Package ‘gggenes’

June 24, 2019

Title Draw Gene Arrow Maps in 'ggplot2'
Version 0.4.0
Description Provides a 'ggplot2' geom and helper functions for drawing gene arrow maps.
Depends R (>= 3.3.0)
Imports grid (>= 3.3.0), ggplot2 (>= 2.2.1), ggfittext (>= 0.8.0), rlang (>= 0.2.0)
License GPL-2
LazyData true
RoxygenNote 6.1.1
URL https://wilkox.org/gggenes
BugReports https://github.com/wilkox/gggenes/issues
Suggests testthat, knitr, rmarkdown, vdiffr
VignetteBuilder knitr
Encoding UTF-8
NeedsCompilation no
Author David Wilkins [aut, cre], Zachary Kurtz [ctb]
Maintainer David Wilkins <david@wilkox.org>
Repository CRAN
Date/Publication 2019-06-24 05:10:11 UTC

R topics documented:

example_genes .............................................................. 2
geom_gene_arrow .......................................................... 3
geom_gene_label ........................................................... 4
geom_subgene_arrow ......................................................... 6
geom_subgene_label ........................................................ 7
example_genes

A set of example genes.

Description

Genes for example purposes only.

Usage

example_genes
example_subgenes

Format

A data frame with 118 rows and four 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

example_subgenes (237 rows) also contains:

- **subgeme**: the name of the subgene
- **from**: the start position of the subgene segment
- **strand**: the end position of the subgene segment
Description

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

Usage

```r
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
- size

See Also

theme_genes(), make_alignment_dummies(), geom_gene_label()

Examples

```r
ggplotR::ggplot(example_genesL ggplotR::aes(xmin = startL xmax = endL y = moleculeL fill = gene)) +
  geom_gene_arrow() +
  ggplotR::facet_wrap(~ moleculeL scales = "free")
```

Description

`geom_gene_label()` can be used to add a text label to genes drawn with `geom_gene_arrow()`.

Usage

```r
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"),
  ...)
```
geom_gene_label

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)
- `colour`
- `size`
- `alpha`
- `family`
- `fontface`
- `angle`

See Also

- `geom_gene_arrow`

Examples

```r
  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 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
geom_subgene_label

- colour
- fill
- linetype
- size

See Also
gem_gene_arrow(), geom_subgene_label()

Examples

ggplotR::ggplot(example_genesL ggplotR::aes(xmin = startL xmax = endL
    y = molecule)) +
geom_gene_arrow() +
geom_subgene_arrow(data = example_subgenes,
    ggplotR::aes(xmin = startL xmax = endL xsubmin = fromL xsubmax = toL
    y = moleculeL fill = gene)) +
ggplotR::facet_wrap(~ moleculeL 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 = NULLL data = NULLL stat = "identityL
    position = "identityL na.rm = FALSEL show.legend = FALSEL
    inherit.aes = TRUEL padding.x = grid::unit(1L "mm"),
    padding.y = grid::unit(0.1L "lines"), align = "centreL
    min.size = 4L grow = F, reflow = F, height = grid::unit(3L "mm"),
    ...)}

Arguments

t mapping, data, stat, position, na.rm, show.legend, inherit.aes, ...
  Standard geom arguments as for ggplot2::geom_text().

t padding.x, padding.y
  grid::unit() object, giving horizontal or vertical padding around the text. De-
  faults to 1 mm and 0.1 lines respectively.

t align
  Where inside the subgene to place the text label. Default is 'centre'; other op-
  tions 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**
ggrid::unit() 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 geom_gene_arrow().

**Details**

geom_subgene_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**

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

---

**gggenes**

'gggenes': provides a 'ggplot2' geom and helper functions for drawing gene arrow maps.

**Description**

This package provides a 'ggplot2' geom, geom_gene_arrow a theme, theme_genes, and a helper function for visually aligning genes, make_alignment_dummies.
**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 `ggplot::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**

```r
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**

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

```r
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'.

Usage

```r
theme_genes()
```

Details

This theme removes strip text (the text that labels facets when you use `ggplot2::facet_wrap()` or `ggplot::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

```r
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

*Topic **datasets**
  example_genes, 2
  example_subgenes (example_genes), 2

  geom_gene_arrow, 3
  geom_gene_arrow(), 7, 10
  geom_gene_label, 4
  geom_gene_label(), 4
  geom_subgene_arrow, 6
  geom_subgene_label, 7
  geom_subgene_label(), 7
  gggenes, 8
  gggenes-package (gggenes), 8

  make_alignment_dummies, 9
  make_alignment_dummies(), 4

  theme_genes, 10
  theme_genes(), 4