Package ‘ggquiver’

October 13, 2022

Version 0.3.2
Title Quiver Plots for ‘ggplot2’
Description An extension of ‘ggplot2’ to provide quiver plots to visualise vector fields.
This functionality is implemented using a geom to produce a new graphical layer, which
allows aesthetic options. This layer can be overlaid on a map to improve visualisation
of mapped data.
Depends R (>= 3.2.0)
Imports ggplot2
Suggests dplyr, ggmap, maps, pkgdown, testthat
URL https://github.com/mitchelloharawild/ggquiver,
https://pkg.mitchelloharawild.com/ggquiver/
BugReports https://github.com/mitchelloharawild/ggquiver/issues
License GPL-3
Encoding UTF-8
ByteCompile true
RoxygenNote 7.1.2
NeedsCompilation no
Author Mitchell O’Hara-Wild [aut, cre]
Maintainer Mitchell O’Hara-Wild <mail@mitchelloharawild.com>
Repository CRAN
Date/Publication 2021-12-06 08:00:02 UTC

R topics documented:

  geom_quiver .......................................................... 2

Index 5
**geom_quiver**

Quiver plots for ggplot2

**Description**

Displays the direction and length of vectors on a graph.

**Usage**

```r
gem_quiver(
  mapping = NULL,
  data = NULL,
  stat = "quiver",
  position = "identity",
  center = FALSE,
  rescale = FALSE,
  vecsize = NULL,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  ...
)
```

**GeomQuiver**

```r
stat_quiver(
  mapping = NULL,
  data = NULL,
  geom = "quiver",
  position = "identity",
  center = FALSE,
  rescale = FALSE,
  vecsize = NULL,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  ...
)
```

**StatQuiver**

**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 adjust-
ment function.

center  If FALSE (the default), the vector lines will start at the specified x and y. If TRUE,
the arrows will be centered about x and y.

rescale  If FALSE (the default), the vectors will not be rescaled. If TRUE, the vectors given
by (u, v) will be rescaled using the scale function.

vecsize  By default (NULL), vectors sizing is automatically determined. If a grid can be
identified, they will be scaled to the grid, if not, the vectors will not be scaled.
By specifying a numeric input here, the length of all arrows can be adjusted.
Setting vecsize to zero will prevent scaling the arrows.

na.rm  If FALSE (the default), removes missing values with a warning. If TRUE silently
removes missing values.

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.

geom  The geometric object to use display the data

Format
An object of class GeomQuiver (inherits fromGeomSegment, Geom, ggproto, gg) of length 2.
An object of class StatQuiver (inherits from Stat, ggproto, gg) of length 3.

Computed variables

x  centered x start position for velocity arrow
y  centered y start position for velocity arrow
xend  centered x end position for velocity arrow
yend  centered y end position for velocity arrow
Examples

library(ggplot2)
# Quiver plots of mathematical functions
field <- expand.grid(x=seq(0,pi,pi/12), y=seq(0,pi,pi/12))
ggplot(field, aes(x=x,y=y,u=cos(x),v=sin(y))) +
  geom_quiver()

# Removing automatic scaling
ggplot(seals, aes(x=long, y=lat, u=delta_long, v=delta_lat)) +
  geom_quiver(vecsize=NULL) +
  borders("state")

# Centering arrows is useful for plotting on maps.
library(dplyr)
library(ggmap)
wind_data <- wind %>% filter(between(lon, -96, -93) & between(lat, 28.7, 30))
qmplot(lon, lat, data=wind_data, extent="panel", geom = "blank", zoom=8, maptype = "toner-lite") +
  geom_quiver(aes(u=delta_lon, v=delta_lat, colour = spd), center=TRUE)
Index

* datasets
  geom_quiver, 2

aes(), 2
aes_(), 2

borders(), 3

fortify(), 3

geom_quiver, 2
GeomQuiver (geom_quiver), 2

ggplot(), 3

layer(), 3

stat_quiver (geom_quiver), 2
StatQuiver (geom_quiver), 2