Package ‘MRTSampleSizeBinary’

March 4, 2021

Type Package

Title Sample Size Calculator for MRT with Binary Outcomes

Version 0.1.0

Author Eliot Wong-Toi, Thabat Dahdoul, Tianchen Qian

Maintainer Tianchen Qian <t.qian@uci.edu>

Description Provides a sample size calculator for micro-randomized trials (MRTs) with binary outcomes based on methodology developed in Qian et al. (2020) <doi:10.1093/biomet/asaa070>. Also provides a power calculator when the sample size is input by the user.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Suggests testthat, knitr, rmarkdown

Imports ggplot2, Matrix, stats, dplyr

Depends R (>= 2.10)

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2021-03-04 09:40:07 UTC

R topics documented:

alpha_1 ......................................................... 2
beta_1 ......................................................... 2
compute_m_sigma ........................................... 3
compute_ncp ............................................... 3
f_t_1 .......................................................... 4
g_t_1 .......................................................... 4
is_full_column_rank ........................................ 5
alpha_1  Vector that defines the success probability null curve.

**Description**

Vector that defines the success probability null curve.

**Usage**

alpha_1

**Format**

a length 2 vector

The matrix multiplication of this vector with g_t_1 defines the MEE under the null hypothesis.

beta_1  Vector that defines the MEE under the alternative hypothesis.

**Description**

Vector that defines the MEE under the alternative hypothesis.

**Usage**

beta_1

**Format**

a length 2 vector

The matrix multiplication of this vector with f_t_1 defines the MEE under the alternative hypothesis.
**compute_m_sigma**

*Computes "M" and "Sigma" matrices for the sandwich estimator of variance-covariance matrix.*

**Description**

A helper function for mrt_binary_power() and mrt_binary_ss().

**Usage**

```r
compute_m_sigma(avail_pattern, f_t, g_t, beta, alpha, p_t)
```

**Arguments**

- `avail_pattern`: A vector of length T that is the average availability at each time point
- `f_t`: Defines marginal excursion effect MEE(t) under alternative together with beta. Assumed to be matrix of size T*p.
- `g_t`: Defines success probability null curve together with alpha. Assumed to be matrix of size T*q.
- `beta`: Length p vector that defines marginal excursion effect MEE(t) under alternative together with f_t.
- `alpha`: Length q vector that defines success probability null curve together with g_t.
- `p_t`: Length T vector of randomization probabilities at each time point

**Value**

List containing two matrices. The first is the M matrix and the second is the Sigma matrix.

**Examples**

```r
compute_m_sigma(tau_t_1, f_t_1, g_t_1, beta_1, alpha_1, p_t_1)
```

---

**compute_ncp**

*Computes the non-centrality parameter for an F distributed random variable in the context of a MRT with binary outcome.*

**Description**

A helper function for mrt_binary_power() and mrt_binary_ss().

**Usage**

```r
compute_ncp(x, beta, m_matrix, sigma_matrix)
```
Arguments

- `x` Sample size
- `beta` Marginal excursion effect, assumed dimension `p`
- `m_matrix` "Bread" of sandwich estimator for variance
- `sigma_matrix` "Meat" of sandwich estimator for variance

Value

Returns non-centrality parameter for an F distributed random variable.

Examples

```
compute_ncp(300, beta_1, m_matrix_1, sigma_matrix_1)
```

---

**f_t_1**  
A matrix defining the MEE under the alternative hypothesis.

Description

A matrix defining the MEE under the alternative hypothesis.

Usage

`f_t_1`

Format

a 10 by 2 matrix

In this example it is a log-linear trend.

---

**g_t_1**  
A matrix defining the success probability null curve.

Description

A matrix defining the success probability null curve.

Usage

`g_t_1`

Format

a 10 by 2 matrix

In this example it is a log-linear trend.
**is_full_column_rank**

Check if a matrix is full column rank.

**Description**

Used in checking if $p_t f_t$ is in the linear span of $g_t$.

**Usage**

```r
is_full_column_rank(mat)
```

**Arguments**

- `mat` A matrix.

**Value**

Boolean TRUE/FALSE for if matrix is full column rank.

**Examples**

```r
is_full_column_rank(diag(4))
```

---

**max_samp**

Returns default maximum sample size to end `power_vs_n_plot()`.

**Description**

Returns default maximum sample size to end `power_vs_n_plot()`.

**Usage**

```r
max_samp(min_samp)
```

**Arguments**

- `min_samp` The starting sample size of the plot.

**Value**

A default maximum sample size to end `power_vs_n_plot()`.

**Examples**

```r
max_samp(100)
```
min_samp  

Compute minimum sample size.

Description

Returns a default minimum sample size to start power_vs_n_plot() at.

Usage

min_samp(alph, bet)

Arguments

alph  Vector to describe the MEE under the alternative.
bet   Vector to describe the MEE under the null.

Value

A default minimum sample size to start power_vs_n_plot() at.

Examples

min_samp(alpha_1, beta_1)

mrt_binary_power  Calculate power for binary outcome MRT

Description

Returns power of the hypothesis test of marginal excursion effect (see Details) given a specified sample size in the context of an MRT with binary outcomes with small sample correction using F-distribution. See the vignette for more details.

Usage

mrt_binary_power(avail_pattern, f_t, g_t, beta, alpha, p_t, gamma, n)

Arguments

avail_pattern  A vector of length m that is the average availability at each time point
f_t            Defines marginal excursion effect MEE(t) under alternative together with beta. Assumed to be matrix of size m*p.
g_t            Defines success probability null curve together with alpha. Assumed to be matrix of size m*q.
**mrt_binary_ss**

- **beta** Length p vector that defines marginal excursion effect MEE(t) under alternative together with f_t.
- **alpha** Length q vector that defines success probability null curve together with g_t.
- **p_t** Length m vector of Randomization probabilities at each time point.
- **gamma** Desired Type I error
- **n** Sample size

**Value**

Power of the test for fixed null/alternative and sample size.

**Examples**

```r
mrt_binary_power(tau_t_1, f_t_1, g_t_1, beta_1, alpha_1, p_t_1, 0.05, 100)
```

**Description**

Returns sample size needed to achieve a specified power for the hypothesis test of marginal excursion effect (see Details) in the context of an MRT with binary outcomes with small sample correction using F-distribution. See the vignette for more details.

**Usage**

```r
mrt_binary_ss(
  avail_pattern,
  f_t,
  g_t,
  beta,
  alpha,
  p_t,
  gamma,
  b,
  exact = FALSE
)
```

**Arguments**

- **avail_pattern** A vector of length m that is the average availability at each time point
- **f_t** Defines marginal excursion effect MEE(t) under alternative together with beta. Assumed to be matrix of size m*p.
- **g_t** Defines success probability null curve together with alpha. Assumed to be matrix of size m*q.
**beta** Length p vector that defines marginal excursion effect MEE(t) under alternative together with f_t.

**alpha** Length q vector that defines success probability null curve together with g_t.

**p_t** Length m vector of Randomization probabilities at each time point.

**gamma** Desired Type I error

**b** Desired Type II error

**exact** Determines if exact n or ceiling will be returned

**Details**

When the calculator finds out that a sample size less than or equal to 10 is sufficient to attain the desired power, the calculator does not output the exact sample size but produces an error message. This is because the sample size calculator is based on an asymptotic result, and in this situation the sample size result may not be as accurate. (A small sample correction is built in the calculator, but even with the correction the sample size result may still be inaccurate when it is <= 10.) In general, when the output sample size is small, one might reconsider the following: (1) whether you are correctly or conservatively guessing the average of expected availability, (2) whether the duration of study is too long, (3) whether the treatment effect is overestimated, and (4) whether the power is set too low.

**Value**

Sample size to achieve desired power.

**Examples**

```
mrt_binary_ss(tau_t_1, f_t_1, g_t_1, beta_1, alpha_1, p_t_1, 0.05, .2, FALSE)
```

---

**m_matrix_1**

An example matrix for "bread" of sandwich estimator of variance.

**Description**

An example matrix for "bread" of sandwich estimator of variance.

**Usage**

```
m_matrix_1
```

**Format**

A 2 by 2 matrix

Generated from a toy example.
**power_summary**

*Calculate sample size at a range of power levels.*

**Description**

Returns sample sizes needed to achieve a range of power levels for the hypothesis test of marginal excursion effect (see Details) in the context of an MRT with binary outcomes with small sample correction using F-distribution. See the vignette for more details.

**Usage**

```r
power_summary(
  avail_pattern,
  f_t,
  g_t,
  beta,
  alpha,
  p_t,
  gamma,
  power_levels = seq(from = 0.6, to = 0.95, by = 0.05)
)
```

**Arguments**

- `avail_pattern`: A vector of length T that is the average availability at each time point
- `f_t`: Defines marginal excursion effect MEE(t) under alternative together with beta. Assumed to be matrix of size T*p.
- `g_t`: Defines success probability null curve together with alpha. Assumed to be matrix of size T*q.
- `beta`: Length p vector that defines marginal excursion effect MEE(t) under alternative together with f_t.
- `alpha`: Length q vector that defines success probability null curve together with g_t.
- `p_t`: Length T vector of Randomization probabilities at each time point.
- `gamma`: Desired Type I error
- `power_levels`: Vector of powers to find sample size for.

**Details**

The sample size calculator is based on an asymptotic result with a small sample correction. When the calculator finds out that a sample size less than or equal to 10 is sufficient to attain the desired power, the calculator does not output the exact sample size but produces an error message, because in this situation the sample size result may not be as accurate. In general, when the output sample size is small, one might reconsider the following: (1) whether you are correctly or conservatively guessing the average of expected availability, (2) whether the duration of study is too long, (3) whether the treatment effect is overestimated, and (4) whether the power is set too low.
Value

Dataframe containing needed sample size to achieve user-specified power values.

Examples

```r
power_summary(tau_t_1, f_t_1, g_t_1,
              beta_1, alpha_1, p_t_1, 0.05)
```

`power_vs_n_plot` Returns a plot of power vs sample size in the context of a binary outcome MRT. See the vignette for more details.

Description

Returns a plot of power vs sample size in the context of a binary outcome MRT. See the vignette for more details.

Usage

```r
power_vs_n_plot(
  avail_pattern,
  f_t,
  g_t,
  beta,
  alpha,
  p_t,
  gamma,
  min_n = max(min_samp(alpha, beta), 11),
  max_n = max_samp(min_n)
)
```

Arguments

- `avail_pattern` A vector of length T that is the average availability at each time point
- `f_t` Defines marginal excursion effect MEE(t) under alternative together with beta. Assumed to be matrix of size T*p.
- `g_t` Defines success probability null curve together with alpha. Assumed to be matrix of size T*q.
- `beta` Length p vector that defines marginal excursion effect MEE(t) under alternative together with f_t.
- `alpha` Length q vector that defines success probability null curve together with g_t.
- `p_t` Length T vector of Randomization probabilities at each time point.
- `gamma` Desired Type I error
- `min_n` Minimum of range of sample sizes to plot. Should be greater than the sum of the dimensions of alpha and beta.
- `max_n` Maximum of range of sample sizes to plot. Should be greater than min_n.
**p_t_1**

**Value**
Plot of power and sample size

**Examples**

```
    power_vs_n_plot(tau_t_1, f_t_1, g_t_1, beta_1, alpha_1, 
p_t_1, 0.05, 15, 700)
```

**p_t_1**
A vector of randomization probabilities for each time point.

**Description**
A vector of randomization probabilities for each time point.

**Usage**

```
p_t_1
```

**Format**

a length T vector
Vector of randomization probabilities.

**sigma_matrix_1**
An example matrix for "meat" of sandwich estimator of variance.

**Description**
An example matrix for "meat" of sandwich estimator of variance.

**Usage**

```
sigma_matrix_1
```

**Format**

A 2 by 2 matrix
Generated from a toy example.
**tau_t_1**

Vector that holds the average availability at each time point.

**Description**
Vector that holds the average availability at each time point.

**Usage**

```
tau_t_1
```

**Format**

- vector of length T
  
  A vector of length T that is the average availability at each decision point.
Index

* datasets
  alpha_1, 2
  beta_1, 2
  f_t_1, 4
  g_t_1, 4
  m_matrix_1, 8
  p_t_1, 11
  sigma_matrix_1, 11
  tau_t_1, 12

alpha_1, 2
beta_1, 2
compute_m_sigma, 3
compute_ncp, 3
f_t_1, 4
g_t_1, 4
is_full_column_rank, 5
m_matrix_1, 8
max_samp, 5
min_samp, 6
mrt_binary_power, 6
mrt_binary_ss, 7
p_t_1, 11
power_summary, 9
power_vs_n_plot, 10
sigma_matrix_1, 11
tau_t_1, 12