Package 'cocoon'

November 5, 2024

Title Extract, Format, and Print Statistical Output

Version 0.1.0

Maintainer Jeffrey R. Stevens <jeffrey.r.stevens@protonmail.com>

Description Provides functions that format statistical output in a way that
can be inserted into R Markdown documents. This is analogous to the
apa_print() functions in the 'papaja' package but prints Markdown
or LaTeX syntax.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

Suggests BayesFactor, correlation, knitr, rmarkdown, testthat (>=
3.0.0)

Config/testthat/edition 3

Imports dplyr, lifecycle

URL https://github.com/JeffreyRStevens/cocoon,
https://jeffreyrstevens.github.io/cocoon/

BugReports https://github.com/JeffreyRStevens/cocoon/issues

VignetteBuilder knitr

NeedsCompilation no

Author Jeffrey R. Stevens [aut, cre, cph]
(<https://orcid.org/0000-0003-2375-1360>)

Repository CRAN

Date/Publication 2024-11-05 10:10:02 UTC

Contents

---

format_bf   *Format Bayes factors*

Description

`format_bf()` can input either a BayesFactor object or a vector of Bayes factor values. By default, this function rounds Bayes factors greater than 1 to one decimal place and Bayes factors less than 1 to two decimal places. Values greater than 1000 or less than 1/1000 are formatted using scientific notation. Cutoffs can be set that format the values as greater than or less than the cutoffs (e.g., BF > 1000 or BF < 0.001). Numbers of digits, cutoffs, italics, and label subscripts are all customizable.

Usage

```
format_bf(
  x,
  digits1 = 1,
  digits2 = 2,
  cutoff = NULL,
  label = "BF",
  italics = TRUE,
  subscript = "10",
  type = "md"
)
```

Arguments

x	BayesFactor object or vector of numeric Bayes factor values
digits1	Number of digits after the decimal for Bayes factors > 1
digits2	Number of digits after the decimal for Bayes factors < 1
cutoff	Cutoff for using _BF_~10~ > \<cutoff\> or _BF_~10~ < 1 / \<cutoff\> (value must be > 1)
label	Character string for label before Bayes factor. Default is BF. Set `label = ""` to return just the formatted Bayes factor value with no label or operator (=, <, >)

italics	Logical value (default = TRUE) for whether label should be italicized (*BF* or BF)
subscript	Subscript to include with *BF* label ("10", "01", or "" for no subscript)
type	Type of formatting ("md" = markdown, "latex" = LaTeX)

Value

A character string that includes label (by default *BF*~10~) and then the Bayes factor formatted in Markdown or LaTeX. If Bayes factor is above or below cutoff, _BF_~10~ > <cutoff> or _BF_~10~ < 1 / <cutoff> is used.

See Also

Other functions for printing statistical objects: format_corr(), format_stats(), format_stats.BFBayesFactor(), format_stats.easycorrelation(), format_stats.htest(), format_ttest()

Examples

```
# Format BFBayesfactor objects from {BayesFactor} package
format_bf(BayesFactor::lmBF(mpg ~ am, data = mtcars))

# Format Bayes factors > 1
format_bf(12.4444)

# Bayes factors > 1000 will use scientific notation
format_bf(1244.44)

# Control digits for Bayes factors > 1 with digits1
format_bf(1244.44, digits1 = 3)

# Control cutoff for output
format_bf(1244.44, cutoff = 10000)

# Format Bayes factors < 1
format_bf(0.111)

# Bayes factors < 0.001 will use scientific notation
format_bf(0.0001)

# Control digits for Bayes factors < 1 with digits2
format_bf(0.111, digits2 = 3)

# Control cutoff for output
format_bf(0.001, cutoff = 100)

# Return only Bayes factor value (no label)
format_bf(12.4444, label = "")

# Format for LaTeX
format_bf(12.4444, type = "latex")
```

---

format_chr   *Format character strings with italics and type*

Description

Format character strings with italics and type

Usage

```
format_chr(x, italics = TRUE, type = "md")
```

Arguments

x	Character string
italics	Logical value (default = TRUE) for whether text should be italicized
type	Type of formatting ("md" = markdown, "latex" = LaTeX)

Value

A character string that has either Markdown or LaTeX formatting for italics or not.

Examples

```
format_chr("Hello world!")
# Format in LaTeX syntax
format_chr("Hello world!", type = "latex")
# Remove italics
format_chr("Hello world!", italics = FALSE)
```

---

format_corr   *Format correlation statistics*

Description

[Superseded]

With `format_corr()` you can format correlation statistics generated from `cor.test()` output. This is now an internal function superceded by `format_stats()`, which we recommend using instead.

Usage

```
format_corr(x, digits, pdigits, pzero, full, italics, type, ...)
```

Arguments

x	An htest object
digits	Number of digits after the decimal for means, confidence intervals, and test statistics
pdigits	Number of digits after the decimal for p-values, ranging between 1-5 (also controls cutoff for small p-values)
pzero	Logical value (default = FALSE) for whether to include leading zero for p-values
full	Logical value (default = TRUE) for whether to include means and confidence intervals or just test statistic and p-value
italics	Logical value (default = TRUE) for whether *p* label should be italicized
type	Type of formatting ("md" = markdown, "latex" = LaTeX)
...	Additional arguments passed to methods.

Value

A character string of statistical information formatted in Markdown or LaTeX.

See Also

Other functions for printing statistical objects: format_bf(), format_stats(), format_stats.BFBayesFactor(), format_stats.easycorrelation(), format_stats.htest(), format_ttest()

Examples

```
# format_stats(cor.test(mtcars$mpg, mtcars$cyl))
```

---

format_num	*Format numbers*

Description

Format numbers

Usage

```
format_num(x, digits = 1, pzero = TRUE)
```

Arguments

x	Number
digits	Number of digits after the decimal
pzero	Logical value (default = TRUE) for whether to include leading zero numbers less than 1

Value

A character string formatting the number with specified number of digits after the decimal.

Examples

```
format_num(pi, digits = 2)
format_num(pi, digits = 4)
```

---

format_p   *Format p-values*

Description

`format_p()` inputs numeric vectors of p-values. Cutoffs can be set that format the values as less than the cutoffs (e.g., p < 0.001). The default output is APA formatted, but numbers of digits, cutoffs, leading zeros, and italics are all customizable.

Usage

```
format_p(
  x,
  digits = 3,
  pzero = FALSE,
  label = "p",
  italics = TRUE,
  type = "md"
)
```

Arguments

x	Number representing p-value
digits	Number of digits after the decimal for p-values, ranging between 1-5 (also controls cutoff for small p-values)
pzero	Logical value (default = FALSE) for whether to include leading zero for p-values
label	Character string for label before p value. Default is p. Set `label = ""` to return just the formatted p value with no label or operator (=, <, >)
italics	Logical value (default = TRUE) for whether label should be italicized (p)
type	Type of formatting ("md" = markdown, "latex" = LaTeX)

Value

A character string that includes *p* and then the p-value formatted in Markdown or LaTeX. If p-value is below `digits` cutoff, `p < cutoff` is used.

Examples

```
# Format p-value
format_p(0.001)

# Format p-value vector
format_p(c(0.001, 0.01))

# Round digits for p-values greater than cutoff
format_p(0.111, digits = 2)

# Default cutoff is p < 0.001
format_p(0.0001)

# Set cutoff with digits
format_p(0.0001, digits = 2)

# Include leading zero
format_p(0.001, pzero = TRUE)

# Return only Bayes factor value (no label)
format_p(0.001, label = ")

# Format for LaTeX
format_p(0.001, type = "latex")
```

---

format_scientific	*Format numbers in scientific notation*

Description

Format numbers in scientific notation

Usage

```
format_scientific(x, digits = 1, type = "md")
```

Arguments

x	Number
digits	Number of digits after the decimal
type	Type of formatting ("md" = markdown, "latex" = LaTeX)

Value

A character string of a number in scientific notation formatted in Markdown or LaTeX.

Examples

```
format_scientific(1111)
# Control number of digits after decimal with digits
format_scientific(1111, digits = 3)
```

---

format_stats   *Format statistical results*

Description

A generic function that takes objects from various statistical methods to create formatted character strings to insert into R Markdown or Quarto documents. Currently, the generic function works with the following objects:

1. htest objects of correlations, t-tests, and Wilcoxon tests

2. correlations from the {correlation} package.

3. Bayes factors from the {BayesFactor} package. The function invokes specific methods that depend on the class of the first argument.

Usage

```
format_stats(x, ...)
```

Arguments

x   Statistical object.

...   Additional arguments passed to methods. For method-specific arguments, see
                 format_stats.htest() for htest correlations, t-tests, and Wilcoxon tests and
                 format_stats.BFBayesFactor() for Bayes factors from the {BayesFactor}
                 package.

Value

A character string of statistical information formatted in Markdown or LaTeX.

See Also

Other functions for printing statistical objects: format_bf(), format_corr(), format_stats.BFBayesFactor(), format_stats.easycorrelation(), format_stats.htest(), format_ttest()

Examples

```
# Format cor.test() object
format_stats(cor.test(mtcars$mpg, mtcars$cyl))

# Format correlation::correlation() object
format_stats(correlation::correlation(data = mtcars, select = "mpg", select2 = "cyl"))

# Format t.test() object
format_stats(t.test(mtcars$vs, mtcars$am))

# Format BFBayesFactor object from {BayesFactor} package
format_stats(BayesFactor::ttestBF(mtcars$vs, mtcars$am))
```

```
format_stats.BFBayesFactor
```
Format Bayes factors

Description

This method formats Bayes factors from the {BayesFactor} package. By default, this function rounds Bayes factors greater than 1 to one decimal place and Bayes factors less than 1 to two decimal places. Values greater than 1000 or less than 1/1000 are formatted using scientific notation. Cutoffs can be set that format the values as greater than or less than the cutoffs (e.g., BF > 1000 or BF < 0.001). Numbers of digits, cutoffs, italics, and label subscripts are all customizable.

Usage

```
## S3 method for class 'BFBayesFactor'
format_stats(
  x,
  digits1 = 1,
  digits2 = 2,
  cutoff = NULL,
  label = "BF",
  italics = TRUE,
  subscript = "10",
  type = "md",
  ...
)
```

Arguments

x	BayesFactor object or vector of numeric Bayes factor values
digits1	Number of digits after the decimal for Bayes factors > 1
digits2	Number of digits after the decimal for Bayes factors < 1
cutoff	Cutoff for using _BF_~10~ > <cutoff> or _BF_~10~ < 1 / <cutoff> (value must be > 1)

label	Character string for label before Bayes factor. Default is BF. Set `label = ""` to return just the formatted Bayes factor value with no label or operator (=, <, >)
`italics`	Logical value (default = TRUE) for whether label should be italicized (*BF* or BF)
`subscript`	Subscript to include with *BF* label (`"10"`, `"01"`, or `""` for no subscript)
`type`	Type of formatting (`"md"` = markdown, `"latex"` = LaTeX)
`...`	Additional arguments passed to methods.

Value

A character string of statistical information formatted in Markdown or LaTeX.

See Also

Other functions for printing statistical objects: `format_bf()`, `format_corr()`, `format_stats()`, `format_stats.easycorrelation()`, `format_stats.htest()`, `format_ttest()`

Examples

```
# Prepare statistical object
test_bf <- BayesFactor::ttestBF(mtcars$vs, mtcars$am)

# Format Bayes factor
format_stats(test_bf)

# Control cutoff for output
format_stats(test_bf, cutoff = 3)

# Change digits, remove italics and subscript
format_stats(test_bf, digits2 = 1, italics = FALSE, subscript = "")

# Return only Bayes factor value (no label)
format_stats(test_bf, label = "")

# Format for LaTeX
format_stats(test_bf, type = "latex")
```

```
format_stats.easycorrelation
```
                              Format correlation statistics

Description

This functions formats correlation statistics generated from the {correlation} package. This detects whether the object is from a Pearson, Spearman, or Kendall correlation and reports the appropriate correlation label (r, τ, ρ). The default output is APA formatted, but numbers of digits, leading zeros, the presence of confidence intervals, and italics are all customizable.

Usage

```
## S3 method for class 'easycorrelation'
format_stats(
  x,
  digits = 2,
  pdigits = 3,
  pzero = FALSE,
  full = TRUE,
  italics = TRUE,
  type = "md",
  ...
)
```

Arguments

x	An `htest` object
digits	Number of digits after the decimal for means, confidence intervals, and test statistics
pdigits	Number of digits after the decimal for p-values, ranging between 1-5 (also controls cutoff for small p-values)
pzero	Logical value (default = FALSE) for whether to include leading zero for p-values
full	Logical value (default = TRUE) for whether to include means and confidence intervals or just test statistic and p-value
italics	Logical value (default = TRUE) for whether *p* label should be italicized
type	Type of formatting ("md" = markdown, "latex" = LaTeX)
...	Additional arguments passed to methods.

Value

A character string of statistical information formatted in Markdown or LaTeX.

See Also

Other functions for printing statistical objects: [format_bf()](), [format_corr()](), [format_stats()](),
[format_stats.BFBayesFactor()](), [format_stats.htest()](), [format_ttest()]()

Examples

```
# Prepare statistical objects
test_corr <- correlation::correlation(mtcars, select = "mpg", select2 = "disp")
test_corr2 <- correlation::correlation(mtcars, select = "mpg", select2 = "disp", method = "kendall")

# Format correlation
format_stats(test_corr)

# Remove confidence intervals and italics
format_stats(test_corr, full = FALSE, italics = FALSE)
```

```
# Change digits and add leading zero to p-value
format_stats(test_corr, digits = 3, pdigits = 4, pzero = TRUE)

# Format Kendall's tau for LaTeX
format_stats(test_corr2, type = "latex")
```

---

format_stats.htest   *Format hypothesis test statistics*

Description

This method formats hypothesis test statistics from the class htest. Currently, this includes correlations from cor.test() and t-tests and Wilcoxon tests from t.test() and wilcox.test(). For correlations, the function detects whether the object is from a Pearson, Spearman, or Kendall correlation and reports the appropriate correlation label (r, τ, ρ). The default output is APA formatted, but this function allows control over numbers of digits, leading zeros, the presence of means and confidence intervals, italics, degrees of freedom, and mean labels, and output format of Markdown or LaTeX.

Usage

```
## S3 method for class 'htest'
format_stats(
  x,
  digits = NULL,
  pdigits = 3,
  pzero = FALSE,
  full = TRUE,
  italics = TRUE,
  dfs = "par",
  mean = "abbr",
  type = "md",
  ...
)
```

Arguments

x	An htest object
digits	Number of digits after the decimal for means, confidence intervals, and test statistics
pdigits	Number of digits after the decimal for p-values, ranging between 1-5 (also controls cutoff for small p-values)
pzero	Logical value (default = FALSE) for whether to include leading zero for p-values
full	Logical value (default = TRUE) for whether to include means and confidence intervals or just test statistic and p-value

italics	Logical value (default = TRUE) for whether *p* label should be italicized
dfs	Formatting for degrees of freedom ("par" = parenthetical, "sub" = subscript, "none" = do not print degrees of freedom)
mean	Formatting for mean label ("abbr" = M, "word" = Mean)
type	Type of formatting ("md" = markdown, "latex" = LaTeX)
...	Additional arguments passed to methods.

Value

A character string of statistical information formatted in Markdown or LaTeX.

See Also

Other functions for printing statistical objects: `format_bf`(), `format_corr`(), `format_stats`(), `format_stats.BFBayesFactor`(), `format_stats.easycorrelation`(), `format_ttest`()

Examples

```
# Prepare statistical objects
test_corr <- cor.test(mtcars$mpg, mtcars$cyl)
test_corr2 <- cor.test(mtcars$mpg, mtcars$cyl, method = "kendall")
test_ttest <- t.test(mtcars$vs, mtcars$am)
test_ttest2 <- wilcox.test(mtcars$vs, mtcars$am)

# Format correlation
format_stats(test_corr)

# Remove confidence intervals and italics
format_stats(test_corr, full = FALSE, italics = FALSE)

# Change digits and add leading zero to p-value
format_stats(test_corr, digits = 3, pdigits = 4, pzero = TRUE)

# Format Kendall's tau
format_stats(test_corr2)

# Format t-test
  format_stats(test_ttest)

# Remove mean and confidence interval
format_stats(test_ttest, full = FALSE)

# Remove degrees of freedom and spell out "Mean"
format_stats(test_ttest, dfs = "none", mean = "word")

# Format for LaTeX
format_stats(test_ttest2, type = "latex")
```

---

format_sub   *Format subscript text*

Description

Format subscript text

Usage

```
format_sub(subscript = NULL, type = "md")
```

Arguments

subscript	Character string or NULL
type	Type of formatting ("md" = markdown, "latex" = LaTeX)

Value

A character string that is formatted as subscript for either Markdown or LaTeX.

Examples

```
format_sub("Hello world!")
# Format in LaTeX syntax
format_sub("Hello world!", type = "latex")
```

---

format_summary   *Calculate and format summary statistics of central tendency and error*

Description

format_summary() is a general function that allows you to either automatically calculate mean/median and a measure of error from a data vector or specify already calculated a mean/median and either an error interval or error limits. Error measures include confidence intervals, standard deviation, and standard error of the mean. Each of those has a specific function that formats means and those error measures using APA (7th edition) style. So format_meanci(), format_meansd(), format_meanse(), and format_medianiqr() are wrappers around format_summary() for specific error measures with a default style. To just format the mean or median with no error, use format_mean() or format_median(). All measures ignore NAs.

Usage

```
format_summary(
  x = NULL,
  tendency = "mean",
  error = "ci",
  values = NULL,
  digits = 1,
  tendlabel = "abbr",
  italics = TRUE,
  subscript = NULL,
  units = NULL,
  display = "limits",
  cilevel = 0.95,
  errorlabel = TRUE,
  type = "md"
)

format_mean(
  x = NULL,
  tendency = "mean",
  values = NULL,
  digits = 1,
  tendlabel = "abbr",
  italics = TRUE,
  subscript = NULL,
  units = NULL,
  display = "none",
  type = "md"
)

format_meanci(
  x = NULL,
  tendency = "mean",
  error = "ci",
  values = NULL,
  digits = 1,
  tendlabel = "abbr",
  italics = TRUE,
  subscript = NULL,
  units = NULL,
  display = "limits",
  cilevel = 0.95,
  errorlabel = TRUE,
  type = "md"
)

format_meanse(
  x = NULL,
```

```
  tendency = "mean",
  error = "se",
  values = NULL,
  digits = 1,
  tendlabel = "abbr",
  italics = TRUE,
  subscript = NULL,
  units = NULL,
  display = "par",
  errorlabel = TRUE,
  type = "md"
)

format_meansd(
  x = NULL,
  tendency = "mean",
  error = "sd",
  values = NULL,
  digits = 1,
  tendlabel = "abbr",
  italics = TRUE,
  subscript = NULL,
  units = NULL,
  display = "par",
  errorlabel = TRUE,
  type = "md"
)

format_median(
  x = NULL,
  tendency = "median",
  values = NULL,
  digits = 1,
  tendlabel = "abbr",
  italics = TRUE,
  subscript = NULL,
  units = NULL,
  display = "none",
  type = "md"
)

format_medianiqr(
  x = NULL,
  tendency = "median",
  error = "iqr",
  values = NULL,
  digits = 1,
  tendlabel = "abbr",
```

```
    italics = TRUE,
    subscript = NULL,
    units = NULL,
    display = "par",
    errorlabel = TRUE,
    type = "md"
)
```

Arguments

x	Numeric vector of data to calculate mean and error
tendency	Character vector specifying measure of central tendency ("mean" = mean, "median" = median)
error	Character vector specifying error type ("ci" = confidence interval, "se" = standard error of the mean, "sd" = standard deviation, "iqr" = interquartile range)
values	Numeric vector of mean and interval or mean and lower and upper limits
digits	Number of digits after the decimal for means and error
tendlabel	Formatting for tendency label ("abbr" = M, "word" = Mean, "none" = no label)
italics	Logical value (default = TRUE) for whether mean label should be italicized
subscript	Character string to include as subscript with mean label
units	Character string that gives units to include after mean value
display	Character vector specifying how to display error ("limits" = [lower limit, upper limit], "pm" = ±interval, "par" = (interval), "none" = do not display error)
cilevel	Numeric scalar from 0-1 defining confidence level (defaults to 0.95)
errorlabel	Logical value (default = TRUE) for whether error label (e.g., 95% CI) should be included
type	Type of formatting ("md" = markdown, "latex" = LaTeX)

Value

A character string of mean and error formatted in Markdown or LaTeX. To return only the mean (no error), set `display = "none"`.

Examples

```
# Print mean and 95% confidence limits for fuel efficiency
format_meanci(mtcars$mpg)

# Print mean and standard deviation
format_meansd(mtcars$mpg)

# Print mean and standard error of the mean
format_meanse(mtcars$mpg)

# Print mean
format_mean(mtcars$mpg)
```

```
# Print mean and 95% confidence limits with no label for "95% CI"
format_meanci(mtcars$mpg, errorlabel = FALSE)

# Print mean and standard error of the mean as plus/minus interval
format_meanse(mtcars$mpg, error = "se", display = "pm")

# Print mean and 90% confidence limits with units
format_meanci(mtcars$mpg, units = "cm", cilevel = 0.9)

# Print three-digit mean with subscript in LaTeX
format_summary(mtcars$mpg, digits = 3, subscript = "control", display = "none", type = "latex")
```

---

format_ttest   *Format t-test statistics*

Description

[Superseded]

With format_ttest() you can format t-tests generated from t.test() and wilcox.test() output. This is now an internal function superceded by format_stats(), which we recommend using instead.

Usage

```
format_ttest(x, digits, pdigits, pzero, full, italics, dfs, mean, type)
```

Arguments

x	An htest object
digits	Number of digits after the decimal for means, confidence intervals, and test statistics
pdigits	Number of digits after the decimal for p-values, ranging between 1-5 (also controls cutoff for small p-values)
pzero	Logical value (default = FALSE) for whether to include leading zero for p-values
full	Logical value (default = TRUE) for whether to include means and confidence intervals or just test statistic and p-value
italics	Logical value (default = TRUE) for whether *p* label should be italicized
dfs	Formatting for degrees of freedom ("par" = parenthetical, "sub" = subscript, "none" = do not print degrees of freedom)
mean	Formatting for mean label ("abbr" = M, "word" = Mean)
type	Type of formatting ("md" = markdown, "latex" = LaTeX)

Value

A character string of statistical information formatted in Markdown or LaTeX.

See Also

Other functions for printing statistical objects: format_bf(), format_corr(), format_stats(),
format_stats.BFBayesFactor(), format_stats.easycorrelation(), format_stats.htest()

Examples

```
format_stats(t.test(formula = mtcars$mpg ~ mtcars$vs))
```

Index