

Package: pubformat (via r-universe)

October 3, 2026

Title Publication-Ready Formatting for Statistical Results

Version 0.1.0.9000

Description Provides tools for converting statistical results into consistent, publication-ready character output. Current functionality includes formatting p-values, confidence intervals, and correlation coefficients using publication-friendly conventions. The package emphasizes consistent reporting while preserving the underlying statistical results and statistical decision-making.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

URL <https://github.com/Reynolds826/pubformat>

BugReports <https://github.com/Reynolds826/pubformat/issues>

VignetteBuilder knitr

Repository <https://reynolds826.r-universe.dev>

Date/Publication 2026-10-02 13:15:01 UTC

RemoteUrl <https://github.com/reynolds826/pubformat>

RemoteRef HEAD

RemoteSha 96a257750b9048c7ad550dddf7ac1202f8b683bd

Contents

format_ci	2
format_cor	3
format_p	4
format_percent	5
format_r	6

Index**8**

format_ci	<i>Format confidence intervals for publication</i>
-----------	--

Description

Formats lower and upper confidence interval limits using publication-friendly conventions. By default, confidence intervals are displayed using APA-style square brackets.

Usage

```
format_ci(
  lower,
  upper,
  level = 0.95,
  digits = 2,
  prefix = TRUE,
  leading_zero = TRUE
)
```

Arguments

lower	Numeric vector containing lower confidence limits.
upper	Numeric vector containing upper confidence limits.
level	A single numeric value between 0 and 1 specifying the confidence level. Default is 0.95.
digits	A single non-negative integer specifying the number of decimal places to display. Default is 2.
prefix	Logical. If TRUE, the confidence level and "CI" are included. Default is TRUE.
leading_zero	Logical. If TRUE, values between -1 and 1 retain a leading zero. Default is TRUE.

Details

Confidence intervals are formatted using square brackets with the lower and upper limits separated by a comma.

The function does not infer the type of statistic being reported. For statistics that cannot exceed 1 in absolute value, such as correlations, `leading_zero = FALSE` can be used.

Value

A character vector containing formatted confidence intervals.

Examples

```
format_ci(1.08, 1.87)
format_ci(-0.42, 0.18)
format_ci(0.21, 0.48, leading_zero = FALSE)
format_ci(c(1.08, 2.15), c(1.87, 3.02))
```

format_cor

*Format correlation coefficients for publication***Description**

Formats Pearson, Spearman, and Kendall correlation coefficients using publication-friendly conventions. The appropriate statistical symbol is selected automatically from the requested correlation method.

Usage

```
format_cor(
  r,
  method = "pearson",
  digits = 2,
  prefix = TRUE,
  leading_zero = FALSE
)
```

Arguments

<code>r</code>	Numeric vector containing correlation coefficients. Values must be between -1 and 1.
<code>method</code>	Character string specifying the correlation method. Must be one of "pearson", "spearman", or "kendall". Default is "pearson".
<code>digits</code>	A single non-negative integer specifying the number of decimal places to display. Default is 2.
<code>prefix</code>	Logical. If TRUE, the method-appropriate statistical symbol and equals sign are included. Default is TRUE.
<code>leading_zero</code>	Logical. If FALSE, leading zeros are omitted for values between -1 and 1. Default is FALSE.

Details

`format_cor()` supports three correlation methods:

- Pearson: r
- Spearman: ρ
- Kendall: τ

The function formats correlation coefficients only. It does not calculate correlations, determine statistical significance, or format associated p-values.

Missing values are returned as NA.

Value

A character vector containing formatted correlation coefficients.

Examples

```
format_cor(0.42)
format_cor(0.42, method = "pearson")
format_cor(0.42, method = "spearman")
format_cor(0.42, method = "kendall")
format_cor(c(0.42, -0.31, 0), method = "spearman")
```

format_p

Format p-values for publication

Description

Formats numeric p-values using publication-friendly conventions. Values are displayed with a fixed number of decimal places, leading zeros are omitted, and very small p-values are reported using less-than notation.

Usage

```
format_p(pval, digits = 3, threshold = NULL, sig = FALSE, prefix = TRUE)
```

Arguments

pval	Numeric vector containing p-values.
digits	A single non-negative integer specifying the number of decimal places to display. Default is 3.
threshold	Optional numeric reporting threshold. Values strictly below this threshold are displayed using less-than notation. If NULL (the default), the threshold is automatically set to $10^{-\text{digits}}$. For example, digits = 3 produces a threshold of .001.
sig	Logical. If TRUE, conventional significance codes are appended. Default is FALSE.
prefix	Logical. If TRUE, "p" is included before the formatted value. Default is TRUE.

Details

The reporting threshold controls only how a p-value is displayed. It does not represent an alpha level or statistical decision rule.

With digits = 3:

$p = .001$ is reported as " $p = .001$ " $p = .0009$ is reported as " $p < .001$ "

Significance codes use the conventional strict thresholds:

*** $p < .001$ ** $p < .01$

- $p < .05$. $p < .10$ ns otherwise

Value

A character vector containing formatted p-values.

Examples

```
format_p(0.048)
format_p(0.0002)
format_p(0.001)
format_p(0.048, sig = TRUE)
format_p(c(0.048, 0.006, 0.0002, 0.213))
```

format_percent	<i>Format percentages for publication</i>
----------------	---

Description

Formats proportions or percentages using publication-friendly conventions. By default, input values are interpreted as proportions and converted to percentages.

Usage

```
format_percent(x, digits = 1, input = "proportion", symbol = TRUE)
```

Arguments

x	Numeric vector containing proportions or percentages.
digits	A single non-negative integer specifying the number of decimal places to display. Default is 1.
input	Character string specifying how x should be interpreted. Must be either "proportion" or "percent". Default is "proportion".
symbol	Logical. If TRUE, the percent symbol is appended to each value. Default is TRUE.

Details

When input = "proportion", values must fall between 0 and 1 and are multiplied by 100 before formatting.

When input = "percent", values are assumed to already be expressed as percentages and are not rescaled.

Missing values are returned as NA.

Value

A character vector containing formatted percentages.

Examples

```
format_percent(0.42)
format_percent(0.423, digits = 2)
format_percent(42, input = "percent")
format_percent(c(0.25, 0.50, 0.75))
```

format_r	<i>Format correlation coefficients for publication</i>
----------	--

Description

Formats correlation coefficients using publication-friendly conventions. By default, values are displayed with two decimal places and leading zeros are omitted, consistent with common APA-style reporting for statistics that cannot exceed 1 in absolute value.

Usage

```
format_r(r, digits = 2, prefix = TRUE, leading_zero = FALSE)
```

Arguments

r	Numeric vector containing correlation coefficients. Values must be between -1 and 1.
digits	A single non-negative integer specifying the number of decimal places to display. Default is 2.
prefix	Logical. If TRUE, "r = " is included before the formatted value. Default is TRUE.
leading_zero	Logical. If FALSE, leading zeros are omitted for values between -1 and 1. Default is FALSE.

Details

`format_r()` formats correlation coefficients only. It does not determine statistical significance or format associated p-values.

Missing values are returned as NA.

Value

A character vector containing formatted correlation coefficients.

Examples

```
format_r(0.42)
format_r(-0.31)
format_r(c(0.42, -0.31, 0))
format_r(0.42, leading_zero = TRUE)
```

Index

`format_ci`, [2](#)
`format_cor`, [3](#)
`format_p`, [4](#)
`format_percent`, [5](#)
`format_r`, [6](#)