

Package ‘ggstance’

November 16, 2016

Title Horizontal 'ggplot2' Components

Version 0.3

Description A 'ggplot2' extension that provides flipped components: horizontal versions of 'Stats' and 'Geoms', and vertical versions of 'Positions'.

Depends R (>= 3.2.2)

Imports ggplot2 (>= 2.2), purrr (>= 0.2), lazyeval, plyr

Suggests magrittr, testthat, vdiff

License GPL-3

LazyData true

RoxygenNote 5.0.1

Collate 'flip-aes.R' 'geom-barh.R' 'legend-draw.R' 'geom-boxplot.R'
'geom-crossbarh.R' 'geom-errorbarh.R' 'geom-histogramh.R'
'geom-linerangeh.R' 'geom-pointrangeh.R' 'geom-violinh.R'
'ggstance.R' 'position-dodgev.R' 'position-jitterdodgev.R'
'position-stackv.R' 'position.R' 'stat-bin.R'
'stat-boxplot.R' 'stat-counth.R' 'stat-summaryh.R'
'stat-xdensity.R'

NeedsCompilation no

Author Lionel Henry [aut, cre],
Hadley Wickham [aut],
Winston Chang [aut],
RStudio [cph]

Maintainer Lionel Henry <lionel@rstudio.com>

Repository CRAN

Date/Publication 2016-11-16 08:16:37

R topics documented:

draw_key	2
geom_barh	3

geom_boxplot	4
geom_crossbar	5
geom_histogram	6
geom_violin	7
position_dodge	8
stat_bin	9
stat_boxplot	10
stat_count	11
stat_summary	12
stat_xdensity	13

Index	15
--------------	-----------

draw_key	<i>Horizontal key drawing functions</i>
----------	---

Description

Horizontal key drawing functions

Usage

```
draw_key_hpath(data, params, size)
```

```
draw_key_pointrangeh(data, params, size)
```

```
draw_key_crossbarh(data, params, size)
```

```
draw_key_boxplot(data, params, size)
```

Arguments

data	A single row data frame containing the scaled aesthetics to display in this key
params	A list of additional parameters supplied to the geom.
size	Width and height of key in mm.

Value

A grid grob.

geom_barh	<i>Bars, rectangles with bases on y-axis</i>
-----------	--

Description

Horizontal version of [geom_bar\(\)](#).

Usage

```
geom_barh(mapping = NULL, data = NULL, stat = "count",
           position = "stackv", ..., width = NULL, binwidth = NULL,
           na.rm = FALSE, show.legend = NA, inherit.aes = TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by aes or aes_ . 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	The data to be displayed in this layer. There are three options: If <code>NULL</code> , the default, the data is inherited from the plot data as specified in the call to ggplot . A <code>data.frame</code> , or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify for which variables will be created. A function will be called with a single argument, the plot data. The return value must be a <code>data.frame</code> , and will be used as the layer data.
stat	The statistical transformation to use on the data for this layer, as a string.
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
...	other arguments passed on to layer . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired geom/stat.
width	Bar width. By default, set to 90% of the resolution of the data.
binwidth	<code>geom_bar</code> no longer has a <code>binwidth</code> argument - if you use it you'll get an warning telling to you use geom_histogram instead.
na.rm	If <code>FALSE</code> , the default, missing values are removed with a warning. If <code>TRUE</code> , missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? <code>NA</code> , the default, includes if any aesthetics are mapped. <code>FALSE</code> never includes, and <code>TRUE</code> always includes.
inherit.aes	If <code>FALSE</code> , overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders .

geom_boxploth

*Horizontal box and whiskers plot.***Description**

Horizontal version of `geom_boxplot()`.

Usage

```
geom_boxploth(mapping = NULL, data = NULL, stat = "boxploth",
  position = "dodgev", ..., outlier.colour = NULL, outlier.color = NULL,
  outlier.shape = 19, outlier.size = 1.5, outlier.stroke = 0.5,
  notch = FALSE, notchwidth = 0.5, varwidth = FALSE, na.rm = FALSE,
  show.legend = NA, inherit.aes = TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by <code>aes</code> or <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	The data to be displayed in this layer. There are three options: If <code>NULL</code> , the default, the data is inherited from the plot data as specified in the call to <code>ggplot</code> . A <code>data.frame</code> , or other object, will override the plot data. All objects will be fortified to produce a data frame. See <code>fortify</code> for which variables will be created. A function will be called with a single argument, the plot data. The return value must be a <code>data.frame</code> ., and will be used as the layer data.
stat	The statistical transformation to use on the data for this layer, as a string.
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
...	other arguments passed on to <code>layer</code> . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired <code>geom/stat</code> .
outlier.colour, outlier.color, outlier.shape, outlier.size, outlier.stroke	Default aesthetics for outliers. Set to <code>NULL</code> to inherit from the aesthetics used for the box. In the unlikely event you specify both US and UK spellings of colour, the US spelling will take precedence.
notch	if <code>FALSE</code> (default) make a standard box plot. If <code>TRUE</code> , make a notched box plot. Notches are used to compare groups; if the notches of two boxes do not overlap, this suggests that the medians are significantly different.
notchwidth	for a notched box plot, width of the notch relative to the body (default 0.5)

varwidth	if FALSE (default) make a standard box plot. If TRUE, boxes are drawn with widths proportional to the square-roots of the number of observations in the groups (possibly weighted, using the weight aesthetic).
na.rm	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders .

geom_crossbarh	<i>Horizontal intervals: lines, crossbars & errorbars.</i>
----------------	--

Description

Horizontal versions of [geom_linerange\(\)](#), [geom_pointrange\(\)](#), [geom_errorbar\(\)](#) and [geom_crossbar\(\)](#).

Usage

```
geom_crossbarh(mapping = NULL, data = NULL, stat = "identity",
  position = "identity", ..., fatten = 2.5, na.rm = FALSE,
  show.legend = NA, inherit.aes = TRUE)
```

```
geom_errorbarh(mapping = NULL, data = NULL, stat = "identity",
  position = "identity", ..., na.rm = FALSE, show.legend = NA,
  inherit.aes = TRUE)
```

```
geom_linerangeh(mapping = NULL, data = NULL, stat = "identity",
  position = "identity", ..., na.rm = FALSE, show.legend = NA,
  inherit.aes = TRUE)
```

```
geom_pointrangeh(mapping = NULL, data = NULL, stat = "identity",
  position = "identity", ..., fatten = 4, na.rm = FALSE,
  show.legend = NA, inherit.aes = TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by aes or aes_ . 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	The data to be displayed in this layer. There are three options: If NULL, the default, the data is inherited from the plot data as specified in the call to ggplot .

	A <code>data.frame</code> , or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify for which variables will be created.
	A function will be called with a single argument, the plot data. The return value must be a <code>data.frame</code> , and will be used as the layer data.
<code>stat</code>	The statistical transformation to use on the data for this layer, as a string.
<code>position</code>	Position adjustment, either as a string, or the result of a call to a position adjustment function.
<code>...</code>	other arguments passed on to layer . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired geom/stat.
<code>fatten</code>	A multiplicative factor used to increase the size of the middle bar in <code>geom_crossbar()</code> and the middle point in <code>geom_pointrange()</code> .
<code>na.rm</code>	If <code>FALSE</code> , the default, missing values are removed with a warning. If <code>TRUE</code> , missing values are silently removed.
<code>show.legend</code>	logical. Should this layer be included in the legends? <code>NA</code> , the default, includes if any aesthetics are mapped. <code>FALSE</code> never includes, and <code>TRUE</code> always includes.
<code>inherit.aes</code>	If <code>FALSE</code> , overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders .

<code>geom_histogramh</code>	<i>Horizontal histograms and frequency polygons.</i>
------------------------------	--

Description

Horizontal version of [geom_histogram\(\)](#).

Usage

```
geom_histogramh(mapping = NULL, data = NULL, stat = "binh",
  position = "stackv", ..., binwidth = NULL, bins = NULL, na.rm = FALSE,
  show.legend = NA, inherit.aes = TRUE)
```

Arguments

<code>mapping</code>	Set of aesthetic mappings created by aes or aes_ . 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.
<code>data</code>	The data to be displayed in this layer. There are three options: If <code>NULL</code> , the default, the data is inherited from the plot data as specified in the call to ggplot . A <code>data.frame</code> , or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify for which variables will be created.

A function will be called with a single argument, the plot data. The return value must be a `data.frame`, and will be used as the layer data.

<code>stat</code>	The statistical transformation to use on the data for this layer, as a string.
<code>position</code>	Position adjustment, either as a string, or the result of a call to a position adjustment function.
<code>...</code>	other arguments passed on to <code>layer</code> . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired geom/stat.
<code>binwidth</code>	The width of the bins. The default is to use bins bins that cover the range of the data. You should always override this value, exploring multiple widths to find the best to illustrate the stories in your data. The bin width of a date variable is the number of days in each time; the bin width of a time variable is the number of seconds.
<code>bins</code>	Number of bins. Overridden by <code>binwidth</code> . Defaults to 30
<code>na.rm</code>	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
<code>show.legend</code>	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes.
<code>inherit.aes</code>	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. <code>borders</code> .

<code>geom_violinh</code>	<i>Horizontal violin plot.</i>
---------------------------	--------------------------------

Description

Horizontal version of `geom_violin()`.

Usage

```
geom_violinh(mapping = NULL, data = NULL, stat = "xdensity",
  position = "dodgev", ..., draw_quantiles = NULL, trim = TRUE,
  scale = "area", na.rm = FALSE, show.legend = NA, inherit.aes = TRUE)
```

Arguments

<code>mapping</code>	Set of aesthetic mappings created by <code>aes</code> or <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.
<code>data</code>	The data to be displayed in this layer. There are three options: If NULL, the default, the data is inherited from the plot data as specified in the call to <code>ggplot</code> .

	A data.frame, or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify for which variables will be created.
	A function will be called with a single argument, the plot data. The return value must be a data.frame., and will be used as the layer data.
stat	The statistical transformation to use on the data for this layer, as a string.
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
...	other arguments passed on to layer . These are often aesthetics, used to set an aesthetic to a fixed value, like color = "red" or size = 3. They may also be parameters to the paired geom/stat.
draw_quantiles	If not(NULL) (default), draw horizontal lines at the given quantiles of the density estimate.
trim	If TRUE (default), trim the tails of the violins to the range of the data. If FALSE, don't trim the tails.
scale	if "area" (default), all violins have the same area (before trimming the tails). If "count", areas are scaled proportionally to the number of observations. If "width", all violins have the same maximum width.
na.rm	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders .

position_dodgev	<i>Vertical Positions</i>
-----------------	---------------------------

Description

Vertical versions of [position_dodge\(\)](#), [position_jitterdodge\(\)](#), [position_fill\(\)](#), [position_stack\(\)](#),

Usage

```
position_dodgev(height = NULL)
```

```
position_jitterdodgev(jitter.height = NULL, jitter.width = 0,
  dodge.height = 0.75)
```

```
position_stackv()
```

```
position_fillv()
```


Arguments

height	Dodging height, when different to the height of the individual elements. This is useful when you want to align narrow geoms with taller geoms.
jitter.height	degree of jitter in y direction. Defaults to 0.
jitter.width	degree of jitter in x direction. Defaults to 40% of the resolution of the data.
dodge.height	the amount to dodge in the y direction. Defaults to 0.75, the default position_dodgev() height.

stat_binh	<i>Horizontal binning.</i>
-----------	----------------------------

Description

Horizontal version of [stat_bin\(\)](#).

Usage

```
stat_binh(mapping = NULL, data = NULL, geom = "barh",
  position = "stackv", ..., binwidth = NULL, bins = NULL, center = NULL,
  boundary = NULL, closed = c("right", "left"), pad = FALSE,
  na.rm = FALSE, show.legend = NA, inherit.aes = TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by aes or aes_ . 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	The data to be displayed in this layer. There are three options: If <code>NULL</code> , the default, the data is inherited from the plot data as specified in the call to ggplot . A <code>data.frame</code> , or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify for which variables will be created. A function will be called with a single argument, the plot data. The return value must be a <code>data.frame</code> , and will be used as the layer data.
geom	Use to override the default connection between <code>geom_histogram/geom_freqpoly</code> and <code>stat_bin</code> .
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
...	other arguments passed on to layer . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired geom/stat.

binwidth	The width of the bins. The default is to use bins bins that cover the range of the data. You should always override this value, exploring multiple widths to find the best to illustrate the stories in your data. The bin width of a date variable is the number of days in each time; the bin width of a time variable is the number of seconds.
bins	Number of bins. Overridden by binwidth. Defaults to 30
center	The center of one of the bins. Note that if center is above or below the range of the data, things will be shifted by an appropriate number of widths. To center on integers, for example, use width = 1 and center = 0, even if 0 is outside the range of the data. At most one of center and boundary may be specified.
boundary	A boundary between two bins. As with center, things are shifted when boundary is outside the range of the data. For example, to center on integers, use width = 1 and boundary = 0.5, even if 0.5 is outside the range of the data. At most one of center and boundary may be specified.
closed	One of "right" or "left" indicating whether right or left edges of bins are included in the bin.
pad	If TRUE, adds empty bins at either end of x. This ensures frequency polygons touch 0. Defaults to FALSE.
na.rm	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders .

stat_boxploth	<i>Horizontal boxplot computation.</i>
---------------	--

Description

Horizontal version of [stat_boxplot\(\)](#).

Usage

```
stat_boxploth(mapping = NULL, data = NULL, geom = "boxploth",
  position = "dodgev", ..., coef = 1.5, na.rm = FALSE, show.legend = NA,
  inherit.aes = TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by aes or aes_ . 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	<p>The data to be displayed in this layer. There are three options:</p> <p>If NULL, the default, the data is inherited from the plot data as specified in the call to <code>ggplot</code>.</p> <p>A <code>data.frame</code>, or other object, will override the plot data. All objects will be fortified to produce a data frame. See <code>fortify</code> for which variables will be created.</p> <p>A function will be called with a single argument, the plot data. The return value must be a <code>data.frame</code>, and will be used as the layer data.</p>
geom	Use to override the default connection between <code>geom_boxplot</code> and <code>stat_boxplot</code> .
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
...	other arguments passed on to <code>layer</code> . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired geom/stat.
coef	length of the whiskers as multiple of IQR. Defaults to 1.5
na.rm	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. <code>borders</code> .

stat_counth	<i>Horizontal counting.</i>
-------------	-----------------------------

Description

Horizontal version of `stat_count()`.

Usage

```
stat_counth(mapping = NULL, data = NULL, geom = "barh",
            position = "stackv", ..., width = NULL, na.rm = FALSE,
            show.legend = NA, inherit.aes = TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by <code>aes</code> or <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	<p>The data to be displayed in this layer. There are three options:</p> <p>If NULL, the default, the data is inherited from the plot data as specified in the call to <code>ggplot</code>.</p> <p>A <code>data.frame</code>, or other object, will override the plot data. All objects will be fortified to produce a data frame. See <code>fortify</code> for which variables will be created.</p> <p>A function will be called with a single argument, the plot data. The return value must be a <code>data.frame</code>, and will be used as the layer data.</p>
geom	Override the default connection between <code>geom_bar</code> and <code>stat_count</code> .
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
...	other arguments passed on to <code>layer</code> . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired <code>geom/stat</code> .
width	Bar width. By default, set to 90% of the resolution of the data.
na.rm	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. <code>borders</code> .

stat_summaryh	<i>Horizontal summary.</i>
---------------	----------------------------

Description

Horizontal version of `stat_summary()`.

Usage

```
stat_summaryh(mapping = NULL, data = NULL, geom = "pointrangeh",
  position = "identity", ..., fun.data = NULL, fun.x = NULL,
  fun.xmax = NULL, fun.xmin = NULL, fun.args = list(), na.rm = FALSE,
  show.legend = NA, inherit.aes = TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by <code>aes</code> or <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	<p>The data to be displayed in this layer. There are three options:</p> <p>If NULL, the default, the data is inherited from the plot data as specified in the call to <code>ggplot</code>.</p> <p>A <code>data.frame</code>, or other object, will override the plot data. All objects will be fortified to produce a data frame. See <code>fortify</code> for which variables will be created.</p> <p>A function will be called with a single argument, the plot data. The return value must be a <code>data.frame</code>, and will be used as the layer data.</p>
geom	Use to override the default connection between <code>geom_histogram/geom_freqpoly</code> and <code>stat_bin</code> .
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
...	other arguments passed on to <code>layer</code> . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired geom/stat.
fun.data	A function that is given the complete data and should return a data frame with variables <code>xmin</code> , <code>x</code> , and <code>xmax</code> .
fun.xmin, fun.x, fun.xmax	Alternatively, supply three individual functions that are each passed a vector of <code>x</code> 's and should return a single number.
fun.args	Optional additional arguments passed on to the functions.
na.rm	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. <code>borders</code> .

stat_xdensity *Density computation on x axis.*

Description

Horizontal version of `stat_ydensity()`.

Usage

```
stat_xdensity(mapping = NULL, data = NULL, geom = "violin",
  position = "dodgev", ..., bw = "nrd0", adjust = 1,
  kernel = "gaussian", trim = TRUE, scale = "area", na.rm = FALSE,
  show.legend = NA, inherit.aes = TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by aes or aes_ . 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	The data to be displayed in this layer. There are three options: If <code>NULL</code> , the default, the data is inherited from the plot data as specified in the call to ggplot . A <code>data.frame</code> , or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify for which variables will be created. A function will be called with a single argument, the plot data. The return value must be a <code>data.frame</code> , and will be used as the layer data.
geom	Use to override the default connection between <code>geom_violin</code> and <code>stat_ydensity</code> .
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
...	other arguments passed on to layer . These are often aesthetics, used to set an aesthetic to a fixed value, like <code>color = "red"</code> or <code>size = 3</code> . They may also be parameters to the paired geom/stat.
bw	The smoothing bandwidth to be used. If numeric, the standard deviation of the smoothing kernel. If character, a rule to choose the bandwidth, as listed in bw.nrd .
adjust	A multiplicate bandwidth adjustment. This makes it possible to adjust the bandwidth while still using the a bandwidth estimator. For example, <code>adjust = 1/2</code> means use half of the default bandwidth.
kernel	Kernel. See list of available kernels in density .
trim	If <code>TRUE</code> (default), trim the tails of the violins to the range of the data. If <code>FALSE</code> , don't trim the tails.
scale	if "area" (default), all violins have the same area (before trimming the tails). If "count", areas are scaled proportionally to the number of observations. If "width", all violins have the same maximum width.
na.rm	If <code>FALSE</code> , the default, missing values are removed with a warning. If <code>TRUE</code> , missing values are silently removed.
show.legend	logical. Should this layer be included in the legends? <code>NA</code> , the default, includes if any aesthetics are mapped. <code>FALSE</code> never includes, and <code>TRUE</code> always includes.
inherit.aes	If <code>FALSE</code> , overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders .

Index

`aes`, [3–7](#), [9–12](#), [14](#)
`aes_`, [3–7](#), [9–12](#), [14](#)

`borders`, [3](#), [5–8](#), [10–14](#)
`bw.nrd`, [14](#)

`density`, [14](#)
`draw_key`, [2](#)
`draw_key_boxploth` (`draw_key`), [2](#)
`draw_key_crossbarh` (`draw_key`), [2](#)
`draw_key_hpath` (`draw_key`), [2](#)
`draw_key_pointrangeh` (`draw_key`), [2](#)

`fortify`, [3](#), [4](#), [6](#), [8](#), [9](#), [11–14](#)

`geom_bar`, [3](#)
`geom_barh`, [3](#)
`geom_boxplot`, [4](#)
`geom_boxploth`, [4](#)
`geom_crossbar`, [5](#)
`geom_crossbarh`, [5](#)
`geom_errorbar`, [5](#)
`geom_errorbarh` (`geom_crossbarh`), [5](#)
`geom_histogram`, [3](#), [6](#)
`geom_histogramh`, [6](#)
`geom_linerange`, [5](#)
`geom_linerangeh` (`geom_crossbarh`), [5](#)
`geom_pointrange`, [5](#)
`geom_pointrangeh` (`geom_crossbarh`), [5](#)
`geom_violin`, [7](#)
`geom_violinh`, [7](#)
`ggplot`, [3–7](#), [9](#), [11–14](#)

`layer`, [3](#), [4](#), [6–9](#), [11–14](#)

`position-vertical` (`position_dodgev`), [8](#)
`position_dodge`, [8](#)
`position_dodgev`, [8](#)
`position_fill`, [8](#)
`position_fillv` (`position_dodgev`), [8](#)
`position_jitterdodge`, [8](#)
`position_jitterdodgev`
 (`position_dodgev`), [8](#)
`position_stack`, [8](#)
`position_stackv` (`position_dodgev`), [8](#)

`stat_bin`, [9](#)
`stat_binh`, [9](#)
`stat_boxplot`, [10](#)
`stat_boxploth`, [10](#)
`stat_count`, [11](#)
`stat_counth`, [11](#)
`stat_summary`, [12](#)
`stat_summaryh`, [12](#)
`stat_xdensity`, [13](#)
`stat_ydensity`, [13](#)