Package ‘MatchItEXT’

October 28, 2020

Title A Supplementary Function Set to 'MatchIt'

Version 0.0.1

Description
This function set is supplementary to 'MatchIt'. Its functions conduct several computations that 'MatchIt' does not provide. It takes the 'MatchIt' result object and/or the original data to compute standardized mean differences (SMD) between groups before and after matching. It also calculates ratio of variances and ratio of residual variances. In addition, it draws SMD comparison plots and QQ plots of distance measure score to help diagnose the matching result.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports dplyr, ggplot2, ggpubr, Hmisc, MatchIt, MASS, sure, tibble, tidyr, methods, stats

Suggests knitr, qpdf, rmarkdown, spelling, testthat

Language en-US

Depends R (>= 2.10)

VignetteBuilder knitr

NeedsCompilation no

Author Danjie Zou [aut, cre] (<https://orcid.org/0000-0003-2280-0335>)

Maintainer Danjie Zou <zoudanjie7@gmail.com>

Repository CRAN

Date/Publication 2020-10-28 09:20:11 UTC

R topics documented:

  compute_res_var_ratio .................................................. 2
  compute_smd ................................................................. 3
  compute_sub_smd ............................................................ 4
compute_res_var_ratio

**compute_res_var_ratio**  Compute ratio of variances of residuals for covariates

**Description**

This function computes the ratio of variances of residuals for covariates, which was proposed by Rubin (2001). Applicable covariate types include continuous, binary and ordinal. Multinomial variables are not applicable to this function due to the absence of single residual. Usually a k-category multinomial variable will have k-1 residuals if multinomial logistic or probit regression is applied. For continuous variable, glm(family= gaussian) is used; for binary variable, glm(family= binomial(link= logit)) is used; for ordinal variable, MASS::polr(method = "logistic") is used, then single residual is obtained by using sure::resids().

**Usage**

```r
compute_res_var_ratio(
  original_data = NULL,
  mi_obj = NULL,
  type_vec = NULL,
  discard = FALSE
)
```

**Arguments**

- `original_data`: A data frame containing original data
- `mi_obj`: A matchit object derived from MatchIt package
- `type_vec`: A vector specifying covariate types, valid values: ’ordinal’, ’3’, 3; ’binary’, ’2’, 2; ’continuous’, ’1’, 1; ’excluded’, ’0’, 0, NA. The last one means not to compute the ratio for this covariate. The length of this vector should be the same as that of covariate vector used in propensity score estimation.
- `discard`: A logical value. TRUE means some observations are discarded before matching (with respect to discarded argument in matchit function), then the ratio before matching is based on the data after discard; FALSE means no observation is discarded before matching, then the ratio before matching is based on the original intact data.

**References**

compute_smd

**See Also**

parse_formula() compute_var_ratio()

**Examples**

```r
m_out <- MatchIt::matchit(treat ~ re74 + re75 + age + educ + hispan + black,
                           data = MatchIt::lalonde, method = "nearest")
# use parse_formula() to check grouping variable and covariates
parse_formula(m_out)
compute_res_var_ratio(original_data = MatchIt::lalonde, mi_obj = m_out,
                      type_vec = c(0, 1, 1, 1, 2, 2))
```

---

**compute_smd**  
*Compute standardized mean differences before and after matching*

**Description**

This function accepts a MatchIt object (i.e., the result of matchit function), and calculates standardized mean differences before and after matching. Note exact matching and subclassification are not applicable to this function. For subclassification, use compute_sub_smd() instead. In addition, SMD can be calculated on the basis of the standard deviation of original treatment group, which is the formula used in matchit function, or on the basis of the simple pooled standard deviation of original treatment and control group. The default is `sd = "pooled"`, but it can be switched to "treatment".

**Usage**

```r
compute_smd(mi_obj = NULL, sd = "pooled")
```

**Arguments**

- `mi_obj`: A matchit object derived from MatchIt pacakge
- `sd`: The standard deviation used as the denominator in the formula, either "pooled" or "treatment"

**Value**

Return a data frame containing SMD and other information

**References**

https://doi.org/10.1080/00273171.2011.568786

https://doi.org/10.18637/jss.v042.i08
compute_sub_smd

See Also

 compute_sub_smd()

Examples

# take lalonde data as an example
# run matchit() to obtain the matching result (i.e., a matchit object)
m_out <- MatchIt::matchit(treat ~ re74 + re75 + age + educ + hispan + black, data = MatchIt::lalonde, method = "nearest")
# use matching result and compute_smd() to obtain a SMD data
# frame
opt_smd <- compute_smd(m_out, sd = "treatment")

compute_sub_smd Compute standardized mean differences for subclassification result

Description

This function accepts a MatchIt object (i.e., the result of matchit function), and calculates the overall standardized mean difference after subclassification. Note only subclassification result is applicable to this function. For other matching results except for exact matching, use compute_smd() instead. In addition, SMD can be calculated on the basis of the standard deviation of original treatment group, which is the formula used in matchit function, or on the basis of the simple pooled standard deviation of original treatment and control group. The default is sd = "pooled", but it can be switched to "treatment".

Usage

compute_sub_smd(mi_obj = NULL, sd = "pooled")

Arguments

 mi_obj A matchit object derived from MatchIt package
 sd The standard deviation used as the denominator in the formula

Value

Return a scalar (the overall SMD)

References


compute_var_ratio

See Also

compute_smd()

Examples

# take lalonde data as an example
# run matchit() to obtain the matching result
m_out <- MatchIt::matchit(treat ~ re74 + re75 + age + educ + hispan +
black, data = MatchIt::lalonde, method = "subclass", subclass = 7)
compute_sub_smd(m_out, sd = "treatment")

compute_var_ratio

Compute the ratio of the variances of propensity score in the two
groups

Description

This function accepts a MatchIt object (i.e., the result of matchit function), and calculates variance of propensity score in the two groups before and after matching. The variance ratio is an indicator proposed by Rubin (2001) to assess the similarity of distributions between groups.

Usage

compute_var_ratio(mi_obj = NULL)

Arguments

mi_obj A matchit object derived from MatchIt package

Value

Return a vector of variances and variance ratios

References


See Also

compute_res_var_ratio()

Examples

m_out <- MatchIt::matchit(treat ~ re74 + re75 + age + educ + hispan +
black, data = MatchIt::lalonde, method = "nearest")
compute_var_ratio(m_out)
### parse_formula

*Parse formula to obtain grouping variable and covariate vector*

**Description**

This function parses the formula used in matchit() to obtain the grouping variable and the covariate vector.

**Usage**

```r
parse_formula(mi_obj = NULL)
```

**Arguments**

- **mi_obj**
  - A matchit object derived from MatchIt package

**Value**

Return a list including the grouping variable and covariate vector.

**Examples**

```r
m_out <- MatchIt::matchit(treat ~ re74 + re75 + age + educ + hispan + black, data = MatchIt::lalonde, method = "nearest")
parse_formula(m_out)
```

### plot_ps_qq

*Draw QQ plots of propensity score between groups before and after matching*

**Description**

This function accepts a MatchIt object (i.e., the result of matchit function) and draw side-by-side QQ plots of propensity score between groups before and after matching, for the purpose of comparison.

Note only the results of Subclassification, Nearest Neighbor Matching, Optimal Matching, Full Matching, and Genetic Matching are acceptable. The results of Exact Matching is not applicable to this function.

**Usage**

```r
plot_ps_qq(mi_obj = NULL)
```

**Arguments**

- **mi_obj**
  - A matchit object derived from MatchIt package
plot_smd

Value

Return a list of relevant data, code, and QQ plot

See Also

plot_smd()

Examples

# take lalonde data as an example
m_out <- MatchIt::matchit(treat ~ re74 + re75 + age + educ + hispan +
  black, data = MatchIt::lalonde, method = "nearest")
plot_ps_qq(m_out)

plot_smd

Draw a dot-line plot that compares the SMD change before and after matching

Description

This function accepts the data set from compute_smd() and draws a dot-line plot to compare the SMD change before and after matching. This function differs from plot(summary(data, standardize = T) in MatchIt package in the following aspects: (1) it uses the original SMD values, rather than the absolute SMD values; (2) overall SMD line is colored in blue; (3) increased SMD line is colored in brick red; (4) decreased SMD line is colored in gray. This function depends on ggplot2. If users are not satisfied with the plot, it can be revised with ggplot2. Relevant data and codes are stored in the returned list.

Usage

plot_smd(smd_data = NULL)

Arguments

smd_data a data frame derived from compute_smd

Value

Return a list of relevant data, code, and plot

See Also

plot_ps_qq()
Examples

m_out <- MatchIt::matchit(treat ~ re74 + re75 + age + educ + hispan + black, data = MatchIt::lalonde, method = "nearest")
opt_smd <- compute_smd(m_out, sd = "treatment")
plot_smd_data <- plot_smd(opt_smd)
Index

* **QQ**
  - plot_ps_qq, 6

* **SMD**
  - compute_smd, 3
  - compute_sub_smd, 4
  - plot_smd, 7

* **covariate**
  - parse_formula, 6

* **grouping**
  - parse_formula, 6

* **plot**
  - plot_ps_qq, 6
  - plot_smd, 7

* **ratio**
  - compute_res_var_ratio, 2
  - compute_var_ratio, 5

* **residual**
  - compute_res_var_ratio, 2

* **subclasscation**
  - compute_sub_smd, 4

* **variance**
  - compute_res_var_ratio, 2
  - compute_var_ratio, 5

  compute_res_var_ratio, 2
  compute_smd, 3, 7
  compute_sub_smd, 4
  compute_var_ratio, 5

  parse_formula, 6
  plot_ps_qq, 6
  plot_smd, 7

  SMD (compute_smd), 3
  sub_SMD (compute_sub_smd), 4