Package ‘tipr’

November 16, 2020

Type Package
Title Tipping Point Analyses
Version 0.2.0
Author Lucy D’Agostino McGowan
Maintainer Lucy D’Agostino McGowan <lucydagostino@gmail.com>
Description The strength of evidence provided by epidemiological and observational studies is inherently limited by the potential for unmeasured confounding. We focus on three key quantities: the observed bound of the confidence interval closest to the null, a plausible residual effect size for an unmeasured continuous or binary confounder, and a realistic mean difference or prevalence difference for this hypothetical confounder. Building on the methods put forth by Lin, Psaty, & Kronmal (1998) DOI:10.2307/2533848, we can use these quantities to assess how an unmeasured confounder may tip our result to insignificance, rendering the study inconclusive.
License MIT + file LICENSE
Encoding UTF-8
LazyData true
RoxygenNote 7.1.1
Suggests testthat, broom, dplyr, MASS
Imports glue, tibble, purrr
NeedsCompilation no
Repository CRAN
Date/Publication 2020-11-16 18:50:02 UTC

R topics documented:

  tip ................................................................. 2
  tipr ............................................................... 3
  tip_with_binary ................................................ 4

Index 6
Tip a result with a continuous confounder.

Description

choose one of the following, and the other will be estimated:

- \texttt{smd}
- \texttt{outcome\_association}

Usage

\begin{verbatim}
tip(
d, smd = NULL, outcome\_association = NULL, verbose = TRUE, lb_name = "conf\_low", ub_name = "conf\_high"
)
tip\_with\_continuous(
d, smd = NULL, outcome\_association = NULL, verbose = TRUE, lb_name = "conf\_low", ub_name = "conf\_high"
)
tip\_c(
d, smd = NULL, outcome\_association = NULL, verbose = TRUE, lb_name = "conf\_low", ub_name = "conf\_high"
)
\end{verbatim}

Arguments

\begin{description}
\item[d] Data frame. A data frame with the observed effect(s). This should have at least two columns with the lower and upper confidence bounds. These columns are assumed to be called \texttt{conf\_low} and \texttt{conf\_high}. If this is not the case, the names can be changed using the \texttt{lb\_name} and \texttt{ub\_name} parameters.

\item[smd] Numeric. Estimated scaled mean difference between the unmeasured confounder in the exposed population and unexposed population
\end{description}
outcome_association
Numeric positive value. estimated association between the unmeasured confounder and the outcome

verbose
Logical. Indicates whether to print informative message. Default: TRUE

lb_name
Character. Column name of \( d \) that holds the lower confidence bound. Default: conf.low based on broom defaults.

ub_name
Character. Column name of \( d \) that holds the upper confidence bound. Default: conf.high based on broom defaults.

Value
Data frame.

Examples

```r
d <- data.frame(conf.low = 1.2, conf.high = 1.5)
## to estimate the association between an unmeasured confounder and outcome
## needed to tip analysis
tip(d, smd = -2)

## to estimate the number of unmeasured confounders specified needed to tip
## the analysis
tip(d, smd = -2, outcome_association = .99)

## Example with broom
if (requireNamespace("broom", quietly = TRUE) && requireNamespace("dplyr", quietly = TRUE)) {
  glm(am ~ mpg, data = mtcars, family = "binomial") %>%
  broom::tidy(conf.int = TRUE, exponentiate = TRUE) %>%
  dplyr::filter(term == "mpg") %>%
  tip(outcome_association = 2.5)
}
```

Description
The tipr package.

References


**Description**

Choose two of the following three to specify, and the third will be estimated:

- `exposed_p`
- `unexposed_p`
- `outcome_association`

Alternatively, specify all three and the function will return the number of unmeasured confounders specified needed to tip the analysis.

**Usage**

```r
# With a binary confounder

# For a single data frame

tip_with_binary(  
  d,  
  exposed_p = NULL,  
  unexposed_p = NULL,  
  outcome_association = NULL,  
  verbose = TRUE,  
  lb_name = "conf.low",  
  ub_name = "conf.high"
)

# Tip with a binary confounder

tip_b(  
  d,  
  exposed_p = NULL,  
  unexposed_p = NULL,  
  outcome_association = NULL,  
  verbose = TRUE,  
  lb_name = "conf.low",  
  ub_name = "conf.high"
)
```

**Arguments**

- `d`  
  Data frame. A data frame with the observed effect(s). This should have at least two columns with the lower and upper confidence bounds. These columns are assumed to be called `conf.low` and `conf.high`. If this is not the case, the names can be changed using the `lb_name` and `ub_name` parameters.

- `exposed_p`  
  Numeric between 0 and 1. Estimated prevalence of the unmeasured confounder in the exposed population
unexposed_p Numeric between 0 and 1. Estimated prevalence of the unmeasured confounder in the unexposed population
outcome_association Numeric positive value. Estimated association between the unmeasured confounder and the outcome
verbose Logical. Indicates whether to print informative message. Default: TRUE
lb_name Character. Column name of d that holds the lower confidence bound. Default: conf.low based on broom defaults.
ub_name Character. Column name of d that holds the upper confidence bound. Default: conf.high based on broom defaults.

Details
tip_b() is an alias for tip_with_binary().

Examples
d <- data.frame(conf.low = 1.2, conf.high = 1.5)
## to estimate the association between an unmeasured confounder and outcome
## needed to tip analysis
tip_with_binary(d, exposed_p = 0.5, unexposed_p = 0)

## to estimate the number of unmeasured confounders specified needed to tip
## the analysis
tip_with_binary(d, exposed_p = 0.5, unexposed_p = 0, outcome_association = 1.1)

## Example with broom
if (requireNamespace("broom", quietly = TRUE) &&
requireNamespace("dplyr", quietly = TRUE)) {
  glm(am ~ mpg, data = mtcars, family = "binomial") %>%
  broom::tidy(conf.int = TRUE, exponentiate = TRUE) %>%
  dplyr::filter(term == "mpg") %>%
  tip_with_binary(exposed_p = 1, outcome_association = 1.15)
}
Index

tip. 2

tip_b(tip_with_binary). 4
tip_b(). 5
tip_c(tip). 2
tip_with_binary. 4
tip_with_binary(). 5
tip_with_continuous(tip). 2
tipr. 3