Package ‘cats’

March 11, 2022

Type Package
Title Cohort Platform Trial Simulation
Version 1.0.2
Author Elias Laurin Meyer [aut, cre],
Peter Mesenbrink [ctb],
Franz Koenig [ctb]
Maintainer Elias Laurin Meyer <elias.meyer@meduniwien.ac.at>
Description Cohort plAtform Trial Simulation whereby every cohort consists of two arms,
control and experimental treatment. Endpoints are co-primary binary endpoints and
decisions are made using either Bayesian or frequentist decision rules.
Realistic trial trajectories are simulated and the operating characteristics of the
designs are calculated.
License MIT + file LICENSE
Encoding UTF-8
Depends R (>= 3.5.0)
RoxygenNote 7.1.2
Imports dplyr, purrr, ggplot2, plotly, tidyr, parallel, doParallel,
foreach, openxlsx, forcats, epitools, zoo, mvtnorm
Suggests knitr, rmarkdown, DT, gtools
NeedsCompilation no
Repository CRAN
Date/Publication 2022-03-11 10:20:07 UTC

R topics documented:

make_decision_trial ........................................ 2
simulate_trial .................................................. 5
trial_ocs ....................................................... 9

Index 12
make_decision_trial  Checks whether decision criteria are met and updates trial results accordingly.

Description

Given a res_list object, checks the supplied decision criteria and saves the results in the res_list file.

Usage

```r
make_decision_trial(
  res_list,
  which_cohort,
  Bayes_Sup1 = NULL,
  Bayes_Fut1 = NULL,
  Bayes_Sup2 = NULL,
  Bayes_Fut2 = NULL,
  w = 0.5,
  analysis_number,
  beta_prior = 0.5,
  hist_lag,
  endpoint_number,
  analysis_time,
  dataset,
  hist_miss = TRUE,
  sharing_type,
  ...
)
```

Arguments

- **res_list**: List item containing individual cohort trial results so far in a format used by the other functions in this package
- **which_cohort**: Current cohort that should be evaluated
- **Bayes_Sup1**: List of matrices with rows corresponding to number of multiple Bayesian posterior two-arm combination criteria for superiority of histology endpoint 1
- **Bayes_Fut1**: List of matrices with rows corresponding to number of multiple Bayesian posterior two-arm combination criteria for futility of histology endpoint 1
- **Bayes_Sup2**: List of matrices with rows corresponding to number of multiple Bayesian posterior two-arm combination criteria for superiority of histology endpoint 2
- **Bayes_Fut2**: List of matrices with rows corresponding to number of multiple Bayesian posterior two-arm combination criteria for futility of histology endpoint 2
- **w**: If dynamic borrowing, what is the prior choice for w. Default is 0.5.
- **analysis_number**: 1st, second or third analysis?
\textit{make\_decision\_trial}

- \texttt{beta\_prior}  \hspace{1cm} Prior parameter for all Beta Distributions. Default is 0.5.
- \texttt{hist\_lag}  \hspace{1cm} Histology Lag
- \texttt{endpoint\_number}  \hspace{1cm} Should histology endpoint 1 or 2 be evaluated?
- \texttt{analysis\_time}  \hspace{1cm} Platform Time of Analysis
- \texttt{dataset}  \hspace{1cm} Dataset to be used for analysis
- \texttt{hist\_miss}  \hspace{1cm} Whether or not to exclude missing histology data
- \texttt{sharing\_type}  \hspace{1cm} Type of Data Sharing to perform
- ...  \hspace{1cm} Further arguments inherited from simulate\_trial

**Value**

List containing original res\_list and results of decision rules

**Examples**

```r
# Example 1

df <- matrix(nrow = 100, ncol = length(cols))
colnames(df) <- cols
df <- as.data.frame(df)
df$PatID <- 1:100
df$ArrivalTime <- sort(runif(100, min = 0, max = 5))
df$Cohort <- sample(1:2, 100, replace = TRUE)
df$Arm <- sample(c("Combo", "Plac"), 100, replace = TRUE)
df$RespHist1 <- sample(0:1, 100, replace = TRUE)
df$RespHist2 <- sample(0:1, 100, replace = TRUE)
df$HistMissing <- sample(0:1, 100, replace = TRUE, prob = c(0.95, 0.05))

# Initialize res\_list Object

res_list <-
rep(
  list(
    list(
      Meta = list(
        decision = rep("none", 3),
        decision_hist1 = rep("none", 3),
        decision_hist2 = rep("none", 3),
        start_n = 0,
        start_time = 0,
        pat\_enrolled = 0
      ),
      Arms = rep(
        list(
          list(
            rr = NULL
          ),
          list(
            rr = NULL
          ),
          list(
            rr = NULL
          )
        ),
        rr = NULL
      )
    ),
    list(
      Meta = list(
        decision = rep("none", 3),
        decision_hist1 = rep("none", 3),
        decision_hist2 = rep("none", 3),
        start_n = 0,
        start_time = 0,
        pat\_enrolled = 0
      ),
      Arms = rep(
        list(
          list(
            rr = NULL
          ),
          list(
            rr = NULL
          ),
          list(
            rr = NULL
          )
        ),
        rr = NULL
      )
    )
  ),
  list(
    list(
      Meta = list(
        decision = rep("none", 3),
        decision_hist1 = rep("none", 3),
        decision_hist2 = rep("none", 3),
        start_n = 0,
        start_time = 0,
        pat\_enrolled = 0
      ),
      Arms = rep(
        list(
          list(
            rr = NULL
          ),
          list(
            rr = NULL
          ),
          list(
            rr = NULL
          )
        ),
        rr = NULL
      )
    ),
    list(
      Meta = list(
        decision = rep("none", 3),
        decision_hist1 = rep("none", 3),
        decision_hist2 = rep("none", 3),
        start_n = 0,
        start_time = 0,
        pat\_enrolled = 0
      ),
      Arms = rep(
        list(
          list(
            rr = NULL
          ),
          list(
            rr = NULL
          ),
          list(
            rr = NULL
          )
        ),
        rr = NULL
      )
    )
  )
)
```

hist_observed = 0
)
),
2
)
)
),
2
)

arm_names <- c("Comb", "Plac")

for (i in 1:2) {
  names(res_list[[i]]) <- paste0("Cohort", i)
  names(res_list[[i]]$Arms) <- arm_names

  res_list[[i]]$Arms$Comb$rr <- matrix(c(0.2, 0.2), ncol = 2)
  res_list[[i]]$Arms$Plac$rr <- matrix(c(0.1, 0.1), ncol = 2)
}

sharing_type <- "all"
analysis_number <- 3
which_cohort <- 1
endpoint_number <- 2
hist_lag <- 1
analysis_time <- 6

# Comparison IA1
Bayes_Sup11 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup11[1,] <- c(0.00, 0.95)
Bayes_Sup11[2,] <- c(0.10, 0.80)

# Comparison IA2
Bayes_Sup12 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup12[1,] <- c(0.00, 0.95)
Bayes_Sup12[2,] <- c(NA, NA)

# Comparison IA3
Bayes_Sup13 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup13[1,] <- c(0.00, 0.95)
Bayes_Sup13[2,] <- c(0.10, 0.80)

Bayes_Sup1 <- Bayes_Sup2 <- list(list(Bayes_Sup11), list(Bayes_Sup12), list(Bayes_Sup13))

# DO NOT RUN
res_list2 <-
make_decision_trial(
  res_list = res_list, which_cohort = which_cohort,
  analysis_number = analysis_number, endpoint_number = endpoint_number,
  Bayes_Sup1 = Bayes_Sup1, Bayes_Sup2 = Bayes_Sup2,
  dataset = df, analysis_time = analysis_time, hist_lag = hist_lag,
  sharing_type = sharing_type
)
simulate_trial

Simulates the cohort trial.

Description
Simulates the cohort trial.

Usage

```r
simulate_trial(
  n_fin,
  cohorts_start = 1,
  composite = "or",
  rr_comb1,
  rr_plac1,
  rr_comb2,
  rr_plac2,
  random_type = NULL,
  random = FALSE,
  correlation,
  prob_comb1_rr = NULL,
  prob_plac1_rr = NULL,
  prob_comb2_rr = NULL,
  prob_plac2_rr = NULL,
  stage_data = FALSE,
  cohort_random = NULL,
  cohorts_max = 4,
  sr_drugs_pos = 1,
  sr_pats = cohorts_max * (n_fin + 3 * cohorts_max),
  sr_first_pos = FALSE,
  cohort_offset = 0,
  sharing_type = "all",
  safety_prob = 0,
  cohorts_sim = Inf,
  missing_prob = 0,
  cohort_fixed = NULL,
  accrual_type = "fixed",
  accrual_param = 9,
  hist_lag = 48,
  analysis_times = c(0.5, 0.75, 1),
  time_trend = time_trend,
  ...
)
```

Arguments

- `n_fin` Sample size per cohort at final
**coHORTS_START**: Number of cohorts to start the platform with.

**COMPOSITE**: Rule for deriving the composite endpoint. By default "or", otherwise "and".

**rr_comb1**: Response rates of treatment, histology endpoint 1.

**rr_plac1**: Response rate of the SoC, histology endpoint 1.

**rr_comb2**: Response rates of treatment, histology endpoint 2.

**rr_plac2**: Response rate of the SoC, histology endpoint 2.

**random_type**: How should the response rates be drawn randomly? Options are:
- "absolute": Specify absolute response rates that will be drawn with a certain probability
- "risk_difference": Specify absolute response rates for placebo which will be drawn randomly, plus specify vectors for absolute treatment effects of mono therapies over placebo and for combo over the mono therapies.
- "risk_ratio": Specify absolute response rates for placebo which will be drawn randomly, plus specify vectors for relative treatment effects of mono therapies over placebo and for combo over the mono therapies.
- "odds_ratios": Specify response rate for placebo, specify odds-ratios for mono therapies (via rr_back and rr_mono) and respective probabilities. On top, specify interaction for the combination therapy via rr_comb with prob_rr_comb.
  Set: odds_comb = odds_plac * or_mono1 * or_mono2 * rr_comb. If rr_comb > 1 -> synergistic, if rr_comb = 1 -> additive. If rr_comb < 1 -> antagonistic. Default is "NULL".

**random**: Should the response rates of the arms be randomly drawn from rr_exp? Default is FALSE.

**correlation**: Correlation between histology endpoints.

**prob_comb1_rr**: If random == TRUE, what are the probabilities with which the elements of rr_comb1 should be drawn?

**prob_plac1_rr**: If random == TRUE, what are the probabilities with which the elements of rr_plac1 should be drawn?

**prob_comb2_rr**: If random == TRUE, what are the probabilities with which the elements of rr_comb2 should be drawn?

**prob_plac2_rr**: If random == TRUE, what are the probabilities with which the elements of rr_plac2 should be drawn?

**stage_data**: Should individual stage data be passed along? Default is TRUE.

**coHORT_Random**: If not NULL, indicates that new arms/cohorts should be randomly started. For every timestep, there is a cohort_random probability that a new cohort will be started.

**coHORTS_MAX**: Maximum number of cohorts that are allowed to be added throughout the trial.

**sr_drugs_pos**: Stopping rule for successful experimental arms; Default = 1.

**sr_pats**: Stopping rule for total number of patients; Default = cohorts_max * n_fin + error term based on randomization.

**sr_first_pos**: Stopping rule for first successful cohort; if TRUE, after first cohort was found to be successful, no further cohorts will be included but cohorts will finish evaluating, unless other stopping rules reached prior. Default is FALSE.
**Value**

List containing: Responses and patients on experimental and control arm, total treatment successes and failures and final p-value

**Examples**

```r
random <- TRUE
r1 <- 0.10
prob_comb1_rr <- 1
rr_comb2 <- 0.45
prob_comb2_rr <- 1
rr_plac1 <- 0.10
prob_plac1_rr <- 1
rr_plac2 <- 0.20
prob_plac2_rr <- 1
correlation <- 0.8
cohorts_start <- 2
cohorts_max <- 5
safety_prob <- 0
sharing_type <- "concurrent"
sr_drugs_pos <- 5
sr_first_pos <- FALSE
n_fin <- 100
stage_data <- TRUE
cohort_random <- 0.01
cohort_offset <- 0
cohorts_sim <- Inf
random_type <- "absolute"
```
```r
missing_prob <- 0.2
cohort_fixed <- 5
hist_lag <- 48
analysis_times <- c(0.5, 0.75, 1)
accrual_type <- "fixed"
accrual_param <- 9
time_trend <- 0.001
composite <- "or"

# Comparison IA1
Bayes_Sup11 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup11[,1] <- c(0.00, 0.95)
Bayes_Sup11[,2] <- c(0.10, 0.80)
# Comparison IA2
Bayes_Sup12 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup12[,1] <- c(0.00, 0.95)
Bayes_Sup12[,2] <- c(0.10, 0.80)
# Comparison IA3
Bayes_Sup13 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup13[,1] <- c(0.00, 0.95)
Bayes_Sup13[,2] <- c(0.10, 0.80)
Bayes_Sup1 <- Bayes_Sup2 <- list(list(Bayes_Sup11), list(Bayes_Sup12), list(Bayes_Sup13))

# Comparison IA1
Bayes_Fut11 <- matrix(nrow = 1, ncol = 2)
Bayes_Fut11[,1] <- c(0.00, 0.20)
# Comparison IA2
Bayes_Fut12 <- matrix(nrow = 1, ncol = 2)
Bayes_Fut12[,1] <- c(0.00, 0.30)
# Comparison IA3
Bayes_Fut13 <- matrix(nrow = 1, ncol = 2)
Bayes_Fut13[,1] <- c(NA, NA)
# Endpoint 1+2
Bayes_Fut1 <- Bayes_Fut2 <- list(list(Bayes_Fut11), list(Bayes_Fut12), list(Bayes_Fut13))

simulate_trial(
  n_fin = n_fin, random_type = random_type, composite = composite,
  rr_comb1 = rr_comb1, rr_comb2 = rr_comb2, rr_plac1 = rr_plac1, rr_plac2 = rr_plac2,
  random = random, prob_comb1_rr = prob_comb1_rr, prob_comb2_rr = prob_comb2_rr,
  prob_plac1_rr = prob_plac1_rr, prob_plac2_rr = prob_plac2_rr, correlation = correlation,
  stage_data = stage_data, cohort_random = cohort_random, cohorts_max = cohorts_max,
  sr_drugs_pos = sr_drugs_pos, sharing_type = sharing_type, Bayes_Fut1 = Bayes_Fut1,
  safety_prob = safety_prob, Bayes_Sup1 = Bayes_Sup1, Bayes_Sup2 = Bayes_Sup2,
  cohort_offset = cohort_offset, sr_first_pos = sr_first_pos, Bayes_Fut2 = Bayes_Fut2,
  missing_prob = missing_prob, cohort_fixed = cohort_fixed, accrual_type = accrual_type,
  accrual_param = accrual_param, hist_lag = hist_lag, analysis_times = analysis_times,
  time_trend = time_trend, cohorts_start = cohorts_start, cohorts_sim = cohorts_sim
)
```
Calculates the operating characteristics of the cohort trial

Description
Given the trial specific design parameters, performs a number of simulations of the trial and saves the result in an Excel file

Usage

```r
trial_ocs(
  iter,
  coresnum = 1,
  save = FALSE,
  path = NULL,
  filename = NULL,
  ret_list = FALSE,
  ret_trials = FALSE,
  plot_ocs = FALSE,
  export = NULL,
  ...
)
```

Arguments

- `iter`: Number of program simulations that should be performed
- `coresnum`: How many cores should be used for parallel computing
- `save`: Indicator whether simulation results should be saved in an Excel file
- `path`: Path to which simulation results will be saved; if NULL, then save to current path
- `filename`: Filename of saved Excel file with results; if NULL, then name will contain design parameters
- `ret_list`: Indicator whether function should return list of results
- `ret_trials`: Indicator whether individual trial results should be saved as well
- `plot_ocs`: Should OCs stability plots be drawn?
- `export`: Should any other variables be exported to the parallel tasks?
- `...`: All other design parameters for chosen program

Value
List containing: Responses and patients on experimental and control arm, total treatment successes and failures and final p-value
Examples

random <- TRUE

rr_comb1 <- 0.25
prob_comb1_rr <- 1
rr_comb2 <- 0.20
prob_comb2_rr <- 1
rr_plac1 <- 0.10
prob_plac1_rr <- 1
rr_plac2 <- 0.10
prob_plac2_rr <- 1

correlation <- 0.8

cohorts_start <- 2
cohorts_max <- 5
safety_prob <- 0
sharing_type <- "concurrent"
sr_drugs_pos <- 5
sr_first_pos <- FALSE
n_fin <- 100
stage_data <- TRUE
cohort_random <- 0.01
cohort_offset <- 0
cohorts_sim <- Inf
random_type <- "absolute"
missing_prob <- 0.2
cohort_fixed <- 5
hist_lag <- 48
analysis_times <- c(0.5, 0.75, 1)
accrual_type <- "fixed"
accrual_param <- 9
time_trend <- 0.001
composite <- "or"

# Comparison IA1
Bayes_Sup11 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup11[1,] <- c(0.00, 0.95)
Bayes_Sup11[2,] <- c(0.10, 0.80)
# Comparison IA2
Bayes_Sup12 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup12[1,] <- c(0.00, 0.95)
Bayes_Sup12[2,] <- c(0.10, 0.80)
# Comparison IA3
Bayes_Sup13 <- matrix(nrow = 2, ncol = 2)
Bayes_Sup13[1,] <- c(0.00, 0.95)
Bayes_Sup13[2,] <- c(0.10, 0.80)

Bayes_Sup1 <- Bayes_Sup2 <- list(list(Bayes_Sup11), list(Bayes_Sup12), list(Bayes_Sup13))

ocs <- trial_ocs(
n_fin = n_fin, random_type = random_type, composite = composite,
rr_comb1 = rr_comb1, rr_comb2 = rr_comb2, rr_plac1 = rr_plac1, rr_plac2 = rr_plac2,
random = random, prob_comb1_rr = prob_comb1_rr, prob_comb2_rr = prob_comb2_rr,
prob_plac1_rr = prob_plac1_rr, prob_plac2_rr = prob_plac2_rr,
stage_data = stage_data, cohort_random = cohort_random, cohorts_max = cohorts_max,
sr_drugs_pos = sr_drugs_pos, sharing_type = sharing_type, correlation = correlation,
safety_prob = safety_prob, Bayes_Sup1 = Bayes_Sup1, Bayes_Sup2 = Bayes_Sup2,
cohort_offset = cohort_offset, sr_first_pos = sr_first_pos,
missing_prob = missing_prob, cohort_fixed = cohort_fixed, accrual_type = accrual_type,
accrual_param = accrual_param, hist_lag = hist_lag, analysis_times = analysis_times,
time_trend = time_trend, cohorts_start = cohorts_start, cohorts_sim = cohorts_sim,
iter = 2, coresnum = 1, save = FALSE, ret_list = TRUE, plot_ocs = TRUE)
Index

make_decision_trial, 2
simulate_trial, 5
trial_ocs, 9