Package 'tidyHeatmap'

May 8, 2026

Type Package

Title A Tidy Implementation of Heatmap

Version 1.13.1

Author Stefano Mangiola [aut, cre],
Anthony Papenfuss [ctb]

Maintainer Stefano Mangiola <mangiolastefano@gmail.com>

Description This is a tidy implementation for heatmap. At the moment it is based on the (great) package 'ComplexHeatmap'. The goal of this package is to interface a tidy data frame with this powerful tool. Some of the advantages are: Row and/or columns colour annotations are easy to integrate just specifying one parameter (column names). Custom grouping of rows is easy to specify providing a grouped tbl. For example: df %>% group_by(...). Labels size adjusted by row and column total number. Default use of Brewer and Viridis palettes.

License GPL-3

URL https://www.r-project.org,
https://github.com/stemangiola/tidyHeatmap

BugReports https://github.com/stemangiola/tidyHeatmap/issues

Depends R (>= 4.3.0)

Imports methods, stats, utils, dplyr (>= 0.8.5), magrittr (>= 1.5), tidyr (>= 1.0.3), rlang (>= 0.4.5), purrr (>= 0.3.3), tibble, ComplexHeatmap (>= 2.2.0), viridis (>= 0.5.1), circlize (>= 0.4.8), RColorBrewer (>= 1.1), grid, grDevices, lifecycle (>= 0.2.0), dendextend, patchwork

Suggests spelling, testthat, vdiffr, BiocManager, knitr, rmarkdown, qpdf, covr, roxygen2, forcats, ggplot2

VignetteBuilder knitr

RdMacros lifecycle

Biarch true

biocViews AssayDomain, Infrastructure

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Language en-US

NeedsCompilation no

Repository CRAN

Date/Publication 2025-10-30 07:50:02 UTC

Contents

`+.InputHeatmap` *Add two tidyHeatmap objects together*

Description

Add two tidyHeatmap objects together

Usage

```
## S3 method for class 'InputHeatmap'
e1 + e2
```

Arguments

`e1`	First tidyHeatmap object
`e2`	Second tidyHeatmap object

Value

A combined ComplexHeatmap object

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

add_annotation *add_annotation*

Description

add_annotation() takes a tbl object and easily produces a ComplexHeatmap plot, with integration with tibble and dplyr frameworks.

Usage

```
add_annotation(
  my_input_heatmap,
  annotation,
  type = rep("tile", length(quo_names(annotation))),
  palette_discrete = list(),
  palette_continuous = list(),
  size = NULL,
  ...
)
```

Arguments

my_input_heatmap	A 'InputHeatmap' formatted as | <SAMPLE> | <TRANSCRIPT> | <COUNT> | <...> |
annotation	Vector of quotes
type	A character vector of the set c(\"tile\", \"point\", \"bar\", \"line\")
palette_discrete	A list of character vectors. This is the list of palettes that will be used for horizontal and vertical discrete annotations. The discrete classification of annotations depends on the column type of your input tibble (e.g., character and factor).
palette_continuous	A list of character vectors. This is the list of palettes that will be used for horizontal and vertical continuous annotations. The continuous classification of annotations depends on the column type of your input tibble (e.g., integer, numerical, double).
size	A grid::unit object, e.g. unit(2, "cm"). This is the height or width of the annotation depending on the orientation.
...	The arguments that will be passed to top_annotation or left_annotation of the ComplexHeatmap container

Details

To be added.

Value

A ‘ComplexHeatmap‘ object

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

---

add_attr *Add attribute to abject*

Description

Add attribute to abject

Usage

```
add_attr(var, attribute, name)
```

Arguments

var	A tibble
attribute	An object
name	A character name of the attribute

Value

A tibble with an additional attribute

---

add_class *Add class to abject*

Description

Add class to abject

Usage

```
add_class(var, name)
```

Arguments

var	A tibble
name	A character name of the attribute

Value

A tibble with an additional attribute

`annotation_bar` *Adds a bar annotation layer to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

annotation_bar() from a 'InputHeatmap' object, adds a bar annotation layer.

Usage

```
annotation_bar(.data, .column, palette = NULL, size = NULL, ...)

## S4 method for signature 'InputHeatmap'
annotation_bar(.data, .column, palette = NULL, size = NULL, ...)
```

Arguments

.data	a 'InputHeatmap' object created calling 'tidyHeatmap::heatmap()'
.column	Vector of quotes
palette	A character vector of colors, or a function such as colorRamp2 (see examples).
size	A grid::unit object, e.g. unit(2, "cm"). This is the height or width of the annotation depending on the orientation.
...	The arguments that will be passed to anno_barplot and HeatmapAnnotation if you want to fine tune the aesthetics.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> annotation_bar(inflection)
```

---

annotation_group *Add group annotation strips to a tidyHeatmap*

Description

'annotation_group()' adds group annotation strips to a tidyHeatmap object, allowing you to visually group rows or columns based on categorical variables. This is useful for highlighting biological or experimental groups in your heatmap.

Usage

```
annotation_group(.data, ...)

## S4 method for signature 'InputHeatmap'
annotation_group(
  .data,
  ...,
  palette_grouping = list(),
  group_label_fontsize = 8,
  show_group_name = TRUE,
  group_strip_height = grid::unit(9, "pt")
)
```

Arguments

.data	A tidyHeatmap object
...	Grouping columns (unquoted, like dplyr::group_by)

palette_grouping
: List of color vectors for each grouping. Each element should be a vector of colors for the corresponding grouping variable.

group_label_fontsize
: Font size for group labels

show_group_name
: Logical, show the group annotation name

group_strip_height
: Height of group strip as a grid unit (default: 9pt)

Value

A tidyHeatmap object with group annotation strips added

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
# Basic usage with row grouping
tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
  ) |>
  annotation_group(CAPRA_TOTAL)

# With custom colors and formatting
tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
  ) |>
  annotation_group(
    CAPRA_TOTAL,
    palette_grouping = list(c("#E64B35", "#4DBBD5")),
    group_label_fontsize = 10,
    show_group_name = FALSE
  )

# Multiple grouping variables
tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
```

```
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
  ) |>
  annotation_group(
    CAPRA_TOTAL,
    `Cell type`,
    palette_grouping = list(
      c("#E64B35", "#4DBBD5"),  # colors for CAPRA_TOTAL
      c("#00A087", "#F39B7F")   # colors for Cell type
    )
  )
```

---

annotation_line *Adds a line annotation layer to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

annotation_line() from a 'InputHeatmap' object, adds a line annotation layer.

Usage

```
annotation_line(.data, .column, palette = NULL, size = NULL, ...)

## S4 method for signature 'InputHeatmap'
annotation_line(.data, .column, palette = NULL, size = NULL, ...)
```

Arguments

.data	a 'InputHeatmap' object created calling 'tidyHeatmap::heatmap()'
.column	Vector of quotes
palette	A character vector of colors, or a function such as colorRamp2 (see examples).
size	A grid::unit object, e.g. unit(2, "cm"). This is the height or width of the annotation depending on the orientation.
...	The arguments that will be passed to anno_lines and HeatmapAnnotation if you want to fine tune the aesthetics.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> annotation_line(inflection)
```

---

annotation_numeric *Adds a numeric annotation layer to an 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

'annotation_numeric()' from an 'InputHeatmap' object adds a numeric annotation layer.

Usage

```
annotation_numeric(
  .data,
  .column,
  palette = NULL,
  size = NULL,
  labels_format = function(x) sprintf("%.1f", x),
  ...
)

## S4 method for signature 'InputHeatmap'
annotation_numeric(
  .data,
  .column,
  palette = NULL,
  size = NULL,
```

```
  labels_format = function(x) sprintf("%.1f", x),
  ...
)
```

Arguments

.data	A 'tbl_df' formatted as | <ELEMENT> | <FEATURE> | <VALUE> | <...> |
.column	Vector of quotes
palette	A character vector of colours, or a function such as colorRamp2 (see examples).
size	A grid::unit object, e.g. unit(2, "cm"). This is the height or width of the annotation depending on the orientation.
labels_format	A function to format the numeric labels. By default, it formats numbers to two decimal places using 'sprintf(" supply any function that takes a numeric vector and returns a character vector for customised formatting.
...	The arguments that will be passed to anno_numeric and HeatmapAnnotation if you want to fine tune the aesthetics.

Details

[Maturing]

It uses 'ComplexHeatmap' as the visualisation tool.

Value

An 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'.

An 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'.

Source

[Mangiola and Papenfuss, 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> annotation_numeric(inflection)
```

```
# Align to the right
hm |> annotation_numeric(inflection, align_to = "right")
```

`annotation_point` *Adds a point annotation layer to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

annotation_point() from a 'InputHeatmap' object, adds a point annotation layer.

Usage

```
annotation_point(.data, .column, palette = NULL, size = NULL, ...)

## S4 method for signature 'InputHeatmap'
annotation_point(.data, .column, palette = NULL, size = NULL, ...)
```

Arguments

`.data`	a 'InputHeatmap' object created calling 'tidyHeatmap::heatmap()'
`.column`	Vector of quotes
`palette`	A character vector of colors, or a function such as colorRamp2 (see examples).
`size`	A grid::unit object, e.g. unit(2, "cm"). This is the height or width of the annotation depending on the orientation.
`...`	The arguments that will be passed to `anno_points` and `HeatmapAnnotation` if you want to fine tune the aesthetics.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> annotation_point(inflection)
```

`annotation_tile` *Adds a tile annotation layer to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

annotation_tile() from a 'InputHeatmap' object, adds a tile annotation layer.

Usage

```
annotation_tile(.data, .column, palette = NULL, size = NULL, ...)

## S4 method for signature 'InputHeatmap'
annotation_tile(.data, .column, palette = NULL, size = NULL, ...)
```

Arguments

`.data`	a 'InputHeatmap' object created calling 'tidyHeatmap::heatmap()'
`.column`	Vector of quotes
`palette`	A character vector of colors, or a function such as colorRamp2 (see examples).
`size`	A grid::unit object, e.g. unit(2, "cm"). This is the height or width of the annotation depending on the orientation.
`...`	The arguments that will be passed to `anno_block` and `HeatmapAnnotation` if you want to fine tune the aesthetics.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A ‘InputHeatmap‘ object that gets evaluated to a ‘ComplexHeatmap‘

A ‘InputHeatmap‘ object that gets evaluated to a ‘ComplexHeatmap‘

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> annotation_tile(CAPRA_TOTAL)


hm |>
  annotation_tile(
    inflection,
    palette = circlize::colorRamp2(c(0, 3,10), c("white", "green", "red"))
 )
```

`annot_to_list` *annot_to_list*

Description

annot_to_list

Usage

```
annot_to_list(.data)
```

Arguments

`.data`	A data frame

Value

A list

`as_ComplexHeatmap` *Creates a 'ComplexHeatmap' object for less standard plot manipulation (e.g. changing legend position)*

Description

as_ComplexHeatmap() takes a 'InputHeatmap' object and produces a 'Heatmap' object

Usage

```
as_ComplexHeatmap(tidyHeatmap)

## S4 method for signature 'InputHeatmap'
as_ComplexHeatmap(tidyHeatmap)
```

Arguments

`tidyHeatmap`	A 'InputHeatmap' object from tidyHeatmap::heatmap() call

Details

[Maturing]

Value

A 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
tidyHeatmap::N52 |>
tidyHeatmap::heatmap(
 .row = symbol_ct,
 .column = UBR,
 .value = `read count normalised log`,
) |>
as_ComplexHeatmap()
```

`as_matrix` *Get matrix from tibble*

Description

Get matrix from tibble

Usage

```
as_matrix(tbl, rownames = NULL, do_check = TRUE)
```

Arguments

`tbl`	A tibble
`rownames`	A character string of the rownames
`do_check`	A boolean

Value

A matrix

`check_if_duplicated_genes` *Check whether there are duplicated genes/transcripts*

Description

Check whether there are duplicated genes/transcripts

Usage

```
check_if_duplicated_genes(.data, .sample, .transcript, .abundance)
```

Arguments

.data	A tibble of read counts
.sample	A character name of the sample column
.transcript	A character name of the transcript/gene column
.abundance	A character name of the read count column

Value

A tbl

---

check_if_wrong_input *Check whether there are NA counts*

Description

Check whether there are NA counts

Usage

```
check_if_wrong_input(.data, list_input, expected_type)
```

Arguments

.data	A tibble of read counts
list_input	A list
expected_type	A character string

Value

A tbl

---

drop_class *Remove class to abject*

Description

Remove class to abject

Usage

```
drop_class(var, name)
```

Arguments

var	A tibble
name	A character name of the class

Value

A tibble with an additional attribute

---

error_if_log_transformed

Check whether a numeric vector has been log transformed

Description

Check whether a numeric vector has been log transformed

Usage

```
error_if_log_transformed(x, .abundance)
```

Arguments

x	A numeric vector
.abundance	A character name of the transcript/gene abundance column

Value

NA

---

get_abundance_norm_if_exists

Get column names either from user or from attributes

Description

Get column names either from user or from attributes

Usage

```
get_abundance_norm_if_exists(.data, .abundance)
```

Arguments

.data	A tibble
.abundance	A character name of the abundance column

Value

A list of column enquo or error

`get_elements` *Get column names either from user or from attributes*

Description

Get column names either from user or from attributes

Usage

```
get_elements(.data, .element, of_samples = TRUE)
```

Arguments

`.data`	A tibble
`.element`	A character name of the sample column
`of_samples`	A boolean

Value

A list of column enquo or error

`get_elements_features` *Get column names either from user or from attributes*

Description

Get column names either from user or from attributes

Usage

```
get_elements_features(.data, .element, .feature, of_samples = TRUE)
```

Arguments

`.data`	A tibble
`.element`	A character name of the sample column
`.feature`	A character name of the transcript/gene column
`of_samples`	A boolean

Value

A list of column enquo or error

get_elements_features_abundance

Get column names either from user or from attributes

Description

Get column names either from user or from attributes

Usage

```
get_elements_features_abundance(
  .data,
  .element,
  .feature,
  .abundance,
  of_samples = TRUE
)
```

Arguments

.data	A tibble
.element	A character name of the sample column
.feature	A character name of the transcript/gene column
.abundance	A character name of the read count column
of_samples	A boolean

Value

A list of column enquo or error

get_heatmap_data

Retrieve heatmap data and dendrograms as plotted

Description

get_heatmap_data() extracts the heatmap matrix as it appears in the plot along with the row and column dendrograms, all with consistent naming.

Usage

```
get_heatmap_data(.data)

## S4 method for signature 'InputHeatmap'
get_heatmap_data(.data)
```

Arguments

.data	A ‘InputHeatmap‘ object from tidyHeatmap::heatmap()

Details

[Maturing]

This function converts the InputHeatmap to ComplexHeatmap, draws it to perform clustering, then extracts the ordered matrix and dendrograms exactly as they appear in the heatmap plot.

Value

A list containing:

- matrix: The abundance matrix with rows and columns ordered as in the heatmap
- row_dend: The row dendrogram object
- column_dend: The column dendrogram object

A list containing the ordered matrix, row dendrogram, and column dendrogram

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm <- tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
  ) |>
  annotation_group(
    CAPRA_TOTAL,
    palette_grouping = list(c("#E64B35", "#4DBBD5")),
    group_label_fontsize = 10,
    show_group_name = FALSE
  )

# Multiple grouping variables
tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
  ) |>
```

```
  annotation_group(
    CAPRA_TOTAL,
    `Cell type`,
    palette_grouping = list(
      c("#E64B35", "#4DBBD5"),  # colors for CAPRA_TOTAL
      c("#00A087", "#F39B7F")   # colors for Cell type
    )
  )

# Get heatmap data as plotted
result <- hm |> get_heatmap_data()
ordered_matrix <- result$matrix
row_dendrogram <- result$row_dend
column_dendrogram <- result$column_dend
```

`get_sample_counts` *Get column names either from user or from attributes*

Description

Get column names either from user or from attributes

Usage

```
get_sample_counts(.data, .sample, .abundance)
```

Arguments

`.data`	A tibble
`.sample`	A character name of the sample column
`.abundance`	A character name of the read count column

Value

A list of column enquo or error

get_sample_transcript *Get column names either from user or from attributes*

Description

Get column names either from user or from attributes

Usage

```
get_sample_transcript(.data, .sample, .transcript)
```

Arguments

.data	A tibble
.sample	A character name of the sample column
.transcript	A character name of the transcript/gene column

Value

A list of column enquo or error

get_sample_transcript_counts *Get column names either from user or from attributes*

Description

Get column names either from user or from attributes

Usage

```
get_sample_transcript_counts(.data, .sample, .transcript, .abundance)
```

Arguments

.data	A tibble
.sample	A character name of the sample column
.transcript	A character name of the transcript/gene column
.abundance	A character name of the read count column

Value

A list of column enquo or error

`get_x_y_annotation_columns` *get_x_y_annotation_columns*

Description

get_x_y_annotation_columns

Usage

```
get_x_y_annotation_columns(.data, .column, .row, .abundance)
```

Arguments

`.data`	A ‘tbl‘ formatted as | <SAMPLE> | <TRANSCRIPT> | <COUNT> | <...> |
`.column`	The name of the column horizontally presented in the heatmap
`.row`	The name of the column vertically presented in the heatmap
`.abundance`	The name of the transcript/gene abundance column

Value

A list

`heatmap` *Creates a ‘InputHeatmap‘ object from ‘tbl_df‘ on evaluation creates a ‘ComplexHeatmap‘*

Description

heatmap() takes a tbl object and easily produces a ComplexHeatmap plot, with integration with tibble and dplyr frameworks.

Usage

```
heatmap(
  .data,
  .row,
  .column,
  .value,
  transform = NULL,
  scale = "none",
  palette_value = c("#440154FF", "#21908CFF", "#fefada"),
  palette_grouping = list(),
  .scale = NULL,
  ...
```

```
)

heatmap_(
  .data,
  .row,
  .column,
  .value,
  transform = NULL,
  scale = "none",
  palette_value = c("#440154FF", "#21908CFF", "#fefada"),
  palette_grouping = list(),
  .scale = NULL,
  ...
)

## S4 method for signature 'tbl'
heatmap(
  .data,
  .row,
  .column,
  .value,
  transform = NULL,
  scale = "none",
  palette_value = c("#440154FF", "#21908CFF", "#fefada"),
  palette_grouping = list(),
  .scale = NULL,
  ...
)

## S4 method for signature 'tbl_df'
heatmap(
  .data,
  .row,
  .column,
  .value,
  transform = NULL,
  scale = "none",
  palette_value = c("#440154FF", "#21908CFF", "#fefada"),
  palette_grouping = list(),
  .scale = NULL,
  ...
)
```

Arguments

.data	A ‘tbl_df‘ formatted as | <ELEMENT> | <FEATURE> | <VALUE> | <...> |
.row	The name of the column to use as rows
.column	The name of the column to use as columns

.value	The name of the column to use as values
transform	A function to transform the data (optional)
scale	One of "none", "row", "column", or "both" for scaling
palette_value	A character vector of colors or a function for value colors
palette_grouping	A list of color palettes for grouping annotations
.scale	DEPRECATED: Use scale instead
...	Additional arguments passed to ComplexHeatmap

Details

[Maturing]

This function takes a tbl as an input and creates a ‘ComplexHeatmap‘ plot. The information is stored in a ‘InputHeatmap‘ object that is updated along the pipe statement, for example adding annotation layers.

Value

An ‘InputHeatmap‘ object that gets evaluated to a ‘ComplexHeatmap‘

A ‘InputHeatmap‘ object

A ‘InputHeatmap‘ object

A ‘InputHeatmap‘ object

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
tidyHeatmap::N52 |>
  dplyr::group_by( `Cell type`) |>
  tidyHeatmap::heatmap(
   .row = symbol_ct,
   .column = UBR,
   .value = `read count normalised log`,
  )
```

ifelse2_pipe — *This is a generalisation of ifelse that accepts an object and return an objects*

Description

This is a generalisation of ifelse that accepts an object and return an objects

Usage

```
ifelse2_pipe(.x, .p1, .p2, .f1, .f2, .f3 = NULL)
```

Arguments

.x	A tibble
.p1	A boolean
.p2	ELSE IF condition
.f1	A function
.f2	A function
.f3	A function

Value

A tibble

ifelse_pipe — *This is a generalisation of ifelse that accepts an object and return an objects*

Description

This is a generalisation of ifelse that accepts an object and return an objects

Usage

```
ifelse_pipe(.x, .p, .f1, .f2 = NULL)
```

Arguments

.x	A tibble
.p	A boolean
.f1	A function
.f2	A function

Value

A tibble

`input_heatmap` *input_heatmap*

Description

input_heatmap() takes a tbl object and easily produces a ComplexHeatmap plot, with integration with tibble and dplyr frameworks.

Usage

```
input_heatmap(
  .data,
  .horizontal,
  .vertical,
  .abundance,
  transform = NULL,
  scale = "none",
  palette_value = c("#440154FF", "#21908CFF", "#fefada"),
  palette_grouping = list(),
  ...
)
```

Arguments

`.data`	A ‘tbl‘ formatted as | <SAMPLE> | <TRANSCRIPT> | <COUNT> | <...> |
`.horizontal`	The name of the column horizontally presented in the heatmap
`.vertical`	The name of the column vertically presented in the heatmap
`.abundance`	The name of the transcript/gene abundance column
`transform`	A function, used to transform .value, for example log1p
`scale`	A character string. Possible values are c(\"none\", \"row\", \"column\", \"both\")
`palette_value`	A character vector, or a function for higher customisation (colorRamp2). This is the palette that will be used as gradient for abundance. If palette_value is a vector of hexadecimal colours, it should have 3 values. If you want more customisation, you can pass to palette_value a function, that is derived as for example ‘colorRamp2(c(-2, 0, 2), palette_value)‘
`palette_grouping`	A list of character vectors. This is the list of palettes that will be used for grouping
`...`	Further arguments to be passed to ComplexHeatmap::Heatmap

Details

To be added.

Value

A 'ComplexHeatmap' object

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

`layer_arrow_down` *Adds a layers of symbols above the heatmap tiles to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

layer_arrow_down() from a 'InputHeatmap' object, adds a bar annotation layer.

Usage

```
layer_arrow_down(.data, ..., .size = NULL)

## S4 method for signature 'InputHeatmap'
layer_arrow_down(.data, ..., .size = NULL)
```

Arguments

`.data`	A 'InputHeatmap'
`...`	Expressions that return a logical value, and are defined in terms of the variables in .data. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
`.size`	A column name or a double. The size of the elements of the layer.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A ‘InputHeatmap‘ object that gets evaluated to a ‘ComplexHeatmap‘

A ‘InputHeatmap‘ object that gets evaluated to a ‘ComplexHeatmap‘

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> layer_arrow_down()
```

---

layer_arrow_up *Adds a layers of symbols above the heatmap tiles to a ‘InputHeatmap‘, that on evaluation creates a ‘ComplexHeatmap‘*

Description

layer_arrow_up() from a ‘InputHeatmap‘ object, adds a bar annotation layer.

Usage

```
layer_arrow_up(.data, ..., .size = NULL)

## S4 method for signature 'InputHeatmap'
layer_arrow_up(.data, ..., .size = NULL)
```

Arguments

.data	A 'InputHeatmap'
...	Expressions that return a logical value, and are defined in terms of the variables in .data. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
.size	A column name or a double. The size of the elements of the layer.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> layer_arrow_up()
```

`layer_asterisk` *Adds a layer of symbols above the heatmap tiles to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

layer_asterisk() from a 'InputHeatmap' object, adds a symbol annotation layer over the heatmap tiles.

Usage

```
layer_asterisk(.data, ..., .size = NULL)

## S4 method for signature 'InputHeatmap'
layer_asterisk(.data, ..., .size = NULL)
```

Arguments

`.data`	A 'InputHeatmap'
`...`	Expressions that return a logical value, and are defined in terms of the variables in .data. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
`.size`	A column name or a double. The size of the elements of the layer.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> layer_asterisk()
```

`layer_diamond` *Adds a layers of symbols above the heatmap tiles to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

layer_diamond() from a 'InputHeatmap' object, adds a bar annotation layer.

Usage

```
layer_diamond(.data, ..., .size = NULL)

## S4 method for signature 'InputHeatmap'
layer_diamond(.data, ..., .size = NULL)
```

Arguments

`.data`	A 'InputHeatmap'
`...`	Expressions that return a logical value, and are defined in terms of the variables in .data. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
`.size`	A column name or a double. The size of the elements of the layer.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> layer_diamond()
```

---

layer_point *Adds a layers of symbols above the heatmap tiles to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

layer_point() from a 'InputHeatmap' object, adds a bar annotation layer.

Usage

```
layer_point(.data, ..., .size = NULL)

## S4 method for signature 'InputHeatmap'
layer_point(.data, ..., .size = NULL)
```

Arguments

.data	A 'InputHeatmap'
...	Expressions that return a logical value, and are defined in terms of the variables in .data. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
.size	A column name or a double. The size of the elements of the layer.

Details

[Maturing]

It uses ‘ComplexHeatmap‘ as visualisation tool.

Value

A ‘InputHeatmap‘ object that gets evaluated to a ‘ComplexHeatmap‘

A ‘InputHeatmap‘ object that gets evaluated to a ‘ComplexHeatmap‘

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> layer_point()
```

---

layer_square *Adds a layers of symbols above the heatmap tiles to a ‘InputHeatmap‘, that on evaluation creates a ‘ComplexHeatmap‘*

Description

layer_square() from a ‘InputHeatmap‘ object, adds a bar annotation layer.

Usage

```
layer_square(.data, ..., .size = NULL)

## S4 method for signature 'InputHeatmap'
layer_square(.data, ..., .size = NULL)
```

Arguments

.data	A 'InputHeatmap'
...	Expressions that return a logical value, and are defined in terms of the variables in .data. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
.size	A column name or a double. The size of the elements of the layer.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> layer_square()
```

`layer_star` *Adds a layer of symbols above the heatmap tiles to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

layer_star() from a 'InputHeatmap' object, adds a symbol annotation layer over the heatmap tiles.

Usage

```
layer_star(.data, ..., .size = NULL)

## S4 method for signature 'InputHeatmap'
layer_star(.data, ..., .size = NULL)
```

Arguments

`.data`	A 'InputHeatmap'
`...`	Expressions that return a logical value, and are defined in terms of the variables in .data. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
`.size`	A column name or a double. The size of the elements of the layer.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> layer_star()
```

`layer_text` *Adds a layers of texts above the heatmap tiles to a 'InputHeatmap', that on evaluation creates a 'ComplexHeatmap'*

Description

layer_text() from a 'InputHeatmap' object, adds a text annotation layer.

Usage

```
layer_text(.data, ..., .value, .size = NULL, .color = NULL)

## S4 method for signature 'InputHeatmap'
layer_text(.data, ..., .value, .size = NULL, .color = NULL)
```

Arguments

`.data`	A 'InputHeatmap'
`...`	Expressions that return a logical value, and are defined in terms of the variables in .data. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
`.value`	A column name or character string.
`.size`	A column name or a double. The size of the elements of the layer.
`.color`	A column name or character string. The color of the elements of the layer.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
library(dplyr)

hm =
  tidyHeatmap::N52 |>
  mutate(my_text = "t") |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> layer_text(.value = "a")
hm |> layer_text(.value = my_text)
```

N52 *Example data set N52*

Description

Example data set N52

Usage

```
N52
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 520 rows and 15 columns.

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

`parse_formula` *.formula parser*

Description

.formula parser

Usage

```
parse_formula(fm)
```

Arguments

`fm`	a formula

Value

A character vector

`pasilla` *Example data set Pasilla*

Description

Example data set Pasilla

Usage

```
pasilla
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 504 rows and 8 columns.

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

prepend *From rlang deprecated*

Description

From rlang deprecated

Usage

```
prepend(x, values, before = 1)
```

Arguments

x	An array
values	An array
before	A boolean

Value

An array

quo_names *Convert array of quosure (e.g. c(col_a, col_b)) into character vector*

Description

Convert array of quosure (e.g. c(col_a, col_b)) into character vector

Usage

```
quo_names(v)
```

Arguments

v	A array of quosures (e.g. c(col_a, col_b))

Value

A character vector

save_pdf *Save plot on PDF file*

Description

save_pdf() takes as input a Heatmap from ComplexHeatmap and save it to PDF file

Usage

```
save_pdf(
  .heatmap,
  filename,
  width = NULL,
  height = NULL,
  units = c("in", "cm", "mm")
)
```

Arguments

.heatmap	A 'Heatmap'
filename	A character string. The name of the output file/path
width	A 'double'. Plot width
height	A 'double'. Plot height
units	A character string. units ("in", "cm", or "mm")

Details

[Maturing]

It simply save an 'Heatmap' to a PDF file use pdf() function in the back end

Value

NA

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
tidyHeatmap::heatmap(
  dplyr::group_by(tidyHeatmap::pasilla,location, type),
  .column = sample,
  .row = symbol,
  .value = `count normalised adjusted`,
 ) |>
 save_pdf(tempfile())
```

---

save_pdf,Heatmap-method

save_pdf

Description

save_pdf

Usage

```
## S4 method for signature 'Heatmap'
save_pdf(
  .heatmap,
  filename,
  width = NULL,
  height = NULL,
  units = c("in", "cm", "mm")
)
```

Arguments

.heatmap	A ‘Heatmap‘
filename	A character string. The name of the output file/path
width	A ‘double‘. Plot width
height	A ‘double‘. Plot height
units	A character string. units ("in", "cm", or "mm")

`save_pdf,InputHeatmap-method` *save_pdf*

Description

save_pdf

Usage

```
## S4 method for signature 'InputHeatmap'
save_pdf(
  .heatmap,
  filename,
  width = NULL,
  height = NULL,
  units = c("in", "cm", "mm")
)
```

Arguments

`.heatmap`	A 'Heatmap'
`filename`	A character string. The name of the output file/path
`width`	A 'double'. Plot width
`height`	A 'double'. Plot height
`units`	A character string. units ("in", "cm", or "mm")

`scale_robust` *Scale counts in a robust way against sd == 0*

Description

Scale counts in a robust way against sd == 0

Usage

```
scale_robust(y)
```

Arguments

`y`	A numerical array

Value

A scaled and centred numerical array

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

`select_closest_pairs` *Sub function of remove_redundancy_elements_though_reduced_dimensions*

Description

Sub function of remove_redundancy_elements_though_reduced_dimensions

Usage

```
select_closest_pairs(df)
```

Arguments

`df`	A tibble

Value

A tibble with pairs to drop

`split_rows` *Split the heatmap row-wise depending on the biggest branches in the cladogram.*

Description

split_rows() from a 'InputHeatmap' object, split the row cladogram.

split_columns() from a 'InputHeatmap' object, split the column cladogram.

Usage

```
split_rows(.data, number_of_groups)

## S4 method for signature 'InputHeatmap'
split_rows(.data, number_of_groups)

split_columns(.data, number_of_groups)

## S4 method for signature 'InputHeatmap'
split_columns(.data, number_of_groups)
```

Arguments

`.data`	A 'InputHeatmap'
`number_of_groups`	An integer. The number of groups to split the cladogram into.

Details

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

[Maturing]

It uses 'ComplexHeatmap' as visualisation tool.

Value

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

A 'InputHeatmap' object that gets evaluated to a 'ComplexHeatmap'

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
hm =
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> split_rows(2)



hm =
```

```
  tidyHeatmap::N52 |>
  tidyHeatmap::heatmap(
    .row = symbol_ct,
    .column = UBR,
    .value = `read count normalised log`
)

hm |> split_columns(2)
```

---

wrap_heatmap *Wrap tidyHeatmap (ComplexHeatmap) in a patchwork-compliant patch*

Description

In order to add tidyHeatmap (ComplexHeatmap) element to a patchwork they can be converted to a compliant representation using the ‘wrap_heatmap()‘ function. This allows you to position either grobs, ggplot objects, patchwork objects, or even base graphics (if passed as a formula) in either the full area, the full plotting area (anything between and including the axis label), or the panel area (only the actual area where data is drawn).

Usage

```
wrap_heatmap(
  panel = NULL,
  plot = NULL,
  full = NULL,
  clip = TRUE,
  ignore_tag = FALSE,
  padding = NULL
)

## S4 method for signature 'InputHeatmap'
wrap_heatmap(
  panel = NULL,
  plot = NULL,
  full = NULL,
  clip = TRUE,
  ignore_tag = FALSE,
  padding = NULL
)
```

Arguments

panel, plot, full
: A grob, ggplot, patchwork, formula, raster, or nativeRaster object to add to the respective area.

clip	Should the grobs be clipped if expanding outside its area
ignore_tag	Should tags be ignored for this patch. This is relevant when using automatic tagging of plots and the content of the patch does not qualify for a tag.
padding	A grid::unit object. It defined the padding distance for the plot. It is helpful when the heatmap is assembled with other ggplots through patchwork.

Value

A wrapped_patch object

A wrapped_patch object

Source

[Mangiola and Papenfuss., 2020](https://joss.theoj.org/papers/10.21105/joss.02472)

References

Mangiola, S. and Papenfuss, A.T., 2020. "tidyHeatmap: an R package for modular heatmap production based on tidy principles." Journal of Open Source Software. doi:10.21105/joss.02472.

Examples

```
tidyHeatmap::N52 |>
tidyHeatmap::heatmap(
 .row = symbol_ct,
 .column = UBR,
 .value = `read count normalised log`,
) |>
wrap_heatmap()
```

Index