Package ‘jpgrid’

March 26, 2023

Type Package

Title Functions for the Grid Square Codes in Japan

Version 0.3.1

Description Provides functions for grid square codes in Japan

License MIT + file LICENSE

URL https://github.com/UchidaMizuki/jpgrid,
https://uchidamizuki.github.io/jpgrid/

BugReports https://github.com/UchidaMizuki/jpgrid/issues

Depends R (>= 4.1.0)

Imports dplyr (>= 0.8.0), geosphere, purrr (>= 1.0.0), rlang (>= 0.3.0), stars, sf, stringr (>= 1.4.0), tibble, tidyr (>= 1.0.0), units, vctrs, lifecycle, pillar, tidyselect, cli

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

NeedsCompilation no

Author Mizuki Uchida [aut, cre]

Maintainer Mizuki Uchida <uchidamizuki@vivaldi.net>

Repository CRAN

Date/Publication 2023-03-26 13:00:01 UTC
as_ttbl_grid

Convert a data frame into a tbl_grid object

Description

[Deprecated]

Usage

as_ttbl_grid(x, var = NULL, grid_size = NULL, strict = TRUE, ...)

Arguments

x An object to be converted into an object class tbl_grid.
var A variable to specify the grid object. By default, the first column of the grid object is taken.
grid_size A grid size.
strict A logical scalar. Should the number of digits in the grid square code match a given number of digits?
... Additional arguments passed to tibble::new_tibble().
bbox_to_grid

Details
It is recommended to use grid_as_sf().

The tbl_grid object is a data frame with grid objects in the columns. as_tbl_grid converts a data frame into a tbl_grid object.

Value
A tbl_grid object.

bbox_to_grid  Converting bbox to grid square codes

Description
Converting bbox to grid square codes

Usage
bbox_to_grid(bbox, grid_size)

Arguments
bbox  A bbox.
grid_size  A grid size.

Value
A grid vector.

coords  Conversion between grid square codes and coordinates (longitude and latitude)

Description
Conversion between grid square codes and coordinates (longitude and latitude)

Usage
coords_to_grid(X, Y, grid_size)
grid_to_coords(grid, center = TRUE)
Arguments

X  A numeric vector of longitude.
Y  A numeric vector of latitude.
grid_size  A grid size.
grid  A grid class vector.
center  Should the center point of the grid be returned? Otherwise the end points will be returned. TRUE by default.

Value

coords_to_grid() returns a grid vector.
grid_to_coords() returns a tbl_df.

geometry_to_grid  Converting sf geometries to grid square codes

Description

Converting sf geometries to grid square codes

Usage

geometry_to_grid(geometry, grid_size, options = "ALL_TOUCHED=TRUE", ...)

Arguments

geometry  A sf vector.
grid_size  A grid size.
options  Options vector for GDALRasterize passed on to stars::st_rasterize().
...  Passed on to stars::st_rasterize().

Value

A list of grid vectors.
grid_as_sf

Converting data frame containing grid square codes to sf

Description

Converting data frame containing grid square codes to sf

Usage

grid_as_sf(
  x,
  as_points = FALSE,
  crs = sf::NA_crs_,
  grid_column_name = NULL,
  ...
)

Arguments

x A data frame or a grid.
as_points Return the center points of the grids or not?
crs Coordinate reference system.
grid_column_name A scalar character.
... passed on to sf::st_as_sf().

Value

A sf object.

grid_as_stars

Converting data frame containing regional grids to stars

Description

Converting data frame containing regional grids to stars

Usage

grid_as_stars(
  x,
  coords = NULL,
  crs = sf::NA_crs_,
  grid_column_name = NULL,
  ...
)

Arguments

- **x**: A data frame or a grid.
- **coords**: The column names or indices that form the cube dimensions.
- **crs**: Coordinate reference system.
- **grid_column_name**: A scalar character.
- **...**: Passed on to `stars::st_as_stars()`.

Value

A `stars` object.

---

**grid_city**

*List of grid square codes by Japanese municipalities*

Description

List of grid square codes by Japanese municipalities

Usage

`grid_city`

Format

An object of class `tbl_df` (inherits from `tbl.data.frame`) with 462915 rows and 6 columns.

Source

[https://www.stat.go.jp/data/mesh/m_itiran.html](https://www.stat.go.jp/data/mesh/m_itiran.html)

---

**grid_class**

*Grid square code vector*

Description

[Deprecated]
Usage

grid_80km(x, strict = TRUE)
grid_10km(x, strict = TRUE)
grid_1km(x, strict = TRUE)
grid_500m(x, strict = TRUE)
grid_250m(x, strict = TRUE)
grid_125m(x, strict = TRUE)
grid_100m(x, strict = TRUE)
grid_auto(x, strict = TRUE)

Arguments

x        A list or vector.
strict   A logical scalar. Should the number of digits in the grid square code match a
given number of digits?

Details

It is recommended to use grid_parse() or grid_convert().
A series of functions return grid class for each grid size. grid_auto() returns automatically
determine grid size by the largest grid size.

Value

A grid vector.

grid_convert  Convert the grid size of grid objects

Description

Convert the grid size of grid objects

Usage

grid_convert(grid, grid_size)
grid_distance

Arguments

- **grid**: A grid vector.
- **grid_size**: A grid size.

Value

A grid vector.

Examples

```r
grid_500m <- parse_grid(c("533945263", "533935863", "533945764"), "500m")
grid_convert(grid_500m, "10km")
```

grid_distance

*Distance between grid square codes*

Description

If `grid` and `grid_to` are both vectors, the distance between `grid` and `grid_to` is calculated. If `grid` is a list, the path distance of each element is calculated.

Usage

```r
grid_distance(
  grid,
  grid_to = NULL,
  close = FALSE,
  type = c("keep_na", "ignore_na", "skip_na")
)
```

Arguments

- **grid**: A grid vector or a list of grid vector.
- **grid_to**: A grid vector.
- **close**: Should the path of each element be closed when `grid` is a list?
- **type**: How is the NA grid treated when `grid` is a list? "skip_na" skips the NA grid and connects the paths. "keep_na" by default.

Value

A double vector.
**grid_line**

*Draw line segments between grid square codes*

**Description**

If `grid` and `grid_to` are both vectors, the line between `grid` and `grid_to` is drawn (using Bresenham’s line algorithm). If `grid` is a list, The path lines for each element in the grid will be drawn.

**Usage**

```r
grid_line(grid, grid_to = NULL, close = FALSE, skip_na = FALSE)
```

**Arguments**

- `grid` A grid vector or a list of grid vector.
- `grid_to` A grid vector.
- `close` Should the path of each element be closed when `grid` is a list?
- `skip_na` Should skip the `NA` grid and connects the paths? FALSE by default.

**Value**

A list of grid vectors.

---

**grid_move**

*Moving on grid square codes*

**Description**

Moving on grid square codes

**Usage**

```r
grid_move(grid, n_X, n_Y)
```

**Arguments**

- `grid` A grid vector.
- `n_X` Number of moving cells in the longitude direction.
- `n_Y` Number of moving cells in the latitude direction.

**Value**

A grid vector.
grid_neighbor  Neighborhood grid square codes

Description
Neighborhood grid square codes

Usage
grid_neighbor(grid, n = 1L, moore = TRUE, simplify = TRUE)

Arguments
- grid: A grid vector.
- n: A numeric vector of degrees.
- moore: Moore neighborhood (TRUE) or Von Neumann neighborhood (FALSE).
- simplify: Should simplify the format of the return?

Value
A list of grid vectors.

grid_subdivide  Subdivide grid square codes

Description
grid_subdivide() makes the grid square codes finer.

Usage
grid_subdivide(grid, grid_size)

Arguments
- grid: A grid vector.
- grid_size: A grid size.

Value
A list of grid vector.
**is_grid**  
*Test if the object is a grid*

**Description**

Test if the object is a grid

**Usage**

```r
is_grid(x, grid_size = NULL)
```

**Arguments**

- `x`: An object.
- `grid_size`: A grid size.

**Value**

TRUE if the object inherits from the grid class.

---

**jpgrid**  
*Functions for the Grid Square Codes in Japan*

**Description**

Provides functions for grid square codes in Japan ([https://www.stat.go.jp/english/data/mesh/index.html](https://www.stat.go.jp/english/data/mesh/index.html)). Generates the grid square codes from longitude/latitude, geometries, and the grid square codes of different scales, and vice versa.

**Author(s)**

**Maintainer**: Mizuki Uchida <uchidamizuki@vivaldi.net>

**See Also**

### parse_grid

*Parse grid square codes*

**Description**

Parse grid square codes

**Usage**

```r
parse_grid(x, grid_size = NULL, strict = TRUE)
```

**Arguments**

- **x**: A character vector of grid square codes.
- **grid_size**: A grid size.
- **strict**: A logical scalar. Should the number of digits in the grid square code match a given number of digits? By default, TRUE.

**Examples**

```r
parse_grid("53394526313")
parsed_grid("53394526313", "80km")
parsed_grid("53394526313", "80km", strict = FALSE)
```

### XY

*Conversion between grid square codes and coordinates (longitude and latitude)*

**Description**

[Deprecated]

**Usage**

```r
grid_to_XY(grid, center = TRUE)
 XY_to_grid(X, Y, grid_size)
```
Arguments

- **grid**: A grid class vector.
- **center**: Should the center point of the grid be returned? Otherwise the end points will be returned. TRUE by default.
- **X**: A numeric vector of longitude.
- **Y**: A numeric vector of latitude.
- **grid_size**: A grid size.

Value

- `grid_to_XY()` returns a tbl_df.
- `XY_to_grid()` returns a grid vector.
Index

* datasets
  grid_city, 6
  as_tbl_grid, 2
  bbox_to_grid, 3
  coords, 3
  coords_to_grid(coords), 3
  geometry_to_grid, 4
  grid_100m(grid_class), 6
  grid_10km(grid_class), 6
  grid_125m(grid_class), 6
  grid_1km(grid_class), 6
  grid_250m(grid_class), 6
  grid_500m(grid_class), 6
  grid_80km(grid_class), 6
  grid_as_sf, 5
  grid_as_stars, 5
  grid_auto(grid_class), 6
  grid_city, 6
  grid_class, 6
  grid_convert, 7
  grid_distance, 8
  grid_line, 9
  grid_move, 9
  grid_neighbor, 10
  grid_subdivide, 10
  grid_to_coords(coords), 3
  grid_to_XY(XY), 12
  is_grid, 11
  jpgrid, 11
  jpgrid-package (jpgrid), 11
  parse_grid, 12
  sf::st_as_sf(), 5
  stars::st_as_stars(), 6
  stars::st_rasterize(), 4
  tibble::new_tibble(), 2
  XY, 12
  XY_to_grid(XY), 12