Package 'tmsens'

August 29, 2024

Title Sensitivity Analysis Using the Trimmed Means Estimator

Version 0.3.1

Description Sensitivity analysis using the trimmed means estimator.

License GPL (>= 3)

URL https://github.com/dea-hazewinkel/tmsens,

https://dea-hazewinkel.github.io/tmsens/,

https://mrcieu.r-universe.dev/tmsens

BugReports https://github.com/dea-hazewinkel/tmsens/issues

Suggests testthat

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Audinga-Dea Hazewinkel [aut, cre]
   (<https://orcid.org/0000-0002-6923-4388>),
   Tom Palmer [aut] (<https://orcid.org/0000-0003-4655-4511>),
   Kate Tilling [aut] (<https://orcid.org/0000-0002-1010-8926>),
   Kaitlin Wade [aut] (<https://orcid.org/0000-0003-3362-6280>),
   Jack Bowden [aut] (<https://orcid.org/0000-0003-2628-3304>)

Maintainer Audinga-Dea Hazewinkel <dea.hazew@gmail.com>

Repository CRAN

Date/Publication 2024-08-29 09:30:03 UTC

Contents

---

tmsens-package   *tmsens: Sensitivity Analysis Using the Trimmed Mean Estimator*

Description

This package implements sensitivity analysis using the trimmed mean estimator.

Author(s)

Maintainer: Audinga-Dea Hazewinkel `<dea.hazew@gmail.com>` ([ORCID](#))

Authors:

- Tom Palmer `<remlapmot@hotmail.com>` ([ORCID](#))
- Kate Tilling `<kate.tilling@bristol.ac.uk>` ([ORCID](#))
- Kaitlin Wade `<kaitlin.wade@bristol.ac.uk>` ([ORCID](#))
- Jack Bowden `<j.bowden2@exeter.ac.uk>` ([ORCID](#))

See Also

Useful links:

- <https://github.com/dea-hazewinkel/tmsens>
- <https://dea-hazewinkel.github.io/tmsens/>
- <https://mrcieu.r-universe.dev/tmsens>
- Report bugs at <https://github.com/dea-hazewinkel/tmsens/issues>

---

summary.tm   *Summarizing Trimmed Means Linear Model fits:*

Description

summary method for class `"tm"`.

Usage

```
## S3 method for class 'tm'
summary(object, ...)
```

Arguments

`object`	an object of class `"tm"`
`...`	user specified arguments

Value

`summary.tm` returns an list of summary statistics of the fitted trimmed means linear model in `object`, with components

`call`	the matched call
`n`	the number of observations per treatment group
`dropout`	the proportion of dropout per treatment group
`trimfrac`	the proportion of data that was trimmed away per treatment group
`trimside`	specifies if lower or higher value trimming was performed
`n_after_trimming`	
	the number of observations per treatment group after trimming
`coefficients`	an array of coefficients with corresponding p-values and 95% confidence intervals
`Analysis_details`	
	reiterates trimming fraction and side, and, for `adjest=TRUE` specifies if the adjustment was performed on the comparator or treatment group.
`SD_outcome`	an array of the standard deviation per treatment group, for the observed outcomes and for the trimmed outcomes

See Also

`tm`. The function `coef` extracts the array of regression coefficients with corresponding p-values and 95% confidence intervals.

Examples

```
set.seed(123456)
test_dat <- as.data.frame(cbind(c(rep(0,500),rep(1,500)),
                          c(sort(rnorm(500,0,1)),sort(rnorm(500,1,1.5)))),
                          rbinom(1000,2,0.4), rnorm(1000,0,1)))
colnames(test_dat) <- c("TR", "Y", "U", "U2")
test_dat$Y[1:200] <- NA
tm_obj <- tm(formula= Y ~ TR + U + U2, GR = "TR", trF = 0.5,
             side = "LOW", n_perm = 1000, adj_est = TRUE, data = test_dat)
summary(tm_obj)
coef(tm_obj)
```

---

tm   *Fitting Trimmed Mean Linear Models:*

Description

`tm` performs a trimmed means analysis for data with a continuous outcome/response and a binary treatment/exposure variable. Outcomes are sorted and trimmed per treatment group, and a linear regression is fitted using `lm`.

Usage

```
tm(
  formula,
  GR,
  trF = NULL,
  side = c("LOW", "HIGH"),
  n_perm = 1000,
  adj_est = FALSE,
  data
)
```

Arguments

formula	an object of class [formula](#), specifying the model, of the form outcome ~ terms, where terms must include the binary treatment variable, with additional variables optional.
GR	a string denoting the name of the binary treatment variable. This function assumes the lowest value to be the comparator/reference group
trF	a number between 0 and 1, specifying the trimming fraction: the proportion of the data that is trimmed away for each treatment group. trF should be equal to or greater than the largest observed dropout proportion. If left unspecified, a default trimming fraction of 0.5 is assumed.
side	specifies if higher value trimming ("HIGH") or lower value trimming ("LOW") should be performed.
n_perm	the number of permutations performed to obtain the p-value and 95% confidence intervals for the estimates. Default is 1000.
adj_est	logical. If TRUE the adjusted trimmed means estimate is computed. The default is FALSE.
data	a data frame containing the variables in the model. data should contain at least the following: a numeric outcome variable and a binary treatment variable (numeric, character or factor).

Value

tm returns an object of class tm. The function summary is used to obtain a summary of the results. The generic accessor function coefficients extracts the regression coefficients with corresponding p-values and 95% confidence intervals.

An object of class "tm" is a list containing the following components:

call	the matched call
n	the number of observations per treatment group
dropout	the proportion of dropout per treatment group
trimfrac	the proportion of data that was trimmed away per treatment group
trimside	specifies if lower or higher value trimming was performed
n_after_trimming	
	the number of observations per treatment group after trimming

coefficients	an array of coefficients with corresponding p-values and 95% confidence intervals
Analysis_details	
	reiterates trimming fraction and side, and, for adjest=TRUE specifies if the adjustment was performed on the comparator or treatment group.
SD_outcome	an array of the standard deviation per treatment group, for the observed outcomes and for the trimmed outcomes

Details

The trimmed means estimate is subject to two assumptions: the strong MNAR assumption requires that all dropouts (unobserved outcome values) are located in the fraction of the distribution that is trimmed away; the location shift assumption requires the group variances of the full sample to be equal. The adjusted trimmed means estimator relaxes the latter, but assumes normally distributed outcomes. The adjustment is performed on the group with the smallest dropout proportion.

The p-value and 95% confidence intervals for the trimmed means estimate and the adjusted trimmed means estimate are obtained in a permutation approach.

Examples

```
set.seed(123456)
test_dat <- as.data.frame(cbind(c(rep(0, 500), rep(1, 500)),
                         c(sort(rnorm(500, 0, 1)), sort(rnorm(500, 1, 1.5)))),
                         rbinom(1000, 2, 0.4), rnorm(1000, 0, 1)))
colnames(test_dat) <- c("TR", "Y", "U", "U2")
test_dat$Y[1:200] <- NA
# Note that we usually recommend setting n_perm to a larger value, e.g., 1000
tm_obj <- tm(formula= Y ~ TR + U + U2,
             GR = "TR", trF = 0.5, side = "LOW",
             n_perm = 100, adj_est = TRUE, data = test_dat)
print(tm_obj)
summary(tm_obj)
```

---

tm_bias	*Calculating Bias For Trimmed Mean Linear Models:*

Description

tm_bias calculates the bias and the bias-adjusted estimate for a trimmed means analysis ([tm](tm)) of a given dataset, for a user-specified trimming fraction and dropout spread. tm_bias calculates, under assumption of normally distributed outcomes, the bias components resulting from violation of the location shift assumption and violation of the strong MNAR assumption.

Usage

```
tm_bias(
  formula,
  GR,
  trF,
  side = c("LOW", "HIGH"),
  spread_TG = "max_bias",
  spread_CG = "max_bias",
  data
)
```

Arguments

formula	an object of class [formula](#), specifying the model, of the form `outcome ~ terms`, where `terms` must include the binary treatment variable, with additional variables optional.
GR	a string denoting the name of the binary treatment variable. This function assumes the lowest value to be the comparator/reference group
trF	a number between 0 and 1, specifying the trimming fraction: the proportion of the data that is trimmed away for each treatment group. `trF` should be equal to or greater than the largest observed dropout proportion. If left unspecified, a default trimming fraction of 0.5 is assumed.
side	specifies if higher value trimming ("HIGH") or lower value trimming ("LOW") should be performed.
spread_TG	a number between 0 and 1, specifying the dropout spread for the treatment group. `spread_TG` should be equal to or greater than the observed dropout proportion. If left unspecified, the worst-case scenario is assumed, in which dropout is located on the side of the distribution opposite from the one that is being trimmed (spread_TG="max_bias").
spread_CG	a number between 0 and 1, specifying the dropout spread for the comparator group. `spread_CG` should be equal to or greater than the observed dropout proportion. If left unspecified, the worst-case scenario is assumed, in which dropout is located on the side of the distribution opposite from the one that is being trimmed (spread_CG="max_bias").
data	a data frame containing the variables in the model. `data` should contain at least the following: a numeric outcome variable and a binary treatment variable (numeric, character or factor).

Value

`tm_bias` returns an object of class `tm_bias`.

An object of class "`tm_bias`" is a list containing the following components:

call	the matched call
bias_components	
	an array of bias components, including location shift assumption bias (LS), Strong MNAR bias in the treatment group (TG) and the comparator group (CG)

`total_bias`	the sum of all bias components
`TM_estimate`	the trimmed means estimate of the treatment effect
`bias_adj_TM_estimate`	
	the bias adjusted trimmed means estimate
`analysis_details`	
	the user-specified trimming fraction, trimming side, and dropout spread in the treatment (TG) and comparator groups (CG)
`observed_TG_SD`	observed standard deviation of the treatment group (TG) outcome
`observed_CG_SD`	observed standard deviation of the comparator group (CG) outcome
`inferred_TG_SD`	inferred full sample standard deviation of the treatment group (TG) outcome
`max_bias_CG`	an array of bias components, total bias, the bias adjusted estimate, and inferred full sample group standard deviations, calculated under the assumption of worst-case scenario dropout, with dropout in the comparator group (CG) on the opposite side of the distribution from the one that is being trimmed
`max_bias_TG`	an array of bias components, total bias, the bias adjusted estimate, and inferred full sample group standard deviations, calculated under the assumption of worst-case scenario dropout, with dropout in the treatment group (TG) on the opposite side of the distribution from the one that is being trimmed

Details

The trimmed means estimate is subject to two assumptions: the strong MNAR assumption requires that all dropouts (unobserved outcome values) are located in the fraction of the distribution that is trimmed away; the location shift assumption requires the group variances of the full sample to be equal. The bias resulting from the violation of either assumption can be calculated under assumption of normally distributed outcomes.

Obtaining the strong MNAR assumption bias requires an additional assumption about the distribution of the dropouts: it is assumed that the dropouts are spread homogeneously across the specified dropout spread. For example, under lower value trimming (side="LOW"), and a treatment group dropout spread of 0.6 (spread_TG=0.6), any value in the bottom 60% of the treatment group outcome distribution is equally likely to be missing.

The specified dropout spread for a given treatment group has implications for the unobserved full sample variance that is inferred from the observed data. For example, for an observed dropout of 0.4 and an assumed dropout spread of 0.5, the inferred full sample variance will be larger than for an assumed dropout spread of e.g., 0.8.

In addition to calculating the bias for a user-specified dropout spread, `tm_bias` also calculates the maximal bias. For example, for lower value trimming (side="LOW"), the worst-case scenario would involve lower value dropout in the treatment group (TG) and higher value dropout in the comparator group (CG), and vice versa. Bias components are calculated for both scenarios. If the dropout spread (spread_TG, spread_CG) is left unspecified for either treatment group, the function will return only these quantities.

Examples

```
test_dat <- as.data.frame(cbind(c(rep(0, 500), rep(1, 500)),
  c(sort(rnorm(500, 0, 1)), sort(rnorm(500, 1, 1.5)))))
```

```
colnames(test_dat) <- c("TR", "Y")
test_dat$Y[which(test_dat$TR == 0)[1:150]] <- NA
test_dat$Y[which(test_dat$TR == 1)[sample(seq(1, 400), 200, replace = FALSE)]] <- NA
tm_bias_obj <- tm_bias(formula = Y ~ TR, "TR", trF = 0.5,
                         side = "LOW", spread_TG = 0.4,
                         spread_CG = 0.6, data = test_dat)
print(tm_bias_obj)
```

Index