

Package ‘Keyboard’

January 20, 2025

Type Package

Title Bayesian Designs for Early Phase Clinical Trials

Version 0.1.3

Author Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

Maintainer Xiaomeng Yuan <xiaomeng.yuan@stjude.org>

Imports Iso, ggplot2

Description We developed a package 'Keyboard' for designing single-agent, drug-combination, or phase I/II dose-finding clinical trials. The 'Keyboard' designs are novel early phase trial designs that can be implemented simply and transparently, similar to the 3+3 design, but yield excellent performance, comparable to those of more-complicated, model-based designs (Yan F, Mandrekar SJ, Yuan Y (2017) <doi:10.1158/1078-0432.CCR-17-0220>, Li DH, Whitmore JB, Guo W, Ji Y. (2017) <doi:10.1158/1078-0432.CCR-16-1125>, Liu S, Johnson VE (2016) <doi:10.1093/biostatistics/kxv040>, Zhou Y, Lee JJ, Yuan Y (2019) <doi:10.1002/sim.8475>, Pan H, Lin R, board' package provides tools for designing, conducting, and analyzing single-agent, drug-combination, and phase I/II dose-finding clinical trials. For more details about how to use this package, please refer to Li C, Sun H, Cheng C, Tang L, and Pan H. (2022) ``A software tool for both the maximum tolerated dose and the optimal biological dose finding trials in early phase designs''. Manuscript submitted for publication.

Depends R (>= 3.4.0), Rcpp (>= 0.12.0), methods

License GPL-2

Encoding UTF-8

RoxygenNote 7.2.1

Repository CRAN

NeedsCompilation no

Date/Publication 2022-08-11 10:10:17 UTC

Contents

decision.finding	2
----------------------------	---

example.kb	3
get.boundary.comb.kb	4
get.boundary.kb	7
get.decision.obd.kb	10
get.decision.obd2.kb	14
get.oc.comb.kb	18
get.oc.kb	20
get.oc.obd.kb	24
get.oc.obd2.kb	28
next.comb.kb	31
plot_kb	33
select.mtd.comb.kb	35
select.mtd.kb	38
select.obd.kb	40
summary_kb	43

Index 46

decision.finding	<i>Return Dose Escalation or De-escalation Decision Given Number of Toxicity and Responses</i>
------------------	--

Description

This function returns the decision given the decision table, the total number of patients, the number of patients who experienced toxicity and the number of responses.

Usage

```
decision.finding(out.matrix, n, t, r)
```

Arguments

out.matrix	A decision matrix.
n	The number of enrolled subjects.
t	The number of subjects experienced dose limiting toxicities (DLT).
r	The number of responses.

Details

decision.finding() is a helper function to get.decision.obd.kb() and get.decision.obd2.kb(). decision.finding() returns the decision given the decision table, the total number of patients, the number of patients who experienced toxicity and the number of responses. If the number of subjects enrolled is n, then the number of possible combination for subjects who experienced toxicity and efficacy is (n+1) x (n+1). For n = 30, the number of entries is 961. Finding the decision given a scenario is tedious. decision.finding() resolves this challenge by helping users to obtain the decision easily with the input of the decision table (output from get.decision.obd.kb() and get.decision.obd2.kb()), the total number of patients, the number of patients who experienced toxicity and the number of responses.

Value

The function `decision.finding()` returns the decision.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

Examples

```
output.matrix <- get.decision.obd.kb( toxicity.low = 0.15,
                                     toxicity.moderate= 0.25,
                                     toxicity.high = 0.35,
                                     efficacy.low = 0.25,
                                     efficacy.moderate = 0.45,
                                     efficacy.high = 0.65,
                                     target.toxicity=0.20,
                                     target.efficacy=0.40,
                                     cohortsize=3, ncohort=10)$decision.matrix
decision <- decision.finding (out.matrix=output.matrix, n=6, t=1, r=3)
decision
```

example.kb

Illustration of the Keyboard Design (Single-agent)

Description

This function serves solely as an example to illustrate how the Keyboard design determines dosing decisions when conducting a single-agent trial.

Usage

```
example.kb(center, half_width, s1, s2, a, b)
```

Arguments

center	The center of the target key (between 0 and 1).
half_width	The width, or tolerable deviation, to each side of the center (half of the target key's width).
s1	The lower boundary of the strongest key.
s2	The upper boundary of the strongest key.
a	The alpha parameter of the beta distribution.
b	The beta parameter of the beta distribution.

Details

It is straightforward to visualize how the Keyboard design determines whether to stay at the current dose, escalate, or de-escalate. The "target key" (shown in blue) is the proper dosing interval determined for the trial by the investigators. The "strongest key" (in red) is the interval with the highest posterior probability (given the dosage data observed thus far) of the current dose's true DLT rate. If the strongest key is to the left of the target key, then we escalate the dose; if the strongest key is to the right of the target key, then we de-escalate the dose; and if the strongest key is the target key, then we retain the current dose. Thus, posterior probabilities are what directs dosage. Graphically, the strongest key is the one with the largest area under the posterior distribution curve of the DLT rate of the current dose.

Value

A dose decision plot with the posterior distribution of the DLT rate of the current dose and the positions of the target and the strongest keys.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

- Yan F, Mandrekar SJ, Yuan Y. Keyboard: A Novel Bayesian Toxicity Probability Interval Design for Phase I Clinical Trials. *Clinical Cancer Research*. 2017; 23:3994-4003. <http://clincancerres.aacrjournals.org/content/23/15/3text.pdf>
- Pan H, Lin R, Yuan Y. Keyboard design for phase I drug-combination trials. *Contemporary Clinical Trials*. 2020. <https://doi.org/10.1016/j.cct.2020.105972>

Examples

```
## Clear all plots before switching between graphical parameters
opar <- par(no.readonly = TRUE)
on.exit(par(opar))
par(mfrow = c(3, 1)) # for many plots in the same screen
par(mar = c(5, 5, 2, 2)) # for only one plot per page

example.kb(center = 0.19, half_width = 0.03, s1 = 0.4, s2 = 0.46, a = 3, b = 4) # de-escalation
example.kb(center = 0.19, half_width = 0.03, s1 = 0.04, s2 = 0.1, a = 2, b = 12) # escalation
example.kb(center = 0.19, half_width = 0.03, s1 = 0.16, s2 = 0.22, a = 2, b = 5) # stay
```

get.boundary.comb.kb *Dose Escalation or De-escalation Boundaries for Drug-combination Trials*

Description

This function generates the optimal dose escalation or de-escalation boundaries when conducting a drug-combination trial with the Keyboard design.

Usage

```

get.boundary.comb.kb(
  target,
  ncohort,
  cohortsize,
  n.earlystop = 100,
  marginL = 0.05,
  marginR = 0.05,
  cutoff.eli = 0.95,
  offset = 0.05,
  extrasafe = TRUE
)

```

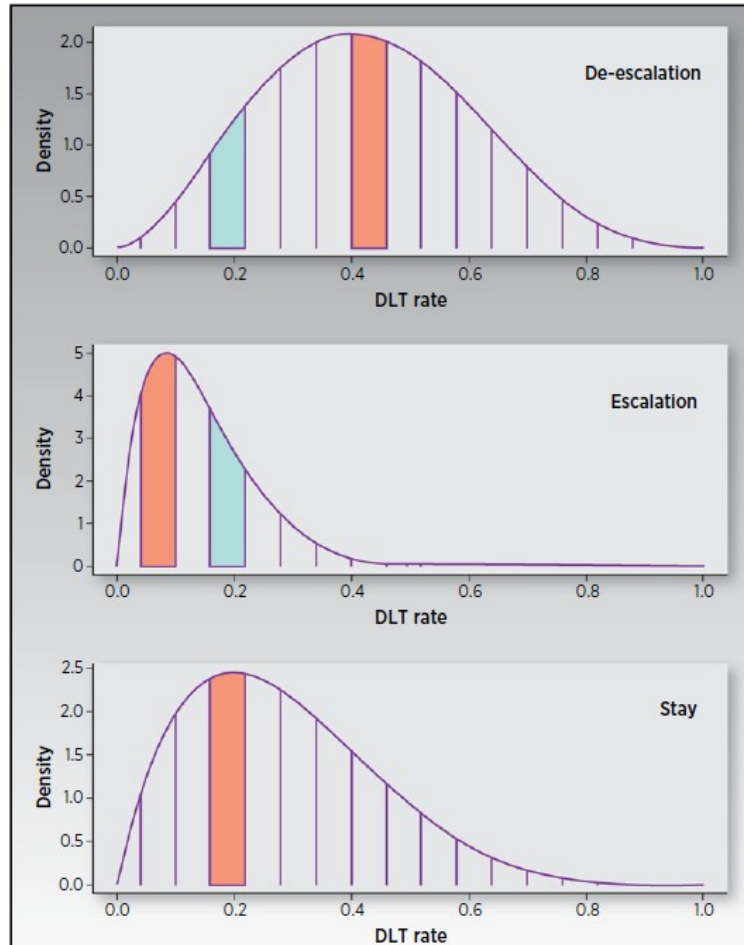
Arguments

target	The target dose-limiting toxicity (DLT) rate.
ncohort	A scalar specifying the total number of cohorts in the trial.
cohortsize	The number of patients in the cohort.
n.earlystop	The early stopping parameter. If the number of patients treated at the current dose reaches n.earlystop, then stop the trial and select the MTD based on the observed data. The default value is 100.
marginL	The difference between the target and the lower bound of the "target key" (proper dosing interval) to be defined. The default is 0.05.
marginR	The difference between the target and the upper bound of the "target key" (proper dosing interval) to be defined. The default is 0.05.
cutoff.eli	The cutoff to eliminate an overly toxic dose and all higher doses for safety. The recommended value is 0.95.
offset	A small positive number (between 0 and 0.5) to control how strict the stopping rule is when extrasafe=TRUE. A larger value leads to a stricter stopping rule. The default value is 0.05.
extrasafe	Set extrasafe=TRUE to impose a stricter stopping rule for extra safety, expressed as the stopping boundary value in the result.

Details

The Keyboard design relies on the posterior distribution of the toxicity probability to guide dosage. To determine whether to escalate or de-escalate the dose, given the observed data at the current dose, we first identify an interval that has the highest posterior probability, referred to as the "strongest key". This key represents where the true dose-limiting toxicity (DLT) rate of the current dose is most likely located. If the strongest key is to the left of the "target key", then we escalate the dose because the data suggest that the current dose is likely to underdose patients; if the strongest key is to the right of the target key, then we de-escalate the dose because the observed data suggest that the current dose is likely to overdose the patients; and if the strongest key is the target key, then we retain the current dose because the observed data support that the current dose is most likely to be

in the proper dosing interval. Graphically, the strongest key is the one with the largest area under the posterior distribution curve of the DLT rate of the current dose.



An attractive feature of the Keyboard design is that its dose escalation and de-escalation rules can be tabulated before the onset of the trial. Thus, when conducting the trial, no calculation or model fitting is needed, and we need to count only the number of DLTs observed at the current dose; the decision to escalate or de-escalate the dose is based on the pre-tabulated decision rules.

Given all observed data, we use matrix isotonic regression to obtain an estimate of the toxicity rate of the combination of dose level j of drug A and dose level k of drug B and to select as the MTD the combination with the toxicity estimate that is closest to the target. When there are ties, we randomly choose one as the MTD.

For patient safety, we apply the following Bayesian overdose control rule after each cohort: if at least 3 patients have been treated at the given dose and the observed data indicate that the probability of the current combination dose's toxicity rate being above the target toxicity rate is more than 95%, then we exclude this dose and beyond to avoid exposing future patients to these overly toxic doses. The probability threshold can be specified with `cutoff.e1i`. If the lowest dose combination (1, 1) is overly toxic, then the trial terminates early, and no dose is selected as the MTD.

Value

The function returns a matrix, including the dose escalation, de-escalation, and elimination boundaries.

Note

In most clinical applications, the target DLT rate is often a rough guess, but finding a dose level with a DLT rate reasonably close to the target rate (which ideally would be the MTD) is what interests the investigator.

References

Yan F, Mandrekar SJ, Yuan Y. Keyboard: A Novel Bayesian Toxicity Probability Interval Design for Phase I Clinical Trials. *Clinical Cancer Research*. 2017; 23:3994-4003. <http://clincancerres.aacrjournals.org/content/23/15/3994.full.pdf>

Pan H, Lin R, Yuan Y. Keyboard design for phase I drug-combination trials. *Contemporary Clinical Trials*. 2020. <https://doi.org/10.1016/j.cct.2020.105972>

See Also

Other drug-combination functions: [get.oc.comb.kb\(\)](#), [next.comb.kb\(\)](#), [select.mtd.comb.kb\(\)](#)

Examples

```
### Drug-combination trial ###

bound <- get.boundary.comb.kb(target=0.3, ncohort=10, cohortsize=3)
print(bound)
```

get.boundary.kb

Dose Escalation or De-escalation Boundaries for Single-agent Trials

Description

This function generates the optimal dose escalation or de-escalation boundaries when conducting a single-agent trial with the Keyboard design.

Usage

```
get.boundary.kb(
  target,
  ncohort,
  cohortsize,
  marginL = 0.05,
  marginR = 0.05,
  cutoff.eli = 0.95,
  n.earlystop = 100,
```

```

    extrasafe = FALSE,
    offset = 0.05
)

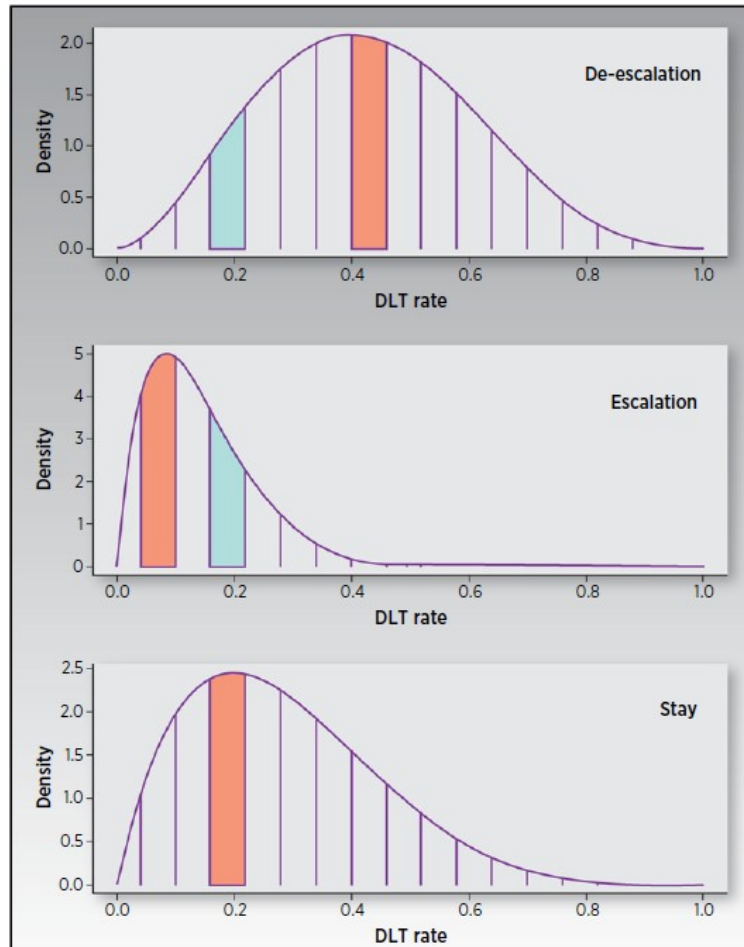
```

Arguments

target	The target dose-limiting toxicity (DLT) rate.
ncohort	The total number of cohorts.
cohortsize	The number of patients in the cohort.
marginL	The difference between the target and the lower bound of the "target key" (proper dosing interval) to be defined. The default is 0.05.
marginR	The difference between the target and the upper bound of the "target key" (proper dosing interval) to be defined. The default is 0.05.
cutoff.eli	The cutoff value to eliminate an overly toxic dose and all higher doses for safety. The recommended value is 0.95.
n.earlystop	The early stopping parameter. If the number of patients treated at the current dose reaches n.earlystop, then stop the trial and select the MTD based on the observed data. The default value is 100.
extrasafe	Set extrasafe=TRUE to impose a stricter stopping rule for extra safety, expressed as the stopping boundary value in the result.
offset	A small positive number (between 0 and 0.5) to control how strict the stopping rule is when extrasafe=TRUE. A larger value leads to a stricter stopping rule. The default value is 0.05.

Details

The Keyboard design relies on the posterior distribution of the toxicity probability to guide dosage. To make a decision of dose escalation or de-escalation, given the observed data at the current dose, we first identify an interval that has the highest posterior probability, referred to as the "strongest key". This key represents where the true dose-limiting toxicity (DLT) rate of the current dose is most likely located. If the strongest key is to the left of the "target key", then we escalate the dose because the data suggest that the current dose is most likely too low; if the strongest key is to the right of the target key, then we de-escalate the dose because the observed data suggest that the current dose is likely too high; and if the strongest key is the target key, then we retain the current dose because the observed data support the notion that the current dose is most likely to be in the proper dosing interval. Graphically, the strongest key is the one with the largest area under the posterior distribution curve of the DLT rate of the current dose.



An attractive feature of the Keyboard design is that its dose escalation and de-escalation rules can be tabulated before the onset of the trial. Thus, when conducting the trial, no calculation or model fitting is needed, and we need to count only the number of DLTs observed at the current dose; the decision to escalate or de-escalate the dose is based on the pre-tabulated decision rules.

Given all observed data, the Keyboard design uses isotonic regression to obtain an efficient statistical estimate of the maximum tolerated dose (MTD) by utilizing the fact that toxicity presumably increases with the dose.

For patient safety, we apply the following Bayesian overdose control rule after each cohort: if at least 3 patients have been treated at the given dose and the observed data indicate that the probability of the current dose's toxicity rate being above the target toxicity rate is more than 95%, then we stop the trial to avoid exposing future patients to these overly toxic doses. The probability threshold can be specified with `cutoff.eli`. When a dose is eliminated, the design recommends the next lower dose for treating the next cohort of patients. If the lowest dose is overly toxic, then the trial terminates early and no dose is selected as the MTD.

Value

The function returns a matrix, which includes the dose escalation, de-escalation, and elimination boundaries.

Note

In most clinical applications, the target DLT rate is often a rough guess, but finding a dose level with a DLT rate reasonably close to the target rate (which ideally would be the MTD) is of interest.

References

Yan F, Mandrekar SJ, Yuan Y. Keyboard: A Novel Bayesian Toxicity Probability Interval Design for Phase I Clinical Trials. *Clinical Cancer Research*. 2017; 23:3994-4003. <http://clincancerres.aacrjournals.org/content/23/15/3994.full.pdf>

See Also

Other single-agent functions: [get.oc.kb\(\)](#), [select.mtd.kb\(\)](#)

Examples

```
### Single-agent trial ###  
  
bound <- get.boundary.kb(target=0.3, ncohort=10, cohortsize=3)  
print(bound)
```

get.decision.obd.kb *Boundary Table and Decision Matrix for Phase I/II Trials.*

Description

This function generates a boundary table and decision matrix for single-agent phase I/II trials using Keyboard design.

Usage

```
get.decision.obd.kb(  
  toxicity.low,  
  toxicity.moderate,  
  toxicity.high,  
  efficacy.low,  
  efficacy.moderate,  
  efficacy.high,  
  target.toxicity,  
  target.efficacy,  
  cohortsize,  
  ncohort,  
  cutoff eli.toxicity = 0.95,
```

```

    cutoff.eli.efficacy = 0.3
)

```

Arguments

`toxicity.low` The upper boundary for the low toxicity interval.
`toxicity.moderate` The upper boundary for the moderate toxicity interval.
`toxicity.high` The upper boundary for the high toxicity interval.
`efficacy.low` The upper boundary for the low efficacy interval.
`efficacy.moderate` The upper boundary for the moderate efficacy interval.
`efficacy.high` The upper boundary for the high efficacy interval.
`target.toxicity` The target DLT rate.
`target.efficacy` The target efficacy rate.
`cohortsize` The number of patients in the cohort.
`ncohort` The total number of cohorts.
`cutoff.eli.toxicity` The cutoff value to eliminate a dose with an unacceptably high toxicity for safety. The default value is 0.95.
`cutoff.eli.efficacy` The cutoff value to eliminate a dose with unacceptably low efficacy. The default value is 0.3.

Details

The Keyboard design relies on the posterior distribution of the toxicity and efficacy to guide dosage transition. To determine whether to escalate or de-escalate the dose given the observed data at the current dose and a boundary table by investigators, we identify the interval that has the highest joint unit probability mass, which we refer to as the winner key. The decision corresponding to the winner key is the decision for dose assignment.

The prespecified boundary table is determined by the investigators. The four subintervals for toxicity are low, moderate, high, and unacceptable. Generating these four intervals requires three investigators-provided boundaries, which can be specified as follows:

1. The low interval for toxicity is (0, `toxicity.low`); `toxicity.low` is the upper boundary for the low toxicity interval, which can be specified by using the argument `toxicity.low`.
2. The moderate interval for toxicity is (`toxicity.low`, `toxicity.moderate`); `toxicity.low` is the upper boundary for the low toxicity interval, which can be specified by using the argument `toxicity.low` and `toxicity.moderate` is the upper boundary for the moderate toxicity interval, which can be specified by using the argument `toxicity.moderate`.

3. The high interval for toxicity is (toxicity.moderate, toxicity.high); toxicity.moderate is the upper boundary for the moderate toxicity interval, which can be specified by using the argument toxicity.moderate and toxicity.high is the upper boundary for the high toxicity interval, which can be specified by using the argument toxicity.high.
4. The unacceptable interval for toxicity is (toxicity.high, 1); toxicity.high is the upper boundary for the high toxicity interval, which can be specified by using the argument toxicity.high.

Similarly, there are four subintervals for efficacy as follows:

1. The low interval for efficacy is (0, efficacy.low); efficacy.low the upper boundary for the low efficacy interval, which can be specified by using the argument efficacy.low.
2. The moderate interval for efficacy is (efficacy.low, efficacy.moderate); efficacy.low is the upper boundary for the low efficacy interval, which can be specified by using the argument efficacy.low and efficacy.moderate is the upper boundary for the moderate efficacy interval, which can be specified by using the argument efficacy.moderate.
3. The high interval for efficacy is (efficacy.moderate, efficacy.high); efficacy.moderate is the upper boundary for the moderate efficacy interval, which can be specified by using the argument efficacy.moderate and efficacy.high is the upper boundary for the high efficacy interval, which can be specified by using the argument efficacy.high.
4. The superb interval for efficacy is (efficacy.high, 1); efficacy.high is the upper boundary for the high efficacy interval, which can be specified by using the argument efficacy.high.

One could suppose that there are d doses in the trial and the current dose is i . Define the number of patients as n_i , the number of patients who experienced toxicity as x_i and the number of responses as y_i . The trial data can be represented as follows:

$$D = (n_i, x_i, y_i), i = 1, \dots, d$$

Assuming that the toxicity probability is p_i and the efficacy probability is q_i at dose level i , the probability unit intervals for toxicity and efficacy can be partitioned into subintervals (a, b) and (c,d). (a, b) is the subinterval for toxicity probability, and (c, d) is the subinterval for efficacy probability. (a, b) x (c,d) is one combination interval. There are 16 combination intervals in total given the 4 subintervals for toxicity rate and the 4 subintervals for efficacy intervals. Investigators would then be required to provide 16 decisions that correspond to these 16 combination intervals. Decision "D" denotes de-escalation, so the next cohort of patients will be treated at the next lower dose level. Decision "E" denotes escalation, so the next cohort of patients will be treated at the next higher dose level. Decision "S" denotes stay, so the next cohort of patients will be treated at the current dose level. The following is an example of a prespecified boundary table with a target toxicity rate of 0.2 and a target efficacy rate of 0.4:

		Efficacy.low (0,0.25)	Efficacy.moderate (0.25,0.45)	Efficacy.high (0.45,0.65)	Efficacy.superb (0.65,1)
Toxicity.low	(0,0.15)	E	E	E	E
Toxicity.moderate	(0.15,0.25)	E	E	E	S
Toxicity. high	(0.25,0.35)	D	S	S	S
Toxicity. unacceptable	(0.35,1.0)	D	D	D	D

For example, the combination interval $(0.25, 0.35) \times (0, 0.25)$ corresponds to decision "D". Therefore the next cohort of patients will be treated at the next lower level if the observed toxicity rate of the current dose falls in $(0.25, 0.35)$ and the observed efficacy rate falls in $(0, 0.25)$.

Bayesian rule is used to calculate the joint unit probability mass (JUPM) for the toxicity and efficacy combination intervals. For a given combination interval, JUPM is calculated as follows:

$$JUPM_{(a,b)}^{(c,d)} = Pr p_j \in (a, b), q_j \in (c, d) | D / (b - a) * (d - c)$$

$Pr p_j \in (a, b), q_j \in (c, d) | D$ is the posterior probability of p_i and q_i falling in the subinterval (a, b) and (c, d) . Assume the priors for both p_i and q_i follow independent beta distributions $beta(\alpha_p, \beta_p)$ and $beta(\alpha_q, \beta_q)$ independently. The posterior distributions for p_i and q_i are $beta(\alpha_p + x_i, \beta_p + n_i - x_i)$ and $beta(\alpha_q + y_i, \beta_q + n_i - y_i)$. Using these posterior distributions, calculate the JUPM for all 16 combination intervals and find the winning combination interval (a^*, b^*) and (c^*, d^*) with the largest JUPM. The decision that corresponds with this winning combination interval is used to treat the next cohort of patients.

Two dose exclusion rules are applied in the trial design: safety rule and futility rule. Safety rule: if at least 3 patients have been treated at a given dose and the observed data indicate that the probability of the current dose's toxicity rate exceeding the target toxicity rate is more than 95%, then we eliminate the current dose and any higher doses from the trial to avoid exposing future patients to unacceptably toxic doses. The probability threshold can be specified with `cutoff.eli.toxicity`. When a dose is eliminated, the design recommends the next lower dose for treating the next cohort of patients. If the lowest dose is unacceptably toxic, then the trial terminates early and no dose is selected as the OBD. This corresponds to a dose assignment of "DUT", which means to de-escalate due to unacceptable high toxicity and exclude the current dose and any dose higher than this dose from the trial.

Futility rule: if at least 3 patients have been treated at a given dose and the observed data indicate that the probability of the current dose's efficacy rate exceeding the target efficacy rate is less than 30%, then we eliminate this dose from the trial to avoid exposing future patients to these futile doses. The probability threshold can be specified with `cutoff.eli.efficacy`. This corresponds to two possible dose assignments: "EUE" and "DUE", which both exclude the current dose from the trial. "EUE" denotes escalation due to unacceptable low efficacy; "DUE" denotes de-escalation due to unacceptable low efficacy.

An attractive feature of the Keyboard design is that its dose escalation and de-escalation rule can be tabulated before implementing the trial. Thus, when conducting the trial, no real-time calculation or model fitting is needed, and we need to count only the number of patients, the number of DLTs, and the number of responses observed at the current dose and the decision to escalate or des-escalate the dose based on the pre-tabulated decision rules.

Value

`get.decision.obd.kb()` returns a prespecified boundary table and a dose assignment decision matrix:

1. Prespecified boundary table (`$boundary.table`)
2. Decision matrix (`$decision.matrix`)

Note

This method is adopted from Li et al (2017)

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Li DH, Whitmore JB, Guo W, Ji Y. Toxicity and efficacy probability interval design for phase I adoptive cell therapy dose-finding clinical trials. *Clinical Cancer Research*. 2017; 23:13-20. <https://clincancerres.aacrjournals.org/content/23/1/13.long>

See Also

Other single-agent phase I/II functions: [get.decision.obd2.kb\(\)](#), [get.oc.obd.kb\(\)](#), [get.oc.obd2.kb\(\)](#), [select.obd.kb\(\)](#)

Examples

```
toxicity.low <- 0.15
toxicity.moderate <- 0.25
toxicity.high <- 0.35
efficacy.low <- 0.25
efficacy.moderate <- 0.45
efficacy.high <- 0.65
target.toxicity <- 0.20
target.efficacy <- 0.40
cohortsize <- 3
ncohort <- 10

decision.obd <- get.decision.obd.kb( toxicity.low = toxicity.low,
  toxicity.moderate= toxicity.moderate,
  toxicity.high = toxicity.high,
  efficacy.low = efficacy.low,
  efficacy.moderate = efficacy.moderate,
  efficacy.high = efficacy.high,
  target.toxicity=target.toxicity,
  target.efficacy=target.efficacy,
  cohortsize=cohortsize, ncohort=ncohort)
print(decision.obd)
```

get.decision.obd2.kb *Automatic Boundary and Decision Tables for Phase I/II Trials.*

Description

This function automatically generates a boundary table and a decision table for single-agent Phase I/II trials using Keyboard design.

Usage

```

get.decision.obd2.kb(
  target.toxicity,
  target.efficacy,
  cohortsize,
  ncohort,
  cutoff.eli.toxicity = 0.95,
  cutoff.eli.efficacy = 0.3,
  decision = c("E", "E", "S", "S", "S", "S", "D", "D", "D")
)

```

Arguments

target.toxicity	The target toxicity rate.
target.efficacy	The target efficacy rate.
cohortsize	The number of patients in the cohort.
ncohort	The total number of cohorts.
cutoff.eli.toxicity	The cutoff value to eliminate a dose with unacceptable high toxicity for safety. The default value is 0.95.
cutoff.eli.efficacy	The cutoff value to eliminate a dose with unacceptable low efficacy. The default value is 0.3.
decision	The pre-specified decision matrix for the 9 combination intervals: (low toxicity, low efficacy),(low toxicity, moderate efficacy),(low toxicity, high efficacy), (moderate toxicity, low efficacy),(moderate toxicity, moderate efficacy),(moderate toxicity, high efficacy),(high toxicity, low efficacy),(high toxicity, moderate efficacy),(high toxicity, high efficacy)

Details

The Keyboard design utilizes the posterior distributions of the toxicity and efficacy to guide dosage transition. To determine whether to escalate or de-escalate given the observed data at the current dose, we must generate a boundary table and a decision matrix. The boundary table requires a lower boundary and an upper boundary for both toxicity and efficacy and a decision matrix, which provides dose escalation or de-escalation suggestions.

A Bayesian toxicity probability interval design from Yuan's group is used to determine the lower and upper boundaries for toxicity based on the target toxicity rate.

The boundaries for efficacy are calculated based on the efficacy failure rate, which is $1 - \text{target.efficacy}$. The efficacy failure rate is defined as $1 - \text{target.efficacy}$ and treated similarly to toxicity. The lower and upper boundaries for efficacy failure are determined using a Bayesian toxicity probability interval design. To compute the lower and upper boundaries for efficacy, we use the following:

$$\text{efficacy.lower.boundary} = 1 - \text{efficacy.failure.upper.boundary}$$

$$efficacy.upper.boundary = 1 - efficacy.failure.lower.boundary$$

There are 3 subintervals for toxicity:

1. The low interval for toxicity is (0, toxicity.lower.boundary).
2. The moderate interval for toxicity is (toxicity.lower.boundary, toxicity.upper.boundary).
3. The high interval for toxicity is (toxicity.upper.boundary, 1).

There are 3 subintervals for efficacy:

1. The low interval for efficacy is (0, efficacy.lower.boundary).
2. The moderate interval for efficacy is (efficacy.lower.boundary, efficacy.upper.boundary).
3. The high interval for efficacy is (efficacy.upper.boundary, 1).

Assuming that the toxicity probability is p_i and the efficacy probability is q_i at dose level i , the probability unit intervals for toxicity and efficacy can be partitioned into subintervals (a, b) and (c, d). (a, b) is a subinterval for toxicity probability and (c, d) is a subinterval for efficacy probability. (a, b) x (c, d) is a combination interval. 3 toxicity intervals and 3 efficacy intervals result in 9 combination intervals. Besides the target toxicity rate and the target efficacy rate, a matrix is required that contains 9 decisions for these 9 combination intervals ordered as ((low toxicity, low efficacy), (low toxicity, moderate efficacy), (low toxicity, high efficacy), (moderate toxicity, low efficacy), (moderate toxicity, moderate efficacy), (moderate toxicity, high efficacy), (high toxicity, low efficacy), (high toxicity, moderate efficacy), (high toxicity, high efficacy)). Possible decisions are "E", "D", "S". Decision "D" denotes de-escalation, so the next cohort of patients will be treated at the next lower dose level. Decision "E" denotes escalation, so the next cohort of patients will be treated at the next higher dose level. Decision "S" denotes stay, so the next cohort of patients will be treated at the current dose level.

Shown here is an example of boundary tables designed using this method with a target toxicity rate of 0.2 and a target efficacy rate of 0.4:

		Efficacy.low (0,0.27)	Efficacy.moderate (0.27,0.52)	Efficacy.high (0.52,1)
Toxicity.low	(0,0.16)	E	E	S
Toxicity.moderate	(0.16,0.24)	S	S	S
Toxicity. high	(0.24,1)	D	D	D

For example, the interval combination (0, 0.16) x (0.27,0.52) corresponds to a decision "S". This means that the next cohort of patients will be treated at the current dose level if the observed toxicity rate of current dose falls in (0, 0.16) and the observed efficacy rate falls in (0.27,0.52) .

Suppose that there are d doses in the trial and that the current dose is i . Define the number of patients as n_i , the number of patients who experienced toxicity as x_i and the number of responses as y_i . The trial data can be shown as follows:

$$D = (n_i, x_i, y_i), i = 1, \dots, d$$

Bayesian rule is used to calculate the joint unit probability mass (JUPM) for the toxicity and efficacy combination intervals. For a given combination interval, JUPM is calculated as follows:

$$JUPM_{(a,b)}^{(c,d)} = Pr p_j \in (a, b), q_j \in (c, d) | D / (b - a) * (d - c)$$

$Pr p_j \in (a, b), q_j \in (c, d) | D$ are the posterior probabilities of p_i and q_i falling in the subinterval (a, b) and (c, d) . Assume the prior for both p_i and q_i follows independent beta distributions $beta(\alpha_p, \beta_p)$ and $beta(\alpha_q, \beta_q)$ respectively. The posterior distributions for p_i and q_i are $beta(\alpha_p + x_i, \beta_p + n_i - x_i)$ and $beta(\alpha_q + y_i, \beta_q + n_i - y_i)$. Using these posterior distributions, calculate the JUPM for all 16 combination intervals and find the winning combination interval (a^*, b^*) and (c^*, d^*) , which achieves the largest JUPM. The decision that corresponds to this winning combination interval is used to treat the next cohort of patients.

Two dose exclusion rules are applied in the trial design: safety rule and futility rule. Safety rule: if at least 3 patients have been treated at a given dose and given the observed data indicate that the probability of the toxicity rate of the current dose exceeding the target toxicity rate is more than 95%, we eliminate the current and any higher dose from the trial to prevent exposing future patients to with unacceptable high toxicity. The probability threshold can be specified with `cutoff.eli.toxicity`. When a dose is eliminated, the design recommends the next lower dose for treating the next cohort of patients. If the lowest dose is overly toxic, then the trial terminates early and no dose is selected as the OBD. This corresponds to a dose assignment of "DUT", which means de-escalation due to unacceptable high toxicity and excluding the current dose and any dose higher than this dose from the trial.

Futility rule: if at least 3 patients have been treated at a given dose and the observed data indicate that the probability of the current dose's efficacy rate exceeding the target efficacy rate is less than 30%, then we eliminate this dose from the trial to avoid exposing future patients to these futile doses. The probability threshold can be specified with `cutoff.eli.efficacy`. This corresponds to two possible dose assignments: "EUE" and "DUE", both excluding this dose from the trial. "EUE" denotes escalation due to unacceptable low efficacy and "DUE" denotes de-escalation due to unacceptable low efficacy.

An attractive feature of the Keyboard design is that its dose escalation and de-escalation rule can be tabulated before implementing the trial. Thus, when conducting the trial, no real-time calculation or model fitting is needed, and one needs to count only the number of patients, the number of DLTs, and the number of responses observed at the current dose and the decision to escalate or de-escalate is based on the pre-tabulated decision rules.

Value

`get.decision.obd2.kb()` returns a prespecified boundary table and a dose assignment decision table:

1. Boundary table (`$boundary.table`)
2. Decision matrix (`$decision.matrix`)

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Liu S. and Yuan, Y. Bayesian Optimal Interval Designs for Phase I Clinical Trials, *Journal of the Royal Statistical Society: Series C*. 2015; 64, 507-523.

Yuan Y., Hess K.R., Hilsenbeck S.G. and Gilbert M.R. Bayesian Optimal Interval Design: A Simple and Well-performing Design for Phase I Oncology Trials. *Clinical Cancer Research*. 2016; 22, 4291-4301.

Li DH, Whitmore JB, Guo W, Ji Y. Toxicity and efficacy probability interval design for phase I adoptive cell therapy dose-finding clinical trials. *Clinical Cancer Research*. 2017; 23:13-20. <https://clincancerres.aacrjournals.org/content/23/1/13.long>

See Also

Other single-agent phase I/II functions: [get.decision.obd.kb\(\)](#), [get.oc.obd.kb\(\)](#), [get.oc.obd2.kb\(\)](#), [select.obd.kb\(\)](#)

Examples

```
decision.obd2.kb <- get.decision.obd2.kb(target.toxicity=0.2,  
    target.efficacy=0.4, cohortsize=3, ncohort=10)  
print(decision.obd2.kb)
```

get.oc.comb.kb

Operating Characteristics for Drug-combination Trials

Description

This function generates the operating characteristics of the Keyboard design for drug-combination trials.

Usage

```
get.oc.comb.kb(  
  target,  
  p.true,  
  ncohort,  
  cohortsize,  
  n.earllystop = 100,  
  marginL = 0.05,  
  marginR = 0.05,  
  startdose = c(1, 1),  
  cutoff.eli = 0.95,  
  extrasafe = FALSE,  
  offset = 0.05,  
  ntrial = 1000  
)
```

Arguments

target	The target dose-limiting toxicity (DLT) rate.
p.true	A J*K matrix (J<=K) containing the true toxicity probabilities of combinations with J dose levels of agent A and K dose levels of agent B.
ncohort	A scalar specifying the total number of cohorts in the trial.
cohortsiz	The number of patients in the cohort.
n.earlystop	The early stopping parameter. If the number of patients treated at the current dose reaches n.earlystop, then we stop the trial and select the MTD based on the observed data. The default value is 100.
marginL	The difference between the target and the lower bound of the "target key" (proper dosing interval) to be defined. The default is 0.05.
marginR	The difference between the target and the upper bound of the "target key" (proper dosing interval) to be defined. The default is 0.05.
startdose	The starting dose combination level for the drug-combination trial. The default is c(1, 1).
cutoff.eli	The cutoff value to eliminate an overly toxic dose and all higher doses for safety. The default value is 0.95.
extrasafe	Set extrasafe=TRUE to impose a stricter stopping rule. The default is FALSE.
offset	A small positive number (between 0 and 0.5) to control how strict the stopping rule is when extrasafe=TRUE. A larger value leads to a stricter stopping rule. The default value is 0.05.
ntrial	The total number of trials to be simulated. The default value is 1000.

Value

The function returns the operating characteristics of the Keyboard combination design as a list:

1. the true toxicity probability at each dose level (`$p.true`),
2. the selection percentage at each dose level (`$selpercent`),
3. the percentage of correct selection (`$pcs`),
4. the number of patients treated at each dose level (`$nptsdose`),
5. the number of toxicities observed at each dose level (`$ntoxdose`),

6. the total number of toxicities observed in the trial (`$totaltox`),
7. the total number of patients in the trial (`$totaln`),
8. the total percentage of patients treated at the MTD (`$npercent`).

Uses

This function uses [get.boundary.comb.kb](#)

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Yan F, Mandrekar SJ, Yuan Y. Keyboard: A Novel Bayesian Toxicity Probability Interval Design for Phase I Clinical Trials. *Clinical Cancer Research*. 2017; 23:3994-4003. <http://clincancerres.aacrjournals.org/content/23/15/3994.full.pdf>

Pan H, Lin R, Yuan Y. Keyboard design for phase I drug-combination trials. *Contemporary Clinical Trials*. 2020. <https://doi.org/10.1016/j.cct.2020.105972>

See Also

Other drug-combination functions: [get.boundary.comb.kb\(\)](#), [next.comb.kb\(\)](#), [select.mtd.comb.kb\(\)](#)

Examples

```
### Drug-combination trial ###  
  
p.true <- matrix(c(0.01, 0.03, 0.10, 0.20, 0.30,  
                 0.03, 0.05, 0.15, 0.30, 0.60,  
                 0.08, 0.10, 0.30, 0.60, 0.75), byrow=TRUE, ncol=5)  
  
oc.comb <- get.oc.comb.kb(target=0.3, p.true=p.true, ncohort=20, cohortsize=3,  
                        n.earlystop=12, startdose=c(1, 1), ntrial=100)
```

get.oc.kb

Operating Characteristics for Single-agent Trials

Description

This function generates the operating characteristics of the Keyboard design for single-agent trials.

Usage

```

get.oc.kb(
  target,
  p.true,
  ncohort,
  cohortsize,
  n.earlystop = 100,
  startdose = 1,
  marginL = 0.05,
  marginR = 0.05,
  cutoff.eli = 0.95,
  extrasafe = FALSE,
  offset = 0.05,
  ntrial = 1000
)

```

Arguments

target	The target dose-limiting toxicity (DLT) rate.
p.true	A vector containing the true toxicity probabilities of the investigational dose levels.
ncohort	A scalar specifying the total number of cohorts in the trial.
cohortsize	The number of patients in the cohort. The default is 1.
n.earlystop	The early stopping parameter. If the number of patients treated at the current dose reaches n.earlystop, then stop the trial and select the MTD based on the observed data. The default value is 100.
startdose	The starting dose level for the trial. The default is 1.
marginL	The difference between the target and the lower limit of the "target key" (proper dosing interval) to be defined. The default is 0.05.
marginR	The difference between the target and the upper limit of the "target key" (proper dosing interval) to be defined. The default is 0.05.
cutoff.eli	The cutoff value to eliminate an overly toxic dose and all higher doses for safety. The default value is 0.95.
extrasafe	Set extrasafe=TRUE to impose a stricter stopping rule. The default is FALSE.
offset	A small positive number (between 0 and 0.5) to control how strict the stopping rule is when extrasafe=TRUE. A larger value leads to a stricter stopping rule. The default value of 0.05 generally works well.
ntrial	The total number of trials to be simulated. The default value is 1000.

Details

The operating characteristics of the Keyboard design are generated by simulating trials under the pre-specified true toxicity probabilities of the investigational doses.

The Keyboard design defines a series of equal-width dosing intervals (or "keys") to present the potential locations of the true toxicity of a dose and guide the dose escalation or de-escalation between cohorts of patients. The Keyboard design starts by specifying a proper dosing interval, referred to as the "target key", based on which it forms a series of equally wide keys that span the remainder of the range from 0 to 1.

The Keyboard design relies on the posterior distribution of the toxicity probability to guide dosage. To make the decision of dose escalation or de-escalation, given the observed data at the current dose, we identify the key that has the highest posterior probability, referred to as the "strongest key". This key represents where the true dose-limiting toxicity (DLT) rate of the current dose is most likely located. If the strongest key is to the left of the target key, then we escalate the dose because the data suggest that the current dose is most likely to underdose; if the strongest key is to the right of the target key, then we de-escalate the dose because the observed data suggest that the current dose is likely to overdose; and if the strongest key is the target key, then we retain the current dose because the observed data support the notion that the current dose is most likely to be in the proper dosing interval. Graphically, the strongest key is the one with the largest area under the posterior distribution curve of the DLT rate of the current dose.

An attractive feature of the Keyboard design is that its dose escalation and de-escalation rule can be tabulated before the onset of the trial. Thus, when conducting the trial, no calculation or model fitting is needed, and we need to count only the number of DLTs observed at the current dose; the decision to escalate or de-escalate the dose is based on the pre-tabulated decision rules.

Given all observed data, the Keyboard design uses an isotonic regression to obtain an efficient statistical estimate of the maximum tolerated dose (MTD) by utilizing the fact that toxicity presumably increases with the dose.

For patient safety, the following dose-elimination rule is evaluated after each cohort: if at least 3 patients have been treated at the given dose and the observed data indicate that there is more than a 95% chance that the current dose is above the maximum tolerated dose (MTD), then we eliminate this dose and beyond from the trial to prevent exposing future patients to these overly toxic doses. The probability threshold for elimination can be specified with `cutoff.e1i`. When a dose is eliminated, the design recommends the next lower dose for treating the next patient.

The Keyboard design has two built-in stopping rules:

- (1) stop the trial if the lowest dose is eliminated due to toxicity, and no dose should be selected as the MTD; and
- (2) stop the trial and select the MTD if the number of patients treated at the current dose reaches `n.earllystop`.

The first stopping rule is a safety rule to protect patients from being exposed to overly toxic doses. The rationale for the second stopping rule is that when enough (i.e., `n.earllystop`) patients are assigned to a dose, then the dose-finding algorithm has approximately converged. Thus, we can stop the trial early and select the MTD to minimize the sample size and the trial duration.

For some applications, investigators may prefer a stricter stopping rule than rule (1) for extra safety when the lowest dose is overly toxic. This can be achieved by setting `extrasafe=TRUE`, which imposes the following, stricter, safety stopping rule:

Stop the trial if

- (i) the number of patients treated at the lowest dose ≥ 3 , and
- (ii)

$$Pr((\text{toxicityrateofthelowestdose} > \text{target}) | \text{data}) > \text{cutoff.f.eli} - \text{offset}$$

As a tradeoff, the strong stopping rule will decrease the MTD selection percentage when the lowest dose is the true MTD.

Value

The function returns the operating characteristics of the Keyboard design (single-agent) as a list, which includes:

1. the selection percentage at each dose level (`$selpercent`),
2. the average number of patients treated at each dose level (`$npatients`),
3. the average number of toxicities observed at each dose level (`$ntox`),
4. the average number of toxicities (`$totaltox`),
5. the average number of patients (`$totaln`),
6. the percentage of early stopping due to toxicity without selecting the MTD (`$percentstop`),
7. the risk of overdosing 60% or more of patients (`$overdose60`),
8. the risk of overdosing 80% or more of patients (`$overdose80`),
9. a data.frame object containing simulation parameters, such as `target`, `p.true`, etc. (`$simu.setup`).

Uses

This function uses [get.boundary.kb](#) and [select.mtd.kb](#).

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Yan F, Mandrekar SJ, Yuan Y. Keyboard: A Novel Bayesian Toxicity Probability Interval Design for Phase I Clinical Trials. *Clinical Cancer Research*. 2017; 23:3994-4003. <http://clincancerres.aacrjournals.org/content/23/15/3994.full.pdf>

See Also

Other single-agent functions: [get.boundary.kb\(\)](#), [select.mtd.kb\(\)](#)

Examples

```
### Single-agent trial ###  
  
oc <- get.oc.kb(target=0.3, p.true=c(0.05, 0.15, 0.3, 0.45, 0.6),  
              ncohort=20, cohortsize=3, ntrial=1000)  
  
oc
```

get.oc.obd.kb	<i>Generate Operating Characteristics to Find the Optimal Biological Dose</i>
---------------	---

Description

This function generates operating characteristics to find the optimal biological dose (OBD).

Usage

```
get.oc.obd.kb(  
  toxicity.low,  
  toxicity.moderate,  
  toxicity.high,  
  efficacy.low,  
  efficacy.moderate,  
  efficacy.high,  
  target.toxicity,  
  target.efficacy,  
  ncohort = 10,  
  cohortsize = 3,  
  n.early = 100,  
  startdose = 1,  
  p.true,  
  q.true,  
  ntrial = 1000,  
  seed = 6,  
  p1 = 0.15,  
  p2 = 0.4,  
  q1 = 0.3,  
  q2 = 0.6,  
  cutoff.eli.toxicity = 0.95,  
  cutoff.eli.efficacy = 0.3,  
  w1.toxicity = 0.33,  
  w2.toxicity = 1.09,  
  indicator = target.toxicity  
)
```


Arguments

toxicity.low	The upper boundary for the low toxicity interval.
toxicity.moderate	The upper boundary for the moderate toxicity interval.
toxicity.high	The upper boundary for the high toxicity interval.
efficacy.low	The upper boundary for the low efficacy interval.
efficacy.moderate	The upper boundary for the moderate efficacy interval.
efficacy.high	The upper boundary for the high efficacy interval.
target.toxicity	The target DLT rate.
target.efficacy	The target efficacy rate.
ncohort	The total number of cohorts.
cohortsiz	The number of patients in the cohort.
n.early	The early stopping parameter. If the number of patients treated at the current dose reaches n.early, then we stop the trial and select the optimal biological dose (OBD) based on the observed data. The default value is 100.
startdose	The starting dose level.
p.true	A vector containing the true toxicity probabilities of the investigational dose levels.
q.true	A vector containing the true efficacy probabilities of the investigational dose levels.
ntrial	The total number of trials to be simulated.
seed	The random seed for simulation.
p1	The cutoff lower limit for safety utility function 1, described in the Details section.
p2	The cutoff upper limit for safety utility function 1, described in the Details section.
q1	The cutoff lower limit for efficacy utility function 1, described in the Details section.
q2	The cutoff upper limit for efficacy utility function 1, described in the Details section.
cutoff.eli.toxicity	The cutoff value to eliminate a dose with unacceptable high toxicity for safety. The default value is 0.95.
cutoff.eli.efficacy	The cutoff value for the futility rule, the acceptable lowest efficacy. The default value is 0.30.
w1.toxicity	The weight for toxicity utility function 2 and 3, described in the Details section.
w2.toxicity	The weight for toxicity utility function 3, described in the Details section.
indicator	The indicator cutoff value for utility function 3, described in the Details section.

Details

Large trials are simulated to characterize the operating characteristics of the Keyboard design under the prespecified true toxicity probabilities and true efficacy probabilities of the investigational doses. The dose assignment follows the rules described in the function `get.decision.obd.kb()`.

The following stopping rules are built in the Keyboard design:

1. Stop the trial if the lowest dose is eliminated from the trial due to high unacceptable toxicity.
2. Stop the trial if the number of patients treated at the current dose exceeds `n.earllystop`.

Value

`get.oc.obd.kb()` returns the operating characteristics of the Keyboard design as a list, including:

1. the selection percentage at each dose level using utility function 1 (`$selpercent1`),
2. the selection percentage at each dose level using utility function 2 (`$selpercent2`),
3. the selection percentage at each dose level using utility function 3 (`$selpercent3`),
4. the number of patients treated at each dose level (`$npatients`),
5. the number of dose-limiting toxicities (DLTs) observed at each dose level (`$ntox`),
6. the number of responses observed at each dose level (`$neff`),
7. the average number of DLTs (`$totaltox`),
8. the average number of responses (`$totaleff`),
9. the average number of patients (`$totaln`),
10. the percentage of early stopping without selecting the OBD using utility function 1 (`$percentstop1`),
11. the percentage of early stopping without selecting the OBD using utility function 2 (`$percentstop2`),
12. the percentage of early stopping without selecting the OBD using utility function 3 (`$percentstop3`),
13. `data.frame` (`$simu.setup`) containing simulation parameters, such as `target`, `p.true`, etc.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Li DH, Whitmore JB, Guo W, Ji Y. Toxicity and efficacy probability interval design for phase I adoptive cell therapy dose-finding clinical trials. *Clinical Cancer Research*. 2017; 23:13-20. <https://clincancerres.aacrjournals.org/content/23/1/13.long>

Liu S, Johnson VE. A robust Bayesian dose-finding design for phase I/II clinical trials. *Biostatistics*. 2016; 17(2):249-63. <https://academic.oup.com/biostatistics/article/17/2/249/1744018>

Zhou Y, Lee JJ, Yuan Y. A utility-based Bayesian optimal interval (U-BOIN) phase I/II design to identify the optimal biological dose for targeted and immune therapies. *Statistics in Medicine*. 2019; 38:S5299-5316. <https://onlinelibrary.wiley.com/doi/epdf/10.1002/sim.8361>

See Also

Other single-agent phase I/II functions: [get.decision.obd.kb\(\)](#), [get.decision.obd2.kb\(\)](#), [get.oc.obd2.kb\(\)](#), [select.obd.kb\(\)](#)

Examples

```
toxicity.low <- 0.15
toxicity.moderate <- 0.25
toxicity.high <- 0.35
efficacy.low <- 0.25
efficacy.moderate <- 0.45
efficacy.high <- 0.65
target.toxicity <- 0.30
target.efficacy <- 0.40
p.true <- c(0.08, 0.30, 0.60, 0.80)
q.true <- c(0.25, 0.40, 0.25, 0.50)
oc.obd.kb <- get.oc.obd.kb(toxicity.low = toxicity.low,
  toxicity.moderate = toxicity.moderate,
  toxicity.high = toxicity.high,
  efficacy.low = efficacy.low,
  efficacy.moderate = efficacy.moderate,
  efficacy.high = efficacy.high,
  target.toxicity = target.toxicity,
  target.efficacy = target.efficacy,
  p.true = p.true, q.true = q.true)

oc.obd.kb
summary_kb(oc.obd.kb)
plot_kb(oc.obd.kb)
plot_kb(oc.obd.kb$selpersent1)
plot_kb(oc.obd.kb$selpersent2)
plot_kb(oc.obd.kb$selpersent3)
plot_kb(oc.obd.kb$npatients)
plot_kb(oc.obd.kb$ntox)
plot_kb(oc.obd.kb$neff)
```

`get.oc.obd2.kb`*Generate Operating Characteristics for OBD Finding Automaticlly*

Description

This function reports operating characteristics for optimal biological dose (OBD) finding automatically.

Usage

```
get.oc.obd2.kb(  
  target.toxicity,  
  target.efficacy,  
  ncohort = 10,  
  cohortsize = 3,  
  n.early = 100,  
  startdose = 1,  
  p.true,  
  q.true,  
  ntrial = 1000,  
  seed = 6,  
  p1 = 0.15,  
  p2 = 0.4,  
  q1 = 0.3,  
  q2 = 0.6,  
  cutoff.eli.toxicity = 0.95,  
  cutoff.eli.efficacy = 0.3,  
  w1.toxicity = 0.33,  
  w2.toxicity = 1.09,  
  indicator = target.toxicity  
)
```

Arguments

<code>target.toxicity</code>	The target DLT rate.
<code>target.efficacy</code>	The target efficacy rate.
<code>ncohort</code>	The total number of cohorts.
<code>cohortsize</code>	The number of patients in the cohort.
<code>n.early</code>	The early stopping parameter. If the number of patients treated at the current dose reaches <code>n.early</code> , then we stop the trial and select the optimal biological dose (OBD) based on the observed data. The default value is 100.
<code>startdose</code>	The starting dose level.
<code>p.true</code>	A vector containing the true toxicity probabilities of the investigational dose levels.

q.true	A vector containing the true efficacy probabilities of the investigational dose levels.
ntrial	The total number of trials to be simulated.
seed	The random seed for simulation.
p1	The cutoff lower limit for safety utility function 1, described in the Details section.
p2	The cutoff upper limit for safety utility function 1, described in the Details section.
q1	The cutoff lower limit for efficacy utility function 1, described in the Details section.
q2	The cutoff upper limit for efficacy utility function 1, described in the Details section.
cutoff.eli.toxicity	The cutoff to eliminate a dose with unacceptable high toxicity for safety. The recommended value is 0.95.
cutoff.eli.efficacy	The cutoff for the futility rule, the acceptable lowest efficacy. The recommended value is 0.30.
w1.toxicity	The weight for toxicity utility function 2 and 3, described in the Details section.
w2.toxicity	The weight for toxicity utility function 3, described in the Details section.
indicator	The indicator cutoff for utility function 3, described in the Details section.

Details

A large number of trials are simulated to characterize the operating characteristics of the Keyboard design under the prespecified true toxicity probabilities and true efficacy probabilities of the investigational doses. The dose assignment follows the rules described in the function `get.decision.obd.kb()`.

The following stopping rules are built in the Keyboard design:

1. Stop the trial if the lowest dose is eliminated from the trial due to high unacceptable toxicity.
2. Stop the trial if the number of patients treated at current dose is larger than or equal to `n.earlystop`.

Value

`get.oc.obd2.kb()` returns the operating characteristics of the Keyboard design as a list, including:

1. the selection percentage at each dose level using utility function 1 (`$selpercent1`),
2. the selection percentage at each dose level using utility function 2 (`$selpercent2`),
3. the selection percentage at each dose level using utility function 3 (`$selpercent3`),
4. the number of patients treated at each dose level (`$npatients`),

5. the number of dose-limiting toxicities (DLTs) observed at each dose level (`$ntox`),
6. the number of responses observed at each dose level (`$neff`),
7. the average number of DLTs (`$totaltox`),
8. the average number of responses (`$totaleff`),
9. the average number of patients (`$totaln`),
10. the percentage of early stopping without selecting the OBD using utility function 1 (`$percentstop1`),
11. the percentage of early stopping without selecting the OBD using utility function 2 (`$percentstop2`),
12. the percentage of early stopping without selecting the OBD using utility function 3 (`$percentstop3`),
13. `data.frame($simu.setup)` containing simulation parameters, such as `target`, `p.true`, etc.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Li DH, Whitmore JB, Guo W, Ji Y. Toxicity and efficacy probability interval design for phase I adoptive cell therapy dose-finding clinical trials. *Clinical Cancer Research*. 2017; 23:13-20. <https://clincancerres.aacrjournals.org/content/23/1/13.long>

Liu S, Johnson VE. A robust Bayesian dose-finding design for phase I/II clinical trials. *Biostatistics*. 2016; 17(2):249-63. <https://academic.oup.com/biostatistics/article/17/2/249/1744018>

Zhou Y, Lee JJ, Yuan Y. A utility-based Bayesian optimal interval (U-BOIN) phase I/II design to identify the optimal biological dose for targeted and immune therapies. *Statistics in Medicine*. 2019; 38:S5299-5316. <https://onlinelibrary.wiley.com/doi/epdf/10.1002/sim.8361>

See Also

Other single-agent phase I/II functions: [get.decision.obd.kb\(\)](#), [get.decision.obd2.kb\(\)](#), [get.oc.obd.kb\(\)](#), [select.obd.kb\(\)](#)

Examples

```
target.toxicity <- 0.30
target.efficacy <- 0.40
p.true <- c(0.08, 0.30, 0.60, 0.80)
q.true <- c(0.25, 0.40, 0.25, 0.50)
oc.obd2.kb <- get.oc.obd2.kb(target.toxicity=target.toxicity,
                             target.efficacy= target.efficacy, ncohort=20,
                             cohortsize= 3, p.true= p.true, q.true= q.true)
```

```

oc.obd2.kb
summary_kb(oc.obd2.kb)
plot_kb(oc.obd2.kb)
plot_kb(oc.obd2.kb$selpercent1)
plot_kb(oc.obd2.kb$selpercent2)
plot_kb(oc.obd2.kb$selpercent3)
plot_kb(oc.obd2.kb$npatients)
plot_kb(oc.obd2.kb$ntox)
plot_kb(oc.obd2.kb$neff)

```

next.comb.kb

Find the Next Dose Combination

Description

This function determines the dose combination for the next cohort of patients in drug-combination trials.

Usage

```

next.comb.kb(
  target,
  npts,
  ntox,
  dose.curr,
  n.earllystop = 100,
  marginL = 0.05,
  marginR = 0.05,
  cutoff.eli = 0.95,
  extrasafe = FALSE,
  offset = 0.05
)

```

Arguments

target	The target dose-limiting toxicity (DLT) rate.
npts	A J*K matrix (J<=K) containing the number of patients treated at each dose combination.
ntox	A J*K matrix (J<=K) containing the number of patients who experienced a dose-limiting toxicity at each dose combination.
dose.curr	The current dose combination, i.e., the dose combination that was used to treat the most recently enrolled cohort of patients.
n.earllystop	The early stopping parameter. If the number of patients treated at the current dose reaches n.earllystop, then we stop the trial and select the MTD based on the observed data. The default value is 100.

marginL	The difference between the target and the lower limit of the "target key" (proper dosing interval) to be defined. The default is 0.05.
marginR	The difference between the target and the upper limit of the "target key" (proper dosing interval) to be defined. The default is 0.05.
cutoff.eli	The cutoff value to eliminate an overly toxic dose and all higher doses for safety. The default value is 0.95.
extrasafe	Set extrasafe=TRUE to impose a stricter stopping rule. The default is FALSE.
offset	A small positive number (between 0 and 0.5) to control how strict the stopping rule is when extrasafe=TRUE. A larger value leads to a stricter stopping rule. The default value is 0.05.

Details

Given the observed data thus far, this function determines the dose combination for treating the next cohort of new patients. The observed data are as follows: the number of patients treated at each dose combination (npts), the number of patients who experienced dose-limiting toxicities (DLTs) at each dose combination (ntox) and the level of the current dose (dose.curr). The number of patients for doses that have not been used in the trial is zero.

Value

This function returns the recommended dose for treating the next cohort of patients (`$next_dc`).

Uses

This function uses [get.boundary.comb.kb](#).

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Yan F, Mandrekar SJ, Yuan Y. Keyboard: A Novel Bayesian Toxicity Probability Interval Design for Phase I Clinical Trials. *Clinical Cancer Research*. 2017; 23:3994-4003. <http://clincancerres.aacrjournals.org/content/23/15/3text.pdf>

Pan H, Lin R, Yuan Y. Keyboard design for phase I drug-combination trials. *Contemporary Clinical Trials*. 2020. <https://doi.org/10.1016/j.cct.2020.105972>

See Also

Other drug-combination functions: [get.boundary.comb.kb\(\)](#), [get.oc.comb.kb\(\)](#), [select.mtd.comb.kb\(\)](#)

Examples

```
### Drug-combination trial ###

n <- matrix(c(3, 0, 0, 0, 0,
              7, 6, 0, 0, 0,
              0, 0, 0, 0, 0), ncol=5, byrow=TRUE)
y <- matrix(c(0, 0, 0, 0, 0,
              1, 1, 0, 0, 0,
              0, 0, 0, 0, 0), ncol=5, byrow=TRUE)

nxt.comb <- next.comb.kb(target=0.3, npts=n, ntox=y, dose.curr=c(2, 2))
summary_kb(nxt.comb)
```

plot_kb

*Plot Objects Generated by the Keyboard Design Package***Description**

This function plots the objects returned by the other functions in the Keyboard package.

Usage

```
plot_kb(x, ..., name = deparse(substitute(x)))
```

Arguments

x	The object (returned by another function) to be plotted.
...	Ignored arguments.
name	The name of the object to be plotted. (Users should not input this parameter).

Details

(1) For `get.oc.kb()` and `get.oc.comb.kb()`, the operating characteristics of the Keyboard design, which include:

- (i) the overall dose selection percentage of the simulations generated,
- (ii) the number of patients treated at each dose, and
- (iii) the toxicities presented at each dose.

(2) For `get.oc.obd.kb()` and `get.oc.obd2.kb()`, the operating characteristics of the Keyboard design for phase I/II, which includes:

- (i) the overall dose selection percentage of the simulations generated,
- (ii) the number of patients treated at each dose,
- (iii) the toxicities presented at each dose, and

(iv) the efficacies presented at each dose.

(3) For `select.mtd.kb()` and `select.mtd.comb.kb()`: an infographic showing the estimates of the toxicity probability for each dose, with corresponding 95% credible intervals.

Value

This function returns a figure.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

See Also

[summary_kb](#)

Examples

```
### Single-agent trial ###

## Get the operating characteristics for a Keyboard single-agent trial
oc <- get.oc.kb(target=0.3, p.true=c(0.05, 0.15, 0.3, 0.45, 0.6),
               ncohort=10, cohortsize=3, ntrial=1000)
summary_kb(oc)

plot_kb(oc)
plot_kb(oc$selpercent)
plot_kb(oc$npatients)
plot_kb(oc$ntox)

## Select the MTD based on a trial's data
n <- c(3, 3, 15, 9, 0)
y <- c(0, 0, 4, 4, 0)
selmtd <- select.mtd.kb(target=0.3, npts=n, ntox=y)
summary_kb(selmtd)

plot_kb(selmtd)

### Drug-combination trial ###

## Get the operating characteristics for a Keyboard drug-combination trial
p.true <- matrix(c(0.01, 0.03, 0.10, 0.20, 0.30,
                  0.03, 0.05, 0.15, 0.30, 0.60,
                  0.08, 0.10, 0.30, 0.60, 0.75), byrow=TRUE, ncol=5)
oc.comb <- get.oc.comb.kb(target=0.3, p.true=p.true, ncohort=20,
                         cohortsize=3, n.earlystop=12, startdose=c(1, 1), ntrial=100)
summary_kb(oc.comb)

plot_kb(oc.comb) # use previous plot or next plot button to switch plots
plot_kb(oc.comb$selpercent)
```

```

plot_kb(oc.comb$nptsdose)
plot_kb(oc.comb$ntoxdose)

## Select the MTD based on a trial's data
n <- matrix(c(3, 5, 0, 0, 0,
              7, 6, 15, 0, 0,
              0, 0, 4, 0, 0), ncol=5, byrow=TRUE)
y <- matrix(c(0, 1, 0, 0, 0,
              1, 1, 4, 0, 0,
              0, 0, 2, 0, 0), ncol=5, byrow=TRUE)
sel.comb <- select.mtd.comb.kb(target=0.3, npts=n, ntox=y)
summary_kb(sel.comb)

plot_kb(sel.comb)

### oc.obd.kb
toxicity.low <- 0.15
toxicity.moderate <- 0.33
toxicity.high <- 0.40
efficacy.low <- 0.20
efficacy.moderate <- 0.40
efficacy.high <- 0.60
target.toxicity <- 0.30
target.efficacy <- 0.40
p.true <- c(0.08,0.30,0.60,0.80)
q.true <- c(0.25,0.40,0.25,0.50)
oc.obd.kb <- get.oc.obd.kb(toxicity.low = toxicity.low,
                           toxicity.moderate= toxicity.moderate,
                           toxicity.high = toxicity.high,
                           efficacy.low = efficacy.low,
                           efficacy.moderate = efficacy.moderate,
                           efficacy.high = efficacy.high,
                           target.toxicity=target.toxicity,
                           target.efficacy= target.efficacy,
                           p.true= p.true, q.true= q.true)
summary_kb(oc.obd.kb)
plot_kb(oc.obd.kb)
plot_kb(oc.obd.kb$selpercent1)
plot_kb(oc.obd.kb$selpercent2)
plot_kb(oc.obd.kb$selpercent3)
plot_kb(oc.obd.kb$npatients)
plot_kb(oc.obd.kb$ntox)
plot_kb(oc.obd.kb$neff)

```

Description

This function selects the maximum tolerated dose (MTD) after the drug-combination trial is completed.

Usage

```
select.mtd.comb.kb(
  target,
  npts,
  ntox,
  cutoff.eli = 0.95,
  extrasafe = FALSE,
  offset = 0.05
)
```

Arguments

target	The target dose-limiting toxicity (DLT) rate.
npts	A J*K matrix (J<=K) containing the number of patients treated at each dose combination.
ntox	A J*K matrix (J<=K) containing the number of patients who experienced a dose-limiting toxicity at each dose combination.
cutoff.eli	The cutoff value to eliminate an overly toxic dose and all higher doses for safety. The default value is 0.95.
extrasafe	Set extrasafe=TRUE to impose a stricter stopping rule. The default is FALSE.
offset	A small positive number (between 0 and 0.5) to control how strict the stopping rule is when extrasafe=TRUE. A larger value leads to a stricter stopping rule. The default value is 0.05.

Details

The Keyboard design starts by specifying a target toxicity interval (referred to as the "target key") such that any dose with a toxicity probability within that interval can be practically viewed as the MTD. Based on this interval's width, the Keyboard design forms a series of equally wide keys that span the rest of the range from 0 to 1.

Given all the observed data once the trial is completed, this function selects the MTD based on matrix isotonic estimates of the toxicity probability of each dose combination, selecting the dose whose estimate is closest to the target. When there are ties, one is randomly chosen. These (matrix) isotonic estimates are obtained with [biviso](#).

For patient safety, the following dose elimination rule is evaluated after each cohort: if at least 3 patients have been treated at the given dose and the observed data indicate that there is more than a 95% chance that the current dose is above the MTD, then we eliminate this dose and beyond from the trial to prevent exposing future patients to these overly toxic doses. The probability threshold for elimination can be specified with `cutoff.eli`. When a dose is eliminated, the design recommends the next lower dose for treating the next patient. If the lowest dose combination (1, 1) is overly toxic, then the trial terminates early and no dose is selected as the MTD.

For some applications, investigators may prefer a stricter stopping rule for extra safety when the lowest dose is overly toxic. This can be achieved by setting `extrasafe=TRUE`, which imposes the following stricter safety stopping rule:

Stop the trial if

- (i) the number of patients treated at the lowest dose ≥ 3 , and
- (ii)

$$Pr[(toxicityrateofthelowestdose > target)|data] > cutoff.eli - offset$$

As a tradeoff, the strong stopping rule will decrease the MTD selection percentage when the lowest dose is the true MTD.

Value

The function returns a list with:

1. the target toxicity probability (`$target`),
2. the selected MTD (`$MTD`),
3. a matrix with the isotonic estimates of the DLT probability at each dose (`$p_est`).

Uses

This function uses [biviso](#).

Note

The MTD selection and dose escalation/de-escalation rules are two independent components of the trial design. When appropriate, another dose selection procedure (e.g., one based on a fitted logistic model) can be used to select the MTD after completing the trial using the Keyboard design.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Yan F, Mandrekar SJ, Yuan Y. Keyboard: A Novel Bayesian Toxicity Probability Interval Design for Phase I Clinical Trials. *Clinical Cancer Research*. 2017; 23:3994-4003. <http://clincancerres.aacrjournals.org/content/23/15/3text.pdf>

Pan H, Lin R, Yuan Y. Keyboard design for phase I drug-combination trials. *Contemporary Clinical Trials*. 2020. <https://doi.org/10.1016/j.cct.2020.105972>

See Also

Other drug-combination functions: [get.boundary.comb.kb\(\)](#), [get.oc.comb.kb\(\)](#), [next.comb.kb\(\)](#)

Examples

```
### Drug-combination trial ###

## Select the MTD based on the data from a 3 x 5 combination trial
n <- matrix(c(3, 5, 0, 0, 0,
              7, 6, 15, 0, 0,
              0, 0, 4, 0, 0), ncol=5, byrow=TRUE)
y <- matrix(c(0, 1, 0, 0, 0,
              1, 1, 4, 0, 0,
              0, 0, 2, 0, 0), ncol=5, byrow=TRUE)

sel.comb <- select.mtd.comb.kb(target=0.3, npts=n, ntox=y)

summary_kb(sel.comb)
plot_kb(sel.comb)
```

select.mtd.kb

Maximum Tolerated Dose (MTD) Selection for Single-agent Trials

Description

This function selects the maximum tolerated dose (MTD) after the single-agent trial is completed.

Usage

```
select.mtd.kb(
  target,
  npts,
  ntox,
  cutoff.eli = 0.95,
  extrasafe = FALSE,
  offset = 0.05
)
```

Arguments

target	The target dose-limiting toxicity (DLT) rate.
npts	A vector containing the number of patients treated at each dose level.
ntox	A vector containing the number of patients at each dose level who experienced a DLT at each dose level.
cutoff.eli	The cutoff to eliminate an overly toxic dose and all higher doses for safety. The default value is 0.95.
extrasafe	Set extrasafe=TRUE to impose a stricter stopping rule. The default is FALSE.
offset	A small positive number (between 0 and 0.5) to control how strict the stopping rule is when extrasafe=TRUE. A larger value leads to a stricter stopping rule. The default value is 0.05.

Details

The Keyboard design starts by specifying a target toxicity interval (referred to as the "target key") such that any dose with a toxicity probability within that interval can be practically viewed as the MTD. Based on this interval's width, the Keyboard design forms a series of equally wide keys that span the rest of range from 0 to 1.

This function selects the MTD based on isotonic estimates of toxicity probabilities, selecting dose level j^* for which the isotonic estimate of the DLT rate is closest to the target. If there are ties, then we select from the ties the highest dose level when the estimate of the DLT rate is smaller than the target, or the lowest dose level when the estimate of the DLT rate is greater than the target. The isotonic estimates are obtained by applying the pooled-adjacent-violators algorithm (PAVA) [Barlow, 1972].

For some applications, investigators may prefer a stricter stopping rule to ensure the lowest dose is not overly toxic. This can be achieved by setting `extrasafe=TRUE`, which imposes the following stricter safety stopping rule:

Stop the trial if

- (i) the number of patients treated at the lowest dose ≥ 3 , and
- (ii)

$$Pr((\text{toxicityrateofthelowestdose} > \text{target})|data) > \text{cutoff.f.eli} - \text{offset}$$

As a tradeoff, the strong stopping rule will decrease the MTD selection percentage when the lowest dose actually is the MTD.

Value

The function returns a list with:

1. the target toxicity probability (`$target`),
2. the selected MTD (`$MTD`),
3. the isotonic estimates of the DLT probability at each dose and corresponding 95% credible interval (`$p_est`),
4. the probability of overdosing defined as $Pr(\text{toxicity} > \text{target}|data)$ (`$p_overdose`).

Note

The MTD selection and dose escalation/de-escalation rules are two independent components of the trial design. When appropriate, another dose selection procedure (e.g., one based on a fitted logistic model) can be used to select the MTD after completing the trial using the Keyboard design.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

Yan F, Mandrekar SJ, Yuan Y. Keyboard: A Novel Bayesian Toxicity Probability Interval Design for Phase I Clinical Trials. *Clinical Cancer Research*. 2017; 23:3994-4003. <http://clincancerres.aacrjournals.org/content/23/15/3994.full.pdf>

See Also

Other single-agent functions: [get.boundary.kb\(\)](#), [get.oc.kb\(\)](#)

Examples

```
### Single-agent trial ###  
  
n <- c(3, 3, 15, 9, 0)  
y <- c(0, 0, 4, 4, 0)  
  
selmtd <- select.mtd.kb(target=0.3, npts=n, ntox=y)  
  
selmtd
```

select.obd.kb

Select the Optimal Biological Dose (OBD) for Single-agent Phase I/II Trials

Description

This function selects the optimal biological dose (OBD) at the end of a single-agent phase I/II trial.

Usage

```
select.obd.kb(  
  target.toxicity,  
  target.efficacy,  
  npts,  
  ntox,  
  neff,  
  p1 = 0.15,  
  p2 = 0.4,  
  q1 = 0.3,  
  q2 = 0.6,  
  cutoff.eli.toxicity = 0.95,  
  cutoff.eli.efficacy = 0.3,  
  w1.toxicity = 0.33,  
  w2.toxicity = 1.09,  
  indicator = target.toxicity  
)
```


Arguments

target.toxicity	The target dose-limiting toxicity (DLT) rate.
target.efficacy	The target efficacy rate.
npts	The vector containing the total number of patients treated at each dose level.
ntox	The vector containing the number of subjects who experienced toxicities at each dose level.
neff	The vector containing the number of subjects who experienced efficacies at each dose level.
p1	The cutoff lower limit for safety utility function 1, described in the Details section.
p2	The cutoff upper limit for safety utility function 1, described in the Details section.
q1	The cutoff lower limit for efficacy utility function 1, described in the Details section.
q2	The cutoff upper limit for efficacy utility function 1, described in the Details section.
cutoff.eli.toxicity	The cutoff value to eliminate a dose with unacceptable high toxicity for safety. The default value is 0.95.
cutoff.eli.efficacy	The cutoff value for futility rule, the acceptable lowest efficacy. The default value is 0.3.
w1.toxicity	The weight for toxicity utility function 2 and 3, described in the Details section.
w2.toxicity	The weight for toxicity utility function 3, described in the Details section.
indicator	The indicator cutoff value for utility function 3, described in the Details section.

Details

select.obd.kb() selects the OBD that is the most desirable based on benefit-risk tradeoff considering both toxicity and efficacy outcomes. A utility score is used to quantify the desirability of all admissible doses. Calculation of utility scores requires the posterior probabilities for toxicity p_i and efficacy q_i , which can be computed by using $beta(\alpha_p + x_i, \beta_p + n_i - x_i)$ and $beta(\alpha_q + y_i, \beta_q + n_i - y_i)$ assuming that the prior for both p_i and q_i follows independent beta distributions $beta(\alpha_p, \beta_p)$ and $beta(\alpha_q, \beta_q)$. Three criteria are used to calculate the desirability in this function.

The first criterion relies on a utility function for toxicity $f_1(p)$, where p denotes the toxicity rate, and on a utility function for efficacy $f_2(q)$, where q denotes the efficacy rate. $f_1(p)$ is 1 if $p \in [0, p1]$; $f_1(p)$ is 0 if $p \in [p2, 1]$; $f_1(p)$ is $1 - (p - p1)/(p2 - p1)$ if $p \in [p1, p2]$. $f_2(p)$ is 1 if $p \in (0, p1)$. Here, $p1$ is the cutoff lower limit and $p2$ is the cutoff upper limit for safety utility function 1 $f_1(p)$.

Similarly, $f_2(q)$ is 1 if $p \in [0, q1]$; $f_2(q)$ is 0 if $p \in [q2, 1]$; $f_2(q)$ is $1 - (p - q1)/(q2 - q1)$ if $p \in [q1, q2]$. $f_2(p)$ is 1 if $p \in (0, q1)$. Here, $q1$ is the cutoff lower limit and $q2$ is the cutoff upper limit for safety utility function $f_2(q)$.

The utility score that quantifies benefit-risk tradeoff at the current dose i is calculated as follows:

$$U(p_i, q_i) = f_1(p) * f_2(q)$$

The second criterion depends on a marginal toxicity probability $\pi_{T,i} = \text{beta}(\alpha_p + x_i, \beta_p + n_i - x_i)$ and a marginal efficacy probability $\pi_{E,i} = \text{beta}(\alpha_q + y_i, \beta_q + n_i - y_i)$. Then the utility score is calculated as follows:

$$U_i = \pi_{E,i} - w_1 * \pi_{T,i}$$

The third criterion also depends on a marginal toxicity probability $\pi_{T,i} = \text{beta}(\alpha_p + x_i, \beta_p + n_i - x_i)$ and a marginal efficacy probability $\pi_{E,i} = \text{beta}(\alpha_q + y_i, \beta_q + n_i - y_i)$, but it has an additional penalty when the posterior toxicity probability is high by using an indicator function. Then utility score using this function is calculated as follows:

$$U_j = \pi_{E,i} - w_1 * \pi_{T,i} - w_2 * \pi_{T,i} * I(\pi_{T,i} > \rho)$$

Here, the recommended ρ is the target toxicity rate.

Once the utility score is computed for all the doses, the optimal biological dose is calculated as follows:

$$d = \text{argmax}_i[U(p_i, q_i)|D]$$

Value

select.obd.kb() returns the selected dose:

1. Selected OBD level using utility function 1 (\$obd1), as described in the Details section.
2. Selected OBD level using utility function 2 (\$obd2), as described in the Details section.
3. Selected OBD level using utility function 3 (\$obd3), as described in the Details section.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

References

- Li DH, Whitmore JB, Guo W, Ji Y. Toxicity and efficacy probability interval design for phase I adoptive cell therapy dose-finding clinical trials. *Clinical Cancer Research*. 2017; 23:13-20. <https://clincancerres.aacrjournals.org/content/23/1/13.long>
- Liu S, Johnson VE. A robust Bayesian dose-finding design for phase I/II clinical trials. *Biostatistics*. 2016; 17(2):249-63. <https://academic.oup.com/biostatistics/article/17/2/249/1744018>
- Zhou Y, Lee JJ, Yuan Y. A utility-based Bayesian optimal interval (U-BOIN) phase I/II design to identify the optimal biological dose for targeted and immune therapies. *Statistics in Medicine*. 2019; 38:S5299-5316. <https://onlinelibrary.wiley.com/doi/epdf/10.1002/sim.8361>

See Also

Other single-agent phase I/II functions: [get.decision.obd.kb\(\)](#), [get.decision.obd2.kb\(\)](#), [get.oc.obd.kb\(\)](#), [get.oc.obd2.kb\(\)](#)

Examples

```
target.toxicity<-0.3
target.efficacy<-0.4
npts <- c(3,6,12,3,3)
ntox <- c(1,2,4,2,3)
neff <- c(0,0,5,1,1)
obd <- select.obd.kb (target.toxicity=target.toxicity,
                    target.efficacy= target.efficacy, npts = npts,
                    ntox = ntox, neff = neff)
print(obd)
```

summary_kb

Summarize Objects Generated by the Keyboard Design Package

Description

This function generates a descriptive summary for objects returned by other functions.

Usage

```
summary_kb(object, ...)
```

Arguments

object	The object (returned by another function) to be described.
...	Ignored arguments.

Details

Unpacks objects returned by other functions with descriptions of their results. The following functions are supported:

- (1) [get.oc.kb\(\)](#), [get.oc.comb.kb\(\)](#), [get.oc.obd.kb\(\)](#), and [get.oc.obd2.kb\(\)](#), which yield the operating characteristics of trials simulated using the Keyboard design.
- (2) [select.mtd.kb\(\)](#) and [select.mtd.comb.kb\(\)](#), which yield the MTD and other statistics.
- (3) [next.comb.kb\(\)](#), which indicates the dose combination to administer to the next cohort.
- (4) [select.obd.kb\(\)](#), which yields the OBD.

Value

This function helps users to interpret the objects returned by other Keyboard package functions.

Author(s)

Xiaomeng Yuan, Chen Li, Hongying Sun, Li Tang and Haitao Pan

See Also

[plot_kb](#)

Examples

```
### Single-agent trial ###

## Summarize the object returned by get.oc.kb()
oc.single <- get.oc.kb(target=0.3, p.true=c(0.05, 0.15, 0.3, 0.45, 0.6),
                      ncohort=10, cohortsize=3, ntrial=1000)
summary_kb(oc.single)

## Summarize the object returned by select.mtd.kb()
n <- c(3, 3, 15, 9, 0)
y <- c(0, 0, 4, 4, 0)
sel.single <- select.mtd.kb(target=0.3, npts=n, ntox=y)
summary_kb(sel.single)

## Summarize the object returned by select.mtd.comb.kb()
n <- matrix(c(6, 3, 0, 0,
              6, 24, 9, 0,
              0, 0, 0, 0), ncol=4, byrow=TRUE)
y <- matrix(c(0, 0, 0, 0,
              1, 5, 4, 0,
              0, 0, 0, 0), ncol=4, byrow=TRUE)
sel.comb <- select.mtd.comb.kb(target=0.25, npts=n, ntox=y)
summary_kb(sel.comb)

## Summarize the object returned by next.comb.kb()
n <- matrix(c(3, 0, 0, 0,
              0, 0, 0, 0,
              0, 0, 0, 0), ncol=4, byrow=TRUE)
y <- matrix(c(0, 0, 0, 0,
              0, 0, 0, 0,
              0, 0, 0, 0), ncol=4, byrow=TRUE)
nxt.comb <- next.comb.kb(target=0.25, npts=n, ntox=y, dose.curr=c(1, 1))
summary_kb(nxt.comb)

## get.oc.obd
toxicity.low <- 0.15
toxicity.moderate <- 0.25
toxicity.high <- 0.35
efficacy.low <- 0.25
efficacy.moderate <- 0.45
```

```
efficacy.high <- 0.65
target.toxicity<-0.20
target.efficacy<-0.40
p.true <-c(0.08,0.20,0.60,0.80)
q.true <- c(0.25,0.40,0.25,0.50)
oc.obd.kb <- get.oc.obd.kb(toxicity.low = toxicity.low,
                           toxicity.moderate= toxicity.moderate,
                           toxicity.high = toxicity.high,
                           efficacy.low = efficacy.low,
                           efficacy.moderate = efficacy.moderate,
                           efficacy.high = efficacy.high,
                           target.toxicity=target.toxicity,
                           target.efficacy= target.efficacy,
                           p.true= p.true, q.true= q.true)
summary_kb(oc.obd.kb)

## OBD selection
target.toxicity<-0.3
target.efficacy<-0.4
npts <- c(3,6,12,3,3)
ntox <- c(1,2,4,2,3)
neff <- c(0,0,5,1,1)
obd <- select.obd.kb (target.toxicity=target.toxicity,
                      target.efficacy= target.efficacy, npts = npts,
                      ntox = ntox, neff = neff)
summary_kb(obd)
```

Index

* drug-combination functions

get.boundary.comb.kb, 4
get.oc.comb.kb, 18
next.comb.kb, 31
select.mtd.comb.kb, 35

* single-agent functions

get.boundary.kb, 7
get.oc.kb, 20
select.mtd.kb, 38

* single-agent phase I/II functions

get.decision.obd.kb, 10
get.decision.obd2.kb, 14
get.oc.obd.kb, 24
get.oc.obd2.kb, 28
select.obd.kb, 40

biviso, 36, 37

decision.finding, 2

example.kb, 3

get.boundary.comb.kb, 4, 20, 32, 37
get.boundary.kb, 7, 23, 40
get.decision.obd.kb, 10, 18, 27, 30, 43
get.decision.obd2.kb, 14, 14, 27, 30, 43
get.oc.comb.kb, 7, 18, 32, 37
get.oc.kb, 10, 20, 40
get.oc.obd.kb, 14, 18, 24, 30, 43
get.oc.obd2.kb, 14, 18, 27, 28, 43

next.comb.kb, 7, 20, 31, 37

plot_kb, 33, 44

select.mtd.comb.kb, 7, 20, 32, 35
select.mtd.kb, 10, 23, 38
select.obd.kb, 14, 18, 27, 30, 40
summary_kb, 34, 43