Package 'journalR'

December 14, 2025

Type Package

Title Formatting Tools for Scientific Journal Writing

Version 0.2.1

Maintainer Sam Byrne <ssbyrne@uw.edu>

Description Scientific journal numeric formatting policies implemented in code.
Emphasis on formatting mean/upper/lower sets of values. Convert raw numeric
triplet value vectors to formatted text for journal submission. For example
c(2e6, 1e6, 3e6) becomes ``2.00 million (1.00--3.00)".
Lancet and Nature have built-in styles for rounding and punctuation marks.
Users may extend journal styles arbitrarily.
Three metrics are supported; proportions, percentage points, and counts.
Magnitudes for all metrics are discovered automatically.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.2.0)

Imports checkmate, data.table, glue

Suggests devtools (>= 2.4.5), testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 7.3.2

URL https://github.com/epi-sam/journalR

BugReports https://github.com/epi-sam/journalR/issues

NeedsCompilation no

Author Sam Byrne [aut, cre, cph] (ORCID:
<https://orcid.org/0009-0008-1067-307X>)

Repository CRAN

Date/Publication 2025-12-14 17:00:07 UTC

Contents

`en_dash` *En-dash*

Description

Standard for "x – y" and Lancet negative: ("\U2013")

Usage

```
en_dash()
```

Value

[chr] en-dash character

See Also

Other marks: `mid_dot()`, `thin_space()`

Examples

```
en_dash()
```

`fmt_magnitude` *Format magnitude*

Description

Format a numeric vector into a string with specified magnitude (billion, million, thousand).

Usage

```
fmt_magnitude(
  x,
  digits = 1,
  nsmall = 1,
  decimal.mark = ".",
  mag = NULL,
  label_thousands = FALSE
)
```

Arguments

`x`	[num] numeric vector
`digits`	[int: default 1L] passed to 'round()'
`nsmall`	[int: default 1L] passed to 'format()'
`decimal.mark`	[chr: default "."] decimal mark passed to 'format()'
`mag`	[chr c("b", "m", "t")] magnitude (billion, million, thousand) passed to set_magnitude()
`label_thousands`	[lgl: default FALSE] allow thousands magnitude? Not Lancet-valid. Passed to 'set_magnitude()'

Details

Unaware of schema, just a hard-coded git-er-done function.

Caution - thousands magnitude is not Lancet compliant.

Value

[chr] formatted string

See Also

Other vector_formats: `format_oxford_comma()`, `fround()`, `fround_count()`, `fround_dtype()`, `fround_dtype_lancet()`, `fround_props()`

Other magnitudes: `set_magnitude()`

Examples

```
fmt_magnitude(123456789)
```

`format_journal_clu` *Format central, lower, upper value triplets for journal presentation*

Description

Defaults are generic. This function allows special formtting marks to be applied by journal. Use ‘format_lancet_clu()‘ for Lancet-specific formatting. Use ‘format_nature_clu()‘ for Nature-specific formatting.

Usage

```
format_journal_clu(central, lower, upper, d_type, style_name = "nature")
```

Arguments

`central`	[num] central, point_estimate value vector
`lower`	[num] lower bound vector
`upper`	[num] upper bound vector
`d_type`	[chr c(prop, pp, count)] data type - proportion, percentage
`style_name`	[chr: default 'nature'] style name - controls rounding and formatting.

Details

Takes three vectors as main arguments for data.table-friendly vectorization.

‘central‘ could be mean, median, point_estimate

Transform c(central = 0.994, lower = 0.984, upper = 0.998) to "99.4 (98.4–99.8)"

Accounts for negative values, and UIs that cross zero. Checks if central, lower, upper values are in the correct order.

Value

[chr] formatted string vector

See Also

Other styled_formats: `format_journal_df()`, `format_lancet_clu()`, `format_lancet_df()`, `format_means_df()`, `format_nature_clu()`, `format_nature_df()`, `fround_clu_triplet()`, `new_style()`

Examples

```
format_journal_clu(
 central = c(0.994, -0.994)
 , lower = c(0.984, -0.998)
 , upper = c(0.998, -0.984)
 , d_type = "prop"
)
```

`format_journal_df` *Return a table with formatted central, lower, upper*

Description

Assumes a single data-type (d_type) for the whole table (e.g. 'prop', 'pp', 'count')

Usage

```
format_journal_df(
  df,
  d_type,
  new_var = "clu_fmt",
  style_name = "nature",
  central_var = "mean",
  lower_var = "lower",
  upper_var = "upper",
  remove_clu_columns = TRUE
)
```

Arguments

`df`	[data.frame, data.table]
`d_type`	[chr c('prop', 'pp', or 'count')] a single data type
`new_var`	[chr: default 'clu_fmt'] name of new formatted column
`style_name`	[chr: default 'nature'] style name - controls rounding and formatting.
`central_var`	[chr: default 'mean'] name of central tendency variable
`lower_var`	[chr: default 'lower'] name of lower bound variable
`upper_var`	[chr: default 'upper'] name of upper bound variable
`remove_clu_columns`	[lgl: default TRUE] remove central, lower, upper variables after formatting?

Value

[data.frame] data.frame, data.table with new 'clu_fmt' column

See Also

Other styled_formats: format_journal_clu(), format_lancet_clu(), format_lancet_df(), format_means_df(), format_nature_clu(), format_nature_df(), fround_clu_triplet(), new_style()

Examples

```
df <- data.frame(
 location_id = c(1, 2, 3)
 , mean = c(0.1234, 0, -0.3456)
 , lower = c(0.1134, -0.2245, -0.4445)
 , upper = c(0.1334, 0.2445, 0.3556)
)
format_journal_df(df, d_type = "prop")

DF <- data.frame(
 location_id = c(1, 2, 3)
 , mean = c(0.1234, 0, -0.3456)
 , lower = c(0.1134, -0.2245, -0.4445)
 , upper = c(0.1334, 0.2445, 0.3556)
)
format_journal_df(DF, d_type = "prop")
```

---

format_lancet_clu *Format central, lower, upper value triplets for Lancet journal presentation*

Description

Format central, lower, upper value triplets for Lancet journal presentation

Usage

```
format_lancet_clu(central, lower, upper, d_type)
```

Arguments

central	[num] central, point_estimate value vector
lower	[num] lower bound vector
upper	[num] upper bound vector
d_type	[chr c(prop, pp, count)] data type - proportion, percentage

Value

[chr] formatted string vector

See Also

Other styled_formats: format_journal_clu(), format_journal_df(), format_lancet_df(), format_means_df(), format_nature_clu(), format_nature_df(), fround_clu_triplet(), new_style()

Examples

```
format_lancet_clu(
   central  = c(0.994, -0.994)
   , lower  = c(0.984, -0.998)
   , upper  = c(0.998, -0.984)
   , d_type = "prop"
)
```

`format_lancet_df` *Return a table with formatted central, lower, upper for Lancet journal*

Description

Assumes a single data-type (d_type) for the whole table (e.g. 'prop', 'pp', 'count')

Usage

```
format_lancet_df(
  df,
  d_type,
  new_var = "clu_fmt",
  central_var = "mean",
  lower_var = "lower",
  upper_var = "upper",
  remove_clu_columns = TRUE
)
```

Arguments

`df`	[data.table] with central, lower, upper columns
`d_type`	[chr c(prop', 'pp', 'count')] data type - proportion, percentage point or count
`new_var`	[chr: default 'clu_fmt'] name of new formatted column
`central_var`	[chr: default 'mean'] name of central tendency e.g. 'point_estimate'
`lower_var`	[chr: default 'lower']
`upper_var`	[chr: default 'upper']
`remove_clu_columns`	[lgl: default TRUE] remove central, lower, upper columns?

Value

[data.frame, data.table] with mean_95_UI_formatted column, and central, lower, upper columns removed (if specified)

See Also

Other styled_formats: `format_journal_clu()`, `format_journal_df()`, `format_lancet_clu()`, `format_means_df()`, `format_nature_clu()`, `format_nature_df()`, `fround_clu_triplet()`, `new_style()`

Examples

```
df <- data.frame(
   location_did    = 1
   , location_name = "Global"
   , me_name       = "vacc_dpt1"
   , mean          = 55.8e6
   , lower         = 50.7e6
   , upper         = 60.7e6
)
format_lancet_df(df = df, d_type = "count", central_var = 'mean')
```

`format_means_df` *Format multiple data.frame 'mean_*' columns for presentation (by data type).*

Description

Format one or more 'mean_' columns by magnitude, data_type, and style.

Usage

```
format_means_df(df, d_type, central_var = "mean", style_name = "nature")
```

Arguments

`df`	[data.table] input data.table with one or more 'mean_' columns
`d_type`	[chr c('prop', 'pp', or 'count')] a single data type
`central_var`	[chr: default 'mean'] prefix of mean variable names to format. Implemented as e.g. "^mean[_]*" to capture 'mean', 'mean_1990', 'mean_2000', etc.
`style_name`	[chr: default 'nature'] style name - controls rounding and formatting.

Details

BEWARE: Does not have sophisticated count-type data handling like 'format_journal_clu()'. This is a simple formatter for multiple mean columns. Use with caution.

Value

[data.table] copy of input data.table with formatted mean column(s)

See Also

Other styled_formats: `format_journal_clu()`, `format_journal_df()`, `format_lancet_clu()`, `format_lancet_df()`, `format_nature_clu()`, `format_nature_df()`, `fround_clu_triplet()`, `new_style()`

Examples

```
df <- data.frame(
  location_id = c(1, 2, 3)
  , mean_1990 = c(100, 1e6, 1e9)
  , mean_2000 = c(200, 2e6, 2e-1)
 )
format_means_df(df, d_type = "count")
```

`format_nature_clu` *Format central, lower, upper value triplets for Nature journal presentation*

Description

Format central, lower, upper value triplets for Nature journal presentation

Usage

```
format_nature_clu(central, lower, upper, d_type)
```

Arguments

`central`	[num] central, point_estimate value vector
`lower`	[num] lower bound vector
`upper`	[num] upper bound vector
`d_type`	[chr c(prop, pp, count)] data type - proportion, percentage

Value

[chr] formatted string vector

See Also

Other styled_formats: `format_journal_clu()`, `format_journal_df()`, `format_lancet_clu()`, `format_lancet_df()`, `format_means_df()`, `format_nature_df()`, `fround_clu_triplet()`, `new_style()`

Examples

```
format_nature_clu(
   central  = c(0.994, -0.994)
   , lower  = c(0.984, -0.998)
   , upper  = c(0.998, -0.984)
   , d_type = "prop"
)
```

format_nature_df *Return a table with formatted central, lower, upper for Nature journal*

Description

Return a table with formatted central, lower, upper for Nature journal

Usage

```
format_nature_df(
  df,
  d_type,
  new_var = "clu_fmt",
  central_var = "mean",
  lower_var = "lower",
  upper_var = "upper",
  remove_clu_columns = TRUE
)
```

Arguments

df	[data.table]
d_type	[chr c('prop', 'pp', or 'count')] a single data type
new_var	[chr: default 'clu_fmt'] name of new formatted column
central_var	[chr: default 'mean'] name of central tendency variable
lower_var	[chr: default 'lower'] name of lower bound variable
upper_var	[chr: default 'upper'] name of upper bound variable
remove_clu_columns	[lgl: default TRUE] remove central, lower, upper variables after

Value

[data.table] copy of input data.table with new 'clu_fmt' column

See Also

Other styled_formats: format_journal_clu(), format_journal_df(), format_lancet_clu(), format_lancet_df(), format_means_df(), format_nature_clu(), fround_clu_triplet(), new_style()

Examples

```
df <- data.frame(
   location_did    = 1
   , location_name = "Global"
   , me_name       = "vacc_dpt1"
   , mean          = 55.8e6
```

```
   , lower         = 50.7e6
   , upper         = 60.7e6
)
format_nature_df(df = df, d_type = "count", central_var = 'mean')
```

`format_oxford_comma` *Format vector of items with Oxford comma*

Description

Format vector of items with Oxford comma

Usage

```
format_oxford_comma(vec, sep = "and")
```

Arguments

vec	[any] vector of items to format
sep	[chr: default "and"] separator before last item

Value

[chr] formatted string with Oxford comma

See Also

Other vector_formats: `fmt_magnitude()`, `fround()`, `fround_count()`, `fround_dtype()`, `fround_dtype_lancet()`, `fround_props()`

Examples

```
format_oxford_comma(1:2)
format_oxford_comma(1:3)
format_oxford_comma(1:3, "or")
```

fround *Format and round*

Description

Unaware of data-type or schema, just a hard-coded git-er-done function.

Usage

```
fround(x, digits = 1L, nsmall = 1L, decimal.mark = ".")
```

Arguments

x	[num] numeric vector
digits	[integer] passed to ‘round()‘
nsmall	[integer] passed to ‘format()‘
decimal.mark	[chr] passed to ‘format()‘

Value

[chr] formatted string

See Also

Other vector_formats: fmt_magnitude(), format_oxford_comma(), fround_count(), fround_dtype(), fround_dtype_lancet(), fround_props()

Examples

```
fround(0.123456789)
fround(0.123456789, digits = 3)
fround(0.123456789, digits = 3, nsmall = 4)
```

fround_dtype *Format and round with data-type suffix*

Description

Unaware of schema, just a hard-coded git-er-done function.

Usage

```
fround_dtype(x, d_type = "prop", digits = 1L, nsmall = 1L, decimal.mark = ".")
```

Arguments

x	[num] numeric value
d_type	[chr c('prop', 'pp', or 'count')] data type - proportion, percentage point or count
digits	[integer: default 1L] passed to 'round()'
nsmall	[integer: default 1L] passed to 'format()'
decimal.mark	[chr: default "."] decimal mark passed to 'format()'

Value

[chr] formatted string

See Also

Other vector_formats: `fmt_magnitude()`, `format_oxford_comma()`, `fround()`, `fround_count()`, `fround_dtype_lancet()`, `fround_props()`

Examples

```
fround_dtype(0.123456789)
fround_dtype(0.123456789, 'pp', 3, 4)
fround_dtype(c(55.8346, 123.456789), 'count', 3, 4, ".")
```

`fround_dtype_lancet` *Format and round with data-type suffix*

Description

Lancet-specific wrapper for 'fround_dtype()', using mid-dot as decimal mark.

Usage

```
fround_dtype_lancet(
  x,
  d_type = "prop",
  digits = 1L,
  nsmall = 1L,
  decimal.mark = mid_dot()
)
```

Arguments

x	[num] numeric value
d_type	[chr c('prop', 'pp', or 'count')] data type - proportion, percentage point or count
digits	[integer: default 1L] passed to 'round()'
nsmall	[integer: default 1L] passed to 'format()'
decimal.mark	[chr: default mid_dot()] decimal mark passed to 'format()'

Value

[chr] formatted string

See Also

Other vector_formats: `fmt_magnitude()`, `format_oxford_comma()`, `fround()`, `fround_count()`, `fround_dtype()`, `fround_props()`

Examples

```
fround_dtype_lancet(0.123456789)
fround_dtype_lancet(0.123456789, 'pp', 3, 4)
fround_dtype_lancet(c(55.8346, 123.456789), 'count', 3, 4, ".")
```

---

get_data_types *Get data types*

Description

Centrally managed definition for all allowed data types.

Usage

```
get_data_types()
```

Value

[chr] vector of allowed data types

See Also

Other data_types: `get_data_type_labels()`

Examples

```
get_data_types()
```

get_data_type_labels *Get data type labels*

Description

Centrally managed definition for all data type labels.

Usage

```
get_data_type_labels(d_type)
```

Arguments

d_type	[chr]

Value

[list] named list of data type labels

See Also

Other data_types: `get_data_types()`

Examples

```
get_data_type_labels('prop')
```

get_style *Get a style from the styles dictionary*

Description

Accessor function to retrieve a style from the package's styles dictionary.

Usage

```
get_style(style_name)
```

Arguments

style_name	[chr] name of the style to retrieve

Value

[list] the requested style as a named list

See Also

Other styles: assert_style_schema(), get_style_schema(), new_style(), set_style(), style_lancet(), style_nature()

Examples

```
get_style("lancet")
```

`get_style_schema` *Get style schema*

Description

Centrally managed definition for all required journal format styles.

Usage

```
get_style_schema()
```

Value

[list] named list of style elements and their expected types

See Also

Other styles: assert_style_schema(), get_style(), new_style(), set_style(), style_lancet(), style_nature()

Examples

```
get_style_schema()
```

`mid_dot` *Mid-dot*

Description

Lancet numeric decimal standard: "·" ("\U00B7")

Usage

```
mid_dot()
```

Value

[chr] mid-dot character

See Also

Other marks: `en_dash()`, `thin_space()`

Examples

```
mid_dot()
```

`new_style` *Make a new style by args*

Description

Wrapper function to create and set a new style in one step.

Usage

```
new_style(
  style_name,
  prop_digits_round = 1,
  prop_nsmall = 1,
  count_method = "sigfig",
  count_digits_sigfig = 3,
  count_pad_sigfigs = TRUE,
  count_nsmall = 1,
  count_big.mark = ",",
  decimal.mark = ".",
  neg_mark_mean = "-",
  neg_mark_UI = "-",
  UI_only = FALSE,
  UI_text = "",
  assert_clu_order = TRUE,
  is_lancet = FALSE,
  label_thousands = FALSE,
  round_5_up = TRUE
)
```

Arguments

`style_name`	[chr] name of the style to set
`prop_digits_round`	[int: default 1] number of digits to round proportions to
`prop_nsmall`	[int: default 1] minimum number of digits to the right of the decimal point - proportions
`count_method`	[chr: c("sigfig", "decimal", "int")] choose how to report counts - prioritize sigfigs across mean/lower/upper, hard-set decimals, or leave numbers in integer space.

`count_digits_sigfig`
: [int: default 3] number of significant figures for counts

`count_pad_sigfigs`
: [lgl: default TRUE] signif(5.00, 3) is "5" - do you want to pad the trailing 0s back on - usually TRUE?

`count_nsmall`
: [int: default 1] passed to ‘format()‘ if ‘count_method‘ == ’decimal’

`count_big.mark`
: [chr: default ","] character to use for counts thousand, million, billion separator e.g. ","

`decimal.mark`
: [chr: default "."] decimal mark e.g. "." or ‘mid_dot()‘ for Lancet.

`neg_mark_mean`
: [chr: default "-"] string to describe central value negatives - e.g. "-1 (-2 to 4)" could become "Negtive 1 (-2 to 4)"

`neg_mark_UI`
: [chr: default "-"] string to describe negative sign in UI brackets e.g. "1 (-2 to 4)" could become "1 (–2 to 4)" (en-dash)

`UI_only`
: [lgl: default FALSE] Return only UI from ‘format_journal_df()‘ family functions?

`UI_text`
: [chr: default ""] Text to appear inside UI brackets before numbers e.g. "2 (1 – 4)" could become "2 (95%UI 1 – 4)"

`assert_clu_order`
: [lgl: default TRUE] whether to assert CLU relationships (ensure lower < central < upper)

`is_lancet`
: [lgl: default FALSE] TRUE to handle edge-case Lancet count formatting policies

`label_thousands`
: [lgl: default FALSE] whether format counts as e.g. 10,000 as ’10 thousand’

`round_5_up`
: [lgl: default TRUE] In R, ‘round(1245, 3)‘ is "1240". Do you want to round to "1250" instead? Default TRUE to conform with common expectations.

Value

[chr] invisible vector of input objects

See Also

Other styles: `assert_style_schema()`, `get_style()`, `get_style_schema()`, `set_style()`, `style_lancet()`, `style_nature()`

Other styled_formats: `format_journal_clu()`, `format_journal_df()`, `format_lancet_clu()`, `format_lancet_df()`, `format_means_df()`, `format_nature_clu()`, `format_nature_df()`, `fround_clu_triplet()`

Examples

```
new_style(style_name = "my_style")
```

set_magnitude *Define magnitude, magnitude label and denominator for a vector of numeric values.*

Description

Support function used on _central_ (e.g. mean) values for later formatting functions.

Usage

```
set_magnitude(x, mag = NULL, label_thousands = FALSE, verbose = TRUE)
```

Arguments

x	[num] numeric vector
mag	[chr: default NULL c(NULL, "T", "B", "M")] NULL (auto-detect), otherwise user-override (not recommended) - (M)illions or (B)illions or (T)housands (thousands are not Lancet-valid)
label_thousands	[lgl: default FALSE] allow (T)housands magnitude? Not Lancet-valid.
verbose	[lgl: default TRUE] warn if label_thousands is TRUE

Details

'Thousands' label is not a Lancet-valid, which uses ddd<narrow-space>ddd format. See 'fround_count()' for details.

Value

[data.frame] with vector elements: mag, mag_label, and denom Each vector element has one item per length(x)

See Also

[fround_count()]

Other magnitudes: `fmt_magnitude()`

Examples

```
set_magnitude(c(1e-6, 1, 1e3, 1e6, 1e9))
```

`set_style` *Set a new style by list*

Description

Set a new style by list

Usage

```
set_style(style_name, style_entry)
```

Arguments

`style_name`	[chr] name of the style to set
`style_entry`	[list] named list representing the style entry

Value

[chr] invisible vector of input objects, to allow easier un-locking

See Also

Other styles: `assert_style_schema()`, `get_style()`, `get_style_schema()`, `new_style()`, `style_lancet()`, `style_nature()`

Examples

```
set_style(
   style_name    = "my_style"
   , style_entry = list(
      prop_digits_round     = 2
      , count_digits_sigfig = 3
      , count_method        = "sigfig"
      , count_pad_sigfigs   = TRUE
      , prop_nsmall         = 1
      , count_nsmall        = 1
      , decimal.mark        = "."
      , neg_mark_UI         = "-"
      , count_big.mark      = ","
      , neg_mark_mean       = "a decrease of"
      , UI_only             = FALSE
      , UI_text             = ""
      , assert_clu_order    = TRUE
      , is_lancet           = FALSE
      , label_thousands     = FALSE
      , round_5_up          = TRUE
   )
)
```

style_lancet *Lancet style schema*

Description

Pre-defined style schema for Lancet journal formatting

Usage

```
style_lancet()
```

Value

[list] named list representing the lancet style

See Also

Other styles: assert_style_schema(), get_style(), get_style_schema(), new_style(), set_style(), style_nature()

Examples

```
style_lancet()
```

style_nature *Nature style schema*

Description

The default style for the package.

Usage

```
style_nature()
```

Details

Pre-defined style schema for Nature journal formatting.

Value

[list] named list representing the nature style

See Also

Other styles: assert_style_schema(), get_style(), get_style_schema(), new_style(), set_style(), style_lancet()

Examples

```
style_nature()
```

`thin_space` *Thin space*

Description

Lancet thin space separator for counts 10,000 – 999,9999 instead of comma ",": ("\U2009")

Usage

```
thin_space()
```

Value

[chr] thin space character

See Also

Other marks: `en_dash()`, `mid_dot()`

Examples

```
thin_space()
```

Index