Package ‘papaja’

October 14, 2022

Title Prepare American Psychological Association Journal Articles with R Markdown

Version 0.1.1

Description Tools to create dynamic, submission-ready manuscripts, which conform to American Psychological Association manuscript guidelines. We provide R Markdown document formats for manuscripts (PDF and Word) and revision letters (PDF). Helper functions facilitate reporting statistical analyses or create publication-ready tables and plots.

Date 2022-07-04

URL https://github.com/crsh/papaja

BugReports https://github.com/crsh/papaja/issues

Depends R (>= 3.6), tinylabels (>= 0.1.0)

Imports bookdown (>= 0.9.1), broom (>= 0.7.0), glue (>= 1.4.0), knitr (>= 1.26), methods, rmarkdown (>= 2.4), rmdfiltr (>= 0.1.3), utils, yaml, zip

Suggests afex, BayesFactor, beeswarm, boot, car, dplyr, effectsize (>= 0.4.4), emmeans, ggforce, ggplot2, latex2exp, lme4, lmerTest, MBESS, multcomp, nlme, R.rsp, skimmer, spelling, testthat

SystemRequirements Rendering the document template requires pandoc (>= 2.0; https://pandoc.org) and for PDFs a TeX distribution, such as TinyTeX (>= 0.12; https://yihui.org/tinytex/)

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.1.2

VignetteBuilder knitr, R.rsp

Language en-US

NeedsCompilation no

Author Frederik Aust [aut, cre] (<https://orcid.org/0000-0003-4900-788X>), Marius Barth [aut] (<https://orcid.org/0000-0002-3421-6665>), Birk Diedenhofen [ctb],
Christoph Stahl [ctb],
Joseph V. Casillas [ctb],
Rudolf Siegel [ctb]

Maintainer Frederik Aust <frederik.aust@uni-koeln.de>
Repository CRAN
Date/Publication 2022-07-05 00:10:02 UTC

R topics documented:

add_equals .............................................. 3
apa6_pdf .............................................. 3
apa_barplot ......................................... 5
apa_beeplot ......................................... 8
apa_df ................................................. 11
apa_factorial_plot ................................. 12
apa_interval ....................................... 16
apa_lineplot ....................................... 18
apa_num ............................................. 21
apa_p .................................................. 24
apa_prepare_doc ................................. 25
apa_print ........................................... 26
apa_print.aov ..................................... 27
apa_print.BFBayesFactor ....................... 31
apa_print.emmGrid .................................. 34
apa_print.glht ....................................... 37
apa_print.htest ..................................... 38
apa_print.list .................................... 40
apa_print.lm ......................................... 42
apa_print.lme ....................................... 45
apa_print.merMod ................................... 47
apa_print.papaja_wsci ......................... 48
apa_table ........................................... 49
beautify_terms ..................................... 53
cite_r ................................................ 54
conf_int ............................................. 55
fetch_web_refs .................................... 56
fetch_zoteroRefs .................................. 56
glue_apa_results ................................ 58
hd_int ............................................... 59
in_paren ............................................ 60
papaja ............................................... 61
quote_from_tex .................................... 62
remove_comments .................................. 62
revision_letter_pdf ......................... 63
r.refs ................................................ 63
sanitize_terms .................................... 65
se ....................................................... 66
add.equals

Add Equals Where Necessary

Description

This is an internal function that prepends every element of a character vector with an 'equals' sign if the respective element does not contain one of c("=", "<", ">").

Usage

add.equals(x)

Arguments

x A character vector.

Value

Character vector

Examples

add.equals(c("42", "<= 42", "> 42", "+42"))

apa6_pdf APA manuscript (6th edition)

Description

Template for creating an article according to APA guidelines (6th edition) in PDF or DOCX format.
Usage

\begin{verbatim}
apa6_pdf(
    fig_caption = TRUE,
    number_sections = FALSE,
    toc = FALSE,
    keep_tex = TRUE,
    md_extensions = NULL,
    includes = NULL,
    ...
)

apa6_docx(
    fig_caption = TRUE,
    number_sections = FALSE,
    md_extensions = NULL,
    ...
)

apa6_word(...)

apa6_doc(...)
\end{verbatim}

Arguments

- `fig_caption`: TRUE to render figures with captions
- `number_sections`: TRUE to number section headings
- `toc`: TRUE to include a table of contents in the output
- `keep_tex`: Keep the intermediate tex file used in the conversion to PDF
- `md_extensions`: Markdown extensions to be added or removed from the default definition of R Markdown. See the \texttt{rmarkdown_format} for additional details.
- `includes`: Named list of additional content to include within the document (typically created using the \texttt{includes} function).
- `...`: Further arguments to pass to \texttt{pdf_document2} or \texttt{word_document2}.

Details

When creating PDF documents the YAML option \texttt{classoption} is passed to the class options of the LaTeX \texttt{apa6} document class. In this case, additional options are available. Refer to the \texttt{apa6} document class documentation to find out about class options such as paper size or draft watermarks.

Please refer to the \texttt{papaja} online-manual for additional information on available YAML front matter settings. Note that the available settings for DOCX documents are more limited than for PDF documents.

When creating PDF documents the output device for figures defaults to \texttt{c("pdf", "png")}, so that each figure is saved in all four formats at a resolution of 300 dpi.
apa_barplot

Value

R Markdown output format to pass to \texttt{rmarkdown::render()}

Functions

- \texttt{apa6\_word}: Format to create .docx-files. Alias of \texttt{apa6\_docx}.
- \texttt{apa6\_doc}: Format to create .docx-files. Alias of \texttt{apa6\_docx}.

See Also

\texttt{bookdown::pdf\_document2()} \texttt{bookdown::word\_document2()}

\begin{verbatim}
apa_barplot data, ...

## Default S3 method:
apa_barplot(
data, id, factors = NULL, dv, tendency = mean, dispersion = conf_int, level = 0.95,
fun_aggregate = mean, na.rm = TRUE, use = \texttt{\textquotesingle all.obs\textquotesingle}, reference = 0,
intercept = NULL, args_x_axis = NULL, args_y_axis = NULL, args_title = NULL,
args_rect = NULL, args_error_bars = NULL, args_legend = NULL, xlab = NULL,
 ylab = NULL,

\end{verbatim}

Description

Create one or more bar plots from a \texttt{data.frame} containing data from a factorial design and set APA-friendly defaults.

Usage

\begin{verbatim}
apa_barplot(data, ...)

## Default S3 method:
apa_barplot(
data, id, factors = NULL, dv, tendency = mean, dispersion = conf_int,
level = 0.95, fun_aggregate = mean, na.rm = TRUE, use = \texttt{\textquotesingle all.obs\textquotesingle},
reference = 0, intercept = NULL, args_x_axis = NULL, args_y_axis = NULL,
args_title = NULL, args_rect = NULL, args_error_bars = NULL, args_legend = NULL,
xlab = NULL, ylab = NULL,

\end{verbatim}
apa_barplot

main = NULL,
... )

## S3 method for class 'afex_aov'
apa_barplot(
data,
tendency = mean,
dispersion = conf_int,
fun_aggregate = mean,
... )

Arguments
data A data.frame that contains the data, or an object of class afex_aov.

Arguments passed on to apa_factorial_plot

id Character. Variable name that identifies subjects.
factors Character. A vector of up to four variable names that is used to stratify the data.
dv Character. The name of the dependent variable.
tendency Closure. A function that will be used as measure of central tendency.
dispersion Closure. A function that will be used to construct error bars (i.e., whiskers).
Defaults to conf_int() for 95% between-subjects confidence intervals. See details for more options, especially for within-subjects confidence intervals.
level Numeric. Defines the width of the interval if confidence intervals are plotted. Defaults to 0.95. for 95% confidence intervals. Ignored if dispersion is not a confidence-interval function. See details.
fun_aggregate Closure. The function that will be used to aggregate observations within subjects and factors before calculating descriptive statistics for each cell of the design. Defaults to mean.
na.rm Logical. Specifies if missing values are removed. Defaults to TRUE.
use Character. Specifies a method to exclude cases if there are missing values after aggregating. Possible options are "all.obs" or "complete.obs".
reference Numeric. A reference point that determines the y coordinate of the x axis. Useful if there exists a 'nil' value; defaults to 0.
intercept Numeric. Adds a horizontal line at height intercept to the plot. Can be either a single value or a matrix. For the matrix case, multiple lines are drawn, where the dimensions of the matrix determine the number of lines to be drawn.
args_x_axis An optional list that contains further arguments that may be passed to axis() for customizing the x axis.
args_y_axis An optional list that contains further arguments that may be passed to axis() for customizing the y axis.
args_title An optional list that contains further arguments that may be passed to title().
args_rect  An optional list that contains further arguments that may be passed to `rect()`.

args_error_bars  An optional list that contains further arguments that may be passed to `arrows()`.

args_legend  An optional list that contains further arguments that may be passed to `legend()`.

xlab  Character or expression. Label for x axis.

ylab  Character or expression. Label for y axis.

main  Character or expression. For up to two factors, simply specify the main title. If you stratify the data by more than two factors, either specify a single value that will be added to automatically generated main title, or specify an array of multiple titles, one for each plot area.

Details

The measure of dispersion can be either `conf_int()` for between-subjects confidence intervals, `se()` for standard errors, or any other standard function. For within-subjects confidence intervals, specify `wsci()` or `within_subjects_conf_int()`.

If between- or within-subjects confidence intervals are requested, you can also specify the area of the cumulative distribution function that will be covered. For instance, if you want a 98% confidence interval, specify `level = 0.98`. The default is `level = 0.95` for 95% confidence intervals.

Customization of plot elements:

`apa_factorial_plot()` and its descendants `apa_barplot()`, `apa_lineplot()`, and `apa_beeplot()` are wrapper functions that sequentially call:

- `plot.new()`,
- `plot.window()`,
- `axis()` (once for x axis, once for y axis),
- `title()` for axis labels and titles,
- `rect()` for bars in bar plots,
- `points()` for bee swarms,
- `lines()` for lines connecting central tendency points,
- `arrows()` for error bars,
- `points()` for tendency points,
- `legend()` for a legend, and
- `lines()` for intercepts.

These calls can be customized by setting the respective parameters `args_*** = list(...)`.

Value

A named (nested) list of plot options including raw and derived data. *Note that the structure of the return value is about to change in a forthcoming release of papaja.*

See Also

Other plots for factorial designs: `apa_beeplot()`, `apa_factorial_plot()`, `apa_lineplot()`
Examples

```r
apa_barplot(
  data = npk
  , id = "block"
  , dv = "yield"
  , factors = c("N")
)

apa_barplot(
  data = npk
  , id = "block"
  , dv = "yield"
  , factors = c("N", "P")
)

apa_barplot(
  data = npk
  , id = "block"
  , dv = "yield"
  , factors = c("N", "P", "K")
  , ylim = c(0, 80)
  , level = .34
  , las = 1
)
```

---

**apa_beeplot**

*Bee-swarm Plots for Factorial Designs that Conform to APA Guidelines*

**Description**

Create one or more beeswarm plots from a `data.frame` containing data from a factorial design and set APA-friendly defaults.

**Usage**

```r
apa_beeplot(data, ...)```

# Default S3 method:
```r
apa_beeplot(
  data, id,
  factors = NULL, dv,
  tendency = mean,
  dispersion = conf_int,
  level = 0.95,
  ...)```

fun_aggregate = mean,
na.rm = TRUE,
use = "all.obs",
intercept = NULL,
args_x_axis = NULL,
args_y_axis = NULL,
args_title = NULL,
args_points = NULL,
args_swarm = NULL,
args_error_bars = NULL,
args_legend = NULL,
jit = 0.3,
xlab = NULL,
ylab = NULL,
main = NULL,
...)

## S3 method for class 'afex_aov'
apa_beeplot(
  data,
tendency = mean,
dispersion = conf_int,
fun_aggregate = mean,
...)

Arguments

data A data.frame that contains the data, or an object of class afex_aov.

... Arguments passed on to apa_factorial_plot

id Character. Variable name that identifies subjects.
factors Character. A vector of up to four variable names that is used to stratify the data.
dv Character. The name of the dependent variable.
tendency Closure. A function that will be used as measure of central tendency.
dispersion Closure. A function that will be used to construct error bars (i.e., whiskers). Defaults to conf_int() for 95% between-subjects confidence intervals. See details for more options, especially for within-subjects confidence intervals.
level Numeric. Defines the width of the interval if confidence intervals are plotted. Defaults to 0.95 for 95% confidence intervals. Ignored if dispersion is not a confidence-interval function. See details.
fun_aggregate Closure. The function that will be used to aggregate observations within subjects and factors before calculating descriptive statistics for each cell of the design. Defaults to mean.
na.rm Logical. Specifies if missing values are removed. Defaults to TRUE.
**apa_beeplot**

**use**  
Character. Specifies a method to exclude cases if there are missing values after aggregating. Possible options are "all.obs" or "complete.obs".

**intercept**  
Numeric. Adds a horizontal line at height intercept to the plot. Can be either a single value or a matrix. For the matrix case, multiple lines are drawn, where the dimensions of the matrix determine the number of lines to be drawn.

**args_x_axis**  
An optional list that contains further arguments that may be passed to `axis()` for customizing the x axis.

**args_y_axis**  
An optional list that contains further arguments that may be passed to `axis()` for customizing the y axis.

**args_title**  
An optional list that contains further arguments that may be passed to `title()`.

**args_points**  
An optional list that contains further arguments that may be passed to `points()`.

**args_swarm**  
An optional list that contains further arguments to customize the `points()` of the beeswarm.

**args_error_bars**  
An optional list that contains further arguments that may be passed to `arrows()`.

**args_legend**  
An optional list that contains further arguments that may be passed to `legend()`

**jit**  
Numeric. Determines the amount of horizontal displacement. Defaults to 0.3, defaults to 0.4 if `plot = "bars"`.

**xlab**  
Character or expression. Label for x axis.

**ylab**  
Character or expression. Label for y axis.

**main**  
Character or expression. For up to two factors, simply specify the main title. If you stratify the data by more than two factors, either specify a single value that will be added to automatically generated main title, or specify an array of multiple titles, one for each plot area.

**Details**

The measure of dispersion can be either `conf_int()` for between-subjects confidence intervals, `se()` for standard errors, or any other standard function. For within-subjects confidence intervals, specify `wsci()` or `within_subjects_conf_int()`.

If between- or within-subjects confidence intervals are requested, you can also specify the area of the cumulative distribution function that will be covered. For instance, if you want a 98% confidence interval, specify `level = 0.98`. The default is `level = 0.95` for 95% confidence intervals.

**Customization of plot elements:**

`apa_factorial_plot()` and its descendants `apa_barplot()`, `apa_lineplot()`, and `apa_beeplot()` are wrapper functions that sequentially call:

- `plot.new()`.
- `plot.window()`.
- `axis()` (once for x axis, once for y axis).
- `title()` for axis labels and titles.
- `rect()` for bars in bar plots.
- `points()` for bee swarms.
- `lines()` for lines connecting central tendency points.
• `arrows()` for error bars,
• `points()` for tendency points,
• `legend()` for a legend, and
• `lines()` for intercepts.

These calls can be customized by setting the respective parameters `args_*** = list(...)`. 

Value

A named (nested) list of plot options including raw and derived data. *Note that the structure of the return value is about to change in a forthcoming release of papaja.*

See Also

Other plots for factorial designs: `apa_barplot()`, `apa_factorial_plot()`, `apa_lineplot()`

Examples

```r
apa_beeplot(
  data = npk
  , id = "block"
  , dv = "yield"
  , factors = c("N")
)

apa_beeplot(
  data = npk
  , id = "block"
  , dv = "yield"
  , factors = c("N", "P")
  , args.legend = list(x = "center")
)

apa_beeplot(
  data = npk
  , id = "block"
  , dv = "yield"
  , factors = c("N", "P", "K")
  , ylim = c(0, 80)
  , level = .34
  , las = 1
)
```

TypeSet Degrees of Freedom

Description

This is a function for processing degrees of freedom. It takes care that trailing digits are only printed if non-integer values are given.
**Usage**

```
apa_df(x, digits = 2L, elementwise = TRUE)
print_df(x, digits = 2L, elementwise = TRUE)
```

**Arguments**

- **x** Numeric. The degrees of freedom to report.
- **digits** Integer. The desired number of digits after the decimal point to be used if `x` contains non-integer values.
- **elementwise** Logical. Determines whether the number of trailing digits should be determined for each element of `x` separately (the default), or for the complete vector `x`.

**Value**

An object of the same class as `x` with all numeric values converted to character.

**See Also**

`apa_num()`, `apa_p()`

**Examples**

```
apa_df(c(1, 1.23151))
```

---

**apa_factorial_plot**

*Plots for Factorial Designs that Conform to APA Guidelines*

**Description**

Create one or more plots by sequentially calling functions from the `graphics` package. `apa_factorial_plot()` is the workhorse function that is called by `apa_barplot()`, `apa_beeplot()`, and `apa_lineplot()`.

**Usage**

```
apa_factorial_plot(data, ...)
```

## Default S3 method:

```
apa_factorial_plot(  
data,  
id,  
factors = NULL,  
dv,  
tendency = mean,  
dispersion = conf_int,  
level = 0.95,  
fun_aggregate = mean,
```
na.rm = TRUE,
use = "all.obs",
reference = 0,
intercept = NULL,
args_x_axis = NULL,
args_y_axis = NULL,
args_title = NULL,
args_rect = NULL,
args_points = NULL,
args_lines = NULL,
args_swarm = NULL,
args_error_bars = NULL,
args_legend = NULL,
plot = NULL,
jit = 0.3,
xlab = NULL,
ylab = NULL,
main = NULL,
...
)

## S3 method for class 'afex_aov'
apa_factorial_plot(
da, 
tendency = mean,
dispersion = conf_int,
fun_aggregate = mean,
...
)

Arguments

data A data.frame that contains the data, or an object of class afex_aov.
...
Arguments passed on to graphics::plot.window
xlim numeric vectors of length 2, giving the x and y coordinates ranges.
ylim numeric vectors of length 2, giving the x and y coordinates ranges.
log character; indicating which axes should be in log scale.
asp numeric, giving the aspect ratio y/x, see ‘Details’.
id Character. Variable name that identifies subjects.
factors Character. A vector of up to four variable names that is used to stratify the data.
dv Character. The name of the dependent variable.
tendency Closure. A function that will be used as measure of central tendency.
dispersion Closure. A function that will be used to construct error bars (i.e., whiskers). Defaults to conf_int() for 95% between-subjects confidence intervals. See details for more options, especially for within-subjects confidence intervals.
level Numeric. Defines the width of the interval if confidence intervals are plotted. Defaults to 0.95, for 95% confidence intervals. Ignored if dispersion is not a confidence-interval function. See details.

fun_aggregate Closure. The function that will be used to aggregate observations within subjects and factors before calculating descriptive statistics for each cell of the design. Defaults to mean.

na.rm Logical. Specifies if missing values are removed. Defaults to TRUE.

use Character. Specifies a method to exclude cases if there are missing values after aggregating. Possible options are "all.obs" or "complete.obs".

reference Numeric. A reference point that determines the y coordinate of the x axis. Useful if there exists a ‘nil’ value; defaults to 0.

intercept Numeric. Adds a horizontal line at height intercept to the plot. Can be either a single value or a matrix. For the matrix case, multiple lines are drawn, where the dimensions of the matrix determine the number of lines to be drawn.

args_x_axis An optional list that contains further arguments that may be passed to axis() for customizing the x axis.

args_y_axis An optional list that contains further arguments that may be passed to axis() for customizing the y axis.

args_title An optional list that contains further arguments that may be passed to title().

args_rect An optional list that contains further arguments that may be passed to rect().

args_points An optional list that contains further arguments that may be passed to points().

args_lines An optional list that contains further arguments that may be passed to lines().

args_swarm An optional list that contains further arguments to customize the points() of the beeswarm.

args_error_bars An optional list that contains further arguments that may be passed to arrows().

args_legend An optional list that contains further arguments that may be passed to legend().

plot Character. A vector specifying which elements of the plot should be plotted. Available options are c("points", "error_bars", "bars", "swarms", "lines").

jit Numeric. Determines the amount of horizontal displacement. Defaults to 0.3, defaults to 0.4 if plot = "bars".

xlab Character or expression. Label for x axis.

ylab Character or expression. Label for y axis.

main Character or expression. For up to two factors, simply specify the main title. If you stratify the data by more than two factors, either specify a single value that will be added to automatically generated main title, or specify an array of multiple titles, one for each plot area.

Details

The measure of dispersion can be either conf_int() for between-subjects confidence intervals, se() for standard errors, or any other standard function. For within-subjects confidence intervals, specify wsci() or within_subjects_conf_int().
If between- or within-subjects confidence intervals are requested, you can also specify the area of the cumulative distribution function that will be covered. For instance, if you want a 98% confidence interval, specify `level = 0.98`. The default is `level = 0.95` for 95% confidence intervals.

**Customization of plot elements:**

`apa_factorial_plot()` and its descendants `apa_barplot()`, `apa_lineplot()`, and `apa_beeplot()` are wrapper functions that sequentially call:

- `plot.new()`,
- `plot.window()`,
- `axis()` (once for x axis, once for y axis),
- `title()` for axis labels and titles,
- `rect()` for bars in bar plots,
- `points()` for bee swarms,
- `lines()` for lines connecting central tendency points,
- `arrows()` for error bars,
- `points()` for tendency points,
- `legend()` for a legend, and
- `lines()` for intercepts.

These calls can be customized by setting the respective parameters `args_*** = list(...)`. 

**Value**

A named (nested) list of plot options including raw and derived data. *Note that the structure of the return value is about to change in a forthcoming release of papaja.*

**See Also**

Other plots for factorial designs: `apa_barplot()`, `apa_beeplot()`, `apa_lineplot()`

**Examples**

```r
apa_factorial_plot(
  data = npk,
  id = "block",
  dv = "yield",
  factors = c("N", "P", "K"),
  las = 1,
  plot = c("error_bars", "points", "swarms"),
  ylim = c(0, 100)
)
```
apa_interval

Typeset Interval Estimate

Description

Creates a character string to report interval estimates, such as frequentist confidence or Bayesian credible intervals.

Usage

apa_interval(
  x,
  ...,  
  conf.int = NULL,
  interval_type = NULL,
  enclose_math = FALSE
)

print_interval(
  x,
  ...,  
  conf.int = NULL,
  interval_type = NULL,
  enclose_math = FALSE
)

### S3 method for class 'numeric'
apa_interval(
  x,
  y = NULL,
  conf.int = NULL,
  interval_type = NULL,
  enclose_math = FALSE,
  ...
)

### S3 method for class 'matrix'
apa_interval(
  x,
  conf.int = NULL,
  interval_type = NULL,
  enclose_math = FALSE,
  ...
)

### S3 method for class 'data.frame'
apa_interval(x, ...)
apa_interval

## S3 method for class 'list'
apa_interval(x, ...)

apa_confint(x, ..., interval_type = "CI")

print_confint(x, ..., interval_type = "CI")

apa_hdint(x, ..., interval_type = "HDI")

print_hdint(x, ..., interval_type = "HDI")

Arguments

- **x**
  Numeric. A vector (of length 2, unless y is also specified) with, a two-column matrix, or a data.frame, which can coerced to a matrix.

- **...**
  Arguments passed on to `apa_num`

- **conf.int**
  Numeric. Confidence level of the interval. Ignored if level can be inferred from attributes of x, see Details.

- **interval_type**
  Character. Abbreviation indicating the type of interval estimate, e.g. CI.

- **enclose_math**
  Logical. Indicates whether the interval should be enclosed in $ (i.e., a math environment).

- **y**
  Numeric. An optional vector of the same length as x.

Details

If possible the confidence level of the interval is inferred from attributes of x. For a vector of length 2, the attribute `conf.level` is consulted; for a matrix or data.frame the column names are used, if they are of the format "2.5\n. If x is a matrix or data.frame the row names are used as names for the returned list of intervals.

Value

A single interval is returned as a character vector of length 1; multiple intervals are returned as a named list of character vectors of length 1.

See Also

`apa_num()`

Examples

apa_confint(1, 2, conf.int = 0.95)
apa_confint(c(1, 2), conf.int = 0.95)
apa_confint(matrix(c(1, 2), ncol = 2), conf.int = 0.95)
apa_confint(confint(lm(cars)))
apa_confint(confint(lm(cars)), digits = 3)
Line Plots for Factorial Designs that Conform to APA Guidelines

Description

Creates one or more line plots from a data.frame containing data from a factorial design and set APA-friendly defaults.

Usage

apa_lineplot(data, ...)

## Default S3 method:
apa_lineplot(
data,
id,
factors = NULL,
dv,
tendency = mean,
dispersion = conf_int,
level = 0.95,
fun_aggregate = mean,
na.rm = TRUE,
use = "all.obs",
intercept = NULL,
args_x_axis = NULL,
args_y_axis = NULL,
args_title = NULL,
args_points = NULL,
args_lines = NULL,
args_error_bars = NULL,
args_legend = NULL,
jit = 0.3,
xlab = NULL,
ylab = NULL,
main = NULL,
...
)

## S3 method for class 'afex_aov'
apa_lineplot(
data,
tendency = mean,
dispersion = conf_int,
fun_aggregate = mean,
...
)
Arguments

- **data**: A `data.frame` that contains the data, or an object of class `afex_aov`.
- **...**: Arguments passed on to `apa_factorial_plot`

- **id**: Character. Variable name that identifies subjects.
- **factors**: Character. A vector of up to four variable names that is used to stratify the data.
- **dv**: Character. The name of the dependent variable.
- **tendency**: Closure. A function that will be used as measure of central tendency.
- **dispersion**: Closure. A function that will be used to construct error bars (i.e., whiskers). Defaults to `conf_int()` for 95% between-subjects confidence intervals. See details for more options, especially for within-subjects confidence intervals.
- **level**: Numeric. Defines the width of the interval if confidence intervals are plotted. Defaults to 0.95 for 95% confidence intervals. Ignored if `dispersion` is not a confidence-interval function. See details.
- **fun_aggregate**: Closure. The function that will be used to aggregate observations within subjects and factors before calculating descriptive statistics for each cell of the design. Defaults to `mean`.
- **na.rm**: Logical. Specifies if missing values are removed.Defaults to `TRUE`.
- **use**: Character. Specifies a method to exclude cases if there are missing values after aggregating. Possible options are "all.obs" or "complete.obs".
- **intercept**: Numeric. Adds a horizontal line at height `intercept` to the plot. Can be either a single value or a matrix. For the matrix case, multiple lines are drawn, where the dimensions of the matrix determine the number of lines to be drawn.
- **args_x_axis**: An optional list that contains further arguments that may be passed to `axis()` for customizing the x axis.
- **args_y_axis**: An optional list that contains further arguments that may be passed to `axis()` for customizing the y axis.
- **args_title**: An optional list that contains further arguments that may be passed to `title()`.
- **args_points**: An optional list that contains further arguments that may be passed to `points()`.
- **args_lines**: An optional list that contains further arguments that may be passed to `lines()`.
- **args_error_bars**: An optional list that contains further arguments that may be passed to `arrows()`.
- **args_legend**: An optional list that contains further arguments that may be passed to `legend()`.
- **jit**: Numeric. Determines the amount of horizontal displacement. Defaults to 0.3, defaults to 0.4 if `plot` = "bars".
- **xlab**: Character or expression. Label for x axis.
- **ylab**: Character or expression. Label for y axis.
- **main**: Character or expression. For up to two factors, simply specify the main title. If you stratify the data by more than two factors, either specify a single value that will be added to automatically generated main title, or specify an array of multiple titles, one for each plot area.
Details

The measure of dispersion can be either `conf_int()` for between-subjects confidence intervals, `se()` for standard errors, or any other standard function. For within-subjects confidence intervals, specify `wsci()` or `within_subjects_conf_int()`.

If between- or within-subjects confidence intervals are requested, you can also specify the area of the cumulative distribution function that will be covered. For instance, if you want a 98% confidence interval, specify `level = 0.98`. The default is `level = 0.95` for 95% confidence intervals.

Customization of plot elements:
`apa_factorial_plot()` and its descendants `apa_barplot()`, `apa_lineplot()`, and `apa_beeplot()` are wrapper functions that sequentially call:
- `plot.new()`,
- `plot.window()`,
- `axis()` (once for x axis, once for y axis),
- `title()` for axis labels and titles,
- `rect()` for bars in bar plots,
- `points()` for bee swarms,
- `lines()` for lines connecting central tendency points,
- `arrows()` for error bars,
- `points()` for tendency points,
- `legend()` for a legend, and
- `lines()` for intercepts.

These calls can be customized by setting the respective parameters `args_*** = list(...)`.

Value

A named (nested) list of plot options including raw and derived data. Note that the structure of the return value is about to change in a forthcoming release of papaja.

See Also

Other plots for factorial designs: `apa_barplot()`, `apa_beeplot()`, `apa_factorial_plot()`

Examples

```r
apa_lineplot(
  data = npk,
  id = "block",
  dv = "yield",
  factors = c("N")
)

apa_lineplot(
  data = npk,
  id = "block",
  dv = "yield",
  factors = c("N", "P")
)
apa_num

Typeset Numerical Values for Printing and Reporting

Description

Converts numerical values to character strings for printing and reporting.

Usage

apa_num(x, ...)

printnum(x, ...)

print_num(x, ...)

## Default S3 method:
apa_num(x, na_string = getOption("papaja.na_string"), ...)

## S3 method for class 'list'
apa_num(x, ...)

## S3 method for class 'integer'
apa_num(
  x,
  numerals = TRUE,
  capitalize = FALSE,
  zero_string = "no",
  na_string = getOption("papaja.na_string"),
  ...
)

## S3 method for class 'numeric'
apa_num(

x,
gt1 = TRUE,
zero = TRUE,
na_string = getOption("papaja.na_string"),
use_math = TRUE,
add.equals = FALSE,

)

## S3 method for class 'data.frame'
apa_num(x, margin = 2, ...)

## S3 method for class 'matrix'
apa_num(x, margin = 2, ...)

## S3 method for class 'tiny_labelled'
apa_num(x, ...)

Arguments

x
Can be either a single value, vector, matrix, data.frame.

... Arguments passed on to base::formatC
digits the desired number of digits after the decimal point (format = "f") or significant digits (format = "g", = "e" or = "fg").
Default: 2 for integer, 4 for real numbers. If less than 0, the C default of 6 digits is used. If specified as more than 50, 50 will be used with a warning unless format = "f" where it is limited to typically 324. (Not more than 15–21 digits need be accurate, depending on the OS and compiler used. This limit is just a precaution against segfaults in the underlying C runtime.)
width the total field width; if both digits and width are unspecified, width defaults to 1, otherwise to digits + 1. width = 0 will use width = digits, width < 0 means left justify the number in this field (equivalent to flag = "-"). If necessary, the result will have more characters than width. For character data this is interpreted in characters (not bytes nor display width).
format equal to "d" (for integers), "f", "e", "E", "g", "G", "fg" (for reals), or "s" (for strings). Default is "d" for integers, "g" for reals.
"f" gives numbers in the usual xxx.xxx format; "e" and "E" give n.ddde+nn or n.dddE+nn (scientific format); "g" and "G" put x[i] into scientific format only if it saves space to do so and drop trailing zeros and decimal point - unless flag contains "#" which keeps trailing zeros for the "g", "G" formats.
"fg" (our own hybrid format) uses fixed format as "f", but digits as the minimum number of significant digits. This can lead to quite long result strings, see examples below. Note that unlike signif this prints large numbers with more significant digits than digits. Trailing zeros are dropped in this format, unless flag contains ",#".
flag for formatC, a character string giving a format modifier as in Kernighan and Ritchie (1988, page 243) or the C+99 standard.
"0" pads leading zeros;
"-" does left adjustment,
"+" ensures a sign in all cases, i.e., "+" for positive numbers,
" " if the first character is not a sign, the space character " " will be used instead.
"#" specifies "an alternative output form", specifically depending on format.
"'" on some platform–locale combination, activates "thousands' grouping" for decimal conversion,
"I" in some versions of 'glibc' allow for integer conversion to use the locale's alternative output digits, if any.
There can be more than one of these flags, in any order. Other characters used to have no effect for character formatting, but signal an error since R 3.4.0.

gmode "double" (or "real"), "integer" or "character". Default: Determined from the storage mode of x.
big.mark character; if not empty used as mark between every big.interval decimals before (hence big) the decimal point.
big.interval see big.mark above; defaults to 3.
small.mark character; if not empty used as mark between every small.interval decimals after (hence small) the decimal point.
small.interval see small.mark above; defaults to 5.
decimal.mark the character to be used to indicate the numeric decimal point.
preserve.width string specifying if the string widths should be preserved where possible in those cases where marks (big.mark or small.mark) are added. "common", the default, corresponds to format-like behavior whereas "individual" is the default in formatC(). Value can be abbreviated.
zero.print logical, character string or NULL specifying if and how zeros should be formatted specially. Useful for pretty printing 'sparse' objects.
replace.zero logical; if zero.print is a character string, indicates if the exact zero entries in x should be simply replaced by zero.print. Otherwise, depending on the widths of the respective strings, the (formatted) zeroes are partly replaced by zero.print and then padded with " " to the right were applicable. In that case (false replace[,zero]), if the zero.print string does not fit, a warning is produced (if warn.non.fitting is true). This works via prettyNum(), which calls .format.zeros(*, replace=replace.zero) three times in this case, see the 'Details'.
drop0trailing logical, indicating if trailing zeros, i.e., "0" after the decimal mark, should be removed; also drops "e+00" in exponential formats. This is simply passed to prettyNum(), see the 'Details'.

na_string Character. String to print if any element of x is NA.
numerals Logical. Indicates if integers should be converted to words.
capitalize Logical. Indicates if first letter should be capitalized. Ignored if numerals = TRUE.
zero_string Character. Word to print if x is a zero integer.
gt1 Logical. Indicates if the statistic can, in principle, have an absolute value greater than 1. If FALSE, leading zeros are omitted.

zero Logical. Indicates if the statistic can, in principle, be 0. If FALSE, a string of the form < 0.001 is returned instead of 0.

use_math Logical. Indicates whether to use $ in the output so that Inf or scientific notation is rendered correctly.

add_equals Logical. Indicates if the output string should be prepended with an =.

margin Integer. If x is a matrix or data.frame, the function is applied either across rows (margin = 1) or columns (margin = 2). See apply().

Details

If x is a vector, all arguments can be vectors according to which each element of the vector is formatted. Parameters are recycled if length of x exceeds the length of the parameter vectors. If x is a matrix or data.frame, the vectors specify the formatting of either rows or columns according to the value of margin.

We recommend to use apa_num(), rather than printnum() or print_num(), which are aliases kept only for backward compatibility.

Value

An object of the same class as x with all numeric values converted to character.

See Also

apa_p(), apa_df()

Examples

apa_num(1/3)
apa_num(1/3, gt1 = FALSE)
apa_num(1/3, digits = 5)

apa_num(0)
apa_num(0, zero = FALSE)

---

**Description**

Convenience wrapper for apa_num to print p values.
Usage

apa_p(x, digits = 3L, na_string = "", add_equals = FALSE)

drinn(x, digits = 3L, na_string = "", add_equals = FALSE)

print_p(x, digits = 3L, na_string = "", add_equals = FALSE)

Arguments

x Numeric. The p value(s) to report.

digits Integer. The desired number of digits after the decimal point, passed on to formatC.

na_string Character. String to print if any element of x is NA.

add_equals Logical. Indicates if the output string should be prepended with an =.

Value

An object of the same class as x with all numeric values converted to character.

See Also

apa_num(), apa_df()

Examples

apa_p(0.05)
apa_p(0.0005)
apa_p(0.99999999)
apa_p(c(.001, 0), add_equals = TRUE)

Prepare APA document (deprecated)

Description

Prepares objects to be used in the rendering process and creates title page and abstract for MS Word documents. This function has been defunct. It is no longer needed.

Usage

apa_prepare_doc()
Details

The function creates and locks a non-exported object `apa_lang` that is used by other `papaja`-functions. `apa_lang` is a list containing localizations for document elements such as abstract and title. The selected language is defined by the `lang`-parameter in the documents yaml-header. Currently, English (default) and German ("german") are supported.

If the output document is MS Word (output: \link{apa6_word}) the function additionally creates a title page and adds the abstract. You should, therefore, always call `apa_prepare_doc` at the very beginning of the R Markdown document.

Value

Returns NULL invisibly.

See Also

`apa6_docx()`

---

Description

A generic function that takes objects from various statistical methods to create formatted character strings to report the results in accordance with APA manuscript guidelines. The function invokes particular methods, which depend on the `class` of the first argument.

Usage

`apa_print(x, ...)`

Arguments

- `x` A model object.
- `...` Additional arguments passed to methods.

Value

`apa_print()`-methods return a named list of class `apa_results` containing the following elements:

- `estimate` One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.
- `statistic` One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.
full_result  One or more character strings comprised 'estimate' and 'statistic'. A single string is returned in a vector; multiple strings are returned as a named list.

table  A data.frame of class apa_results_table that contains all elements of estimate and statistics. This table can be passed to apa_table() for reporting.

Column names in apa_results_table are standardized following the broom glossary (e.g., term, estimate conf.int, statistic, df, df.residual, p.value). Additionally, each column is labelled (e.g., $\hat{\eta}^2_G$ or $t$) using the tinylabels package and these labels are used as column names when an apa_results_table is passed to apa_table().

See Also

Other apa_print: apa_print.BFBayesFactor(), apa_print.aov(), apa_print.emmGrid(), apa_print.glht(), apa_print.htest(), apa_print.list(), apa_print.lme(), apa_print.lm(), apa_print.merMod()

Examples

# List methods for apa_print()
methods("apa_print")

# S3 method for class 'anova'
apa_print(aov, x, estimate = getOption("papaja.estimate_anova", "ges"), observed = NULL, intercept = FALSE, mse = TRUE, in_paren = FALSE, ...)

# S3 method for class 'summary.aov'
apa_print(x, estimate = getOption("papaja.estimate_anova", "ges"), observed = NULL,
intercept = FALSE,
mse = TRUE,
in_paren = FALSE,
...

## S3 method for class 'aovlist'
apa_print(
  x,
  estimate = getOption("papaja.estimate_anova", "ges"),
  observed = NULL,
  intercept = FALSE,
  mse = TRUE,
  in_paren = FALSE,
  ...
)

## S3 method for class 'summary.aovlist'
apa_print(
  x,
  estimate = getOption("papaja.estimate_anova", "ges"),
  observed = NULL,
  intercept = FALSE,
  mse = TRUE,
  in_paren = FALSE,
  ...
)

## S3 method for class 'Anova.mlm'
apa_print(
  x,
  estimate = getOption("papaja.estimate_anova", "ges"),
  observed = NULL,
  correction = getOption("papaja.sphericity_correction"),
  intercept = FALSE,
  mse = TRUE,
  in_paren = FALSE,
  ...
)

## S3 method for class 'summary.Anova.mlm'
apa_print(
  x,
  estimate = getOption("papaja.estimate_anova", "ges"),
  observed = NULL,
  correction = getOption("papaja.sphericity_correction"),
  intercept = FALSE,
  mse = TRUE,
Arguments

x An object containing the results from an analysis of variance ANOVA
estimate Character, function, or data frame. Determines which estimate of effect size is to be used. See details.
observed Character. The names of the factors that are observed, i.e., not manipulated. Necessary only for calculating generalized eta squared; otherwise ignored. If x is of class afex_aov, observed is automatically deduced from x.
intercept Logical. Indicates if the intercept term should be included in output.
mse Logical. Indicates if mean squared errors should be included in output. The default is TRUE, but this can be changed either by supplying a different value in the function call or by changing the global default via options(papaja.mse = FALSE).
in_paren Logical. Whether the formatted string is to be reported in parentheses. If TRUE,
parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.

Further arguments that may be passed to `apa_num` to format estimates (i.e., columns `estimate` and `conf.int`).

correction Character. For repeated-measures ANOVA, the type of sphericity correction to be used. Possible values are "GG" for the Greenhouse-Geisser method (the default), "HF" for the Huyn-Feldt method, or "none" for no correction.

test Character. For MANOVA, the multivariate test statistic to be reported, see `summary.manova`.

Details

The factor names are sanitized to facilitate their use as list names (see Value section). Parentheses are omitted and other non-word characters are replaced by `_`.

Argument `estimate` determines which measure of effect size is to be used: It is currently possible to provide one of three characters to specify the to-be-calculated effect size: "ges" for generalized \( \eta^2 \), "pes" for partial \( \eta^2 \), and "es" for \( \eta^2 \). Note that \( \eta^2 \) is calculated correctly if and only if the design is balanced.

It is also possible to provide a `data.frame` with columns `estimate`, `conf.low`, and `conf.high`, which allows for including custom effect-size measures.

A third option is to provide a function from the `effectsize` package that will be used to calculate effect-size measures from `x`. If `effectsize` is installed (and `papaja` is loaded), this is the new default. This default can be changed via `options(papaja.estimate_anova = ...)`.

Value

`apa_print()` methods return a named list of class `apa_results` containing the following elements:

- `estimate`: One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is `NULL`.

- `statistic`: One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is `NULL`.

- `full_result`: One or more character strings comprised ‘estimate’ and ‘statistic’. A single string is returned in a vector; multiple strings are returned as a named list.

- `table`: A `data.frame` of class `apa_results_table` that contains all elements of `estimate` and `statistic`. This table can be passed to `apa_table()` for reporting.

Column names in `apa_results_table` are standardized following the `broom` glossary (e.g., term, `estimate conf.int statistic df df.residual p.value`). Additionally, each column is labelled (e.g., \( \hat{\eta}^2_G \) or \( \text{t} \)) using the `tinylabels` package and these labels are used as column names when an `apa_results_table` is passed to `apa_table()`.

References

See Also

aov(), car::Anova(), apa_print.list()

Other apa_print: apa_print.BFBayesFactor(), apa_print.emmGrid(), apa_print.glht(), apa_print.htest(), apa_print.list(), apa_print.lme(), apa_print.lm(), apa_print.merMod(), apa_print()

Examples

npk_aov <- aov(yield ~ block + N * P * K, npk)
apa_print(npk_aov)

# Use the effectsize package to calculate partial eta-squared with
# confidence intervals
apa_print(npk_aov, estimate = effectsize::omega_squared)

Description

These methods take result objects from the BayesFactor package to create formatted character strings to report the results in accordance with APA manuscript guidelines.

Usage

## S3 method for class 'BFBayesFactor'
apa_print(
  x,
  stat_name = NULL,
  est_name = NULL,
  subscript = NULL,
  escape_subscript = FALSE,
  scientific_threshold = NULL,
  reciprocal = FALSE,
  log = FALSE,
  mcmc_error = any(x@bayesFactor$error > 0.05),
  iterations = 10000,
  standardized = FALSE,
  central_tendency = median,
  interval = hd_int,
  interval_type = "HDI",
  bf_r1 = NULL,
  bf_1r = NULL,
  ...
)
## S3 method for class 'BFBayesFactorTop'

`apa_print(x, reciprocal = FALSE, ...)`

**Arguments**

- `x`: Output object. See details.
- `stat_name`: Character. If `NULL` (the default), the name given in `x` is used for the *test statistic*, otherwise the supplied name is used. See details.
- `est_name`: Character. If `NULL` (the default), the name given in `x` (or a formally correct adaptation) is used for the *estimate*, otherwise the supplied name is used. See details.
- `subscript`: Character. Index used to specify the model comparison for the Bayes factors, e.g., `"+0"` yields $BF_{+0}$. If `NULL` default to "10".
- `escape_subscript`: Logical. If `TRUE` special LaTeX characters, such as `%` or `_`, in subscript are escaped.
- `scientific_threshold`: Numeric. Named vector of length 2 taking the form `c(min = 1/10, max = 1e6)`. Bayes factors that exceed these thresholds will be printed in scientific notation.
- `reciprocal`: Logical. If `TRUE` the reciprocal of all Bayes factors is taken before results are formatted. The advantage over specifying `x = t(x)` is that the default (only the default) index specifying the model comparison is automatically reversed, see subscript.
- `log`: Logical. If `TRUE` the logarithm of the Bayes factor is reported.
- `mcmc_error`: Logical. If `TRUE` estimation error of the Bayes factor(s) is reported.
- `iterations`: Numeric. Number of iterations of the MCMC sampler to estimate HDIs from the posterior.
- `standardized`: Logical. Whether to return standardized or unstandardized effect size estimates.
- `central_tendency`: Function to calculate central tendency of MCMC samples to obtain a point estimate from the posterior.
- `interval`: Function to calculate an interval estimate of MCMC samples from the posterior. The returned object must be either a named vector or matrix with (column) names giving the interval bounds (e.g., `2.5%` and `97.5%`) or with an attribute `conf.level` (e.g., `0.95`).
- `interval_type`: Character. Used to specify the type of interval in the formatted text.
- `bf_r1`: Numeric. Vector of the same length as `x` giving Bayes factors in favor of an order constraint relative to the unconstrained model (see details). Must be on log-scale if `log = TRUE`.
- `bf_1r`: Numeric. Same as `bf_r1` (see details).
- `...`: Arguments passed on to `apa_num.numeric`
- `gt1`: Logical. Indicates if the statistic can, in principle, have an absolute value greater than 1. If `FALSE`, leading zeros are omitted.
apa_print.BFBayesFactor

zero Logical. Indicates if the statistic can, in principle, be 0. If FALSE, a string of the form < 0.001 is returned instead of 0.

na_string Character. String to print if any element of x is NA.

use_math Logical. Indicates whether to use $ in the output so that Inf or scientific notation is rendered correctly.

add.equals Logical. Indicates if the output string should be prepended with an =.

Details

stat_name and est_name are placed in the output string and are thus passed to pandoc or LaTeX through knitr. To the extent it is supported by the final document type, you can pass LaTeX-markup to format the final text (e.g., \text{M}_\Delta yields $M_\Delta$).

For models with order constraint, the evidence for the order constraint relative to the null model can be obtained by multiplying the Bayes factor \text{BF}_{r1} for the order constraint relative to the unconstrained model (bf_r1) with the Bayes factor \text{BF}_{10} for the unconstrained model relative to the null model,

$$\frac{p(y \mid \mathcal{M}_r)}{p(y \mid \mathcal{M}_0)} = \frac{p(y \mid \mathcal{M}_r)}{p(y \mid \mathcal{M}_1)} \times \frac{p(y \mid \mathcal{M}_1)}{p(y \mid \mathcal{M}_0)}$$

\text{BF}_{r1} can be calculated from the prior and posterior odds of the order constraint (e.g., Morey & Wagenmakers, 2014). If bf_r1 (or bf_1r) is specified they are multiplied with the corresponding Bayes factor supplied in x before the reciprocal is taken and the results are formatted. Note, that it is not possible to determine whether x gives \text{BF}_{10} or \text{BF}_{01} and, hence, bf_r1 and bf_1r are treated identically; the different argument names only serve to ensure the expressiveness of the code. It is the user’s responsibility to ensure that the supplied Bayes factor is correct!

Value

apa_print()::methods return a named list of class apa_results containing the following elements:

estimate One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.

statistic One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.

full_result One or more character strings comprised ‘estimate’ and ‘statistic’. A single string is returned in a vector; multiple strings are returned as a named list.

table A data.frame of class apa_results_table that contains all elements of estimate and statistics. This table can be passed to apa_table() for reporting.

Column names in apa_results_table are standardized following the broom glossary (e.g., term, estimate conf.int, statistic, df, df.residual, p.value). Additionally, each column is labelled (e.g., $\hat{\eta}^2_G$ or $t$) using the tinylabels package and these labels are used as column names when an apa_results_table is passed to apa_table().
References


See Also

Other `apa_print`: `apa_print.aov()`, `apa_print.emmGrid()`, `apa_print.glht()`, `apa_print.htest()`, `apa_print.list()`, `apa_print.lme()`, `apa_print.lm()`, `apa_print.merMod()`, `apa_print()`

Examples

```r
# ANOVA

data(sleep, package = "BayesFactor")
bayesian_anova <- BayesFactor::anovaBF(
  extra ~ group + ID
  , data = sleep
  , whichRandom = "ID"
  , progress = FALSE
)

# Paired t-test
ttest_paired <- BayesFactor::ttestBF(
  x = sleep$extra[sleep$group == 1]
  , y = sleep$extra[sleep$group == 2]
  , paired = TRUE
)

# Results for paired t-tests are indistinguishable
# from one-sample t-tests. We therefore specify the
# appropriate 'est_name' manually.
apa_print(ttest_paired,
  est_name = "M_D"
  , iterations = 1000)

apa_print(ttest_paired
  , iterations = 1000
  , interval = function(x) quantile(x, probs = c(0.025, 0.975))
  , interval_type = "CrI"
)
```
Description

Takes various `emmeans` objects to create formatted character strings to report the results in accordance with APA manuscript guidelines. `emmeans` supports a wide range of analyses, not all of which are currently (fully) supported. Proceed with caution.

Usage

```r
## S3 method for class 'emmGrid'
apa_print(x, infer = TRUE, conf.int = 0.95, ...)

## S3 method for class 'summary_emm'
apa_print(
  x,
  contrast_names = NULL,
  est_name = "\hat{\theta}",
  in_paren = FALSE,
  ...
)

## S3 method for class 'lsmobj'
apa_print(x, ...)

## S3 method for class 'summary.ref.grid'
apa_print(x, ...)
```

Arguments

- **x**: Object
- **infer**: A vector of one or two logical values. The first determines whether confidence intervals are displayed, and the second determines whether t-tests and P-values are displayed. If only one value is provided, it is used for both.
- **conf.int**: Numeric. Confidence level for confidence intervals.
- **contrast_names**: Character. An optional vector of names to label the calculated contrasts.
- **est_name**: Character. If NULL (default) the name of the estimate is inferred from the function call of the model object supplied to `emmeans`.
- **in_paren**: Logical. Whether the formatted string is to be reported in parentheses. If TRUE, parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.

Details

When p-values and confidence intervals are adjusted for multiple testing, the correction method is added as an index to the output (e.g., `p_{(Tukey(3))}`). Values in parenthesis indicate the size of the family of tests or the rank of the set of linear functions (for the Scheffé method).
If possible, each family of tests is additionally marked in the returned table by alphabetic superscripts.

Generally, the `summary_emm` objects returned by `emmeans::summary_emm` omit information that may be needed to add some of the information on the adjustments made to p-values and confidence intervals. It is therefore preferable to pass `emmGrid`-objects if possible. For example, by using `emmeans(object, 1 ~ x1, adjust = "scheffe")`.

### Value

`apa_print()`-methods return a named list of class `apa_results` containing the following elements:

- **estimate**: One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is `NULL`.

- **statistic**: One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is `NULL`.

- **full_result**: One or more character strings comprised ‘estimate’ and ‘statistic’. A single string is returned in a vector; multiple strings are returned as a named list.

- **table**: A `data.frame` of class `apa_results_table` that contains all elements of `estimate` and `statistics`. This table can be passed to `apa_table()` for reporting.

Column names in `apa_results_table` are standardized following the `broom` glossary (e.g., term, estimate conf.int. statistic, df, df.residual, p.value). Additionally, each column is labelled (e.g., $\hat{\eta}_G^2$ or $t$) using the `tinylabels` package and these labels are used as column names when an `apa_results_table` is passed to `apa_table()`.

### See Also

Other `apa_print`: `apa_print.BFBayesFactor()`, `apa_print.aov()`, `apa_print.glht()`, `apa_print.htest()`, `apa_print.list()`, `apa_print.lme()`, `apa_print.lm()`, `apa_print.merMod()`, `apa_print()`

### Examples

```r
# From the emmeans manual:
library(emmeans)
warp.lm <- lm(breaks ~ wool*tension, data = warpbreaks)
warp.emm <- emmeans(warp.lm, ~ tension | wool)
warp.contr <- contrast(warp.emm, "poly")
apa_print(warp.contr)

# In this example, because degrees of freedom are equal across all rows
# of the output, it is possible to move that information to the variable
# labels. This is useful if a compact results table is required:

df_into_label(apa_print(warp.contr))
```
Description

*These methods are not properly tested and should be considered experimental.*

Usage

```r
## S3 method for class 'glht'
apa_print(x, test = multcomp::adjusted(), ...)
```

```r
## S3 method for class 'summary.glht'
apa_print(x, conf.int = 0.95, in_paren = FALSE, ...)
```

Arguments

- `x` Object
- `test` Function. Computes p-values (adjusted for multiple comparisons).
- `...` Further arguments to pass to `apa_num` to format the estimate.
- `conf.int` Numeric. If `NULL` (default) the function tries to obtain confidence intervals from `x`. Other confidence intervals can be supplied as a vector of length 2 (lower and upper boundary, respectively) with attribute `conf.level`, e.g., when calculating bootstrapped confidence intervals.
- `in_paren` Logical. Whether the formatted string is to be reported in parentheses. If `TRUE`, parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.

Value

`apa_print()`-methods return a named list of class `apa_results` containing the following elements:

- `estimate` One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is `NULL`.
- `statistic` One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is `NULL`.
- `full_result` One or more character strings comprised ‘estimate’ and ‘statistic’. A single string is returned in a vector; multiple strings are returned as a named list.
- `table` A `data.frame` of class `apa_results_table` that contains all elements of `estimate` and `statistics`. This table can be passed to `apa_table()` for reporting.

Column names in `apa_results_table` are standardized following the `broom` glossary (e.g., `term`, `estimate conf.int`, `statistic df`, `df.residual`, `p.value`). Additionally, each column is labelled (e.g., `$\hat{\eta}^2_G$` or `$t$`) using the `tinylabels` package and these labels are used as column names when an `apa_results_table` is passed to `apa_table()`.
See Also
Other apa_print: apa_print.BFBayesFactor(), apa_print.aov(), apa_print.emmGrid(), apa_print.h.test(), apa_print.list(), apa_print.lme(), apa_print.lm(), apa_print.merMod(), apa_print()

Examples

# From the multcomp::glht() examples:
library(multcomp)
amod <- aov(breaks ~ tension, data = warpbreaks)
glht_out <- glht(amod, linfct = mcp(tension = "Tukey"))
apa_print(glht_out)

# In this example, because degrees of freedom are equal across all rows
# of the output, it is possible to move that information to the variable
# labels. This is useful if a compact results table is required:
df_into_label(apa_print(glht_out))

---

apa_print.h.test Typeset Statistical Results from Hypothesis Tests

Description

Takes htest objects from various statistical methods (e.g., t.test(), wilcox.test(), cor.test()) to create formatted character strings to report the results in accordance with APA manuscript guidelines.

Usage

## S3 method for class 'htest'
apa_print(
x, 
stat_name = NULL, 
est_name = NULL, 
n = NULL, 
conf.int = NULL, 
in_paren = FALSE, 
...
)

Arguments

x An htest object. See details.
stat_name Character. If NULL (the default), the name given in x (or a formally correct adaptation, such as $\chi^2$ instead of "x-squared") is used for the test statistic; otherwise the supplied name is used. See details.
est_name  Character. If NULL (the default), the name given in x (or a formally correct adaptation, such as $r_S$ instead of "rho") is used for the estimate, otherwise the supplied name is used. See details.

n  Numeric. Sample size; required when reporting $\chi^2$ tests, otherwise this parameter is ignored.

conf.int  Numeric. If NULL (the default), the function tries to obtain confidence intervals from x. Other confidence intervals can be supplied as a vector of length 2 (lower and upper boundary, respectively) with attribute conf.level set, e.g., when calculating bootstrapped confidence intervals.

in_paren  Logical. Whether the formatted string is to be reported in parentheses. If TRUE, parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.

Details

The function should work on a wide range of htest objects. Due to the large number of functions that produce these objects and their idiosyncrasies, the returned strings should be compared to the original object. If you experience inaccuracies you may report these [here](https://github.com/crsh/papaja/issues) (please include a reproducible example in your report).

stat_name and est_name are placed in the output string and are thus passed to pandoc or LaTeX through knitr. Thus, to the extent it is supported by the final document type, you can pass LaTeX-markup to format the final text (e.g., $\tau$ yields $\tau$).

Value

apa_print()-methods return a named list of class apa_results containing the following elements:

estimate  One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.

statistic  One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.

full_result  One or more character strings comprised ‘estimate‘ and ‘statistic‘. A single string is returned in a vector; multiple strings are returned as a named list.

table  A data.frame of class apa_results_table that contains all elements of estimate and statistics. This table can be passed to apa_table() for reporting.

Column names in apa_results_table are standardized following the broom glossary (e.g., term, estimate conf.int, statistic, df, df.residual, p.value). Additionally, each column is labelled (e.g., $\hat{\eta}^2_G$ or $t$) using the tinylabels package and these labels are used as column names when an apa_results_table is passed to apa_table().
See Also

Other apa_print: apa_print.BFBayesFactor(), apa_print.aov(), apa_print.emmGrid(), apa_print.glht(), apa_print.list(), apa_print.lme(), apa_print.lm(), apa_print.merMod(), apa_print()

Examples

# Comparisons of central tendencies
t_stat <- t.test(extra ~ group, data = sleep)
apa_print(t_stat)
apa_print(t_stat, stat_name = "tee")

wilcox_stat <- wilcox.test(extra ~ group, data = sleep, exact = FALSE)
apa_print(wilcox_stat)

# Correlations
x <- c(44.4, 45.9, 41.9, 53.3, 44.7, 44.1, 50.7, 45.2, 60.1)
y <- c(2.6, 3.1, 2.5, 5.0, 3.6, 4.0, 5.2, 2.8, 3.8)
cor_stat <- cor.test(x, y, method = "spearman")
apa_print(cor_stat)

# Contingency tables
## Data from Fleiss (1981), p. 139.
smokers <- c(83, 90, 129, 70)
patients <- c(86, 93, 136, 82)
prop_stat <- prop.test(smokers, patients)
apa_print(prop_stat, n = sum(patients))

apa_print.list

Typeset Statistical Results from Linear-Model Comparisons

Description

This method performs comparisons of lm-objects and creates formatted character strings and a model comparison table to report the results in accordance with APA manuscript guidelines.

Usage

## S3 method for class 'list'
apa_print(
  x,
anova_fun = stats::anova,
  conf.int = 0.9,
  boot_samples = 10000,
  progress_bar = interactive(),
  observed = TRUE,
in_paren = FALSE,
...
Arguments

- **x**: List. A list containing to-be-compared `lm` objects. If the list is completely named, element names are used as model names in the output object.
- **anova_fun**: Function. Function to compare model-objects contained in `x`.
- **conf.int**: Numeric. Confidence level for the bootstrap confidence interval for $\Delta R^2$ (range [0, 1]); ignored if `boot_samples = 0`.
- **boot_samples**: Numeric. Number of bootstrap samples to estimate confidence intervals for $\Delta R^2$.
- **progress_bar**: Logical. Determines whether a progress bar is printed while bootstrapping.
- **observed**: Logical. Indicates whether predictor variables were observed. See details.
- **in_paren**: Logical. Whether the formatted string is to be reported in parentheses. If `TRUE`, parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.
- **...**: Additional arguments passed to the function specified as `anova_fun`.

Details

As demonstrated by Algina, Keselman & Penfield (2007), asymptotic confidence intervals for $\Delta R^2$ are often unreliable. Confidence intervals for model comparisons of `lm` objects are, therefore, estimated using their modified percentile bootstrap method. Note that the accuracy of the confidence intervals depends on the number of predictors $p$, their distribution, and the sample size $n$:

"When the predictor distribution is multivariate normal, one can obtain accurate CIs for $\rho^2$ with $n \geq 50$ when $p = 3$. For $p = 6$ and for $p = 9$, $n \geq 100$ is advisable. When the predictor distribution is nonnormal in form, sample size requirements vary with type of nonnormality." (p. 939, Algina, Keselman & Penfield, 2010)

If `MBESS` is available, confidence intervals for $R^2$ are computed using `MBESS::ci.R2()` to obtain a confidence region that corresponds to the confidence level `conf.int`, the default being a 90% CI (see Steiger, 2004). If `observed = FALSE`, it is assumed that predictors are fixed variables, i.e., "the values of the [predictors] were selected a priori as part of the research design" (p. 15, Kelly, 2007); put differently, it is assumed that predictors are not random. The confidence intervals for the regression coefficients in the model comparison table correspond to the $\alpha$-level chosen for $R^2$ and $\Delta R^2$ (e.g., 90% CI or $\alpha = 0.10$ for $R^2$ and $\Delta R^2$ yields a 95% CI for regression coefficients, Steiger, 2004).

Value

`apa_print()`-methods return a named list of class `apa_results` containing the following elements:

- **estimate**: One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.
- **statistic**: One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.
- **full_result**: One or more character strings comprised ‘estimate’ and ‘statistic’. A single string is returned in a vector; multiple strings are returned as a named list.
A data frame of class `apa_results_table` that contains all elements of estimate and statistics. This table can be passed to `apa_table()` for reporting.

Column names in `apa_results_table` are standardized following the `broom` glossary (e.g., `term`, `estimate`, `conf.int`, `statistic`, `df`, `df.residual`, `p.value`). Additionally, each column is labelled (e.g., $\hat{\eta}^2_G$ or $t$) using the `tinylabels` package and these labels are used as column names when an `apa_results_table` is passed to `apa_table()`.

### References


### See Also

- `stats::anova()`

### Other `apa_print`: `apa_print.BFBayesFactor()`, `apa_print.aov()`, `apa_print.emmGrid()`, `apa_print.glht()`, `apa_print.htest()`, `apa_print.lme()`, `apa_print.lm()`, `apa_print.merMod()`, `apa_print()`

### Examples

```r
mod1 <- lm(Sepal.Length ~ Sepal.Width, data = iris)
mod2 <- update(mod1, formula = . ~ . + Petal.Length)
mod3 <- update(mod2, formula = . ~ . + Petal.Width)

# No bootstrapped Delta R^2 CI
apa_print(list(Baseline = mod1, Length = mod2, Both = mod3), boot_samples = 0)
```

### Description

These methods take (generalized) linear model objects to create formatted character strings to report the results in accordance with APA manuscript guidelines.
Usage

```r
## S3 method for class 'lm'
apa_print(
x,
est_name = NULL,
standardized = FALSE,
conf.int = 0.95,
observed = TRUE,
in_paren = FALSE,
...
)

## S3 method for class 'glm'
apa_print(
x,
est_name = NULL,
standardized = FALSE,
conf.int = 0.95,
observed = TRUE,
in_paren = FALSE,
...
)

## S3 method for class 'summary.glm'
apa_print(x, ...)

## S3 method for class 'summary.lm'
apa_print(x, ...)
```

Arguments

- `x` glm or lm object.
- `est_name` Character. If NULL (the default) the name given in `x` (or a formally correct adaptation, such as "b" instead of "b" for standardized regression coefficients) is used. Otherwise the supplied name is used. See details.
- `standardized` Logical. Indicates if coefficients were standardized (e.g., using `scale()`), and leading zeros should be omitted if appropriate. See details.
- `conf.int` Numeric. Either a single value (range [0, 1]) giving the confidence level or a two-column matrix with confidence region bounds as column names (e.g., "2.5 %" and "97.5 %") and coefficient names as row names (in the same order as they appear in `summary(x)$coefficients`). See details.
- `observed` Logical. Indicates whether predictor variables were observed. See details.
- `in_paren` Logical. Whether the formatted string is to be reported in parentheses. If TRUE, parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.
- `...` Arguments passed on to `apa_num`
Details

The coefficient names are sanitized to facilitate their use as list names. Parentheses are omitted and other non-word characters are replaced by _ (see sanitize_terms()).

est_name is placed in the output string and is then passed to pandoc or LaTeX through knitr. Thus, to the extent it is supported by the final document type, you can pass LaTeX-markup to format the final text (e.g., "\\beta" yields $\beta$).

If standardized = TRUE, scale() is removed from coefficient names (see examples). This option is currently ignored for glm-objects.

If conf.int is a single value, confidence intervals are calculated using stats::confint().

If x is an lm object and the MBESS package is available, confidence intervals for $R^2$ are computed using MBESS::ci.R2() to obtain a confidence region that corresponds to the $\alpha$-level chosen for the confidence intervals of regression coefficients (e.g., 95% CI or $\alpha = 0.05$ for regression coefficients yields a 90% CI for $R^2$, see Steiger, 2004). If observed = FALSE, it is assumed that predictors are fixed variables, i.e., "the values of the [predictors] were selected a priori as part of the research design" (p. 15, Kelly, 2007); put differently, it is assumed that predictors are not random.

Value

apa_print() -methods return a named list of class apa_results containing the following elements:

- **estimate**: One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.

- **statistic**: One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.

- **full_result**: One or more character strings comprised ‘estimate’ and ‘statistic’. A single string is returned in a vector; multiple strings are returned as a named list.

- **table**: A data.frame of class apa_results_table that contains all elements of estimate and statistics. This table can be passed to apa_table() for reporting.

Column names in apa_results_table are standardized following the broom glossary (e.g., term, estimate, conf.int, statistic, df, df.residual, p.value). Additionally, each column is labelled (e.g., $\hat{\eta}^2_G$ or $t$) using the tinylabels package and these labels are used as column names when an apa_results_table is passed to apa_table().

References


See Also

`stats::confint()`, `MBESS::ci.pvaf()`

Other `apa_print`: `apa_print.BFBayesFactor()`, `apa_print.aov()`, `apa_print.emmGrid()`, `apa_print.glht()`, `apa_print.htest()`, `apa_print.list()`, `apa_print.lme()`, `apa_print.merMod()`, `apa_print()`

Examples

```r
# Data from Dobson (1990), p. 9.
c1 <- c(4.17, 5.58, 5.18, 6.11, 4.50, 4.61, 5.17, 4.53, 5.33, 5.14)
c2 <- c(4.81, 4.17, 4.41, 3.59, 5.87, 3.83, 6.03, 4.89, 4.32, 4.69)
group <- gl(2, 10, 20, labels = c("CtI", "Trt"))
weight <- c(c1, c2)
lm_fit <- lm(weight ~ group)

apa_print(lm_fit)

trt <- rep(trt, 2)  # More data is always better
c1 <- rep(c1, 2)
lm_fit2 <- lm(scale(trt) ~ scale(c1))

apa_print(lm_fit2, standardized = TRUE)

# It is possible to simplify the regression table with transmute_df_into_label():
transmute_df_into_label(apa_print(lm_fit2, standardized = TRUE))

# Dobson (1990) Page 93: Randomized Controlled Trial:
counts <- c(18, 17, 15, 20, 10, 20, 25, 13, 12)
outcome <- gl(3, 1, 9)
treatment <- gl(3, 3)
da.AD <- data.frame(treatment, outcome, counts)
glm.D93 <- glm(counts ~ outcome + treatment, family = poisson())

apa_print(glm.D93)
```

### apa_print.lme

**Typeset Statistical Results from Nonlinear Hierarchical Models**

#### Description

These methods take mixed-effects models fitted with the `nlme` package and create formatted character strings report the results in accordance with APA manuscript guidelines.

#### Usage

```
## S3 method for class 'lme'
apa_print(x, conf.int = 0.95, in_paren = FALSE, est_name = NULL, ...)

## S3 method for class 'anova.lme'
apa_print(x, in_paren = FALSE, ...)
```
Arguments

- `x` A (non-)linear mixed-effects model fitted with `nlme::lme()` or `nlme::nlme()`.
- `conf.int` Numeric specifying the required confidence level or a named list of additional arguments that are passed to `nlme::intervals.lme()`.
- `in_paren` Logical. Whether the formatted string is to be reported in parentheses. If TRUE, parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.
- `est_name` An optional character. The label to be used for fixed-effects coefficients.
- `...` Further arguments that may be passed to `apa_num` to format estimates (i.e., columns estimate and conf.int).

Value

`apa_print()`-methods return a named list of class `apa_results` containing the following elements:

- `estimate` One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.
- `statistic` One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.
- `full_result` One or more character strings comprised ‘estimate’ and ‘statistic’. A single string is returned in a vector; multiple strings are returned as a named list.
- `table` A data.frame of class `apa_results_table` that contains all elements of `estimate` and `statistic`. This table can be passed to `apa_table()` for reporting.

Column names in `apa_results_table` are standardized following the `broom` glossary (e.g., term, estimate conf.int, statistic, df, df.residual, p.value). Additionally, each column is labelled (e.g., $\hat{\eta}^2_G$ or $t$) using the `tinylabels` package and these labels are used as column names when an `apa_results_table` is passed to `apa_table()`.

See Also

Other `apa_print`: `apa_print.BFBayesFactor()`, `apa_print.aov()`, `apa_print.emmGrid()`, `apa_print.glht()`, `apa_print.htest()`, `apa_print.list()`, `apa_print.lm()`, `apa_print.merMod()`, `apa_print()`

Examples

```r
library(nlme)
fm1 <- lme(distance ~ age, data = Orthodont, method = "ML") # random is ~ age
apa_print(fm1, conf.int = .9) # random int ~ age
# ANOVA-like tables
single_anova <- anova(fm1)
apa_print(single_anova)
```
Typeset Statistical Results from Hierarchical GLM

Description

These methods take objects from various R functions that calculate hierarchical (generalized) linear models to create formatted character strings to report the results in accordance with APA manuscript guidelines.

Usage

```r
## S3 method for class 'merMod'
apa_print(
x,
effects = "fixed",
conf.int = 0.95,
in_paren = FALSE,
est_name = NULL,
...)

## S3 method for class 'mixed'
apa_print(x, ...)
```

Arguments

- `x`: A fitted hierarchical (generalized) linear model, either from `lme4::lmer()`, `lmerTest::lmer()`, `afex::mixed()`, or `lme4::glmer()`.
- `effects`: Character. Determines which information is returned. Currently, only fixed-effects terms ("fixed") are supported.
- `conf.int`: Numeric specifying the required confidence level or a named list specifying additional arguments that are passed to `lme4::confint.merMod()`, see details.
- `in_paren`: Logical. Whether the formatted string is to be reported in parentheses. If TRUE, parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.
- `est_name`: An optional character. The label to be used for fixed-effects coefficients.
- `...`: Further arguments that may be passed to `apa_num` to format estimates (i.e., columns `estimate` and `conf.int`).

Details

Confidence intervals are calculated by calling `lme4::confint.merMod()`. By default, Wald confidence intervals are calculated, but this may change in the future.
Value

apa_print()-methods return a named list of class apa_results containing the following elements:

estimate One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.

statistic One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is NULL.

full_result One or more character strings comprised ‘estimate’ and ‘statistic’. A single string is returned in a vector; multiple strings are returned as a named list.

table A data.frame of class apa_results_table that contains all elements of estimate and statistics. This table can be passed to apa_table() for reporting.

Column names in apa_results_table are standardized following the broom glossary (e.g., term, estimate conf.int, statistic, df, df.residual, p.value). Additionally, each column is labelled (e.g., $\hat{\eta}^2_G$ or $t$) using the tinylabels package and these labels are used as column names when an apa_results_table is passed to apa_table().

See Also

Other apa_print: apa_print.BFBayesFactor(), apa_print.aov(), apa_print.emmGrid(), apa_print.glht(), apa_print.htest(), apa_print.list(), apa_print.lme(), apa_print.lm(), apa_print()

Examples

# Fit a linear mixed model using the lme4 package
# or the lmerTest package (if dfs and p values are desired)
library(lmerTest)
fm1 <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)
# Format statistics for fixed-effects terms (the default)
apa_print(fm1)

apa_print.papaja_wsci  Typeset Within-Subjects Confidence Intervals

Description

This method takes an output object from wsci and creates a table and character strings to report means and within-subjects confidence intervals in a table or in text.

Usage

## S3 method for class 'papaja_wsci'
apa_print(x, ...)

Arguments

- **x**: An object of class papaja_wsci.
- **...**: Arguments passed on to `apa_num`

Value

`apa_print()`-methods return a named list of class `apa_results` containing the following elements:

- **estimate**: One or more character strings giving point estimates, confidence intervals, and confidence level. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is `NULL`.
- **statistic**: One or more character strings giving the test statistic, parameters (e.g., degrees of freedom), and p-value. A single string is returned in a vector; multiple strings are returned as a named list. If no estimate is available the element is `NULL`.
- **full_result**: One or more character strings comprised 'estimate' and 'statistic'. A single string is returned in a vector; multiple strings are returned as a named list.
- **table**: A data.frame of class `apa_results_table` that contains all elements of `estimate` and `statistic`. This table can be passed to `apa_table()` for reporting.

Column names in `apa_results_table` are standardized following the `broom` glossary (e.g., `term`, `estimate`, `conf.int`, `statistic`, `df`, `df.residual`, `p.value`). Additionally, each column is labelled (e.g., \(\hat{\eta}^2_G\) or \(t\)) using the `tinylabels` package and these labels are used as column names when an `apa_results_table` is passed to `apa_table()`.

---

**Description**

Formats matrices and data frames to report them as tables in R Markdown documents according to APA guidelines.

**Usage**

```r
apa_table(x, ...)
```

```
## S3 method for class 'apa_results_table'
apa_table(x, escape = FALSE, ...)
```

```
## S3 method for class 'apa_results'
apa_table(x, ...)
```

```
## S3 method for class 'matrix'
apa_table()
```
x,
caption = NULL,
note = NULL,
stub_indents = NULL,
added_stub_head = NULL,
col_spanners = NULL,
midrules = NULL,
placement = "tbp",
landscape = FALSE,
font_size = NULL,
escape = TRUE,
span_text_columns = TRUE,
...
format.args = NULL
}

## S3 method for class 'list'
apa_table(
  x,
caption = NULL,
note = NULL,
stub_indents = NULL,
added_stub_head = NULL,
col_spanners = NULL,
midrules = NULL,
placement = "tbp",
landscape = FALSE,
font_size = NULL,
escape = TRUE,
merge_method = "indent",
span_text_columns = TRUE,
...
format.args = NULL
)

## S3 method for class 'data.frame'
apa_table(
  x,
caption = NULL,
note = NULL,
stub_indents = NULL,
added_stub_head = NULL,
col_spanners = NULL,
midrules = NULL,
placement = "tbp",
landscape = FALSE,
font_size = NULL,
escape = TRUE,
span_text_columns = TRUE,
        ..., format.args = NULL)

Arguments

  x  Object to print, either a matrix, data.frame, or list. See details.
  ... Arguments passed on to knitr::kable

format A character string. Possible values are latex, html, pipe (Pandoc’s pipe tables), simple (Pandoc’s simple tables), and rst. The value of this argument will be automatically determined if the function is called within a knitr document. The format value can also be set in the global option knitr.table.format. If format is a function, it must return a character string.

digits Maximum number of digits for numeric columns, passed to round(). This can also be a vector of length ncol(x), to set the number of digits for individual columns.

row.names Logical: whether to include row names. By default, row names are included if rownames(x) is neither NULL nor identical to 1:nrow(x).

col.names A character vector of column names to be used in the table.

align Column alignment: a character vector consisting of 'l' (left), 'c' (center) and/or 'r' (right). By default or if align = NULL, numeric columns are right-aligned, and other columns are left-aligned. If length(align) == 1L, the string will be expanded to a vector of individual letters, e.g. 'crlc' becomes c('c', 'l', 'c'), unless the output format is LaTeX.

label The table reference label. By default, the label is obtained from knitr::opts_current$get('label'). To disable the label, use label = NA.

escape Logical. If TRUE special LaTeX characters, such as % or _, in column names, row names, caption, note and table contents are escaped.

caption Character. Caption to be printed above the table.

note Character. Note to be printed below the table.

stub_indents List. A named list of vectors that contain indices of rows to indent. The name of each list element containing the vector is used as title for indented sections.

added_stub_head Character. Used as stub head (name of first column) if row.names = TRUE is passed to kable; ignored if row names are omitted from the table.

col_spanners List. A named list of vectors of length 2 that contain the indices of the first and last column to span. The name of each list element is used as grouping column name. Currently ignored in Word documents.

midrules Numeric. Vector of line numbers in table (not counting column headings) that should be followed by a horizontal rule; currently ignored in Word documents.

placement Character. Indicates whether table should be placed, for example, at the current location (h), at the top (t), bottom (b), or on a separate page (p). Arguments can be combined to indicate order of preference (htb); currently ignored when longtable = TRUE, landscape = TRUE, and in Word documents.
landscape Logical. If TRUE the table is printed in landscape mode; currently ignored in Word documents.

font_size Character. Font size to use for table contents (can be tiny, scriptsize, footnotesize, small, normalsize (default), large, Large, LARGE, huge, Huge). Ignored in Word documents.

span_text_columns Logical. If TRUE tables span across text columns in two-column PDF documents (e.g. when setting classoption: jou). Otherwise ignored.

format_args List. A named list of arguments to be passed to apa_num to format numeric values.

merge_method Character. Determines how to merge tables if x is a list of matrices or data frames with a common structure. Can be either indent or table_spanner. See details.

Details

When using apa_table, the type of the output (LaTeX or Word) is determined automatically by the rendered document type. In interactive R session the output defaults to LaTeX.

If x is a list, all list elements are merged by columns into a single table and the names of list elements are added according to the setting of merge_method.

By default, the width of the table is set to accommodate its contents. In some cases, this may cause the table to exceed the page width. To address this, tables can be rotated 90 degrees by setting landscape = TRUE or, by explicitly using “paragraph columns” with fixed column widths, such that the contents is automatically broken into multiple lines. For example, set align = "lm[5cm]l" to limit the second column to a width of 5 cm. Similarly, to space columns equally use align = paste0("m","1/(ncol(x) + 1), \linewidth")

Note that placement options are not supported in appendices of apa6 documents and will be printed to the document. To omit the printed options set placement = NULL.

Value

A character vector of the table source code of class knit_asis, see knitr::asis_output().

See Also

knitr::kable(), apa_num()

Examples

my_table <- t(apply(cars, 2, function(x) # Create data
    round(c(Mean = mean(x), SD = sd(x), Min = min(x), Max = max(x)), 2))
)

apa_table(
    my_table
    , align = c("l", rep("r", 3))
    , caption = "A summary table of the cars dataset."
beautify_terms

)

apa_table(
  cbind(my_table, my_table)
  , align = c("l", rep("r", 8))
  , caption = "A summary table of the cars dataset."
  , note = "This table was created using apa\_table()"
  , added_stub_head = "Variables"
  , col_spanners = list(`Cars 1` = c(2, 5), `Cars 2` = c(6, 9))
)

apa_table(
  list(`Cars 1` = my_table, `Cars 2` = my_table)
  , caption = "A summary table of the cars dataset."
  , added_stub_head = "Variables"
)

---

beautify_terms Prettify Term Names

**Description**

Remove parentheses, replace colons with $\times$. Useful to prettify term names in apa_print() tables.

**Usage**

beautify_terms(x, ...)

## S3 method for class 'character'
beautify_terms(x, standardized = FALSE, retain_period = FALSE, ...)

## S3 method for class 'numeric'
beautify_terms(x, standardized = FALSE, ...)

## S3 method for class 'factor'
beautify_terms(x, standardized = FALSE, ...)

## S3 method for class 'data.frame'
beautify_terms(x, ...)

**Arguments**

- **x**
  Character. Vector of term names to be prettified.

- **...**
  Additional arguments passed to apa_num(), for numeric values in x, ignored otherwise.

- **standardized**
  Logical. If TRUE, the name of the function scale() will be removed from term names.
**retained_period**

Logical. If TRUE, any periods in term names will be retained, otherwise they will be replaced by a space.

**Value**

A character vector or data.frame (if x is a data.frame) containing term names modified for nicer printing.

**Examples**

```
beautify_terms("a:b")
beautify_terms("scale(x)", standardized = TRUE)
beautify_terms("snake_case")
```

---

**cite_r**

*Cite R and R Packages*

**Description**

Creates character strings to cite R and R packages.

**Usage**

```
cite_r(
  file = NULL,
  prefix = "R-",
  footnote = FALSE,
  pkgs = NULL,
  omit = TRUE,
  ...
)
```

**Arguments**

- **file**
  Character. Path and name of the .bib-file holding the references. If NULL, only R is cited.

- **prefix**
  Character. Prefix used for all R-package reference handles.

- **footnote**
  Logical. Indicates if packages should be cited in a footnote. Ignored if no package information is available.

- **pkgs**
  Character. Vector of package names to cite or omit depending on omit.

- **omit**
  Logical. If TRUE, pkgs constitutes a list of packages not to cite (a blacklist). If FALSE, pkgs constitutes a list of packages to cite (a whitelist).

- **...**
  Additional arguments, which are currently ignored.
Details

If footnote = FALSE, a character string citing R and R packages including version numbers is returned. Otherwise a named list with the elements r and pkgs is returned. The former element holds a character string citing R and a reference to a footnote; the latter element contains a character string that creates the footnote. For correct rendering, the footnote string needs to be a separate paragraph in the R Markdown document.

Value

If footnote = FALSE a character string is returned, otherwise a named list with the elements r and pkgs.

See Also

r.refs(), knitr::write_bib()

Examples

cite_r()

calculates the deviation that is needed to construct confidence intervals for a vector of observations.

Usage

conf_int(x, level = 0.95, na.rm = TRUE)

conf.int(x, level = 0.95, na.rm = TRUE)

ci(x, level = 0.95, na.rm = TRUE)

Arguments

x Numeric. A vector of observations from your dependent variable.
level Numeric. Defines the width of the interval if confidence intervals are plotted. Defaults to 0.95 for 95% confidence intervals.
na.rm Logical. Specifies if missing values should be removed.

Value

Returns a single numeric value, the deviation of the symmetric confidence bounds from the mean based on the t distribution.
fetch_web.refs  Fetch a .bib-reference file from the web (defunct)

Description

Downloads and saves a .bib-reference file from the web, so it can be used to cite references in a Markdown-document using pandoc or LaTeX. This function has been defunct. Please use `download` from the `downloader` instead.

Usage

```r
fetch_web_refs(x, bib_name)
```

Arguments

- `x` Character. URL of the .bib-file to fetch.
- `bib_name` Character. The path and name of the file to be created.

Details

If the function is called in an RMarkdown-document the file name specified as `bib_name` can be used in the YAML header as bibliography.

Value

Returns NULL invisibly.

See Also

cite_r(), r.refs(), knitr::write_bib()

fetch_zotero.refs  Save a collection from a Zotero-Account to a BibTeX-file (defunct)

Description

Downloads and saves a Zotero reference library (or a subset) and saves it as BibTeX file. This function has been defunct. Use `ReadZotero()` from the `RefManageR` package instead.

Usage

```r
fetch_zotero.refs(  
  x,  
  bib_name,  
  API_key = NULL,  
  collection = NULL,  
  lib_type = "user"  
)
```
fetch_zotero_refs

Arguments

- x: Character. Zotero user or group ID, see details.
- bib_name: Character. Name of the BibTeX-file references are saved to.
- API_key: Character. Zotero API key, see details.
- collection: Character. Optional ID of a collection in the Zotero library, see details.
- lib_type: Character. Specifies if the supplied ID is associated with a Zotero user or group.

Details

This function retrieves references through the Zotero web API. x takes a Zotero user or group ID that can be retrieved from the Zotero.org user or group Feeds/API settings. An authentication key (API_key) is required to access nonpublic Zotero libraries. Authentication keys can also be generated in the Zotero.org user or group Feeds/API settings.

If the requested reference collection is larger than 100 records, multiple API calls are initiated because the number of retrieved records is limited to 100 per API call. Frequent API calls will result in a temporary access block. Thus, there is an (currently unknown) upper limit to the length of reference collections that can be retrieved through this function. It is generally recommended to comment out calls to this function in R Markdown documents during periods of frequent knitting to limit the number of API calls and limit the number of references to those needed for the current document by setting up collections in your Zotero library.

Collection keys (collection), i.e. identifiers of reference library subsets, can be retrieved by accessing them via a web browser. They keys are contained in the URL: https://www.zotero.org/<USERNAME>/items/collectionKey/<COLLECTIONKEY>

Zotero web API calls can be slow, especially for large reference collections. If available, this function will use the downloader-package, which speeds up reference downloads considerably.

Value

Returns bib_name invisibly.

Author(s)

Christoph Stahl, Frederik Aust

See Also

cite_r(), r_refs()
glue_apa_results  
Create a New apa_results Object

Description

Typeset the contents of an object according to the specified expression strings and create a new or extend an existing apa_results object.

Usage

```r
glue_apa_results(x = NULL, term_names = NULL, ...)  
add_glue_to_apa_results(..., est_glue, stat_glue, container, sublist = NULL, term_names = NULL, in_paren = FALSE, est_first = TRUE, simplify = TRUE)
```

Arguments

- `x`  
  An environment, list or data frame used to look up values for substitution.
- `term_names`  
  Character. Used as names for the estimate-, statistics-, and full_result sub-lists, if multiple estimates or statistics are glued. Defaults to `attr(x, "sanitized_term_names")`.
- `...`  
  [expressions]  
  Unnamed arguments are taken to be expression string(s) to format. Multiple inputs are concatenated together before formatting. Named arguments are taken to be temporary variables available for substitution.
- `est_glue`  
  Character. (Named vector of) expressions string(s) to format. Each string creates a new (named) element in the estimate sub-list.
- `stat_glue`  
  Character. (Named vector of) expressions string(s) to format. Each string creates a new (named) element in the statistic sub-list.
- `container`  
  List of class apa_results to add the glued results to.
- `sublist`  
  Character. Name of (new) sub-list in estimate statistics, and full_result to append glued results to (e.g., modelfit).
- `in_paren`  
  Logical. Whether the formatted string is to be reported in parentheses. If TRUE, parentheses in the formatted string (e.g., those enclosing degrees of freedom) are replaced with brackets.
- `est_first`  
  Logical. Determines in which order estimate and statistic are glued together to full_result.
simplify Logical. Determines whether the estimate, statistic, and full_result sub-lists should be simplified if only one term is available from the model object.

Value

Returns a list of class apa_results, see apa_print().

Examples

# Tidy and typeset output
iris_lm <- lm(Sepal.Length ~ Petal.Length + Petal.Width, iris)
tidy_iris_lm <- broom::tidy(iris_lm, conf.int = TRUE)
tidy_iris_lm$p.value <- apa_p(tidy_iris_lm$p.value)

glance_iris_lm <- broom::glance(iris_lm)
glance_iris_lm$p.value <- apa_p(glance_iris_lm$p.value, add_equal = TRUE)
glance_iris_lm$df <- apa_num(as.integer(glance_iris_lm$df))
glance_iris_lm$df.residual <- apa_num(as.integer(glance_iris_lm$df.residual))

# Create `apa_results`-list
lm_results <- glue_apa_results(
  x = tidy_iris_lm,
  df = glance_iris_lm$df.residual,
  est_glue = "$b = <<estimate>>, 95% CI [<<conf.low>>, [<<conf.high>>]]$",
  stat_glue = "$t(<<df>>) = <<statistic>>$, $p <<p.value>>$",
  term_names = make.names(names(coef(iris_lm)))
)

# Add modelfit information
add_glue_to_apa_results(
  .x = glance_iris_lm,
  container = lm_results,
  sublist = "modelfit",
  est_glue = c(
    r2 = "$R^2 = <<r.squared>>$",
    aic = ""
  ),
  stat_glue = c(
    r2 = "$F(<<df>>, <<df.residual>>) = <<statistic>>$, $p <<add_equal(p.value)>>$",
    aic = "$\text{AIC} = <<AIC>>$"
  )
)

hd_int

Highest-Density Intervals

Description

Calculates the highest-density interval of a vector of values.
Usage

hd_int(x, level = 0.95)

Arguments

x  Numeric. A vector of observations.
level  Numeric. Defines the width of the interval. Defaults to 95% highest-density intervals.

Description

Takes a single character or a list of characters and replaces parentheses with brackets. Can be used to prepare a string of statistics (e.g. containing degrees of freedom) for reporting within parentheses.

Usage

in_paren(x)

Arguments

x  Character. Single character or list of characters.

Value

An object of the same type as x, where all parentheses have been replaced by brackets.

See Also

apa_print()

Examples

t_stat <- t.test(extra ~ group, data = sleep)
t_test_res <- apa_print(t_stat)
in_paren(t_test_res$stat)
in_paren(t_test_res[1:3])
papaja

Prepare APA Journal Articles with R Markdown

Description

papaja is an award-winning R package that facilitates creating computationally reproducible, submission-ready manuscripts which conform to the American Psychological Association (APA) manuscript guidelines (6th Edition).

Details

papaja provides

- an R Markdown template that can be used with (or without) RStudio to create PDF documents (using the apa6 LaTeX class) or Word documents (using a .docx-reference file).
- Functions to typeset the results from statistical analyses (e.g., apa_print()),
- functions to create tables (apa_table()), and
- functions to create figures in accordance with APA guidelines (e.g., apa_factorial_plot()).

System requirements

To use papaja you need either an up-to-date version of RStudio or pandoc. If you want to create PDF- in addition to DOCX-documents you additionally need a TeX distribution. We recommend TinyTex, which can be installed from within R via the tinytex package.

Please refer to the papaja manual for detailed installation instructions.

Getting help

For a comprehensive introduction to papaja, see the current draft of the manual. If you have a specific question that is not answered in the manual, feel free to ask a question on Stack Overflow using the papaja tag. If you believe you have found a bug or would like to request a new feature, open an issue on Github and provide a minimal complete verifiable example.

Authors

Frederik Aust (frederik.aust at uni-koeln.de). Marius Barth (marius.barth at uni-koeln.de).

Maintainer

Frederik Aust (frederik.aust at uni-koeln.de).
quote_from_tex  Quote from TeX document

Description
Includes a labelled quote from a LaTeX document 'asis'.

Usage
quote_from_tex(x, file)

Arguments
  x               Character. One or more quote labels.
  file            Character. Path to LaTeX file from which to quote.

Details
Searches the LaTeX document specified in file for labelled quotes, i.e. paragraphs that are enclosed in \% <@~{#quote-label} and \% ~@>} tags in LaTeX comments on separate lines. The labelled quote is then inserted and rendered asis.

Value
A character vector of LaTeX document text of class knit_asis, see knitr::asis_output().

remove_comments  Remove Comments

Description
Removes markdown comments from an R Markdown file.

Usage
remove_comments(x, file)

Arguments
  x               Character. Path to an R Markdown file.
  file            Character. Name of the new R Markdown file without comments.

Value
No return value, called to write text to file.
### Revision Letter

**Description**

Template for creating a journal revision letters.

**Usage**

```r
revision_letter_pdf(keep_tex = TRUE, ...)
```

**Arguments**

- `keep_tex`: Logical. Whether to keep the intermediate tex file used in the conversion to PDF.
- `...`: Arguments passed on to `bookdown::pdf_document2`

**Value**

R Markdown output format to pass to `rmarkdown::render()`

**See Also**

- `bookdown::html_document2()`

---

### r.refs

**Create a Reference File for R and R Packages**

**Description**

Creates a .bib-reference file for the installed R version and R-packages, so they can be cited in an R Markdown-document.

**Usage**

```r
r.refs(
  file,
  append = TRUE,
  prefix = "R-",
  type_pref = c("Article", "Book"),
  tweak = TRUE
)

create_bib(
  x,
```
file,
append = TRUE,
prefix = "R-",
type_pref = c("Article", "Book"),
tweak = TRUE
)

Arguments

file Character. Path and name of the file to be created or updated.
append Logical. Indicates if existing bibliography should be complemented or overwritten. See details.
prefix Character. Prefix for all R-package reference handles.
type_pref Character. A vector of BibTeX entry types in the order by which to prioritize packages CITATION entries. See details.
tweak Logical. Indicates whether to fix some known problems in citations (based on write_bib).
x Character. Names of packages to include in bibliography.

Details

r.refs is a wrapper for create_bib to create a bibliography for R and all attached or cached packages.

By default, if a file exists at the specified location, r.refs reads the file and appends missing citation information to the end of the file (create_bib always overwrites existing files). It is recommended to use a bibliography-file dedicated to R-references.

Beware that chunks loading packages should generally not be cached. r.refs will make all packages loaded in cached chunks citable, but it won’t know when you remove a package from a cached chunk. This can result in unused package references in your bibliography-file that will be cited when using cite_r.

If a package provides citation information in a CITATION file, a reference is selected based on the preferred order of reference types specified in type_pref. By default, available articles are cited rather than books. If no reference of the specified types is available, the first reference is used. If multiple references of the preferred type are given all of them are cited. Finally, if no CITATION file exists a reference is generated from the DESCRIPTION file by citation.

Value

Invisibly returns the bibliography written to file.

See Also

cite_r(), knitr::write_bib(), utils::citation(), utils::toLatex()
**sanitize_terms**

**Sanitize Term Names**

**Description**

Remove characters from term names that will be difficult to address using the \$-operator. *This function is not exported.*

**Usage**

```r
sanitize_terms(x, standardized = FALSE)
```

```r
## S3 method for class 'character'
sanitize_terms(x, standardized = FALSE)
```

```r
## S3 method for class 'factor'
sanitize_terms(x, standardized = FALSE)
```

```r
## S3 method for class 'data.frame'
sanitize_terms(x, standardized = FALSE)
```

```r
## S3 method for class 'list'
sanitize_terms(x, standardized = FALSE)
```

**Arguments**

- **x** Character. Vector of term names to be sanitized.
- **standardized** Logical. If TRUE, the name of the function `[scale()]` will be removed from term names.

**Value**

An object of the same class as `x` containing sanitized term names as characters.

**Examples**

```r
sanitize_terms(c("(Intercept)", "Factor A", "Factor B", "Factor A:Factor B", "scale(FactorA)"))
```
se

Standard Error of the Mean

**Description**

Calculates the standard error of the mean.

**Usage**

```r
se(x, na.rm = TRUE)
```

**Arguments**

- `x`: Numeric. A vector of observations.
- `na.rm`: Logical. Specifies if missing values should be removed.

**Value**

The standard error of the mean as numeric vector of length 1.

---

simple_codebook

**Simple Codebook**

**Description**

Generate a simple codebook in CSV-format from a (labelled) `data.frame`.

**Usage**

```r
simple_codebook(x, ...)
```

**Arguments**

- `x`: `data.frame`. Data to be documented.
- `...`: Arguments passed on to `utils::write.table`
  - `file`: either a character string naming a file or a `connection` open for writing. """ indicates output to the console.
  - `append`: logical. Only relevant if `file` is a character string. If `TRUE`, the output is appended to the file. If `FALSE`, any existing file of the name is destroyed.
  - `quote`: a logical value (TRUE or FALSE) or a numeric vector. If TRUE, any character or factor columns will be surrounded by double quotes. If a numeric vector, its elements are taken as the indices of columns to quote. In both cases, row and column names are quoted if they are written. If FALSE, nothing is quoted.
sort_terms

Sort ANOVA or Regression Table by Predictors/Effects

Description

Sort rows in ANOVA or regression tables produced by `apa_print()` by complexity (i.e., main effects, two-way interactions, three-way interactions, etc.).
**Usage**

```r
sort_terms(x, colname)
```

**Arguments**

- `x` A data frame. For example, the table element produced by `apa_print()`.
- `colname` Character. Column name of the data.frame containing the terms to sort.

**Value**

Returns the same data.frame with reordered rows.

**Examples**

```r
npk_aov <- aov(yield ~ block + N * P * K, npk)
npk_aov_results <- apa_print(npk_aov)
sort_terms(npk_aov_results$table, "term")
```

---

**summary.papaja_wsci**  
*Summarize Within-Subjects Confidence Intervals*

**Description**

Calculate upper and lower limits of within-subjects confidence intervals calculated with `wsci()` and return them along their respective means.

**Usage**

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

**Arguments**

- `object` An object of class papaja_wsci, generated with function `wsci()`.
- `...` Further arguments that may be passed, currently ignored.

**Value**

A data.frame containing means as well as lower and upper confidence bounds for each cell of the design.
Description

**ggplot2** theme with a white panel background, no grid lines, large axis and legend titles, and increased text padding for better readability.

Usage

```r
theme_apa(base_size = 12, base_family = "", box = FALSE)
```

Arguments

- `base_size` Numeric. Base font size; other font sizes and margins are adjusted relative to this.
- `base_family` Character. Base font family.
- `box` Logical. Indicates whether to draw a black panel border.

Value

Object of class theme and gg, see `ggplot2::theme()`.

See Also

`ggplot2::theme_bw()`, `ggplot2::theme()`

Examples

```
# Copied from ?ggtheme
mtcars2 <- within(mtcars, {
  vs <- factor(vs, labels = c("V-shaped", "Straight"))
  am <- factor(am, labels = c("Automatic", "Manual"))
  cyl <- factor(cyl)
  gear <- factor(gear)
})

library("ggplot2")
p1 <- ggplot(mtcars2) +
  geom_point(aes(x = wt, y = mpg, colour = gear)) +
  labs(
    title = "Fuel economy declines as weight increases",
    subtitle = "(1973-1974)",
    x = "Weight (1000 lbs)",
    y = "Fuel economy (mpg)",
    colour = "Gears"
  )
```

transmute_df_into_label

Transmute Degrees-of-Freedom Columns into Variable Labels

**Description**

Takes the output from `apa_print()` methods and modifies the results table by transmuting information about degrees of freedom into the variable labels of test-statistic columns.

**Usage**

```r
transmute_df_into_label(x, check_df = TRUE, ...)
```

```r
df_into_label(x, check_df = TRUE, ...)

## S3 method for class 'apa_results'
transmute_df_into_label(x, check_df = TRUE, ...)

## S3 method for class 'apa_results_table'
transmute_df_into_label(x, check_df = TRUE, ...)
```

**Arguments**

- `x` Either the complete output object created by `apa_print()` methods, or the table component of such objects.
- `check_df` Logical. If TRUE (the default), checks if degrees-of-freedom-columns contain non-integer values.
- `...` further arguments passed from an to other methods

**Value**

An object of the same class as `x`, where a redundant column with degrees of freedom has been incorporated into the column label of the column `statistic`.

**Examples**

```r
apa_out <- apa_print(aov(yield ~ N * P, npk))

# Standard output with separate columns for degrees of freedom:
apa_out$table

# Modified output where degrees of freedom are incorporated into the variable label of column 'statistic':
transmute_df_into_label(apa_out)$table
```
Within-Subjects Confidence Intervals

Description
Calculate Cousineau-Morey within-subjects confidence intervals.

Usage
wsci(data, id, factors, dv, level = 0.95, method = "Morey")
within_subjects_conf_int(data, id, factors, dv, level = 0.95, method = "Morey")

Arguments
- **data**: A data.frame that contains the data.
- **id**: Character. Variable name that identifies subjects.
- **factors**: Character. A vector of variable names that is used to stratify the data.
- **dv**: Character. The name of the dependent variable.
- **level**: Numeric. Defines the width of the interval. Defaults to 0.95 for 95% confidence intervals.
- **method**: Character. The method that is used to calculate CIs. Currently, "Morey" and "Cousineau" are supported. Defaults to "Morey".

Value
A data.frame with additional class papaja_wsci. The summary() method for this class returns a data.frame with means along lower and upper limit for each cell of the design.

References

Examples
```r
wsci(
  data = npk,
  id = "block",
  dv = "yield",
  factors = c("N", "P")
)
```
$.apa_results_table

Extract Parts of an APA Results Table

Description

These methods are only defined for backward compatibility with older versions of papaja. In the past, the column names of apa_results_tables were less standardized than they are today. In order to maintain backwards compatibility, it is still possible to extract columns with the old column names, because we here provide aliased indexing. Note that aliased indexing will be defunct in a future release of papaja.

Usage

## S3 method for class 'apa_results_table'

x$name

## S3 method for class 'apa_results_table'

x[[i, exact = TRUE]]

## S3 method for class 'apa_results_table'

x[i, j, ..., drop = TRUE]

Arguments

- **x**: object from which to extract element(s) or in which to replace element(s).
- **name**: A literal character string or a name (possibly backtick quoted). For extraction, this is normally (see under ‘Environments’) partially matched to the names of the object.
- **i, j, ...**: Indices specifying elements to extract. See base::Extract() for details.
- **exact**: Controls possible partial matching of [[ when extracting by a character vector (for most objects, but see under ‘Environments’). The default is no partial matching. Value NA allows partial matching but issues a warning when it occurs. Value FALSE allows partial matching without any warning.
- **drop**: For matrices and arrays. If TRUE the result is coerced to the lowest possible dimension (see the examples). This only works for extracting elements, not for the replacement. See drop for further details.

Value

A (vector of) character values as extracted from an object of class apa_results_table.
Index

* **apa_print**
  - `apa_print`, 26
  - `apa_print.aov`, 27
  - `apa_print.BFBayesFactor`, 31
  - `apa_print.emmGrid`, 34
  - `apa_print.glht`, 37
  - `apa_print.h.test`, 38
  - `apa_print.list`, 40
  - `apa_print.lm`, 42
  - `apa_print.lme`, 45
  - `apa_print.merMod`, 47
  - `apa_print.BFBayesFactorTop`, 31

* **plots for factorial designs**
  - `apa_barplot`, 5
  - `apa_beeplot`, 8
  - `apa_factorial_plot`, 12
  - `apa_lineplot`, 18
  - `.apa_results_table` ($`.apa_results_table`), 72
  - `[[`.apa_results_table` ($`.apa_results_table`), 72
  - `$`.apa_results_table` , 72
  - `add.equals`, 3
  - `add_glue_to_apa_results` (glue_apa_results), 58
  - `afex::mixed()`, 47
  - `aov()`, 31
  - `apa6_doc` (apa6_pdf), 3
  - `apa6_docx` (apa6_pdf), 3
  - `apa6_docx()`, 26
  - `apa6_pdf`, 3
  - `apa6_word` (apa6_pdf), 3
  - `apa_barchart`, 5, 11, 15, 20
  - `apa_barchart()`, 7, 10, 12, 15, 20
  - `apa_beeplot`, 7, 8, 15, 20
  - `apa_beeplot()`, 7, 10, 12, 15, 20
  - `apa_confint` (apa_interval), 16
  - `apa_df`, 11
  - `apa_df()`, 24, 25
  - `apa_factorial_plot`, 6, 7, 9, 11, 12, 19, 20
  - `apa_factorial_plot()`, 7, 10, 15, 20, 61
  - `apa_haint` (apa_interval), 16
  - `apa_interval`, 16
  - `apa_lineplot`, 7, 11, 15, 18
  - `apa_lineplot()`, 7, 10, 12, 15, 20
  - `apa_num`, 17, 21, 24, 30, 35, 37, 39, 43, 46, 47, 49, 52
  - `apa_num()`, 12, 17, 25, 53
  - `apa_num.numeric`, 32
  - `apa_p`, 24
  - `apa_p()`, 12, 24
  - `apa_prepare_doc`, 25
  - `apa_print`, 26, 31, 34, 36, 38, 40, 42, 45, 46, 48
  - `apa_print()`, 53, 59–61, 67, 68, 70
  - `apa_print.afex_aov` (apa_print.aov), 27
  - `apa_print.anova` (apa_print.aov), 27
  - `apa_print.anova.lme` (apa_print.lme), 45
  - `apa_print.Anova.mlm` (apa_print.aov), 27
  - `apa_print.aov`, 27, 28, 34, 36, 38, 40, 42, 45, 46, 48
  - `apa_print.aovlist` (apa_print.aov), 27
  - `apa_print.BFBayesFactor`, 27, 31, 34, 36, 38, 40, 42, 45, 46, 48
  - `apa_print.BFBayesFactorTop` (apa_print.BFBayesFactor), 31
  - `apa_print.emmGrid`, 27, 31, 34, 36, 38, 40, 42, 45, 46, 48
  - `apa_print.glht`, 27, 31, 34, 36, 37, 40, 42, 45, 46, 48
  - `apa_print.glht` (apa_print.lme), 42
  - `apa_print.h.test`, 27, 31, 34, 36, 38, 40, 42, 45, 46, 48
  - `apa_print.list`, 27, 31, 34, 36, 38, 40, 42, 45, 46, 48
  - `apa_print.list()` , 31
  - `apa_print.lm`, 27, 31, 34, 36, 38, 40, 42, 42, 46, 48
  - `apa_print.lme`, 27, 31, 34, 36, 38, 40, 42, 45, 46, 48
INDEX

45, 48
apa_print.lsmobj(apa_print.emmGrid), 34
apa_print.manova(apa_print.aov), 27
apa_print.merMod, 27, 31, 34, 36, 38, 40, 42, 45, 46, 47
apa_print.mixed(apa_print.merMod), 47
apa_print.papaja_wsci, 48
apa_print.summary.Anova.mlm(apa_print.aov), 27
apa_print.summary.aovlist(apa_print.aov), 27
apa_print.summary.aov(apa_print.aov), 27
apa_print.summary.glht(apa_print.emmGrid), 34
apa_print.summary.emm(apa_print.emmGrid), 34
apa_table, 49
apa_table(), 61
apply(), 24
arrows(), 7, 10, 11, 14, 15, 19, 20
axis(), 6, 7, 10, 14, 15, 19, 20
backtick, 72
base::Extract(), 72
base::formatC, 22
beautify_terms, 53
bookdown::html_document2(), 63
bookdown::pdf_document2(), 5
bookdown::word_document2(), 5
car::Anova(), 31
ci (conf_int), 55
citation, 64
cite_r, 54, 64
cite_r(), 56, 57, 64
class, 26
conf.int (conf_int), 55
conf_int, 55
conf_int(), 6, 7, 9, 10, 13, 14, 19, 20
connection, 66
cor.test(), 38
create_bib, 64
create_bib(r.refs), 63
df_into_label( transmute_df_into_label), 70
drop, 72
fetch_web_refs, 56
fetch_zotero_refs, 56
file, 67
format, 23
formatC, 25
ggplot2::theme(), 69
ggplot2::theme_bw(), 69
glue_apa_results, 58
graphics::plot.window, 13
hd_int, 59
in_paren, 60
includes, 4
kable, 51
knitr::asis_output(), 52, 62
knitr::kable, 51
knitr::kable(), 52
knitr::write_bib(), 55, 56, 64
legend(), 7, 10, 11, 14, 15, 19, 20
lines(), 7, 10, 11, 14, 15, 19, 20
lm, 40, 41
lme4::confint.merMod(), 47
lme4::glmer(), 47
lme4::lmer(), 47
lmerTest::lmer(), 47
MBESS::ci.pvaf(), 45
MBESS::ci.R2(), 41, 44
name, 72
names, 72
nlme::intervals.lme(), 46
nlme::lme(), 46
nlme::nlme(), 46
opts_current, 51
papaja, 61
pdf_document2, 4
plot.new(), 7, 10, 15, 20
plot.window(), 7, 10, 15, 20
points(), 7, 10, 11, 14, 15, 19, 20
print_confint (apa_interval), 16
print_df (apa_df), 11
print_hdint (apa_interval), 16
print_interval (apa_interval), 16
print_num (apa_num), 21
print_p (apa_p), 24
printnum (apa_num), 21
printp (apa_p), 24
quote_from_tex, 62
r_refs, 63
r_refs(), 55–57
rect(), 7, 10, 14, 15, 20
remove_comments, 62
revision_letter_pdf, 63
rmarkdown::render(), 5, 63
rmarkdown_format, 4
sanitize_terms, 65
sanitize_terms(), 44
scale(), 53, 65
se, 66
se(), 7, 10, 14, 20
signif, 22
simple_codebook, 66
sort_terms, 67
stats::anova(), 42
stats::confint(), 44, 45
summary.manova, 30
summary.papaja_wsci, 68
t.test(), 38
theme_apa, 69
title(), 6, 7, 10, 14, 15, 19, 20
transmute_df_into_label, 70
utils::citation(), 64
utils::toLatex(), 64
utils::write.csv(), 67
utils::write.table, 66
wilcox.test(), 38
within_subjects_conf_int (wsci), 71
within_subjects_conf_int(), 7, 10, 14, 20
word_document2, 4
write_bib, 64
wsci, 48, 71
wsci(), 7, 10, 14, 20, 68