

Package ‘csquares’

June 24, 2024

Title Concise Spatial Query and Representation System (c-Squares)

Version 0.0.3

Author Pepijn de Vries [aut, cre] (<<https://orcid.org/0000-0002-7961-6646>>)

Description Encode and decode c-squares, from and to simple feature (sf) or spatiotemporal arrays (stars) objects. Use c-squares codes to quickly join or query spatial data.

Imports dplyr, methods, purrr, rlang, sf, stars, stringr, tidyr

Suggests ggplot2

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.3

Depends R (>= 4.1.0)

LazyData true

Collate 'helpers.R' 'as_csquares.R' 'as_stars.R' 'csquares-package.r'
'csquares_methods.R' 'init.R' 'new_csquares.R' 'orca.R'
'st_as_sf.R' 'summarise.R' 'tidyverse.R' 'validate.R'

NeedsCompilation no

Maintainer Pepijn de Vries <pepijn.devries@outlook.com>

Repository CRAN

Date/Publication 2024-06-24 15:50:09 UTC

Contents

as_csquares	2
format.csquares	3
new_csquares	4
orca	4
st_as_sf	5
st_as_stars.csquares	6
summarise	7
validate_csquares	8

as_csquares	<i>Convert lon-lat coordinates into c-square codes</i>
-------------	--

Description

Takes WGS84 longitude and latitude coordinates and finds the closest matching c-squares for a given resolution.

Usage

```
as_csquares(x, resolution, csquares, ...)

## Default S3 method:
as_csquares(x, resolution, csquares, ...)

## S3 method for class 'character'
as_csquares(x, resolution, csquares, validate = TRUE, ...)

## S3 method for class 'numeric'
as_csquares(x, resolution = 1, csquares, ...)

## S3 method for class 'data.frame'
as_csquares(x, resolution = 1, csquares, ...)

## S3 method for class 'sf'
as_csquares(x, resolution = 1, csquares, ...)

## S3 method for class 'stars'
as_csquares(x, resolution = 1, csquares, ...)
```

Arguments

x	An object to be coerced to a csquares object. x can be a vector of character strings representing c-squares code. It can also be a numeric matrix with two columns containing the x and y coordinates. x can also be a simple features object (sf) or a spatial arrays object (stars).
resolution	Resolution (in WGS84 degrees) to be used for creating c-squares codes. As per c-square specifications, the resolution should be 10 or less, yet greater than 0. It should be a tenfold of 1 or 5. Valid resolutions are therefore: 10, 5, 1, 0.5, 0.1, etc.
csquares	If x is not a vector of character strings (but for instance a data.frame), the csquares argument should specify the name of the element of x containing the c-square codes as character strings.
...	Currently ignored

`validate` A logical value indicating whether the created object needs to be validated. Defaults to TRUE. Validation can be time-consuming so set to FALSE to save computing time.

Value

Returns a csquares object that contains c-squares codes.

Author(s)

Pepijn de Vries

Examples

```
as_csquares(cbind(x = 5.2399066, y = 52.7155812), resolution = 1)
as_csquares(orca, csquares = "csquares")
```

format.csquares *Basic csquares methods*

Description

Basic methods for csquares objects for formatting and printing the objects

Usage

```
## S3 method for class 'csquares'
format(x, ...)

show(x, ...)

## S3 method for class 'csquares'
show(x, ...)

## S3 method for class 'csquares'
print(x, ...)
```

Arguments

`x` A csquares object to be handled by the s3 methods
`...` Ignored

Value

Returns (a formatted version of) `x`

new_csquares *Create a c-squares raster from a bounding box*

Description

Creates a spatial raster ([stars](#)) with c-square codes for a specified bounding box, using a specified resolution. The raster will be conform c-squares specifications.

Usage

```
new_csquares(x, resolution = 1, crs = 4326)
```

Arguments

x	An object of class bbox or an object that can be coerced to a bbox. It defines the bounding box for the c-squares grid created by this function.
resolution	Resolution (in WGS84 degrees) to be used for creating c-squares codes. As per c-square specifications, the resolution should be 10 or less, yet greater than 0. It should be a tenfold of 1 or 5. Valid resolutions are therefore: 10, 5, 1, 0.5, 0.1, etc.
crs	The projection to be used for the created grid. By default it is WGS84 (EPSG:4326).

Value

Returns a [stars](#) and csquares object based on the provided bounding box and resolution.

Author(s)

Pepijn de Vries

Examples

```
library(sf)
nc <- st_read(system.file("shape/nc.shp", package = "sf"))
new_csquares(nc)
```

orca *Killer whale realm*

Description

Killer whale realm

Usage

```
orca
```

Format

orca:

The orca object is a Killer whale realm data set extracted from the data as provided by Costello (2017) and published by Costello *et al.* (2017). It is a data frame with 2,058 rows and two columns:

csquares c-squares codes indicating spatial grid cells

orcinus_orca logical values indicating whether the corresponding c-squares grid cell belongs to the killer whales (*Orcinus orca*) biogeographic realm or not.

References

- Costello, M.J. (2017); University of Auckland [doi:10.17608/k6.auckland.5086654](https://doi.org/10.17608/k6.auckland.5086654) Licence [CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)
- Costello M.J., Tsai P., Wong P.S., Cheung A.K.L, Basher Z. & Chaudhary C. (2017); "Marine biogeographic realms and species endemism" Nature Communications 8, 1057 [doi:10.1038/s41467017011212](https://doi.org/10.1038/s41467017011212)

 st_as_sf

Create a simple features object from c-squares

Description

Converts a character string of c-squares in a spatially explicit simple features object (*sf*). It can also convert data.frames with a column of c-squares codes to an *sf* object.

Usage

```
st_as_sf.csquares(x, csquares = "csquares")
```

```
st_as_sf.c.csquares(x)
```

Arguments

x	A vector of character strings. Each element should hold a valid c-square code. x can also be a data.frame with a column of c-square codes. (Note that wildcard characters are not supported)
csquares	In case x is a data.frame, csquare should specify the column name that holds the c-square codes.

Value

In case of `st_as_sf.c.csquares` a list of geometries (*sfc*, (MULTI)POLYGONS) is returned. In case of `st_as_sf.csquares` an object of class (*sf*) is returned.

Author(s)

Pepijn de Vries

Examples

```
library(sf)
st_as_sfc(as_csquares("7500:110:3|7500:110:1|1500:110:3|1500:110:1"))
st_as_sf(as_csquares("7500:110:3|7500:110:1|1500:110:3|1500:110:1"))
```

st_as_stars.csquares *Coerce csquares object into a stars object*

Description

Take a csquares object created with [new_csquares](#) or [as_csquares](#) and coerce it to a spatiotemporal array ([stars](#)).

Usage

```
st_as_stars.csquares(x, ...)
```

Arguments

x	An object of class csquares created with new_csquares or as_csquares
...	ignored.

Value

Returns a spatiotemporal array ([stars](#)) object based on x.

Author(s)

Pepijn de Vries

Examples

```
library(stars)
st_as_stars(as_csquares("7500:110:3|7500:110:1|1500:110:3|1500:110:1"))
st_as_stars(as_csquares(orca, csquares = "csquares"))
```

summarise	<i>Summarise c-square data to a lower resolution</i>
-----------	--

Description

This function acts very much like `dplyr::summarise`, but instead of using a column to group, c-square codes are used to aggregate to a lower resolution and summarise the data to that lower resolution.

Usage

```
summarise.csquares(x, ..., .by, tiers_down = 1L)
```

Arguments

<code>x</code>	The object to be summarised. Can be a <code>data.frame</code> , <code>sf</code> , or <code>stars</code> object.
<code>...</code>	<p><data-masking> Name-value pairs of summary functions. The name will be the name of the variable in the result.</p> <p>The value can be:</p> <ul style="list-style-type: none"> • A vector of length 1, e.g. <code>min(x)</code>, <code>n()</code>, or <code>sum(is.na(y))</code>. • A data frame, to add multiple columns from a single expression. <p>[Deprecated] Returning values with size 0 or >1 was deprecated as of 1.1.0. Please use <code>reframe()</code> for this instead.</p>
<code>.by</code>	The column name that holds the c-squares codes that need to be aggregated.
<code>tiers_down</code>	The number of tiers down from the current resolution to which you wish to summarise. If the current resolution is 5x5 degrees, the tier down would be 10x10 degrees (as is the case in the example below).

Value

Returns the summarised object inheriting its class from `x`

Author(s)

Pepijn de Vries

Examples

```
library(dplyr)
orca |>
  as_csquares(csquares = "csquares") |>
  summarise(
    .by = "csquares",
    orcinus_orca = any(na.omit(.data$orcinus_orca)))
```

validate_csquares *Test if a csquares object is valid*

Description

Tests if a csquares object is correctly specified and can be translated into valid coordinates

Usage

```
validate_csquares(x)
```

Arguments

x An object of class csquares to be evaluated.

Value

Returns a logical value indicating whether the csquares object is valid or not.

Author(s)

Pepijn de Vries

Examples

```
validate_csquares(  
  as_csquares("7500:110:3|7500:110:1|1500:110:3|1500:110:1")  
)
```


Index

* datasets

orca, 4

as_csquares, 2, 6

bbox, 4

dplyr::summarise, 7

format.csquares, 3

new_csquares, 4, 6

orca, 4

print.csquares (format.csquares), 3

reframe(), 7

sf, 2, 5, 7

sfc, 5

show (format.csquares), 3

st_as_sf, 5

st_as_sfc (st_as_sf), 5

st_as_stars.csquares, 6

stars, 2, 4, 6, 7

summarise, 7

validate_csquares, 8