

Package ‘ggtangle’

October 10, 2026

Title Draw Network with Data

Version 0.1.4

Description Extends the 'ggplot2' plotting system to support network visualization. Inspired by the 'Method 1' in 'ggtree' (G Yu (2018) <[doi:10.1093/molbev/msy194](https://doi.org/10.1093/molbev/msy194)>), 'ggtangle' is designed to work with network associated data.

Depends R (>= 4.1.0)

Imports ggfun (>= 0.1.7), ggplot2, ggrepel, igraph, rlang, yulab.utils (>= 0.2.2)

Suggests applot, cli, ggiraph, ggnewscale, ggtree, quarto, scatterpie (>= 0.2.4)

VignetteBuilder quarto

License Artistic-2.0

Encoding UTF-8

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Guangchuang Yu [aut, cre] (ORCID: <<https://orcid.org/0000-0002-6485-8781>>)

Maintainer Guangchuang Yu <guangchuangyu@gmail.com>

Repository CRAN

Date/Publication 2026-10-10 04:00:03 UTC

Contents

cnetplot	2
drag_network	4
geom_cnet_label	5
geom_edge	5
geom_edge_interactive	6
geom_edge_text	7
geom_edge_text_interactive	8
ggplot.igraph	9

layout_circular	10
layout_cnet_category_rail	10
layout_fishbone	11
layout_linear	11

Index	13
--------------	-----------

cnetplot	<i>category-item network plot</i>
----------	-----------------------------------

Description

category-item network plot

Usage

```
cnetplot(  
  x,  
  layout = igraph::layout_nicely,  
  showCategory = 5,  
  color_category = "#E5C494",  
  size_category = 1,  
  color_item = "#B3B3B3",  
  size_item = 1,  
  color_edge = "grey",  
  size_edge = 0.5,  
  categorySizeBy = ~itemNum,  
  node_label = "all",  
  node_label_size = NULL,  
  foldChange = NULL,  
  fc_threshold = NULL,  
  highlight = "none",  
  highlight_alpha = 0.3,  
  ...  
)
```

```
## S3 method for class 'list'  
cnetplot(  
  x,  
  layout = "nicely",  
  showCategory = 5,  
  color_category = "#E5C494",  
  size_category = 1,  
  color_item = "#B3B3B3",  
  size_item = 1,  
  color_edge = "grey",  
  size_edge = 0.5,  
  categorySizeBy = ~itemNum,
```

```

    node_label = "all",
    node_label_size = NULL,
    foldChange = NULL,
    fc_threshold = NULL,
    highlight = "none",
    highlight_alpha = 0.3,
    ...
)

```

Arguments

<code>x</code>	input object
<code>layout</code>	network layout
<code>showCategory</code>	categories to display. Use a single number to show the first <i>n</i> categories, a numeric vector to select categories by index (for example, <code>c(1, 10, 53)</code>), or a character vector to select categories by name.
<code>color_category</code>	color of category node
<code>size_category</code>	relative size of the category
<code>color_item</code>	color of item node
<code>size_item</code>	relative size of the item (e.g., genes)
<code>color_edge</code>	color of edge, e.g., "black". If <code>color = "category"</code> , then edges will be colored based on the category information.
<code>size_edge</code>	relative size of edge
<code>categorySizeBy</code>	An expression (e.g., <code>itemNum</code> , <code>p.adjust</code>) or a formula (e.g., <code>~ -log10(p.adjust)</code>) to set the category node size.
<code>node_label</code>	one of 'all', 'none', 'category', 'item', 'exclusive' or 'share'
<code>node_label_size</code>	size of the node label text. When <code>NULL</code> (default) the built-in label size is used.
<code>foldChange</code>	numeric values to color the item (e.g., foldChange of gene expression values)
<code>fc_threshold</code>	threshold for absolute fold change to filter items
<code>highlight</code>	selected category to be highlighted
<code>highlight_alpha</code>	transparent value for not selected to be highlight
<code>...</code>	additional parameters. One important parameter is 'curvature' (default is 0), which can be used to curve the edges (e.g., <code>curvature = 0.2</code>). For circular layouts, this uses a cubic Bezier arc layer with the same radial control-point geometry as the former <code>ggraph::geom_edge_arc()</code> implementation. In circular mode, any non-zero curvature enables this arc layer; the arc geometry is intentionally kept compatible with the former backend.

Details

For list input, `showCategory` accepts either a single integer for the first *n* categories, a numeric vector of category indices, or a character vector of category names.

Examples

```
x <- list(A = letters[1:10], B = letters[5:12])
attr(x, "p.adjust") <- c(A = 0.01, B = 0.2)
p <- cnetplot(x, node_label = "none", categorySizeBy = ~ -log10(p.adjust))
p <- cnetplot(x, showCategory = c(1, 2))
```

drag_network

Drag the nodes of a network to update the layout of the network

Description

Drag the nodes of a network to update the layout of the network

Usage

```
drag_network(p, g = NULL)
```

Arguments

p the network diagram as a ggplot/gg/ggraph object.

g an corresponding igraph object. Default is to extract from the 'ggraph' attribute.

Value

an updated ggplot/gg/ggraph object

Examples

```
## Not run:
library(igraph)
library(gggraph)

flow_info <- data.frame(from = c(1,2,3,3,4,5,6),
                        to = c(5,5,5,6,7,6,7))
g = graph_from_data_frame(flow_info)
p <- gggraph(g, layout='nicely') + geom_node_point() + geom_edge_link()
pp <- drag_network(p)

## End(Not run)
```

geom_cnet_label	<i>geom_cnet_label</i>
-----------------	------------------------

Description

add labels of cnetplot

Usage

```
geom_cnet_label(mapping = NULL, data = NULL, node_label = "all", ...)
```

Arguments

mapping	aes mapping, default is NULL
data	plot data, default is NULL
node_label	which type of node label to be displayed, see also cnetplot
...	parameters that passed to geom_text_repel

Author(s)

Guangchuang Yu

geom_edge	<i>layer to draw edges of a network</i>
-----------	---

Description

layer to draw edges of a network

Usage

```
geom_edge(mapping = NULL, data = NULL, geom = geom_segment, ...)
```

Arguments

mapping	aesthetic mapping, default is NULL
data	data to plot, default is NULL
geom	geometric layer to draw lines
...	additional parameter passed to 'geom'

Value

line segments layer

Examples

```

flow_info <- data.frame(from = LETTERS[c(1,2,3,3,4,5,6)],
                        to = LETTERS[c(5,5,5,6,7,6,7)])

dd <- data.frame(
  label = LETTERS[1:7],
  v1 = abs(rnorm(7)),
  v2 = abs(rnorm(7)),
  v3 = abs(rnorm(7))
)

g = igraph::graph_from_data_frame(flow_info)

p <- ggplot(g) + geom_edge()
library(ggplot2)
library(scatterpie)

p %<+% dd +
  geom_scatterpie(cols = c("v1", "v2", "v3")) +
  geom_text(aes(label=label), nudge_y = .2) +
  coord_fixed()

```

`geom_edge_interactive` *layer to draw edges of a network interactively*

Description

layer to draw edges of a network interactively

Usage

```

geom_edge_interactive(
  mapping = NULL,
  data = NULL,
  geom = ggiraph::geom_segment_interactive,
  ...
)

```

Arguments

<code>mapping</code>	aesthetic mapping, default is <code>NULL</code>
<code>data</code>	data to plot, default is <code>NULL</code>
<code>geom</code>	geometric layer to draw lines
<code>...</code>	additional parameter passed to 'geom'

Value

line segments layer

Examples

```

library(ggiraph)
flow_info <- data.frame(from = LETTERS[c(1,2,3,3,4,5,6)],
                        to = LETTERS[c(5,5,5,6,7,6,7)])

dd <- data.frame(
  label = LETTERS[1:7],
  v1 = abs(rnorm(7)),
  v2 = abs(rnorm(7)),
  v3 = abs(rnorm(7))
)

g = igraph::graph_from_data_frame(flow_info)

p <- ggplot(g) + geom_edge_interactive()
library(ggplot2)
library(scatterpie)

p2 <- p %<+% dd +
  geom_scatterpie(cols = c("v1", "v2", "v3")) +
  geom_text_interactive(aes(label=label, tooltip = label, data_id = label), nudge_y = .2) +
  coord_fixed()

girafe(ggobj = p2)

```

geom_edge_text

*layer to draw edge labels of a network***Description**

layer to draw edge labels of a network

Usage

```

geom_edge_text(
  mapping = NULL,
  data = NULL,
  geom = geom_text,
  angle_calc = "none",
  label_dodge = NULL,
  ...
)

```

Arguments

mapping	aesthetic mapping, default is NULL
data	data to plot, default is NULL
geom	geometric layer to draw text, default is geom_text

angle_calc	how to calculate angle ('along' or 'none')
label_dodge	dodge distance
...	additional parameter passed to 'geom'

Value

text layer

geom_edge_text_interactive

layer to draw edge labels of a network interactively

Description

layer to draw edge labels of a network interactively

Usage

```
geom_edge_text_interactive(
  mapping = NULL,
  data = NULL,
  geom = ggiraph::geom_text_interactive,
  angle_calc = "none",
  label_dodge = NULL,
  ...
)
```

Arguments

mapping	aesthetic mapping, default is NULL
data	data to plot, default is NULL
geom	geometric layer to draw text, default is geom_text
angle_calc	how to calculate angle ('along' or 'none')
label_dodge	dodge distance
...	additional parameter passed to 'geom'

Value

text layer

ggplot.igraph	<i>Plot an igraph object</i>
---------------	------------------------------

Description

Plot an igraph object

Usage

```
## S3 method for class 'igraph'
ggplot(
  data = NULL,
  mapping = aes(),
  layout = "nicely",
  ...,
  environment = parent.frame()
)

## S3 method for class 'mechgraph'
ggplot(
  data = NULL,
  mapping = aes(),
  layout = "nicely",
  ...,
  environment = parent.frame()
)
```

Arguments

data	an igraph object (or a mechgraph object for the ggplot.mechgraph method) to be converted and plotted.
mapping	default aesthetics, default is aes().
layout	network layout. Supported values are the layout names handled by layout_circular , layout_linear , layout_fishbone and other ggtangle/igraph layouts, or a custom layout function. Default is "nicely".
...	additional parameters passed to the layout function.
environment	environment in which to evaluate the mapping, default is parent.frame().

layout_circular	<i>Circular layout</i>
-----------------	------------------------

Description

Circular layout

Usage

```
layout_circular(graph, sort.by = NULL, ...)
```

Arguments

graph	A graph object.
sort.by	The attribute to sort the nodes by. Default is NULL.
...	Additional arguments passed to <code>igraph::layout_in_circle</code> .

Value

A matrix of coordinates.

Author(s)

Guangchuang Yu

layout_cnet_category_rail	<i>Category-rail layout for cnetplot networks</i>
---------------------------	---

Description

Places category vertices on a right-hand rail and item vertices on a left-hand arc. This layout is intended for readable category-item tracing; use `layout = layout_cnet_category_rail` with `cnetplot()`.

Usage

```
layout_cnet_category_rail(graph, ...)
```

Arguments

graph	An <code>igraph</code> object with <code>.isCategory</code> vertex attribute.
...	Unused additional arguments.

Value

A two-column layout matrix.

layout_fishbone	<i>Glycan Layout Algorithm (Fishbone/SNFG-like)</i>
-----------------	---

Description

Calculates node coordinates for a glycan structure to mimic SNFG style. Supports customizable direction and branch length.

Usage

```
layout_fishbone(graph, direction = "left", length = 1, angle_sep = 30, ...)
```

Arguments

graph	An igraph object.
direction	The direction of the main chain growth ("left", "right", "up", "down"). Default is "left".
length	The distance between nodes. Default is 1.
angle_sep	The angle separation for branches in degrees. Default is 30.
...	Additional arguments.

Value

A matrix of x, y coordinates.

layout_linear	<i>Linear layout</i>
---------------	----------------------

Description

Linear layout

Usage

```
layout_linear(graph, sort.by = NULL, ...)
```

Arguments

graph	A graph object.
sort.by	The attribute to sort the nodes by. Default is NULL.
...	Additional arguments.

Value

A matrix of coordinates.

Author(s)

Guangchuang Yu

Index

`cnetplot`, [2](#), [5](#)
`cnetplot()`, [10](#)

`drag_network`, [4](#)

`geom_cnet_label`, [5](#)
`geom_edge`, [5](#)
`geom_edge_interactive`, [6](#)
`geom_edge_text`, [7](#)
`geom_edge_text_interactive`, [8](#)
`ggplot.igraph`, [9](#)
`ggplot.mechgraph` (`ggplot.igraph`), [9](#)

`layout_circular`, [9](#), [10](#)
`layout_cnet_category_rail`, [10](#)
`layout_fishbone`, [9](#), [11](#)
`layout_linear`, [9](#), [11](#)