Package: geovizr (via r-universe)

June 10, 2026

Type Package

Title Interactive Cartography

Version 1.0.0

Description Create a wide range of interactive, zoomable vector maps. This package is an 'R' binding for the 'geoviz' JavaScript library <https://github.com/riatelab/geoviz/>, itself based on the 'd3.js' ecosystem <doi:10.1109/TVCG.2011.185>. Like the original javascript library, the package takes advantage of d3's many features: proportional symbols, pictograms, typologies, choropleth maps, Spikes, tiles, Dorling cartograms, etc. It can also be used to create pretty static vectorial maps in svg format, suitable for editorial cartography.

License GPL (>= 3)

URL https://riatelab.github.io/geovizr/

BugReports https://github.com/riatelab/geovizr/issues/

Encoding UTF-8

Depends R (>= 4.1.0)

Imports htmlwidgets, geojsonsf, jsonlite, sf

Language en-US

Config/roxygen2/version 8.0.0

Config/pak/sysreqs libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev make libuv1-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev

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

Date/Publication 2026-06-10 07:26:10 UTC

RemoteUrl https://github.com/riatelab/geovizr

RemoteRef HEAD

RemoteSha f53e6b5a9f13b0724f15f4a4ac13c2078c9a3c0d

Contents

`geovizr` *Package description*

Description

Create a wide range of interactive, zoomable vector maps. This package is an R binding for the geoviz javascript library (https://github.com/riatelab/geoviz), itself based on the d3.js ecosystem ported by Mike Bostock (https://d3js.org/). Like the original javascript library, the package takes advantage of d3's many features: proportional symbols, pictograms, typologies, choropleth maps, Spikes, tiles, Dorling cartograms, etc. It can also be used to create pretty static vectorial maps in svg format, suitable for editorial cartography.

Author(s)

Maintainer: Nicolas Lambert <nicolas.lambert@cnrs.fr> (ORCID)

Authors:

- Nicolas Lambert <nicolas.lambert@cnrs.fr> (ORCID)

Other contributors:

- Timothée Giraud <timothee.giraud@cnrs.fr> (ORCID) [contributor]

See Also

Useful links:

- https://riatelab.github.io/geovizr/
- Report bugs at https://github.com/riatelab/geovizr/issues/

viz_blur *Blur filter*

Description

The viz_blur function creates an SVG blur filter and adds it to the map definitions. It returns a reference usable in SVG styling (e.g. url(#id)).

Usage

```
viz_blur(map, id = NULL, stdDeviation = 1.5, ...)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique filter id.
stdDeviation	numeric. Optional. Standard deviation controlling blur intensity (default 1.5). Higher values produce a stronger blur effect.
...	Additional parameters

Value

A modified ‘geoviz‘ map object with a new effect added. Rendering is performed using viz_render()

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
aus <- world[world$ISO3 == "AUS", ]

viz_create(margin = 5, width = 650) |>
  viz_blur(id = "my_blur_effect", stdDeviation = 2) |>
  viz_path(datum = aus, fill = "#38896F", filter = "url(#my_blur_effect)") |>
  viz_render()
```

`viz_choro` *Choropleth layer*

Description

The `viz_choro` function creates a choropleth map from a spatial data frame by classifying a numeric variable and mapping it to a color palette. It supports multiple classification methods and automatic legend generation.

Usage

```
viz_choro(
  map,
  data = NULL,
  var,
  method = "quantile",
  nb = 6,
  breaks = NULL,
  colors = NULL,
  middle = FALSE,
  sd = 1,
  reverse = FALSE,
  missing = "white",
  legend = TRUE,
  leg_type = "vertical",
  leg_pos = c(10, 10),
  ...
)
```

Arguments

`map`	A `geovizr` map object created using `viz_create()`.
`data`	A spatial dataframe Use `data` to enable iteration over features.
`var`	character. Name of the numeric variable used for classification. You can also use `fill` or `stroke` directly instead of `var`.
`method`	character. Optional. Classification method (default "quantile"). One of: "quantile", "q6", "equal", "jenks", "msd", "geometric", "headtail", "pretty", "arithmetic", "nestedmeans".
`nb`	numeric. Optional. Number of classes (default 6).
`breaks`	numeric vector. Optional. Manual class breaks. Overrides `nb` and `method`.
`colors`	character or vector. Optional. Color palette or vector of colors. Can use named palettes from the Dicopal library. See https://observablehq.com/@neocartocnrs/dicopal-library
`middle`	logical. Optionnal Only for the MSD method: is the mean in a central class or not? (default FALSE)

sd	number. Only for the MSD method: the number of standard deviations taken into account (default 1)
reverse	logical. Optional. Reverse color palette (default FALSE).
missing	character or logical. Optional. Color for missing values (default "white").
legend	logical. Optional. Whether to display a legend (default TRUE).
leg_type	character. Optional. Legend orientation (default "vertical"). One of: "horizontal", "vertical".
leg_pos	numeric vector. Optional. Legend position (default c(10, 10)).
...	Additional parameters passed to path rendering. With the leg_ prefix, you can configure the legend. For example: leg_title, leg_subtitle, leg_note, etc.

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(
  projection = "EqualEarth", background = "white",
  zoomable = TRUE
) |>
  viz_choro(
    data = world, var = "gdppc", method = "quantile", nb = 5,
    leg_values_round = 0, leg_title = "GDP\nper\ncapita",
    leg_subtitle = "(in $/inh.)", colors = "PinkYl"
  ) |>
  viz_render()
```

---

viz_circle *Circle layer*

Description

The viz_circle function draws circles on the map from a spatial data frame or from a single position. It can be used to create proportional symbol maps with optional collision avoidance.

Usage

```
viz_circle(
  map,
  data = NULL,
  id = NULL,
  pos = c(0, 0),
  r = 10,
  k = 50,
  fixmax = NULL,
  dodge = FALSE,
  iteration = 200,
  sort = NULL,
  descending = NULL,
  coords = "geo",
  fill = NULL,
  stroke = "white",
  tip = FALSE,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	object. Optional. A spatial data frame.
id	character. Optional. Unique layer id.
pos	numeric vector. Optional. Position of a single circle (default c(0, 0)).
r	numeric or character. Optional. Circle radius (default 10). Can be a fixed value or the name of a field containing numerical values.
k	numeric. Optional. Radius of the largest circle (default 50).
fixmax	numeric. Optional. Value corresponding to the circle of radius k. Useful to ensure comparability between maps.
dodge	logical. Optional. Avoid circle overlap (default FALSE).
iteration	numeric. Optional. Number of iterations for dodging (default 200).
sort	character or function. Optional. Field name or function to sort circles.
descending	logical. Optional. Sorting order.
coords	character. Optional. Coordinate system (default "geo"). Use "svg" if coordinates are already expressed in the SVG coordinate space.
fill	character or function. Optional. Fill color.
stroke	character or function. Optional. Stroke color (default "white").
tip	logical or function. Optional. Tooltip definition (default FALSE). Use TRUE to display all fields.
...	Additional SVG attributes (e.g. strokeDasharray, strokeWidth, opacity, strokeLinecap, etc.).

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_circle(data = world, r = 30, fill = "#38896F") |>
  viz_render()
```

`viz_clipPath` *ClipPath layer*

Description

The `viz_clipPath` function creates an SVG clipPath definition and adds it to the SVG defs. It returns a reference usable in SVG styling (e.g. `url(#id)`). WARNING: the clipPath is global to the web page, not only the map.

Usage

```
viz_clipPath(map, id = NULL, datum = list(type = "Sphere"), permanent = FALSE)
```

Arguments

`map`	A `geovizr` map object created using `viz_create()`.
`id`	character. Optional. Unique clipPath id.
`datum`	object. Optional. Geometry used for clipping (default `list(type = "Sphere")`).
`permanent`	logical or character. Optional. Whether the clipPath is static (default FALSE).

Value

A modified 'geoviz' map object with a new effect added. Rendering is performed using `viz_render()`

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
aus <- world[world$ISO3 == "AUS", ]
```

```
viz_create(margin = 5, projection = "meractor") |>
  viz_clipPath(id = "ausclip", datum = aus) |>
  viz_tile(url = "worldStreet", clipPath = "url(#ausclip)") |>
  viz_render()
```

`viz_create` *Create a geoviz map container*

Description

The `viz_create` function initializes a geovizr map by creating an SVG container and defining its main configuration (projection, domain, size, margins, and layout). It is the first step in any map construction workflow.

Usage

```
viz_create(
  id = "map",
  width = NULL,
  resize = TRUE,
  domain = NULL,
  responsive = TRUE,
  projection = NULL,
  background = NULL,
  fontFamily = NULL,
  margin = c(0, 0, 0, 0),
  zoomable = NULL,
  control = TRUE,
  warning = TRUE
)
```

Arguments

`id`	character. Optional. ID of the SVG container (default "map").
`width`	numeric. Optional. It allows you to manually set the width of the map. (default size of the device). Note that if responsive = TRUE, the SVG always has the size defined by the width in map coordinates, which does not correspond to the actual displayed size of the map. Similarly, if resize = TRUE, the size of the map is updated every time the window is resized.
`resize`	logical. If TRUE, the widget automatically redraw when the container size changes. Everything is recalculated every time the window is resized (default FALSE)
`domain`	spatial dataframe. Optional. Geographic domain to display.
`responsive`	logical. Optional. Whether the SVG map resizes with the container Nothing is recalculated. It's just the image that is enlarged or reduced. (default TRUE).

projection	character. Optional. D3 Map projection (e.g. "mercator", "equalearth", "Polar", "Spilhaus",...).
background	character. Optional. Background color.
fontFamily	character. Optional. Font family applied to the entire map.
margin	numeric or vector. Optional. Map margins (default 0). Can be a single value or c(top, right, bottom, left).
zoomable	logical or numeric or character. Optional. Enables zoom interaction. Can define zoom extent or use "versor" for spherical zoom.
control	logical or numeric vector. Optional. Adds zoom control panel and optionally defines its position by using a vector of 2 values.
warning	logical. Optional. Whether to display warnings on the map (default TRUE).

Value

A list containing map parameters and an initially empty list of layers to be displayed.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", zoomable = TRUE) |>
  viz_path(data = world, fill = "#38896F") |>
  viz_render()
```

---

viz_dotdensity *Dot density layer*

Description

The viz_dotdensity function creates a dot density map by distributing points proportionally to a numeric variable. Each dot represents a fixed quantity of the underlying data.

Usage

```
viz_dotdensity(
  map,
  data = NULL,
  var,
  stroke = "none",
  r = 1,
  dotval = NULL,
  fill = "black",
```

```
  legend = TRUE,
  leg_pos = NULL,
  leg_text = NULL,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration over features.
var	character. Name of the numeric variable used to generate dot density values.
stroke	character. Optional. Stroke color for dots (default "none").
r	numeric. Optional. Radius of each dot (default 1).
dotval	numeric. Optional. Value represented by a single dot. If not provided, it is computed automatically.
fill	character. Optional. Fill color of dots (default "black").
legend	logical. Optional. Whether to display legend (default TRUE).
leg_pos	numeric vector. Optional. Legend position (default c(10, svg.height - 10)).
leg_text	character. Optional. Legend text (default dot value).
...	Additional parameters passed to rendering or SVG container options.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
cities <- st_read(
  system.file("gpkg/cities.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_dotdensity(data = cities, var = "population") |>
  viz_render()
```

viz_earth *Earth raster layer*

Description

The viz_earth function displays PNG images representing the Earth's surface (Natural Earth dataset or custom source). The image is projected on-the-fly. The images are hosted on GitHub. You will need an internet connection. This function is only relevant at the world scale.

Usage

```
viz_earth(
  map,
  id = NULL,
  url = "NE2_50M_SR_W",
  resolution = 1,
  tileSize = 1024,
  opacity = 1,
  dx = 0,
  dy = 0,
  clipPath = NULL,
  max_canvas_size = NULL
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
url	character. Optional. Image source. Can be one of ("GRAY_50M_SR","GRAY_50M_SR_OB","GRAY_5 "NE1_50M_SR_W","NE2_50M_SR","NE2_50M_SR_W","OB_50M","PRIMSA_SR_50M","SR_50M' or a custom URL. Default is "NE2_50M_SR_W".
resolution	numeric. Optional. Resolution factor (default 1). Increase for sharper rendering (e.g. 2 for Retina, 3 for high-quality export).
tileSize	numeric. Optional. Tile size (default 1024).
opacity	numeric. Optional. Layer opacity (default 1).
dx	numeric. Optional. Horizontal shift (default 0).
dy	numeric. Optional. Vertical shift (default 0).
clipPath	a spatial dataframe used to clip the image (default uses map outline).
max_canvas_size	numeric. Optional. Maximum raster size (in pixels) before tiling (e.g. 2048).

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using viz_render().

Examples

```
viz_create(projection = "EqualEarth", background = "white") |>
  viz_earth(url = "NE2_50M_SR_W", resolution = 2) |>
  viz_render()
```

`viz_footer` *Source of the map*

Description

The `viz_footer` function adds a text below a geoviz map.

Usage

```
viz_footer(
  map,
  id = NULL,
  text = "Author, source...",
  fill = "#9e9696",
  background_fill = "white",
  background_stroke = "white",
  background_strokeWidth = 1,
  dominantBaseline = "central",
  textAnchor = "middle",
  lineSpacing = 0,
  margin = 1,
  fontSize = 10,
  fontFamily = NULL,
  dx = 0,
  dy = 0,
  ...
)
```

Arguments

`map`	A `geovizr` map object created using `viz_create()`.
`id`	character. Optional. Unique layer id.
`text`	character. Optional. Footer text to display (default "Author, source...").
`fill`	character. Optional. Text color (default "#9e9696").
`background_fill`	character. Optional. Background fill color (default "white").
`background_stroke`	character. Optional. Background stroke color (default "white").
`background_strokeWidth`	numeric. Optional. Background stroke width (default 1).

dominantBaseline	character. Optional. Vertical text alignment (default "central", values "hanging","middle","central","bottom").
textAnchor	character. Optional. Horizontal text alignment (default "middle", values "start","middle","end").
lineSpacing	numeric. Optional. Line spacing (default 0).
margin	numeric. Optional. Margin (default 1).
fontSize	numeric. Optional. Font size (default 10).
fontFamily	character. Optional. Font family (default container font).
dx	numeric. Optional. X shift (default 0).
dy	numeric. Optional. Y shift (default 0).
...	Additional SVG attributes applied to text and background elements.

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_footer(text = "Source, author, note, etc.", fill = "#38896F") |>
  viz_render()
```

---

viz_graticule *Graticule layer*

Description

The `viz_graticule` function draws a graticule (latitude and longitude lines) on the map. The spacing between lines can be uniform or specified separately for parallels and meridians.

Usage

```
viz_graticule(
  map,
  id = NULL,
  step = 10,
  stroke = "#9ad5e6",
  fill = "none",
  strokeWidth = 0.8,
```

```
  strokeLinecap = "square",
  strokeLinejoin = "round",
  strokeDasharray = 2,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
step	numeric or vector. Optional. Gap between graticules (default 10). Can be a single value or a vector of two values (e.g. c(10, 20)).
stroke	character. Optional. Stroke color (default "#9ad5e6").
fill	character. Optional. Fill color (default "none").
strokeWidth	numeric. Optional. Stroke width (default 0.8).
strokeLinecap	character. Optional. Stroke line cap (default "square").
strokeLinejoin	character. Optional. Stroke line join (default "round").
strokeDasharray	numeric or vector. Optional. Stroke dash pattern (default 2).
...	Additional SVG attributes (e.g. opacity, etc.).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_graticule(step = c(10, 20), stroke = "#38896F") |>
  viz_render()
```

---

viz_gridchoro *Grid-based choropleth layer*

Description

The gridchoro function builds a choropleth map based on a spatial grid aggregation. It first generates a grid from the input spatial data frame, aggregates values within each cell, and maps them using a color classification.

Usage

```
viz_gridchoro(
  map,
  data = NULL,
  var,
  grid = "square",
  step = 15,
  level = 1,
  ratio_factor = 1,
  coords = "geo",
  missing = "white",
  fixmax = NULL,
  legend = TRUE,
  leg_type = "vertical",
  leg_pos = c(10, 10),
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration over features.
var	character or character vector. Variable(s) used for aggregation. If two variables are provided, a ratio is computed.
grid	character. Optional. Type of grid used for aggregation (default "square"). One of: "square", "dot", "diamond", "hexbin", "hex", "triangle", "arbitrary", "random", "h3", "h3geo", "hexgeo", "hexbingeo", "square_sph".
step	numeric. Optional. Grid resolution (default 15).
level	numeric. Optional. Subdivision level for hierarchical grids (default 1).
ratio_factor	numeric. Optional. Multiplication factor applied to ratio values (default 1).
coords	character. Optional. Coordinate system (default "geo"). One of: "geo", "svg".
missing	character. Optional. Fill color for missing values (default "white").
fixmax	numeric. Optional. Maximum value for color scaling. Useful for map comparability.
legend	logical. Optional. Whether to display legend (default TRUE).
leg_type	character. Optional. Legend type (default "vertical"). One of: "horizontal", "vertical".
leg_pos	numeric vector. Optional. Legend position (default c(10, 10)).
...	Additional parameters passed to choropleth rendering (same as viz_choro). With the leg_ prefix, you can configure the legend. For example: leg_title, leg_subtitle, leg_note, etc.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
cities <- st_read(
  system.file("gpkg/cities.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_gridchoro(data = cities, var = "population") |>
  viz_render()
```

---

viz_gridprop *Grid-based proportional symbols layer*

Description

The viz_gridprop function builds a proportional symbol map aggregated on a spatial grid. It first generates a grid from the input spatial data frame, then displays aggregated values within each cell using proportional symbols.

Usage

```
viz_gridprop(
  map,
  data = NULL,
  var,
  grid = "square",
  step = 15,
  level = 1,
  coords = "geo",
  missing = "white",
  symbol = "circle",
  k = 10,
  width = 30,
  straight = 0,
  fixmax = NULL,
  legend = TRUE,
  leg_type = "separate",
  leg_pos = c(10, 10),
  leg_title = NULL,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration over features.
var	character. Name of the numeric variable used for aggregation and symbol scaling.
grid	character. Optional. Type of grid used for aggregation (default "square"). One of: "square", "dot", "diamond", "hexbin", "hex", "triangle", "arbitrary", "random", "h3", "h3geo", "hexgeo", "hexbingeo", "square_sph".
step	numeric. Optional. Grid resolution (default 15).
level	numeric. Optional. Subdivision level for hierarchical grids (default 1).
coords	character. Optional. Coordinate system (default "geo"). One of: "geo", "svg".
missing	character. Optional. Color for missing values (default "white").
symbol	character. Optional. Symbol type (default "circle"). One of: "circle", "square", "spike", "halfcircle".
k	numeric. Optional. Size of the largest symbol (default 10).
width	numeric. Optional. Width of spike symbols (default 30).
straight	numeric. Optional. Curvature of spike symbols (default 0). Value between 0 (curved) and 1 (straight).
fixmax	numeric. Optional. Maximum value for symbol scaling. Useful for map comparability.
legend	logical. Optional. Whether to display legend (default TRUE).
leg_type	character. Optional. Legend type (default "separate"). One of: "nested", "separate".
leg_pos	numeric vector. Optional. Legend position (default c(10, 10)).
leg_title	character. Optional. Legend title (default var).
...	Additional parameters passed to choropleth rendering (same as viz_choro). With the leg_ prefix, you can configure the legend. For example: leg_title, leg_subtitle, leg_note, etc.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
cities <- st_read(
  system.file("gpkg/cities.gpkg", package = "geovizr"),
  quiet = TRUE
)
```

```
viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_gridprop(data = cities, var = "population") |>
  viz_render()
```

`viz_halfcircle` *Half-circle layer*

Description

The `viz_halfcircle` function draws rotatable half-circles on the map from a spatial data frame or from a single position. It can be used to represent values with semi-circular proportional symbols.

Usage

```
viz_halfcircle(
  map,
  data = NULL,
  id = NULL,
  pos = c(0, 0),
  dx = 0,
  dy = 0,
  angle = 0,
  r = 10,
  innerRadius = 0,
  cornerRadius = 2,
  k = 50,
  fixmax = NULL,
  sort = NULL,
  descending = NULL,
  coords = "geo",
  fill = NULL,
  stroke = NULL,
  tip = FALSE,
  ...
)
```

Arguments

`map`	A `geovizr` map object created using `viz_create()`.
`data`	object. Optional. A spatial data frame.
`id`	character. Optional. Unique layer id.
`pos`	numeric vector. Optional. Position of a single half-circle (default c(0, 0)).
`dx`	numeric. Optional. Horizontal shift (default 0).
`dy`	numeric. Optional. Vertical shift (default 0).

angle	numeric. Optional. Rotation angle in degrees (default 0).
r	numeric or character. Optional. Outer radius (default 10). Can be a fixed value or the name of a field containing numerical values.
innerRadius	numeric. Optional. Inner radius (default 0).
cornerRadius	numeric. Optional. Corner radius (default 2).
k	numeric. Optional. Radius of the largest half-circle (default 50).
fixmax	numeric. Optional. Value corresponding to the half-circle of radius k. Useful to ensure comparability between maps.
sort	character or function. Optional. Field name or function to sort half-circles.
descending	logical. Optional. Sorting order.
coords	character. Optional. Coordinate system (default "geo"). Use "svg" if coordinates are already expressed in the SVG coordinate space.
fill	character or function. Optional. Fill color.
stroke	character or function. Optional. Stroke color.
tip	logical or function. Optional. Tooltip definition (default FALSE). Use TRUE to display all fields.
...	Additional SVG attributes (e.g. strokeDasharray, strokeWidth, opacity, strokeLinecap, etc.).

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_halfcircle(data = world, r = 30, fill = "#38896F") |>
  viz_render()
```

---

viz_header *Map title*

Description

The viz_header function adds a title above a geoviz map.

Usage

```
viz_header(
  map,
  id = NULL,
  text = "Map title",
  fill = "#9e9696",
  background_fill = "white",
  background_stroke = "white",
  background_strokeWidth = 1,
  dominantBaseline = "central",
  textAnchor = "middle",
  lineSpacing = 0,
  margin = 8,
  fontSize = 26,
  fontFamily = NULL,
  dx = 0,
  dy = 0,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
text	character. Optional. Title text to display (default "Map title").
fill	character. Optional. Text color (default "#9e9696").
background_fill	character. Optional. Background fill color (default "white").
background_stroke	character. Optional. Background stroke color (default "white").
background_strokeWidth	numeric. Optional. Background stroke width (default 1).
dominantBaseline	character. Optional. Vertical text alignment. One of "hanging", "middle", "central", "bottom" (default "central").
textAnchor	character. Optional. Horizontal text alignment. One of "start", "middle", "end" (default "middle").
lineSpacing	numeric. Optional. Space between lines (default 0).
margin	numeric. Optional. Margin around header (default 8).
fontSize	numeric. Optional. Font size (default 26).
fontFamily	character. Optional. Font family (default container font).
dx	numeric. Optional. X offset (default 0).
dy	numeric. Optional. Y offset (default 0).
...	Additional SVG attributes applied to the text or background elements.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_header(text = "Hello World", fill = "#38896F") |>
  viz_render()
```

---

viz_leg_box *Add a box legend*

Description

The `viz_leg_box` function creates a box legend. The function adds a legend layer to the map.

Usage

```
viz_leg_box(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  rect_width = 25,
  rect_height = 17,
  rect_fill = "#5d6266",
  rect_stroke = "#303030",
  rect_strokeWidth = 0.1,
  label = "",
  label_fill = "#363636",
  label_fontSize = 10,
  label_dx = 5,
  label_dominantBaseline = "central",
  title = "Legend",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
```

```
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. ID of the layer.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
rect_width	numeric. Optional. Width of the box (default 25).
rect_height	numeric. Optional. Height of the box (default 17).
rect_fill	character. Optional. Box fill color (default "#5d6266").
rect_stroke	character. Optional. Stroke color (default "#303030").
rect_strokeWidth	numeric. Optional. Stroke width (default 0.1).
label	character. Optional. Text displayed (default "").
label_fill	character. Optional. Text color (default "#363636").
label_fontSize	numeric. Optional. Text size (default 10).
label_dx	numeric. Optional. Horizontal shift (default 5).
label_dominantBaseline	character. Optional. Dominant baseline (default "central").
title	character. Optional. Legend title (default "Legend").
title_fill	character. Optional. Title color (default "#363636").
title_fontSize	numeric. Optional. Title font size (default 16).
subtitle	character. Optional. Subtitle of the legend (default "").
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle font size (default 12).
note	character. Optional. Note displayed above the legend (default "").
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note font size (default 10).
frame	logical. Optional. Draw a frame around the legend (default FALSE).
frame_margin	numeric. Optional. Frame margin (default 15).
frame_fill	character. Optional. Frame fill color (default "white").

frame_stroke	character. Optional. Frame stroke color (default "black").
frame_fillOpacity	numeric. Optional. Frame fill opacity (default 0.5).
...	Additional SVG attributes passed to elements (e.g. rect_*, label_*, title_*, subtitle_*, note_*, frame_*, text_*).

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_box(pos = c(20, 20), label = "Hello") |>
  viz_render()
```

---

viz_leg_choro_horizontal

Add a horizontal choropleth legend

Description

The `viz_leg_choro_horizontal` function creates a horizontal legend for choropleth layers. The function adds a legend layer to the map.

Usage

```
viz_leg_choro_horizontal(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  breaks = c(1, 2, 3, 4, 5),
  colors = c("#fee5d9", "#fcae91", "#fb6a4a", "#cb181d"),
  rect_width = 50,
  rect_height = 14,
  rect_spacing = 0,
  rect_fill = "#5d6266",
  rect_stroke = "#303030",
  rect_strokeWidth = 0.1,
  values_textAnchor = "middle",
```

```
  values_dx = 0,
  values_dy = 5,
  values_fill = "#363636",
  values_fontSize = 10,
  values_factor = 1,
  values_decimal = ".",
  values_thousands = " ",
  title = "Legend",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. ID of the layer.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
breaks	numeric vector. Optional. Break values (default c(1, 2, 3, 4, 5)).
colors	character vector. Optional. Colors (default c("#fee5d9", "#fcae91", "#fb6a4a", "#cb181d")).
rect_width	numeric. Optional. Width of the boxes (default 50).
rect_height	numeric. Optional. Height of the boxes (default 14).
rect_spacing	numeric. Optional. Spacing between boxes (default 0).
rect_fill	character. Optional. Box fill color (default "#5d6266").
rect_stroke	character. Optional. Stroke color (default "#303030").
rect_strokeWidth	numeric. Optional. Stroke width (default 0.1).
values_textAnchor	character. Optional. Text anchor (default "middle").
values_dx	numeric. Optional. Horizontal shift (default 0).
values_dy	numeric. Optional. Vertical shift (default 5).

values_fill	character. Optional. Text color (default "#363636").
values_fontSize	numeric. Optional. Font size (default 10).
values_factor	numeric. Optional. Multiplication factor for displayed values (default 1).
values_decimal	character. Optional. Decimal separator (default ".").
values_thousands	character. Optional. Thousands separator (default " ").
title	character. Optional. Legend title (default "Legend").
title_fill	character. Optional. Title color (default "#363636").
title_fontSize	numeric. Optional. Title font size (default 16).
subtitle	character. Optional. Subtitle of the legend (default "").
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle font size (default 12).
note	character. Optional. Note displayed above the legend (default "").
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note font size (default 10).
frame	logical. Optional. Draw a frame around the legend (default FALSE).
frame_margin	numeric. Optional. Frame margin (default 15).
frame_fill	character. Optional. Frame fill color (default "white").
frame_stroke	character. Optional. Frame stroke color (default "black").
frame_fillOpacity	numeric. Optional. Frame fill opacity (default 0.5).
...	Additional SVG attributes passed to elements (e.g. rect_*, title_*, subtitle_*, note_*, frame_*, text_*).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_choro_horizontal(pos = c(20, 20)) |>
  viz_render()
```

viz_leg_choro_vertical *Add a vertical choropleth legend*

Description

The viz_leg_choro_vertical function creates a vertical legend for choropleth layers. The function adds a legend layer to the map.

Usage

```
viz_leg_choro_vertical(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  breaks = c(1, 2, 3, 4, 5),
  colors = c("#fee5d9", "#fcae91", "#fb6a4a", "#cb181d"),
  rect_width = 25,
  rect_height = 17,
  rect_spacing = 0,
  rect_fill = "#5d6266",
  rect_stroke = "#303030",
  rect_strokeWidth = 0.1,
  values_textAnchor = "start",
  values_dx = 5,
  values_dy = 0,
  values_fill = "#363636",
  values_fontSize = 10,
  values_factor = 1,
  values_decimal = ".",
  values_thousands = " ",
  title = "Legend",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
```

```
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. ID of the layer.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
breaks	numeric vector. Optional. Break values (default c(1, 2, 3, 4, 5)).
colors	character vector. Optional. Colors (default c("#fee5d9", "#fcae91", "#fb6a4a", "#cb181d")).
rect_width	numeric. Optional. Width of the boxes (default 25).
rect_height	numeric. Optional. Height of the boxes (default 17).
rect_spacing	numeric. Optional. Spacing between boxes (default 0).
rect_fill	character. Optional. Box fill color (default "#5d6266").
rect_stroke	character. Optional. Stroke color (default "#303030").
rect_strokeWidth	numeric. Optional. Stroke width (default 0.1).
values_textAnchor	character. Optional. Text anchor (default "start").
values_dx	numeric. Optional. Horizontal shift (default 5).
values_dy	numeric. Optional. Vertical shift (default 0).
values_fill	character. Optional. Text color (default "#363636").
values_fontSize	numeric. Optional. Font size (default 10).
values_factor	numeric. Optional. Multiplication factor for displayed values (default 1).
values_decimal	character. Optional. Decimal separator (default ".").
values_thousands	character. Optional. Thousands separator (default " ").
title	character. Optional. Legend title (default "Legend").
title_fill	character. Optional. Title color (default "#363636").
title_fontSize	numeric. Optional. Title font size (default 16).
subtitle	character. Optional. Subtitle of the legend (default "").
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle font size (default 12).
note	character. Optional. Note displayed above the legend (default "").
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note font size (default 10).
frame	logical. Optional. Draw a frame around the legend (default FALSE).

frame_margin	numeric. Optional. Frame margin (default 15).
frame_fill	character. Optional. Frame fill color (default "white").
frame_stroke	character. Optional. Frame stroke color (default "black").
frame_fillOpacity	numeric. Optional. Frame fill opacity (default 0.5).
...	Additional SVG attributes passed to elements (e.g. rect_*, values_*, title_*, subtitle_*, note_*, frame_*, text_*).

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_choro_vertical(pos = c(20, 20)) |>
  viz_render()
```

---

viz_leg_circles *Add a proportional circles legend*

Description

The `viz_leg_circles` function adds a legend for proportional circles. It draws circles scaled by values, with optional reference lines and labels. The function adds a legend layer to the SVG container.

Usage

```
viz_leg_circles(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  data = c(1, 1000),
  k = 50,
  fixmax = NULL,
  nb = 4,
  circle_fill = "none",
  circle_stroke = "#363636",
```

```
  circle_spacing = 5,
  line_stroke = "#363636",
  line_strokeDasharray = "1",
  line_strokeWidth = 0.7,
  line_length = 10,
  values_textAnchor = "start",
  values_dx = 5,
  values_dy = 0,
  values_fill = "#363636",
  values_fontSize = 10,
  values_factor = 1,
  values_decimal = ".",
  values_thousands = " ",
  title = "",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
data	numeric vector. Required. Input values.
k	numeric. Optional. Radius of largest circle (default 50).
fixmax	numeric. Optional. Value mapped to radius k.
nb	numeric. Optional. Number of circles (default 4).
circle_fill	character. Optional. Circle fill (default "none").
circle_stroke	character. Optional. Circle stroke (default "#363636").
circle_spacing	numeric. Optional. Spacing between circles (default 5).
line_stroke	character. Optional. Line stroke color (default "#363636").
line_strokeDasharray	character. Optional. Line dash style (default "1").

line_strokeWidth
: numeric. Optional. Line width (default 0.7).

line_length
: numeric. Optional. Line length (default 10).

values_textAnchor
: character. Optional. Text anchor (default "start").

values_dx
: numeric. Optional. X shift (default 5).

values_dy
: numeric. Optional. Y shift (default 0).

values_fill
: character. Optional. Label color (default "#363636").

values_fontSize
: numeric. Optional. Font size (default 10).

values_factor
: numeric. Optional. Scaling factor (default 1).

values_decimal
: character. Optional. Decimal separator (default ".").

values_thousands
: character. Optional. Thousands separator (default " ").

title
: character. Optional. Title (default "").

title_fill
: character. Optional. Title color (default "#363636").

title_fontSize
: numeric. Optional. Title size (default 16).

subtitle
: character. Optional. Subtitle (default "").

subtitle_fill
: character. Optional. Subtitle color (default "#363636").

subtitle_fontSize
: numeric. Optional. Subtitle size (default 12).

note
: character. Optional. Note (default "").

note_fill
: character. Optional. Note color (default "#363636").

note_fontSize
: numeric. Optional. Note size (default 10).

frame
: logical. Optional. Draw frame (default FALSE).

frame_margin
: numeric. Optional. Frame margin (default 15).

frame_fill
: character. Optional. Frame fill (default "white").

frame_stroke
: character. Optional. Frame stroke (default "black").

frame_fillOpacity
: numeric. Optional. Frame opacity (default 0.5).

...
: Additional SVG attributes.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
```

```
viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_circles(pos = c(20, 20)) |>
  viz_render()
```

`viz_leg_circles_nested` *Add a nested proportional circles legend*

Description

The `viz_leg_circles_nested` function adds a legend for nested proportional circles. It displays circles with hierarchical/nested visual structure for comparative values. The function adds a legend layer to the SVG container.

Usage

```
viz_leg_circles_nested(
  map,
  id = NULL,
  pos = c(0, 5),
  gap = 2,
  data = c(1, 1000),
  k = 50,
  fixmax = NULL,
  nb = 4,
  circle_fill = "none",
  circle_stroke = "#363636",
  line_stroke = "#363636",
  line_strokeDasharray = "1",
  line_strokeWidth = 0.7,
  line_length = 10,
  values_textAnchor = "start",
  values_dx = 5,
  values_dy = 0,
  values_fill = "#363636",
  values_fontSize = 10,
  values_factor = 1,
  values_decimal = ".",
  values_thousands = " ",
  title = "",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
```

```
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 5)).
gap	numeric. Optional. Gap between elements (default 2).
data	numeric vector. Required. Input values.
k	numeric. Optional. Radius of largest circle (default 50).
fixmax	numeric. Optional. Value mapped to radius k.
nb	numeric. Optional. Number of circles (default 4).
circle_fill	character. Optional. Fill color (default "none").
circle_stroke	character. Optional. Stroke color (default "#363636").
line_stroke	character. Optional. Line stroke color (default "#363636").
line_strokeDasharray	character. Optional. Dash style (default "1").
line_strokeWidth	numeric. Optional. Line width (default 0.7).
line_length	numeric. Optional. Line length (default 10).
values_textAnchor	character. Optional. Text anchor (default "start").
values_dx	numeric. Optional. X shift (default 5).
values_dy	numeric. Optional. Y shift (default 0).
values_fill	character. Optional. Label color (default "#363636").
values_fontSize	numeric. Optional. Font size (default 10).
values_factor	numeric. Optional. Scaling factor (default 1).
values_decimal	character. Optional. Decimal separator (default ".").
values_thousands	character. Optional. Thousands separator (default " ").
title	character. Optional. Title (default "").
title_fill	character. Optional. Title color (default "#363636").

title_fontSize	numeric. Optional. Title size (default 16).
subtitle	character. Optional. Subtitle (default "").
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle size (default 12).
note	character. Optional. Note (default "").
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note size (default 10).
frame	logical. Optional. Draw frame (default FALSE).
frame_margin	numeric. Optional. Frame margin (default 15).
frame_fill	character. Optional. Frame fill (default "white").
frame_stroke	character. Optional. Frame stroke (default "black").
frame_fillOpacity	numeric. Optional. Frame opacity (default 0.5).
...	Additional SVG attributes.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_circles_nested(pos = c(20, 20)) |>
  viz_render()
```

---

viz_leg_gradient_vertical

Add a vertical gradient legend

Description

The `viz_leg_gradient_vertical` function creates a vertical gradient legend. It draws a series of colored rectangles with three labels aligned at top, middle, and bottom. The function adds a legend layer to the map.

Usage

```
viz_leg_gradient_vertical(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 5,
  colors = c("#fee5d9", "#fcae91", "#fb6a4a", "#cb181d"),
  rect_width = 8,
  rect_height = 25,
  rect_spacing = 0,
  rect_stroke = "white",
  values_fontSize = 12,
  values_fill = "black",
  text_high = "High",
  text_intermediate = "Intermediate",
  text_low = "Low",
  reverse = FALSE,
  frame = FALSE,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. ID of the layer.
pos	numeric vector of length 2. Optional. Position of the legend (default c(0, 0)).
gap	numeric. Optional. Gap between title/subtitle and rectangles (default 5).
colors	character vector. Optional. Colors of the gradient (default c("#fee5d9", "#fcae91", "#fb6a4a", "#cb181d")).
rect_width	numeric. Optional. Rectangle width (default 8).
rect_height	numeric. Optional. Rectangle height (default 25).
rect_spacing	numeric. Optional. Spacing between rectangles (default 0).
rect_stroke	character. Optional. Stroke color of rectangles (default "white").
values_fontSize	numeric. Optional. Font size of labels (default 12).
values_fill	character. Optional. Label color (default "black").
text_high	character. Optional. Label at top (default "High").
text_intermediate	character. Optional. Label at middle (default "Intermediate").
text_low	character. Optional. Label at bottom (default "Low").
reverse	logical. Optional. Reverse the order of colors (default FALSE).
frame	logical. Optional. Draw a frame around the legend (default FALSE).
...	Additional SVG attributes passed to elements.

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_gradient_vertical(pos = c(20, 20)) |>
  viz_render()
```

---

viz_leg_mushrooms *Add a proportional half-circles (mushrooms) legend*

Description

The `viz_leg_mushrooms` function adds a legend for proportional half-circles. It displays two sets of nested half-circles (top and bottom), each scaled according to input values. The function adds a legend layer to the SVG container and returns the layer identifier.

Usage

```
viz_leg_mushrooms(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  line_stroke = "#363636",
  line_strokeDasharray = "1",
  line_strokeWidth = 0.7,
  line_length = 10,
  top_data = NULL,
  top_k = 50,
  top_fixmax = NULL,
  top_nb = 4,
  top_circle_fill = "none",
  top_circle_stroke = "black",
  top_circle_cornerRadius = 5,
  top_values_textAnchor = "start",
  top_values_dx = 5,
  top_values_dy = 0,
  top_values_factor = 1,
```

```
  top_values_decimal = ".",
  top_values_thousands = " ",
  top_title = "top_title",
  bottom_data = NULL,
  bottom_k = 50,
  bottom_fixmax = NULL,
  bottom_nb = 4,
  bottom_circle_fill = "none",
  bottom_circle_stroke = "black",
  bottom_circle_cornerRadius = 5,
  bottom_values_textAnchor = "start",
  bottom_values_dx = 5,
  bottom_values_dy = 0,
  bottom_values_factor = 1,
  bottom_values_decimal = ".",
  bottom_values_thousands = " ",
  bottom_title = "bottom_title",
  title = "Legend",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
line_stroke	character. Optional. Line stroke color (default "#363636").
line_strokeDasharray	character. Optional. Dash pattern (default "1").
line_strokeWidth	numeric. Optional. Line width (default 0.7).
line_length	numeric. Optional. Line length (default 10).
top_data	numeric vector. Optional. Input values for top half-circles.

top_k numeric. Optional. Radius of largest top half-circle (default 50).

top_fixmax numeric. Optional. Value mapped to radius k.

top_nb numeric. Optional. Number of top half-circles (default 4).

top_circle_fill
character. Optional. Fill color (default "none").

top_circle_stroke
character. Optional. Stroke color (default "black").

top_circle_cornerRadius
numeric. Optional. Corner radius (default 5).

top_values_textAnchor
character. Optional. Text anchor (default "start").

top_values_dx numeric. Optional. X shift (default 5).

top_values_dy numeric. Optional. Y shift (default 0).

top_values_factor
numeric. Optional. Scaling factor (default 1).

top_values_decimal
character. Optional. Decimal separator (default ".").

top_values_thousands
character. Optional. Thousands separator (default " ").

top_title character. Optional. Title of top legend part (default "top_title").

bottom_data numeric vector. Optional. Input values for bottom half-circles.

bottom_k numeric. Optional. Radius of largest bottom half-circle (default 50).

bottom_fixmax numeric. Optional. Value mapped to radius k.

bottom_nb numeric. Optional. Number of bottom half-circles (default 4).

bottom_circle_fill
character. Optional. Fill color (default "none").

bottom_circle_stroke
character. Optional. Stroke color (default "black").

bottom_circle_cornerRadius
numeric. Optional. Corner radius (default 5).

bottom_values_textAnchor
character. Optional. Text anchor (default "start").

bottom_values_dx
numeric. Optional. X shift (default 5).

bottom_values_dy
numeric. Optional. Y shift (default 0).

bottom_values_factor
numeric. Optional. Scaling factor (default 1).

bottom_values_decimal
character. Optional. Decimal separator (default ".").

bottom_values_thousands
character. Optional. Thousands separator (default " ").

bottom_title	character. Optional. Title of bottom legend part (default "bottom_title").
title	character. Optional. Legend title (default "Legend").
title_fill	character. Optional. Title color (default "#363636").
title_fontSize	numeric. Optional. Title size (default 16).
subtitle	character. Optional. Subtitle (default "").
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle size (default 12).
note	character. Optional. Note (default "").
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note size (default 10).
frame	logical. Optional. Draw frame (default FALSE).
frame_margin	numeric. Optional. Frame margin (default 15).
frame_fill	character. Optional. Frame fill (default "white").
frame_stroke	character. Optional. Frame stroke (default "black").
frame_fillOpacity	numeric. Optional. Frame opacity (default 0.5).
...	Additional SVG attributes.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_mushrooms(pos = c(20, 20)) |>
  viz_render()
```

viz_leg_spikes *Add a spike legend*

Description

The viz_leg_spikes function adds a legend for spike marks. It draws a series of spikes with heights proportional to values. The function adds a legend layer to the SVG container.

Usage

```
viz_leg_spikes(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  data = c(1, 1000),
  k = 50,
  fixmax = NULL,
  nb = 4,
  spike_width = 30,
  spike_straight = 0,
  spike_spacing = 3,
  spike_fill = "none",
  spike_stroke = "black",
  values_textAnchor = "start",
  values_dx = 5,
  values_dy = 0,
  values_fill = "#363636",
  values_fontSize = 10,
  values_factor = 1,
  values_decimal = ".",
  values_thousands = " ",
  title = "",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
```

```
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. ID of the layer.
pos	numeric vector of length 2. Optional. Position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
data	numeric vector. Optional. Input values (default c(1, 200, 500, 1000)).
k	numeric. Optional. Height of the highest spike (default 50).
fixmax	numeric. Optional. Value mapped to height k.
nb	numeric. Optional. Number of spikes (default 4).
spike_width	numeric. Optional. Width of spikes (default 30).
spike_straight	numeric. Optional. Shape of spikes (0 = curved, 1 = straight, default 0).
spike_spacing	numeric. Optional. Spacing between spikes (default 3).
spike_fill	character. Optional. Fill color (default "none").
spike_stroke	character. Optional. Stroke color (default "black").
values_textAnchor	character. Optional. Text anchor (default "start").
values_dx	numeric. Optional. Shift in x (default 0).
values_dy	numeric. Optional. Shift in y (default 5).
values_fill	character. Optional. Label color (default "#363636").
values_fontSize	numeric. Optional. Font size (default 10).
values_factor	numeric. Optional. Multiplicative factor (default 1).
values_decimal	character. Optional. Decimal separator (default ".").
values_thousands	character. Optional. Thousands separator (default " ").
title	character. Optional. Title (default "").
title_fill	character. Optional. Title color (default "#363636").
title_fontSize	numeric. Optional. Title size (default 16).
subtitle	character. Optional. Subtitle (default "").
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle size (default 12).
note	character. Optional. Note (default "").
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note size (default 10).
frame	logical. Optional. Draw frame (default FALSE).
frame_margin	numeric. Optional. Frame margin (default 15).

frame_fill	character. Optional. Frame fill (default "white").
frame_stroke	character. Optional. Frame stroke (default "black").
frame_fillOpacity	numeric. Optional. Frame opacity (default 0.5).
...	Additional SVG attributes.

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_spikes(pos = c(20, 20)) |>
  viz_render()
```

---

viz_leg_squares *Add a proportional squares legend*

Description

The `viz_leg_squares` function adds a legend for proportional squares. It displays squares scaled according to input values, optionally comparable via fixmax. The function adds a legend layer to the SVG container.

Usage

```
viz_leg_squares(
  map,
  id = NULL,
  pos = c(0, 0),
  data = c(1, 1000),
  gap = 2,
  k = 50,
  fixmax = NULL,
  nb = 4,
  square_fill = "none",
  square_stroke = "#363636",
  square_spacing = 5,
  line_stroke = "#363636",
```

```
  line_strokeDasharray = "1",
  line_strokeWidth = 0.7,
  line_length = 10,
  values_textAnchor = "start",
  values_dx = 5,
  values_dy = 0,
  values_fill = "#363636",
  values_fontSize = 10,
  values_factor = 1,
  values_decimal = ".",
  values_thousands = " ",
  title = "",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
data	numeric vector. Required. Input values.
gap	numeric. Optional. Gap between elements (default 2).
k	numeric. Optional. Side length of largest square (default 50).
fixmax	numeric. Optional. Value mapped to side k.
nb	numeric. Optional. Number of squares (default 4).
square_fill	character. Optional. Fill color (default "none").
square_stroke	character. Optional. Stroke color (default "#363636").
square_spacing	numeric. Optional. Spacing between squares (default 5).
line_stroke	character. Optional. Line stroke color (default "#363636").
line_strokeDasharray	character. Optional. Dash pattern (default "1").

line_strokeWidth
: numeric. Optional. Line width (default 0.7).

line_length
: numeric. Optional. Line length (default 10).

values_textAnchor
: character. Optional. Text anchor (default "start").

values_dx
: numeric. Optional. X shift (default 5).

values_dy
: numeric. Optional. Y shift (default 0).

values_fill
: character. Optional. Label color (default "#363636").

values_fontSize
: numeric. Optional. Font size (default 10).

values_factor
: numeric. Optional. Scaling factor (default 1).

values_decimal
: character. Optional. Decimal separator (default ".").

values_thousands
: character. Optional. Thousands separator (default " ").

title
: character. Optional. Title (default "").

title_fill
: character. Optional. Title color (default "#363636").

title_fontSize
: numeric. Optional. Title size (default 16).

subtitle
: character. Optional. Subtitle (default "").

subtitle_fill
: character. Optional. Subtitle color (default "#363636").

subtitle_fontSize
: numeric. Optional. Subtitle size (default 12).

note
: character. Optional. Note (default "").

note_fill
: character. Optional. Note color (default "#363636").

note_fontSize
: numeric. Optional. Note size (default 10).

frame
: logical. Optional. Draw frame (default FALSE).

frame_margin
: numeric. Optional. Frame margin (default 15).

frame_fill
: character. Optional. Frame fill (default "white").

frame_stroke
: character. Optional. Frame stroke (default "black").

frame_fillOpacity
: numeric. Optional. Frame opacity (default 0.5).

...
: Additional SVG attributes.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
```

```
viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_squares(pos = c(20, 20)) |>
  viz_render()
```

`viz_leg_squares_nested` *Add a proportional nested squares legend*

Description

The `viz_leg_squares_nested` function adds a legend for proportional nested squares. It displays squares scaled according to input values, optionally comparable via fixmax. The function adds a legend layer to the SVG container.

Usage

```
viz_leg_squares_nested(
  map,
  id = NULL,
  pos = c(0, 5),
  gap = 2,
  data = c(1, 1000),
  k = 50,
  fixmax = NULL,
  nb = 4,
  square_fill = "none",
  square_stroke = "#363636",
  square_spacing = 5,
  line_stroke = "#363636",
  line_strokeDasharray = "1",
  line_strokeWidth = 0.7,
  line_length = 10,
  values_textAnchor = "start",
  values_dx = 5,
  values_dy = 0,
  values_fill = "#363636",
  values_fontSize = 10,
  values_factor = 1,
  values_decimal = ".",
  values_thousands = " ",
  title = "",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
```

```
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 5)).
gap	numeric. Optional. Gap between elements (default 2).
data	numeric vector. Required. Input values.
k	numeric. Optional. Side length of largest square (default 50).
fixmax	numeric. Optional. Value mapped to side k.
nb	numeric. Optional. Number of squares (default 4).
square_fill	character. Optional. Fill color (default "none").
square_stroke	character. Optional. Stroke color (default "#363636").
square_spacing	numeric. Optional. Spacing between squares (default 5).
line_stroke	character. Optional. Line stroke color (default "#363636").
line_strokeDasharray	character. Optional. Dash pattern (default "1").
line_strokeWidth	numeric. Optional. Line width (default 0.7).
line_length	numeric. Optional. Line length (default 10).
values_textAnchor	character. Optional. Text anchor (default "start").
values_dx	numeric. Optional. X shift (default 5).
values_dy	numeric. Optional. Y shift (default 0).
values_fill	character. Optional. Label color (default "#363636").
values_fontSize	numeric. Optional. Font size (default 10).
values_factor	numeric. Optional. Scaling factor (default 1).
values_decimal	character. Optional. Decimal separator (default ".").
values_thousands	character. Optional. Thousands separator (default " ").

`title`	character. Optional. Title (default "").
`title_fill`	character. Optional. Title color (default "#363636").
`title_fontSize`	numeric. Optional. Title size (default 16).
`subtitle`	character. Optional. Subtitle (default "").
`subtitle_fill`	character. Optional. Subtitle color (default "#363636").
`subtitle_fontSize`	numeric. Optional. Subtitle size (default 12).
`note`	character. Optional. Note (default "").
`note_fill`	character. Optional. Note color (default "#363636").
`note_fontSize`	numeric. Optional. Note size (default 10).
`frame`	logical. Optional. Draw frame (default FALSE).
`frame_margin`	numeric. Optional. Frame margin (default 15).
`frame_fill`	character. Optional. Frame fill (default "white").
`frame_stroke`	character. Optional. Frame stroke (default "black").
`frame_fillOpacity`	numeric. Optional. Frame opacity (default 0.5).
`...`	Additional SVG attributes.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_squares_nested(pos = c(20, 20)) |>
  viz_render()
```

`viz_leg_symbol_horizontal` *Add a symbol horizontal legend*

Description

The `viz_leg_symbol_horizontal` function adds a horizontal legend for symbol layers. It displays a set of categorized symbols with optional shapes and ordering. The function creates a legend layer in the SVG container and returns the layer identifier.

Usage

```
viz_leg_symbol_horizontal(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  types = c("A", "B", "C", "D"),
  symbols = c("circle", "square", "triangle", "pentagon"),
  alphabetical = TRUE,
  symbol_size = 10,
  symbol_rotate = 0,
  symbol_spacing = 4,
  symbol_fill = "#2e2e2e",
  symbol_stroke = "#303030",
  symbol_strokeWidth = 0.5,
  symbol_background = FALSE,
  values_textAnchor = "middle",
  values_dx = 0,
  values_dy = 5,
  values_fill = "#363636",
  values_fontSize = 10,
  title = "Legend",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
types	character vector. Optional. Category labels (default c("A","B","C","D")).
symbols	character vector. Optional. Symbol shapes (default c("circle","square","triangle","pentagon")).
alphabetical	logical. Optional. Sort types alphabetically (default TRUE).

`symbol_size` numeric. Optional. Symbol size (default 10).

`symbol_rotate` numeric. Optional. Rotation angle (default 0).

`symbol_spacing` numeric. Optional. Spacing between symbols (default 4).

`symbol_fill` character. Optional. Symbol fill color (default "#2e2e2e").

`symbol_stroke` character. Optional. Symbol stroke color (default "#303030").

`symbol_strokeWidth` numeric. Optional. Symbol stroke width (default 0.5).

`symbol_background` logical. Optional. Add background circle (default FALSE).

`values_textAnchor` character. Optional. Text anchor (default "middle").

`values_dx` numeric. Optional. X shift (default 0).

`values_dy` numeric. Optional. Y shift (default 5).

`values_fill` character. Optional. Text color (default "#363636").

`values_fontSize` numeric. Optional. Font size (default 10).

`title` character. Optional. Legend title (default "Legend").

`title_fill` character. Optional. Title color (default "#363636").

`title_fontSize` numeric. Optional. Title size (default 16).

`subtitle` character. Optional. Subtitle (default "").

`subtitle_fill` character. Optional. Subtitle color (default "#363636").

`subtitle_fontSize` numeric. Optional. Subtitle size (default 12).

`note` character. Optional. Note (default "").

`note_fill` character. Optional. Note color (default "#363636").

`note_fontSize` numeric. Optional. Note size (default 10).

`frame` logical. Optional. Draw frame (default FALSE).

`frame_margin` numeric. Optional. Frame margin (default 15).

`frame_fill` character. Optional. Frame fill (default "white").

`frame_stroke` character. Optional. Frame stroke (default "black").

`frame_fillOpacity` numeric. Optional. Frame opacity (default 0.5).

`...` Additional SVG attributes.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_symbol_horizontal(pos = c(20, 20)) |>
  viz_render()
```

`viz_leg_symbol_vertical` *Add a vertical symbol legend*

Description

The `viz_leg_symbol_vertical` function adds a vertical legend for symbol layers. It displays categorized symbols arranged vertically with optional shapes and ordering. The function creates a legend layer in the SVG container and returns the layer identifier.

Usage

```
viz_leg_symbol_vertical(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  types = c("A", "B", "C", "D"),
  symbols = c("circle", "square", "triangle", "pentagon"),
  alphabetical = TRUE,
  symbol_size = 10,
  symbol_rotate = 0,
  symbol_spacing = 4,
  symbol_fill = "#2e2e2e",
  symbol_stroke = "#303030",
  symbol_strokeWidth = 0.5,
  symbol_background = FALSE,
  values_textAnchor = "middle",
  values_dx = 5,
  values_dy = 0,
  values_fill = "#363636",
  values_fontSize = 10,
  title = "Legend",
  title_fill = "#363636",
  title_fontSize = 16,
```

```
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
types	character vector. Optional. Category labels (default c("A","B","C","D")).
symbols	character vector. Optional. Symbol shapes (default c("circle","square","triangle","pentagon")).
alphabetical	logical. Optional. Sort types alphabetically (default TRUE).
symbol_size	numeric. Optional. Symbol size (default 10).
symbol_rotate	numeric. Optional. Rotation angle (default 0).
symbol_spacing	numeric. Optional. Spacing between symbols (default 4).
symbol_fill	character. Optional. Symbol fill color (default "#2e2e2e").
symbol_stroke	character. Optional. Symbol stroke color (default "#303030").
symbol_strokeWidth	numeric. Optional. Symbol stroke width (default 0.5).
symbol_background	logical. Optional. Add background circle (default FALSE).
values_textAnchor	character. Optional. Text anchor (default "middle").
values_dx	numeric. Optional. X shift (default 5).
values_dy	numeric. Optional. Y shift (default 0).
values_fill	character. Optional. Text color (default "#363636").
values_fontSize	numeric. Optional. Font size (default 10).
title	character. Optional. Legend title (default "Legend").
title_fill	character. Optional. Title color (default "#363636").
title_fontSize	numeric. Optional. Title size (default 16).

subtitle	character. Optional. Subtitle (default "").
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle size (default 12).
note	character. Optional. Note (default "").
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note size (default 10).
frame	logical. Optional. Draw frame (default FALSE).
frame_margin	numeric. Optional. Frame margin (default 15).
frame_fill	character. Optional. Frame fill (default "white").
frame_stroke	character. Optional. Frame stroke (default "black").
frame_fillOpacity	numeric. Optional. Frame opacity (default 0.5).
...	Additional SVG attributes.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)

world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_symbol_vertical(pos = c(20, 20)) |>
  viz_render()
```

---

viz_leg_typo_horizontal

Add a horizontal typology legend

Description

The `viz_leg_typo_horizontal` function creates a horizontal legend for typology layers. The function adds a legend layer to the map.

Usage

```
viz_leg_typo_horizontal(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  types = c("A", "B", "C", "D"),
  colors = c("#e41a1c", "#377eb8", "#4daf4a", "#984ea3"),
  alphabetical = TRUE,
  rect_width = 50,
  rect_height = 14,
  rect_spacing = 0,
  rect_fill = "#5d6266",
  rect_stroke = "#303030",
  rect_strokeWidth = 0.1,
  values_textAnchor = "middle",
  values_dx = 0,
  values_dy = 5,
  values_fill = "#363636",
  values_fontSize = 10,
  title = "Legend",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. ID of the layer.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
types	character vector. Optional. Types (default c("A", "B", "C", "D")).
colors	character vector. Optional. Colors (default c("#e41a1c", "#377eb8", "#4daf4a", "#984ea3")).
alphabetical	logical. Optional. Alphabetical order (default TRUE).

rect_width	numeric. Optional. Width of the boxes (default 50).
rect_height	numeric. Optional. Height of the boxes (default 14).
rect_spacing	numeric. Optional. Spacing between boxes (default 0).
rect_fill	character. Optional. Box fill color (default "#5d6266").
rect_stroke	character. Optional. Stroke color (default "#303030").
rect_strokeWidth	numeric. Optional. Stroke width (default 0.1).
values_textAnchor	character. Optional. Text anchor (default "middle").
values_dx	numeric. Optional. Horizontal shift (default 0).
values_dy	numeric. Optional. Vertical shift (default 5).
values_fill	character. Optional. Text color (default "#363636").
values_fontSize	numeric. Optional. Font size (default 10).
title	character. Optional. Legend title (default "Legend").
title_fill	character. Optional. Title color (default "#363636").
title_fontSize	numeric. Optional. Title font size (default 16).
subtitle	character. Optional. Subtitle of the legend.
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle font size (default 12).
note	character. Optional. Note displayed above the legend.
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note font size (default 10).
frame	logical. Optional. Draw a frame around the legend (default FALSE).
frame_margin	numeric. Optional. Frame margin (default 15).
frame_fill	character. Optional. Frame fill color (default "white").
frame_stroke	character. Optional. Frame stroke color (default "black").
frame_fillOpacity	numeric. Optional. Frame fill opacity (default 0.5).
...	Additional SVG attributes passed to elements (e.g. rect_*, title_*, subtitle_*, note_*, frame_*, text_*).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_typo_horizontal(pos = c(20, 20)) |>
  viz_render()
```

---

viz_leg_typo_vertical *Add a vertical typology legend*

Description

The viz_leg_typo_vertical function creates a vertical legend for typology layers. The function adds a legend layer to the map.

Usage

```
viz_leg_typo_vertical(
  map,
  id = NULL,
  pos = c(0, 0),
  gap = 2,
  types = c("A", "B", "C", "D"),
  colors = c("#e41a1c", "#377eb8", "#4daf4a", "#984ea3"),
  alphabetical = TRUE,
  rect_width = 25,
  rect_height = 17,
  rect_spacing = 3,
  rect_fill = "#5d6266",
  rect_stroke = "#303030",
  rect_strokeWidth = 0.1,
  values_textAnchor = "start",
  values_dx = 5,
  values_dy = 0,
  values_fill = "#363636",
  values_fontSize = 10,
  title = "Legend",
  title_fill = "#363636",
  title_fontSize = 16,
  subtitle = "",
  subtitle_fill = "#363636",
  subtitle_fontSize = 12,
```

```
  note = "",
  note_fill = "#363636",
  note_fontSize = 10,
  frame = FALSE,
  frame_margin = 15,
  frame_fill = "white",
  frame_stroke = "black",
  frame_fillOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. ID of the layer.
pos	numeric vector of length 2. Optional. Legend position (default c(0, 0)).
gap	numeric. Optional. Gap between elements (default 2).
types	character vector. Optional. Types (default c("A", "B", "C", "D")).
colors	character vector. Optional. Colors (default c("#e41a1c", "#377eb8", "#4daf4a", "#984ea3")).
alphabetical	logical. Optional. Alphabetical order (default TRUE).
rect_width	numeric. Optional. Width of the boxes (default 25).
rect_height	numeric. Optional. Height of the boxes (default 17).
rect_spacing	numeric. Optional. Spacing between boxes (default 3).
rect_fill	character. Optional. Box fill color (default "#5d6266").
rect_stroke	character. Optional. Stroke color (default "#303030").
rect_strokeWidth	numeric. Optional. Stroke width (default 0.1).
values_textAnchor	character. Optional. Text anchor (default "middle").
values_dx	numeric. Optional. Horizontal shift (default 0).
values_dy	numeric. Optional. Vertical shift (default 5).
values_fill	character. Optional. Text color (default "#363636").
values_fontSize	numeric. Optional. Font size (default 10).
title	character. Optional. Legend title (default "Legend").
title_fill	character. Optional. Title color (default "#363636").
title_fontSize	numeric. Optional. Title font size (default 16).
subtitle	character. Optional. Subtitle of the legend (default "").
subtitle_fill	character. Optional. Subtitle color (default "#363636").
subtitle_fontSize	numeric. Optional. Subtitle font size (default 12).

note	character. Optional. Note displayed above the legend (default "").
note_fill	character. Optional. Note color (default "#363636").
note_fontSize	numeric. Optional. Note font size (default 10).
frame	logical. Optional. Draw a frame around the legend (default FALSE).
frame_margin	numeric. Optional. Frame margin (default 15).
frame_fill	character. Optional. Frame fill color (default "white").
frame_stroke	character. Optional. Frame stroke color (default "black").
frame_fillOpacity	numeric. Optional. Frame fill opacity (default 0.5).
...	Additional SVG attributes passed to elements (e.g. rect_*, title_*, subtitle_*, note_*, frame_*, text_*).

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_leg_typo_vertical(pos = c(20, 20)) |>
  viz_render()
```

---

viz_minimap *Minimap inset layer*

Description

The `viz_minimap` function adds a minimap inset to a geoviz map. It displays a basemap and optionally highlights a location geometry (polygon or point). This is useful for showing spatial context in map layouts. NB: The map's projection must provide an invert() function.

Usage

```
viz_minimap(
  map,
  id = NULL,
  basemap_data = NULL,
  basemap_fill = "white",
  basemap_fillOpacity = 0.5,
```

```
  basemap_stroke = "none",
  width = 200,
  projection = "EqualEarth",
  precision = 10,
  pos = c(10, 10),
  location_type = "polygon",
  location_r = 5,
  location_fill = NULL,
  location_stroke = NULL,
  location_strokeWidth = 1.2,
  domain = NULL,
  margin = NULL,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id. If NULL, a random id is generated.
basemap_data	object. Optional. GeoJSON basemap. Default is land.
basemap_fill	character. Optional. Fill color of the basemap (default "white").
basemap_fillOpacity	numeric. Optional. Fill opacity of the basemap (default 0.5).
basemap_stroke	character. Optional. Stroke color of the basemap (default "none").
width	numeric. Optional. Width of the minimap (default 200).
projection	character. Optional. Projection used for the minimap (default "EqualEarth").
precision	numeric. Optional. Geometry simplification precision (default 10).
pos	numeric vector. Optional. Position of the minimap (default c(10, 10)).
location_type	character. Optional. Type of location geometry: "polygon" or "point" (default "polygon").
location_r	numeric. Optional. Radius when location_type = "point" (default 5).
location_fill	character. Optional. Fill color of the location geometry.
location_stroke	character. Optional. Stroke color of the location geometry.
location_strokeWidth	numeric. Optional. Stroke width of the location geometry (default 1.2).
domain	list. Optional. Projection domain.
margin	list. Optional. Margin configuration.
...	Additional styling options passed as prefixed arguments. Supported prefixes include: outline_* for outline styling properties, basemap_* for basemap styling properties, location_* for location geometry styling properties.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
afr <- world[world$region == "Africa", ]
viz_create(projection = "Mercator", background = "white", domain = afr) |>
  viz_path(data = world, fill = "#9e9696") |>
  viz_minimap(
    width = 200,
    projection = "EqualEarth",
    pos = c(20, 20),
    location_stroke = "red"
  ) |>
  viz_render()
```

---

viz_north *North arrow*

Description

The viz_north function adds a north arrow a geoviz map.

Usage

```
viz_north(
  map,
  id = NULL,
  pos = NULL,
  scale = 1,
  rotate = NULL,
  fill = "black",
  fillOpacity = 1,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Position [x, y] on the page (default c(svg.width - 30, 30)).
scale	numeric. Optional. Scaling factor for the arrow (default 1).
rotate	numeric. Optional. Rotation angle. If NA, it is automatically calculated.
fill	character. Optional. Fill color (default "black").
fillOpacity	numeric. Optional. Fill opacity (default 1).
...	Additional SVG attributes applied to the arrow.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "Mercator", background = "white") |>
  viz_path(datum = world[world$region == "Africa", ], fill = "#f1f3f5") |>
  viz_north(fill = "#38896F") |>
  viz_render()
```

`viz_outline` *Outline layer*

Description

The `viz_outline` function draws the Earth outline according to the current map projection. This layer can be used as a background or clipping reference.

Usage

```
viz_outline(
  map,
  id = NULL,
  stroke = "none",
  strokeWidth = 1,
  fill = "#B5DFFD",
  ...
)
```

Arguments

`map`	A `geovizr` map object created using `viz_create()`.
`id`	character. Optional. Unique layer id.
`stroke`	character. Optional. Stroke color (default "none").
`strokeWidth`	numeric. Optional. Stroke width (default 1).
`fill`	character. Optional. Fill color (default "#B5DFFD").
`...`	Additional SVG attributes (e.g. `strokeDasharray`, `opacity`, `strokeLinecap`, etc.).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_outline(fill = "#38896F") |>
  viz_render()
```

`viz_path` *Path layer*

Description

The `viz_path` function draws geometries from a spatial data frame as SVG paths. This function can be used to display polygons, lines, or points, and supports styling, simplification, and interaction.

Usage

```
viz_path(
  map,
  data = NULL,
  datum = NULL,
  id = NULL,
  coords = "geo",
  clip = TRUE,
  fill = NULL,
  stroke = NULL,
  strokeWidth = 1,
  tip = FALSE,
  simplify = FALSE,
  rewind = FALSE,
  rewindPole = FALSE,
  clipOutline = 0,
  r = 3,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration (e.g. for styling with functions).
datum	A spatial dataframe Use datum if no iteration is needed.
id	character. Optional. Unique layer id.
coords	character. Optional. Coordinate system (default "geo"). Use "svg" if coordinates are already expressed in the SVG coordinate space.
clip	logical. Optional. Whether to clip geometries with the outline (default TRUE).
fill	character or function. Optional. Fill color. Functions can be used for choropleth maps or typologies.
stroke	character or function. Optional. Stroke color. Functions can be used for choropleth maps or typologies.
strokeWidth	numeric or function. Optional. Stroke width (default 1).
tip	logical or function. Optional. Tooltip definition (default FALSE). Use TRUE to display all fields.
simplify	numeric, vector, or logical. Optional. Geometry simplification (default FALSE). TRUE for automatic simplification, numeric for fixed tolerance, or vector c(k1, k2) for dynamic simplification depending on zoom level.
rewind	logical. Optional. Rewind polygon rings to correct winding order (default FALSE).
rewindPole	logical. Optional. Special rewinding for geometries crossing poles or the dateline (default FALSE).
clipOutline	numeric or logical. Optional. Clip geometries near the antimeridian and poles (default 0). If TRUE, uses 0.01 degrees.
r	numeric. Optional. Radius for point geometries (default 3). You can also use the shorthand r.
...	Additional SVG attributes (e.g. strokeDasharray, opacity, strokeLinecap, etc.).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(data = world, fill = "#38896F", stroke = "white") |>
  viz_render()
```

viz_pattern *Pattern layer*

Description

The viz_pattern function creates a reusable SVG pattern for thematic or cartographic styling. Patterns can be applied to any SVG shape (e.g. paths, rectangles) and support multiple textures such as lines, crosses, dots, waves, triangles, or zigzags. Patterns can also be clipped to a spatial data frame geometry or to the Earth outline.

Usage

```
viz_pattern(
  map,
  id = NULL,
  spacing = 6,
  angle = 0,
  fill = NULL,
  stroke = "#786d6c",
  strokeWidth = 2,
  strokeOpacity = 0.1,
  fillOpacity = 1,
  strokeDasharray = NULL,
  strokeLinecap = "butt",
  strokeLinejoin = "miter",
  strokeMiterlimit = 4,
  opacity = 1,
  visibility = "visible",
  display = NULL,
  pattern = "lines",
  data = NULL,
  clipOutline = FALSE,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique pattern id (auto-generated if not provided).
spacing	numeric. Optional. Distance between pattern elements (default 6).
angle	numeric. Optional. Pattern rotation in degrees (default 0).
fill	character or NULL. Optional. Fill color of pattern elements.
stroke	character. Optional. Stroke color (default "#786d6c").
strokeWidth	numeric. Optional. Stroke width (default 2).
strokeOpacity	numeric. Optional. Stroke opacity (default 0.1).

fillOpacity	numeric. Optional. Fill opacity (default 1).
strokeDasharray	character or NULL. Optional. Stroke dash pattern.
strokeLinecap	character. Optional. Line cap style (default "butt").
strokeLinejoin	character. Optional. Line join style (default "miter").
strokeMiterlimit	numeric. Optional. Miter limit (default 4).
opacity	numeric. Optional. Overall opacity (default 1).
visibility	character. Optional. SVG visibility property (default "visible").
display	character or NULL. Optional. SVG display property.
pattern	character. Optional. Pattern type (default "lines"). One of: "lines", "cross", "dots", "waves", "triangles", "zigzag".
data	object or NULL. Optional. Spatial data frame used to clip the pattern.
clipOutline	logical. Optional. Clip pattern to Earth outline (default FALSE).
...	Additional SVG attributes (e.g. strokeDasharray, opacity, strokeLinecap, etc.).

Value

A modified ‘geoviz‘ map object with a new effect added. Rendering is performed using `viz_render()`

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
africa <- world[world$region == "Africa", ]

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_pattern(
    data = africa, stroke = "#38896F", pattern = "cross",
    angle = 45, strokeWidth = 2, strokeOpacity = 0.6
  ) |>
  viz_render()
```

`viz_picto` *Pictogram layer*

Description

The `viz_picto` function creates a pictogram (symbol-based) map layer from a spatial data frame. It allows mapping qualitative variables to custom symbols and supports legend generation and ordering.

Usage

```
viz_picto(
  map,
  data = NULL,
  var,
  symbols = NULL,
  alphabetical = TRUE,
  legend = TRUE,
  leg_type = "vertical",
  leg_pos = c(10, 10),
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration over features.
var	character. Name of the categorical variable used for symbol assignment, or directly the name of a symbol.
symbols	character vector. Optional. Vector of available symbols. Symbols available: "circle", "square", "triangle", "pentagon", "hexagon", "roundsquare", "pillow", "drop", "egg", "star12", "star8", "star", "diamond", "trapzium", "plus", "minus", "arrow", "stop", "vbar", "crescent", "donut", "heart", "clover", "fist", "check", "plane", "rocket", "boat", "pin", "hospital", "flower", "cloud", "human", "tent", "beer", "boom", "nuke", "target", "missing"
alphabetical	logical. Optional. Whether to sort legend items alphabetically (default TRUE).
legend	logical. Optional. Whether to display a legend (default TRUE).
leg_type	character. Optional. Legend orientation (default "vertical"). One of: "horizontal", "vertical".
leg_pos	numeric vector. Optional. Legend position (default c(10, 10)).
...	Additional parameters passed to rendering functions (e.g. strokeWidth). With the leg_ prefix, you can configure the legend. For example: leg_title, leg_subtitle, leg_note, etc.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(
```

```
  projection = "EqualEarth", zoomable = TRUE
) |>
  viz_path(
    datum = world, fill = "#f1f3f5"
  ) |>
  viz_picto(
    data = world, var = "region",
    symbols = c("human", "heart", "fist", "clover", "rocket", "plane"),
    leg_type = "horizontal",
    leg_pos = c(400, 350),
    leg_symbol_spacing = 40,
    fill = "#38896F"
  ) |>
  viz_render()
```

`viz_prop` *Proportional symbols layer*

Description

The `viz_prop` function creates a proportional symbol layer from a spatial data frame. It can display different symbol types (circle, square, spike, half-circle) sized according to a numeric variable, and optionally includes a legend.

Usage

```
viz_prop(
  map,
  data = NULL,
  var,
  symbol = "circle",
  k = 50,
  fixmax = NULL,
  width = 30,
  straight = 0,
  dodge = FALSE,
  legend = TRUE,
  leg_type = "nested",
  leg_pos = c(10, 10),
  ...
)
```

Arguments

`map`	A `geovizr` map object created using `viz_create()`.#' @param map A `geoviz` map created with `viz_create`.
`data`	A spatial dataframe
`var`	character. Variable name containing numeric values used for scaling symbols.

symbol	character. Optional. Symbol type (default "circle"). One of: "circle", "square", "spike", "halfcircle".
k	numeric. Optional. Size of the largest symbol (default 50).
fixmax	numeric. Optional. Value corresponding to the symbol of size k. Useful to ensure comparability between maps.
width	numeric. Optional. Width of spike symbols (default 30).
straight	numeric. Optional. Curvature of spike symbols (default 0). Value between 0 (curved) and 1 (straight).
dodge	logical. Optional. Avoid symbol overlap (default FALSE, circle only).
legend	logical. Optional. Whether to display a legend (default TRUE).
leg_type	character. Optional. Legend type (default "nested"). One of: "nested", "separate".
leg_pos	numeric vector. Optional. Legend position (default c(10, 10)).
...	Additional parameters passed to symbol rendering (e.g. strokeWidth). With the leg_ prefix, you can configure the legend. For example: leg_title, leg_subtitle, leg_note, etc.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)

world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(
  projection = "EqualEarth", background = "white",
  zoomable = TRUE
) |>
  viz_path(
    datum = world, fill = "#f1f3f5"
  ) |>
  viz_prop(
    data = world, var = "pop", symbol = "circle", fill = "#38896F",
    k = 25, leg_values_round = 0, leg_title = "Population",
    leg_subtitle = "(million inh.)", leg_values_factor = 1 / 1000000
  ) |>
  viz_render()
```

viz_propchoro *Proportional + choropleth combined layer*

Description

The viz_propchoro function combines proportional symbols and choropleth mapping on the same spatial data frame. It allows simultaneous encoding of two numeric variables using symbols (size) and colors (classification).

Usage

```
viz_propchoro(
  map,
  data = NULL,
  var1,
  var2,
  var = NULL,
  symbol = "circle",
  k = 50,
  fixmax = NULL,
  dodge = FALSE,
  width = 30,
  straight = 0,
  method = "quantile",
  nb = 6,
  breaks = NULL,
  colors = NULL,
  middle = FALSE,
  sd = 1,
  reverse = FALSE,
  missing = "white",
  legend = TRUE,
  leg1_type = "nested",
  leg2_type = "vertical",
  leg1_pos = c(10, 10),
  leg2_pos = NULL,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration over features.
var1	character. Name of the numeric variable used for absolute values (driving symbol size).

var2	character. Name of the numeric variable used for relative values (driving color classification, e.g. percentages).
var	character. Optional. If provided, the same variable is used for both size and color encoding.
symbol	character. Optional. Symbol type (default "circle"). One of: "circle", "spike", "halfcircle".
k	numeric. Optional. Size of the largest symbol (default 50).
fixmax	numeric. Optional. Value corresponding to the symbol of size k. Useful for ensuring comparability between maps.
dodge	logical. Optional. Avoid symbol overlap (default FALSE, circle only).
width	numeric. Optional. Width of spike symbols (default 30).
straight	numeric. Optional. Curvature of spike symbols (default 0). Value between 0 (curved) and 1 (straight).
method	character. Optional. Classification method for choropleth (default "quantile"). One of: "quantile", "q6", "equal", "jenks", "msd", "geometric", "headtail", "pretty", "arithmetic", "nestedmeans".
nb	numeric. Optional. Number of classes (default 6).
breaks	numeric vector. Optional. Manual class breaks overriding nb and method.
colors	character or vector. Optional. Color palette or vector of colors from the Dicopal library. See https://observablehq.com/@neocartocnrs/dicopal-library
middle	logical. Optionnal Only for the MSD method: is the mean in a central class or not? (default FALSE)
sd	number. Only for the MSD method: the number of standard deviations taken into account (default 1)
reverse	logical. Optional. Reverse color palette (default FALSE).
missing	character or logical. Optional. Color for missing values (default "white").
legend	logical. Optional. Whether to display legends (default TRUE).
leg1_type	character. Optional. Symbol legend type (default "nested").
leg2_type	character. Optional. Choropleth legend type (default "vertical").
leg1_pos	numeric vector. Optional. Position of symbol legend (default c(10, 10)).
leg2_pos	numeric vector. Optional. Position of choropleth legend.
...	Additional parameters passed to rendering functions (e.g. strokeWidth). With leg1_ and leg2_ prefixes, you can configure the legends. For example: leg1_title, leg2_subtitle, leg1_note, etc.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(
  projection = "EqualEarth", background = "white",
  zoomable = TRUE
) |>
  viz_path(
    datum = world, fill = "#f1f3f5"
  ) |>
  viz_propchoro(
    data = world, var1 = "pop", k = 25,
    leg1_values_round = 0, var2 = "gdppc",
    leg1_title = "Population",
    leg1_subtitle = "(million inh.)",
    leg1_values_factor = 1 / 1000000,
    leg2_values_round = 0,
    leg2_title = "GDP per inh.",
    colors = "Temps"
  ) |>
  viz_render()
```

---

viz_proptypo *Proportional + typology combined layer*

Description

The viz_proptypo function combines proportional symbols and typology mapping on the same spatial data frame. It allows simultaneous encoding of two variables using symbol size (quantitative) and categorical coloring.

Usage

```
viz_proptypo(
  map,
  data = NULL,
  var1,
  var2,
  var = NULL,
  symbol = "circle",
  k = 50,
  fixmax = NULL,
  dodge = FALSE,
  width = 30,
```

```
  straight = 0,
  colors = NULL,
  order = NULL,
  alphabetical = TRUE,
  missing = "white",
  legend = TRUE,
  leg1_type = "nested",
  leg2_type = "vertical",
  leg1_pos = c(10, 10),
  leg2_pos = NULL,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration over features.
var1	character. Name of the numeric variable used for absolute values (driving symbol size).
var2	character. Name of the numeric variable used for categorical mapping.
var	character. Optional. If provided, the same variable is used for both size and color encoding.
symbol	character. Optional. Symbol type (default "circle"). One of: "circle", "spike", "halfcircle".
k	numeric. Optional. Size of the largest symbol (default 50).
fixmax	numeric. Optional. Value corresponding to the symbol of size k. Useful for ensuring comparability between maps.
dodge	logical. Optional. Avoid symbol overlap (default FALSE).
width	numeric. Optional. Width of spike symbols (default 30).
straight	numeric. Optional. Curvature of spike symbols (default 0). Value between 0 (curved) and 1 (straight).
colors	character or vector. Optional. Color palette or vector of colors from the Dicopal library. See https://observablehq.com/@neocartocnrs/dicopal-library
order	character vector. Optional. Explicit order of categorical values.
alphabetical	logical. Optional. Whether to sort categories alphabetically (default TRUE).
missing	character or logical. Optional. Color for missing values (default "white").
legend	logical. Optional. Whether to display legends (default TRUE).
leg1_type	character. Optional. Symbol legend type (default "nested").
leg2_type	character. Optional. Typology legend type (default "vertical").
leg1_pos	numeric vector. Optional. Position of symbol legend (default c(10, 10)).
leg2_pos	numeric vector. Optional. Position of typology legend.
...	Additional parameters passed to rendering functions (e.g. strokeWidth). With leg1_ and leg2_ prefixes, you can configure the legends. For example: leg1_title, leg2_subtitle, leg1_note, etc.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(
  projection = "EqualEarth", background = "white",
  zoomable = TRUE
) |>
  viz_path(
    datum = world, fill = "#f1f3f5"
  ) |>
  viz_proptypo(
    data = world, var1 = "pop", k = 25,
    leg1_values_round = 0, var2 = "region",
    leg1_title = "Population",
    symbol = "square",
    leg1_subtitle = "(million inh.)",
    leg1_values_factor = 1 / 1000000,
    leg2_title = "Continents",
    colors = "Set3"
  ) |>
  viz_render()
```

---

viz_radialGradient *Radial gradient*

Description

The `viz_radialGradient` function creates an SVG radialGradient definition and adds it to the SVG defs. It returns a reference usable in SVG styling (e.g. `url(#id)`). It is typically used to create smooth radial color transitions.

Usage

```
viz_radialGradient(
  map,
  id = NULL,
  color1 = "#63b0af",
  color2 = "#428c8b",
  offset1 = 50,
  offset2 = 100,
  fx = 50,
```

```
  fy = 50,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique gradient id.
color1	character. Optional. First color of the gradient (default "#63b0af").
color2	character. Optional. Second color of the gradient (default "#428c8b").
offset1	numeric. Optional. Offset of the first color stop (default 50).
offset2	numeric. Optional. Offset of the second color stop (default 100).
fx	numeric. Optional. Focal point x-position (default 50).
fy	numeric. Optional. Focal point y-position (default 50).
...	Additional parameters

Value

A modified ‘geoviz‘ map object with a new effect added. Rendering is performed using viz_render()

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
aus <- world[world$ISO3 == "AUS", ]

viz_create() |>
  viz_radialGradient(
    id = "my_gradient", color1 = "#63b0af",
    color2 = "#428c8b"
  ) |>
  viz_path(datum = aus, fill = "url(#my_gradient)") |>
  viz_render()
```

---

viz_render *Display the map*

Description

The viz_render function allows to display the map

Usage

```
viz_render(map)
```

Arguments

map	A geovizr map object created using viz_create().

Value

An htmlwidget object representing an interactive map.

Examples

```
library(sf)

world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create() |>
  viz_outline() |>
  viz_path(data = world) |>
  viz_render()
```

---

viz_rhumbs *Rhumb lines layer*

Description

The viz_rhumbs function draws rhumb lines (loxodromes), similar to those found on old portolan charts. These lines represent paths of constant bearing.

Usage

```
viz_rhumbs(
  map,
  id = NULL,
  nb = 16,
  pos = c(10, 10),
  coords = "svg",
  stroke = "#394a70",
  strokeWidth = 1,
  strokeOpacity = 0.3,
  strokeDasharray = c(3, 2),
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
nb	numeric. Optional. Number of lines (default 16).
pos	numeric vector. Optional. Position of the lines (default c(10, 10)). If coords = "svg", values are in SVG coordinates. If coords = "geo", values are longitude and latitude.
coords	character. Optional. Coordinate system for pos (default "svg"). If coords = "geo" and the map is zoomable, lines follow zoom interactions.
stroke	character. Optional. Stroke color (default "#394a70").
strokeWidth	numeric. Optional. Stroke width (default 1).
strokeOpacity	numeric. Optional. Stroke opacity (default 0.3).
strokeDasharray	numeric or vector. Optional. Stroke dash pattern (default c(3, 2)).
...	Additional SVG attributes (e.g. opacity, strokeLinecap, etc.).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_rhumbs(
    nb = 32, coords = "geo", pos = c(0, 0),
    fill = "#38896F"
  ) |>
  viz_render()
```

---

viz_save *Save the map*

Description

The viz_save function allows to download the map

Usage

```
viz_save(map)
```

Arguments

map	a geoviz object create with the viz_create

Value

A file path to the downloaded SVG map.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create() |>
  viz_outline() |>
  viz_path(data = world) |>
  viz_save()
```

---

viz_scalebar *Scale bar*

Description

The viz_scalebar function adds a scalebar on the map.

Usage

```
viz_scalebar(
  map,
  id = NULL,
  pos = NULL,
  translate = "",
  units = "km",
  label = "",
  tickSize = 0.2,
  tickPadding = 5,
  distance = "",
  tickValues = "",
  labelAnchor = "start",
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
pos	numeric vector of length 2. Optional. Position [x, y] (default c(10, svg.height - 20)).
translate	numeric vector of length 2. Optional. Translation without changing size (default NA).
units	character. Optional. Units ("m","km","mi","ft") (default "km").
label	character. Optional. Label to display.
tickSize	numeric. Optional. Tick size (default 0.2).
tickPadding	numeric. Optional. Tick padding (default 5).
distance	numeric. Optional. Distance represented by the scalebar.
tickValues	numeric vector. Optional. Custom tick values.
labelAnchor	character. Optional. Label anchor ("start","middle","end") (default "start").
...	Additional SVG attributes applied to the scalebar.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "Mercator", background = "white") |>
  viz_path(datum = world[world$region == "Africa", ], fill = "#f1f3f5") |>
  viz_scalebar() |>
  viz_render()
```

---

viz_shadow *Shadow layer*

Description

The viz_shadow function adds a shadow effect to map elements, typically used to enhance depth perception for shapes, symbols, or extruded visualizations. The layer is rendered in SVG and can be applied globally or to specific geometries.

Usage

```
viz_shadow(
  map,
  id = NULL,
  dx = 0,
  dy = 0,
  blur = 0,
  opacity = 1,
  fill = NULL,
  stroke = NULL,
  strokeWidth = NULL,
  coords = "geo",
  ...
)
```

Arguments

map	A geovizr map object created using viz_create.
id	character. Optional. Unique layer id.
dx	numeric. Optional. Horizontal shadow offset (default 0).
dy	numeric. Optional. Vertical shadow offset (default 0).
blur	numeric. Optional. Blur intensity of the shadow (default 0).
opacity	numeric. Optional. Shadow opacity (default 1).
fill	character or function. Optional. Shadow fill color.
stroke	character or function. Optional. Shadow stroke color.
strokeWidth	numeric. Optional. Shadow stroke width.
coords	character. Optional. Coordinate system (default "geo"). Use "svg" if coordinates are already expressed in SVG space.
...	Additional parameters

Value

A modified ‘geoviz‘ map object with a new effect added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
aus <- world[world$ISO3 == "AUS", ]

viz_create(margin = 5) |>
  viz_shadow(id = "my_shadow_effect", stdDeviation = 2.5, dx = 5, dy = 5) |>
  viz_path(datum = aus, fill = "#38896F", filter = "url(#my_shadow_effect)") |>
  viz_render()
```

viz_sketch *Sketch layer*

Description

The viz_sketch function renders a spatial dataframe as hand-drawn (sketchy) SVG shapes. It uses SVG filters (feTurbulence and feDisplacementMap) to simulate a pencil-like rendering style.

Usage

```
viz_sketch(
  map,
  data = NULL,
  id = NULL,
  fill = "none",
  stroke = "#000",
  strokeWidth = 1,
  simplify = NULL,
  baseFrequency = 0.03,
  feDisplacementMap = 5,
  fillStyle = "dashed",
  roughness = 5,
  hachureGap = 3,
  bowing = 30,
  fillWeight = 0.12,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	object. Optional. A spatial data frame (or equivalent geometry input).
id	character. Optional. Unique layer id (auto-generated if not provided).
fill	character. Optional. Fill color (default "none").
stroke	character. Optional. Stroke color (default "#000").
strokeWidth	numeric. Optional. Stroke width (default 1).
simplify	numeric or vector or logical. Optional. Geometry simplification
baseFrequency	numeric. Optional. Base frequency of the turbulence filter (default 0.03). Controls noise density.
feDisplacementMap	numeric. Optional. Displacement intensity of the sketch effect (default 5).
fillStyle	character. Optional. Fill style (default "dashed").
roughness	numeric. Optional. Roughness level (reserved for future use, default 5).
hachureGap	numeric. Optional. Gap between hachure lines (reserved, default 3).

bowing	numeric. Optional. Line bowing effect (reserved, default 30).
fillWeight	numeric. Optional. Fill stroke weight (reserved, default 0.12).
...	Additional SVG attributes (e.g. strokeDasharray, opacity, strokeLinecap, etc.).

Value

A modified 'geoviz' map object with a new effect added. Rendering is performed using viz_render()

Examples

```
library(sf)

world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(width = 750, background = "white", projection = "EqualEarth") |>
  viz_sketch(data = world, stroke = "#38896F", strokeWidth = 1) |>
  viz_render()
```

---

viz_smooth *Smoothed density (isobands) layer*

Description

The viz_smooth function creates smoothed density contours (isobands) from a set of spatial points. It is a wrapper around contour-based density estimation, with styling optimized for smooth thematic visualizations.

Usage

```
viz_smooth(
  map,
  data = NULL,
  var = NULL,
  id = NULL,
  nb = 1e+05,
  bandwidth = NULL,
  fixbandwidth = FALSE,
  thresholds = NULL,
  cellSize = NULL,
  stroke = "white",
  strokeOpacity = 0.8,
  strokeWidth = 0.3,
  shadow = TRUE,
  fill = NULL,
```

```
  colors = "RdPu",
  opacity = NULL,
  fillOpacity = 0.6,
  tip = NULL,
  tipstyle = NULL,
  coords = "geo",
  legend = TRUE,
  leg_pos = c(10, 10),
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration over features.
var	character or numeric. Optional. Variable used to weight points.
id	character. Optional. Layer identifier.
nb	numeric. Optional. Number of sampled points used for density estimation (default 100000).
bandwidth	numeric. Optional. Bandwidth used for density computation.
fixbandwidth	logical. Optional. Whether to scale bandwidth by zoom factor (default FALSE).
thresholds	numeric. Optional. Number of contour levels.
cellSize	numeric. Optional. Grid cell size for density computation.
stroke	character. Optional. Stroke color (default "white").
strokeOpacity	numeric. Optional. Stroke opacity (default 0.8).
strokeWidth	numeric. Optional. Stroke width (default 0.3).
shadow	logical. Optional. Whether to add a shadow filter (default TRUE).
fill	character or function. Optional. Single fill color (used if colors is not defined).
colors	character. Optional. Name of a Dicopal sequential palette (default "RdPu"). See https://observablehq.com/@neocartocnrs/dicopal-library.
opacity	numeric. Optional. Global opacity.
fillOpacity	numeric. Optional. Fill opacity for polygons (default 0.6).
tip	logical or function. Optional. Tooltip function; if TRUE displays all properties.
tipstyle	list. Optional. Custom tooltip styling.
coords	character. Optional. Coordinate system (default "geo"). One of: "geo", "svg".
legend	logical. Optional. Whether to display legend (default TRUE).
leg_pos	numeric vector. Optional. Legend position (default c(10, 10)).
...	Additional SVG attributes (e.g. strokeDasharray, opacity, strokeLinecap, etc.).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)
cities <- st_read(
  system.file("gpkg/cities.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_smooth(data = cities, var = "population") |>
  viz_render()
```

---

viz_spike *Spike layer*

Description

The viz_spike function draws spikes on the map from a spatial data frame or from a single position. It can be used to represent values with vertical symbols (e.g. for density or intensity maps).

Usage

```
viz_spike(
  map,
  data = NULL,
  id = NULL,
  pos = c(0, 0),
  height = 10,
  width = 30,
  straight = 0,
  k = 100,
  fixmax = NULL,
  sort = NULL,
  descending = NULL,
  coords = "geo",
  fill = NULL,
  stroke = NULL,
  tip = FALSE,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	object. Optional. A spatial data frame.
id	character. Optional. Unique layer id.
pos	numeric vector. Optional. Position of a single spike (default c(0, 0)).
height	numeric or character. Optional. Spike height (default 10). Can be a fixed value or the name of a field containing numerical values.
width	numeric. Optional. Spike width (default 30).
straight	numeric. Optional. Controls spike curvature (default 0). Value between 0 (curved) and 1 (straight).
k	numeric. Optional. Height of the highest spike (default 100).
fixmax	numeric. Optional. Value corresponding to the spike of height k. Useful to ensure comparability between maps.
sort	character or function. Optional. Field name or function to sort spikes.
descending	logical. Optional. Sorting order.
coords	character. Optional. Coordinate system (default "geo"). Use "svg" if coordinates are already expressed in the SVG coordinate space.
fill	character or function. Optional. Fill color.
stroke	character or function. Optional. Stroke color.
tip	logical or function. Optional. Tooltip definition (default FALSE). Use TRUE to display all fields.
...	Additional SVG attributes (e.g. strokeDasharray, strokeWidth, opacity, strokeLinecap, etc.).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_square(data = world, height = 100, fill = "#38896F") |>
  viz_render()
```

viz_square *Square layer*

Description

The viz_square function draws rotatable squares on the map from a spatial data frame or from a single position. It can be used to create proportional symbol maps with square markers.

Usage

```
viz_square(
  map,
  data = NULL,
  id = NULL,
  pos = c(0, 0),
  dx = 0,
  dy = 0,
  angle = 0,
  side = 20,
  k = 100,
  fixmax = NULL,
  sort = NULL,
  descending = NULL,
  coords = "geo",
  fill = NULL,
  stroke = NULL,
  tip = FALSE,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	object. Optional. A spatial data frame.
id	character. Optional. Unique layer id.
pos	numeric vector. Optional. Position of a single square (default c(0, 0)).
dx	numeric. Optional. Horizontal shift (default 0).
dy	numeric. Optional. Vertical shift (default 0).
angle	numeric. Optional. Rotation angle in degrees (default 0).
side	numeric or character. Optional. Square size (default 20). Can be a fixed value or the name of a field containing numerical values.
k	numeric. Optional. Size of the largest square (default 100).
fixmax	numeric. Optional. Value corresponding to the square of size k. Useful to ensure comparability between maps.

sort	character or function. Optional. Field name or function to sort squares.
descending	logical. Optional. Sorting order.
coords	character. Optional. Coordinate system (default "geo"). Use "svg" if coordinates are already expressed in the SVG coordinate space.
fill	character or function. Optional. Fill color.
stroke	character or function. Optional. Stroke color.
tip	logical or function. Optional. Tooltip definition (default FALSE). Use TRUE to display all fields.
...	Additional SVG attributes (e.g. strokeDasharray, strokeWidth, opacity, strokeLinecap, etc.).

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_square(data = world, side = 60, fill = "#38896F") |>
  viz_render()
```

---

viz_symbol *Symbol layer*

Description

The viz_symbol function draws SVG symbols on the map from a spatial data frame. It allows the use of predefined symbols, scaling, rotation, and styling, and can be used for categorical or proportional symbol maps.

Usage

```
viz_symbol(
  map,
  data = NULL,
  id = NULL,
  pos = c(0, 0),
  fill = NULL,
  stroke = "white",
```

```
  strokeWidth = 0.2,
  coords = "geo",
  r = 12,
  scale = NULL,
  symbol = "star",
  missing = "missing",
  rotate = 0,
  skewX = 0,
  skewY = 0,
  background = FALSE,
  tip = FALSE,
  k = 50,
  fixmax = NULL,
  dodge = FALSE,
  iteration = 200,
  sort = NULL,
  descending = NULL,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	object. Optional. A spatial data frame.
id	character. Optional. Unique layer id.
pos	numeric vector. Optional. Position of a single symbol (default c(0, 0)).
fill	character or function. Optional. Fill color.
stroke	character or function. Optional. Stroke color (default "white").
strokeWidth	numeric or function. Optional. Stroke width (default 0.2).
coords	character. Optional. Coordinate system (default "geo"). Use "svg" if coordinates are already expressed in the SVG coordinate space.
r	numeric or character. Optional. Radius defining symbol size (default 12).
scale	numeric. Optional. Global scale factor for symbols.
symbol	character. Optional. Symbol name. "circle", "square", "triangle", "pentagon", "hexagon", "roundsquare", "pillow", "drop", "egg", "star12", "star8", "star", "diamond", "trapzium", "plus", "minus", "arrow", "stop", "vbar", "crescent", "donut", "heart", "clover", "fist", "check", "plane", "rocket", "boat", "pin", "hospital", "flower", "cloud", "human", "tent", "beer", "boom", "nuke", "target", "missing". (default "star"). different symbols are assigned per category.
missing	character. Optional. Symbol used for missing values (default "missing"). Use NULL to hide missing values.
rotate	numeric. Optional. Rotation angle (default 0).
skewX	numeric. Optional. Horizontal skew (default 0).
skewY	numeric. Optional. Vertical skew (default 0).

background	logical. Optional. Add a background circle (default FALSE). You can confifure it with the prefix background_ (e.g. background_fill, background_stroke, etc.).
tip	logical or function. Optional. Tooltip definition (default FALSE). Use TRUE to display all fields.
k	numeric. Optional. Maximum radius scaling value (default 50).
fixmax	numeric. Optional. Value corresponding to the maximum radius k.
dodge	logical. Optional. Avoid symbol overlap (default FALSE).
iteration	numeric. Optional. Number of iterations for dodging (default 200).
sort	character or function. Optional. Field name or function to sort symbols.
descending	logical. Optional. Sorting order.
...	Additional SVG attributes (e.g. strokeDasharray, opacity, strokeLinecap, etc.).

Value

A modified 'geoviz' map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "EqualEarth") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_symbol(data = world, symbol = "star", fill = "#38896F") |>
  viz_render()
```

---

viz_text *Texts and labels*

Description

The viz_text function adds a text on a geoviz map and can also generate labels from a spatial dataframe.

Usage

```
viz_text(
  map,
  id = NULL,
  data = NULL,
  text = "text",
```

```
  textAnchor = NULL,
  dominantBaseline = NULL,
  fontFamily = NULL,
  fontSize = 12,
  lineSpacing = 0,
  pos = c(0, 0),
  dx = 0,
  dy = 0,
  sort = NULL,
  descending = FALSE,
  coords = "geo",
  fill = NULL,
  stroke = NULL,
  strokeWidth = 1,
  strokeLinejoin = "round",
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
data	a spatial dataframe.
text	character or function. Optional. Text to display (default "text").
textAnchor	character or function. Optional. Text anchor ("start","middle","end").
dominantBaseline	character or function. Optional. Baseline alignment ("auto","middle","central","hanging").
fontFamily	character. Optional. Font family from SVG container.
fontSize	numeric. Optional. Font size (default 12).
lineSpacing	numeric. Optional. Line spacing (default 0).
pos	numeric vector of length 2. Optional. Position for single text element (default c(0,0)).
dx	numeric. Optional. X shift (default 0).
dy	numeric. Optional. Y shift (default 0).
sort	character or function. Optional. Field or function used to sort labels.
descending	logical. Optional. Sorting order (default FALSE).
coords	character. Optional. Coordinate system ("geo" or "svg") (default "geo").
fill	character or function. Optional. Fill color.
stroke	character or function. Optional. Stroke color.
strokeWidth	numeric. Optional. Stroke width (default 1).
strokeLinejoin	character or function. Optional. Stroke line join (default "round").
...	Additional SVG attributes applied to text elements.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using `viz_render()`.

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

# Example 1
viz_create(projection = "EqualEarth", width = 750, background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_text(
    pos = c(100, 200), coords = "svg",
    text = "All maps are lies,\nbut some are useful lies",
    fill = "#38896F"
  ) |>
  viz_render()

# Example 2
viz_create(projection = "EqualEarth", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_text(data = world, text = "ISO3", fill = "#38896F") |>
  viz_render()
```

---

viz_tile *Mercator tiles*

Description

The `viz_tile` function adds raster zoomable tiles to a map It requires a Mercator projection.

Usage

```
viz_tile(
  map,
  id = NULL,
  tileSize = 512,
  zoomDelta = 1,
  opacity = 1,
  url = "openstreetmap",
  clipPath = NULL
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
tileSize	numeric. Optional. Tile size (default 512).
zoomDelta	numeric. Optional. Zoom offset (default 1).
opacity	numeric. Optional. Tile opacity (default 1).
url	function or character. Optional. Tile source URL or preset ("openstreetmap", "opentopomap", "worldterrain", "worldimagery", "worldStreet", "worldphysical", "shadedrelief", "stamenterrain", "cartodbvoyager", "stamentoner", "stamentonerbackground", "stamentonerlite", "stamenwatercolor", "hillshade", "worldocean", "natgeo").
clipPath	character. Optional. SVG clip-path definition (e.g. "url(#myclipid)").

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
viz_create(projection = "mercator") |>
  viz_tile(url = "worldphysical") |>
  viz_render()
```

---

viz_tissot *Tissot indicatrices*

Description

The viz_tissot function draws Tissot circles to visualize projection distortions on a map.

Usage

```
viz_tissot(
  map,
  id = NULL,
  step = 20,
  fill = "red",
  stroke = "white",
  strokeOpacity = 0.5,
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
id	character. Optional. Unique layer id.
step	numeric. Optional. Step between circles (default 20).
fill	character. Optional. Fill color (default "red").
stroke	character. Optional. Stroke color (default "white").
strokeOpacity	numeric. Optional. Stroke opacity (default 0.5).
...	Additional SVG attributes (strokeDasharray, strokeWidth, opacity, strokeLinecap...).

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(projection = "Polar", background = "white") |>
  viz_path(datum = world, fill = "#f1f3f5") |>
  viz_tissot(fill = "#38896F") |>
  viz_render()
```

---

viz_typo *Typology layer*

Description

The viz_typo function creates a typology map from a spatial data frame by mapping categorical variables to colors. It supports custom ordering, color palettes, and automatic legend generation.

Usage

```
viz_typo(
  map,
  data = NULL,
  var,
  colors = NULL,
  order = NULL,
  alphabetical = TRUE,
  missing = "white",
  legend = TRUE,
  leg_type = "vertical",
```

```
  leg_pos = c(10, 10),
  ...
)
```

Arguments

map	A geovizr map object created using viz_create().
data	A spatial dataframe Use data to enable iteration over features.
var	character. Name of the categorical variable used for styling. You can also use fill or stroke directly instead of var.
colors	character or vector. Optional. Color palette or vector of colors. Can use named palettes from the Dicopal library (e.g. "Antique", "Bold", "Pastel", "Prism", "Safe", "Vivid", "Accent", "Dark2", "Paired", "Pastel1", "Pastel2", "Set1", "Set2", "Set3"). See https://observablehq.com/@neocartocnrs/dicopal-library
order	character vector. Optional. Explicit ordering of categories.
alphabetical	logical. Optional. Whether to sort legend items alphabetically (default TRUE).
missing	character or logical. Optional. Color for missing values (default "white").
legend	logical. Optional. Whether to display a legend (default TRUE).
leg_type	character. Optional. Legend orientation (default "vertical"). One of: "horizontal", "vertical".
leg_pos	numeric vector. Optional. Legend position (default c(10, 10)).
...	Additional parameters passed to path rendering. With the leg_ prefix, you can configure the legend. For example: leg_title, leg_subtitle, leg_note, etc.

Value

A modified ‘geoviz‘ map object with a new layer added. Rendering is performed using viz_render().

Examples

```
library(sf)
world <- st_read(
  system.file("gpkg/world.gpkg", package = "geovizr"),
  quiet = TRUE
)

viz_create(
  projection = "EqualEarth", background = "white",
  zoomable = TRUE
) |>
  viz_typo(
    data = world, var = "region", colors = "Pastel",
    leg_title = "Continents"
  ) |>
  viz_render()
```

Index