

Package ‘tabularaster’

December 10, 2017

Type Package

Title Tidy Tools for 'Raster' Data

Version 0.4.0

Description Facilities to work with vector and raster data in efficient repeatable and systematic work flow. Missing functionality in existing packages is included here to allow extraction from raster data with 'simple features' and 'Spatial' types and to make extraction consistent and straightforward. Extract cell numbers from raster data and return the cells, values and weights as a data frame rather than as lists of matrices or vectors. The functions here allow spatial data to be used without special handling for the format currently in use.

License GPL-3

LazyData TRUE

Depends R (>= 3.2.5)

Imports dplyr, methods, raster, sp, spex (>= 0.4.0), spbabel (>= 0.4.8), tibble, viridis

RoxygenNote 6.0.1

Suggests covr, testthat, knitr, rmarkdown, sf, spdplyr

VignetteBuilder knitr

URL <https://github.com/hypertidy/tabularaster>

BugReports <https://github.com/hypertidy/tabularaster/issues>

NeedsCompilation no

Author Michael D. Sumner [aut, cre]

Maintainer Michael D. Sumner <mdsumner@gmail.com>

Repository CRAN

Date/Publication 2017-12-10 21:36:39 UTC

R topics documented:

as_tibble	2
bufext	3
cellnumbers	4
extentFromCells	5
ghrsst	5
index_extent	6
oisst	6
polycano	7
rastercano	7
sharkcano	7
tabularaster	8

Index	9
--------------	----------

as_tibble	<i>Convert a Raster to a data frame.</i>
-----------	--

Description

Generate a data frame version of any raster object. Use the arguments 'cell', 'dim', 'split_date' and 'value' to control the columns that are included in the output.

Usage

```
## S3 method for class 'BasicRaster'
as_tibble(x, cell = TRUE, dim = nlayers(x) > 1L,
  value = TRUE, split_date = FALSE, xy = FALSE, ...)
```

Arguments

x	a RasterLayer, RasterStack or RasterBrick
cell	logical to include explicit cell number
dim	logical to include slice index
value	logical to return the values as a column or not
split_date	logical to split date into components
xy	logical to include the x and y centre coordinate of each cell
...	unused

Details

If the raster has only one layer, the slice index is not added. Use 'dim = FALSE' to not include the slice index value.

Value

a data frame ('tbl_df'/'tibble' form)

Examples

```
#library(tabularaster)
#library(tibble)
#as_tibble(raster::raster(volcano))
#as_tibble(raster::setZ(raster::raster(volcano), Sys.Date()), cell = TRUE)
```

bufext

Defunct tabularaster function

Description

Ensure a raster extent aligns to whole parts.

Usage

```
bufext(e1, e2)
```

Arguments

e1	input extent
e2	grain size

Details

This function is defunct, please use `spex::buffer_extent`.

Examples

```
library(spex)
library(raster)
buffer_extent(extent(0.1, 2.2, 0, 3), 2)
```

 cellnumbers

Extract cell numbers from a Raster object.

Description

Provide the 'cellnumbers' capability of `raster::extract` and friends directly, returning a data frame of query-object identifiers 'object_' and the cell number.

Usage

```
cellnumbers(x, query, ...)
```

Arguments

x	Raster object
query	Spatial object or matrix of coordinates
...	arguments passed on to <code>raster::cellFromPolygon</code> for polygon input

Details

Raster data is inherently 2-dimensional, with a time or 'level' dimension treated like a layers of these 2D forms. The 'raster' package cell number is counted from 1 at the top-left, across the rows and down. This corresponds the the standard "raster graphics" convention used by 'GDAL' and the 'sp' package, and many other implementations. Note that this is different to the convention used by the `graphics::image` function.

Currently this function only operates as if the input is a single layer objects, it's not clear if adding an extra level of grouping for layers would be sensible.

The dots argument can be used to set `weights=TRUE` for the polygon case, this is otherwise ignored.

Value

tbl_df data frame

Examples

```
library(raster)
library(dplyr)
r <- raster(volcano) %>% aggregate(factor = 4)
cellnumbers(r, rasterToContour(r, level = 120))
#library(dplyr)

#cr <- cut(r, pretty(values(r)))
#p <- raster::rasterToPolygons(cr, dissolve = TRUE)
#p <- spex::qm_rasterToPolygons_sp(cr)
#tt <- cellnumbers(cr, p)
#library(dplyr)
#tt %>% mutate(v = extract(r, cell_)) %>%
```

```
#group_by(object_) %>%
#summarize(mean(v))
#head(pretty(values(r)), -1)
```

extentFromCells	<i>Extent from cells</i>
-----------------	--------------------------

Description

Use cell numbers to obtain their extent from a raster.

Usage

```
extentFromCells(x, cells)
```

Arguments

x	raster
cells	cell numbers, relative to x

Details

This is a missing function in the raster package for cell abstraction.

Value

raster extent

ghrsst	<i>Sea surface temperature data.</i>
--------	--------------------------------------

Description

A smoothed subset of GHRSSST.

Format

A raster created GHRSSST data and raster smoothing.

Details

See "data-raw/ghrsst.R" and "data-raw/ghrsst-readme.txt" for details.
 sst_regions is a simple polygon region layer to sit over the SST data.

Examples

```
library(raster)
plot(ghrsst, col = viridis::viridis(100))
plot(sst_regions, add = TRUE, col = NA)
## cellnumbers(ghrsst, sst_regions)
```

index_extent	<i>Index extent</i>
--------------	---------------------

Description

Extent in index space.

Usage

```
index_extent(x, ex)
```

Arguments

x	raster layer
ex	extent

Details

Convert a geographic extent into purely index space.

Value

extent object

oisst	<i>Optimally interpolated SST in near-native form.</i>
-------	--

Description

See data-raw/oisst.R in the source repository. The file was avhrr-only-v2.20170729.nc, its extent -180, 180, -90, 90 with dimensions 1440x720 in the usual raster configuration.

Format

A data frame of sst values created from OISST data.

polycano	<i>The raster volcano as polygons.</i>
----------	--

Description

See data-raw/rastercano.r in the source repository.

Format

A `sp::SpatialPolygonsDataFrame` with variables: `volcano_elevation`.

rastercano	<i>The raster volcano.</i>
------------	----------------------------

Description

See data-raw/rastercano.r in the source repository.

Format

A raster created from the [volcano](#) data.

sharkcano	<i>Sharkcano, the shark and the volcano.</i>
-----------	--

Description

This is just a free image off the internetz. The image was read in and all non-essential items dropped. The dimensions in `raster::raster` terms is stored in `attr(sharkcano, "rasterdim")`.

Format

A data frame with 117843 rows and 2 variables:

`cell_` integer, cell index

`byte` integer, byte value of shark image pixels

These are cell values on a grid that is 648x958.

References

This is the small version from here, see script in data-raw/sharkcano.r <http://www.freestockphotos.biz/stockphoto/16214>
Thanks to @jennybc for pointers on finding free stuff: <https://github.com/jennybc/free-photos>

Examples

```

library(raster)
rd <- attr(sharkcano, "rasterdim")
rastershark <- raster(matrix(NA_integer_, rd[1], rd[2]))
rastershark[sharkcano$cell_] <- sharkcano$byte ## byte, heh
## I present to you, Sharkcano! (Just wait for the 3D version, Quadshark).
#plot(rastercano)
#contour(rastershark, add = TRUE, labels = FALSE)
#plot(rastershark, col = "black")
## another way
#plot(rastercano)
#points(xyFromCell(rastershark, sharkcano$cell_), pch = ".")

```

tabularaster

Tabular tools for raster, tidy tools for raster.

Description

Tabular tools for raster, tidy tools for raster.

Details

Tabularaster includes these main functions.

<code>as_tibble</code>	<code>as_tibble</code>	convert raster data to data frame form, with control over output and form of dimension/coordinate
<code>cellnumbers</code>		extract a data frame of query identifiers and cell,pixel index numbers
<code>extentFromCells</code>		build an Extent object from cell numbers
<code>index_extent</code>		build an extent in row column form, as opposed to coordinate value form

Index

`as_tibble`, 2, 8

`bufext`, 3

`cellnumbers`, 4, 8

`extent`, 3

`extentFromCells`, 5, 8

`ghrsst`, 5

`graphics::image`, 4

`index_extent`, 6, 8

`oisst`, 6

`polycano`, 7

`raster::cellFromPolygon`, 4

`raster::extract`, 4

`rastercano`, 7

`sharkcano`, 7

`sst_regions` (`ghrsst`), 5

`tabularaster`, 8

`tabularaster-package` (`tabularaster`), 8

`volcano`, 7