Package ‘cartogram’

May 26, 2023

**Title**  Create Cartograms with R

**Version**  0.3.0

**Description**  Construct continuous and non-contiguous area cartograms.

**URL**  https://github.com/sjewo/cartogram

**BugReports**  https://github.com/sjewo/cartogram/issues

**Imports**  methods, sf, packcircles

**Suggests**

**License**  GPL-3

**Encoding**  UTF-8

**RoxygenNote**  7.2.3

**NeedsCompilation**  no

**Author**  Sebastian Jeworutzki [aut, cre]  
(https://orcid.org/0000-0002-2671-5253),  
Timothee Giraud [ctb],  
Nicolas Lambert [ctb],  
Roger Bivand [cph],  
Edzer Pebesma [cph],  
Jakub Nowosad [ctb] (https://orcid.org/0000-0002-1057-3721)

**Maintainer**  Sebastian Jeworutzki <sebastian.jeworutzki@ruhr-uni-bochum.de>

**Repository**  CRAN

**Date/Publication**  2023-05-26 11:30:02 UTC

**R topics documented:**

- cartogram_cont .................................................. 2
- cartogram_dorling .................................................. 3
- cartogram_ncont .................................................. 5

**Index**  7
cartogram_cont  

**Calculate Contiguous Cartogram Boundaries**

**Description**

Construct a continuous area cartogram by a rubber sheet distortion algorithm (Dougenik et al. 1985)

**Usage**

```r
cartogram_cont(
  x, weight, itermax = 15, maxSizeError = 1.0001, prepare = "adjust", threshold = 0.05, verbose = FALSE
)
```

**Arguments**

- **x**: a polygon or multipolygon sf object
- **weight**: Name of the weighting variable in x
- **itermax**: Maximum iterations for the cartogram transformation, if maxSizeError ist not reached
maxSizeError  Stop if meanSizeError is smaller than maxSizeError
prepare  Weighting values are adjusted to reach convergence much earlier. Possible methods are "adjust", adjust values to restrict the mass vector to the quantiles defined by threshold and 1-threshold (default), "remove", remove features with values lower than quantile at threshold, "none", don’t adjust weighting values
threshold  Define threshold for data preparation
verbose  print meanSizeError on each iteration

Value
An object of the same class as x

References

Examples
library(sf)
library(cartogram)

nc = st_read(system.file("shape/nc.shp", package="sf"), quiet = TRUE)

# transform to NAD83 / UTM zone 16N
nc_utm <- st_transform(nc, 26916)

# Create cartogram
nc_utm_carto <- cartogram_cont(nc_utm, weight = "BIR74", itermax = 5)

# Plot
par(mfrow=c(2,1))
plot(nc[,"BIR74"], main="original", key.pos = NULL, reset = FALSE)
plot(nc_utm_carto[,"BIR74"], main="distorted", key.pos = NULL, reset = FALSE)

---

Description
Construct a cartogram which represents each geographic region as non-overlapping circles (Dorling 1996).
Usage

cartogram_dorling(x, weight, k = 5, m_weight = 1, itermax = 1000)

## S3 method for class 'sf'
cartogram_dorling(x, weight, k = 5, m_weight = 1, itermax = 1000)

## S3 method for class 'SpatialPolygonsDataFrame'
cartogram_dorling(x, weight, k = 5, m_weight = 1, itermax = 1000)

Arguments

x a polygon or multipolygon sf object
weight Name of the weighting variable in x
k Share of the bounding box of x filled by the larger circle
m_weight Circles’ movements weights. An optional vector of numeric weights (0 to 1 inclusive) to apply to the distance each circle moves during pair-repulsion. A weight of 0 prevents any movement. A weight of 1 gives the default movement distance. A single value can be supplied for uniform weights. A vector with length less than the number of circles will be silently extended by repeating the final value. Any values outside the range [0, 1] will be clamped to 0 or 1.
itermax Maximum iterations for the cartogram transformation.

Value

Non overlapping proportional circles of the same class as x.

References


Examples

library(sf)
library(cartogram)

nc = st_read(system.file("shape/nc.shp", package="sf"), quiet = TRUE)

# transform to NAD83 / UTM zone 16N
nc_utm <- st_transform(nc, 26916)

# Create cartogram
nc_utm_carto <- cartogram_dorling(nc_utm, weight = "BIR74")

# Plot
par(mfrow=c(2,1))
plot(nc[,"BIR74"], main="original", key.pos = NULL, reset = FALSE)
plot(nc_utm_carto[,"BIR74"], main="distorted", key.pos = NULL, reset = FALSE)
**Description**

Construct a non-contiguous area cartogram (Olson 1976).

**Usage**

```r
cartogram_ncont(x, weight, k = 1, inplace = TRUE)
```

```r
## S3 method for class 'SpatialPolygonsDataFrame'
cartogram_ncont(x, weight, k = 1, inplace = TRUE)
```

```r
## S3 method for class 'sf'
cartogram_ncont(x, weight, k = 1, inplace = TRUE)
```

**Arguments**

- `x`: a polygon or multipolygon sf object
- `weight`: Name of the weighting variable in `x`
- `k`: Factor expansion for the unit with the greater value
- `inplace`: If TRUE, each polygon is modified in its original place, if FALSE multi-polygons are centered on their initial centroid

**Value**

An object of the same class as `x` with resized polygon boundaries

**References**


**Examples**

```r
library(sf)
library(cartogram)

nc = st_read(system.file("shape/nc.shp", package="sf"), quiet = TRUE)

# transform to NAD83 / UTM zone 16N
nc_utm <- st_transform(nc, 26916)

# Create cartogram
nc_utm_carto <- cartogram_ncont(nc_utm, weight = "BIR74")

# Plot
par(mfrow=c(2,1))
plot(nc[,"BIR74"], main="original", key.pos = NULL, reset = FALSE)
plot(st_geometry(nc_utm), main="distorted", reset = FALSE)
plot(nc_utm_carto[,"BIR74"], add = TRUE)
Index

cartogram_cont, 2
cartogram_dorling, 3
cartogram_ncont, 5