Package ‘zonebuilder’

October 14, 2022

Title  Create and Explore Geographic Zoning Systems
Version  0.0.2
Description  Functions, documentation and example data to help divide geographic space into discrete polygons (zones).
The functions are motivated by research into the merits of different zoning systems.
<doi:10.1068/a090169>. A flexible ‘ClockBoard’ zoning system is provided, which breaks-up space by concentric rings
and radial lines emanating from a central point.
By default, the diameter of the rings grow according the triangular number sequence
<doi:10.1080/26375451.2019.1598687> with the first 4 ‘doughnuts’
(or ‘annuli’) measuring 1, 3, 6, and 10 km wide.
These annuli are subdivided into equal segments (12 by default), creating the visual impression of a dartboard. Zones are labelled according to distance to the centre and angular distance from North, creating a simple geographic zoning and labelling system useful for visualising geographic phenomena with a clearly demarcated central location such as cities.
License  GPL-3
BugReports  https://github.com/zonebuilders/zonebuilder/issues
Depends  R (>= 2.10)
Imports  sf, RColorBrewer, graphics, grDevices
Suggests  knitr, rmarkdown, tmap, tmaptools, dplyr, lwgeom,
          leaflet, covr, bookdown
VignetteBuilder  knitr
URL  https://github.com/zonebuilders/zonebuilder,
     https://zonebuilders.github.io/zonebuilder/
Encoding  UTF-8
LazyData  true
RoxygenNote  7.1.1
NeedsCompilation  no
Author  Robin Lovelace [aut, cre] (<https://orcid.org/0000-0001-5679-6536>),
        Martijn Tennekes [aut]
Maintainer  Robin Lovelace <rob00x@gmail.com>
Repository  CRAN
Date/Publication  2021-07-12 22:30:02 UTC

R topics documented:

- `london_area` ............................................ 2
- `zb_100_triangular_numbers` ............................ 3
- `zb_color` .................................................. 3
- `zb_doughnut` ............................................. 4
- `zb_lines` .................................................. 5
- `zb_plot` ................................................... 5
- `zb_quadrat` .............................................. 6
- `zb_segment` .............................................. 7
- `zb_view` .................................................. 8
- `zb_zone` .................................................. 8

Index  11

---

**london_area**  Region representing London in projected coordinate system

**Description**

‘`london_a()`’ and ‘`london_c()`’ return the city boundaries and centre point of London, respectively.

**Usage**

```r
london_a()

london_c()
```

**Note**

‘`london_a()`’ returns a projected version of ‘`lnd` in ‘spDataLarge’. See the ‘data-raw’ folder in the package’s repo to reproduce these datasets. The ‘lonlat’ versions of the data have coordinates in units of degrees.

**Examples**

```r
plot(london_a(), reset = FALSE)
plot(london_c(), add = TRUE)
```
zb_100_triangular_numbers

The first 100 triangular numbers

Description

The first 100 in the sequence of [triangular numbers](https://en.wikipedia.org/wiki/Triangular_number)

Note

See the ‘data-raw’ folder in the package’s repo to reproduce these datasets

zb_color

Generate colors for zones

Description

This function generates colors for zones.

Usage

zb_color(z, palette = c("rings", "hcl", "dartboard"))

Arguments

z
An ‘sf’ object containing zones covering the region

palette
Palette type, one of "hcl" (a palette based on the HCL color space), "rings" (a palette which colors the rings using the YlOrBr color brewer palette), "dartboard" (a palette which resembles a dartboard)

Value

A vector of colors

Examples

z = zb_zone(london_c(), london_a())
zb_color(z)
plot(z[, "circle_id"], col = zb_color(z))
zb_doughnut  

Make doughnuts

Description

Make doughnuts

Usage

zb_doughnut(
  x = NULL,
  area = NULL,
  n_circles = NA,
  distance = 1,
  distance_growth = 1
)

Arguments

x  Centre point. Should be an sf or sfc object containing one point, or a name of a city (which is looked up with OSM geocoding).

area  (optional) Area. Should be an sf or sfc object containing one (multi) polygon

n_circles  Number of rings including the central circle. By default 5, unless area is specified (then it is set automatically to fill the area).

distance  Distance The distances between the circles. For the center circle, it is the distance between the center and the circle. If only one number is specified, distance_growth determines the increment at which the distances grow for the outer circles.

distance_growth  The rate at which the distances between the circles grow. Only applicable when distance is one number and n_circles > 1. See also distance.

Value

An ‘sf’ data frame

Examples

zb_plot(zb_doughnut(london_c(), london_a()))
zb_lines

Create lines radiating at equal angles from a point

**Description**

Create lines radiating at equal angles from a point

**Usage**

```r
zb_lines(point, n, starting_angle = 45, distance = 1e+05)
```

**Arguments**

- `point`: Center point
- `n`: Number of lines
- `starting_angle`: Starting angle
- `distance`: Distance

**Value**

Objects of class ‘sfc’ containing linestring geometries

**Examples**

```r
point = sf::st_centroid(london_a())
n = 4
l = zb_lines(point, n)
plot(l)
```

zb_plot

Plot zones

**Description**

This function opens a static map of the zones

**Usage**

```r
zb_plot(
  z,
  palette = c("rings", "hcl", "dartboard"),
  title = NULL,
  text_size = c(0.3, 1),
  zone_label_thres = 0.002
)
```
zb_quadrat

Arguments

z An ‘sf’ object containing zones covering the region
palette Palette type, one of "hcl" (a palette based on the HCL color space), "rings" (a palette which colors the rings using the YlOrBr color brewer palette), "dartboard" (a palette which resembles a dartboard)
title Plot title
text_size Vector of two numeric values that determine the relative text sizes. The first determines the smallest text size and the second one the largest text size. The largest text size is used for the outermost circle, and the smallest for the central circle in case there are 9 or more circles. If there are less circles, the relative text size is larger (see source code for exact method)
zone_label_thres This number determines in which zones labels are printed, namely each zone for which the relative area size is larger than ‘zone_label_thres’.

Value
A static plot created using R’s base ‘graphics’ package

Examples

zb_plot(zb_zone(london_c()))

zb_quadrat

Divide a region into quadrats

Description
Divide a region into quadrats

Usage

zb_quadrat(x, ncol, nrow = NULL, intersection = TRUE)

Arguments

x
ncol
nrow
intersection

Value
An sf object
zb_segment

Examples

x = london_a()
c = sf::st_centroid(london_a())
plot(zb_quadrat(x, ncol = 2), col = 2:5)
plot(c, add = TRUE, col = "white")
plot(zb_quadrat(x, ncol = 3))
plot(zb_quadrat(x, ncol = 4))
plot(zb_quadrat(x, ncol = 4, intersection = FALSE))

zb_segment

Make segments

Description

Make segments

Usage

zb_segment(x = NULL, area = NULL, n_segments = 12, distance = NA)

Arguments

x Centre point. Should be an sf or sfc object containing one point, or a name of
a city (which is looked up with OSM geocoding).
area (optional) Area. Should be an sf or sfc object containing one (multi) polygon
n_segments (optional) Number of segments. The number of segments. Either one number
which determines the number of segments applied to all circles, or a vector with
a number for each circle (which should be a multiple of 4, see also the argument
labeling). By default, the central circle is not segmented (see the argument
segment_center).
distance Distance The distances between the circles. For the center circle, it is the
distance between the center and the circle. If only one number is specified,
distance_growth determines the increment at which the distances grow for the
outer circles.

Value

An ‘sf’ data frame

Examples

zb_plot(zb_segment(london_c(), london_a()))
zb_view  View zones

Description
This function opens an interactive map of the zones

Usage
zb_view(z, alpha = 0.4, palette = c("rings", "hcl", "dartboard"), title = NULL)

Arguments

- **z**: An ‘sf’ object containing zones covering the region

- **alpha**: Alpha transparency, number between 0 (fully transparent) and 1 (not transparent)

- **palette**: Palette type, one of "hcl" (a palette based on the HCL color space), "rings" (a palette which colors the rings using the YlOrBr color brewer palette), "dartboard" (a palette which resembles a dartboard)

- **title**: The title of the plot

Value
An interactive map created with ‘tmap’

Examples

```r
z = zb_zone(london_c(), london_a())
zbin_view(z, palette = "rings")
```

zb_zone  Generate zones covering a region of interest

Description
This function first divides geographic space into annuli (concentric 2d rings or ‘doughnuts’) and then subdivides each annulus into a number of segments.
zb_zone

Usage

zb_zone(
  x = NULL,
  area = NULL,
  n_circles = NA,
  n_segments = 12,
  distance = 1,
  distance_growth = 1,
  labeling = NA,
  starting_angle = NA,
  segment_center = FALSE,
  intersection = TRUE,
  city = NULL
)

Arguments

x Centre point. Should be an sf or sfc object containing one point, or a name of a city (which is looked up with OSM geocoding).

area (optional) Area. Should be an sf or sfc object containing one (multi) polygon

n_circles Number of rings including the central circle. By default 5, unless area is specified (then it is set automatically to fill the area).

n_segments (optional) Number of segments. The number of segments. Either one number which determines the number of segments applied to all circles, or a vector with a number for each circle (which should be a multiple of 4, see also the argument labeling). By default, the central circle is not segmented (see the argument segment_center).

distance Distance The distances between the circles. For the center circle, it is the distance between the center and the circle. If only one number is specified, distance_growth determines the increment at which the distances grow for the outer circles.

distance_growth The rate at which the distances between the circles grow. Only applicable when distance is one number and n_circles > 1. See also distance.

labeling The labeling of the zones. Either "clock" which uses the clock analogy (i.e. hours 1 to 12) or "NESW" which uses the cardinal directions N, E, S, W. If the number of segments is 12, the clock labeling is used, and otherwise NESW. Note that the number of segments should be a multiple of four. If, for instance the number of segments is 8, than the segments are labeled N1, N2, E1, E2, S1, S2, W1, and W2.

starting_angle The angle of the first of the radii that create the segments (degrees). By default, it is either 15 when n_segments is 12 (i.e. the ClockBoard setting) and -45 otherwise.

segment_center Should the central circle be divided into segments? FALSE by default.

intersection Should the zones be intersected with the area? TRUE by default.

city (optional) Name of the city. If specified, it adds a column ‘city’ to the returned ‘sf’ object.
Details

By default 12 segments are used for each annuli, resulting in a zoning system that can be used to refer to segments in [clock position](https://en.wikipedia.org/wiki/Clock_position), with 12 representing North, 3 representing East, 6 South and 9 Western segments.

Value

An `sf` object containing zones covering the region

Examples

```r
# default settings
z = zb_zone(london_c(), london_a())

zb_plot(z)
if (require(tmap)) {
  zb_view(z)
  z = zb_zone("Berlin")
  zb_view(z)
}

# variations
zb_plot(zb_zone(london_c(), london_a(), n_circles = 2))
zb_plot(zb_zone(london_c(), london_a(), n_circles = 4, distance = 2, distance_growth = 0))
zb_plot(zb_zone(london_c(), london_a(), n_circles = 3, n_segments = c(1,4,8)))
```
Index

* datasets
  - london_area, 2
  - zb_100_triangular_numbers, 3
  - london_a(london_area), 2
  - london_area, 2
  - london_area_lonlat(london_area), 2
  - london_c(london_area), 2
  - london_cent(london_area), 2
  - london_cent_lonlat(london_area), 2
  - sf, 4, 7, 9
  - sfc, 4, 7, 9
  - zb_100_triangular_numbers, 3
  - zb_color, 3
  - zb_doughnut, 4
  - zb_lines, 5
  - zb_plot, 5
  - zb_quadrat, 6
  - zb_segment, 7
  - zb_view, 8
  - zb_zone, 8