Package 'ggdaynight'

<div align="center">May 23, 2024</div>

Type Package

Title Add Day/Night Patterns to 'ggplot2' Plots

Version 0.1.3

Description It provides a custom 'ggplot2' geom to add day/night patterns to plots.
It visually distinguishes daytime and nighttime periods. It is useful for
visualizing data that spans multiple days and for highlighting diurnal patterns.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

URL https://github.com/GabrielSlPires/ggdaynight

BugReports https://github.com/GabrielSlPires/ggdaynight/issues

Imports ggplot2, grid

Depends R (>= 2.10)

RoxygenNote 7.3.1

Suggests testthat (>= 3.0.0), vdiffr

Config/testthat/edition 3

NeedsCompilation no

Author Gabriel S. Pires [aut, cre, cph]
(<https://orcid.org/0000-0002-5354-1957>)

Maintainer Gabriel S. Pires <gabriel.slpires@gmail.com>

Repository CRAN

Date/Publication 2024-05-23 07:10:09 UTC

R topics documented:

---

daynight_temperature   *Sensor Temperature Data*

Description

A dataset containing temperature readings from multiple sensors over a specified period.

Usage

```
daynight_temperature
```

Format

'daynight_temperature' A data frame with 1833 rows and 3 columns:

sensor Sensor identifier (e.g., A, B)
datetime Timestamp of the reading (POSIXct format)
temperature Temperature reading in degrees Celsius

Source

Sensor readings collected on 2024-04-23 in Campinas-SP, Brazil.

---

geom_daynight   *Add Day/Night Pattern to ggplot*

Description

Adds a day/night pattern to a ggplot object. Daytime is represented by rectangles filled with the specified 'day_fill' color and nighttime by rectangles filled with the specified 'night_fill' color. The pattern is created along the x-axis, which must be a datetime variable.

Usage

```
geom_daynight(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  day_fill = "white",
  night_fill = "grey30",
  sunrise = 6,
  sunset = 18,
  ...
)
```

Arguments

mapping	Set of aesthetic mappings created by [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. When using a geom_*() function to construct a layer, the stat argument can be used the override the default coupling between geoms and stats. The stat argument accepts the following:
	• A Stat ggproto subclass, for example StatCount.
	• A string naming the stat. To give the stat as a string, strip the function name of the stat_ prefix. For example, to use stat_count(), give the stat as "count".
	• For more information and other ways to specify the stat, see the [layer stat]() documentation.
position	A position adjustment to use on the data for this layer. This can be used in various ways, including to prevent overplotting and improving the display. The position argument accepts the following:
	• The result of calling a position function, such as position_jitter(). This method allows for passing extra arguments to the position.
	• A string naming the position adjustment. To give the position as a string, strip the function name of the position_ prefix. For example, to use position_jitter(), give the position as "jitter".
	• For more information and other ways to specify the position, see the [layer position]() documentation.
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()]().
day_fill	The fill color for daytime rectangles. Defaults to "white".
night_fill	The fill color for nighttime rectangles. Defaults to "grey30".
sunrise	The hour at which daytime starts. Defaults to 6 (6 AM).

sunset	The hour at which nighttime starts. Defaults to 18 (6 PM).
...	Additional arguments passed to 'geom_rect'.

Value

A ggplot2 layer representing the day/night pattern.

Examples

```
# Basic usage with default parameters
library(ggplot2)
ggplot(daynight_temperature, aes(datetime, temperature)) +
  geom_daynight() +
  geom_point()

# Basic usage with faceting by sensor
ggplot(daynight_temperature, aes(datetime, temperature)) +
  geom_daynight() +
  geom_point() +
  facet_wrap(vars(sensor))

# Usage with lines and color by sensor
ggplot(daynight_temperature, aes(datetime, temperature, color = sensor)) +
  geom_daynight() +
  geom_line()

# Custom day and night fill colors, custom sunrise and sunset times, and adjusted alpha
ggplot(daynight_temperature, aes(datetime, temperature, color = sensor)) +
  geom_daynight(
    day_fill = "yellow", night_fill = "blue",
    sunrise = 5, sunset = 20, alpha = 0.5
  ) +
  geom_line(linewidth = 1)
```

Index