Package ‘grainchanger’

March 8, 2019

Title Moving-Window and Direct Data Aggregation

Version 0.1.0

Description Data aggregation via moving window or direct methods. Aggregate a fine-resolution raster to a grid. The moving window method smooths the surface using a specified function within a moving window of a specified size and shape prior to aggregation. The direct method simply aggregates to the grid using the specified function.

Depends R (>= 3.3)

License GPL-3

Encoding UTF-8

LazyData true

Imports raster, sf, furrr, checkmate, methods

Suggests testthat, spelling, knitr, rmarkdown, covr, ggplot2, landscapetools

RoxygenNote 6.1.0

Language en-GB

VignetteBuilder knitr

URL https://github.com/laurajanegraham/grainchanger

BugReports https://github.com/laurajanegraham/grainchanger/issues

NeedsCompilation no

Author Laura Graham [aut, cre] (<https://orcid.org/0000-0002-3611-7281>), Felix Eigenbrod [ctb] (Input on initial conceptual development), Marco Sciaini [ctb] (Input on package development and structure)

Maintainer Laura Graham <LauraJaneE Graham@gmail.com>

Repository CRAN

Date/Publication 2019-03-08 17:20:05 UTC
R topics documented:

- cat_ls ................................................................. 2
- cont_ls ............................................................... 3
- create_torus ......................................................... 3
- g_sf ................................................................. 4
- nomove_agg ......................................................... 4
- winmove .............................................................. 5
- winmove_agg ......................................................... 7

Index

---

cat_ls

Example categorical raster

Description

An example map to show functionality on categorical surfaces.

Usage

cat_ls

Format

A raster layer object.

Details

Generated with nlm_mpd() from NLMR and classified with util_classify() from landscapetools.

Source


**cont_ls**  

**Example continuous raster**

**Description**

An example map to show functionality on continuous surfaces.

**Usage**

```r
cont_ls
```

**Format**

A raster layer object.

**Details**

Generated with `nlm_mpd()` from NLMR.

**Source**


---

**create_torus**  

**Pad a raster by a specified radius**

**Description**

This function pads a raster by a specified number of cells, creating the effect of a torus. Allows for moving window analysis that avoids edge effects (e.g. on simulated landscapes).

**Usage**

```r
create_torus(dat, r)
```

**Arguments**

- `dat` The raster dataset to pad
- `r` The radius by which to pad the raster

**Value**

raster. Original raster padded by radius r with torus effect
Examples

data(cat_ls)
d = create_torus(dat = cat_ls, r = 5)

---

g_sf

Example grid

Description

An example grid to show functionality when aggregating using an sf object.

Usage

g_sf

Format

An sf object.

Details

Generated with nlm_mpd() and converted to sf.

Source


---

nomove_agg

Direct data aggregation

Description

Calculate the value for a given function for each cell in a larger resolution grid.

Usage

nomove_agg(g, dat, fun, ...)

Arguments

- **g**: the grid across which to calculate the aggregated moving window function (raster, SpatialPolygonsDataFrame, or sf object)
- **dat**: The raster dataset to aggregate
- **fun**: The function to apply. The function fun should take multiple numbers, and return a single number. For example mean, modal, min or max. It should also accept a na.rm argument (or ignore it, e.g. as one of the ‘dots’ arguments. For example, length will fail, but function(x, ...)na.omit(length(x)) works. See Details
- **...**: further arguments passed to or from other methods

Details

`grainchanger` has several built-in functions. Functions currently included are:

- `nm_shei` - Shannon evenness, requires the additional argument `lc_class` (vector or scalar)
- `nm_prop` - Proportion, requires the additional argument `lc_class` (scalar)
- `var_range` - Range (max - min)

Value

Raster (if input is Raster) or numeric vector (if input is sp or sf object) containing values calculated for each coarser cell

Examples

```r
# load required data
data(g_sf)
data(cont_ls)
data(cat_ls)

# aggregate using mean
d = nomove_agg(g_sf, cont_ls, "mean")

# aggregate using Shannon evenness
d = nomove_agg(g_sf, cont_ls, "nm_shei", lc_class = 0:3)
```

Description

Smooth a raster surface using a moving window with a given function, radius and shape.
Usage

```
winmove(dat, d, type, fun, ...)
```

Arguments

dat The raster dataset on which to calculate the moving window function
d numeric. If type=circle, the radius of the circle (in units of the CRS). If
type=rectangle the dimension of the rectangle (one or two numbers). If type=Gauss
the size of sigma, and optionally another number to determine the size of the
matrix returned (default is 3 times sigma)
type The shape of the moving window
fun The function to apply. The function fun should take multiple numbers, and re-
turn a single number. For example mean, modal, min or max. It should also
accept a na.rm argument (or ignore it, e.g. as one of the ‘dots’ arguments. For
example, length will fail, but function(x, ...)na.omit(length(x)) works. See De-
tails
...
... further arguments passed to or from other methods

Details

grainchanger has several built-in functions. Functions currently included are:

- **wm_shei** - Shannon evenness, requires the additional argument lc_class (vector or scalar)
- **wm_prop** - Proportion, requires the additional argument lc_class (scalar)
- **wm_classes** - Unique number of classes in a categorical landscape
- **var_range** - Range (max - min)

Value

A smoothed raster with the moving window values calculated

Examples

```
# load required data
data(cat_ls)
data(cont_ls)

# calculate the moving window mean
d = winmove(cont_ls, 5, "rectangle", "mean")

# calculate the moving window Shannon evenness
d = winmove(cat_ls, 5, "rectangle", "shei", lc_class = 0:3)
```
Description

Calculate the mean moving window value for a given radius, shape and function for each cell in a larger resolution grid.

Usage

\texttt{winmove\_agg(g, dat, d, type, fun, \ldots)}

Arguments

- \texttt{g}: the grid across which to calculate the aggregated moving window function (raster, SpatialPolygonsDataFrame, or sf object)
- \texttt{dat}: The raster dataset to aggregate
- \texttt{d}: numeric. If \texttt{type=circle}, the radius of the circle (in units of the CRS). If \texttt{type=rectangle} the dimension of the rectangle (one or two numbers). If \texttt{type=gauss} the size of sigma, and optionally another number to determine the size of the matrix returned (default is 3 times sigma)
- \texttt{type}: The shape of the moving window
- \texttt{fun}: The function to apply. The function fun should take multiple numbers, and return a single number. For example mean, modal, min or max. It should also accept a na.rm argument (or ignore it, e.g. as one of the ‘dots’ arguments. For example, length will fail, but function(x, \ldots)na.omit(length(x)) works. See Details
- \ldots: further arguments passed to or from other methods

Details

\texttt{grainchanger} has several built-in functions. Functions currently included are:

- \texttt{wm\_shei} - Shannon evenness, requires the additional argument \texttt{lc\_class} (vector or scalar)
- \texttt{wm\_prop} - Proportion, requires the additional argument \texttt{lc\_class} (scalar)
- \texttt{wm\_classes} - Unique number of classes in a categorical landscape
- \texttt{var\_range} - Range (max - min)

Value

Numeric vector containing moving window values calculated for each grid cell
Examples

```r
# load required data
data(g_sf)
data(cont_ls)
data(cat_ls)

#' aggregate using mean
d = winmove_agg(g_sf, cont_ls, 5, "rectangle", "mean")

# aggregate using Shannon evenness
d = winmove_agg(g_sf, cat_ls, 5, "rectangle", "shei", lc_class = 0:3)
```
Index

*Topic aggregate
  nomove_agg, 4
  winmove_agg, 7

*Topic datasets
  cat_ls, 2
  cont_ls, 3
  g_sf, 4

*Topic focal,
  winmove, 5
  winmove_agg, 7

*Topic raster
  create_torus, 3

*Topic spatial,
  nomove_agg, 4
  winmove_agg, 7

*Topic spatial
  winmove, 5

*Topic torus,
  create_torus, 3

  cat_ls, 2
  cont_ls, 3
  create_torus, 3

  g_sf, 4

  nomove_agg, 4

  winmove, 5
  winmove_agg, 7