# Package: mycaas (via r-universe)

**Type** Package

**Title** My Computerized Adaptive Assessment

**Version** 0.0.1

**Maintainer** Andrea Brancaccio <andreabrancaccio01@gmail.com>

**Description** Implementation of adaptive assessment procedures based on Knowledge Space Theory (KST) and Formal Psychological Assessment (FPA) frameworks. An adaptive assessment is a type of evaluation that adjusts the difficulty and nature of subsequent questions based on the test-taker's responses to previous ones. The package contains functions to perform and simulate an adaptive assessment. Moreover, it is integrated with two 'Shiny' interfaces, making it both accessible and user-friendly. The package has been funded by the European Union - NextGenerationEU and by the Ministry of University and Research (MUR), National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.5, project "RAISE - Robotics and AI for Socio-economic Empowerment" (ECS00000035).

**License** MIT + file LICENSE

**Depends** R (>= 2.10)

**Imports** ggplot2, graphics, grDevices, igraph, methods, rlang, rPref, shiny, stats

**Suggests** pks, spelling, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.1

**Language** en-US

**Config/pak/sysreqs** libglpk-dev make libxml2-dev zlib1g-dev

**Repository** https://brancaccioandrea.r-universe.dev

**RemoteUrl** https://github.com/brancaccioandrea/mycaas

**RemoteRef** HEAD

**RemoteSha** dc303588224cbff1b982ff10ea824fd1a57ff2a7

# Contents

---

AA_knowledge_test          *Example of a test to showcase the Adaptive Assessment tools*

---

### Description

The dataset contains a six items test concerning basic knowledge on computerized adaptive assessment. This test has been developed during the creation of this R package.

### Usage

```
AA_knowledge_test
```

### Format

A assessment object based on six multiple choice items.

**states**  Structure with six items and twelve states.

**likelihood**  Uniformly distributed across the states.

**Stopping_rule**  The stopping rules is "likelihood_maximization" with a criterion of .5.

**Question**  Six multiple choice items.

**adaptive**  The value is set on TRUE allowing adaptive administration.

**repetition**  The value is set on FALSE not allowing items repetition.

### Source

Created to serve as an Example for this package.

### Examples

```
data(AA_knowledge_test) #Lazy loading
```

---

| assessment | *Assessment function* |
|---|---|

---

### Description

Function that performs computerized assessment

### Usage

```
assessment(
  likelihood = NA,
  states,
  beta,
  eta,
  questioning_rule = "half_split",
  termination = "likelihood_maximization",
  SC = 0.5,
  N_items = NULL,
  ki = 1,
  textq = NULL,
  textr = NULL,
  repetition = FALSE,
  adaptive = TRUE,
  simulation = TRUE,
  interactiveplot = FALSE
)
```

### Arguments

| | |
|---|---|
| likelihood | A vector of the likelihood distribution on the states in the structure. |
| states | A state-by-problem matrix representing the structure, where an element is one if the item is included in the state, and zero otherwise. |
| beta | Vector of careless error probabilities. |
| eta | Vector of lucky guess error probabilities. |
| questioning_rule | A function which is used a questioning rules for the assessment. the default questioning rule is 'half_split'. |
| termination | Define and select one of the termination criteria: "likelihood_maximization" the assessment terminates when the likelihood of a knowledge state in a knowledge structure became higher of the termination criteria (Heller, and Repitsch, 2012). "items_discrimination" the assessment terminates if the marginal likelihood of all the items is outside the interval of the stopping criteria (Donadello, Spoto, Sambo, Badaloni, Granziol, Vidotto, 2017). |
| SC | The Stopping criterion for the assessment is a numeric vector of values between 0 and 1. When the "termination" parameter is "likelihood_maximization" this is |

a single scalar that corresponds to the likelihood that a knowledge state needed to terminates the assessment. When the "termination" parameter is "items_discrimination" this is a numeric vector of length two, the assessment terminate if the the marginal likelihood of each item is outside of the interval between the two elements.

| | |
|---|---|
| N_items | Number of items in the test. Optional entry in case of adaptive = FALSE. |
| ki | A number indicating the row in the structure to simulate as the true knowledge states. |
| textq | A character vector containing the text of the questions. |
| textr | A list containing for each question the correct and incorrect answers. |
| repetition | Logical value. When the value is TRUE the assessment procedure is allowed to administer the same item more then one time. |
| adaptive | Logical value. When the value is TRUE the assessment proceed with an adaptive procedure, otherwise the items presentation is randomized |
| simulation | Logical value. When the value is TRUE the assessment proceed with simulating a complete assessment with a user knowledge state define in variable ki. Otherwise the assessment collect answer from the user. |
| interactiveplot | |
| | Logical value. When the value is TRUE the knowledge structure is plot using a color thermometer scale to represents the likelihood of the states. |

## Value

The outcome of a single assessment.

## References

Doignon, J.-P., & Falmagne, J.-C. (1999). Knowledge spaces. Springer.

Donadello, I., Spoto, A., Sambo, F., Badaloni, S., Granziol, U., & Vidotto, G. (2017). ATS-PD: An adaptive testing system for psychological disorders. Educational and psychological measurement, 77(5), 792-815.

Heller, J., & Repitsch, C. (2012). Exploiting prior information in stochastic knowledge assessment. Methodology.

## Examples

```
# Example 1: From Brancaccio,de Chiusole, Stefanutti (2023)
# Consider the knowledge space and the parameters used in Brancaccio,
# de Chiusole, Stefanutti (2023) in Example 1

states<-matrix(c( 0,0,0,0,0,
                  0,0,0,0,1,
                  0,0,1,0,1,
                  0,0,0,1,1,
                  0,0,1,1,1,
                  1,0,1,0,1,
                  0,1,0,1,1,
                  1,0,1,1,1,
```

```
                   0,1,1,1,1,
                   1,1,0,1,1,
                   1,1,1,1,1), byrow=TRUE, ncol=5)
beta  <-c(.004,.03,.02,.01,.007)
eta    <-c(5e-06, 5e-05, 4e-05,.007,.08)
likelihood <-rep(1,nrow(states))/nrow(states)
assessment(likelihood,states,beta,eta)


# Example 2: Random items presentation

assessment(N_items = 5 ,adaptive = FALSE)

# Example 3: Interactive mode on in the console

TextQuestion<-c("I frequently use computers.",
"I know how to build an assessment instrument (i.e., a paper-and-pencil questionnaire).",
"I can code and implement a 'for' loop (independently of its purpose).",
"When you administer an adaptive assessment, each question: ",
"I usually programme or code my assessment tools ",
"I add my assessment in a programme (i.e , google form) ")

TextResponse<-list(
correct=list("Yes","Yes","Yes","depend from the previous questions","Yes","Yes"),
incorrect=list(c("No"),c("No"),c("No"),c("is independent from the others questions",
"depend from the gender of the interviewed"),c("No"),c("No"))
)

map <- matrix(c(1,0,0,0,
                0,1,0,0,
                1,0,1,0,
                0,1,0,1,
                1,1,1,0,
                1,1,0,0), ncol=4)

skillmap<-cbind(1:6,map)

states<-pks::delineate(skillmap)$K
rownames(states)<-NULL

eta <- rep(0.1,ncol(states))
beta <- rep(0.1,ncol(states))
likelihood <-rep(1,nrow(states))/nrow(states)

## Not run: token <- assessment(likelihood,states,beta,eta,textq=TextQuestion,
                      textr=TextResponse,simulation=FALSE)
## End(Not run)
```

---

half_split                    *Questioning rule (half split)*

---

**Description**

Rule to select the most informative item at each step of the assessment.

**Usage**

```
half_split(likelihood, states, beta, eta, Q_pool = NA)
```

**Arguments**

| | |
|---|---|
| likelihood | A vector of the likelihood distribution on the states in the structure. |
| states | A state-by-problem matrix representing the structure, where an element is one if the item is included in the state, and zero otherwise. |
| beta | Vector of careless error probabilities. |
| eta | Vector of lucky guess error probabilities. |
| Q_pool | A vector contains the pool of items for the assessment in this moment of the procedure. |

**Value**

The item that maximizes the information (for details see, Doignon and Falmagne, 2012).

**References**

Doignon, J.-P., & Falmagne, J.-C. (1999). Knowledge spaces. Berlin: Springer.

**Examples**

```
# Consider the knowledge space and the parameters used in Brancaccio,
# de Chiusole, Stefanutti (2023) in Example 1

states<-matrix(c( 0,0,0,0,0,
                  0,0,0,0,1,
                  0,0,1,0,1,
                  0,0,0,1,1,
                  0,0,1,1,1,
                  1,0,1,0,1,
                  0,1,0,1,1,
                  1,0,1,1,1,
                  0,1,1,1,1,
                  1,1,0,1,1,
                  1,1,1,1,1), byrow=TRUE, ncol=5)
beta  <-c(.004,.03,.02,.01,.007)
eta   <-c(5e-06, 5e-05, 4e-05,.007,.08)
likelihood <-c(0,0,0,0,0,1/3,0,1/3,0,0,1/3)
Q_pool <- c(2,4,5)
half_split(likelihood,states,beta,eta,Q_pool)
```

---

performance_simulation

*Performance indexes*

---

### Description

Simulation producing efficiency and accuracy performance curves and indexes based on simulated assessment of all the states in structure

### Usage

```
performance_simulation(
  states,
  beta,
  eta,
  likelihood = NA,
  questioning_rule = "half_split",
  nrep = 1
)
```

### Arguments

| | |
|---|---|
| states | A state-by-problem matrix representing the structure, where an element is one if the item is included in the state, and zero otherwise. |
| beta | Vector of careless error probabilities. |
| eta | Vector of lucky guess error probabilities. |
| likelihood | A vector of the likelihood for each state, if omitted the initial likelihood is assumed equally distributed. |
| questioning_rule | |
| | A function which is used a questioning rules for the assessment. the default questioning rule is 'half_split'. |
| nrep | Number of times in which each state is simulated. The default value is one. |

### Value

Efficiency and Accuracy curves plots.

### Examples

```
# Consider the knowledge space and the parameters used in Brancaccio,
# de Chiusole, Stefanutti (2023) in Example 1

states<-matrix(c( 0,0,0,0,0,
                  0,0,0,0,1,
                  0,0,1,0,1,
                  0,0,0,1,1,
```

```
                         0,0,1,1,1,
                         1,0,1,0,1,
                         0,1,0,1,1,
                         1,0,1,1,1,
                         0,1,1,1,1,
                         1,1,0,1,1,
                         1,1,1,1,1), byrow=TRUE, ncol=5)
beta  <-c(.004,.03,.02,.01,.007)
eta   <-c(5e-06, 5e-05, 4e-05,.007,.08)
performance_simulation(states,beta,eta)
```

---

```
plot.assessment              Plot (Method)
```

---

## Description

Function used to plot an assessment object.

## Usage

```
## S3 method for class 'assessment'
plot(x, bg_color = NULL, verices_color = "black", ...)
```

## Arguments

| | |
|---|---|
| x | Assessment object to display. |
| bg_color | Background color. The default is white. |
| verices_color | Color of the nodes label. |
| ... | Additional arguments affecting the summary produced. |

## Value

The function returns the graph representation of the structure.

## Examples

```
#  Example: Random items presentation

token<- assessment(N_items = 5 ,adaptive = FALSE)
plot(token)
```

---

print_question                    *Print the question*

---

## Description

Function to print in the console the item and collect the answer.

## Usage

```
print_question(
  TextQuestion,
  TextResponse = list(correct = "correct", incorrect = c("wrong1", "wrong2", "wrong3"))
)
```

## Arguments

TextQuestion      A character contain the text of the question.

TextResponse      A list containing for each question the correct and incorrect answers.

## Value

The answers of the question as numeric (1) if is correct and (0) if is false.

---

run_Assessment                    *run Assessment shinyApp*

---

## Description

This function run a shinyApp in which the is available: iii) The GUI implementation of the adaptive assessment

## Usage

```
run_Assessment(file = NULL)
```

## Arguments

file              Logical value.

## Value

The results of the assessment

## Examples

```
# Try the test example yourself
data(AA_knowledge_test)
if(interactive()){
run_Assessment(AA_knowledge_test)}
```

---

run_Practice                 *run Practice Assessment shinyApp*

---

## Description

This function run a shinyApp in which the following pages are available: i) a brief introduction of the assessment tools; ii) the GUI to create "assessment" files from a "csv" of items; iii) the GUI implementation of the adaptive assessment; and iv) the GUI implementation of the outcome.

## Usage

```
run_Practice()
```

## Value

The results of the assessment

## Examples

```
# Try to build your test
if(interactive()){
run_Practice()}
```

---

stopping_criterion       *Stopping rule*

---

## Description

Rule to decide when terminate the assessment

## Usage

```
stopping_criterion(
  likelihood,
  states,
  termination = "likelihood_maximization",
  SC = c(0.8)
)
```

## Arguments

| | |
|---|---|
| likelihood | A vector of the likelihood distribution on the states in the structure. |
| states | A state-by-problem matrix representing the structure, where an element is one if the item is included in the state, and zero otherwise. |
| termination | Define and select one of the termination criteria: "likelihood_maximization" the assessment terminates when the likelihood of a knowledge state in a knowledge structure became higher of the termination criteria (Heller, and Repitsch, 2012). "items_discrimination" the assessment terminates if the marginal likelihood of all the items is outside the interval of the stopping criteria (Donadello, Spoto, Sambo, Badaloni, Granziol, Vidotto, 2017). |
| SC | The Stopping criterion for the assessment is a numeric vector of values between 0 and 1. When the "termination" parameter is "likelihood_maximization" this is a single scalar that corresponds to the likelihood that a knowledge state needed to terminates the assessment. When the "termination" parameter is "items_discrimination" this is a numeric vector of length two, the assessment terminate if the the marginal likelihood of each item is outside of the interval between the two elements. |

## Value

Return TRUE if the assessment should terminates under the criteria, otherwise FALSE

## References

Donadello, I., Spoto, A., Sambo, F., Badaloni, S., Granziol, U., & Vidotto, G. (2017). ATS-PD: An adaptive testing system for psychological disorders. Educational and psychological measurement, 77(5), 792-815.

Heller, J., & Repitsch, C. (2012). Exploiting prior information in stochastic knowledge assessment. Methodology.

## Examples

```
# Consider the knowledge space and the parameters used in Brancaccio,
# de Chiusole, Stefanutti (2023) in Example 1

states<-matrix(c( 0,0,0,0,0,
                  0,0,0,0,1,
                  0,0,1,0,1,
                  0,0,0,1,1,
                  0,0,1,1,1,
                  1,0,1,0,1,
                  0,1,0,1,1,
                  1,0,1,1,1,
                  0,1,1,1,1,
                  1,1,0,1,1,
                  1,1,1,1,1), byrow=TRUE, ncol=5)
beta  <-c(.004,.03,.02,.01,.007)
eta   <-c(5e-06, 5e-05, 4e-05,.007,.08)
likelihood <-c(0,0,0,0,0,0,0,.49,0,0,.51)
#stopping criterion based on the likelihood mode
```

```
stopping_criterion(likelihood,states, termination="likelihood_maximization" ,SC=c(0.5))
#stopping criterion based on the items marginal probabilities
stopping_criterion(likelihood,states, termination="items_discrimination" ,SC=c(0.2,0.8))
```

---

summary.assessment          *Summary (Method)*

---

### Description

Function used to produce result summaries of an assessment object.

### Usage

```
## S3 method for class 'assessment'
summary(object, ...)
```

### Arguments

| | |
|---|---|
| object | Assessment object to display |
| ... | Additional arguments affecting the summary produced. |

### Value

The function return a summary of the information in the a assessment object

### Examples

```
#   Example of random items presentation

token<- assessment(N_items = 5 ,adaptive = FALSE, ki=15)
summary(token)
```

---

updating                    *Updating rules*

---

### Description

Multiplicative rule as in Falmagne and Doignon 2010 (Chapter 13, Section 10)

### Usage

```
updating(likelihood, states, q, r_q, beta, eta)
```

## Arguments

| | |
|---|---|
| likelihood | A vector of the likelihood distribution on the states in the structure. |
| states | A state-by-problem matrix representing the structure, where an element is one if the item is included in the state, and zero otherwise. |
| q | Last administered item |
| r_q | Observed response to item q |
| beta | Vector of careless error probabilities |
| eta | Vector of lucky guess error probabilities |

## Value

The updated likelihood distribution on the knowledge states

## Examples

```
# Let consider the knowledge space and the parameters used in Brancaccio,
# de Chiusole, Stefanutti (2023) in Example 1

states<-matrix(c( 0,0,0,0,0,
                  0,0,0,0,1,
                  0,0,1,0,1,
                  0,0,0,1,1,
                  0,0,1,1,1,
                  1,0,1,0,1,
                  0,1,0,1,1,
                  1,0,1,1,1,
                  0,1,1,1,1,
                  1,1,0,1,1,
                  1,1,1,1,1), byrow=TRUE, ncol=5)
beta  <-c(.004,.03,.02,.01,.007)
eta   <-c(5e-06, 5e-05, 4e-05,.007,.08)
likelihood_0 <-rep(1,nrow(states))/nrow(states)
# Item asked
q = 3
# response observed
r_q = 1
likelihood_1 <- updating(likelihood_0,states,q,r_q,beta,eta)
```

# Index