Package: ggkodom (via r-universe)

June 21, 2026

Title Visualize Individual Longitudinal Trajectories

Version 1.0.3

Description A 'ggplot2'-based toolkit for visualizing individual-level longitudinal trajectories. Creates linear kodom plots, circular kodom plots, heatmaps, and state-ribbon charts for repeated-measures data. Each subject gets its own visual lane with measurements colored by value, revealing patterns across subjects and time. The circular variant resembles the Kodom flower.

License MIT + file LICENSE

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports ggplot2 (>= 3.4.0), rlang, scales

Suggests testthat (>= 3.0.0), lme4, mgcv, refund, fdapace, patchwork, knitr, rmarkdown, spelling

VignetteBuilder knitr

Config/testthat/edition 3

Repository https://subroy13.r-universe.dev

Date/Publication 2026-06-15 21:07:23 UTC

RemoteUrl https://github.com/subroy13/ggkodom

RemoteRef HEAD

RemoteSha 6a2970a3b477cc2d6b09180539e55b5a2357bb7a

Contents

`coord_kodom_periodic` *Polar coordinate system for periodic Kodom plots*

Description

A convenience wrapper around `ggplot2::coord_polar()` with defaults appropriate for `geom_kodom_periodic()`: `theta = "x"` (time maps to angle), `start = pi/2` (x = 0 at 12 o'clock), and `direction = -1` (clockwise). Pass `clockwise = FALSE` for a counter-clockwise layout.

Usage

```
coord_kodom_periodic(clockwise = TRUE)
```

Arguments

`clockwise`	Logical. `TRUE` (default) places time clockwise from the top, matching the convention of most clock and calendar displays.

Value

A `ggplot2::coord_polar()` coordinate object.

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = visit_month, id = subject_id, colour = hba1c)) +
  geom_kodom_periodic(period = 12) +
```

```
  scale_x_continuous(breaks = 1:12, labels = month.abb, limits = c(0, 12)) +
  scale_colour_kodom() +
  coord_kodom_periodic() +
  theme_kodom_periodic()
```

`geom_kodom_branch` *Branching swimlane plot for observed paths and counterfactual predictions*

Description

Draws one horizontal path per subject for the observed trajectory, then fans out into sub-lanes — one per medication or intervention arm — for predicted (counterfactual) trajectories. The branching time may differ across subjects.

Usage

```
geom_kodom_branch(
  mapping = NULL,
  data = NULL,
  stat = StatKodomBranch,
  position = "identity",
  ...,
  sort_by = "none",
  n_max = Inf,
  lane_width = 1,
  branch_fraction = 0.7,
  show_points = TRUE,
  show_fork = TRUE,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```

Arguments

`mapping`	Set of aesthetic mappings created by `ggplot2::aes()`.
`data`	A data frame. If `NULL`, inherits from the plot.
`stat`	The stat to use. Defaults to StatKodomLine.
`position`	Position adjustment, almost always `"identity"`.
`...`	Other arguments passed to `ggplot2::layer()`.
`sort_by`	Lane ordering. One of `"none"` (default), `"mean"`, `"mean_asc"`, `"first"`, `"last"` — all refer to the `colour` variable.
`n_max`	Maximum number of subjects to display. A random sample is taken when exceeded. Default `Inf` (all subjects).

lane_width	Positive numeric. Vertical distance between adjacent subject lanes. Default 1.
branch_fraction	Fraction of lane_width devoted to prediction sub-lanes. Default 0.7. With K arms the sub-lane step is branch_fraction / K.
show_points	If TRUE (default), draws a point at every observation. Set to FALSE, or map shape = NA / size = 0, to suppress points.
show_fork	Logical. If TRUE (default), draws a vertical fork connector at each subject's branch point.
na.rm	If TRUE, silently remove rows with missing required aesthetics.
show.legend	Logical. Should this layer appear in the legend?
inherit.aes	If FALSE, overrides the default aesthetics.

Details

Data format. Supply NA in the medication column for all observed rows (pre- and post-branch). Supply a non-NA arm label (e.g. "DrugA") for every predicted row. The stat replaces NA with "observed" in its output so that the linetype scale receives a clean string for every row.

Linetype. Map linetype = <arm_column> in aes() **and** supply a scale_linetype_manual() so that the legend appears and the linetypes are exactly what you want. The stat converts NA (observed rows) to the string "observed" before the scale is applied, so target that key explicitly:

```
aes(linetype = arm, medication = arm)   # both point to the same column
scale_linetype_manual(
  values = c("observed" = "solid", "DrugA" = "dashed", "DrugB" = "dotted")
)
```

Fork connector. A short vertical segment is drawn at each subject's branch point (the first x that appears in a predicted arm), connecting the observed lane to the topmost arm. Suppress with show_fork = FALSE.

Lane layout. Each subject occupies a primary band of width lane_width. Within that band the observed path is at the base; each prediction arm sits at an equal sub-lane fraction above it controlled by branch_fraction. With two arms and branch_fraction = 0.7, arm 1 is at 0.35 * lane_width above the base and arm 2 is at 0.70 * lane_width, leaving 30% clearance before the next subject.

Y-axis labels. Subject IDs can be added by setting custom breaks:

```
scale_y_continuous(
  breaks = seq_len(n_subjects) * lane_width,
  labels = subject_ids
)
```

Value

A ggplot2 layer object.

Aesthetics

- `x` — time (numeric or Date)
- `id` — subject identifier; determines the primary lane position
- `colour` — measured value mapped to colour (interpolated along path)
- `medication` — `NA` for observed rows; a character/factor arm label for predicted rows. Each unique non-NA value becomes one sub-lane.
- `size`, `linewidth`, `alpha`, `shape`, `stroke` — standard path/point aesthetics. `linetype` is set by the stat (medication name / `"observed"`).

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(
  x = time, id = subject_id,
  colour = hba1c, linetype = arm, medication = arm
)) +
  geom_kodom_branch(sort_by = "mean", lane_width = 2) +
  scale_linetype_manual(
    values = c("observed" = "solid", "DrugA" = "dashed", "DrugB" = "dotted")
  ) +
  scale_colour_kodom() +
  theme_kodom()
```

---

geom_kodom_circular *Circular ("Kodom") swimlane plot for longitudinal trajectories*

Description

Draws a radial plot where each subject occupies an angular spoke and time increases outward from the center. Measurements are encoded as a colour gradient along each radial path, interpolated between consecutive observations. The layout resembles a Kadam flower (*Neolamarckia cadamba*), giving the package its name.

Usage

```
geom_kodom_circular(
  mapping = NULL,
  data = NULL,
```

```
  stat = StatKodomCircular,
  position = "identity",
  ...,
  sort_by = "none",
  n_max = Inf,
  gap_fraction = 0.15,
  inner_fraction = 0.3,
  direction = 1L,
  show_points = TRUE,
  reference_rings = c(0),
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```

Arguments

mapping	Set of aesthetic mappings created by ggplot2::aes().
data	A data frame. If NULL, inherits from the plot.
stat	The stat to use. Defaults to StatKodomCircular.
position	Position adjustment, almost always "identity".
...	Other arguments passed to ggplot2::layer().
sort_by	Lane ordering (angular order). One of "none" (default), "mean", "mean_asc", "first", "last".
n_max	Maximum number of subjects to display. Default Inf.
gap_fraction	Fraction of the full circle left empty as a visual gap at the ordering seam. Default 0.15.
inner_fraction	Fraction of the time range used as a hollow inner buffer so short-follow-up subjects remain visible. Default 0.3.
direction	1L = clockwise (default), -1L = counter-clockwise.
show_points	If TRUE (default), draws a point at every observation. Set to FALSE, or map shape = NA / size = 0, to suppress points.
reference_rings	Numeric vector of time points where concentric reference rings should be drawn behind the data paths. By default, draws a ring at 0 (baseline). Set to NULL to suppress rings.
na.rm	If TRUE, silently remove rows with missing required aesthetics.
show.legend	Logical. Should this layer appear in the legend?
inherit.aes	If FALSE, overrides the default aesthetics.

Details

Unlike the original coord_polar() variant in the archive package, this geom performs the polar-to-Cartesian transform inside the stat and renders in ordinary Cartesian space, so all ggplot2 scales, themes, and facets work normally. Use coord_fixed() to preserve the circular shape.

Value

A ggplot2 layer object.

Aesthetics

Required aesthetics are shown in **bold**. `size` and `linewidth` are intentionally independent: each controls a different graphical element.

- `x` — time variable (numeric); mapped to radial distance
- `id` — subject identifier; mapped to angular position
- `colour` — color of both the connecting path and the point border
- `fill` — point interior color for filled shapes (21–25)
- `size` — **point size only**; set to `NA` or `0` to suppress points
- `linewidth` — **path width only**
- `alpha` — transparency applied to both path and points
- `shape` — point shape (default 19). Set to `NA` to suppress points.
- `stroke`, `linetype` — point border width / path line type

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = time, id = subject_id, colour = hba1c)) +
  geom_kodom_circular(sort_by = "mean", gap_fraction = 0.1) +
  scale_colour_kodom() +
  coord_fixed() +
  theme_kodom_circular()
```

`geom_kodom_heatmap` *Swimlane heatmap for longitudinal trajectories*

Description

Divides the time axis into equal-width bins and fills each (subject × bin) cell with an aggregate of the measured value. Each subject occupies one horizontal lane. Use this layout when the cohort is large or when you prefer a compact, aligned grid over individual paths.

Usage

```
geom_kodom_heatmap(
  mapping = NULL,
  data = NULL,
  stat = StatKodomHeatmap,
  position = "identity",
  ...,
  sort_by = "mean",
  n_max = Inf,
  bins = 10L,
  breaks = NULL,
  fun = "mean",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```

Arguments

mapping	Set of aesthetic mappings created by ggplot2::aes().
data	A data frame. If NULL, inherits from the plot.
stat	The stat to use. Defaults to StatKodomHeatmap.
position	Position adjustment, almost always "identity".
...	Other arguments passed to ggplot2::layer().
sort_by	Lane ordering. One of "none", "mean" (default), "mean_asc", "first", "last" — all refer to the fill variable.
n_max	Maximum number of subjects to display. A random sample is taken when exceeded. Default Inf (all subjects).
bins	Number of equal-width time bins. Default 10L. Ignored if breaks is supplied.
breaks	Numeric vector of explicit bin boundaries. Overrides bins.
fun	Aggregation function per cell. One of "mean" (default), "median", "first", "last", "min", "max".
na.rm	If TRUE, silently remove rows with missing required aesthetics.
show.legend	Logical. Should this layer appear in the legend?
inherit.aes	If FALSE, overrides the default aesthetics.

Details

Lane ordering is controlled by sort_by (defaults to "mean", which places the highest-mean subjects at the top). Time bins can be customised via bins (number of equal-width intervals) or breaks (explicit boundaries).

Value

A ggplot2 layer object.

Aesthetics

Required aesthetics are shown in **bold**.

- x — time variable (numeric or Date)
- id — subject identifier; determines lane position on the y axis
- fill — measurement value used for cell color and sorting
- colour — tile border color (default "white")
- linewidth — tile border width (default 0.25)
- alpha — transparency

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = time, id = subject_id, fill = hba1c)) +
  geom_kodom_heatmap(sort_by = "mean", bins = 12) +
  scale_fill_kodom() +
  theme_kodom()
```

---

geom_kodom_line *Swimlane line plot for longitudinal trajectories*

Description

Draws one horizontal colored path per subject. Time maps to x, subject identity maps to id (which the stat converts to integer lane positions on the y axis), and a measured value maps to colour. The colour transitions smoothly between consecutive observations via interpolated sub-segments.

Usage

```
geom_kodom_line(
  mapping = NULL,
  data = NULL,
  stat = StatKodomLine,
  position = "identity",
  ...,
  sort_by = "none",
  n_max = Inf,
  show_points = TRUE,
```

```
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```

Arguments

mapping	Set of aesthetic mappings created by ggplot2::aes().
data	A data frame. If NULL, inherits from the plot.
stat	The stat to use. Defaults to StatKodomLine.
position	Position adjustment, almost always "identity".
...	Other arguments passed to ggplot2::layer().
sort_by	Lane ordering. One of "none" (default), "mean", "mean_asc", "first", "last" — all refer to the colour variable.
n_max	Maximum number of subjects to display. A random sample is taken when exceeded. Default Inf (all subjects).
show_points	If TRUE (default), draws a point at every observation. Set to FALSE, or map shape = NA / size = 0, to suppress points.
na.rm	If TRUE, silently remove rows with missing required aesthetics.
show.legend	Logical. Should this layer appear in the legend?
inherit.aes	If FALSE, overrides the default aesthetics.

Details

Lane ordering is controlled by sort_by. With many subjects, consider suppressing y-axis labels via theme(axis.text.y = element_blank()).

Value

A ggplot2 layer object.

Aesthetics

Required aesthetics are shown in **bold**. size and linewidth are intentionally independent: each controls a different graphical element.

- x — time variable (numeric or Date)
- id — subject identifier; determines lane position on the y axis
- colour — color of both the connecting path and the point border
- fill — point interior color for filled shapes (21–25); no effect on path
- size — **point size only**; set to NA or 0 to suppress points
- linewidth — **path width only**; does not affect observation points
- alpha — transparency applied to both path and points
- shape — point shape (default 19). Set to NA to suppress points.
- stroke, linetype — point border width / path line type

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = time, id = subject_id, colour = hba1c)) +
  geom_kodom_line(sort_by = "mean", n_max = 50) +
  scale_colour_kodom() +
  theme_kodom()
```

---

geom_kodom_periodic *Periodic "Star Trail" swimlane plot for longitudinal trajectories*

Description

Draws concentric arcs representing cyclical longitudinal data. Time maps to angle (one full revolution equals one `period`), and each subject's base radius is determined by its lane rank. To separate successive cycles of the same subject the radius drifts outward slightly as total time increases, forming a star trail or Archimedean spiral.

Usage

```
geom_kodom_periodic(
  mapping = NULL,
  data = NULL,
  stat = StatKodomPeriodic,
  position = "identity",
  ...,
  period = 12,
  spiral_fraction = 0.1,
  inner_fraction = 0.3,
  lane_width = 1,
  sort_by = "none",
  n_max = Inf,
  show_points = TRUE,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```

Arguments

mapping	Set of aesthetic mappings created by ggplot2::aes().
data	A data frame. If NULL, inherits from the plot.
stat	The stat to use. Defaults to StatKodomLine.
position	Position adjustment, almost always "identity".
...	Other arguments passed to ggplot2::layer().
period	Numeric. Length of one complete cycle (e.g. 12 for months, 24 for hours). Default 12.
spiral_fraction	Numeric. Radial expansion per full cycle as a fraction of one *original* lane width (i.e. before lane_width scaling). 0 keeps cycles on the same ring; 0.1 (default) drifts outward by 10% of one lane per cycle.
inner_fraction	Fraction of total lanes used as a hollow inner buffer. Default 0.3. Pair with scale_y_kodom_periodic() to make the hole visible.
lane_width	Positive numeric. Multiplier for the radial distance between adjacent subject rings. Default 1 packs rings at unit spacing. Increase (e.g. 3) to spread rings apart so each subject's arc is more legible — particularly useful when using n_max to show a small subset. The hollow center (inner_fraction) is intentionally not scaled so the hole size stays proportional to the cohort, not the spacing.
sort_by	Lane ordering. One of "none" (default), "mean", "mean_asc", "first", "last" — all refer to the colour variable.
n_max	Maximum number of subjects to display. A random sample is taken when exceeded. Default Inf (all subjects).
show_points	If TRUE (default), draws a point at every observation. Set to FALSE, or map shape = NA / size = 0, to suppress points.
na.rm	If TRUE, silently remove rows with missing required aesthetics.
show.legend	Logical. Should this layer appear in the legend?
inherit.aes	If FALSE, overrides the default aesthetics.

Details

Coordinate system. This geom is designed for use with coord_polar(theta = "x"). Use coord_kodom_periodic() as a convenient wrapper that sets start = pi/2 (12 o'clock at x = 0) and direction = -1 (clockwise) by default.

Axis labels. Because x is continuous time, ggplot2 will automatically expand the limits to cover the entire time range if you use standard scales, which alters the length of a full 360-degree rotation. To make one rotation exactly equal one period, use the included wrapper scale_x_kodom_periodic():

```
scale_x_kodom_periodic(period = 12, breaks = 1:12, labels = month.abb)
```

Value

A ggplot2 layer object.

kodom_colors *Kodom color palette*

Description

Returns colors inspired by the Kadam flower (*Neolamarckia cadamba*): teal (low), gold (mid), red (high). Passing n > 3 interpolates between the three anchors via `grDevices::colorRampPalette()`.

Usage

```
kodom_colors(n = 3L)
```

Arguments

n	Number of colors. 3 returns the anchor set; >3 interpolates.

Value

Character vector of hex color codes.

Examples

```
kodom_colors()
kodom_colors(7)
```

scale_color_kodom *Kodom color scale for the color aesthetic*

Description

A ggplot2 color scale using the Kadam flower palette (teal -> gold -> red). Designed to compose with `geom_kodom_line()`, `geom_kodom_heatmap()`, and `geom_kodom_circular()`.

Usage

```
scale_color_kodom(
  colors = kodom_colors(),
  color_breaks = NULL,
  name = ggplot2::waiver(),
  discretize = FALSE,
  ...
)

scale_colour_kodom(
  colors = kodom_colors(),
  color_breaks = NULL,
```

```
  name = ggplot2::waiver(),
  discretize = FALSE,
  ...
)
```

Arguments

colors	Color vector. Defaults to kodom_colors().
color_breaks	Numeric breakpoints anchoring the gradient via scales::rescale(). For discretize = TRUE, each break adds one band: k breaks produce k + 1 bands. NULL = evenly spaced.
name	Legend title. Defaults to "value".
discretize	FALSE (default) = smooth gradient; TRUE = step bands at color_breaks.
...	Ignored.

Details

Set discretize = TRUE to switch from a smooth gradient to solid color bands separated at color_breaks — useful when values change slowly and a continuous gradient washes out into a single hue.

Value

A ggplot2 scale object.

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = time, id = subject_id, color = value)) +
  geom_kodom_line() +
  scale_color_kodom()

# Discrete bands at clinical thresholds
ggplot(df, aes(x = time, id = subject_id, color = hba1c)) +
  geom_kodom_line() +
  scale_color_kodom(discretize = TRUE, color_breaks = c(5.7, 6.5))
```

scale_fill_kodom *Kodom fill scale for the fill aesthetic*

Description

Identical behavior to scale_color_kodom() but targets the fill aesthetic. Intended for use with geom_kodom_heatmap().

Usage

```
scale_fill_kodom(
  colors = kodom_colors(),
  color_breaks = NULL,
  name = ggplot2::waiver(),
  discretize = FALSE,
  ...
)
```

Arguments

colors	Color vector. Defaults to kodom_colors().
color_breaks	Numeric breakpoints anchoring the gradient via scales::rescale(). For discretize = TRUE, each break adds one band: k breaks produce k + 1 bands. NULL = evenly spaced.
name	Legend title. Defaults to "value".
discretize	FALSE (default) = smooth gradient; TRUE = step bands at color_breaks.
...	Ignored.

Value

A ggplot2 scale object.

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = time, id = subject_id, fill = value)) +
  geom_kodom_heatmap() +
  scale_fill_kodom()
```

scale_x_kodom_periodic

X-scale for periodic longitudinal plots

Description

This is a convenience wrapper around ggplot2::scale_x_continuous() that enforces limits = c(0, period) and uses oob = scales::oob_keep.

Usage

```
scale_x_kodom_periodic(period = 12, breaks = 1:period, ...)
```

Arguments

period	The length of one complete cycle (e.g. 12 for months). Must match the period argument passed to geom_kodom_periodic(). Default 12.
breaks	Passed to scale_x_continuous(). Default provides integer breaks for the period.
...	Additional arguments passed to ggplot2::scale_x_continuous(), such as labels.

Details

Why is this necessary? To make exactly one cycle span exactly one 360-degree rotation in coord_polar, the scale limits *must* be set to the period length (e.g., c(0, 12)). However, standard ggplot2 scale_x_continuous will drop any data outside those limits. By setting oob = scales::oob_keep, we instruct ggplot2 to keep the data that exceeds the period. coord_polar then natively wraps those out-of-bounds values around the circle, creating beautiful, continuous Archimedean spirals!

Value

A ggplot2::scale_x_continuous() object.

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = visit_month, id = subject_id, colour = hba1c)) +
  geom_kodom_periodic(period = 12) +
```

```
  scale_x_kodom_periodic(period = 12, labels = month.abb) +
  scale_y_kodom_periodic() +
  coord_kodom_periodic() +
  theme_kodom_periodic()
```

`scale_y_kodom_periodic` *Radial scale for periodic Kodom plots*

Description

In `coord_polar(theta = "x")`, ggplot2 sets the center of the plot to the *minimum data y value*, not to zero. Because `geom_kodom_periodic()` stores the hollow-center gap as an offset above zero (`inner_fraction * n_lanes`), the gap is invisible without this scale: the auto-range simply absorbs it.

Usage

```
scale_y_kodom_periodic(expand = ggplot2::expansion(mult = c(0, 0.05)))
```

Arguments

`expand`	Passed to `ggplot2::scale_y_continuous()`. Default adds 5% padding beyond the outermost ring and no padding at the center.

Details

This function pins `y = 0` at the center by setting `limits = c(0, NA)`, so the offset computed by `inner_fraction` becomes a visible donut hole. It should be added to every plot that uses `geom_kodom_periodic()`.

Value

A `ggplot2::scale_y_continuous()` object.

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = visit_month, id = subject_id, colour = hba1c)) +
  geom_kodom_periodic(period = 12, inner_fraction = 0.3) +
  scale_x_continuous(breaks = 0:11, labels = month.abb) +
```

```
  scale_y_kodom_periodic() +
  scale_colour_kodom() +
  coord_kodom_periodic() +
  theme_kodom_periodic()
```

---

theme_kodom *Clean theme for ggkodom linear and heatmap plots*

Description

Based on ggplot2::theme_minimal(): vertical x-axis grid lines mark time positions, horizontal y-axis grid lines are suppressed (they would bisect subject lanes and add clutter). Restrained axis text, wide horizontal color-bar legend. Suitable for geom_kodom_line() and geom_kodom_heatmap().

Usage

```
theme_kodom(base_size = 12, legend_position = "top")
```

Arguments

base_size	Base font size (default 12).
legend_position	One of "top", "bottom", "left", "right", "none".

Details

The vertical grid lines can be suppressed or restyled via the usual ggplot2 theme() override:

```
+ theme(panel.grid.major.x = element_blank())
```

Value

A ggplot2::theme() object.

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = time, id = subject_id, colour = value)) +
  geom_kodom_line() +
  scale_colour_kodom() +
  theme_kodom()
```

`theme_kodom_circular` *Theme for circular ggkodom plots*

Description

Fully minimal Cartesian theme for `geom_kodom_circular()`: no axis text, ticks, or grid lines. Reference rings are drawn by the geom itself as annotation layers, so this theme suppresses all panel grid elements.

Usage

```
theme_kodom_circular(base_size = 12)
```

Arguments

`base_size`	Base font size (default 12).

Details

Unlike the original `coord_polar()`-based approach, `geom_kodom_circular()` performs its coordinate transformation inside the stat and renders in ordinary Cartesian space. This means `panel.grid.major.y` lines would appear as horizontal rules, not concentric circles — so both major axes are suppressed here.

Value

A `ggplot2::theme()` object.

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = time, id = subject_id, colour = value)) +
  geom_kodom_circular() +
  scale_colour_kodom() +
  coord_fixed() +
  theme_kodom_circular()
```

`theme_kodom_periodic` *Theme for periodic Kodom plots*

Description

Designed for use with `geom_kodom_periodic()` and `coord_kodom_periodic()`. Suppresses the radial (y) axis — which shows lane numbers that are not meaningful to readers — while keeping the angular (x) axis so that time labels added via `ggplot2::scale_x_continuous()` are visible around the ring.

Usage

```
theme_kodom_periodic(base_size = 12)
```

Arguments

`base_size`	Base font size (default 12).

Value

A `ggplot2::theme()` object.

Examples

```
library(ggplot2)
df <- data.frame(
  subject_id = rep(1:5, each = 4),
  time = rep(1:4, 5),
  visit_month = rep(1:4, 5),
  value = rep(1:4, 5),
  hba1c = rep(1:4, 5),
  arm = rep(c("Treatment", "Control"), c(12, 8))
)
ggplot(df, aes(x = visit_month, id = subject_id, colour = hba1c)) +
  geom_kodom_periodic(period = 12) +
  scale_x_continuous(breaks = 1:12, labels = month.abb, limits = c(0, 12)) +
  scale_colour_kodom() +
  coord_kodom_periodic() +
  theme_kodom_periodic()
```

Index