Package 'ggrastr'

May 24, 2020

Type Package

Title Raster Layers for 'ggplot2'

Version 0.1.8

Description The package ggrastr provides a set of geoms to rasterize only specific layers of the plot while simultaneously keeping all labels and text in vector format. The package allows to keep your plots within the reasonable size limit without loosing vector properties of the scale-sensitive information.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports ggplot2 (>= 2.1.0), Cairo (>= 1.5.9), ggbeeswarm

Depends R (>= 3.2.2)

RoxygenNote 7.1.0

Suggests rmarkdown, knitr

VignetteBuilder knitr

URL https://github.com/VPetukhov/ggrastr

BugReports https://github.com/VPetukhov/ggrastr/issues

NeedsCompilation no

Author Viktor Petukhov [aut], Evan Biederstedt [cre, aut]

Maintainer Evan Biederstedt <evan.biederstedt@gmail.com>

R topics documented:

geom_beeswarm_rast 2
geom_boxplot_jitter 3
geom_point_rast 5
geom_quasirandom_rast 7
geom_tile_rast 9
theme_pdf 11

Index **12**

geom_beeswarm_rast *This geom is similar to* geom_beeswarm*, but creates a raster layer*

Description

This geom is similar to geom_beeswarm, but creates a raster layer

Usage

```
geom_beeswarm_rast(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "quasirandom",
  priority = c("ascending", "descending", "density", "random", "none"),
  cex = 1,
  groupOnX = NULL,
  dodge.width = 0,
  ...,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  raster.width = NULL,
  raster.height = NULL,
  raster.dpi = 300
)
```

Arguments

mapping	Set of aesthetic mappings created by aes() or aes_(). If specified and inherit.aes = TRUE (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 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. A function can be created from a formula (e.g. ~ head(.x,10)).
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.
priority	Method used to perform point layout (see ggbeeswarm::position_beeswarm)
cex	Scaling for adjusting point spacing (see ggbeeswarm::position_beeswarm)
groupOnX	Should jitter be added to the x axis if TRUE or y axis if FALSE (the default NULL causes the function to guess which axis is the categorical one based on the number of unique entries in each) Refer to see ggbeeswarm::position_beeswarm

dodge.width	Amount by which points from different aesthetic groups will be dodged. This requires that one of the aesthetics is a factor. (see ggbeeswarm::position_beeswarm)
...	Other arguments passed on to layer(). These are often aesthetics, used to set an aesthetic to a fixed value, like colour = "red" or size = 3. They may also be parameters to the paired geom/stat.
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. It can also be a named logical vector to finely select the aesthetics to display.
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().
raster.width	Width of the result image (in inches). Default: deterined by the current device parameters.
raster.height	Height of the result image (in inches). Default: deterined by the current device parameters.
raster.dpi	Resolution of the result image.

Examples

```
library(ggplot2)
library(ggrastr)

ggplot(mtcars) + geom_beeswarm_rast(aes(x = factor(cyl), y = mpg), raster.dpi = 600, cex = 1.5)
```

---

geom_boxplot_jitter *This geom is similar to* geom_boxplot*, but allows to jitter outlier points and to raster points layer.*

Description

This geom is similar to geom_boxplot, but allows to jitter outlier points and to raster points layer.

Usage

```
geom_boxplot_jitter(
  mapping = NULL,
  data = NULL,
  stat = "boxplot",
  position = "dodge",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  ...,
  outlier.jitter.width = NULL,
  outlier.jitter.height = 0,
  raster = FALSE,
```

```
  raster.dpi = 300,
  raster.width = NULL,
  raster.height = NULL
)
```

Arguments

mapping	Set of aesthetic mappings created by aes() or aes_(). If specified and inherit.aes = TRUE (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 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. A function can be created from a formula (e.g. ~ head(.x,10)).
stat	Use to override the default connection between geom_boxplot and stat_boxplot.
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
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. It can also be a named logical vector to finely select the aesthetics to display.
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().
...	Other arguments passed on to layer(). These are often aesthetics, used to set an aesthetic to a fixed value, like colour = "red" or size = 3. They may also be parameters to the paired geom/stat.
outlier.jitter.width	Amount of horizontal jitter. The jitter is added in both positive and negative directions, so the total spread is twice the value specified here. Default: boxplot width.
outlier.jitter.height	Amount of horizontal jitter. The jitter is added in both positive and negative directions, so the total spread is twice the value specified here. Default: 0.
raster	Should outlier points be rastered?.
raster.dpi	Resolution of the rastered image. Ignored if raster == FALSE.
raster.width	Width of the result image (in inches). Default: deterined by the current device parameters. Ignored if raster == FALSE.
raster.height	Height of the result image (in inches). Default: deterined by the current device parameters. Ignored if raster == FALSE.

Aesthetics

geom_boxplot() understands the following aesthetics (required aesthetics are in bold):

- x ***or*** y
- lower ***or*** xlower
- upper ***or*** xupper
- middle ***or*** xmiddle
- ymin ***or*** xmin
- ymax ***or*** xmax
- alpha
- colour
- fill
- group
- linetype
- shape
- size
- weight

Learn more about setting these aesthetics in vignette("ggplot2-specs").

Examples

```
library(ggplot2)
library(ggrastr)

yvalues = rt(1000, df=3)
xvalues = as.factor(1:1000 %% 2)
ggplot() + geom_boxplot_jitter(aes(y=yvalues, x=xvalues), outlier.jitter.width = 0.1, raster = TRUE)
```

---

geom_point_rast *This geom is similar to geom_point, but creates a raster layer*

Description

This geom is similar to geom_point, but creates a raster layer

Usage

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

```
  raster.width = NULL,
  raster.height = NULL,
  raster.dpi = 300
)
```

Arguments

mapping	Set of aesthetic mappings created by aes() or aes_(). If specified and inherit.aes = TRUE (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 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. A function can be created from a formula (e.g. ~ head(.x,10)).
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 colour = "red" or size = 3. They may also be parameters to the paired geom/stat.
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. It can also be a named logical vector to finely select the aesthetics to display.
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().
raster.width	Width of the result image (in inches). Default: deterined by the current device parameters.
raster.height	Height of the result image (in inches). Default: deterined by the current device parameters.
raster.dpi	Resolution of the result image.

Aesthetics

geom_point() understands the following aesthetics (required aesthetics are in bold):

- x
- y
- alpha
- colour
- fill

- group
- shape
- size
- stroke

Learn more about setting these aesthetics in vignette("ggplot2-specs").

Examples

```
library(ggplot2)
library(ggrastr)

ggplot() + geom_point_rast(aes(x=rnorm(1000), y=rnorm(1000)), raster.dpi=600)
```

---

geom_quasirandom_rast *This geom is similar to* geom_quasirandom*, but creates a raster layer*

Description

This geom is similar to geom_quasirandom, but creates a raster layer

Usage

```
geom_quasirandom_rast(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "quasirandom",
  width = NULL,
  varwidth = FALSE,
  bandwidth = 0.5,
  nbins = NULL,
  method = "quasirandom",
  groupOnX = NULL,
  dodge.width = 0,
  ...,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  raster.width = NULL,
  raster.height = NULL,
  raster.dpi = 300
)
```

Arguments

mapping	Set of aesthetic mappings created by aes() or aes_(). If specified and inherit.aes = TRUE (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 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. A function can be created from a formula (e.g. ~ head(.x,10)).
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.
width	the maximum amount of spread (default: 0.4)
varwidth	vary the width by the relative size of each group
bandwidth	the bandwidth adjustment to use when calculating density Smaller numbers (< 1) produce a tighter "fit". (default: 0.5)
nbins	the number of bins used when calculating density (has little effect with quasirandom/random distribution)
method	the method used for distributing points (quasirandom, pseudorandom, smiley or frowney)
groupOnX	if TRUE then jitter is added to the x axis and if FALSE jitter is added to the y axis. Prior to v0.6.0, the default NULL causes the function to guess which axis is the categorical one based on the number of unique entries in each. This could result in unexpected results when the x variable has few unique values and so in v0.6.0 the default was changed to always jitter on the x axis unless groupOnX=FALSE. Also consider coord_flip.
dodge.width	Amount by which points from different aesthetic groups will be dodged. This requires that one of the aesthetics is a factor.
...	Other arguments passed on to layer(). These are often aesthetics, used to set an aesthetic to a fixed value, like colour = "red" or size = 3. They may also be parameters to the paired geom/stat.
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. It can also be a named logical vector to finely select the aesthetics to display.
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().
raster.width	Width of the result image (in inches). Default: deterined by the current device parameters.
raster.height	Height of the result image (in inches). Default: deterined by the current device parameters.
raster.dpi	Resolution of the result image.

Aesthetics

geom_point() understands the following aesthetics (required aesthetics are in bold):

- x
- y
- alpha
- colour
- fill
- group
- shape
- size
- stroke

Learn more about setting these aesthetics in vignette("ggplot2-specs").

Examples

```
library(ggplot2)
library(ggrastr)

ggplot(mtcars) + geom_quasirandom_rast(aes(x = factor(cyl), y = mpg), raster.dpi = 600)
```

---

geom_tile_rast *This geom is similar to geom_tile, but creates a raster layer*

Description

This geom is similar to geom_tile, but creates a raster layer

Usage

```
geom_tile_rast(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  ...,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  raster.width = NULL,
  raster.height = NULL,
  raster.dpi = 300
)
```

Arguments

mapping	Set of aesthetic mappings created by aes() or aes_(). If specified and inherit.aes = TRUE (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 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. A function can be created from a formula (e.g. ~ head(.x,10)).
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 colour = "red" or size = 3. They may also be parameters to the paired geom/stat.
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. It can also be a named logical vector to finely select the aesthetics to display.
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().
raster.width	Width of the result image (in inches). Default: deterined by the current device parameters.
raster.height	Height of the result image (in inches). Default: deterined by the current device parameters.
raster.dpi	Resolution of the result image.

Aesthetics

geom_tile() understands the following aesthetics (required aesthetics are in bold):

- x
- y
- alpha
- colour
- fill
- group
- height
- linetype
- size
- width

Learn more about setting these aesthetics in vignette("ggplot2-specs").

Examples

```
library(ggplot2)
library(ggrastr)

coords <- expand.grid(1:100, 1:100)
coords$Value <- 1 / apply(as.matrix(coords), 1, function(x) sum((x - c(50, 50))^2)^0.01)
ggplot(coords) + geom_tile_rast(aes(x=Var1, y=Var2, fill=Value))
```

---

theme_pdf *Pretty theme*

Description

Pretty theme

Usage

```
theme_pdf(show.ticks = TRUE, legend.pos = NULL)
```

Arguments

show.ticks	Show x- and y- ticks.
legend.pos	Vector with x and y position of the legend.

Examples

```
library(ggplot2)
library(ggrastr)

data = rnorm(100)
colors = (1:100/100)
ggplot() + geom_point(aes(x=data, y=data, color=colors)) + theme_pdf(FALSE, legend.pos=c(1, 1))
```

Index

aes(), *2*, *4*, *6, 7*, *10*
aes_(), *2*, *4*, *6, 7*, *10*

borders(), *3, 4*, *6*, *8*, *10*

coord_flip, *8*

fortify(), *2*, *4*, *6*, *8*, *10*

geom_beeswarm, *2*
geom_beeswarm_rast, 2
geom_boxplot, *3*
geom_boxplot_jitter, 3
geom_point, *5*
geom_point_rast, 5
geom_quasirandom, *7*
geom_quasirandom_rast, 7
geom_tile, *9*
geom_tile_rast, 9
ggplot(), *2*, *4*, *6*, *8*, *10*

layer(), *3, 4*, *6*, *8*, *10*

theme_pdf, 11