Package ‘tabs’

April 3, 2025

Title Temporal Altitudinal Biogeographic Shifts

Version 0.1.0

Description A standardized workflow to reconstruct spatial configurations of altitude-bounded biogeographic systems over time. For example, 'tabs' can model how island archipelagos expand or contract with changing sea levels or how alpine biomes shift in response to tree line movements. It provides functionality to account for various geophysical processes such as crustal deformation and other tectonic changes, allowing for a more accurate representation of biogeographic system dynamics. For more information see De Groeve et al. (2025) <doi:10.3897/arphapreprints.e151900>.

License GPL (>= 3)

URL `https://uva_ibed_piac.gitlab.io/tabs/`

BugReports `https://gitlab.com/uva_ibed_piac/tabs/-/issues`

Depends R (>= 2.10)

Imports ape, dplyr, htmltools, htmlwidgets, httr, jsonlite, geojsonio, leaflet, leaftime, leaflet.extras, mapedit, qs2, rlang, sf, stringi, terra, gpkg, RSQLite

Encoding UTF-8

Language en-US

LazyData true

RoxygenNote 7.3.2

NeedsCompilation no

Author Johannes De Groeve [aut, cre] (<`https://orcid.org/0000-0002-1274-3237`>),
Sietze Norder [aut] (<`https://orcid.org/0000-0003-4692-4543`>),
Eline Sterre Rentier [aut] (<`https://orcid.org/0000-0001-6234-1300`>),
Suzette Flantua [ctb] (<`https://orcid.org/0000-0001-6526-3037`>),
Kenneth Rijsdijk [ctb] (<`https://orcid.org/0000-0002-0943-2577`>)

Maintainer Johannes De Groeve <`j.degroeve@uva.nl`>

Repository CRAN

Date/Publication 2025-04-03 15:30:02 UTC

Contents

`bintanja` *Dataset: Bintanja's Global sea level curve (0-3000000 BP)*

Description

Global sea level curve of Bintanja & van de Wal (2008) from 0 to 3000000 BP.

Usage

```
bintanja
```

Format

A vector with:

year_before_after_present years before or after present

sea_level_m meters below or above the reference sea level expressed in m.

Source

Bintanja, R., van de Wal, R. (2008) North American ice-sheet dynamics and the onset of 100,000-year glacial cycles. Nature 454, 869–872 (2008). doi:10.1038/nature07158

Examples

```
curve <- bintanja
```

`cutler` *Dataset: Cutler's Global sea level curve (0-140000 BP)*

Description

Global sea level curve of Cutler et al. (2003) from 0 to 140000 BP.

Usage

```
cutler
```

Format

A vector with:

year_before_after_present years before or after present

sea_level_m meters below or above the reference sea level expressed in m.

Source

Cutler, Kirsten B; Edwards, Ross L; Taylor, Frederick W; Cheng, H; Adkins, Jess F; Gallup, Christina D; Cutler, P M; Burr, George S; Bloom, Arthur L (2003): Rapid sea-level fall and deep-ocean temperature change since the last interglacial period. Earth and Planetary Science Letters, 206(3-4), 253-271. doi:10.1016/S0012821X(02)01107X

Examples

```
curve <- cutler
```

`explore` *Leaflet map of present and paleo configurations for biogeographic systems*

Description

generate a timelapse or exploration visualisation for a reconstructed biomes

Usage

```
explore(x, timelapse = NULL, filename = NULL)
```

Arguments

`x`	tabs. Object of class tabs, after running the reconstruct-function.
`timelapse`	integer, specifies the speed of the html-animation, the higher the number the slower the animation
`filename`	name of the file to save

Details

explore

Value

html file

Examples

```
# load data samples
sporades <- sporades()
topo <- sporades$topo
labs <- sporades$labs
correction <- sporades$correction
curve <- sporades$curve

# subset first and last period
curve <- curve[[c(1,dim(curve)[3])]]

# reconstruct
rec <- reconstruct(topo=topo,
                   region=labs,
                   curve=curve,
                   correction=correction,
                   reclabs='name'
                   )

# metadata
md <- metadata(rec)


# export
dir <- tempdir() # export to temporary directory
export(rec, paste0(dir,'/sporades.qs2'),overwrite=TRUE) # qs2 (faster and less storage than rds)
export(rec, paste0(dir,'/sporades.rds'), overwrite=TRUE) # rds / rdata
export(rec, paste0(dir,'/sporades'), overwrite=TRUE)  # datasets organized in directory
#export(rec, paste0(dir,'/sporades.zip')) # zipped datasets


# import
rec <- import(filename=paste0(dir,'/sporades.qs2'))
rec <- import(filename=paste0(dir,'/sporades.rds'))
rec <- import(filename=paste0(dir,'/sporades'))
```

```
# explore

## tabs class object
if (interactive()){
explore(rec) # comparison present reference and topo-based
explore(rec, timelapse=3) # paleo reconstruction
}

## from exported object
if (interactive()){
explore(paste0(dir,'/sporades.qs2'))
explore(paste0(dir,'/sporades.qs2'), timelapse=1)
}

# get area
area <- get_area(rec) # using object class tabs
area <- get_area(rec$recvect) # using recvect

area <- get_area(paste0(dir,'/sporades.qs2')) # using exported object
unlink(dir)
```

`export` *Export a reconstruction of class tabs in various formats*

Description

export data

Usage

```
export(x, filename, overwrite = FALSE)
```

Arguments

`x`	tabs. Object of class tabs, after running the reconstruct-function.
`filename`	character. Path where files will be exported. Default as directory tree. Use .qs2, .rds, .zip to save as qs2, rds or zipped directory tree.
`overwrite`	boolean. Whether to overwrite the output when filename is specified.

Value

No return value, called for side effects

Author(s)

Johannes De Groeve

Examples

```
# load data samples
sporades <- sporades()
topo <- sporades$topo
labs <- sporades$labs
correction <- sporades$correction
curve <- sporades$curve

# subset first and last period
curve <- curve[[c(1,dim(curve)[3])]]

# reconstruct
rec <- reconstruct(topo=topo,
                   region=labs,
                   curve=curve,
                   correction=correction,
                   reclabs='name'
                   )

# metadata
md <- metadata(rec)


# export
dir <- tempdir() # export to temporary directory
export(rec, paste0(dir,'/sporades.qs2'),overwrite=TRUE) # qs2 (faster and less storage than rds)
export(rec, paste0(dir,'/sporades.rds'), overwrite=TRUE) # rds / rdata
export(rec, paste0(dir,'/sporades'), overwrite=TRUE)  # datasets organized in directory
#export(rec, paste0(dir,'/sporades.zip')) # zipped datasets


# import
rec <- import(filename=paste0(dir,'/sporades.qs2'))
rec <- import(filename=paste0(dir,'/sporades.rds'))
rec <- import(filename=paste0(dir,'/sporades'))

# explore

## tabs class object
if (interactive()){
explore(rec) # comparison present reference and topo-based
explore(rec, timelapse=3) # paleo reconstruction
}

## from exported object
if (interactive()){
explore(paste0(dir,'/sporades.qs2'))
explore(paste0(dir,'/sporades.qs2'), timelapse=1)
}

# get area
area <- get_area(rec) # using object class tabs
```

```
area <- get_area(rec$recvect) # using recvect

area <- get_area(paste0(dir,'/sporades.qs2')) # using exported object
unlink(dir)
```

`features` *Dataset: features*

Description

Features that can be defined as labeling points in the default dataset. point_reference dataset has been retrieved from geoNames.

Usage

```
features
```

Format

data frame

feature_class feature class

feature_class_description description of the feature class

feature feature names

Source

GeoNames `https://download.geonames.org/export/dump/`

Examples

```
f <- features
```

`funza` *Dataset: Funza (29500 up to 1000000 BP)*

Description

The ‘páramos’ UFL, i.e. high altitude mountain ecosystem in Venezuela (Northern Andes), during the last 1 million years, regularized to intervals of 1 ky.

Usage

```
funza
```

Format

A vector with:

year_before_after_present years before or after present

tree_line_m tree line expressed in meters above the present day reference sea level.

Source

Flantua et al. (2019) doi:10.3389/fevo.2021.615223

Examples

```
curve <- funza
```

`get_area` *Get area in meter*

Description

Get the area based on an object of class tabs or recvect object

Usage

```
get_area(filename, verbose = FALSE)
```

Arguments

`filename`	character. Object of class tabs, recvect (SpatVector) or path where outputs of the reconstruct-function were exported. Data exported in the following formats can be read by get_area: directory tree, .qs2 and .rds.
`verbose`	boolean, print messages

Value

tabs object

Author(s)

Johannes De Groeve

Examples

```
# load data samples
sporades <- sporades()
topo <- sporades$topo
labs <- sporades$labs
correction <- sporades$correction
curve <- sporades$curve

# subset first and last period
curve <- curve[[c(1,dim(curve)[3])]]

# reconstruct
rec <- reconstruct(topo=topo,
                   region=labs,
                   curve=curve,
                   correction=correction,
                   reclabs='name'
                   )

# metadata
md <- metadata(rec)


# export
dir <- tempdir() # export to temporary directory
export(rec, paste0(dir,'/sporades.qs2'),overwrite=TRUE) # qs2 (faster and less storage than rds)
export(rec, paste0(dir,'/sporades.rds'), overwrite=TRUE) # rds / rdata
export(rec, paste0(dir,'/sporades'), overwrite=TRUE)  # datasets organized in directory
#export(rec, paste0(dir,'/sporades.zip')) # zipped datasets


# import
rec <- import(filename=paste0(dir,'/sporades.qs2'))
rec <- import(filename=paste0(dir,'/sporades.rds'))
rec <- import(filename=paste0(dir,'/sporades'))

# explore

## tabs class object
if (interactive()){
explore(rec) # comparison present reference and topo-based
explore(rec, timelapse=3) # paleo reconstruction
}
```

```
## from exported object
if (interactive()){
explore(paste0(dir,'/sporades.qs2'))
explore(paste0(dir,'/sporades.qs2'), timelapse=1)
}

# get area
area <- get_area(rec) # using object class tabs
area <- get_area(rec$recvect) # using recvect

area <- get_area(paste0(dir,'/sporades.qs2')) # using exported object
unlink(dir)
```

`get_correction` *Get and prepare correction*

Description

Get and prepare a correction dataset

Usage

```
get_correction(
  correction = NULL,
  topo = NULL,
  curve = NULL,
 units = list(topo = "m", curve = c(names = "yr", value = "m"), correction = "mm/yr"),
  verbose = FALSE
)
```

Arguments

`correction`	SpatRaster. Correction value, vector, grid, or list of grids to account for spatial-(non-)explicit and temporal (non-)linear changes in the topography (e.g., uplift and subsidence rates, sedimentation and erosion ticknesses)
`topo`	SpatRaster. Topographic/Bathymetric model as SpatRaster or path to dataset. The topo projection is the reference for further outputs.
`curve`	SpatRaster. Curve value, vector, grid or list of grids indicating the relative altitude of a biogeographic system per time period compared to the present. A typical example is a sea level curve indicating the relative sea level position above or below sea level compared to the present.
`units`	numeric. Units of topo, curve and correction provided as a list (default: units=list(topo='m', curve=c(names='yr', value='m'), correction='mm/yr'))
`verbose`	boolean. FALSE: No messages are printed. TRUE: Standard verbose mode 2: Very verbose mode, displaying detailed information.

Value

A SpatRaster or vector with corrrection values in a suitable format for the reconstruct function, including a value for each time step, defined by the curve.

Author(s)

Johannes De Groeve

See Also

correction

Examples

```
sporades <- sporades()
topo <- sporades$topo
correction <- sporades$correction
curve <- sporades$curve

cor <- get_correction(correction=correction,
                      topo=topo,
                      curve=curve)
```

`get_curve` *Get and prepare curve*

Description

load a curve of interest

Usage

```
get_curve(curve = NULL, verbose = FALSE)
```

Arguments

`curve`	SpatRaster. Curve value, vector, grid or list of grids indicating the relative altitude of a biogeographic system per time period compared to the present. A typical example is a sea level curve indicating the relative sea level position above or below sea level compared to the present.
`verbose`	Boolean. FALSE: No messages are printed. TRUE: Standard verbose mode. 2: Very verbose mode, displaying detailed information.

Value

A SpatRaster or vector with curve values in a suitable format for the reconstruct function.

Author(s)

Johannes De Groeve

See Also

curve

Examples

```
curve <- get_curve('lambeck')
curve <- get_curve('cutler')
curve <- get_curve('IPCC')
curve <- get_curve('funza')

curve <- get_curve()
curve <- get_curve(10)
curve <- get_curve(c(0,100,200,300,400,500))

# period definition curve
cur <- c(0,100,200,300,400,500) # altitudes
names(cur) <- c(0,1,2,3,4,5) # periods
curve <- get_curve(cur)

# add source attribute
cur <- c(0,100,200,300,400,500) # altitudes
names(cur) <- c(0,1,2,3,4,5) # periods
attr(cur, 'source') <- 'new curve' # curve source
curve <- get_curve(cur)

# custom-curve from data frame
cur <- data.frame(period=0:10,
                  altitude=seq(0,-20,-2),
                  source='custom')
curve <- get_curve(cur)
```

`get_data` *Get and prepare input datasets (topo, labs, curve, correction)*

Description

load and prepare input datasets topo, labs, curve and correction

Usage

```
get_data(
  region = NULL,
  topo = NULL,
  aggregate = FALSE,
```

```
    curve = NULL,
    correction = NULL,
    reclabs = NULL,
   units = list(topo = "m", curve = c(names = "yr", value = "m"), correction = "mm/yr"),
    buffer = NULL,
    fact = 0,
    verbose = FALSE
)
```

Arguments

`region`	SpatVector. Region selection object defined by extent coordinates, a polygon object or path to dataset, an island, archipelago, country, mountain or plate name from the regions-list. If region is not defined a selection window will pop-up to define the area of interest.
`topo`	SpatRaster. Topographic/Bathymetric model as SpatRaster or path to dataset. The topo projection is the reference for further outputs.
`aggregate`	boolean. Whether to aggregate biogeographic shapes.
`curve`	SpatRaster. Curve value, vector, grid or list of grids indicating the relative altitude of a biogeographic system per time period compared to the present. A typical example is a sea level curve indicating the relative sea level position above or below sea level compared to the present.
`correction`	SpatRaster. Correction value, vector, grid, or list of grids to account for spatial-(non-)explicit and temporal (non-)linear changes in the topography (e.g., uplift and subsidence rates, sedimentation and erosion ticknesses)
`reclabs`	character. Dataset or column used for labeling biogeographic shapes. By default the island labeling dataset is used, while if reclabs is set to ‘mnts’ the mountain labeling is used. Otherwise another column from the region object could be used, or a feature from the geonames feature list (e.g., ‘peaks’, ‘peak’) could be specified. Note that any overlapping name from the list geonames features cannot be used. If so, it is recommended to rename your labeling column. Note that in case of a user-defined reclabs column, the concerned column will be replicated in the labs-object under the column name ‘name’.
`units`	numeric. Units of topo, curve and correction provided as a list (default: units=list(topo=’m’, curve=c(names=’yr’, value=’m’), correction=’mm/yr’))
`buffer`	numeric. Draws a buffer around the selected region. For extent, the buffer is 0, otherwise 10000 m.
`fact`	numeric. Spatial resolution factor at which the bathymetric model will be resampled
`verbose`	boolean. FALSE: No messages are printed. TRUE: Standard verbose mode, providing progress bar. 2: Very verbose mode, displaying detailed information.

Value

a list including topo, labs, curve and correction input datasets

Author(s)

Johannes De Groeve

Examples

```
# load data samples
sporades <- sporades()
topo <- sporades$topo
labs <- sporades$labs
correction <- sporades$correction
curve <- sporades$curve

# subset first and last period
curve <- curve[[c(1,dim(curve)[3])]]


data <- get_data(topo=topo,
                 region=labs,
                 curve=curve)
data <- get_data(topo=topo,
                 region=labs,
                 curve=curve,
                 correction=correction)

# run reconstruct using prepared input datasets
rec <- reconstruct(data)
```

`get_region` *Get and prepare region*

Description

Retrieve region of interest

Usage

```
get_region(
  region = NULL,
  buffer = NULL,
  reclabs = NULL,
  aggregate = FALSE,
  verbose = FALSE,
  overwrite = FALSE
)
```

Arguments

`region`	SpatVector. Region selection object defined by extent coordinates, a polygon object or path to dataset, an island, archipelago, country, mountain or plate name from the regions-list. If region is not defined a selection window will pop-up to define the area of interest.
`buffer`	numeric. Draws a buffer around the selected region. For extent, the buffer is 0, otherwise 10000 m.
`reclabs`	character. Dataset or column used for labeling biogeographic shapes. By default the island labeling dataset is used, while if reclabs is set to ‘mnts’ the mountain labeling is used. Otherwise another column from the region object could be used, or a feature from the geonames feature list (e.g., ‘peaks’, ‘peak’) could be specified. Note that any overlapping name from the list geonames features cannot be used. If so, it is recommended to rename your labeling column. Note that in case of a user-defined reclabs column, the concerned column will be replicated in the labs-object under the column name ‘name’.
`aggregate`	boolean. Whether to aggregate biogeographic shapes.
`verbose`	boolean. FALSE: No messages are printed. TRUE: Standard verbose mode. 2: Very verbose mode, displaying detailed information.
`overwrite`	boolean. TRUE: overwrite region and select a new extent via selection window will pop-up.

Value

A SpatVector object with the labeling polygons for the specified extent coming from spatial object, extent or region name

See Also

region

Examples

```
if(interactive()){
# interactive selection
r <- get_region()
# overwrite interactive selection
r <- get_region(overwrite=TRUE)
}

# sample dataset
labs <- sporades()$labs
# labels based on "name" column
r <- get_region(region=labs)
# labels based on specific column
r <- get_region(region=labs[,'plate'],reclabs='plate')
# automatic labeling
r <- get_region(region=labs, reclabs = FALSE)
# aggregate shapes
```

```
r <- get_region(region=labs, aggregate=TRUE)
# define by extent
library(terra)
e <- terra::ext(labs)
r <- get_region(region=e)
e <- as.vector(e)
r <- get_region(region=e)
# add buffer
r <- get_region(region=e,buffer=10000)
r <- get_region(region=e,buffer=100000)
```

`import` *Import a reconstruction as class tabs object*

Description

import TABS object

Usage

```
import(filename)
```

Arguments

`filename`	character. Path where outputs of the reconstruct-function were exported. Data exported in the following formats can be imported: directory tree, .qs2 and .rds.

Value

object of class tabs including a list of input (topo, labs, curve, correction) and output (recvect, recrast, recarea) datasets

Author(s)

Johannes De Groeve

Examples

```
# load data samples
sporades <- sporades()
topo <- sporades$topo
labs <- sporades$labs
correction <- sporades$correction
curve <- sporades$curve

# subset first and last period
```

```
curve <- curve[[c(1,dim(curve)[3])]]

# reconstruct
rec <- reconstruct(topo=topo,
                   region=labs,
                   curve=curve,
                   correction=correction,
                   reclabs='name'
                   )

# metadata
md <- metadata(rec)


# export
dir <- tempdir() # export to temporary directory
export(rec, paste0(dir,'/sporades.qs2'),overwrite=TRUE) # qs2 (faster and less storage than rds)
export(rec, paste0(dir,'/sporades.rds'), overwrite=TRUE) # rds / rdata
export(rec, paste0(dir,'/sporades'), overwrite=TRUE)  # datasets organized in directory
#export(rec, paste0(dir,'/sporades.zip')) # zipped datasets


# import
rec <- import(filename=paste0(dir,'/sporades.qs2'))
rec <- import(filename=paste0(dir,'/sporades.rds'))
rec <- import(filename=paste0(dir,'/sporades'))

# explore

## tabs class object
if (interactive()){
explore(rec) # comparison present reference and topo-based
explore(rec, timelapse=3) # paleo reconstruction
}

## from exported object
if (interactive()){
explore(paste0(dir,'/sporades.qs2'))
explore(paste0(dir,'/sporades.qs2'), timelapse=1)
}

# get area
area <- get_area(rec) # using object class tabs
area <- get_area(rec$recvect) # using recvect

area <- get_area(paste0(dir,'/sporades.qs2')) # using exported object
unlink(dir)
```

`IPCC_global_mean` *Dataset: IPCC future predictions (2021-2100)*

Description

Mean global sea level rise according to different scenarios (ssp1,ssp2,ssp3,ssp5) for 2021-2040, 2041-2060 and 2081-2100. The global means were calculated from global raster datasets available for download in the interactive IPCC atlas.

Usage

```
IPCC_global_mean
```

Format

A list four RSL vectors

year_before_after_present years before or after present

sea_level_m meters below or above the reference sea level expressed in m.

Source

IPCC, Kirsten B; Edwards, Ross L; Taylor, Frederick W; Cheng, H; Adkins, Jess F; Gallup, Christina D; Cutler, P M; Burr, George S; Bloom, Arthur L (2003): Rapid sea-level fall and deep-ocean temperature change since the last interglacial period. Earth and Planetary Science Letters, 206(3-4), 253-271. doi:10.1016/S0012821X(02)01107X `https://interactive-atlas.ipcc.ch/regional-information`

Examples

```
curves <- IPCC_global_mean
```

`lambeck` *Dataset: Lambeck's Global sea level curve (0-35000 BP)*

Description

Global sea level curve of Lambeck et al. (2014) from 0 to 35000 BP.

Usage

```
lambeck
```

Format

A vector with:

year_before_after_present years before or after present

sea_level_m meters below or above the reference sea level expressed in m.

Source

Lambeck, Kurt; Rouby, Hélène; Purcell, Anthony; Sun, Y; Sambridge, Malcom (2014): Sea level and global ice volumes from the Last Glacial Maximum to the Holocene. Proceedings of the National Academy of Sciences, 111(43), 15296-15303. doi:10.1073/pnas.1411762111

Examples

```
curve <- lambeck
```

`metadata` *Metadata of object of class tabs (reconstruction)*

Description

retrieve dataset and column descriptions as well as the sources from reconstruction object

Usage

```
metadata(x)
```

Arguments

`x`	object of class tabs

Value

a list of data frames with a description of columns of vector datasets and the sources of the input datasets

Author(s)

Johannes De Groeve

Examples

```
# load data samples
sporades <- sporades()
topo <- sporades$topo
labs <- sporades$labs
correction <- sporades$correction
curve <- sporades$curve

# subset first and last period
curve <- curve[[c(1,dim(curve)[3])]]

# reconstruct
rec <- reconstruct(topo=topo,
                   region=labs,
```

```
                   curve=curve,
                   correction=correction,
                   reclabs='name'
                   )

# metadata
md <- metadata(rec)


# export
dir <- tempdir() # export to temporary directory
export(rec, paste0(dir,'/sporades.qs2'),overwrite=TRUE) # qs2 (faster and less storage than rds)
export(rec, paste0(dir,'/sporades.rds'), overwrite=TRUE) # rds / rdata
export(rec, paste0(dir,'/sporades'), overwrite=TRUE)  # datasets organized in directory
#export(rec, paste0(dir,'/sporades.zip')) # zipped datasets


# import
rec <- import(filename=paste0(dir,'/sporades.qs2'))
rec <- import(filename=paste0(dir,'/sporades.rds'))
rec <- import(filename=paste0(dir,'/sporades'))

# explore

## tabs class object
if (interactive()){
explore(rec) # comparison present reference and topo-based
explore(rec, timelapse=3) # paleo reconstruction
}

## from exported object
if (interactive()){
explore(paste0(dir,'/sporades.qs2'))
explore(paste0(dir,'/sporades.qs2'), timelapse=1)
}

# get area
area <- get_area(rec) # using object class tabs
area <- get_area(rec$recvect) # using recvect

area <- get_area(paste0(dir,'/sporades.qs2')) # using exported object
unlink(dir)
```

`reconstruct` *Reconstruct present- and paleoconfigurations for biogeographic systems*

Description

reconstruct paleo or present day landscape using a bathymetric model, island labeling dataset and a seacurve

Usage

```
reconstruct(
  x = NULL,
  region = NULL,
  topo = NULL,
  curve = NULL,
  correction = NULL,
  iso = 0,
  reclabs = NULL,
  buffer = NULL,
  aggregate = FALSE,
 units = list(topo = "m", curve = c(names = "yr", value = "m"), correction = "mm/yr"),
  fact = 0,
  noise = 5,
  noiserm = TRUE,
  fillholes = TRUE,
  filename = NULL,
  overwrite = FALSE,
  metrics = c("area"),
  verbose = FALSE
)
```

Arguments

`x`	get_data-object. Prepared object including standardized input datasets (region, topo, curve, correction) with get_data() (optional)
`region`	SpatVector. Region selection object defined by extent coordinates, a polygon object or path to dataset, an island, archipelago, country, mountain or plate name from the regions-list. If region is not defined a selection window will pop-up to define the area of interest.
`topo`	SpatRaster. Topographic/Bathymetric model as SpatRaster or path to dataset. The topo projection is the reference for further outputs.
`curve`	SpatRaster. Curve value, vector, grid or list of grids indicating the relative altitude of a biogeographic system per time period compared to the present. A typical example is a sea level curve indicating the relative sea level position above or below sea level compared to the present.
`correction`	SpatRaster. Correction value, vector, grid, or list of grids to account for spatial-(non-)explicit and temporal (non-)linear changes in the topography (e.g., uplift and subsidence rates, sedimentation and erosion ticknesses)
`iso`	numeric. Vector or list indicating the elevation range of the biogeographic system to reconstruct. By default 0 (coastlines). If provided as a list, the boundary definition of the range can be defined (options are '>' and '>='). By default, the

	elevation range definition includes the indicated lower bound value (i.e., list(0, ’>=’)).
`reclabs`	character. Dataset or column used for labeling biogeographic shapes. By default the island labeling dataset is used, while if reclabs is set to ‘mnts’ the mountain labeling is used. Otherwise another column from the region object could be used, or a feature from the geonames feature list (e.g., ‘peaks’, ‘peak’) could be specified. Note that any overlapping name from the list geonames features cannot be used. If so, it is recommended to rename your labeling column. Note that in case of a user-defined reclabs column, the concerned column will be replicated in the labs-object under the column name ‘name’.
`buffer`	numeric. Draws a buffer around the selected region. For extent, the buffer is 0, otherwise 10000 m.
`aggregate`	boolean. Whether to aggregate biogeographic shapes.
`units`	numeric. Units of topo, curve and correction provided as a list (default: units=list(topo=’m’, curve=c(names=’yr’, value=’m’), correction=’mm/yr’))
`fact`	numeric. Resolution factor, increasing the factor will half the resolution.
`noise`	numeric. Maximum number of unlabeled clumped topo pixels considered as noise. Note that clumps of pixels are only considered as noise when their highest points do not intersect with a reference polygon.
`noiserm`	boolean. Whether noise should be removed.
`fillholes`	boolean. fill the holes in polygons, independent from noise (e.g. lakes)
`filename`	character. Path where files will be exported. Default as directory tree. Use .qs2, .rds, .zip to save as qs2, rds or zipped directory tree.
`overwrite`	boolean. Whether to overwrite the output when filename is specified.
`metrics`	character. metrics to calculate for each biogeographic shape, currently only area is implemented.
`verbose`	boolean. FALSE: No messages are printed. TRUE: Standard verbose mode, providing progress bar. 2: Very verbose mode, displaying detailed information.

Details

==

INPUT

==

input dataset may be topo, curve, correction (optional) and a labs dataset:

TOPO Topographic and/or bathymetric raster used to identify biogeographic shapes for the extent of the selected region.

CURVE The relative altitude of a biogeographic system per time period compared to the present expressed as a numeric vector (e.g., Lambeck, Cutler, Funza) or raster (e.g., st_curve). In the case of st_curve, the curve is returned for the extent of the selected region and resampled to the resolution of the topo dataset. If the curve is not defined, 0 is returned and a reconstruction is made for the present-day sea level.

CORRECTION Correction numeric vector or raster harmonized with the curve and resampled to the resolution of the topo dataset. If the input correction raster or numeric vector is defined as a rate (i.e., a single value, a single raster; thus, assuming temporal linear changes in topography), a correction variable (raster/numeric vector) is returned with the same length as the curve, expressing the cumulative topographic change over time. If the correction parameter is not defined, 0 is returned.

LABS Labeling dataset that is used for naming biogeographic shapes for the extent of the selected region.

Returned variables:

unique_id `integer`: Unique identifier of a biogeographic shape in the labeling dataset.

name `character`: Name of the biogeographic shape in the labeling dataset. By default this will be derived from the Global Shoreline Vector (GSV; Sayre et al. 2019), or from the mountain inventory v2 (GMBA; Snethlage et al. 2022), when reclabs is set to ‘mnts’. Otherwise, if a custom polygon reference and labeling dataset is used, the name-column will store the content of a by the user specified column. NOTE: If the labeling column is specified by the user, that one will be stored as a duplicate in the labs output under its original name.

uniquename `character`: Concatenated name and unique identifier.

refx `numeric`: X-coordinate (SRID=4326) of the highest point of a biogeographic shape in the labeling dataset. If the labeling dataset are points, the x-coordinate of the point is given.

refy `numeric`: Y-coordinate (SRID=4326) of the highest point of a biogeographic shape in the labeling dataset. If the labeling dataset are points, the y-coordinate of the point is given.

refz `numeric`: Meter above/below present sea level of the highest point within a biogeographic shape extracted through intersection with topo. If the labeling dataset are points, the z of the point is given.

refn `integer`: Number of cells at the resolution of the topo within a biogeographic shape in the labeling dataset. If the labeling dataset are points, the number of cells will equal 1.

Depending from the used labeling dataset (GSV, GMDA, GeoNames) additional other columns are returned.

==
OUTPUT
==

RECAREA Table expressing the area in square meters for each biogeographic shape per time period.

RECRAST Raster expressing the reconstruction of a biogeographic region per time period within the extent of the selected region.

RECVECT Spatial vector expressing the reconstruction per time period, identifying each polygon as a different biogeographic shape. The vector layers include a range of default attributes per biogeographic shape.
NOTE: In case `reclabs=FALSE`, biogeographic shapes are merged into a single multipolygon, and all attributes in the vector layers are expressed for the whole region.

Returned variables:

iso `numeric`: Meter above or below the curve value (e.g., sea level position) defining the lower bound of the range of a biogeographic system

period `character`: Lower bound of a time period expressed in years before/after present for a reconstruction at a specific curve value (e.g., sea level position).

curve `numeric`: Curve value (e.g., sea level position) for that period. In case of a raster (e.g., st_curve; spatial-explicit curve) the average curve value is calculated within the region.

unique_id `integer`: Unique identifier of a biogeographic shape for a time period.

area `numeric`: Size of a biogeographic shape in square meters.

n `integer`: Number of cells at the resolution of the topo within a biogeographic shape; will change if the fact parameter is modified.

x `numeric`: X-coordinate in degrees (SRID=4326) of the highest point within a biogeographic shape extracted through intersection with topo. If the highest point could not be extracted, the centroid of the biogeographic shape is used.

y `numeric`: Y-coordinate in degrees (SRID=4326) of the highest point within a biogeographic shape extracted through intersection with topo. If the highest point could not be extracted, the centroid of the biogeographic shape is used.

z `numeric`: Meter above/below present sea level of the highest point within a biogeographic shape extracted through intersection with topo. If the highest point could not be extracted, the centroid of the biogeographic shape is used.

recid `character`: Reconstructed ID, when biogeographic shapes merge over time it is named after the shape with the highest point. The ID is unique across islands with identical names.

recname `character`: Reconstructed name, when biogeographic shapes merge over time it is named after the shape with the highest point. It can have the following formats:

- `S-<PERIOD>-<ID>` S biogeographic shape identified in topo but not in labeling dataset; or drowned biogeographic shapes that were disconnected from a present-day existing biogeographic shape
  - `PERIOD` The most recent period the biogeographic shape emerged.
  - `ID` Identifier.
- `UNKNOWN` The name of the biogeographic shape is intersecting with the labeling dataset but unknown (only for the Global Shoreline Vector - i.e. island labeling dataset).
- `UNNAMED` The name of the biogeographic shape is intersecting with the labeling dataset but unnamed (only for the Global Shoreline Vector - i.e. island labeling dataset).

recnames `json`: JSON object including the name and ID of intersecting reconstructed polygons (from t0 until ti, where t = time period) within a reconstructed polygon of ti.

refnames `json`: JSON object including the name and ID of intersecting labeling points/polygons (from t0 until ti, where t = time period) within a reconstructed polygon of ti.

=========

Value

object of class tabs including a list of input (topo, labs, curve, correction) and output (recvect, recrast, recarea) datasets

Author(s)

Johannes De Groeve

See Also

get started

tabs object structure

Examples

```
# load data samples
sporades <- sporades()
topo <- sporades$topo
labs <- sporades$labs
correction <- sporades$correction
curve <- sporades$curve

# subset first and last period
curve <- curve[[c(1,dim(curve)[3])]]

# reconstruct
rec <- reconstruct(topo=topo,
                   region=labs,
                   curve=curve,
                   correction=correction,
                   reclabs='name'
                   )

# metadata
md <- metadata(rec)


# export
dir <- tempdir() # export to temporary directory
export(rec, paste0(dir,'/sporades.qs2'),overwrite=TRUE) # qs2 (faster and less storage than rds)
export(rec, paste0(dir,'/sporades.rds'), overwrite=TRUE) # rds / rdata
export(rec, paste0(dir,'/sporades'), overwrite=TRUE)  # datasets organized in directory
#export(rec, paste0(dir,'/sporades.zip')) # zipped datasets


# import
rec <- import(filename=paste0(dir,'/sporades.qs2'))
rec <- import(filename=paste0(dir,'/sporades.rds'))
rec <- import(filename=paste0(dir,'/sporades'))

# explore

## tabs class object
if (interactive()){
explore(rec) # comparison present reference and topo-based
explore(rec, timelapse=3) # paleo reconstruction
}
```

```
## from exported object
if (interactive()){
explore(paste0(dir,'/sporades.qs2'))
explore(paste0(dir,'/sporades.qs2'), timelapse=1)
}

# get area
area <- get_area(rec) # using object class tabs
area <- get_area(rec$recvect) # using recvect

area <- get_area(paste0(dir,'/sporades.qs2')) # using exported object
unlink(dir)
```

`regions` *Dataset: regions*

Description

mountain ranges at different hierarchical levels, islands, archipelagoes, countries and plates that can be used for region selection

Usage

```
regions
```

Format

vector

dataset the dataset (islands,mountains) for which the region definition can be used

region region type

name name of the region

name_ascii name in ascii format

Source

Islands: Sayre et al. 2019 doi:10.1080/1755876X.2018.1529714

Plates: Bird, P. (2003) doi:10.1029/2001GC000252

Archipelagoes: Weigelt et al. 2013 doi:10.5061/dryad.fv94v

Countries: from rnaturalearth `https://docs.ropensci.org/rnaturalearth/`

Mountains: Snethlage et al. 2022 doi:10.48601/earthenvt9k21407

Examples

```
r <- regions
```

`setup` *Setup - Download of default datasets*

Description

download default datasets (labs, topo, curve) in default or custom directory. The datasets require 15 GB disk space.

Usage

```
setup()
```

Value

No return value, called for side effects

Author(s)

Johannes De Groeve

Examples

```
if(interactive()){

# download labs, topo and curve and store in custom or default directory
setup()

# reset setup
# if you want the install the datasets under a different path
options(tabs.datasetPath=NULL)
setup()
}
```

`sporades` *Sporades sample dataset including topo, labs, curve and correction*

Description

Sample dataset for the Sporades Archipelago in Greece

Usage

```
sporades()
```

Value

List of input sample datasets (topo, labs, curve, correction)

Author(s)

Johannes De Groeve

Index