

Package: winratiosim (via r-universe)

July 7, 2026

Type Package

Title Power Analysis Simulation for Sample Size Calculation Using the Win Ratio Endpoint

Version 1.0.0

Description Implements simulation-based power analysis for determining sample size in clinical trials using the win ratio endpoint.

License GPL-2 | file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Imports dplyr, foreach, doSNOW, pbmcapply, kableExtra, plyr, magrittr, parallel, stats, utils

URL <https://github.com/yain22/winratiosim>

BugReports <https://github.com/yain22/winratiosim/issues>

Depends R (>= 4.0.0)

Config/pak/sysreqs cmake libfontconfig1-dev libfreetype6-dev libfribidi-dev make libharfbuzz-dev libicu-dev libpng-dev libuv1-dev libxml2-dev

Repository <https://yain22.r-universe.dev>

Date/Publication 2025-07-15 21:42:20 UTC

RemoteUrl <https://github.com/yain22/winratiosim>

RemoteRef HEAD

RemoteSha 666c0e0c394c428fe62a79775f5ed3a2e0bd65d3

Contents

binom.conf.exact	2
Scoring_Conti	3
Scoring_TTE	4
SimData_per_group	5
winratiosim	7
WR_analysis	9

binom.conf.exact *Exact Binomial Confidence Interval Using Clopper-Pearson Method*

Description

Computes an exact two-sided confidence interval for a binomial proportion using the Clopper-Pearson method, based on the F-distribution quantiles. This method guarantees coverage by inverting the binomial test and is commonly used for small sample sizes or when exact methods are preferred over asymptotic approximations.

Usage

```
binom.conf.exact(x, n, alpha = 0.05)
```

Arguments

x	Integer. Number of observed successes.
n	Integer. Total number of trials.
alpha	Numeric. Significance level for the confidence interval (default is 0.05 for a 95% CI).

Value

A named numeric vector with three elements:

PointEst The observed proportion (x/n).

Lower The lower bound of the exact two-sided confidence interval.

Upper The upper bound of the exact two-sided confidence interval.

Examples

```
binom.conf.exact(x = 8, n = 10)  
binom.conf.exact(x = 50, n = 100, alpha = 0.01)
```

Description

Assigns win/loss/tie scores to matched subject pairs based on continuous outcomes, such as hospitalization rate or quality-of-life score. This function is typically used in the second or third layer of a hierarchical win ratio framework, following initial scoring based on time-to-event outcomes.

Usage

```
Scoring_Conti(dataset, higher_better, var1, var2)
```

Arguments

dataset	A data frame containing matched subject pairs and prior win/loss indicators (i.e., column score).
higher_better	Character. Set to "Yes" if higher values of the outcome indicate better health (e.g., QoL), or "No" if lower values are preferred (e.g., hospitalization count).
var1	Character. Name of the variable for subject 1 (e.g., "kccq1" or "HFH_Annual1").
var2	Character. Name of the variable for subject 2 (e.g., "kccq2" or "HFH_Annual2").

Details

The function only assigns scores to pairs where the current score is NA or 0, indicating that prior layers (e.g., death) did not resolve the comparison.

A small tolerance (10^{-10}) is used to define ties for nearly equal values.

Value

A modified version of the input dataset, where:

score Updated to 1 (subject 1 wins), -1 (subject 2 wins), 0 (tie), or NA (missing values prevent comparison).

WR_cat Updated win/loss explanation at this layer (e.g., "kccq winner").

Examples

```
## Not run:
df_scored <- Scoring_Conti(
  dataset = df_after_TTE,
  higher_better = "Yes",
  var1 = "kccq1", var2 = "kccq2"
)
table(df_scored$score)

## End(Not run)
```

 Scoring_TTE

Score Time-to-Event Pairwise Comparisons for Win Ratio

Description

This function assigns win/loss/tie scores to matched subject pairs based on their time-to-event outcomes (e.g., death or other hard endpoints). It is designed to be used as the first layer in a hierarchical win ratio framework. The scoring accounts for censoring and ties due to simultaneous or unknown outcomes.

Usage

```
Scoring_TTE(dataset, var1, var2, censor1, censor2)
```

Arguments

dataset	A data frame containing matched pairs of subjects with outcome and censoring indicators.
var1	Character. Name of the column representing the time-to-event variable for the subject in position 1 (e.g., "deathdays1").
var2	Character. Name of the column representing the time-to-event variable for the subject in position 2 (e.g., "deathdays2").
censor1	Character. Name of the censoring indicator variable (1 = event occurred, 0 = censored) for subject 1.
censor2	Character. Name of the censoring indicator variable for subject 2.

Details

- A win is declared if subject 1 survives significantly longer than subject 2.
- A loss is declared if subject 2 survives significantly longer than subject 1.
- Ties or unresolved scores occur if censoring interferes with outcome comparison or both are censored.

Value

A modified data frame identical to the input dataset but with updated values for:

score Win/loss/tie indicator: 1 if subject 1 wins, -1 if subject 2 wins, NA for tie or unresolvable comparison.

WR_cat Character column indicating whether subject 1 is a "winner" or "loser" due to the time-to-event outcome.

Examples

```
## Not run:
df_scored <- Scoring_TTE(
  dataset = df_FS_input,
  var1 = "deathdays1", var2 = "deathdays2",
  censor1 = "death1", censor2 = "death2"
)
head(df_scored$score)

## End(Not run)
```

SimData_per_group

Simulate Individual-Level Trial Data for a Treatment Group

Description

Generates individual-level simulated data for a treatment or control group in a hierarchical win ratio trial. The simulation includes frailty-adjusted time-to-death, recurrent hospitalization events, censoring times, and longitudinal quality-of-life scores (e.g., KCCQ). It outputs a subject-level summary suitable for further analysis.

Usage

```
SimData_per_group(
  treatment,
  ngroup,
  alpha.JFM,
  theta.JFM,
  lambda,
  ann.icr,
  censorrate,
  xbase,
  xfinal,
  sd.delta.x
)
```

Arguments

treatment	Integer. Treatment group indicator (1 for treatment, 0 for control).
ngroup	Integer. Number of subjects to simulate in this group.
alpha.JFM	Numeric. Alpha parameter for the joint frailty model for mortality.
theta.JFM	Numeric. Theta parameter (frailty variance) for the joint frailty model.
lambda	Numeric. Annual mortality rate.
ann.icr	Numeric. Annual incidence rate of recurrent events (e.g., hospitalizations).
censorrate	Numeric. Annual censoring rate.

<code>xbase</code>	Numeric. Baseline quality-of-life score (e.g., KCCQ).
<code>xfinal</code>	Numeric. Expected final quality-of-life score after follow-up (if no death).
<code>sd.delta.x</code>	Numeric. Standard deviation for change in QoL score.

Details

This function is called internally by `winratiosim()` and assigns the output as a global variable (`surv_1` or `surv_0`) based on the treatment group.

- Mortality is modeled via a frailty-adjusted exponential distribution.
- Recurrent events (gap times) are modeled as exponential with frailty.
- Follow-up is administratively censored at 360 days or lost to follow-up.
- KCCQ scores are simulated based on a normal distribution bounded between -100 and 100.

Value

This function does not return an object directly. Instead, it assigns a data frame named `surv_1` or `surv_0` to the global environment depending on the group. The data frame contains the following columns:

subjid Subject ID

treatment Treatment group indicator

mortd Time to death (in days)

censortime Observed censoring or death time

death Death indicator (1 if died before censoring)

HFH Number of recurrent events (e.g., hospitalizations)

kccq Quality-of-life score (missing if death occurred before day 360)

Examples

```
## Not run:
SimData_per_group(
  treatment = 1, ngroup = 100,
  alpha.JFM = 0, theta.JFM = 1,
  lambda = 0.13, ann.icr = 0.32,
  censorrate = 0.2, xbase = 45, xfinal = 52.5, sd.delta.x = 20
)
head(surv_1)

## End(Not run)
```

Description

Simulates replicated trials using a hierarchical win ratio analysis framework with three prioritized outcome layers: time-to-event (e.g., death), a continuous hospitalization frequency variable, and a continuous quality-of-life score. The simulation incorporates randomization, joint frailty model parameters, censoring, and covariate progression over time. Parallel computation is supported to speed up large-scale simulations.

Usage

```
winratiosim(
  nsim,
  N,
  Randomization.ratio,
  alpha.JFM,
  theta.JFM,
  lambda_trt,
  lambda_ctl,
  ann.icr_trt,
  ann.icr_ctl,
  xbase_trt,
  xfinal_trt,
  xbase_ctl,
  xfinal_ctl,
  sd.delta.x_trt,
  sd.delta.x_ctl,
  censorrates_trt,
  censorrates_ctl,
  nc = 1
)
```

Arguments

<code>nsim</code>	Integer. Number of simulated trials to run.
<code>N</code>	Integer. Total number of subjects in a trial (across both arms).
<code>Randomization.ratio</code>	Numeric vector of length 2 indicating the treatment-to-control allocation ratio (e.g., <code>c(1, 1)</code> for 1:1 randomization).
<code>alpha.JFM</code>	Numeric. Alpha parameter for the joint frailty model.
<code>theta.JFM</code>	Numeric. Theta parameter for the joint frailty model.
<code>lambda_trt</code>	Numeric. Event rate for the treatment group.
<code>lambda_ctl</code>	Numeric. Event rate for the control group.

ann.icr_trt	Numeric. Annual intercurrent event rate for the treatment group.
ann.icr_ctl	Numeric. Annual intercurrent event rate for the control group.
xbase_trt	Numeric. Baseline value of the continuous outcome for the treatment group.
xfinal_trt	Numeric. Final value of the continuous outcome for the treatment group.
xbase_ctl	Numeric. Baseline value of the continuous outcome for the control group.
xfinal_ctl	Numeric. Final value of the continuous outcome for the control group.
sd.delta.x_trt	Numeric. Standard deviation for the change in the continuous outcome in the treatment group.
sd.delta.x_ctl	Numeric. Standard deviation for the change in the continuous outcome in the control group.
censorrates_trt	Numeric. Censoring rate for the treatment group.
censorrates_ctl	Numeric. Censoring rate for the control group.
nc	Integer. Number of CPU cores to use for parallel processing. Default is 1.

Value

A named list with the following elements:

df_FS.analysis.summary A data frame summarizing functional score win results across simulations.

df_WR.analysis.summary A data frame summarizing win ratio results across simulations.

df_sample.size.summary A data frame with the sample sizes used in each arm per simulation.

df_Total_probability A data frame showing average win/tie/loss probabilities across all simulations.

df_Total_count A data frame showing average win/tie/loss counts across all simulations.

References

Lee, S.Y. (2025). A note on the sample size formula for a win ratio endpoint. *Statistics in Medicine*.

Examples

```
## Not run:
power.design_parameters <- list(
  nsim = 10, # nsim = 10000 is the default, nsim = 10 is only for checking the code
  N = 400,
  Randomization.ratio = c(1, 1),
  alpha.JFM = 0, theta.JFM = 1,
  lambda_trt = 0.13, lambda_ctl = 0.15,
  ann.icr_trt = 0.32, ann.icr_ctl = 0.55,
  xbase_trt = 45, xfinal_trt = 45 + 7.5, sd.delta.x_trt = 20,
  xbase_ctl = 45, xfinal_ctl = 45, sd.delta.x_ctl = 20,
  censorrates_trt = 0.2, censorrates_ctl = 0.2,
  nc = 1
)
```

```

result <- do.call(winratiosim, power.design_parameters)
result$df_WR.analysis.summary

## End(Not run)

```

WR_analysis

*Perform Hierarchical Win Ratio and Functional Score Analysis***Description**

Analyzes matched treatment-control pairs across three prioritized outcome layers using a hierarchical win ratio framework. The function computes win/loss/tie counts at each layer, win ratio test statistics, and functional score statistics as described in Finkelstein and Schoenfeld (1999), Pocock et al. (2012), and Yu & Ganju (2022).

Usage

```
WR_analysis(dataset1, dataset2, dataset3)
```

Arguments

dataset1	Data frame containing matched pairs and win indicators for the first (highest priority) layer (typically time-to-event).
dataset2	Data frame for the second layer (e.g., hospitalization or recurrent event score).
dataset3	Data frame for the third layer (e.g., quality-of-life or patient-reported outcomes).

Value

A named list with the following elements:

win.losses.count.summary A data frame with counts and proportions of wins, ties, and losses by layer and in total.

sample.size.summary Sample sizes for treatment, control, and total number of pairwise comparisons.

FS.analysis.summary Test statistic, variance, z-score, and p-value for the functional score test.

WR.analysis.summary Win ratio, variance estimates, confidence intervals, and p-value for the win ratio test.

References

Finkelstein, D. M., & Schoenfeld, D. A. (1999). Combining mortality and longitudinal measures in clinical trials. *Statistics in Medicine*, 18(11), 1341–1354. Pocock, S. J., Ariti, C. A., Collier, T. J., & Wang, D. (2012). The win ratio: a new approach to the analysis of composite endpoints in clinical trials based on clinical priorities. *European Heart Journal*, 33(2), 176–182. Yu, R. X., & Ganju, J. (2022). Sample size formula for a win ratio endpoint. *Statistics in Medicine*, 41(6), 950–963.

Examples

```
## Not run:  
# Assume dataset1, dataset2, dataset3 are prepared as matched pair data sets  
result <- WR_analysis(dataset1, dataset2, dataset3)  
result$WR.analysis.summary  
  
## End(Not run)
```

Index

`binom.conf.exact`, 2

`Scoring_Conti`, 3

`Scoring_TTE`, 4

`SimData_per_group`, 5

`winratiosim`, 7

`WR_analysis`, 9