

Package ‘clockplot’

May 8, 2026

Type Package

Title Plot Event Times on a 24-Hour Clock

Version 0.8.3

Description Provides a novel visualization technique for plotting timestamped events on a 24-hour circular clock face. This is particularly useful for analyzing daily patterns, event clustering, and gaps in temporal data. The package also generalizes this approach to create cyclic charts for other periods, including weekly and monthly cycles, enabling effective event planning and pattern analysis across multiple time frames.

License MIT + file LICENSE

Imports dplyr, tidyr, tibble, ggplot2, hms, rlang

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Depends R (>= 2.10)

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

URL <https://github.com/mahmudstat/clockplot/>

BugReports <https://github.com/mahmudstat/clockplot/issues>

Language en-US

Config/testthat/edition 3

NeedsCompilation no

Author Abdullah Al Mahmud [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0003-2814-8798>>)

Maintainer Abdullah Al Mahmud <almahmud.sbi@gmail.com>

Repository CRAN

Date/Publication 2025-09-13 22:50:02 UTC

Contents

bdquake	2
bdtemp	3
brincity	3
chatdf	4
clock_chart	5
clock_chart_col	6
clock_chart_qlt	7
clock_chart_qnt	8
cyclic_chart	9
day_chart	10
gitcommit	11
plan_day	11
plan_week	12
smsclock	13
week_chart	13
year_chart	14
Index	16

bdquake

Data of Earthquakes in and around Bangladesh

Description

A dataset containing earthquakes magnitude, depth, and location information.

Usage

bdquake

Format

A data frame with 13 rows and 5 variables:

latitude Latitude

longitude Longitude

depth Depth

mag Magnitude

hms Time

Source

[USGS](#)

bdtemp	<i>Data of Average Monthly Temperature of Bangladesh Divisional Cities</i>
--------	--

Description

A dataset containing temperature of Dhaka, Sylhet, and Chittagong Cities.

Usage

bdtemp

Format

A data frame with 36 rows and 3 variables:

Temperature Average monthly temperature

Month Month of the year

City Name of the city

Source

[Weather Base](#)

brintcity	<i>Data of Bangladesh Railway Express Train Schedule</i>
-----------	--

Description

A dataset containing Bangladesh Railway Express Train schedule

Usage

brintcity

Format

A data frame with 84 rows and 10 variables:

SI Serial

Train_no Train No.

Name Train name

Type Type of train

Off_day The day of the week when the train does not run

Origin The station from where the train departs

Departure The time of departure

Destination The station where the train is heading

Arrival The time of arrival

Zone East/West

Source

Bangladesh Railway

chatdf

Data Containing Chat Times of Two Individuals

Description

A dataset containing timings of chat by two individuals Abid and Abir The set contains 25 pushes by each individual

Usage

chatdf

Format

A data frame with 25 rows and 3 variables:

name Name of the chat participant

time Time in HH:MM:SS format

turn Turn of the chat

Source

Randomly generated

Description

There are five types of clock charts, `clock_chart()` being the simplest one. It just shows the event times on a 24 hour clock. The lines are neither colored, nor length modified. `clock_chart_col()` is used to colorize and `clock_chart_len()` to change the length of the hands by a numeric vector. To do both simultaneously, use `clock_chart_qnt()`. To use a qualitative variable as the criterion, use `clock_chart_qlt()`.

Usage

```
clock_chart(data, time, Col = "black")
```

Arguments

<code>data</code>	A data frame
<code>time</code>	Time in 24 hours. The allowed time formats for these family of charts are HH:MM:SS, HH:MM or even H:M (such as 12;30:09 or 9:3), although the SS part is ignored due to having negligible impact on the final plot).
<code>Col</code>	A single color name for the lines. The default is <code>black</code> .

Details

Change the title, subtitle or the caption of the plot with `ggplot2::labs()`.

Value

A `ggplot` object, which can be further modified with `ggplot2` functions and themes.

See Also

[clock_chart_col\(\)](#), [clock_chart_qnt\(\)](#), and [clock_chart_qlt\(\)](#).

Examples

```
p1 <- clock_chart(smsclock, time) # Using package built-in data
p1 + ggplot2::labs(title = "SMS Receiving Times")
# Add clock_chart(brintcity %>% filter(Origin == "Dhaka"), time = Departure)
```

clock_chart_col *Clock Chart, Hands Colored by a Numeric Variable*

Description

This function will plot time of events on a 24 hour clock to show which events took place at what times. The lines are colored by a criteria.

Usage

```
clock_chart_col(data, time, crit, high = "red", low = "green")
```

Arguments

data	A data frame
time	Time in 24 hours. The allowed time formats for these family of charts are HH:MM:SS, HH:MM or even H:M (such as 12;30:09 or 9:3).
crit	a numeric vector by which lines will be colored.
high	The color name for the high values. The default is red
low	The color name for the low values. The default is green. The color names can be vice versa or other colors, depending on the context.

Details

Change the title, subtitle or the caption of the plot with `ggplot2::labs()`. Change the legend title by adding `ggplot2::labs(size = "TITLE")` or `labs(color = "TITLE")`.

Add or modify legend by `theme(legend.position = "POSITION")`; the valid position names in `ggplot2` are `top`, `bottom`, `right`, and `left`, excluding more complex options.

Value

A `ggplot` object, which can be further modified with `ggplot2` functions and themes.

See Also

[clock_chart_qnt\(\)](#), and [clock_chart_qlt\(\)](#).

Examples

```
df <- data.frame(
  time = c("06:00:00", "08:00:00", "17:30:00"),
  value = c(3, 6, 9)
)
clock_chart_col(df, time, crit = value) +
  ggplot2::labs(size = "TITLE")
```

`clock_chart_qlt`*Clock Chart, Hands Colored by a Factor*

Description

This function will plot time of events on a 24 hour clock to show which events took place at what times. The hands are colored by a qualitative (factor) vector. Change the plot's title, subtitle, or caption using `ggplot2::labs()`.

For example: `ggplot2::labs(title = "My Plot", subtitle = "My Subtitle")` You can change the title of the legend by adding `ggplot2::labs(color = "Legend Title")`.

Usage

```
clock_chart_qlt(data, time, crit)
```

Arguments

<code>data</code>	A data frame
<code>time</code>	Time in 24 hours. The allowed time formats for these family of charts are HH:MM:SS, HH:MM or even H:M (such as 12;30:09 or 9:3).
<code>crit</code>	The qualitative vector by which hands will be colored.

Value

A ggplot object, which can be further modified with `ggplot2` functions and themes.

See Also

[clock_chart_col\(\)](#) for coloring and [clock_chart_qnt\(\)](#) for more options.

Examples

```
# A plot showing sms receiving times based on
# criteria (type/sender/invoked)
clock_chart_qlt(smsclock, time = time, crit = sender) +
  ggplot2::labs(color = "Sender", title = "SMS's Received throughout th Day")
```

clock_chart_qnt	<i>Clock Chart, Length and Color Modified by Numeric Variables</i>
-----------------	--

Description

This function will plot time of events on a 24 hour clock to show which events took place at what times. The length and color of the hands are modified according to a numeric vector.

Usage

```
clock_chart_qnt(data, time, len, Col = NULL, high = "red", low = "green")
```

Arguments

data	A data frame
time	Time in 24 hours. The allowed time formats for these family of charts are HH:MM:SS, HH:MM or even H:M (such as 12;30:09 or 9:3).
len	The numeric vector by which hands will be modified and colored.
Col	Optional. A numeric vector to change color. This option will be discontinued.
high	The color name for the high values. The default is red.
low	The color name for the low values. The default is green. The color names can be vice versa or other colors, depending on the context. To use a single color for all lines, use same value for high and low.

Details

Change the title, subtitle or the caption of the plot with `ggplot2::labs()`. Change the legend title by adding `ggplot2::labs(color = "TITLE 1", size = "TITLE2")`. Add or modify legend by `theme(legend.position = "POSITION")`; the valid position names in `ggplot2` are top, bottom, right, and left, excluding more complex options.

Value

A `ggplot` object, which can be further modified with `ggplot2` functions and themes.

See Also

[clock_chart_col\(\)](#) for coloring by a numeric variable, [clock_chart_qlt\(\)](#) for coloring by a qualitative variable, [clock_chart\(\)](#) for the simplest clock chart

Examples

```
p1 <- clock_chart_qnt(
  data = bdquake, time = hms, len = depth,
  Col = mag, high = "red", low = "blue"
)
p1 + ggplot2::labs(
  color = "Depth", size = "Magnitude",
  title = "Earthquakes in Bangladesh since 2023"
)
```

cyclic_chart

Plot Cyclic Data (General Format)

Description

This function plots values corresponding to random periods such as hours, days, months and so on.

Usage

```
cyclic_chart(df, Period, Value, crit, ColV)
```

Arguments

df	A data frame
Period	A list of periodical values such as hours of the day, days of the week, months of the year and so on.
Value	A numeric vector with the values corresponding to the Period
crit	A factor variable based on which the bars would be colored.
ColV	A character vector with the list of colors for the bars. You can use this online tool (https://r-charts.com/color-palette-generator/) to create a beautiful color palette.

Details

This can plot values corresponding to multiple categories, for example, temperature in different cities on the days of a week, or rainfall by month in a year.

Value

A ggplot object, which can be further modified with ggplot2 functions and themes.

See Also

[day_chart\(\)](#) for plotting values on a day by hours, [week_chart\(\)](#) for plotting values in a week by days, [year_chart\(\)](#) for plotting values in a year by months

Examples

```
# Using package built-in bdtemp data
Col <- c("#0040ff", "#00bfff", "#8000ff")
cyclic_chart(bdtemp,
  Period = Month, Value = Temperature,
  crit = City, ColV = Col
)
```

day_chart

Plot Values on a 24-Hour Day, on Specific Hours

Description

This function plots values corresponding to each hour on a rose plot.

Usage

```
day_chart(hvalue, high = "blue", low = "yellow", width = 0.8)
```

Arguments

hvalue	A numeric vector having values at each of 24 hours (starts from 6 am)
high	The color name for the high values. The default is red
low	The color name for the low values. The default is green.
width	Width of bars

Details

The color names can be vice versa or other colors, depending on the context.

Value

A ggplot object, which can be further modified with ggplot2 functions and themes.

See Also

[week_chart\(\)](#) for plotting values in a week by days [year_chart\(\)](#) for plotting values on in a year by months [cyclic_chart\(\)](#) for plotting values by arbitrary period

Examples

```
value <- sample(15:30, 24, replace = TRUE)
day_chart(hvalue = value, high = "blue", low = "yellow", width = 0.8)
```

gitcommit	<i>Data of git commits in this repository</i>
-----------	---

Description

A dataset containing commit information of all the commits in this repository (clockplot) up to 12 Sep, 2024. Some columns, including the commit message have been omitted.

Usage

```
gitcommit
```

Format

A data frame with 110 rows and 6 variables:

id Commit ID
weekday Day of the week
month Month
day Day of the month
time Time—this is our desired column
year Year

Source

[clockplot repository](#)

plan_day	<i>Visually Plan Activities on a Day with Events on Each of 24 Hours</i>
----------	--

Description

This function plots works corresponding to each hour on a rose plot.

Usage

```
plan_day(dwork, width = 1, brdcol = "grey")
```

Arguments

dwork A character vector of length 24, with activity names for each hour starting from 6 AM and ending at 5 AM the next day.
width Width of bars.
brdcol Color of bar border. Use NA for no border.

Value

A ggplot object, which can be further modified with ggplot2 functions and themes.

Examples

```
set.seed(123)
work <- sample(c("Study", "Adda", "Entertainment", "Games", "Exercise", "Meal"),
  size = 24, replace = TRUE
)
plan_day(dwork = work, brdcol = NA)
```

plan_week

Visually Plan Activities on a Week with Events on Each Day

Description

This function plots works corresponding to each day of the week.

Usage

```
plan_week(wtask)
```

Arguments

wtask A factor variable having values on each day of the week.

Value

A ggplot object, which can be further modified with ggplot2 functions and themes.

Examples

```
set.seed(10)
wtask <- c(
  "Desk Work", "Field Work", "Visit", "Monitoring",
  "Rest", "Reporting", "Meeting"
)
plan_week(wtask)
```

`smsclock`*Data of Times of Receiving of SMS*

Description

A dataset containing timings of receiving sms on mobile phone. Real data from 01 to 09 September, 2024

Usage`smsclock`**Format**

A data frame with 82 rows and 7 variables:

Date Date

day Day

time Time in HH:MM:SS format

sender Name of the sms sender

invoked Whether the sms is invoked by the user

type The category of the sms; offer, info, ad, service etc.

Title Message Title

Source

Phone Messages

`week_chart`*Plot Values on Each Day of a Week*

Description

This function plots values corresponding to each day on a rose plot.

Usage

```
week_chart(wvalue, lgnm = "Value", high = "yellow", low = "green", width = 0.9)
```

Arguments

wvalue	A numeric vector having values on each day, starting from Saturday
lgnm	Title of legend
high	The color name for the high values. The default is red
low	The color name for the low values. The default is green. The color names can be vice versa or other colors, depending on the context.
width	The width of bars.

Value

A ggplot object, which can be further modified with ggplot2 functions and themes.

See Also

[day_chart\(\)](#) for plotting values in on a day hours [year_chart\(\)](#) for plotting values on in a year by months [cyclic_chart\(\)](#) for plotting values by arbitrary period

Examples

```
set.seed(10)
wtemp <- sample(10:40, 7)
week_chart(wtemp, high = "yellow") + ggplot2::labs(title = "Random Values by Day")
```

year_chart

Plot Values on Each Month of a Year

Description

This function plots values corresponding to each month on a rose plot.

Usage

```
year_chart(mvalue, lgnm = "Value", width = 0.9, high = "yellow", low = "green")
```

Arguments

mvalue	A numeric vector of length 12, with values for each month (Jan–Dec). If you have it in a data frame, extract it (e.g. data\$mvalue).
lgnm	Title of the legend (default "Value").
width	Width of bars.
high	Color name for high values (default "yellow").
low	Color name for low values (default "green").

Value

A ggplot object, which can be further modified with ggplot2 functions.

See Also

[day_chart\(\)](#) for plotting values on a day by hours [week_chart\(\)](#) for plotting values in a week by days [cyclic_chart\(\)](#) for plotting values by arbitrary period

Examples

```
syltmp <- c(18.4, 20.8, 24.3, 26.0, 26.8, 27.6, 28.0,  
           28.2, 27.9, 26.7, 23.3, 19.7)  
year_chart(mvalue = syltmp)
```

Index

* datasets

- bdquake, [2](#)
- bdtemp, [3](#)
- brintcity, [3](#)
- chatdf, [4](#)
- gitcommit, [11](#)
- smclock, [13](#)

- bdquake, [2](#)
- bdtemp, [3](#)
- brintcity, [3](#)

- chatdf, [4](#)
- clock_chart, [5](#)
- clock_chart(), [8](#)
- clock_chart_col, [6](#)
- clock_chart_col(), [5, 7, 8](#)
- clock_chart_qlt, [7](#)
- clock_chart_qlt(), [5, 6, 8](#)
- clock_chart_qnt, [8](#)
- clock_chart_qnt(), [5-7](#)
- cyclic_chart, [9](#)
- cyclic_chart(), [10, 14, 15](#)

- day_chart, [10](#)
- day_chart(), [9, 14, 15](#)

- gitcommit, [11](#)

- plan_day, [11](#)
- plan_week, [12](#)

- smclock, [13](#)

- week_chart, [13](#)
- week_chart(), [9, 10, 15](#)

- year_chart, [14](#)
- year_chart(), [9, 10, 14](#)