

Package: CbKST (via r-universe)

September 15, 2026

Type Package

Title Competence-Based Knowledge Space Theory

Date 2026-07-16

Version 0.1-1

Description Competence-based knowledge space theory (CbKST) is an extension of knowledge space theory (KST) modeling the latent skills and competencies underlying the observable response behaviour as described by Heller & Stefanutti (2024) [doi:10.1142/9789811280481_0001](https://doi.org/10.1142/9789811280481_0001). The package focuses on the mappings between competence and performance level (skill (multi) map, problem function etc.).

License GPL-3

Depends R (>= 4.5.0)

Imports readODS, openxlsx, tools, kstMatrix (>= 2.3-4)

Suggests litedown

Encoding UTF-8

LazyData true

VignetteBuilder litedown

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Cord Hockemeyer [aut, cre]

Maintainer Cord Hockemeyer <cord.hockemeyer@uni-graz.at>

Config/pak/sysreqs
cmake libglpk-dev make libicu-dev libsvg2-dev libuv1-dev libxml2-dev libx11-dev

Repository <https://chockemeyer.r-universe.dev>

Date/Publication 2026-07-16 06:50:02 UTC

RemoteUrl <https://github.com/cran/CbKST>

RemoteRef HEAD

RemoteSha bd81c5d12c5ec98604eedf55c790daf650360732

Contents

cbkst_comp2perf	2
cbkst_competencestructure	3
cbkst_perf2comp	3
cbkst_performancestructure	4
cbkst_problemfunction	5
cbkst_simple_perf2comp	6
exampledata	6
read_skillmultimap	7
Index	8

cbkst_comp2perf	<i>Determine the performance state delineated by a competence state</i>
-----------------	---

Description

cbkst_comp2perf() determines the performance state, i.e. the subset of items solvable for a person in a given competence state.

Usage

cbkst_comp2perf(comp, multimap)

Arguments

- comp A binary vector representing the competence state
- multimap A data frame of class cbkst_skillmultimap

Value

A binary vector representing the corresponding performance state

See Also

Other Mapping states: [cbkst_perf2comp\(\)](#), [cbkst_simple_perf2comp\(\)](#)

Examples

cbkst_comp2perf(c(1,1,0,0), exampledata\$mu)

cbkst_competencestructure

Determine the skill structure delineated by a skill multimap and an optional performance structure

Description

cbkst_performancestructure() determines the performance state, i.e. the Determine the structure of all skill states producing some performance state

Usage

```
cbkst_competencestructure(multimap, perf = NULL)
```

Arguments

multimap	A data frame of class cbkst_skillmultimap
perf	Optional performance structure as kmstructure object

Value

A kmstructure object representing the resulting skill structure

See Also

Other Mapping structures: [cbkst_performancestructure\(\)](#), [cbkst_problemfunction\(\)](#)

Examples

```
cbkst_competencestructure(exampladata$mu)
```

cbkst_perf2comp

Determine the competence state behind a given performance state

Description

cbkst_perf2comp() determines competence states underlying a given performance state under the condition of a skill multimap and optionally a competence structure.

Usage

```
cbkst_perf2comp(perf, multimap, comp = NULL)
```

Arguments

perf	A named binary vector representing the performance state
multimap	A data frame of class cbkst_skillmultimap
comp	Optional constraining competence structure

Value

A matrix representing the minimal competence states for item mastery according to perf

See Also

Other Mapping states: [cbkst_comp2perf\(\)](#), [cbkst_simple_perf2comp\(\)](#)

Examples

```
perf <- c(1,1,1,0,0)
names(perf) <- c("z", "y", "x", "w", "v")
cbkst_perf2comp(perf, exampledata$multi)
```

cbkst_performancestructure

*Determine the performance structure delineated by a skill multimap
and an optional competence structure*

Description

cbkst_performancestructure() determines the performance state, i.e. the Determine the structure of all performance states reachable from some skill state

Usage

```
cbkst_performancestructure(multimap, comp = NULL)
```

Arguments

multimap	A data frame of class cbkst_skillmultimap
comp	Optimal competence structure as kmstructure object

Value

A kmstructure object representing the resulting performance structure

See Also

Other Mapping structures: [cbkst_competencestructure\(\)](#), [cbkst_problemfunction\(\)](#)

Examples

```
cbkst_performancestructure(exampladata$multi)
```

cbkst_problemfunction *Determine the problem function for a skill multimap and an optional competence structure*

Description

cbkst_problemfunction() determines a problem function. It is basically stored as a matrix with one row for each competence state. These rows consist of two parts, a binary vector describing the competence state followed by a binary vector representing the corresponding knowledge state (see Details).

Usage

```
cbkst_problemfunction(multimap, comp = NULL)
```

Arguments

multimap	A data frame of class cbkst_skillmultimap
comp	Optional constraining competence structure

Value

A cbkst_problemfunction object (see Details)

If comp is NULL, the power set of competences is taken.

cbkst_problemfunction objects are lists consisting of three elements, nos (number of skills), nop (number of problems), and pf, (problem function matrix as described above).

See Also

Other Mapping structures: [cbkst_competencestructure\(\)](#), [cbkst_performancestructure\(\)](#)

Examples

```
cbkst_problemfunction(exampladata$multi)
```

`cbkst_simple_perf2comp`
Determine the competence state behind a given performance state

Description

`cbkst_simnple_perf2comp()` determines competence state underlying a given performance state under the condition of a skill map.

Usage

```
cbkst_simple_perf2comp(perf, skillmap)
```

Arguments

<code>perf</code>	A binary vector representing the performance state
<code>skillmap</code>	A data frame of class <code>cbkst_skillmap</code>

Value

A binary vector representing the corresponding competence state

See Also

Other Mapping states: [cbkst_comp2perf\(\)](#), [cbkst_perf2comp\(\)](#)

Examples

```
perf <- c(1,1,1,0,0)
names(perf) <- c("z", "y", "x", "w", "v")
cbkst_simple_perf2comp(perf, exampledata$mu)
```

`exampledata`
Small example data set

Description

This small example data set contains a skill map `mu`, a skill multimap `multi`, a competence space `cspace`, and a performance space `pspace`.

Usage

```
exampledata
```

Format

List with four elements: mu (skill map), multi (skill multimap), cspace (competence space), and pspace (performance space)

read_skillmultimap	<i>Read a skill multimap from file</i>
--------------------	--

Description

read_skillmultimap() reads a skill multimap as data frame.

Usage

```
read_skillmultimap(file)
```

Arguments

file	Filename to read
------	------------------

Value

Data frame of class cbkst_skillmultimap. In case of a skill map, it also has the class cbkst_skillmap

Data file and resulting data table have the following format: The first column contains the item and the subsequent rows denote for each skill if it is contained in $\mu(q)$. ODS and XLSX files are recognised.

Examples

```
fpath <- system.file("extdata", "skillmap.ods", package="CbKST")
read_skillmultimap(fpath)
```

Index

- * **Files**

- read_skillmultimap, 7

- * **Mapping states**

- cbkst_comp2perf, 2

- cbkst_perf2comp, 3

- cbkst_simple_perf2comp, 6

- * **Mapping structures**

- cbkst_competencestructure, 3

- cbkst_performancestructure, 4

- cbkst_problemfunction, 5

- * **datasets**

- exampledata, 6

cbkst_comp2perf, 2

cbkst_comp2perf(), 4, 6

cbkst_competencestructure, 3

cbkst_competencestructure(), 4, 5

cbkst_perf2comp, 3

cbkst_perf2comp(), 2, 6

cbkst_performancestructure, 4

cbkst_performancestructure(), 3, 5

cbkst_problemfunction, 5

cbkst_problemfunction(), 3, 4

cbkst_simple_perf2comp, 6

cbkst_simple_perf2comp(), 2, 4

exampledata, 6

read_skillmultimap, 7