> get.boundary

function (target, ncohort, cohortsize, n.earlystop = 100, p.saf = "default",

p.tox = "default", cutoff.eli = 0.95, extrasafe = FALSE,

offset = 0.05, print = TRUE)

{

if (p.saf == "default")

p.saf = 0.6 \* target

if (p.tox == "default")

p.tox = 1.4 \* target

if (target < 0.05) {

cat("Error: the target is too low! \n")

return()

}

if (target > 0.6) {

cat("Error: the target is too high! \n")

return()

}

if ((target - p.saf) < (0.1 \* target)) {

cat("Error: the probability deemed safe cannot be higher than or too close to the target! \n")

return()

}

if ((p.tox - target) < (0.1 \* target)) {

cat("Error: the probability deemed toxic cannot be lower than or too close to the target! \n")

return()

}

if (offset >= 0.5) {

cat("Error: the offset is too large! \n")

return()

}

if (n.earlystop <= 6) {

cat("Warning: the value of n.earlystop is too low to ensure good operating characteristics. Recommend n.earlystop = 9 to 18 \n")

return()

}

npts = ncohort \* cohortsize

ntrt = NULL

b.e = NULL

b.d = NULL

elim = NULL

for (n in 1:npts) {

lambda1 = log((1 - p.saf)/(1 - target))/log(target \*

(1 - p.saf)/(p.saf \* (1 - target)))

lambda2 = log((1 - target)/(1 - p.tox))/log(p.tox \* (1 -

target)/(target \* (1 - p.tox)))

cutoff1 = floor(lambda1 \* n)

cutoff2 = ceiling(lambda2 \* n)

ntrt = c(ntrt, n)

b.e = c(b.e, cutoff1)

b.d = c(b.d, cutoff2)

elimineed = 0

if (n < 3) {

elim = c(elim, NA)

}

else {

for (ntox in 1:n) {

if (1 - pbeta(target, ntox + 1, n - ntox + 1) >

cutoff.eli) {

elimineed = 1

break

}

}

if (elimineed == 1) {

elim = c(elim, ntox)

}

else {

elim = c(elim, NA)

}

}

}

for (i in 1:length(b.d)) {

if (!is.na(elim[i]) && (b.d[i] > elim[i]))

b.d[i] = elim[i]

}

boundaries = rbind(ntrt, b.e, b.d, elim)[, 1:min(npts, n.earlystop)]

rownames(boundaries) = c("Number of patients treated", "Escalate if # of DLT <=",

"Deescalate if # of DLT >=", "Eliminate if # of DLT >=")

colnames(boundaries) = rep("", min(npts, n.earlystop))

if (print) {

cat("Escalate dose if the observed toxicity rate at the current dose <= ",

lambda1, "\n")

cat("Deescalate dose if the observed toxicity rate at the current dose >= ",

lambda2, "\n\n")

cat("This is equivalent to the following decision boundaries\n")

print(boundaries[, (1:floor(min(npts, n.earlystop)/cohortsize)) \*

cohortsize])

if (cohortsize > 1) {

cat("\n")

cat("A more completed version of the decision boundaries is given by\n")

print(boundaries)

}

cat("\n")

if (!extrasafe)

cat("Default stopping rule: stop the trial if the lowest dose is eliminated.\n")

}

if (extrasafe) {

stopbd = NULL

ntrt = NULL

for (n in 1:npts) {

ntrt = c(ntrt, n)

if (n < 3) {

stopbd = c(stopbd, NA)

}

else {

for (ntox in 1:n) {

if (1 - pbeta(target, ntox + 1, n - ntox +

1) > cutoff.eli - offset) {

stopneed = 1

break

}

}

if (stopneed == 1) {

stopbd = c(stopbd, ntox)

}

else {

stopbd = c(stopbd, NA)

}

}

}

stopboundary = rbind(ntrt, stopbd)[, 1:min(npts, n.earlystop)]

rownames(stopboundary) = c("The number of patients treated at the lowest dose ",

"Stop the trial if # of DLT >= ")

colnames(stopboundary) = rep("", min(npts, n.earlystop))

if (print) {

cat("\n")

cat("In addition to the default stopping rule (i.e., stop the trial if the lowest dose is eliminated), \n")

cat("the following more strict stopping safety rule will be used for extra safety: \n")

cat(" stop the trial if (1) the number of patients treated at the lowest dose >= 3 AND",

"\n", "(2) Pr(the toxicity rate of the lowest dose >",

target, "| data) > ", cutoff.eli - offset, ",\n",

"which corresponds to the following stopping boundaries:\n")

print(stopboundary)

}

}

if (!print)

return(boundaries)

}

<environment: namespace:BOIN>

**以下是两个用这个function输出的结果举例：**

> get.boundary(0.3,8,3)

Escalate dose if the observed toxicity rate at the current dose <= 0.2364907

Deescalate dose if the observed toxicity rate at the current dose >= 0.3585195

This is equivalent to the following decision boundaries

Number of patients treated 3 6 9 12 15 18 21 24

Escalate if # of DLT <= 0 1 2 2 3 4 4 5

Deescalate if # of DLT >= 2 3 4 5 6 7 8 9

Eliminate if # of DLT >= 3 4 5 7 8 9 10 11

A more completed version of the decision boundaries is given by

Number of patients treated 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

Escalate if # of DLT <= 0 0 0 0 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4 4 5 5 5

Deescalate if # of DLT >= 1 1 2 2 2 3 3 3 4 4 4 5 5 6 6 6 7 7 7 8 8 8 9 9

Eliminate if # of DLT >= NA NA 3 3 4 4 5 5 5 6 6 7 7 8 8 8 9 9 9 10 10 11 11 11

Default stopping rule: stop the trial if the lowest dose is eliminated.

> get.boundary(0.3,8,3,extrasafe = T)

Escalate dose if the observed toxicity rate at the current dose <= 0.2364907

Deescalate dose if the observed toxicity rate at the current dose >= 0.3585195

This is equivalent to the following decision boundaries

Number of patients treated 3 6 9 12 15 18 21 24

Escalate if # of DLT <= 0 1 2 2 3 4 4 5

Deescalate if # of DLT >= 2 3 4 5 6 7 8 9

Eliminate if # of DLT >= 3 4 5 7 8 9 10 11

A more completed version of the decision boundaries is given by

Number of patients treated 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

Escalate if # of DLT <= 0 0 0 0 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4 4 5 5 5

Deescalate if # of DLT >= 1 1 2 2 2 3 3 3 4 4 4 5 5 6 6 6 7 7 7 8 8 8 9 9

Eliminate if # of DLT >= NA NA 3 3 4 4 5 5 5 6 6 7 7 8 8 8 9 9 9 10 10 11 11 11

In addition to the default stopping rule (i.e., stop the trial if the lowest dose is eliminated),

the following more strict stopping safety rule will be used for extra safety:

stop the trial if (1) the number of patients treated at the lowest dose >= 3 AND

(2) Pr(the toxicity rate of the lowest dose > 0.3 | data) > 0.9 ,

which corresponds to the following stopping boundaries:

The number of patients treated at the lowest dose 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

Stop the trial if # of DLT >= NA NA 2 3 3 4 4 4 5 5 6 6 6 7 7 8 8 8 9 9 9 10 10 10