TAM: An R Package for Item Response Modelling

Thomas Kiefer\textsuperscript{1}, Alexander Robitzsch\textsuperscript{1}, Margaret Wu\textsuperscript{2}

\textsuperscript{1}Federal Institute for Educational Research, Innovation and Development of the Austrian School System (BIFIE), Salzburg, Austria,
\textsuperscript{2}Educational Measurement Solutions, Victoria, Australia

useR! 2015 Aalborg
July 2, 2015
Co-Authors

Alexander Robitzsch.
https://www.bifie.at/user/robitzsch-alexander

Professor Margaret Wu.
Long-term Statistician for the OECD, Scientist at the Work-based Education Research Centre of the Victoria University Melbourne, Australia and Co-Founder of Educational Measurement Solutions.
http://www.edmeasurement.com.au
Overview

Introduction
Item Response Theory
Why TAM?
Conclusion
Introduction
Psychometrics is the (statistical) field of measuring psychological concepts.

A field of application is the educational large-scale assessment (LSA).

The psychological concept in an LSA is defined in a competency construct.

The competence construct is a – often very broad – definition of the students trait in question.
**Psychometrics** is the (statistical) field of measuring psychological concepts.

A field of application is the educational large-scale assessment (LSA).

The psychological concept in an LSA is defined in a competency construct.

The competence construct is a – often very broad – definition of the students trait in question.

---

**Figure:** Model for the Competency construct of mathematical literacy in PISA (OECD, 2013)
• Traits are measured using *items*.
• Items apply to a specific *domain* within the competence construct.
• Items are scored *dichotomously* (right / wrong) or *polytomously* (partial credit).
• Items are either *closed response* or *constructed response* format.
• A measurement is the students score to that or an equivalent item of the domain.

Figure: A typical item used in PISA

(https://nces.ed.gov/surveys/pisa/releaseditems.asp, June '15)
Traits are measured using items.

Items apply to a specific domain within the competence construct.

Items are scored dichotomously (right / wrong) or polytomously (partial credit).

Items are either closed response or constructed response format.

A measurement is the students score to that or an equivalent item of the domain.

Figure: Another typical item used in PISA

(https://nces.ed.gov/surveys/pisa/releaseditems.asp, June ’15)
Measurement

- To measure a domain sufficiently precise a large amount of items is used.
- Individual students are presented with a reasonably small representative sample (a booklet) of all possible items.
- Statistical Inference is obtained using Item Response Theory (IRT).

<table>
<thead>
<tr>
<th>idstud</th>
<th>female</th>
<th>migra</th>
<th>M192Q01</th>
<th>M406Q01</th>
<th>M406Q02</th>
<th>M423Q01</th>
<th>M496Q01</th>
<th>M496Q02</th>
<th>M564Q01</th>
<th>M564Q02</th>
<th>M571Q01</th>
<th>M603Q01</th>
<th>M603Q02</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

R Package TAM (Kiefer, Robitzsch, & Wu)
To measure a domain sufficiently precise a large amount of items is used.

Individual students are presented with a reasonably small representative sample (a booklet) of all possible items.

Statistical Inference is obtained using Item Response Theory (IRT).
Item Response Theory
IRT Models

IRT models are generalized nonlinear mixed effects models:

- the score $Y_{pi} \in \{0, 1\}$ of a student $p$ to an item $i$ is the dependent variable,
- given a randomly sampled student’s trait, e.g. $\theta_p \sim N(\mu, \sigma^2)$, the responses are assumed to be independent Bernoulli distributed,
- given $\theta_p$, the predictor $\eta_{pi} = \logit (P(Y_{pi} = 1))$ is a linear combination of item characteristics

$$\eta_{pi} = \sum_{k=0}^{K} b_k X_{ik} + \theta_p + \varepsilon_{pi},$$

- let $X_{ik} = -1$, if $i = k$, and $X_{ik} = 0$, otherwise - thus obtain the Rasch model

$$P(Y_{pi} = 1 \mid \theta_p) = \frac{\exp(\theta_p - b_i)}{1 + \exp(\theta_p - b_i)};$$

(De Boeck & Wilson, 2004; Lord & Novick, 1968)
Extended IRT Models: Item side

IRT models are extended towards different aspects:

- With respect to **discriminatory power** and **guessing ratio** of an item
- With respect to **polytomous scores**

(Andersen, 1977; Birnbaum, 1968; Muraki, 1993; Rasch, 1960)
Extended IRT Models: Item side

IRT models are extended towards different aspects:

- With respect to **discriminatory power** and **guessing ratio** of an item

\[
P(Y_{pi} = 1 \mid \theta_p) = \frac{\exp(a_i(\theta_p - b_i))}{1 + \exp(a_i(\theta_p - b_i))}
\]

- With respect to **polynomial scores**

(R Package TAM (Kiefer, Robitzsch, & Wu) Item Response Theory)
Extended IRT Models: Item side

IRT models are extended towards different aspects:

- With respect to **discriminatory power** and **guessing ratio** of an item

\[
P(Y_{pi} = 1 \mid \theta_p) = c_i + (1 - c_i) \frac{\exp(a_i (\theta_p - b_i))}{1 + \exp(a_i (\theta_p - b_i))},
\]

- With respect to **polytomous scores**

(Ander sen, 1977; Birnbaum, 1968; Muraki, 1993; Rasch, 1960)
Extended IRT Models: Item side

IRT models are extended towards different aspects:

- With respect to **discriminatory power and guessing ratio** of an item
- With respect to **polynomial scores**

\[
P(Y_{pi} = k | \theta_p) = \frac{\exp (a_{ik} \theta_p - b_{ik})}{\sum_{k=0}^{K} \exp (a_{ik} \theta_p - b_{ik})}.
\]

(Andersen, 1977; Birnbaum, 1968; Muraki, 1993; Rasch, 1960)
Extended IRT Models: Person side

IRT models are extended towards different aspects:

- With respect to known student characteristics constituting the population (e.g., sex, migration status)
- With respect to construct dimensionality
- With respect to discrete skill classes (continuous distributions can be easily approximated by discrete ones)

Regression Coefficients

V1
[1,] 0.704

Variance:
[,]1
[1,] 1.613

(Adams, Wilson, & Wang, 1997; Xu & von Davier, 2007; ?)
Extended IRT Models: Person side

IRT models are extended towards different aspects:

- With respect to known student characteristics constituting the population (e.g., sex, migration status)

  \[ \theta_p \sim N (\mathbf{Z}_\beta, \sigma^2) , \]

- With respect to construct dimensionality
- With respect to discrete skill classes (continuous distributions can be easily approximated by discrete ones)

Regression Coefficients

<table>
<thead>
<tr>
<th></th>
<th>V1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1.0263</td>
</tr>
<tr>
<td>female</td>
<td>0.3342</td>
</tr>
<tr>
<td>migra</td>
<td>-0.7008</td>
</tr>
</tbody>
</table>

Variance:

\[
\begin{bmatrix}
1 & 0.1988 \\
0.1988 & 1
\end{bmatrix}
\]

(A dams et al., 1997; Xu & von Davier, 2007; )

R Package TAM (Kiefer, Robitzsch, & Wu)
Extended IRT Models: Person side

IRT models are extended towards different aspects:

- With respect to known student characteristics constituting the population (e.g., sex, migration status)
- With respect to construct dimensionality

\[ P(Y_{pi} = 1 | \theta_p) = \frac{\exp(\sum_d a_{id} \theta_{pd} - b_i)}{1 + \exp(\sum_d a_{id} \theta_{pd} - b_i)}, \quad \theta_p \sim N^d(\mu, \Sigma) \]

- With respect to discrete skill classes (continuous distributions can be easily approximated by discrete ones)

<table>
<thead>
<tr>
<th>Regression Coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>[,1] [,2] [,3]</td>
</tr>
<tr>
<td>Intercept</td>
</tr>
<tr>
<td>1.4055 0.3483 1.2977</td>
</tr>
<tr>
<td>female</td>
</tr>
<tr>
<td>0.2428 0.3026 0.4821</td>
</tr>
<tr>
<td>migra</td>
</tr>
<tr>
<td>-0.6371 -0.5466 -0.7526</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variance:</th>
</tr>
</thead>
<tbody>
<tr>
<td>[,1] [,2] [,3]</td>
</tr>
<tr>
<td>[1,] 1.354 1.291 1.831</td>
</tr>
<tr>
<td>[2,] 1.291 1.327 1.812</td>
</tr>
<tr>
<td>[3,] 1.831 1.812 2.621</td>
</tr>
</tbody>
</table>
Extended IRT Models: Person side

IRT models are extended towards different aspects:

- With respect to known student characteristics constituting the population (e.g., sex, migration status)
- With respect to construct dimensionality
- With respect to discrete skill classes (continuous distributions can be easily approximated by discrete ones)

\[
P(Y_{pi} = 1 \mid \theta_{p(l)}) = \frac{\exp(\theta_{p(l)} - b_{i(l)})}{1 + \exp(\theta_{p(l)} - b_{i(l)})}, \quad \theta_{p(l)} \in \{\theta_{p(1)}, \ldots, \theta_{p(L)}\}.
\]

<table>
<thead>
<tr>
<th>Full Trait distribution</th>
<th>Group1</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1,] 0.0667</td>
<td></td>
</tr>
<tr>
<td>[2,] 0.1730</td>
<td></td>
</tr>
<tr>
<td>[3,] 0.5206</td>
<td></td>
</tr>
<tr>
<td>[4,] 0.1730</td>
<td></td>
</tr>
<tr>
<td>[5,] 0.0667</td>
<td></td>
</tr>
</tbody>
</table>

SD Trait:

\[
\text{Group1 1.407}
\]
Why TAM?
Here’s why!

- **Open source solution** for everyday work in an educational assessment context (such as BIFIE);
- **Estimation processes at BIFIE prior to TAM:**
  1. Data preparation in R,
  2. Call to third-party software for IRT analyses (e.g., ConQuest).
- **ConQuest:**
  - Absence of standard API,
  - commercial black-box software.
- **R packages:**
  - **mirt** recently became suitable for use in LSA; still lacks some flexibility in specifying dependencies among item parameters.
  - Other R packages (e.g., **eRm**, **ltm**, **psychotools**) lack model classes or processing speed (or both) required for population-sized context.
- **TAM** is flexible due to **design matrices**; yet reasonably fast.
- **Bonus:** gain some deeper understanding of the estimation processes.

(Adams & Wu, 2007; Chalmers, 2012; Wu, Adams, Wilson, & Haldane, 2007)
Model Syntax

IRT models can be set up using *model syntax* statements (based on *lavaan*).

- Relevant aspects for specifying IRT models in *TAM* group into four types.
- The Rasch model is specified by a minimally complex input.
- Presented examples are necessarily limited; *tamaan* also allows for MODEL PRIOR, DO loops, and a lot more model classes.

```r
> ## Toy example
> head(dat)

  A1 A2 A3 A4 B1 B2 B3 B4 C1 C2 C3 C4
2  1  1  1  1  1  1  1  1  1  1  1  1
22 1  1  0  0  1  0  1  1  1  0  1  0
23 1  1  0  1  1  0  1  1  1  1  1  1
41 1  1  1  1  1  1  1  1  1  1  1  1
43 1  0  0  1  0  0  1  1  1  0  1  0
63 1  1  0  0  1  0  1  1  1  1  1  1
```

*R Package TAM (Kiefer, Robitzsch, & Wu)*
Model Syntax

IRT models can be set up using model syntax statements (based on lavaan).

- Relevant aspects for specifying IRT models in TAM group into four types.
- The Rasch model is specified by a minimally complex input.
- Presented examples are necessarily limited; tamaan also allows for MODEL PRIOR, DO loops, and a lot more model classes.

```r
> ## Basic setup
> tammodel <- "
   ANALYSIS:

   LAVAAN MODEL:

   ITEM TYPE:

   MODEL CONSTRAINT:

   "
> ## estimate model
> # mod <- tamaan(tammodel, resp = dat)
```
Model Syntax

IRT models can be set up using model syntax statements (based on lavaan).

- Relevant aspects for specifying IRT models in TAM group into four types.
- The Rasch model is specified by a minimaly complex input.
- Presented examples are necessarily limited; tamaan also allows for MODEL PRIOR, DO loops, and a lot more model classes.

```r
# Rasch model ----
tammodel <- "
ANALYSIS:
  TYPE = TRAIT;
LAVAAN MODEL:
  F1 =~ A1__C4
  F1 ~~ F1
ITEM TYPE:
  ALL(Rasch);
"

# estimate model
mod <- tamaan(tammodel, resp = dat, control = list(progress = FALSE))
mod$variance

V1
V1 1.190302
```

R Package TAM (Kiefer, Robitzsch, & Wu) Why TAM? 14
Model Syntax

IRT models can be set up using model syntax statements (based on lavaan).

- Relevant aspects for specifying IRT models in TAM group into four types.
- The Rasch model is specified by a minimaly complex input.
- Presented examples are necessarily limited; tamaan also allows for MODEL PRIOR, DO loops, and a lot more model classes.

```r
> ## Rasch model ----
> tammodel <- "
ANALYSIS:
  TYPE = TRAIT;
LAVAAN MODEL:
  F1 =~ A1__C4
  F1 ~~ F1
ITEM TYPE:
  ALL(Rasch);
"

> # estimate model
> mod <- tamaan(tammodel, resp = dat, control = list(progress = FALSE))
> mod$variance

V1
V1 1.190302
```

R Package TAM (Kiefer, Robitzsch, & Wu)  Why TAM?
Extending the \texttt{lavaan} syntax \texttt{tamaan} additionally implements convenient operators for specifications on the item side, such as sum over multiple entities "\_\_" and guessing parameters "\?=".

```r
## 3PL Model ----
> tammodel <- "
ANALYSIS:
  TYPE = TRAIT;
LAVAAN MODEL:
  F1 =~ A1__C4
  F1 ~~ 1 * F1
  A1 ?= g1
  B1 + C1 ?= gBC * g1
"

> # estimate model
> mod <- tamaan(tammodel, resp = dat, control = list(progress = FALSE))
> round(mod$.item$guess,2)

[1] 0.68 0.00 0.00 0.00 0.24 0.00 0.00 0.00 0.24 0.00 0.00 0.00
```

\textit{R Package TAM (Kiefer, Robitzsch, & Wu)}
Catchy definitions of model constraints for parameters are available.

```r
> ## MODEL CONSTRAINTS ----
> tammodel <- "
> ANALYSIS:
> TYPE = TRAIT;
> LAVAAN MODEL:
>   F1 =~ load1__load10 * A1__C2
>   F1 ~~ 1 * F1
> MODEL CONSTRAINT:
>   load2 == 1.1*load1
>   load3 == 0.9 * load1 + (-.1) * load0
>   load8 == load0
>   load9 == load0
> "
> # estimate
> mod <- tamaan(tammodel , resp = dat, control = list(progress = FALSE))
> head(tamaanify(tammodel, dat)$L[, 1, ], 3)

   load1 load0 load4 load5 load6 load7 load10
A1  1.0  0.0  0.0  0.0  0.0  0.0  0.0
A2  1.1  0.0  0.0  0.0  0.0  0.0  0.0
A3  0.9 -0.1  0.0  0.0  0.0  0.0  0.0
```

R Package TAM (Kiefer, Robitzsch, & Wu)
Model Syntax II

Using the options in analysis **TYPE**, Latent Class Analysis (LCA) models can be specified.

```r
## LCA Model ----
tammodel <- "
ANALYSIS:
  TYPE=LCA;
  NCLASSES(3);
  NSTARTS(5, 20);
LAVAAN MODEL:
  F =~ A1__C4
"

# estimate model
mod <- tamaan(tammodel, resp = dat,
+   control = list(progress = FALSE))
head(mod$lcaprobs, 3)
```

<table>
<thead>
<tr>
<th>item</th>
<th>itemno</th>
<th>Cat</th>
<th>Class1</th>
<th>Class2</th>
<th>Class3</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>1</td>
<td>0</td>
<td>0.3075322</td>
<td>0.165856</td>
<td>0.000045</td>
</tr>
<tr>
<td>A2</td>
<td>2</td>
<td>0</td>
<td>0.6924778</td>
<td>0.834144</td>
<td>0.999945</td>
</tr>
<tr>
<td>A2</td>
<td>2</td>
<td>0</td>
<td>0.487154</td>
<td>0.443722</td>
<td>0.028797</td>
</tr>
</tbody>
</table>
Messy task: at multiple integration nodes \( \theta_p \), efficiently compute,

\[
P(Y_{pi} = k \mid \theta_p) = \frac{\exp(\sum_d b_{ikd} \theta_{pd} + a_{ik} \xi_i)}{\sum_{k=0}^K \exp(\sum_d b_{ikd} \theta_{pd} + a_{ik} \xi_i)}, \forall i, k, p.
\]

```r
> calc_prob <- function(iIndex, A, AXsi, B, xsi, theta, nnodes, maxK){
+  AXsi.tmp <- array(tensor(A[iIndex, , , drop = FALSE], xsi, 3, 1),
+  dim = c(length(iIndex), maxK, nnodes))
+  AXsi[iIndex,] = AXsi.tmp[,,1]
+  Btheta <- array(0, dim = c(length(iIndex), maxK, nnodes))
+  for( dd in 1:ncol(theta)){
+    Btheta <- Btheta + array(B[iIndex, , dd, drop = FALSE] %o% theta[, dd],
+                               dim = dim(Btheta))
+  }
+  rprobs <- (rr <- exp(Btheta + AXsi.tmp)) /
+  aperm(array(rep(colSums(aperm(rr ,c(2, 1, 3)), dims = 1, na.rm=TRUE)), maxK),
+        dim = dim(rr)[c(1, 3, 2)])
+  return(list("rprobs" = rprobs, "AXsi" = AXsi))
+}
```
Messy and time consuming task: efficiently compute the posterior distribution

\[
f(\theta | Y) = \frac{f(Y | \theta) f(\theta)}{f(Y)}.
\]

```r
# compute posterior distribution
calc_posterior_TK <- function(rprobs, gwt, nitems){
  fx <- gwt
  for (i in 1:nitems){
    r.ii <- rprobs[i,,]
    fx <- fx * r.ii[resp[,i] + 1,]
  }
  hwt <- fx / rowSums(fx)
  return(hwt)
}
```
Messy and time consuming task: efficiently compute the posterior distribution

\[ f(\theta | Y) = \frac{f(Y | \theta) f(\theta)}{f(Y)}. \]

```c
for(i=0; i<nresp; i++){
    for(k=0; k<nnodes; k++){
        res[i+nresp*k] = REAL(sFx)[i+nresp*k];
    }
}

for(i=0; i<nitems; i++){
    // extract non-missing value list
    len = LENGTH(VECTOR_ELT(sRespIndList, i));
    ni = INTEGER(VECTOR_ELT(sRespIndList, i)); //ni indices in R, therefore '-1'

    //compute fx
    for(k=0; k<len; k++){
        for(l=0; l<nnodes; l++){
            res[ ni[k] + l*nresp - 1 ] = res[ ni[k] + l*nresp - 1 ] *
                                         rii[ i + resp[ni[k]+i*nresp - 1]*nitems + l*nitems*ncats ];
        }
    }
}
```
<table>
<thead>
<tr>
<th></th>
<th>TAM</th>
<th>mirt</th>
<th>eRm</th>
<th>ConQuest*</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>30</td>
<td>00:00:057</td>
<td>00:00:425</td>
<td>00:01:229</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>00:00:088</td>
<td>00:00:320</td>
<td>00:05:550</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>00:00:193</td>
<td>00:00:689</td>
<td>00:21:136</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>00:00:579</td>
<td>00:01:876</td>
<td>02:31:158</td>
</tr>
<tr>
<td>3000</td>
<td>30</td>
<td>00:00:229</td>
<td>00:00:412</td>
<td>00:01:610</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>00:00:297</td>
<td>00:00:639</td>
<td>00:06:447</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>00:00:568</td>
<td>00:01:195</td>
<td>00:22:977</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>00:01:422</td>
<td>00:03:041</td>
<td>02:31:919</td>
</tr>
<tr>
<td>10000</td>
<td>30</td>
<td>00:00:561</td>
<td>00:00:958</td>
<td>00:02:821</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>00:00:656</td>
<td>00:01:447</td>
<td>00:08:969</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>00:01:242</td>
<td>00:02:461</td>
<td>00:28:209</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>00:03:325</td>
<td>00:06:053</td>
<td>02:47:708</td>
</tr>
<tr>
<td>70000</td>
<td>30</td>
<td>00:02:723</td>
<td>00:04:691</td>
<td>00:12:943</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>00:04:240</td>
<td>00:09:244</td>
<td>00:29:783</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>00:06:620</td>
<td>00:14:547</td>
<td>01:07:383</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>00:17:789</td>
<td>00:31:569</td>
<td>04:31:620</td>
</tr>
</tbody>
</table>
## Framework of Packages

<table>
<thead>
<tr>
<th></th>
<th>TAM</th>
<th>CDM</th>
<th>sirt</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main functions</strong></td>
<td>tam, tam.mml, tam.mml.2pl/.3pl</td>
<td>din, gdina, supplementary</td>
<td>gdm</td>
</tr>
<tr>
<td><strong>Standard generics</strong></td>
<td>summary, plot, logLik, anova, residuals</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Quasi-standard</strong></td>
<td>IRT.expectedCounts.*</td>
<td>IRT.factor.scores.<em>, IRT.irfprob.</em></td>
<td>IRT.modelfit.<em>, IRT.posterior.</em></td>
</tr>
</tbody>
</table>
Conclusion
There already are several R packages for IRT analysis. None of which is suitable for LSA.

**TAM** is unmatedly flexible (just a glimpse is presented) and competitive in means of processing speed.

For convenient recovery of its flexibility, **TAM** offers and extends **lavaan**’s model syntax.

**TAM** implements generic functions for objects from a wide range of packages (also beyond the scope of **TAM**, **CDM** and **sirt**).

In the Future:

- keep **TAM** competitive in terms of flexibility and processing speed,
- extend and round up the model syntax,
- extend the quasi standard generics,
- provide a **Vignette**.
Thank you for your attention!

Thomas Kiefer

B|F|E Salzburg
t.kiefer@bifie.at
www.bifie.at/user/kiefer-thomas


