Package: tsspiral (via r-universe)

October 15, 2026

Title Spiral Visualisation for Time Series

Version 0.1.1

Description Provides a grammar of graphics geometry for visualising time series using spiral layouts, with each year represented as a separate ring.

License GPL (>= 3)

Encoding UTF-8

Depends R (>= 4.1.0)

RoxygenNote 7.3.3

URL https://github.com/thiyangt/tsspiral, https://thiyangt.github.io/tsspiral/

BugReports https://github.com/thiyangt/tsspiral/issues

Imports ggplot2, dplyr, tibble, lubridate, purrr, tidyr, rlang

Suggests testthat (>= 3.0.0), tsibble

Config/testthat/edition 3

NeedsCompilation no

Author Thiyanga S. Talagala [aut, cre] (ORCID: <https://orcid.org/0000-0002-0656-9789>)

Maintainer Thiyanga S. Talagala <ttalagala@sjp.ac.lk>

Repository https://cran.r-universe.dev

Date/Publication 2026-10-15 17:20:02 UTC

RemoteUrl https://github.com/cran/tsspiral

RemoteRef HEAD

RemoteSha 343b1c87c393373c3dfbd3436e34c38d085fa572

Contents

calendar_hourly_polar 2
geom_tsspiral 4
geom_tsspiralbar 6
geom_tsspiralmiss 7
temporal_hierarchical 9

Index **12**

`calendar_hourly_polar` *Calendar of Hourly Polar Bar Charts*

Description

Creates a monthly calendar visualization in which each calendar day contains a 24-hour polar bar chart. Each bar represents one hour of the day, with the angular position corresponding to the hour and the bar length and colour representing the observed value.

Usage

```
calendar_hourly_polar(
  data,
  datetime,
  value,
  month = NULL,
  radius = 0.35,
  bar_width = 0.85,
  cell_width = 1,
  cell_height = 1
)
```

Arguments

`data`	A data frame containing the hourly observations.
`datetime`	A date-time column containing the observation times.
`value`	A numeric column containing the values to be visualized.
`month`	A date or character value specifying the month to plot. For example, `"2026-01-01"`. If `NULL`, the month containing the earliest observation is used.
`radius`	Numeric. Maximum radius of the hourly bars within each calendar cell. Defaults to `0.35`.
`bar_width`	Numeric between 0 and 1. Controls the angular width of each hourly bar relative to the 24-hour sector. Defaults to `0.85`.
`cell_width`	Numeric. Width of each calendar cell. Defaults to `1`.
`cell_height`	Numeric. Height of each calendar cell. Defaults to `1`.

Details

Missing hourly observations are retained in the calendar but are not drawn as bars. The calendar is arranged from Monday to Sunday, with each day positioned according to its calendar date.

Each day is represented by a separate polar bar chart containing up to 24 bars. Hour 0 starts at the top of the circle and the hours proceed clockwise through hour 23.

The bar length is scaled according to the range of the observed values in the selected month, while colour represents the original value. Missing hourly observations are left blank.

The function is useful for examining intraday patterns while retaining the context of the calendar. For example, daily temperature, air quality, electricity demand, traffic, or other regularly recorded hourly measurements can be displayed.

Value

A ggplot object containing a calendar of hourly polar bar charts.

Examples

```
set.seed(123)

hourly_data <- tibble::tibble(
  datetime = seq(
    from = as.POSIXct("2026-01-01 00:00"),
    to = as.POSIXct("2026-01-31 23:00"),
    by = "hour"
  )
) |>
  dplyr::mutate(
    hour = lubridate::hour(datetime),
    day = lubridate::day(datetime),
    temperature =
      25 +
      4 * sin(2 * pi * hour / 24) +
      0.05 * day +
      stats::rnorm(length(datetime), 0, 0.8)
  )

calendar_hourly_polar(
  hourly_data,
  datetime,
  temperature,
  month = "2026-01-01"
)
```

geom_tsspiral *Spiral Time-Series Geometry*

Description

Create a spiral time-series plot using a fixed 365-day calendar.

Usage

```
geom_tsspiral(
  mapping = NULL,
  data = NULL,
  stat = StatTSSpiral,
  position = "identity",
  na.rm = FALSE,
  ring_spacing = 1,
  ...
)
```

Arguments

`mapping`	Set of aesthetic mappings created by `ggplot2::aes()`.
`data`	A data frame, tibble, or tsibble containing the time-series data.
`stat`	Statistical transformation used by the layer. Defaults to `StatTSSpiral`.
`position`	Position adjustment. Defaults to `"identity"`.
`na.rm`	Logical. Should missing values be removed silently?
`ring_spacing`	Numeric value controlling the distance between yearly rings.
`...`	Other arguments passed to `ggplot2::layer()`.

Details

`geom_tsspiral()` displays each year of a time series as a radial ring. The angular position represents the day of the year. The spiral always contains 365 angular divisions.

The spiral starts at 12 o'clock on January 1 and proceeds clockwise through the year.

The input frequency is detected automatically. Daily observations are used directly. Weekly, monthly, quarterly, and annual observations are expanded to daily observations before plotting.

Calendar labels are displayed only around the outermost ring:

- daily data: month names,
- monthly data: month names,
- quarterly data: quarter names,
- weekly data: week numbers,
- annual data: month names.

The spiral uses a fixed 365-day calendar. January 1 starts at 12 o'clock and the calendar proceeds clockwise.

For day d, the angular position is

$$\theta_d = \frac{\pi}{2} - \frac{2\pi(d-1)}{365}.$$

Consequently, January 1 is positioned at 12 o'clock and later days move clockwise around the spiral.

Leap years do not create a 366th angular division. February 29 is mapped to the February 28 position and the days after February 29 are shifted by one day.

Monthly, quarterly, weekly, and annual observations are expanded to their corresponding daily intervals. Consequently, a monthly observation fills the complete angular region corresponding to that month rather than appearing as a thin radial slice.

Value

A ggplot2 layer.

Examples

```
library(ggplot2)
#' # daily data
dat <- data.frame(
  date = seq.Date(
    as.Date("2020-01-01"),
    as.Date("2023-12-31"),
    by = "day"
  )
)
dat$value <- sin(
  2 * pi * as.numeric(format(dat$date, "%j")) / 365
) + rnorm(nrow(dat), sd = 0.2)

ggplot(
  dat,
  aes(
    x = date,
    y = value,
    fill = value
  )
) +
  geom_tsspiral(ring_spacing = 2) +
  theme_void()
```

geom_tsspiralbar *Spiral Time-Series Bar Geometry*

Description

Create a spiral bar plot for time-series data.

Usage

```
geom_tsspiralbar(
  mapping = NULL,
  data = NULL,
  stat = StatTSSpiralBar,
  position = "identity",
  na.rm = FALSE,
  ring_spacing = 1,
  ...
)
```

Arguments

mapping	Set of aesthetic mappings created by ggplot2::aes().
data	A data frame, tibble, or tsibble.
stat	Statistical transformation used by the layer.
position	Position adjustment.
na.rm	Logical. Should missing values be removed silently?
ring_spacing	Numeric value controlling the distance between yearly rings.
...	Other arguments passed to ggplot2::layer().

Details

Each year is represented by a radial ring. The angular position represents time within the year, while the radial height of each bar represents the magnitude of the time-series value.

January 1 starts at 12 o'clock and the spiral proceeds clockwise.

Value

A ggplot2 layer.

Examples

```
library(ggplot2)

set.seed(123)

dat <- data.frame(
```

```
    date = seq.Date(
      as.Date("2021-01-01"),
      as.Date("2023-12-31"),
      by = "day"
    ),
    value = rnorm(1095)
  )

  ggplot(
    dat,
    aes(
      x = date,
      y = value,
      fill = value
    )
  ) +
    geom_tsspiralbar(
      ring_spacing = 2
    ) +
    scale_fill_viridis_c() +
    theme_void()
```

---

geom_tsspiralmiss *Spiral Missingness Geometry*

Description

Create a spiral plot showing missing observations in a time series.

Usage

```
geom_tsspiralmiss(
  mapping = NULL,
  data = NULL,
  stat = StatTSSpiralMiss,
  position = "identity",
  na.rm = FALSE,
  ring_spacing = 1,
  observed_colour = "grey85",
  missing_colour = "black",
  ...
)
```

Arguments

mapping	Set of aesthetic mappings created by `ggplot2::aes()`.
data	A data frame, tibble, or tsibble containing the time-series data.

stat	Statistical transformation used by the layer. Defaults to StatTSSpiralMiss.
position	Position adjustment. Defaults to "identity".
na.rm	Logical. Should missing values be removed silently?
ring_spacing	Numeric value controlling the distance between yearly rings.
observed_colour	Colour used for observed periods.
missing_colour	Colour used for missing periods.
...	Other arguments passed to ggplot2::layer().

Details

geom_tsspiralmiss() displays each year of a time series as a radial ring using a fixed 365-day calendar. Observed periods are shown in one colour and missing periods are shown in another.

The spiral starts at 12 o'clock on January 1 and proceeds clockwise through the year.

The input frequency is detected automatically. Daily observations are used directly. Weekly, monthly, quarterly, and annual observations are expanded to daily observations before plotting.

The geometry is designed specifically for exploring the temporal structure of missing observations.

January 1 is positioned at 12 o'clock and time proceeds clockwise. Each year is represented by one radial ring, with 365 fixed angular divisions.

A period is classified as observed when a corresponding observation is present in the input data. Periods without an observation are classified as missing.

Leap years do not create a 366th angular division. February 29 is mapped to the February 28 position and dates after February 29 are shifted by one day.

Value

A ggplot2 layer.

Examples

```
library(ggplot2)

set.seed(123)

df <- data.frame(
  date = seq.Date(
    as.Date("2020-01-01"),
    as.Date("2023-12-31"),
    by = "day"
  )
)
df$date[1:10] <- NA

ggplot(
  df,
  aes(x = date)
) +
```

```
  geom_tsspiralmiss(
    ring_spacing = 2
  ) +
  theme_void()
```

`temporal_hierarchical` *Aligned Temporal Hierarchical Spiral Plot*

Description

Visualise a time series at multiple temporal resolutions using concentric rings with a common temporal coordinate system.

Usage

```
temporal_hierarchical(
  data,
  date,
  value,
  fun = mean,
  ring_width = 1,
  ring_gap = 0.05,
  na.value = "grey90",
  daily = TRUE,
  weekly = TRUE,
  monthly = TRUE,
  quarterly = TRUE,
  yearly = TRUE,
  direction = 1,
  start_angle = -pi/2
)
```

Arguments

`data`	A data frame containing a date variable and a numeric variable.
`date`	Unquoted name of the date column.
`value`	Unquoted name of the numeric variable to visualise.
`fun`	Aggregation function used for weekly, monthly, quarterly, and yearly aggregation. Defaults to `mean`.
`ring_width`	Width of each ring. Defaults to `1`.
`ring_gap`	Gap between rings. Defaults to `0.05`.
`na.value`	Colour used for missing values. Defaults to `"grey90"`.
`daily`	Logical; draw the daily ring. Defaults to `TRUE`.

`weekly`	Logical; draw the weekly ring. Defaults to `TRUE`.
`monthly`	Logical; draw the monthly ring. Defaults to `TRUE`.
`quarterly`	Logical; draw the quarterly ring. Defaults to `TRUE`.
`yearly`	Logical; draw the yearly ring. Defaults to `TRUE`.
`direction`	Direction of the temporal axis. `1` produces a clockwise direction and `-1` produces an anticlockwise direction.
`start_angle`	Starting angle in radians. Defaults to `-pi / 2`, placing the beginning of the series at the top.

Details

The outer ring represents daily observations, followed inward by weekly, monthly, quarterly, and yearly observations. Temporal boundaries are aligned across all rings.

All temporal levels use a common angular coordinate based on the complete date range. Consequently, observations at different temporal resolutions are spatially aligned.

For example, the monthly January ring occupies the same angular region as the corresponding daily observations, weeks, quarter, and year.

The outer-to-inner hierarchy is:

- Daily
- Weekly
- Monthly
- Quarterly
- Yearly

Value

A `ggplot2` object.

Examples

```
set.seed(123)

dat <- data.frame(
  date = seq.Date(
    from = as.Date("2020-01-01"),
    to = as.Date("2023-12-31"),
    by = "day"
  )
)

dat$value <- 20 +
  5 * sin(
    2 * pi * lubridate::yday(dat$date) / 365.25
  ) +
  rnorm(nrow(dat), 0, 1)
```

```
temporal_hierarchical(
  data = dat,
  date = date,
  value = value
)
```

Index

calendar_hourly_polar, 2

geom_tsspiral, 4
geom_tsspiralbar, 6
geom_tsspiralmiss, 7
ggplot2::aes(), *4*, *6*, *7*
ggplot2::layer(), *4*, *6*, *8*

temporal_hierarchical, 9