

# Package ‘gggenes’

December 14, 2025

**Title** Draw Gene Arrow Maps in 'ggplot2'

**Version** 0.6.0

**Description** A 'ggplot2' extension for drawing gene arrow maps.

**Depends** R (>= 3.6)

**Imports** grid (>= 3.3.0), ggplot2 (>= 2.2.1), ggfittext (>= 0.8.0),  
rlang (>= 0.2.0), lifecycle, cli (>= 3.6.1)

**License** GPL-2

**LazyData** true

**RoxygenNote** 7.3.3

**URL** <https://wilcox.org/gggenes/>

**BugReports** <https://github.com/wilcox/gggenes/issues/>

**Suggests** testthat (>= 3.0.0), knitr, rmarkdown, vdiff, spelling

**VignetteBuilder** knitr

**Encoding** UTF-8

**Language** en-US

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** David Wilkins [aut, cre],  
Zachary Kurtz [ctb]

**Maintainer** David Wilkins <david@wilcox.org>

**Repository** CRAN

**Date/Publication** 2025-12-14 06:20:02 UTC

## Contents

example_dummies	2
example_features	2
example_genes	3
example_terminators	4

geom_feature . . . . .	4
geom_feature_label . . . . .	6
geom_gene_arrow . . . . .	7
geom_gene_label . . . . .	9
geom_subgene_arrow . . . . .	10
geom_subgene_label . . . . .	12
geom_terminator . . . . .	13
geom_terminator_label . . . . .	15
make_alignment_dummies . . . . .	16
theme_genes . . . . .	17

<b>Index</b>	<b>18</b>
--------------	-----------

---

example_dummies	<i>A set of example dummy alignment genes.</i>
-----------------	--

---

**Description**

Dummy genes, similar to those generated with `make_alignment_dummies()`, for example purposes only.

**Usage**

`example_dummies`

**Format**

A data frame with eight rows and three variables:

- molecule** the genome
- start** the start position of the dummy
- end** the end position of the dummy
- gene** the name of the dummy gene

---

example_features	<i>A set of example genetic features.</i>
------------------	---

---

**Description**

Genetic features for example purposes only.

**Usage**

`example_features`

`example_features_polar`

**Format**

A data frame with 23 rows and five variables:

**molecule** the genome

**name** the name of the feature

**type** the type of the feature

**position** the position of the feature

**forward** is the feature oriented, and if so in the forward direction?

An object of class `data.frame` with 6 rows and 5 columns.

---

example_genes	<i>A set of example genes.</i>
---------------	--------------------------------

---

**Description**

Genes for example purposes only.

**Usage**

example\_genes

example\_genes\_polar

example\_subgenes

example\_subgenes\_polar

**Format**

A data frame with 72 rows and six variables:

**molecule** the genome

**gene** the name of the gene

**start** the start position of the gene

**end** the end position of the gene

**strand** the strand of the gene

**orientation** the orientation of the gene

example\_subgenes (143 rows) also contains:

**subgene** the name of the subgene

**from** the start position of the subgene segment

**to** the end position of the subgene segment

An object of class `data.frame` with 18 rows and 6 columns.

An object of class `data.frame` with 143 rows and 9 columns.

An object of class `data.frame` with 40 rows and 9 columns.

---

example_terminators	<i>A set of example terminators.</i>
---------------------	--------------------------------------

---

### Description

Terminators for example purposes only.

### Usage

```
example_terminators
example_terminators_polar
```

### Format

A data frame with two rows and three variables:

**molecule** the genome  
**name** the name of the terminator  
**position** the position of the terminator

An object of class `data.frame` with 2 rows and 3 columns.

---

geom_feature	<i>A 'ggplot2' geom to draw point genetic features</i>
--------------	--

---

### Description

`geom_feature()` draws lines to indicate the positions of point genetic features, for example restriction sites, origins of replication or transcription start sites.

### Usage

```
geom_feature(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = FALSE,
  feature_height = unit(3, "mm"),
  feature_width = unit(3, "mm"),
  arrowhead_width = unit(2, "mm"),
  ...
)
```

## Arguments

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...  
 As standard for ggplot2, inherit.aes is set to FALSE by default, as features are not likely to share any plot aesthetics other than y.

feature\_height `grid::unit()` object giving the height of a feature above the molecule line. Can be set as a negative value to draw features below the line. Defaults to 3 mm.

feature\_width `grid::unit()` object giving the width of a feature (distance from the elbow to the tip of the arrow). Only relevant for oriented features. Defaults to 3 mm.

arrowhead\_width `grid::unit()` object giving the width of the arrowhead indicating the direction of an oriented feature. Only relevant for oriented features. Defaults to 2 mm.

## Details

Features are drawn as vertical lines extending from the horizontal line representing the molecule. The position of the feature is expressed with the x aesthetic. Optionally, the forward aesthetic can be used to specify an orientation for the feature (e.g. the direction of transcription), in which case an angled arrowhead will be added. The forward aesthetic assumes that the x-axis is oriented in the normal direction, i.e. increasing from left to right; if it is not, the values in forward will need to be inverted manually.

## Aesthetics

- x (required; position of the feature)
- y (required; molecule)
- forward (optional; if TRUE, or a value coercible to TRUE, the feature will be drawn with an arrowhead pointing right, if FALSE, pointing left, if NA, the feature will be drawn as a vertical line)
- alpha
- colour
- linetype
- linewidth (the former size aesthetic has been deprecated and will be removed in future versions)

## See Also

[geom\\_feature\\_label\(\)](#), [geom\\_terminator\(\)](#)

## Examples

```
ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
                                             y = molecule, fill = gene)) +
  geom_gene_arrow() +
  geom_feature(data = example_features, ggplot2::aes(x = position, y = molecule,
                                                    forward = forward)) +
  ggplot2::facet_wrap(~ molecule, scales = "free")
```

---

geom_feature_label	A 'ggplot2' geom to add text labels to point genetic features
--------------------	---

---

## Description

geom\_feature\_label() adds text labels to features drawn with geom\_feature().

## Usage

```
geom_feature_label(  
  mapping = NULL,  
  data = NULL,  
  stat = "identity",  
  position = "identity",  
  na.rm = FALSE,  
  show.legend = FALSE,  
  inherit.aes = FALSE,  
  feature_height = unit(4, "mm"),  
  label_height = unit(3, "mm"),  
  ...  
)
```

## Arguments

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...

As standard for ggplot2. inherit.aes is set to FALSE by default, as features are not likely to share any plot aesthetics other than y.

**feature\_height** grid::unit() object giving the height of the feature being labelled, and hence the distance of the label above or below the molecule line. Can be set as a negative value for features drawn below the line. Defaults to 4 mm, to align labels with the default height of geom\_feature().

**label\_height** grid::unit() object giving the height of the label text. Defaults to 3 mm.

## Details

Standard 'ggplot2' aesthetics for text are supported (see Aesthetics).

## Aesthetics

- x (required; position of the feature)
- y (required; molecule)
- label (required; the label text)
- forward (optional; will draw text in the appropriate location for features with angled arrow-heads)
- colour

- size
- alpha
- family
- fontface
- angle

### See Also

[geom\\_feature\(\)](#)

### Examples

```
ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
                                             y = molecule, fill = gene)) +
  geom_gene_arrow() +
  geom_feature(data = example_features, ggplot2::aes(x = position, y = molecule,
                                                    forward = forward)) +
  geom_feature_label(data = example_features,
                    ggplot2::aes(x = position, y = molecule, label = name,
                                forward = forward)) +
  ggplot2::facet_wrap(~ molecule, scales = "free")
```

---

geom\_gene\_arrow

*A 'ggplot2' geom to draw genes as arrows*

---

### Description

`geom_gene_arrow()` draws genes as arrows, allowing gene maps to be drawn.

### Usage

```
geom_gene_arrow(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  arrowhead_width = grid::unit(4, "mm"),
  arrowhead_height = grid::unit(4, "mm"),
  arrow_body_height = grid::unit(3, "mm"),
  ...
)
```

## Arguments

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...

As standard for ggplot2.

arrowhead\_width

grid::unit() object giving the width of the arrowhead. Defaults to 4 mm. If the gene is drawn smaller than this width, only the arrowhead will be drawn, compressed to the length of the gene.

arrowhead\_height

grid::unit() object giving the height of the arrowhead. Defaults to 4 mm.

arrow\_body\_height

grid::unit() object giving the height of the body of the arrow. Defaults to 3 mm.

## Details

This geom draws genes as arrows along a horizontal line representing the molecule. The start and end locations of the gene are expressed with the `xmin` and `xmax` aesthetics, while the molecule can be specified with the `y` aesthetic. Optionally, an additional forward aesthetic can be used to reverse the orientation of some or all genes from that implied by `xmin` and `xmax`.

Unless the plot is faceted with a free x scale, all the molecules will share a common x axis. This means that if the locations are very different across different molecules, the genes might appear very small and squished together with a lot of unnecessary empty space. To get around this, either facet the plot with `scales = "free_x"`, or normalise the gene locations if their exact locations are not important.

See `make_alignment_dummies()` for a method to align genes between molecules.

## Aesthetics

- `xmin,xmax` (start and end of the gene; will be used to determine gene orientation)
- `y` (molecule)
- `forward` (if any value that is not TRUE, or coercible to TRUE, the gene arrow will be drawn in the opposite direction to that determined by `xmin` and `xmax`)
- `alpha`
- `colour`
- `fill`
- `linetype`
- `linewidth` (the former `size` aesthetic has been deprecated and will be removed in future versions)

## See Also

`theme_genes()`, `make_alignment_dummies()`, `geom_gene_label()`



**Examples**

```
ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
                                              y = molecule, fill = gene)) +
  geom_gene_arrow() +
  ggplot2::facet_wrap(~ molecule, scales = "free")
```

---

geom_gene_label	<i>A 'ggplot2' geom to add text labels to gene arrows</i>
-----------------	---

---

**Description**

geom\_gene\_label() can be used to add a text label to genes drawn with geom\_gene\_arrow().

**Usage**

```
geom_gene_label(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  na.rm = FALSE,
  show.legend = FALSE,
  inherit.aes = TRUE,
  padding.x = grid::unit(1, "mm"),
  padding.y = grid::unit(0.1, "lines"),
  align = "centre",
  min.size = 4,
  grow = F,
  reflow = F,
  height = grid::unit(3, "mm"),
  ...
)
```

**Arguments**

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...	Standard geom arguments as for ggplot2::geom_text().
padding.x, padding.y	grid::unit() object, giving horizontal or vertical padding around the text. Defaults to 1 mm and 0.1 lines respectively.
align	Where inside the gene to place the text label. Default is 'centre'; other options are 'left' and 'right'.
min.size	Minimum font size, in points. If provided, text that would need to be shrunk below this size to fit inside the gene arrow will not be drawn. Defaults to 4 pt.
grow	If TRUE, text will be grown as well as shrunk to fill the arrow.

reflow	If TRUE, text will be reflowed (wrapped) to better fit the arrow.
height	grid::unit() object giving the maximum height of the text. Defaults to 3 mm, which is the default height of gene arrows drawn with geom_gene_arrow().

### Details

geom\_gene\_label() uses the 'ggfittext' package to fit text to genes. All text drawing options available in ggfittext::geom\_fit\_text() (growing, reflowing, etc.) are also available here. For full details on how these options work, see the documentation for ggfittext::geom\_fit\_text().

Standard 'ggplot2' aesthetics for text are supported (see Aesthetics).

### Aesthetics

- xmin,xmax (start and end of the gene; required)
- y (molecule; required)
- label (the label text; required)
- colour
- size
- alpha
- family
- fontface
- angle

### See Also

geom\_gene\_arrow

### Examples

```
ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
  y = molecule, fill = gene, label = gene)) +
  geom_gene_arrow() +
  geom_gene_label() +
  ggplot2::facet_wrap(~ molecule, ncol = 1, scales = "free") +
  theme_genes()
```

---

geom\_subgene\_arrow      A 'ggplot2' geom to draw subgene segments of gene arrows

---

### Description

geom\_subgene\_arrow() draws subgenes segments within gene arrows drawn with geom\_gene\_arrow().

**Usage**

```
geom_subgene_arrow(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  arrowhead_width = grid::unit(4, "mm"),
  arrowhead_height = grid::unit(4, "mm"),
  arrow_body_height = grid::unit(3, "mm"),
  ...
)
```

**Arguments**

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...

As standard for 'ggplot2'.

arrowhead\_width

grid::unit() object giving the width of the arrowhead. Defaults to 4 mm. If the gene is drawn smaller than this width, only the arrowhead will be drawn, compressed to the length of the gene.

arrowhead\_height

grid::unit() object giving the height of the arrowhead. Defaults to 4 mm.

arrow\_body\_height

grid::unit() object giving the height of the body of the arrow. Defaults to 3 mm.

**Details**

The start and end locations of the subgene are given with the xsubmin and xsubmax aesthetics. geom\_subgene\_arrow() requires some information about the 'parent' gene, provided with the same aesthetics used for geom\_gene\_arrow(): start and end locations of the 'parent' gene with the xmin and xmax aesthetics, the molecule with the y aesthetic, and optionally the direction with the forward aesthetic. If the geometry of the parent gene has been changed with arrowhead\_width, arrowhead\_height or arrow\_body\_height, identical parameters should be given to geom\_subgene\_arrow().

**Aesthetics**

- xmin,xmax (start and end of the gene; will be used to determine gene orientation)
- xsubmin,xsubmax (start and end of subgene segment). Should be consistent with xmin/xmax
- y (molecule)
- forward (if FALSE, or coercible to FALSE, the gene arrow will be drawn in the opposite direction to that determined by xmin and xmax)
- alpha
- colour

- fill
- linetype
- linewidth (the former size aesthetic has been deprecated and will be removed in future versions)

### See Also

[geom\\_gene\\_arrow\(\)](#), [geom\\_subgene\\_arrow\(\)](#)

### Examples

```
ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
                                             y = molecule)) +
  geom_gene_arrow() +
  geom_subgene_arrow(data = example_subgenes,
                    ggplot2::aes(xmin = start, xmax = end, xsubmin = from, xsubmax = to,
                                y = molecule, fill = gene)) +
  ggplot2::facet_wrap(~ molecule, scales = "free")
```

---

geom\_subgene\_label      *A 'ggplot2' geom to add text labels to subgenes*

---

### Description

`geom_subgene_label()` can be used to add a text label to subgenes drawn with `geom_subgene_arrow()`.

### Usage

```
geom_subgene_label(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  na.rm = FALSE,
  show.legend = FALSE,
  inherit.aes = TRUE,
  padding.x = grid::unit(1, "mm"),
  padding.y = grid::unit(0.1, "lines"),
  align = "centre",
  min.size = 4,
  grow = F,
  reflow = F,
  height = grid::unit(3, "mm"),
  ...
)
```

**Arguments**

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...	Standard geom arguments as for <code>ggplot2::geom_text()</code> .
padding.x, padding.y	<code>grid::unit()</code> object, giving horizontal or vertical padding around the text. Defaults to 1 mm and 0.1 lines respectively.
align	Where inside the subgene to place the text label. Default is 'centre'; other options are 'left' and 'right'.
min.size	Minimum font size, in points. If provided, text that would need to be shrunk below this size to fit inside the subgene will not be drawn. Defaults to 4 pt.
grow	If TRUE, text will be grown as well as shrunk to fill the subgene.
reflow	If TRUE, text will be reflowed (wrapped) to better fit the subgene.
height	<code>grid::unit()</code> object giving the maximum height of the text. Defaults to 3 mm, which is the default height of gene arrows (and therefore of subgenes) drawn with <code>geom_gene_arrow()</code> .

**Details**

`geom_subgene_label()` uses the 'ggfittest' package to fit text to genes. All text drawing options available in `ggfittest::geom_fit_text()` (growing, reflowing, etc.) are also available here. For full details on how these options work, see the documentation for `ggfittest::geom_fit_text()`.

Standard 'ggplot2' aesthetics for text are supported (see Aesthetics.)

**Aesthetics**

- xsubmin, xsubmax (start and end of the subgene; required)
- y (molecule; required)
- colour
- size
- alpha
- family
- fontface
- angle

---

geom\_terminator

A 'ggplot2' geom to draw transcription terminators

---

**Description**

`geom_terminator()` draws a 'T-shaped' glyph representing the position of a transcription terminator.

**Usage**

```
geom_terminator(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = FALSE,
  terminator_height = unit(3, "mm"),
  terminator_width = unit(3, "mm"),
  ...
)
```

**Arguments**

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...  
 As standard for ggplot2. inherit.aes is set to FALSE by default, as terminators are not likely to share any plot aesthetics other than y.

terminator\_height  
 grid::unit() object giving the height of the vertical 'pillar' of the terminator glyph above the molecule line. Can be set as a negative value to draw terminators below the line. Defaults to 3 mm.

terminator\_width  
 grid::unit() object giving the width of the horizontal 'beam' of the terminator glyph. Defaults to 3 mm.

**Aesthetics**

- x (required; position of the terminator)
- y (required; molecule)
- alpha
- color
- linetype
- linewidth (the former size aesthetic has been deprecated and will be removed in future versions)

**See Also**

[geom\\_terminator\\_label\(\)](#), [geom\\_feature\(\)](#)

**Examples**

```
ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
                                             y = molecule, fill = gene)) +
  geom_gene_arrow() +
  geom_terminator(data = example_terminators, ggplot2::aes(x = position, y = molecule)) +
  ggplot2::facet_wrap(~ molecule, scales = "free")
```

---

geom\_terminator\_label A 'ggplot2' geom to add text labels to transcription terminators

---

## Description

geom\_terminator\_label() adds text labels to terminators drawn with geom\_terminator().

## Usage

```
geom_terminator_label(  
  mapping = NULL,  
  data = NULL,  
  stat = "identity",  
  position = "identity",  
  na.rm = FALSE,  
  show.legend = FALSE,  
  inherit.aes = FALSE,  
  terminator_height = unit(4, "mm"),  
  label_height = unit(3, "mm"),  
  ...  
)
```

## Arguments

mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...

As standard for ggplot2. inherit.aes is set to FALSE by default, as terminators are not likely to share any plot aesthetics other than y.

terminator\_height

grid::unit() object giving the height of the terminator being labelled, and hence the distance of the label above or below the molecule line. Can be set as a negative value for terminators drawn below the line. Defaults to 4 mm, to align labels with the default height of geom\_terminator().

label\_height grid::unit() object giving the height of the label text. Defaults to 3 mm.

## Details

Standard 'ggplot2' aesthetics for text are supported (see Aesthetics).

## Aesthetics

- x (required; position of the terminator)
- y (required; molecule)
- label (required; the label text)
- colour
- size

- alpha
- family
- fontface
- angle

### See Also

[geom\\_terminator\(\)](#)

### Examples

```
ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
                                             y = molecule, fill = gene)) +
  geom_gene_arrow() +
  geom_terminator(data = example_terminators,
                  ggplot2::aes(x = position, y = molecule)) +
  geom_terminator_label(data = example_terminators,
                        ggplot2::aes(x = position, y = molecule, label = name)) +
  ggplot2::facet_wrap(~ molecule, scales = "free")
```

---

make\_alignment\_dummies

*Prepare dummy data to visually align a single gene across faceted molecules*

---

### Description

make\_alignment\_dummies() helps you to visually align genes across molecules that have been faceted with a free x scale. The output of this function is a data frame of dummy genes. If these dummy genes are added to a 'ggplot2' plot with ggplot2::geom\_blank(), they will extend the x axis range in such a way that the start or end of a selected gene is visually aligned across the facets.

### Usage

```
make_alignment_dummies(data, mapping, on, side = "left")
```

### Arguments

data	Data frame of genes. This is almost certainly the same data frame that will later be passed to ggplot2::ggplot().
mapping	Aesthetic mapping, created with ggplot2::aes(). Must contain the following aesthetics: xmin, xmax, y, and id (a unique identifier for each gene).
on	Name of gene to be visually aligned across facets. This gene must be present in 'data', in the column mapped to the id aesthetic.
side	Should the visual alignment be of the 'left' (default) or 'right' side of the gene?



## Examples

```
dummies <- make_alignment_dummies(example_genes, ggplot2::aes(xmin = start,
  xmax = end, y = molecule, id = gene), on = "genE")

ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
  y = molecule, fill = gene)) +
  geom_gene_arrow() +
  ggplot2::geom_blank(data = dummies) +
  ggplot2::facet_wrap(~ molecule, scales = "free", ncol = 1)
```

---

theme\_genes

*A 'ggplot2' theme for drawing gene maps*

---

## Description

This theme removes extraneous plot elements for drawing an 'arrows-on-a-string' style gene map in 'ggplot2'. `theme_genes_flipped()` is like `theme_genes()`, but for flipped coordinates.

## Usage

```
theme_genes()

theme_genes_flipped()
```

## Details

This theme removes strip text (the text that labels facets when you use `ggplot2::facet_wrap()` or `ggplot2::facet_grid()`). This makes it easier to draw molecules on different x scales by setting the y aesthetic to the molecule, then faceting with `facet_grid(~ molecule, scales = "free")`.

## See Also

[geom\\_gene\\_arrow\(\)](#)

## Examples

```
ggplot2::ggplot(example_genes, ggplot2::aes(xmin = start, xmax = end,
  y = molecule, fill = gene)) +
  geom_gene_arrow() +
  ggplot2::facet_wrap(~ molecule, scales = "free") +
  theme_genes()
```

# Index

## \* datasets

- example\_dummies, [2](#)
- example\_features, [2](#)
- example\_genes, [3](#)
- example\_terminators, [4](#)

  

- example\_dummies, [2](#)
- example\_features, [2](#)
- example\_features\_polar
  - (example\_features), [2](#)
- example\_genes, [3](#)
- example\_genes\_polar(example\_genes), [3](#)
- example\_subgenes(example\_genes), [3](#)
- example\_subgenes\_polar(example\_genes),  
[3](#)
- example\_terminators, [4](#)
- example\_terminators\_polar
  - (example\_terminators), [4](#)

  

- geom\_feature, [4](#)
- geom\_feature(), [7](#), [14](#)
- geom\_feature\_label, [6](#)
- geom\_feature\_label(), [5](#)
- geom\_gene\_arrow, [7](#)
- geom\_gene\_arrow(), [12](#), [17](#)
- geom\_gene\_label, [9](#)
- geom\_gene\_label(), [8](#)
- geom\_subgene\_arrow, [10](#)
- geom\_subgene\_label, [12](#)
- geom\_subgene\_label(), [12](#)
- geom\_terminator, [13](#)
- geom\_terminator(), [5](#), [16](#)
- geom\_terminator\_label, [15](#)
- geom\_terminator\_label(), [14](#)

  

- make\_alignment\_dummies, [16](#)
- make\_alignment\_dummies(), [8](#)

  

- theme\_genes, [17](#)
- theme\_genes(), [8](#)
- theme\_genes\_flipped(theme\_genes), [17](#)