Package 'manymome.table'

July 22, 2025

Title Publication-Ready Tables for 'manymome' Results

Version 0.4.0

Description Converts results from the 'manymome' package, presented in Cheung and Cheung (2023) <doi:10.3758/s13428-023-02224-z>, to publication-ready tables.

URL https://sfcheung.github.io/manymome.table/

BugReports https://github.com/sfcheung/manymome.table/issues

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, tinytest, lavaan, officer

VignetteBuilder knitr

Depends R (>= 2.10)

Imports manymome, flextable

NeedsCompilation no

Author Shu Fai Cheung [aut, cre] (ORCID: <https://orcid.org/0000-0002-9871-9448>), Sing-Hang Cheung [aut] (ORCID: <https://orcid.org/0000-0001-5182-0752>)

Maintainer Shu Fai Cheung <shufai.cheung@gmail.com>

Repository CRAN

Date/Publication 2024-12-10 15:10:02 UTC

Contents

`as_flextable.cond_indirect_effects` *Convert an 'cond_indirect_effects' Object to a 'flextable' Object*

Description

The 'as_flextable' method for the output of 'manymome::many_indirect_effects()'.

Usage

```
## S3 method for class 'cond_indirect_effects'
as_flextable(
  x,
  pvalue = FALSE,
  se = TRUE,
  var_labels = NULL,
  digits = 2,
  pval_digits = 3,
  use_arrow = TRUE,
  indirect_raw = TRUE,
  indirect_raw_ci = indirect_raw,
  indirect_raw_se = indirect_raw,
  footnote = TRUE,
  show_wvalues = TRUE,
  show_indicators = FALSE,
  show_path = TRUE,
  pcut = 0.001,
  level = 0.95,
  ...
)
```

Arguments

`x`	The object to be converted. Should be of the class `cond_indirect_effects` from the package `manymome`.
`pvalue`	If bootstrap confidence intervals are stored, whether asymmetric *p*-values are reported. Default is `FALSE`. See `manymome::print.cond_indirect_effects()` for the computational details.
`se`	Whether standard errors are reported if confidence intervals are stored. Default is `TRUE`. See `manymome::print.cond_indirect_effects()` for the computation details.
`var_labels`	A named vectors. Used to replace variable names by other names when generating the table. For example, `c(x = "I.V", y = "D.V.")` replaces `x` by `"I.V"` and `y` by `"D.V."` in the output.
`digits`	The number of digits to be displayed for most numerical columns, such as effect estimates, standard errors, and confidence intervals. Default is 2.

pval_digits
: The number of digits to be displayed for the *p*-value column, if present. Default is 3.

use_arrow
: If TRUE, the default, use the arrow symbol in the paths.

indirect_raw
: If TRUE, the default, report unstandardized effects even if standardization was done.

indirect_raw_ci
: If TRUE, report the confidence intervals of unstandardized effects even if standardization was done and confidence intervals were stored. Default to be equal to indirect_raw. NOTE: Not used for now. Always FALSE.

indirect_raw_se
: If TRUE, report the standard errors of unstandardized effects even if standardization was done and confidence intervals were stored. Default to be equal to indirect_raw. NOTE: Not used for now. Always FALSE.

footnote
: If TRUE, the default, add footnote(s) regarding the results to the bottom of the table.

show_wvalues
: Whether the values of moderators will be shown. If FALSE, no values will be shown, even for categorical moderators. Default is TRUE.

show_indicators
: Whether the values of indicators (dummy variables) will be shown for categorical moderators. Default is FALSE.

show_path
: Whether the paths being moderated will be displayed. Default is TRUE.

pcut
: Any *p*-value less than pcut will be displayed as <[pcut], "[pcut]" replaced by the value of pcut. Default is .001.

level
: The level of confidence for the confidence intervals computed from the original standard errors (e.g., the standard errors in stats::lm() or lavaan). Used only for paths without mediators and both x- and y-variables are not standardized. Default is .95.

...
: Additional arguments. To be passed to flextable::autofit() in preparing the final table. For example, if some lines are too lone and wrapped, try adding add_w = .2.

Details

It converts an cond_indirect_effects object, which is usually created by manymome::cond_indirect_effects(), to a flextable object. The output can be further modified by functions from the flextable package.

Value

A flextable object.

Examples

```
library(manymome)
library(flextable)

# List of indirect effects
```

```
dat <- data_med_mod_a
lm_m <- lm(m ~ x*w + c1 + c2, dat)
lm_y <- lm(y ~ m + x + c1 + c2, dat)
fit_lm <- lm2list(lm_m, lm_y)

# Should set R to 5000 or 10000 in real research
boot_out_lm <- do_boot(fit_lm,
                       R = 100,
                       seed = 54532,
                       parallel = FALSE,
                       progress = FALSE)

out_xmy_on_w <- cond_indirect_effects(wlevels = "w",
                                      x = "x",
                                      y = "y",
                                      m = "m",
                                      fit = fit_lm,
                                      boot_ci = TRUE,
                                      boot_out = boot_out_lm)

std_xmy_on_w <- cond_indirect_effects(wlevels = "w",
                                      x = "x",
                                      y = "y",
                                      m = "m",
                                      fit = fit_lm,
                                      boot_ci = TRUE,
                                      boot_out = boot_out_lm,
                                      standardized_x = TRUE,
                                      standardized_y = TRUE)

ft1 <- as_flextable(out_xmy_on_w,
                    var_labels = c(w = "Moderator"))
ft1

ft2 <- as_flextable(std_xmy_on_w,
                    var_labels = c(w = "Moderator"),
                    se = FALSE,
                    digits = 3)
ft2
```

`as_flextable.indirect_list` *Convert an 'indirect_list' Object to a 'flextable' Object*

Description

The 'as_flextable' method for the output of 'manymome::many_indirect_effects()'.

Usage

```
## S3 method for class 'indirect_list'
as_flextable(
  x,
  pvalue = FALSE,
  se = TRUE,
  var_labels = NULL,
  digits = 2,
  pval_digits = 3,
  use_arrow = TRUE,
  indirect_raw = TRUE,
  indirect_raw_ci = indirect_raw,
  indirect_raw_se = indirect_raw,
  group_by_x = TRUE,
  group_by_y = TRUE,
  y_first = TRUE,
  total_indirect = TRUE,
  footnote = TRUE,
  pcut = 0.001,
  ...
)
```

Arguments

x	The object to be converted. Should be of the class `indirect_list` from the package `manymome`.
pvalue	If bootstrap confidence intervals are stored, whether asymmetric *p*-values are reported. Default is `FALSE`. See `manymome::print.indirect_list()` for the computational details.
se	Whether standard errors are reported if confidence intervals are stored. Default is `TRUE`. See `manymome::print.indirect_list()` for the computation details.
var_labels	A named vectors. Used to replace variable names by other names when generating the table. For example, `c(x = "I.V", y = "D.V.")` replaces `x` by `"I.V"` and `y` by `"D.V."` in the output.
digits	The number of digits to be displayed for most numerical columns, such as effect estimates, standard errors, and confidence intervals. Default is 2.
pval_digits	The number of digits to be displayed for the *p*-value column, if present. Default is 3.
use_arrow	If `TRUE`, the default, use the arrow symbol in the paths.
indirect_raw	If `TRUE`, the default, report unstandardized effects even if standardization was done.
indirect_raw_ci	If `TRUE`, report the confidence intervals of unstandardized effects even if standardization was done and confidence intervals were stored. Default to be equal to `indirect_raw`. NOTE: Not used for now. Always `FALSE`.

indirect_raw_se
: If TRUE, report the standard errors of unstandardized effects even if standardization was done and confidence intervals were stored. Default to be equal to indirect_raw. NOTE: Not used for now. Always FALSE.

group_by_x
: If TRUE, the default, the rows will be grouped by x-variables if the paths have more than one x-variable. Default is TRUE.

group_by_y
: If TRUE, the default, the rows will be grouped by y-variables if the paths have more than one y-variable. Default is TRUE.

y_first
: If group by both x- and y-variables, group by y-variables first if TRUE, the default. Otherwise, group by x-variables.

total_indirect
: If TRUE, the default, total indirect effect will be computed and added to the output.

footnote
: If TRUE, the default, add footnote(s) regarding the results to the bottom of the table.

pcut
: Any *p*-value less than pcut will be displayed as <[pcut], "[pcut]" replaced by the value of pcut. Default is .001.

...
: Additional arguments. To be passed to flextable::autofit() in preparing the final table. For example, if some lines are too lone and wrapped, try adding add_w = .2.

Details

It converts an indirect_list object, which is usually created by manymome::many_indirect_effects(), to a flextable object. The output can be further modified by functions from the package flextable.

Value

A flextable object.

Examples

```
library(flextable)
library(manymome)

data(data_med_complicated)
lm_m11 <- lm(m11 ~ x1 + x2, data_med_complicated)
lm_m2 <- lm(m2 ~ x1 + x2, data_med_complicated)
lm_y1 <- lm(y1 ~ m11 + m2 + x1 + x2, data_med_complicated)
fit <- lm2list(lm_m11, lm_m2, lm_y1)

# All indirect paths
paths <- all_indirect_paths(fit,
                            x = c("x1", "x2"),
                            y = c("y1"))

# Indirect paths from x1 to y1
paths_x1y1 <- all_indirect_paths(fit,
                            x = c("x1"),
                            y = c("y1"))
```

```
# Indirect effect estimates
ind <- many_indirect_effects(paths,
                             fit = fit)
ft_ind <- as_flextable(ind)
ft_ind
ft_ind <- as_flextable(ind, group_by_x = FALSE)
ft_ind

ind_x1y1 <- many_indirect_effects(paths_x1y1,
                                  fit = fit)
ft_ind_x1y1 <- as_flextable(ind_x1y1)
ft_ind_x1y1

# Should set R to 5000 or 10000 in real research
boot_out_lm <- do_boot(fit,
                       R = 100,
                       seed = 54532,
                       parallel = FALSE,
                       progress = FALSE)
ind_x1y1_ci <- many_indirect_effects(paths_x1y1,
                                     fit = fit,
                                     boot_ci = TRUE,
                                     boot_out = boot_out_lm)
ft_ind_x1y1_ci <- as_flextable(ind_x1y1_ci)
ft_ind_x1y1_ci
```

Index