.05
```

The value for the *p*-value (*p*-value: > 0.05) suggests model fit.

B) Rasch model (1PL model) and Birnbaum model (2PL model)

In the second step, the item and person parameters for the (reduced) data set are analyzed according to the

Rasch model (1PL model) and the Birnbaum model (2PL model). The output gives estimates for the Rasch model (1PL) and the Birnbaum model (2PL) with respect to item difficulties and to the logarithmic likelihood. For reasons of space, only the logarithmic likelihoods are reproduced:

```
Log.Lik
onePL -138.29
twoPL -131.46
```

C) Comparison of models

The values for AIC, especially for BIC, show that with the more complex Birnbaum model no better model fit can be achieved than with the Rasch model; moreover, the two models do not differ significantly (LRT = 13.65, *p*.value=0.091).

```
Likelihood Ratio Table
AIC    BIC log.Lik    LRT df p.value
onePL 292.57 303.78 -138.29
twoPL 294.82 417.34 -131.46 13.65    8 0.091
```

Sub process b3.4 provide model information graphically - The sub process *b3.4 provide model information graphically* (see Figure 6) has the task of presenting visual information about individual items and the entire test. For this purpose, the following information is provided by process *b3.4*:

A) Item Characteristics Curves
B) Item information functions
C) Test information function

In analogy to *cpm.4.CSE/IRT* (Zendler, 2018), sub process *b3.4 provide model information graphically* is executed. The output of sub process *b3.4* is depicted in Figure 6.

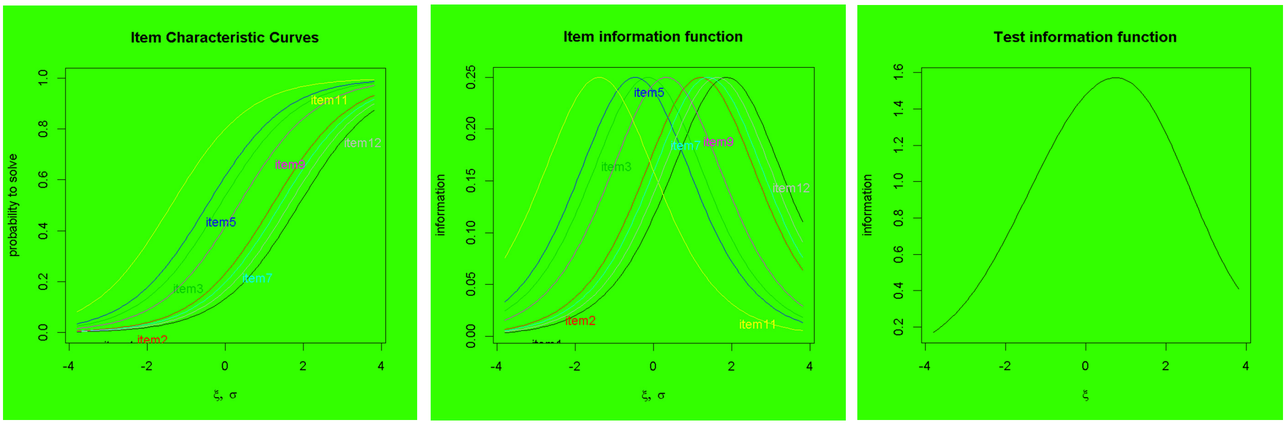


Figure 6. Output of sub process b3.4 interpret items by criteria

3.4 Sub process B4 interpret items by criteria

The sub process *B4 interpret items by criteria* has the task (see Figure 2) of assigning competence levels to items and interpreting them by criteria. For this purpose, (A) the item difficulties σ_i are standardized and then (B) interpreted by a *post-hoc* analysis (cf. Beaton & Allen, 1992; Moosbrugger & Kelava, 2012, p. 258). As input for sub process *B4*, the Rasch model compliant items are used including their difficulty parameters, which were determined in sub process *b3.3* for a formulated competence.

In analogy to *cpm.4.CSE/IRT*, sub process *B4 interpret items by criteria* is executed (*cpm.4.CSE/IRT*, Zendler, 2018), which produces information to A) Standardization and B) Post hoc analysis

A) Standardization

Table 4 shows the items with the standardized item difficulties σ_i, the item difficulties $0(\sigma_i)$ fixed to the sum of 0, their z-values $z(\sigma_i)$ and their values $PT(\sigma_i)$) in the standardization context of PISA / TIMSS.

Table 4

Items and anchor items with standardized item difficulties

Items	σ_i	$0(\sigma_i)$	$z(\sigma_i)$	$PT(\sigma_i)$
item1	1.870	1.320	1.220	622
Item2	1.195	0.645	0.600	560
item3	-0.148	-0.698	-0.645	436
item5	**-0,471**	**-1.021**	**-0.943**	**406**
item7	1.398	0.826	0.783	578
Item9	0.328	-0.222	-0.205	480
Item11	-1.392	-1.942	-1.794	321
item12	**1.620**	**1.070**	**0.989**	**599**

B) Post hoc analysis

The *post hoc* analysis interprets the items criteria-orientedly. For this purpose, a competence scale (divided into sections) for item difficulties in the standardization context of PISA / TIMSS is first determined. Then, for each section, a so-called anchor item (including its difficulty parameter) is defined, which describes the competence level based on its specific requirement.

Figure 7 shows the scale for the competence *Diagram types* with competence levels 1, 2, 3 and 4 and the located items. From the Figure it can be seen that items item5 and item12 with the difficulty parameters $PT(\sigma_5)$ = 406 and $PT(\sigma_{12})$ = 599 are suitable for defining the competence levels 1 and 3, respectively. For the competence level 2 and 4, no items are predestined.

The competence levels can be interpreted and defined in a criteria-oriented manner via the requirements of item5 and item12. Item5 “Learners use diagram types for requirements modeling” interprets and defines competence level 3, item12 “Learners have an overview of various modeling languages such as Unified Modeling Language (UML), Event Driven Process Chains (EPK), Petri Nets, IDEF (Integrated DEFinition), SDL (Specification and Description Language), ERM (Entity-Relationship Model)” interprets and defines competence level 1.

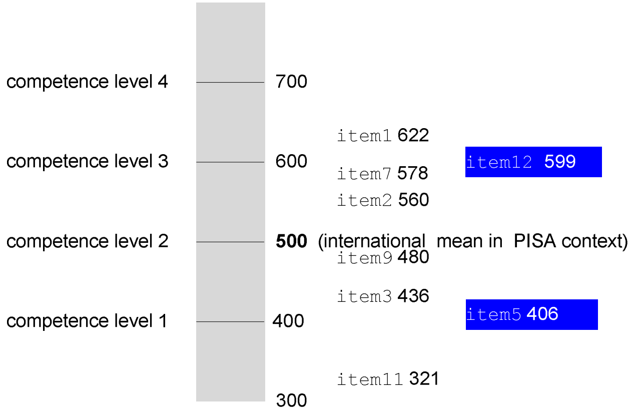


Figure 7. Scale for competence *Diagram types*

The scale for the competence *Diagram types* (see Figure 7) is the final result of using *cpm.4.CSE/IRT*$^{N=small}$ given a very small sample size of N = 30. The results obtained by *cpm.4.CSE/IRT*$^{N=small}$ are similar to the results of *cpm.4.CSE/IRT* given a sample size N = 100. They differ, that items 8 and 10 (instead of item 9) have been eliminated, and items 5 and 12 (instead of item 7 and item 10) define the competence levels.

To compare the results of *cpm.4.CSE/IRT* and *cpm.4.CSE/IRT*$^{N=small}$, appendix contain the items with standardized item difficulties and the scale for the competence *Diagram types* from *cpm.4.CSE/IRT*.

4. Conclusions

In this supplementary article, the process model *cpm.4.CSE/IRT*$^{N=small}$ (*c*ompact *p*rocess *m*odel for *C*ompetence *S*cience *E*ducation based on *IRT* using a small sample) was presented. It is almost identical to *cpm.4.CSE/IRT* with the exception of sub process *b*3.1 *test model assumptions statistically by non-parametric tests. b3.1* was demonstrated using data for the competence *Diagram types* in the competence area of *Modeling* that were already prepared for *cpm.4.CSE/IRT*. However, not the complete data set was used by, but only a sample of $N = 30$, which was randomly drawn.

b3.1 of *cpm.4.CSE/IRT*$^{N=small}$ is equipped with a number of non-parametric tests which allow to (preliminary) validate items of measuring instruments even with small samples. For implementing *cpm.4.CSE/IRT*$^{N=small}$, the open-source software R was used, because R allows to realize and tailor statistical programs to respective problems with little effort, which is not common in commercial products. It is possible to implement R programs with interactive user interfaces as a web application using Shiny (Beeley, 2016; RStudio, 2019).

Moreover, *cpm.4.CSE/IRT*$^{N=small}$ can be used as a process model for the development of competence models in other subjects. The inputs and outputs for the four sub processes are then to be analogized with respect to the other subjects. In addition, the conditions and roles in the sub processes of *cpm.4.CSE/IRT*$^{N=small}$ have to be changed according to the requirements in other subjects.

For further work, it should be observed which tendencies emerge in the context of the Rasch model with small samples. Zwister and Maris (2016) who introduced the non-parametric Rasch model make suggestions.

5. References

Beaton, E., & Allen, N. (1992). Interpreting scales through scale anchoring. *Journal of Educational Statistics, 17*, 191–204. https://doi.org/10.2307/1165169

Beeley, C. (2016). *Web application development with R using Shiny*. Birmingham: Packt Publishing.

Christensen, K. B., & Kreiner, S. (2010). Monte Carlo tests of the Rasch model based on scalability coefficients. *British Journal of Mathematical and Statistical Psychology, 63*, 101–111. https://doi.org/10.1348/000711009X424200

CRAN (Comprehensive R Archive Network). (2019). CRAN task views. Retrieved from https://cran.r-project.org/AS

De Ayala, R. J. (2009). *The theory and practice of item response theory*. New York: Guilford press.

Eid, M., & Schmid, K. (2014). *Testtheorie und testkonstruktion*. Göttingen: Hogrefe.

Jones, P., Smith, R. & Talley, D. M. (2006). Developing test forms for small-scale achievement testing systems. In S. M. Downing, & T. M. Haladyna (Eds.), *Handbook of test development* (pp. 487–525.) Mahwah: Erlbaum.

Koller, I., & Hatzinger, R. (2013). Nonparametric tests for the Rasch model: Explanation, development, and application of quasi-exact tests for small samples. *InterStat, 11*, 1–16.

Koller, I., Alexandrowicz, R., & Hatzinger, R. (2012). *The Rasch model in practice* [Das Rasch modell in der Praxis]. Wien: Facultas.

Linacre, J. M. (1994). Sample size and item calibration stability. *Rasch Measurement Transactions, 7*, 328.

Menzel, C., & Mayer, R. (2005). The IDEF family of languages. In P. Bernus, K. Martins, & G. Schmidt (Eds.), *Handbook on architectures of information systems* (pp. 215–250). Berlin: Springer. https://doi.org/10.1007/3-540-26661-5_10

Moosbrugger, H., & Kelava, A. (Eds.) (2012). *Test theory and questionnaire construction* [Testtheorie und

Fragebogenkonstruktion]. Berlin: Springer.

Ponocny, I. (2002). Nonparametric goodness of fit tests for the Rasch model. *Psychometrika, 66*, 437–460. https://doi.org/10.1007/BF02294444

RStudio. (2019). Shiny. Retrieved from http://shiny.rstudio.com/

Robinson, P. W., & Foster, D. F. (1979). *Experimental psychology: A small-N approach.* New York: Harper & Row.

Smith, P. L., & Little, D. R. (2018). Small is beautiful: In defense of the small-*N* design. *Psychonomic Bulletin and Review, 25*, 2083–2101. https://doi.org/10.3758/s13423-018-1451-8

Verhelst, N. D (2008). An efficient MCMC algorithm to sample binary matrices. *Psychometrika, 73*, 705–728. https://doi.org/10.1007/s11336-008-9062-3

Verhelst, N. D., Hatzinger, R., & Mair, P. (2007). The Rasch sampler. *Journal of Statistical Software, 20*(4), 1–14. https://doi.org/10.18637/jss.v020.i04

Zendler, A., Seitz, C., & Klaudt, D. (2016). Process-based development of competence models to computer science education. *Journal of Educational Computing Research, 54*(2), *563*–592. https://doi.org/10.1177/0735633115622214

Zendler, A. (2018). cpm.4.CSE/IRT: compact process model for measuring competences in Computer Science Education based on IRT models. *Education and Information Technology*, *24*(1), 843–884. https://doi.org/10.1007/s10639-018-9794-3

Zwister, R. J., & Maris, G. (2016). Ordering individuals with sum scores: The introduction of the nonparametric Rasch model. *Psychometrika, 81*(1)*,* 39–59. https://doi.org/10.1007/s11336-015-9481-x

Appendices

A-1 Items with standardized item difficulties from cpm.4.CSE/IRT

Table A-1

Items and anchor items with standardized item difficulties from cpm.4.CSE/IRT

	σ_i	$0(\sigma_i)$	$z(\sigma_i)$	$PT(\sigma_i)$
item1	1.288	0.818	0.927	593
Item2	0.892	0.422	0.478	548
item3	0.009	-0.461	-0.522	448
item5	0.150	-0.320	-0.363	464
item7	**1.413**	**0.943**	**1.069**	**607**
item8	-0.133	-0.603	-0.683	432
Item9	NA	NA	NA	NA
item10	**-0.181**	**-0.651**	**-0.738**	**426**
Item11	-1.046	-1.516	-1,718	328
item12	1.837	1.367	1.550	655

A-2 Scale for the competence Diagram types from cpm.4.CSE/IRT

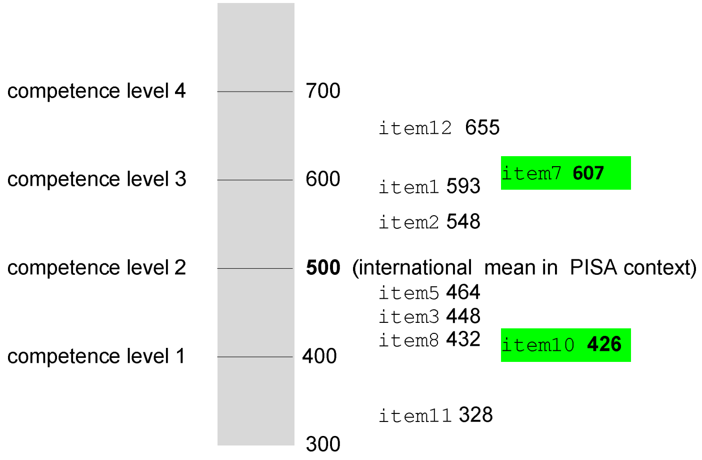


Figure A-1. Scale for the competence *Diagram types*

A-3 R script for cpm.4.CSE/IRT$^{N\text{=small}}$

The following listing shows the R script for the sub process *B3* of *cpm.4.CSE/IRT*$^{N\text{=small}}$. Due to lack of space, the listing is kept to a minimum, without comments, functions are only called with the most necessary parameters. Explanations of the individual functions are contained in the documentations of the eRm and ltm packages (see Mair & Hatzinger, 2007 and Rizopoulos, 2006, respectively).

```
# ----------------------------------------------------------------------------------
# B3 analyze items according to Rasch model / nonparametric tests
#
   chooseCRANmirror()
   install.packages("eRm")
   install.packages("ltm")
   library(eRm)
   library(ltm)
   setwd(".../...")
   load(file = "...")                              # load data set
#
   dat=dat[sample(1:nrow(dat),30,replace=FALSE), ]   # sample of N=30
```

```
#
# --------------------------------------------------------------------------------
# b3.3 test model assumptions statistically by non-parametric tests
# --------------------------------------------------------------------------------
#
# A) Strictly monotonically increasing item-characteristic functions
#
   tpbis =NPtest(dat, n=1000, method="Tpbis",idxt = 1, idxs=c(2:12))
   print(tpbis)
#
# B) Local stochastic independence
#
# 1. T1: too many response patterns of type {00} or {11}.
#
   t1=NPtest(dat, n=1000, method="T1")
   print(t1, alpha=.05)
#
# 2. T2: tests whether the variance of the person scores is too high.
#
   t2=NPtest(dat, n=1000, method="T2",idx=c(3,5,8,10), stat="var")
   print(t2)
#
# C) One-dimensionality
#
# 1. T1m: tests whether there are too many response patterns of type {00} or {11}.
#
   t1m=NPtest(dat, n=1000, method="T1m")
   print(t1m, alpha=.05)
#
# 2. T2m: tests whether the variance of person scores is too low for a subscale.
#
   t2m=NPtest(dat, n=1000, method="T2m",idx=c(3, 5, 8, 10), stat="var")
   print(t2m)
#
# 3. Tmd: non-parametric analogue of the Martin-Löf test): tests – for two sub
#    scales – the homogeneity of items by using the correlation of person scores.
#
   tmd=NPtest(dat,n=1000,method="Tmd",idx1=c(3,6,8,9,10,11), idx2= c(1,2,4,5,7,12))
   print(tmd)
#
# D) Specific objectivity
#
# 1. t10 (comparable to Andersen's likelihood ratio), tests globally whether the
#     number of solved / non-solved items is different for two groups of persons.
#
   t10=NPtest(dat, n=1000, method="T10",splitcr="mean")
   print(t10)
   score=rowSums(dat)
   split=ifelse(score <=median(score),0,1)
   t4=NPtest(dat, n=1000, method="T4",idx =12, group=split==0)
   print(t4)
#
# 2. t4: tests for two groups of persons whether the number of solved / non-solved
#     items is different for a specific item
#
   score=rowSums(dat)
   split=ifelse(score <=median(score),1,0)
   t4=NPtest(dat, n=1000, method="T4",idx =1, group=split==1)
   print(t4)
#
# --------------------------------------------------------------------------------
```