

Package ‘ggvenn’

July 22, 2025

Title Draw Venn Diagram by 'ggplot2'
Version 0.1.10
Author Linlin Yan [aut, cre] (ORCID: <<https://orcid.org/0000-0002-4990-6239>>)
Maintainer Linlin Yan <yanlinlin82@gmail.com>
Description An easy-to-use way to draw pretty venn diagram by 'ggplot2'.
Depends dplyr, grid, ggplot2
License MIT + file LICENSE
Encoding UTF-8
RoxygenNote 7.2.3
NeedsCompilation no
Repository CRAN
Date/Publication 2023-03-31 03:50:02 UTC

Contents

data_frame_to_list	1
geom_venn	2
ggvenn	4
list_to_data_frame	7
Index	8

data_frame_to_list	<i>Utility function for data type conversion.</i>
--------------------	---

Description

Utility function for data type conversion.

Usage

data_frame_to_list(x)

Arguments

x A data.frame with logical columns representing sets.

Value

A list of sets.

Examples

```
d <- tibble(name = 1:6,
             A = c(rep(TRUE, 5), FALSE),
             B = rep(c(FALSE, TRUE), each = 3))
print(d)
data_frame_to_list(d)
```

geom_venn	<i>Plot venn diagram as a ggplot layer object. It supports only data frame as input.</i>
-----------	--

Description

Plot venn diagram as a ggplot layer object. It supports only data frame as input.

Usage

```
geom_venn(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  ...,
  set_names = NULL,
  show_percentage = TRUE,
  digits = 1,
  label_sep = ", ",
  count_column = NULL,
  show_outside = c("auto", "none", "always"),
  auto_scale = FALSE,
  fill_color = c("blue", "yellow", "green", "red"),
  fill_alpha = 0.5,
  stroke_color = "black",
  stroke_alpha = 1,
  stroke_size = 1,
  stroke_linetype = "solid",
  set_name_color = "black",
  set_name_size = 6,
  text_color = "black",
  text_size = 4
)
```

Arguments

mapping	Set of aesthetic mappings created by <code>aes()</code> . If specified and <code>inherit.aes = TRUE</code> (the default), it is combined with the default mapping at the top level of the plot. You must supply mapping if there is no plot mapping.
data	A <code>data.frame</code> or a list as input data.
stat	The statistical transformation to use on the data for this layer, as a string.
position	Position adjustment, either as a string naming the adjustment (e.g. "jitter" to use <code>position_jitter</code>), or the result of a call to a position adjustment function. Use the latter if you need to change the settings of the adjustment.
...	Other arguments passed on to <code>layer()</code> . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>colour = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired <code>geom/stat</code> .
set_names	Set names, use column names if omitted.
show_percentage	Show percentage for each set.
digits	The desired number of digits after the decimal point
label_sep	separator character for displaying elements.
count_column	Specify column for element repeat count.
show_outside	Show outside elements (not belongs to any set).
auto_scale	Allow automatically resizing circles according to element counts.
fill_color	Filling colors in circles.
fill_alpha	Transparency for filling circles.
stroke_color	Stroke color for drawing circles.
stroke_alpha	Transparency for drawing circles.
stroke_size	Stroke size for drawing circles.
stroke_linetype	Line type for drawing circles.
set_name_color	Text color for set names.
set_name_size	Text size for set names.
text_color	Text color for intersect contents.
text_size	Text size for intersect contents.

Value

The `ggplot` object to print or save to file.

See Also

`ggvenn`

Examples

```
library(ggvenn)

# use data.frame as input
d <- tibble(value = c(1, 2, 3, 5, 6, 7, 8, 9),
  `Set 1` = c(TRUE, FALSE, TRUE, TRUE, FALSE, TRUE, FALSE, TRUE),
  `Set 2` = c(TRUE, FALSE, FALSE, TRUE, FALSE, FALSE, FALSE, TRUE),
  `Set 3` = c(TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, TRUE, TRUE),
  `Set 4` = c(FALSE, FALSE, FALSE, FALSE, TRUE, TRUE, FALSE, FALSE))

# ggplot grammar
ggplot(d) +
  geom_venn(aes(A = `Set 1`, B = `Set 2`)) +
  coord_fixed() +
  theme_void()
ggplot(d) +
  geom_venn(aes(A = `Set 1`, B = `Set 2`, C = `Set 3`)) +
  coord_fixed() +
  theme_void()
ggplot(d) +
  geom_venn(aes(A = `Set 1`, B = `Set 2`, C = `Set 3`, D = `Set 4`)) +
  coord_fixed() +
  theme_void()

# set fill color
ggplot(d) +
  geom_venn(aes(A = `Set 1`, B = `Set 2`), fill_color = c("red", "blue")) +
  coord_fixed() +
  theme_void()

# hide percentage
ggplot(d) +
  geom_venn(aes(A = `Set 1`, B = `Set 2`), show_percentage = FALSE) +
  coord_fixed() +
  theme_void()

# change precision of percentages
ggplot(d) +
  geom_venn(aes(A = `Set 1`, B = `Set 2`), digits = 2) +
  coord_fixed() +
  theme_void()

# show elements instead of count/percentage
ggplot(d) +
  geom_venn(aes(A = `Set 1`, B = `Set 2`, C = `Set 3`, D = `Set 4`, label = value)) +
  coord_fixed() +
  theme_void()
```

Description

Plot venn diagram as an independent function. It supports both data frame and list as input.

Usage

```
ggvenn(
  data,
  columns = NULL,
  show_elements = FALSE,
  show_percentage = TRUE,
  digits = 1,
  fill_color = c("blue", "yellow", "green", "red"),
  fill_alpha = 0.5,
  stroke_color = "black",
  stroke_alpha = 1,
  stroke_size = 1,
  stroke_linetype = "solid",
  set_name_color = "black",
  set_name_size = 6,
  text_color = "black",
  text_size = 4,
  label_sep = ", ",
  count_column = NULL,
  show_outside = c("auto", "none", "always"),
  auto_scale = FALSE
)
```

Arguments

<code>data</code>	A data.frame or a list as input data.
<code>columns</code>	A character vector use as index to select columns/elements.
<code>show_elements</code>	Show set elements instead of count/percentage.
<code>show_percentage</code>	Show percentage for each set.
<code>digits</code>	The desired number of digits after the decimal point
<code>fill_color</code>	Filling colors in circles.
<code>fill_alpha</code>	Transparency for filling circles.
<code>stroke_color</code>	Stroke color for drawing circles.
<code>stroke_alpha</code>	Transparency for drawing circles.
<code>stroke_size</code>	Stroke size for drawing circles.
<code>stroke_linetype</code>	Line type for drawing circles.
<code>set_name_color</code>	Text color for set names.
<code>set_name_size</code>	Text size for set names.
<code>text_color</code>	Text color for intersect contents.

<code>text_size</code>	Text size for intersect contents.
<code>label_sep</code>	Separator character for displaying elements.
<code>count_column</code>	Specify column for element repeat count.
<code>show_outside</code>	Show outside elements (not belongs to any set).
<code>auto_scale</code>	Allow automatically resizing circles according to element counts.

Value

The ggplot object to print or save to file.

See Also

`geom_venn`

Examples

```
library(ggvenn)

# use list as input
a <- list(`Set 1` = c(1, 3, 5, 7),
          `Set 2` = c(1, 5, 9),
          `Set 3` = c(1, 2, 8),
          `Set 4` = c(6, 7))
ggvenn(a, c("Set 1", "Set 2"))
ggvenn(a, c("Set 1", "Set 2", "Set 3"))
ggvenn(a)

# use data.frame as input
d <- tibble(value = c(1, 2, 3, 5, 6, 7, 8, 9),
            `Set 1` = c(TRUE, FALSE, TRUE, TRUE, FALSE, TRUE, FALSE, TRUE),
            `Set 2` = c(TRUE, FALSE, FALSE, TRUE, FALSE, FALSE, FALSE, TRUE),
            `Set 3` = c(TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, TRUE, TRUE),
            `Set 4` = c(FALSE, FALSE, FALSE, FALSE, TRUE, TRUE, FALSE, FALSE))
ggvenn(d, c("Set 1", "Set 2"))
ggvenn(d, c("Set 1", "Set 2", "Set 3"))
ggvenn(d)

# set fill color
ggvenn(d, c("Set 1", "Set 2"), fill_color = c("red", "blue"))

# hide percentage
ggvenn(d, c("Set 1", "Set 2"), show_percentage = FALSE)

# change precision of percentages
ggvenn(d, c("Set 1", "Set 2"), digits = 2)

# show elements instead of count/percentage
ggvenn(a, show_elements = TRUE)
ggvenn(d, show_elements = "value")
```

list_to_data_frame	<i>Utility function for data type conversion.</i>
--------------------	---

Description

Utility function for data type conversion.

Usage

```
list_to_data_frame(x)
```

Arguments

x	A list of sets.
---	-----------------

Value

A data.frame with logical columns representing sets.

Examples

```
a <- list(A = 1:5, B = 4:6)
print(a)
list_to_data_frame(a)
```

Index

`aes()`, [3](#)

`data_frame_to_list`, [1](#)

`geom_venn`, [2](#)

`ggvenn`, [4](#)

`layer()`, [3](#)

`list_to_data_frame`, [7](#)