Package 'bubbleHeatmap'

July 22, 2025

Type Package

Title Produces 'bubbleHeatmap' Plots for Visualising Metabolomics Data

Version 0.1.1

Description Plotting package based on the grid system, combining elements of
a bubble plot and heatmap to conveniently display two numerical variables,
(represented by color and size) grouped by categorical variables on the x
and y axes. This is a useful alternative to a forest plot when the data can
be grouped in two dimensions, such as predictors x outcomes. It has
particular advantages for visualising the metabolic measures produced by
the 'Nightingale Health' metabolomics platform, and templates are included
for automatically generating figures from these datasets.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Imports reshape, methods

Depends R (>= 3.5.0), grid

Suggests rmarkdown, knitr

VignetteBuilder knitr

NeedsCompilation no

Author Ruth Boxall [aut, cre],
Michael Holmes [ctb],
Robin Walters [ctb]

Maintainer Ruth Boxall <ruth.boxall@ndph.ox.ac.uk>

Repository CRAN

Date/Publication 2023-05-15 14:50:02 UTC

Contents

---

arrangePanels   *Arrange Multiple bubbleHeatmap gTrees Inside a New Viewport.*

Description

The arrangePanels function creates a new viewport with a grid layout, and directs the input trees to
be drawn at positions within the new layout. The new layout is filled by rows.

Usage

```
arrangePanels(
  name,
  subTrees,
  nrow,
  ncol,
  spacer = unit(0, "cm"),
  margin = unit(0, "cm"),
  just = c(0.5, 0.5)
)
```

Arguments

name	Character string naming the new viewport (will have suffix "VP")
subTrees	gTrees to be arranged in the new panel
nrow	Numeric, number of rows in the new layout
ncol	Numeric, number of columns in the new layout
spacer	Grid [unit](#) object specifying size of space left between plots.
margin	Grid [unit](#) object specifying size of margin around panel.
just	Numeric or character vector specifying justification of tree viewports within panel viewport layout cells. See [viewport](#).

Value

List containing grobList (a list of modified gTrees) and vp, the new viewport object.

---

bubbleHeatmap   *Generate bubbleHeatmap Plot Object*

Description

This function creates a gTree for a bubbleHeatmap plot. These plots have a grid of "bubbles" with the color and size each scaled according to the values of a different variable. They can be used as an alternative to forest plots for visualizing estimates and errors that can be organised in a grid format, correlograms, and any other two variable matrices.

Usage

```
bubbleHeatmap(
  colorMat,
  sizeMat = colorMat,
  treeName = "Plot",
  context = gpar(cex = 0.8),
  colorSeq = c("#053061", "#2166AC", "#4393C3", "#92C5DE", "#D1E5F0", "#FDDBC7",
    "#F4A582", "#D6604D", "#B2182B", "#67001F"),
  unitBase = unit(0.5, "cm"),
  diameter = 0.8,
  colorLim = c(NA, NA),
  sizeLim = c(NA, NA),
  showTopLabels = TRUE,
  showLeftLabels = TRUE,
  leftLabelsTitle = FALSE,
  showRowBracket = FALSE,
  rowTitle = FALSE,
  showColBracket = FALSE,
  colTitle = FALSE,
  plotTitle = FALSE,
  xTitle = FALSE,
  yTitle = FALSE,
  showColorLegend = TRUE,
  showBubbleLegend = TRUE,
  colorBreaks = NULL,
  sizeBreaks = NULL,
  legendHeight = 8,
  legendTitles = c(expression("-log"[10] * "P"), "Estimate (SD)")
)
```

Arguments

colorMat, sizeMat

Numeric matrices containing values to be mapped to bubble color and size respectively. The dimensions, rownames and colnames of these matrices must match.

treeName   Identifying character string for [gTree](#).

context   [gpar](#) object to provide drawing context. Will be applied to the parent viewport in the childrenvp slot of the gTree.

colorSeq   Character vector of colors to be used in creating color scale.

unitBase   [unit](#) object indicating the size of one plot grid box.

diameter   Maximum diameter of bubbles as multiple of unitBase.

colorLim, sizeLim

Numeric vectors indicating limits (min/max) outside which the values of color/size matrices should be truncated. NOTE! These data ranges are separate from the scale ranges, which are taken from the min/max values of colorBreaks and sizeBreaks.

showTopLabels   Logical, should matrix column names be printed above plot?

showLeftLabels Logical, should matrix row names be printed above plot?

leftLabelsTitle

Heading to print over the column of row names.

showRowBracket Logical vector, should vertical bracket be printed to the right of the plot (to label a row of plots in a final figure layout)?

rowTitle   Label to be printed to the right of the plot, outside row bracket if present, to label a row of plots in a final figure layout.

showColBracket Logical, should horizontal bracket be printed below the plot (to label a column of plots in a final figure layout)?

colTitle   Label to be printed below plot, outside col bracket if present, to label a column of plots in a final figure layout.

plotTitle   Plot title string, to be centered above plot.

xTitle   X-axis title string, displayed left of LeftLabels.

yTitle   Y-axis title string, displayed above TopLabels.

showColorLegend, showBubbleLegend

Should plot include legends for color scale/bubble size?

colorBreaks, sizeBreaks

Character vectors of legend tick labels. The min/max values will represent the range of color/size scales. If not supplied then these will be generated using [pretty](#), to a range outside data values (after values truncated to colorLim\sizeLim if applicable).

legendHeight   Numeric, preferred height of legends in multiples of unitBase. This will be supplied to [pretty](#) function. If sizeBreaks vector is supplied then this will be ignored and legend height will be length(sizeBreaks).

legendTitles   Character vector of legend titles. First string should be the size legend title, second for the color legend.

Value

This function creates a bubbleHeatmap using the `grid` graphics package. It does not draw any output but produces a [gTree](#) object which can be drawn using the [grid.draw](#) function. Returned trees always include a [viewport](#) object with a 7x6 layout and a given plot element is always positioned in the same cell, to assist with aligning elements when combining multiple plots in a single figure (see vignette, [matchLayoutHeights](#), [matchLayoutWidths](#)). NOTE: A cairo output device with gradient support is necessary to correctly render the color legend.

See Also

Other build functions: [bubbleHMLegends](#)(), [bubbleHeatmapList](#)(), [multiPlotInput](#)()

Examples

```
# Example 1 - Available plot features#
names <- list(paste0("leftLabels", 1:6), paste0("topLabels", 1:10))
colorMat <- matrix(rnorm(60), nrow = 6, ncol = 10, dimnames = names)
sizeMat <- matrix(abs(rnorm(60)), nrow = 6, ncol = 10, dimnames = names)
tree <- bubbleHeatmap(colorMat, sizeMat,
  treeName = "example",
  leftLabelsTitle = "leftLabelsTitle", showRowBracket = TRUE,
  rowTitle = "rowTitle", showColBracket = TRUE, colTitle = "colTitle",
  plotTitle = "plotTitle", xTitle = "xTitle", yTitle = "yTitle",
  legendTitles = c("legendTitles[1]", "legendTitles[2]")
)
grid.draw(tree)
```

---

bubbleHeatmapList   *Output List of bubbleHeatmap Plots.*

Description

Convenient wrapper for `bubbleHeatmap` and `bubbleHMLegend` for easy output of multiple plots from lists of arguments.

Usage

```
bubbleHeatmapList(inputList)
```

Arguments

inputList   A list of two lists, trees and legend. trees should contain a list of lists of arguments for multiple calls to bubbleHeatmap and legend should contain a list of arguments for a call to bubbleHMLegends. This object can be generated automatically using [multiPlotInput](#).

Value

A list of bubbleHeatmap gTree objects. Any legends specified by the legend list will also be produced as separate trees in the output list.

See Also

Other build functions: [bubbleHMLegends()](bubbleHMLegends), [bubbleHeatmap()](bubbleHeatmap), [multiPlotInput()](multiPlotInput)

---

bubbleHMLegends   *Build Legend Trees for bubbleHeatmap Plots*

Description

Generate a bubble size legend and/or color legend for bubbleHeatmap plots.

Usage

```
bubbleHMLegends(
  sizeBreaks,
  colorBreaks,
  colorSeq,
  unitBase,
  diameter,
  context,
  legendHeight,
  legendTitles,
  showBubbleLegend = TRUE,
  showColorLegend = TRUE,
  name = "Legends",
  orientation = c("horizontal", "vertical")
)
```

Arguments

colorBreaks, sizeBreaks

> Character vectors of legend tick labels. The min/max values will represent the range of color/size scales. If not supplied then these will be generated using pretty, to a range outside data values (after values truncated to colorLim\sizeLim if applicable).

colorSeq   Character vector of colors to be used in creating color scale.

unitBase   unit object indicating the size of one plot grid box.

diameter   Maximum diameter of bubbles as multiple of unitBase.

context   gpar object to provide drawing context. Will be applied to the parent viewport in the childrenvp slot of the gTree.

legendHeight   Numeric, preferred height of legends in multiples of unitBase. This will be supplied to [pretty](#) function. If sizeBreaks vector is supplied then this will be ignored and legend height will be length(sizeBreaks).

legendTitles   Character vector of legend titles. First string should be the size legend title, second for the color legend.

showColorLegend, showBubbleLegend
                Should plot include legends for color scale/bubble size?

name   character string naming the gTree containing the function output.

orientation   one of "horizontal" or "vertical", (default horizontal) indicating whether legends should be arranged stacked or side by side. Ignored if only one legend is required.

Value

A [gTree](#) object representing the bubbleHeatmap legends requested by the showBubbleLegend and showColorLegend parameters.

See Also

Other build functions: [bubbleHeatmapList](#)(), [bubbleHeatmap](#)(), [multiPlotInput](#)()

Examples

```
legends <- bubbleHMLegends(
  sizeBreaks = seq(0, 80, 10), colorBreaks = seq(-2, 2, 0.5), diameter = 0.8,
 context = gpar(cex = 0.8), colorSeq = c("#053061", "#2166AC", "#4393C3", "#92C5DE", "#D1E5F0",
  "#FDDBC7", "#F4A582", "#D6604D", "#B2182B", "#67001F"), unitBase = unit(0.5, "cm"),
  legendHeight = 9, legendTitles = c("Estimate", "NegLog10P"))
grid.draw(legends)
```

---

cetp   *Associations of Nightingale Health Metabolomics Panel with a Cholesterol Esterase Transfer Protein Genetic Risk Score in China Kadoorie Biobank*

Description

Example dataset containing association results of the 225 Nightingale metabolomics traits with a CETP Genetic Risk Score, scaled to 10-mg/dL higher levels of HDL-C.

Usage

cetp

Format

Data frame with 225 rows and 5 variables

trait trait identifier as used in China Kadoorie Biobank, can be used to merge results with ntngale template datasets

estimate association beta

stdErr standard error of association estimate

pval p-value of association

negLog10P -log10(pval)

Details

The sample data included in this package represents associations between Nightingale metabolic measures and a genetic risk score (GRS) for Cholesterol Ester Transfer Protein (CETP) in the China Kadoorie Biobank (CKB), a prospective study of Chinese adults from 10 distinct areas of China. The full design of the CKB cohort and the methods and results of this study have been previously published.

CETP transfers esterified cholesterol from HDL to apolipoprotein B-containing lipoproteins in exchange for triglycerides. This process is a key component of the atheroprotective reverse cholesterol transport pathway, which modulates the return of excess cholesterol from peripheral cells such as macrophages to the liver, where it can be redistributed or excreted. Reduced CETP activity results in higher levels of HDL cholesterol (HDL-C), which has resulted in interest in CETP as a drug target due to the inverse correlation between HDL-C and risk of atherosclerotic disease. In this analysis, five CETP SNP variants were selected on the basis of previously reported associations with HDL cholesterol and CETP activity. Genotyping data for these variants and conventional lipid biochemistry measures were available for a subset of 17,854 CKB participants selected for a CVD case-control study. This data was used to generate a CETP GRS weighted by HDL-C association, derived internally with 100-fold cross-validation. The Nightingale Health metabolomics panel consisting of 225 metabolomics measures was available for 4657 of these individuals, and association with the GRS was assessed by linear regression with adjustment for age and sex, stratified by geographical region. In generation of the GRS and association of metabolomics, outcomes were standardized by rank inverse normal transformation, stratified by region, after adjustment for age and sex.

Source

Millwood IY, Bennett DA, Holmes MV et al. Association of CETP Gene Variants With Risk for Vascular and Nonvascular Diseases Among Chinese Adults. JAMA Cardiol. 2018 Jan 1;3(1):34-43.

---

combinePanels   *Combine Multiple Panels In A New gTree.*

Description

The combinePanels function creates a new gTree containing a viewport with a grid layout, and arranges multiple panels (created with the arrangePanels function) within that grid. The new layout is filled by rows.

Usage

```
combinePanels(
  name,
  panels,
  nrow,
  ncol,
  spacer = unit(0, "cm"),
  margin = unit(0, "cm"),
  just = c(0.5, 0.5)
)
```

Arguments

name	Character string naming the new tree and viewport (viewport will have suffix "VP")
panels	Panels to be arranged in the new tree
nrow	Numeric, number of rows in the new layout
ncol	Numeric, number of columns in the new layout
spacer	Grid [unit](#) object specifying size of space left between panels.
margin	Grid [unit](#) object specifying size of margin around tree.
just	Numeric or character vector specifying justification of panel viewports within new tree viewport layout cells. See [viewport](#).

Value

New [gTree](#) object with children slot containing all trees in the grobLists of input panels, and a childrenvp slot containing all viewports in input panels arranged within a new parent viewport layout.

---

formatData	*Split Data Into Matrices and Reorder Rows and Columns.*

Description

Prepare data variables for plot building by splitting into matrices and ordering rows and columns. Includes pre-defined settings for plotting Nightingale metabolomics data.

Usage

```
formatData(
  data,
  rowName = "rowName",
  colName = "colName",
  plotGroup = "plotGroup",
  colorValue = "colorValue",
```

```
    sizeValue = "sizeValue",
    rowOrder = NULL,
    colOrder = NULL,
    listOrder = NULL,
    nightingale = TRUE
)
```

Arguments

data	Dataset containing variables to be plotted on color and size scales with additional matrix layout specification variables (see details).
rowName	String naming the variable of data which defines rows of each matrix.
colName	String naming the variable of data which defines columns of each matrix.
plotGroup	String naming the variable of data which groups the dataset into plots.
colorValue	String naming the variable of data containing the values to be plotted on the color scale.
sizeValue	String naming the variable of data containing the values to scale bubble sizes.
rowOrder	List of vectors of character strings matching the values of rowName for each plot. List names should match values of plotGroup. Can be NULL (default) if no sorting required or when using inbuilt Nightingale ordering. If used, it is not necessary for all plots to have list entries, but each entry should include all row names for that plot.
colOrder	As rowOrder for columns and colName.
listOrder	Vector of character strings matching values of plotGroup variable, or NULL (default) if no sorting required or when using inbuilt Nightingale ordering.
nightingale	Logical - use Nightingale style settings?

Details

This function takes an input dataset with one row for each plot cell and splits it into a list of plot datasets as defined by the variable plotGroup. Each plot dataset is then recast (using the cast() function from the reshape package) into matrices defined by the colName and rowName variables. Two matrices are produced for each plot, containing the values of colorValue and sizeValue respectively. To use the included Nightingale metabolomics plot templates, the rowName, colName and plotGroup variables can be merged from the options in the package datafiles ntngale225/249.

Value

A list of two lists of numeric matrices. The matrices in colorList contain values of data$colorValue and those in sizeList contain values of data$sizeValue. The matrice layouts are defined by data$rowName and data$colName and each list has one item for each unique value of data$plotGroup.

Examples

```
data <- merge_template(cetp, "ckb_id")
formatData(data, colorValue = "estimate", sizeValue = "negLog10P",
nightingale = TRUE)
```

---

getLayoutSizes   *Recalculate Layout Sizes for bubbleHeatmap gTree*

Description

This function recalculates the viewport layout dimensions for a bubbleHeatmap tree based on the sizes of the children grobs, and returns the updated tree. This is useful for matching widths exactly to grob dimensions in an updated drawing context, and for resetting any previous layout modifications.

Usage

```
getLayoutSizes(tree, unitBase)
```

Arguments

tree	A bubbleHeatmap [gTree]().
unitBase	The unitBase parameter used when the tree was created.

Value

The input bubbleHeatmap [gTree]() with viewport dimensions matched to the size of the children grobs.

---

matchLayoutHeights   *Match Viewport Layout Dimensions.*

Description

Align the dimensions of the viewport layouts in a list of bubbleHeatmap gTrees. The maximum height/width of a given row/column in any input tree will be applied that row/column for all trees in the list. This enables the plots to be aligned in mult-plot figure layouts, for example the "TopLabels" element can be given the same height in two plots that are to be positioned side by side.

Usage

```
matchLayoutHeights(x, r1 = 1, r2 = 5)

matchLayoutWidths(x, c1 = 1, c2 = 5)
```

Arguments

x	A list of bubbleHeatmap trees
r1, r2	Numeric between 1 and 6, the first (r1) and last (r2) viewport layout rows to be aligned.
c1, c2	Numeric between 1 and 9, the first (c1) and last (c2) viewport layout columns to be aligned.

Value

The input list of bubbleHeatmap trees modified so that the maximum height/width of a given row/column in any input tree is applied to that row/column for all trees in the list, for the rows/columns specified by the r/c parameters.

Functions

- `matchLayoutHeights()`: Align layout row heights across multiple bubbleHeatmap trees.

- `matchLayoutWidths()`: Align layout column widths across multiple bubbleHeatmap trees.

---

merge_template   *Merge Nightingale Figure Templates with User Data*

Description

Custom merge function to help users align their data with the template datasets containing the variable location information for the Nightingale figures. User datasets should contain the variables representing bubble color and size, and a key column to merge rows with the template. The key column should either be named "ukbb_id" and contain the UK Biobank Field ID for each variable (e.g. "23474") or named "ckb_id" and contain the China Kadoorie Biobank variable ID, (e.g. "nmr_bohbut").

Usage

```
merge_template(data, id)
```

Arguments

data	User dataset
id	Name of key column, either "ukbb_id" or "ckb_id"

Value

Merged dataset

Examples

```
data <- merge_template(cetp, "ckb_id")
```

---

metabFigure   *Create Nightingale Figure From Template*

Description

Produce a single gTree object representing plots arranged according to the provided Nightingale metabolomics figure template.

Usage

```
metabFigure(
  data,
  colorValue = "estimate",
  sizeValue = "negLog10P",
  legendHeight = 8,
  context = gpar(cex = 0.8),
  unitBase = unit(0.5, "cm"),
  colorLim = c(NA, NA),
  sizeLim = c(NA, NA)
)
```

Arguments

data	Nightingale results data merged with the ntngale225 or ntngale249 template.
colorValue	String naming the variable of data containing the values to be plotted on the color scale.
sizeValue	String naming the variable of data containing the values to scale bubble sizes.
legendHeight	Numeric, preferred height of legends in multiples of 0.5cm. This will be supplied to pretty function.
context	gpar object to provide drawing context. Will be applied to the parent viewport in the childrenvp slot of the gTree.
unitBase	unit object indicating the size of one plot grid box.
colorLim, sizeLim	
	Numeric vectors indicating limits (min/max) outside which the values of color/size matrices should be truncated. NOTE! These data ranges are separate from the scale ranges, which are taken from the min/max values of colorBreaks and sizeBreaks.

Details

Quick function to produce a final Nightingale figure from results data merged with one of the supplied template datasets. This function wraps the functions formatData, multiPlotInput, bubbleHeatmapList and metabFigurePlot applying default settings.

Value

Produces a complete metabolomics figure consisting of 14 bubbleHeatmap blocks in a pre-set layout. Output is not drawn but returned as a [gTree](#) object which can be edited or drawn using the [grid.draw](#) function. NOTE: A cairo output device with gradient support is necessary to correctly render the color legend.

Examples

```
myData <- merge_template(cetp, "ckb_id")
metabTree <- metabFigure(myData)
grid.draw(metabTree)
```

---

metabFigurePlot   *Arrange List of bubbleHeatmap Plots According to Preset Nightingale*
                               Template.

Description

Produce a single [gTree](#) object representing plots arranged according to the provided Nightingale metabolomics figure template.

Usage

```
metabFigurePlot(treeList)
```

Arguments

treeList   List of bubbleHeatmap trees. This function expects a treeList of particular
                 length, containing trees with specified names and dimensions. The easiest way
                 to generate this is via the [formatData](#), [multiPlotInput](#) and [bubbleHeatmapList](#)
                 functions. See vignette for full details of this workflow.

Value

A [gTree](#) object representing the input bubblehHeatmaps arranged into a Nightingale metabolomics figure.

---

multiPlotInput	*Arrange bubbleHeatmap Arguments in Lists for Building Multiple Plots.*

Description

Build bubbleHeatmap arguments into a list of lists to facilitate building of multiple plots in a single call. It is useful for creating sets of plots with the same data scales, similar plots with minor differences in settings, or the supplied Nightingale plot template.

Usage

```
multiPlotInput(
  colorList,
  sizeList = colorList,
  nightingale = TRUE,
  treeNames = names(colorList),
  context = gpar(cex = 0.8),
  colorSeq = c("#053061", "#2166AC", "#4393C3", "#92C5DE", "#D1E5F0", "#FDDBC7",
    "#F4A582", "#D6604D", "#B2182B", "#67001F"),
  unitBase = unit(0.5, "cm"),
  diameter = 0.8,
  colorLim = c(NA, NA),
  sizeLim = c(NA, NA),
  showTopLabels = TRUE,
  showLeftLabels = TRUE,
  leftLabelsTitles = FALSE,
  showRowBracket = FALSE,
  rowTitles = FALSE,
  showColBracket = FALSE,
  colTitles = FALSE,
  plotTitles = FALSE,
  xTitles = FALSE,
  yTitles = FALSE,
  showColorLegend = TRUE,
  showBubbleLegend = TRUE,
  colorBreaks = NULL,
  sizeBreaks = NULL,
  legendHeight = nrow(colorList[[1]]),
  legendTitles = c(expression("-log"[10] * "P"), "Estimate (SD)")
)
```

Arguments

colorList, sizeList

List of matrices, each item representing one plot, containing values to be plotted as the bubble color and size respectively. If sizeList is not supplied then color values will also be used for size.

nightingale   Logical, apply Nightingale style template?

treeNames   Vector of names for gTrees output by bubbleHeatmap, if not supplied then
                     `names(colorList)` will be used.

context   [gpar](#) object to provide drawing context. Will be applied to the parent viewport
                     in the childrenvp slot of gTrees output by bubbleHeatmap.

colorSeq   Vector of colors to distribute over color scale gradient.

unitBase   [unit](#) object indicating the size of one plot grid square.

diameter   Maximum diameter of bubbles as multiple of unitBase.

colorLim, sizeLim
                     Numeric vectors indicating limits (min/max) beyond which the values of color/size
                     matrices should be truncated.

showTopLabels   Logical vector, should matrix column names be printed above each plot? Should
                     have `length == length(colorList)` or `length == 1` to use same value for all
                     plots.

showLeftLabels   Logical vector, should matrix row names be printed above each plot? Should
                     have `length == length(colorList)` or `length == 1` to use same value for all
                     plots.

leftLabelsTitles
                     Vector of headings to go over the row names of each plot. Character strings or
                     FALSE for no title. A single FALSE value can also be used if no left label titles
                     are required.

showRowBracket   Logical vector, should vertical bracket be printed to the right of each plot (to la-
                     bel a row of plots in a final figure layout)? Should have `length == length(colorList)`
                     or `length == 1` to use same value for all plots.

rowTitles   Vector of labels to be printed to the right of each plot, outside row bracket if
                     present (to label a row of plots in a final figure layout). Character strings or
                     FALSE for no title. A single FALSE value can also be used if no row titles are
                     required.

showColBracket   Logical vector, should horizontal bracket be printed below each plot (to label a
                     column of plots in a final figure layout)? Should have `length == length(colorList)`
                     or `length == 1` to use same value for all plots.

colTitles   Vector of labels to be printed to the right of each plot, outside col bracket if
                     present (to label a column of plots in a final figure layout). Character strings or
                     FALSE for no title. A single FALSE value can also be used if no column titles
                     are required.

plotTitles   Vector of plot titles to be centered above each plot. Character strings or FALSE
                     for no title. A single FALSE value can also be used if no plot titles are required.

xTitles   Vector of x axis titles to be printed left of `LeftLabels`. Character strings or
                     FALSE for no title. A single FALSE value can also be used if no x axis titles are
                     required.

yTitles   Vector of y axis titles to be printed above `TopLabels`. Character strings or
                     FALSE for no title. A single FALSE value can also be used if no y axis titles are
                     required.

showColorLegend, showBubbleLegend

> Logicals, should a legend be produced for color scale/bubble size? If vector with `length == length(colorList)` then values will be input for each plot and any legends produced will be an element of a plot tree. A single TRUE value will build the legends outside the plots as separate trees.

colorBreaks, sizeBreaks

> Character vectors of legend tick labels. If not supplied then these will be generated using pretty to a range outside data values (after values truncated to `colorLim\sizeLim` if applicable).

legendHeight   Numeric, preferred height of legends as multiple of `unitBase`. This will be supplied to pretty function. If sizeBreaks is supplied then this will be ignored and legend height will be `length(sizeBreaks)`.

legendTitles   Character vector of legend titles. First string should be the size legend title, second for the color legend.

Value

List with two elements:

trees List of lists, arguments for bubbleHeatmap

legend List of arguments for bubbleHMLegends

The list returned by this plot can be passed directly to [bubbleHeatmapList](#) to produce a list of plot trees.

Nightingale

For more information about using the Nightingale style template, see the vignette. Requesting Nightingale style values overrides any argument for plotTitles, xTitles, yTitles, showTopLabels, showLeftLabels, leftLabelsTitles, showRowBracket, rowTitles, showColBracket, colTitles, showBubbleLegend or showColorLegend.

See Also

Other build functions: [bubbleHMLegends()](#), [bubbleHeatmapList()](#), [bubbleHeatmap()](#)

---

nightingaleTemplates   *Nightingale Health Metabolomics Plot Template Format*

Description

Template datasets containing grouping and labeling variables to information to plot 225 or 249 variable Nightingale Health metabolomics data.

Usage

```
ntngale225
```

```
ntngale249
```

Format

Data frame with 228/249 rows and 5 variables

trait trait identifier as used in China Kadoorie Biobank, can be used to merge template with results data

traitTitle trait description to assist with merging results

rowName row label for trait in plot matrix

colName column label for trait in plot matrix

plotGroup this variable splits the traits into 10 plot matrices that make up the overall figure

An object of class `data.frame` with 249 rows and 6 columns.

Details

Nightingale Health is a Finnish company which has developed a fully automated platform for deriving over 200 quantitative metabolomics measures from a single blood serum sample using NMR technology, at a comparable cost to standard lipid clinical chemistry. These measures include concentration and composition of 14 lipoprotein subclasses, apolipoproteins, fatty acids, amino acids and glycolysis related metabolites. Over 1M samples have already been processed, including biobank participants, population cohorts and clinical studies, with capacity to process an additional 250K samples per year. Over 100K profiles of UK Biobank participants have recently been released, with data on the remaining individuals to follow. The wide availability of standardised Nightingale datasets in global cohorts facilitates collaborative research and is contributing to the identification of new biomarkers across a range of diseases.

---

widthDetails.bubbleHeatmap

Methods for determining the width/height of bubbleHeatmap plots and legends.

Description

Methods for determining the width/height of bubbleHeatmap plots and legends.

Usage

```
## S3 method for class 'bubbleHeatmap'
widthDetails(x)

## S3 method for class 'bubbleLegend'
widthDetails(x)

## S3 method for class 'colorLegend'
widthDetails(x)

## S3 method for class 'bubbleHeatmap'
```

```
heightDetails(x)

## S3 method for class 'bubbleLegend'
heightDetails(x)

## S3 method for class 'colorLegend'
heightDetails(x)
```

Arguments

x   A bubbleHeatmap, bubbleLegend, or colorLegend object.

Value

Grid unit object representing object width or height.

Functions

- widthDetails(bubbleHeatmap): Get width of bubbleHeatmap
- widthDetails(bubbleLegend): Get width of bubbleLegend
- widthDetails(colorLegend): Get width of colorLegend
- heightDetails(bubbleHeatmap): Get height of bubbleHeatmap
- heightDetails(bubbleLegend): Get height of bubbleLegend
- heightDetails(colorLegend): Get height of colorLegend

Index