Package ‘interAdapt’

February 20, 2015

Title interAdapt

Description A shiny application for designing adaptive clinical trials. For more details, see: http://arxiv.org/abs/1404.0734

Version 0.1

Author Aaron Fisher <afisher@jhsph.edu>, Michael Rosenblum <mrosenbl@jhsph.edu>, Harris Jaffee <hj@jhu.edu>

Maintainer Michael Rosenblum <mrosenbl@jhsph.edu>

Depends R (>= 3.0.1)

Imports shiny, knitcitations, RCurl, mvtnorm, knitr

License GPL-2

LazyData true

NeedsCompilation no

Repository CRAN

Date/Publication 2014-08-03 08:03:00

R topics documented:

compute_design_performance .................................................. 1
default_parameter_table ...................................................... 6
runInterAdapt ................................................................. 6

Index 7

compute_design_performance

Compute Design Performance
compute_design_performance

**Description**

Generates decision rules for group sequential trial designs with adaptive enrollment criteria. Tables are also generated which compare the performance of these designs to the performance of standard group sequential designs. We use the notation AD to refer to the design with adaptive enrollment, SC to refer to a standard group sequential design enrolling from the combined population, and SS to refer to a standard group sequential design enrolling from only the subpopulation where there is greater prior evidence of a positive treatment effect. Further details are provided in (Fisher et al. 2014).

**Usage**

```r
compute_design_performance(p_1 = 0.33, p_10 = 0.25, p_20 = 0.2,
                         p_11 = p_10 + 0.125, per_stage_sample_size_combined_AD = 280,
                         per_stage_sample_size_subpop_1_AD = 148, FWER = 0.025,
                         FWER_H0C_proportion = 0.09, Delta = -0.5, iter = 10000,
                         time_limit = 45, total_number_stages = 5,
                         last_stage_subpop_2_enrolled_AD = 3,
                         enrollment_rate_combined_population = 420,
                         per_stage_sample_size_combined_SC = 106, per_stage_sample_size_SS = 100,
                         subpop_2_stop_boundary_constant_AD = 0,
                         H01_futility_boundary_constant_AD = 0,
                         H0C_futility_boundary_constant_SC = -0.1,
                         H01_futility_boundary_constant_SS = -0.1,
                         lower_bound_treatment_effect_subpop_2 = -0.2,
                         upper_bound_treatment_effect_subpop_2 = 0.2, CSV)
```

**Arguments**

- `p_1`: The proportion of the population in subpopulation 1, which is the subpopulation having stronger prior evidence of a positive treatment effect.
- `p_10`: The probability of a successful outcome for subpopulation 1 under assignment to the control arm. This is used in estimating power and expected sample size.
- `p_20`: The probability of a successful outcome for subpopulation 2 under assignment to the control arm. This is used in estimating power and expected sample size.
- `p_11`: The probability of a successful outcome for subpopulation 1 under assignment to the treatment arm. Note that the user does not specify the probability of success under treatment for subpopulation 2 ($p_{2t}$). Instead, `compute_design_performance` considers a range of possible values of $p_{2t}$ (see the `lower_bound_treatment_effect_subpop_2` and `upper_bound_treatment_effect_subpop_2` arguments).
- `per_stage_sample_size_combined_AD`: The number of participants enrolled per stage in the adaptive design while both subpopulations are being enrolled.
- `per_stage_sample_size_subpop_1_AD`: The number of participants required for each stage in the adaptive design after enrollment for subpopulation 2 has been stopped.
- `FWER`: The familywise Type I error rate ($\alpha$) for all designs (see Fisher et al. 2014).
Proportion of $\alpha$ allocated to $H_{0C}$ for the adaptive design. Here, $H_{0C}$ refers to the null hypothesis of no treatment effect in the combined population.

Used as the exponent in defining the efficacy and futility boundaries, as described (Fisher et al. 2014).

The number of simulated trials used to estimate the power, expected sample size, and expected trial duration.

Time limit for simulation in seconds. If the simulation exceeds the time limit, calculations will stop and the user will get an error message ("reached CPU time limit"). See `setTimeLimit`. To avoid this, the number of iterations (`iter`) can be reduced or the time limit can be increased.

Total number of stages used in each design ($K$). The maximum allowable number of stages is 20.

The last stage subpopulation 2 is enrolled, under the adaptive design. We refer to this stage number as $k^*$.

The assumed enrollment rate per year for the combined population. This impacts the expected duration of each trial design. Active enrollments from the two subpopulations are assumed to be independent. The enrollment rates for subpopulations 1 and 2 are assumed proportional, based on $p_1$. This implies that each stage of the adaptive design up to and including stage $k^*$ takes the same amount of time to complete, regardless of whether or not enrollment stops for subpopulation 2. Each stage after $k^*$ will also take the same amount of time to complete.

The number of participants enrolled in each stage of the standard group sequential design enrolling the combined population (SC).

The number of participants enrolled in each stage for standard group sequential design enrolling only subpopulation 1 (SS).

Stopping boundary proportionality constant for subpopulation 2 enrollment in the adaptive design.

Futility boundary proportionality constant for $H_{01}$ in the adaptive design. This is used to calculate the futility boundaries ($l_{1,k}$) for the z-statistics calculated in subpopulation 1 ($Z_{1,k}$) as defined in (Fisher et al. 2014).

Futility boundary proportionality constant for $H_{0C}$ in the standard design always enrolling from the combined population.

Futility boundary proportionality constant for $H_{01}$ in the standard design only enrolling from subpopulation 1.
lower_bound_treatment_effect_subpop_2
Simulations are performed under a range of treatment effect sizes for subpopulation 2 (i.e. \( p_{(2t)} - p_{(2c)} \)). This parameter sets the lower bound for this range. This effectively sets the lower bound for the probability of success under treatment for subpopulation 2 (\( p_{(2t)} \)), since \( p_{(20)} \) is set by the user.

upper_bound_treatment_effect_subpop_2
Simulations are performed under a range of treatment effect sizes for subpopulation 2 (i.e. \( p_{(2t)} - p_{(2c)} \)). This parameter sets the upper bound for this range.

CSV
Rather than manually entering the arguments above, this allows for the arguments to be entered via a tabular csv file. The CSV argument should contain a character vector or list of csv filenames. The table must minimally include the columns "inputId" and "value" (as in the \texttt{default_parameter_table}).

Details
This function is meant to be applied when there is prior evidence that a treatment might work better in a one subpopulation than in another. In this context, a trial with an adaptive enrollment criteria would determine whether or not to continue enrolling patients from each subpopulation based on interim analyses of whether each subpopulation is benefiting. In order for the type I error and the power of the trial to be calculable, the decision rules for changing enrollment must be set before the trial starts. This function generates decision rules for group sequential trial designs with adaptive enrollment criteria, and compares the performance of these designs against standard group sequential designs with fixed enrollment criteria. Performance is compared in terms of power, expected sample size, and expected trial duration.

Value
A list with 5 elements:

- \texttt{performance\_comparison}
  A table comparing the performance of the three trials, in terms of power, expected sample size, and expected duration. See examples.
- \texttt{AD\_design}
  Efficacy and futility boundaries for the group sequential design with adaptive enrollment
- \texttt{SC\_design}
  Efficacy and futility boundaries for the standard group sequential design enrolling from the combined population
- \texttt{SS\_design}
  Efficacy and futility boundaries for the standard group sequential design enrolling subpopulation 1 only
- \texttt{input\_parameters}
  List of input argument values

References
Examples

```r
# Store function output
o <- compute_design_performance()

names(o)

# Decision rules for trial designs
print(o$SS[[1]])
print(o$SC[[1]])
print(o$AD[[1]])

# Plot decision rules
par(mfrow = c(1, 3))
matplot(t(o$SS[[1]][2:3]), type = 'o',
main = 'Standard trial - subpop 1',
ylab = 'Z-statistic',
col = 'blue', pch = 2:3, lty = 3)
legend('topright', c('H01 Efficacy', 'H01 Futility'),
col = 'blue', pch = 2:3, lty = 3)
matplot(t(o$SC[[1]][4:5]), type = 'o',
main = 'Standard trial - combined pop',
ylab = 'Z-statistic', col = 'red', pch = 0:1, lty = 3)
legend('topright', c('H0C Efficacy', 'H0C Futility'),
col = 'red', pch = 0:1, lty = 3)
matplot(t(o$AD[[1]][4:7]), type = 'o',
main = 'Adaptive Enrollment',
ylab = 'Z-statistic', col = c('red', 'red', 'blue', 'blue'),
pch = 0:3, lty = 3)
legend('topright', c('H0C Efficacy', 'H0C Futility',
'H01 Efficacy', 'H01 Futility'), col = c('red',
'red', 'blue', 'blue'), pch = 0:3, lty = 3)

# Check performance
o$performance[[1]]

# Plot performance over a range of treatment effects for subpop2
col1 <- c('black', 'black', 'black', 'green', 'blue')

perform_names <- rownames(o$performance[[1]])

# Index for parts of the table corresponding to power
p_ind <- grep('Power', perform_names)
# Index for parts of the table corresponding to sample size
s_ind <- grep('Sample Size', perform_names)
# Index for parts of the table corresponding to trial duration
d_ind <- grep('DUR', perform_names)

par(mfrow = c(1, 3))
```
```r
lty1<-c(2,3,1,4,5)
matplot(x=t(o$performance[[1]][1,]),t(o$performance[[1]][p_ind,]),
type='l',lty=lty1,col=col1[1:5],xlab='Subpop.2 Tx. Effect',
ylab='Power',main='Power')
legend('bottomleft',perform_names[p_ind],col=col1[1:length(p_ind)],
lty=lty1)

matplot(x=t(o$performance[[1]][1,]),t(o$performance[[1]][s_ind,]),
type='l',lty=1:3,col=col1[3:5],xlab='Subpop.2 Tx. Effect',
ylab='Expected Sample Size',main='Expected Sample Size')
legend('topright',substr(perform_names[d_ind],1,2),col=col1[3:5],
lty=1:3)

matplot(x=t(o$performance[[1]][1,]),t(o$performance[[1]][d_ind,]),
type='l',lty=1:3,col=col1[3:5],xlab='Subpop.2 Tx. Effect',
ylab='Expected Duration',main='Expected Duration')
legend('topright',substr(perform_names[d_ind],1,2),col=col1[3:5],
lty=1:3)
```

**default_parameter_table**

*Table of default parameters*

**Description**

A table of the default parameters entered to `compute_design_performance`.

**Details**

Each row of this table corresponds to one input to `compute_design_performance`, and contains: (1) the name of the argument passed to `compute_design_performance`, (2) a label describing the input in more detail, (3) the minimum value the input can take, (4) the maximum value the input can take, and (5) the input's default value.

**runInterAdapt**

*Run the interAdapt shiny application*

**Description**

`runInterAdapt` Runs the interactive shiny application

**Usage**

`runInterAdapt()`
Index

*Topic Adaptive
  compute_design_performance, 1
*Topic Group
  compute_design_performance, 1
*Topic clinical
  compute_design_performance, 1
*Topic designs
  compute_design_performance, 1
*Topic enrollment,
  compute_design_performance, 1
*Topic sequential
  compute_design_performance, 1
*Topic trial,
  compute_design_performance, 1
compute_design_performance, 1, 6
default_parameter_table, 4, 6
runInterAdapt, 6
setTimeLimit, 3