

Package ‘fasterize’

July 27, 2020

Title Fast Polygon to Raster Conversion

Version 1.0.3

Description Provides a drop-in replacement for rasterize() from the 'raster' package that takes 'sf'-type objects, and is much faster. There is support for the main options provided by the rasterize() function, including setting the field used and background value, and options for aggregating multi-layer rasters. Uses the scan line algorithm attributed to Wylie et al. (1967) <doi:10.1145/1465611.1465619>.

License MIT + file LICENSE

LazyData true

URL <https://github.com/ecohealthalliance/fasterize>

BugReports <https://github.com/ecohealthalliance/fasterize/issues>

RoxygenNote 7.1.1

SystemRequirements C++11

Suggests testthat, microbenchmark, knitr, rmarkdown, sf

Depends R (>= 3.3.0)

Imports methods, Rcpp, raster, sp

LinkingTo Rcpp, RcppArmadillo

Encoding UTF-8

VignetteBuilder knitr

NeedsCompilation yes

Author Noam Ross [aut, cre] (<<https://orcid.org/0000-0002-2136-0000>>),
Michael Sumner [ctb] (<<https://orcid.org/0000-0002-2471-7511>>),
EcoHealth Alliance [cph],
USAID PREDICT [fnd]

Maintainer Noam Ross <ross@ecohealthalliance.org>

Repository CRAN

Date/Publication 2020-07-27 05:10:26 UTC

R topics documented:

fasterize	2
raster,sf-method	3

Index	4
--------------	----------

fasterize	<i>Rasterize an sf object of polygons</i>
-----------	---

Description

Rasterize set of polygons
Fast sf-to-raster conversion

Usage

```
fasterize(
  sf,
  raster,
  field = NULL,
  fun = "last",
  background = NA_real_,
  by = NULL
)
```

Arguments

sf	an <code>sf::sf()</code> object with a geometry column of POLYGON and/or MULTI-POLYGON objects.
raster	A raster object. Used as a template for the raster output. Can be created with <code>raster::raster()</code> . The fasterize package provides a method to create a raster object from an sf object.
field	character. The name of a column in sf, providing a value for each of the polygons rasterized. If NULL (default), all polygons will be given a value of 1.
fun	character. The name of a function by which to combine overlapping polygons. Currently takes "sum", "first", "last", "min", "max", "count", or "any". Future versions may include more functions or the ability to pass custom R/C++ functions. If you need to summarize by a different function, use <code>by=</code> to get a RasterBrick and then <code>raster::stackApply()</code> or <code>raster::calc()</code> to summarize.
background	numeric. Value to put in the cells that are not covered by any of the features of x. Default is NA.
by	character. The name of a column in sf by which to aggregate layers. If set, fasterize will return a RasterBrick with as many layers as unique values of the by column.

Details

This is a high-performance replacement for `raster::rasterize()`.

The algorithm is based on the method described in course materials provided by [Wayne O. Cochran](#).

The algorithm is originally attributed to [Wylie et al. \(1967\)](#).

Value

A raster of the same size, extent, resolution and projection as the provided raster template.

References

Wylie, C., Romney, G., Evans, D., & Erdahl, A. (1967). Half-tone perspective drawings by computer. Proceedings of the November 14-16, 1967, Fall Joint Computer Conference. AFIPS '67 (Fall). <https://dx.doi.org/10.1145/1465611.1465619>

Examples

```
library(sf)
library(fasterize)
p1 <- rbind(c(-180,-20), c(-140,55), c(10, 0), c(-140,-60), c(-180,-20))
hole <- rbind(c(-150,-20), c(-100,-10), c(-110,20), c(-150,-20))
p1 <- list(p1, hole)
p2 <- list(rbind(c(-10,0), c(140,60), c(160,0), c(140,-55), c(-10,0)))
p3 <- list(rbind(c(-125,0), c(0,60), c(40,5), c(15,-45), c(-125,0)))
pols <- st_sf(value = rep(1,3),
              geometry = st_sfc(lapply(list(p1, p2, p3), st_polygon)))
r <- raster(pols, res = 1)
r <- fasterize(pols, r, field = "value", fun="sum")
plot(r)
```

raster,sf-method	<i>Create a raster from an sf object</i>
------------------	--

Description

See `raster::raster()` for more details.

Usage

```
## S4 method for signature 'sf'
raster(x, origin, ...)
```

Arguments

x	an sf object
origin	the origin point of the output raster
...	Additional arguments, see <code>raster::raster()</code> for more details.

Index

`fasterize`, 2

`raster (raster, sf-method)`, 3

`raster, sf-method`, 3

`raster::calc()`, 2

`raster::raster()`, 2, 3

`raster::rasterize()`, 3

`raster::stackApply()`, 2

`sf::sf()`, 2