Package ‘MRTSampleSizeBinary’

October 12, 2022

Type Package
Title Sample Size Calculator for MRT with Binary Outcomes
Version 0.1.1
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 7.1.2
Suggests testthat, knitr, rmarkdown
Imports ggplot2, Matrix, stats, dplyr
Depends R (>= 2.10)
VignetteBuilder knitr
NeedsCompilation no
Repository CRAN
Date/Publication 2022-05-01 14:30:16 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
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.

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**  
**Description**  
A helper function for mrt_binary_power() and mrt_binary_ss().  
**Usage**  
```
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**  
```
compute_m_sigma(tau_t_1, f_t_1, g_t_1, beta_1, alpha_1, p_t_1)
```  
---

**compute_ncp**  
**Description**  
A helper function for mrt_binary_power() and mrt_binary_ss().  
**Usage**  
```
compute_ncp(x, beta, m_matrix, sigma_matrix)
```  
---

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

**Usage**
```
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**
```
compute_m_sigma(tau_t_1, f_t_1, g_t_1, beta_1, alpha_1, p_t_1)
```

---

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

**Usage**
```
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**

```r
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**

```r
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**

```r
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
\[
\text{is_full_column_rank}(\text{mat})
\]

Arguments

\[
\text{mat} \quad \text{A matrix.}
\]

Value

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

Examples
\[
is\_full\_column\_rank(diag(4))
\]

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
\[
\text{max_samp}(\text{min_samp})
\]

Arguments

\[
\text{min_samp} \quad \text{The starting sample size of the plot.}
\]

Value

A default maximum sample size to end power_vs_n_plot().

Examples
\[
\text{max_samp}(100)
\]
**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

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

mrt_binary_ss  Calculate sample size for binary outcome MRT

Description
Returns sample size needed to achieve a specified power for the hypothesis test of marginal excursi-
on 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

mrt_binary_ss(
    avail_pattern, 
    f_t, 
    g_t, 
    beta, 
    alpha, 
    p_t, 
    gamma, 
    b, 
    exact = FALSE, 
    less_than_10_possible = 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.
\texttt{g\_t} \quad \text{Defines success probability null curve together with alpha. Assumed to be matrix of size m*q.}

\texttt{beta} \quad \text{Length p vector that defines marginal excursion effect MEE(t) under alternative together with f\_t.}

\texttt{alpha} \quad \text{Length q vector that defines success probability null curve together with g\_t.}

\texttt{p\_t} \quad \text{Length m vector of Randomization probabilities at each time point.}

\texttt{gamma} \quad \text{Desired Type I error}

\texttt{b} \quad \text{Desired Type II error}

\texttt{exact} \quad \text{Determines if exact n or ceiling will be returned}

\texttt{less\_than\_10\_possible} \quad \text{If TRUE, returns sample size (instead of error) even if the calculated sample size is <= 10. Setting to TRUE is not recommended. Defaults to FALSE.}

\textbf{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.

\textbf{Value}

Sample size to achieve desired power.

\textbf{Examples}

\begin{verbatim}
\texttt{mrt\_binary\_ss(tau\_t\_1, f\_t\_1, g\_t\_1, beta\_1, alpha\_1, p\_t\_1, 0.05, .2, FALSE)}
\end{verbatim}

\begin{verbatim}
\texttt{m\_matrix\_1} \quad \text{An example matrix for "bread" of sandwich estimator of variance.}
\end{verbatim}

\textbf{Description}

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

\textbf{Usage}

\texttt{m\_matrix\_1}
power_summary

Format

A 2 by 2 matrix
Generated from a toy example.

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)
```

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.
**p_t**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>g_t</td>
<td>Defines success probability null curve together with alpha. Assumed to be matrix of size T*q.</td>
</tr>
<tr>
<td>beta</td>
<td>Length p vector that defines marginal excursion effect MEE(t) under alternative together with f_t.</td>
</tr>
<tr>
<td>alpha</td>
<td>Length q vector that defines success probability null curve together with g_t.</td>
</tr>
<tr>
<td>p_t</td>
<td>Length T vector of Randomization probabilities at each time point.</td>
</tr>
<tr>
<td>gamma</td>
<td>Desired Type I error</td>
</tr>
<tr>
<td>min_n</td>
<td>Minimum of range of sample sizes to plot. Should be greater than the sum of the dimensions of alpha and beta.</td>
</tr>
<tr>
<td>max_n</td>
<td>Maximum of range of sample sizes to plot. Should be greater than min_n.</td>
</tr>
</tbody>
</table>

**Value**

Plot of power and sample size

**Examples**

```r
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**

```r
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, 12
  - 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, 12
tau_t_1, 12