Package ‘distanceto’

June 1, 2023

Title Calculate Distance to Features
Version 0.0.3
Description Calculates distances from point locations to features.
The usual approach for eg. resource selection function analyses is to
generate a complete distance to features surface then sample it with your
observed and random points. Since these raster based approaches can be
pretty costly with large areas, and often lead to memory issues in R,
the distanceto package opts to compute these distances using
efficient, vector based approaches. As a helper, there's a decidedly
low-res raster based approach for visually inspecting your region's
distance surface. But the workhorse is distance_to.

URL https://github.com/robitalec/distance-to,
https://robitalec.github.io/distance-to/

BugReports https://github.com/robitalec/distance-to/issues
License GPL (>= 3)
Encoding UTF-8
RoxygenNote 7.2.3
Suggests fasterize, knitr, rmarkdown, raster, tinytest, lwgeom
Imports sf, nabor, geodist
VignetteBuilder knitr
NeedsCompilation no
Author Alec L. Robitaille [aut, cre] (<https://orcid.org/0000-0002-4706-1762>)
Maintainer Alec L. Robitaille <robit.alec@gmail.com>
Repository CRAN
Date/Publication 2023-06-01 12:20:03 UTC

R topics documented:

  distanceto ......................................................... 2
distance_raster ..................................................... 2
distance_to .......................................................... 4
The distanceto package is designed to quickly sample distances from points features to other vector layers. Normally the approach for calculating distance to (something) involves generating distance surfaces using raster based approaches eg. `raster::distance` or `gdal_proximity` and subsequently point sampling these surfaces. Since raster based approaches are a costly method that frequently leads to memory issues or long and slow run times with high resolution data or large study sites, we have opted to compute these distances using vector based approaches. As a helper, there’s a decidedly low-res raster based approach for visually inspecting your region’s distance surface. But the workhorse is `distance_to`.

Details

The distanceto package provides two functions:

- `distance_to`
- `distance_raster`

Author(s)

Maintainer: Alec L. Robitaille <robit.alec@gmail.com> (ORCID)

See Also

Useful links:

- [https://github.com/robitalec/distance-to](https://github.com/robitalec/distance-to)
- [https://robitalec.github.io/distance-to/](https://robitalec.github.io/distance-to/)
- Report bugs at [https://github.com/robitalec/distance-to/issues](https://github.com/robitalec/distance-to/issues)

Description

Generates a distance surface from layer y.
Usage

distance_raster(
  y,
  cellsize,
  extent = NULL,
  expand = NULL,
  measure = NULL,
  check = TRUE
)

Arguments

y  feature layer to measure distance to. Expecting an sf point, line or polygon compatible with sf::st_coordinates such as an sf, sfc or sfg object.
cellsize  size, in unit of projection, of pixels of output distance surface.
extent  optional alternative extent bounding box. See details.
extend  0-1 scaling eg. 5% expansion = 0.05. See details.
measure  method used to measure geographic distances. See geodist::geodist for more information. Ignored if CRS of y indicates projected coordinates.
check  default: TRUE. Checks the cellsize against the size of the feature layers y bounding box or optional extent argument.

Details

Calculates the distance of each pixel to the features in layer y. First, generates a regular grid of points in the bounding box of y or optionally provided extent. Then measures the distance from each point to the nearest feature in layer y using distanceto::distance_to(). Finally, returns the grid of distances, rasterized using the excellent package fasterize.

Note: this function is intended to provide a rough, low-res look at your distance surface. The function distanceto::distance_to() should be used for all precise measurements from points to features, as it avoids the costly procedure of generating an entire distance surface by calculating geographic distances directly between points x and features in layer y.

The features in layer y are expected to be an sf object. If the input CRS of features in layer y is longlat, eg. EPSG 4326, the distance is returned as measured by geodist::geodist. Otherwise, if the input CRS indicates projected coordinates, the distance measured is the euclidean distance.

The extent argument can be used to provide an alternative bounding box to generate the distance surface within. This might be useful, for example, if your features in layer y are in a larger area than you require or if you’d like to generate distance surfaces with a specific extent. The expand argument can be used to expand the bounding box calculated for layer y or provided by argument extent. This is just a simple multiplier on the min and max XY of the bounding box to generate a larger surface.

Value

A distance raster surface.
Examples

```r
# Load sf
library(sf)

# Load nc data
nc <- st_read(system.file("shape/nc.shp", package="sf"))

# Select first 5 of nc
ncsub <- nc[1:5,]

# Note: package 'fasterize' required for distance_raster
if (require(fasterize, quietly = TRUE)) {
  # Generate a distance raster from some of nc within extent of all of nc
  distance_raster(ncsub, 0.1, st_bbox(nc), measure = 'geodesic')
}
```

---

distance_to  

Description

Measures the distance from points x to features in layer y.

Usage

```r
distance_to(x, y, measure = NULL)
```

Arguments

- **x**: points to measure distances from, to layer y. Expecting an sf point compatible with sf::st_coordinates such as an sf, sfc or sfg object with geometry type 'POINT' or 'MULTIPOINT'. CRS of x should match CRS of y.
- **y**: feature layer to measure distance to. Expecting an sf point, line or polygon compatible with sf::st_coordinates such as an sf, sfc or sfg object. CRS of y should match CRS of x.
- **measure**: method used to measure geographic distances between longlat x and y objects. See geodist::geodist for more information. Ignored if CRS of x and y indicated projected coordinates.

Details

Uses the function nabor::knn to determine the distance from each point in x to the nearest feature in layer y. If the input CRS is longlat, eg. EPSG 4326, the distance is returned as measured by geodist::geodist. Otherwise, if the input CRS indicates projected coordinates, the distance returned is the euclidean distance. Both x and y are expected to be sf objects and the distances are returned as vector, easily added to input x with $\leftarrow$ or other methods. If y is a 'POLYGON' or 'MULTIPOLYGON' object, the distance returned for points in x within features in y are set to 0.
distance_to

Value
A vector of distances from points in x to features in layer y.

Examples

# Load sf
library(sf)

# Load nc data
nc <- st_read(system.file("shape/nc.shp", package="sf"))

# Set number of sampling points
npts <- 1e3

# Note: package 'lwgeom' required for st_sample
if (require(lwgeom, quietly = TRUE)) {
  # Sample points in nc
  ncpts <- st_sample(nc, npts)

  # Select first 5 of nc
  ncsub <- nc[1:5,]

  # Measure distance from ncpts to first 5 of nc, printing result
distance_to(ncpts, ncsub, measure = 'geodesic')

  # or add to ncpts
  ncpts$dist <- distance_to(ncpts, ncsub, measure = 'geodesic')
}

Index

_PACKAGE (distanceto), 2

distance_raster, 2
distance_to, 4
distanceto, 2
distanceto-package (distanceto), 2