gbif.range

December 8, 2022

Type Package

Title A toolbox to generate species range maps based on ecoregions and a user-friendly GBIF wrapper

Version 0.2.0

Depends R (>= 4.0.0), raster, terra, rgbif, CoordinateCleaner, rgeos, mclust, ClusterR, FNN, geometry, cluster

Description Workflow to generate taxa range maps from scratch based on ecoregions and an user-friendly GBIF wrapper (no hard-limit of < 100,000 species observations, filetring parameters to easily flag records, access to the GBIF backbone taxonomy)

License GPL (>=3)

BugReports https://github.com/8Ginette8/gbif.range/issues

Encoding UTF-8

Maintainer Yohann Chauvier <yohann.chauvier@wsl.ch>

LazyData true

RoxygenNote 7.1.2

Authors Yohann Chauvier [cre,aut] (https://orcid.org/0000-0001-9399-3192)
Oskar Hagen [aut] (https://orcid.org/0000-0002-7931-6571)
Camille Albouy [aut] (https://orcid.org/0000-0003-1629-2389)
Patrice Descombes [aut] (https://orcid.org/0000-0002-3760-9907)
Fabian Fopp [aut] (https://orcid.org/0000-0003-0648-8484)
Michael P. Nobis [aut] (https://orcid.org/0000-0003-3285-1590)
Philipp Brun [aut] (https://orcid.org/0000-0002-2750-9793)
Lisha Lyu [aut] (https://orcid.org/0000-0001-7855-8109)
Katalin Csillery [aut] (https://orcid.org/0000-0003-0039-9296)
Loïc Pellissier [aut] (https://orcid.org/0000-0002-2289-8259)

Collate 'make_tiles.R'
'get_gbif.R'
'get_taxonomy.R'
'get_doi.R'
'obs_filter.R'
'get_range.R'
'conv_function.R'
'make_ecoregion.R'

R topics documented:

`conv_function` *Create polygon objects in different bioregions.*

Description

Not to be called directly by the user.

Usage

```
conv_function(sp_coord, bwp, bipl, bwpo, temp_dir, g = NULL)
```

`get_doi` *Get a custom DOI for a GBIF filtered dataset*

Description

A small user friendly wrapper of the derived_dataset() function of the rgbif R package, compatible with one or several get_gbif() outputs.

Usage

```
get_doi(
  get.gbif = NULL,
  title = NULL,
  description = NULL,
  source_url = "https://example.com/",
  user = "",
  pwd = "",
  ...
)
```

Arguments

`get.gbif`	data.frame or list. One get_gbif() output or a list of several.
`title`	Title for your derived dataset.
`source_url`	Link to where the dataset is stored.
`user`	Your GBIF username.
`pwd`	Your GBIF password.
`...`	Additonnal parameters for derived_dataset() in rgbif.
`descritpion`	Description of the dataset.

Details

see derived_dataset() function from the rgbif R package.

Value

One citable DOI and its information.

References

Chamberlain, S., Oldoni, D., & Waller, J. (2022). rgbif: interface to the global biodiversity information facility API. 10.5281/zenodo.6023735

See Also

The rgbif package for additional and more general approaches to get GBIF DOI.

Examples

```
# Downloading worldwide the observations of Panthera tigris and Ailuropoda melanoleuca
obs.pt = get_gbif("Panthera tigris")
obs.am = get_gbif("Ailuropoda melanoleuca")

# Just an example on how to retrieve the DOI for only one get_gbif() output
get_doi(obs.pt,title="GBIF_test1",description="A small example 1",
   source_url="https://example.com/",user="",pwd="") # Use your own GBIF credentials here

# Another example on how to retrieve the DOI for several get_gbif() outputs
get_doi(list(obs.pt,obs.am),title="GBIF_test2",description="A small example 2",
   source_url="https://example.com/",user="",pwd="") # Use your own GBIF credentials here
```

---

get_gbif — *Massively download and filter GBIF observations for sound spatial analyses*

Description

Implement an user-friendly workflow to download and clean gbif taxa observations. The function uses the rgbif R package but (1) implements the same search result found if www.gbif.org is employed i.e., based on the input taxa name, all species records related to its accepted name and synonyms are extracted. The function also (2) bypasses the rgbif hard limit on the number of records (100'000 max). For this purpose, a dynamic moving window is created and used across the geographic extent defined by the user. This window automatically fragments the specified study area in succesive tiles of different sizes, until all tiles include < 100'000 observations. The function also (3) automatically applies a post-filtering of observations based on the chosen resolution of the study/analysis and by partly employing the CoordinateCleaner R package. Filtering options may be chosen and involve several choices: study's extent, removal of duplicates, removal of absences, basis of records selection, removal of invalid/uncertain xy coordinates (WGS84), time period selection and removal of raster centroids. By default, the argument hasGeospatialIssue in occ_data() (implemented rgbif function) is set to FALSE.

Usage

```
get_gbif(
  sp_name = NULL,
  conf_match = 90,
  geo = NULL,
  grain = 1000,
  duplicates = FALSE,
  absences = FALSE,
  no_xy = FALSE,
  basis = c("OBSERVATION", "HUMAN_OBSERVATION", "MACHINE_OBSERVATION",
   "MATERIAL_CITATION", "MATERIAL_SAMPLE", "PRESERVED_SPECIMEN", "FOSSIL_SPECIMEN",
    "LIVING_SPECIMEN", "LITERATURE", "UNKNOWN", "OCCURRENCE"),
  add_infos = NULL,
  time_period = c(1000, 3000),
  identic_xy = FALSE,
  wConverted_xy = FALSE,
  centroids = FALSE,
  ntries = 10,
  error.skip = TRUE,
  occ_samp = 99000,
  ...
)
```

Arguments

sp_name	Character. Scientific name to run an online search (i.e. with GBIF-API) to get occurrence records. Works also for genus and higher taxa levels.
conf_match	Numeric from 0 to 100. Determine the confidence threshold of match of 'sp_name' with the GBIF backbone taxonomy. Default is 90.
geo	Object of class Extent, SpatExtent, SpatialPolygon, SpatialPolygonDataframe, or SpatVector (WGS84) to define the study's area extent. Default is NULL i.e. the whole globe.
grain	Numeric. Specify in meters the study resolution. Used to filter gbif records according to their (1) spatial uncertainties and (2) number of coordinate decimals. Records with no information on coordinate uncertainties (column 'coordinateUncertaintyInMeters') are be kept by default. See details.
duplicates	Logical. Should duplicated records be kept?
absences	Logical. Should absence records be kept?
no_xy	Logical. Default is FALSE i.e., only records with coordinates are downloaded. If TRUE, records with no coordinates are also downloaded.
basis	Character. Which basis of records should be selected? Default is all i.e. c('OBSERVATION', 'HUMAN_OBSERVATION', 'MACHINE_OBSERVATION', 'MATERIAL_CITATION', MATERIAL_SAMPLE', 'PRESERVED_SPECIMEN', 'FOSSIL_SPECIMEN', 'LIVING_SPECIMEN', 'LITERATURE',UNKNOWN','OCCURRENCE'). Description may be found here: https://docs.gbif.org/course-data-use/en/basis-of-record.html, https://gbif.github.io/parsers/apidocs/org/gbif/api/vocabulary/BasisOfRecord.html
add_infos	Character. Infos that may be added to the default output information. List of IDs may be found at: https://www.gbif.org/developer/occurrence. Default IDs contain 'taxonKey', 'scientificName', 'acceptedTaxonKey', 'acceptedScientificName', 'individualCount', 'decimalLatitude', 'decimalLongitude', 'basisOfRecord', 'coordinateUncertaintyInMeters', 'countryCode', 'country', 'year',

'datasetKey', 'institutionCode', 'publishingOrgKey', 'taxonomicStatus' and 'taxonRank'.

`time_period` Numerical vector. Observations will be downloaded according to the chosen year range. Default is c(1000,3000). Observations with year = NA are kept by default.

`identic_xy` Logical. Should records with identical xy be kept?

`wConverted_xy` Logical. Should incorrectly converted lon/lat be kept? Uses cd_ddmm() from 'CoordinateCleaner' R package.

`centroids` Logical. Should species records from raster centroids be kept? Uses cd_round() from 'CoordinateCleaner' R package.

`ntries` Numeric. In case of failure from GBIF server or within the rgbif package, how many download attempts should the function request? Default is '10' with a 2 seconds interval between tries. If the attempts failed, an empty data.frame is return by default.

`error.skip` Logical. Should the search process continues if ntries failed ?

`occ_samp` Numeric. Determine how many GBIF occurrences will be sampled per geographic tiles of the fragmented study area. Default is the maximum number of GBIF observations found in a tile (i.e. ~100'000 records). A lower number may be set (<99'000) if the user only wants a sample of the species GBIF observations, hence increasing the download process and the generation of its range map if get_range() is employed afterwards.

`...` Additonnal parameters for the function cd_round() of CoordinateCleaner.

Details

Argument 'grain' used for two distinct gbif records filtering. (1) Records filtering according to gbif 'coordinateUncertaintyInMeters'; every records uncertainty > grain/2 are removed. Note: Records with no information on coordinate uncertainties are kept by default. (2) Records filtering according to the number of longitude/latitude decimals; if 110km < grain <= 11km, lon/lat with >= 1 decimal are kept, if 11km < grain <= 1100m, lon/lat with >= 2 decimals kept; if 1100m < grain <= 110m, lon/lat with >= 3 decimals are kept; if 110m < grain <= 11m, lon/lat with >= 4 decimals are kept; if 11m < grain <= 1.1m, lon/lat with >= 5 decimals are kept etc...

Value

Object of class data.frame with requested GBIF information. Although the function works accurately, error outputs might still occur depending on the 'sp_name' used. Therefore, default information detailed in 'add_infos' is stored so that sanity checks may still be applied afterwards. Although crucial preliminary checks of species records are done by the function, additional post exploration with the CoordinateCleaner R package is still highly recommended.

References

Chauvier, Y., Thuiller, W., Brun, P., Lavergne, S., Descombes, P., Karger, D. N., ... & Zimmermann, N. E. (2021). Influence of climate, soil, and land cover on plant species distribution in the European Alps. Ecological monographs, 91(2), e01433. 10.1002/ecm.1433

Chamberlain, S., Oldoni, D., & Waller, J. (2022). rgbif: interface to the global biodiversity information facility API. 10.5281/zenodo.6023735

Zizka, A., Silvestro, D., Andermann, T., Azevedo, J., Duarte Ritter, C., Edler, D., ... & Antonelli, A. (2019). CoordinateCleaner: Standardized cleaning of occurrence records from biological collection databases. Methods in Ecology and Evolution, 10(5), 744-751. 10.1111/2041-210X.13152

Hijmans, Robert J. "terra: Spatial Data Analysis. R Package Version 1.6-7." (2022). Terra - CRAN

See Also

The (1) rgbif and (2) CoordinateCelaner packages for additional and more general approaches on (1) downloading GBIF observations and (2) post-filtering those.

Examples

```
# Load maptools for the map world
library(maptools)
data(wrld_simpl)

# Load the Alps Extend
data(geo_dat)

# Downloading worldwide the observations of Panthera tigris
obs.pt = get_gbif("Panthera tigris",basis=c("OBSERVATION","HUMAN_OBSERVATION","MACHINE_OBSERVATION"))
plot(wrld_simpl)
points(obs.pt[,c("decimalLongitude","decimalLatitude")],pch=20,col="#238b4550",cex=4)

# Downloading in the Alps the observations of Cypripedium calceolus (with a 100m grain and
# by adding the 'issues' column)
obs.cc = get_gbif("Cypripedium calceolus", geo = shp.lonlat, grain = 100, add_infos = c("issue"))
plot(shp.lonlat)
points(obs.cc[,c("decimalLongitude","decimalLatitude")],pch=20,col="#238b4550",cex=1)

# Downloading worlwide the observations of Ailuropoda melanoleuca (with a 100km grain, after 1990
# and by keeping duplicates and by adding the name of the person who collected the panda records)
obs.am = get_gbif("Ailuropoda melanoleuca", grain = 100000 , duplicates = TRUE,
    time_period = c(1990,3000), add_infos = c("recordedBy","issue"))
plot(wrld_simpl)
points(obs.am[,c("decimalLongitude","decimalLatitude")],pch=20,col="#238b4550",cex=4)

# Downloading worlwide the observations of Phascolarctos cinereus (with a 1km grain, after 1980,
# and keeping raster centroids)
obs.pc = get_gbif("Phascolarctos cinereus", grain = 1000,
    time_period = c(1990,3000), centroids = TRUE)
```

`get_range` *Create a species range map based on a get_gbif() output*

Description

Estimates species ranges based on occurrence data (GBIF or not) and ecoregions (may be an external, in-house or make_ecoregion() input). It first deletes outliers from the observation dataset and then creates a polygon (convex hull) with a user specified buffer around all the observations of one ecoregion. If there is only one observation in an ecoregion, a buffer around this point will be created. If all points in an ecoregion are on a line, the function will also create a buffer around these points, however, the buffer size increases with the number of points in the line. Finally, also note that in case of too many records, get_range can be used with a sub-sample of species observations to ensure a faster polygon process and/or to overcome potential RAM crash of the function.

Usage

```
get_range(
  sp_name = NULL,
  occ_coord = NULL,
  Bioreg = eco.earth,
  Bioreg_name = "ECO_NAME",
  degrees_outlier = 3,
  clustered_points_outlier = 2,
  buffer_width_point = 4,
  buffer_increment_point_line = 0.5,
  buffer_width_polygon = 4,
  dir_temp = paste0("temp", sample(1:99999999, 1)),
  raster = TRUE,
  res = 10
)
```

Arguments

`sp_name` Character. Species name e.g., 'Anemone nemorosa'.

`occ_coord` get_gbif() output or SpatialPoints object (WGS84/lonlat).

`Bioreg` SpatialPolygonsDataFrame containg different ecoregions (convex hulls will be classified on a bioreg basis). Note that this parameter may be fed with an external, generated (function make_ecoregion) or in-house ecoregion shapefile. Three in-house shapefiles are already included in the library: 'eco.earh' (for terrestrial species; Nature conservancy version adapted from Olson & al. 2001), 'eco.marine' (for marine species; Spalding & al. 2007, 2012) and 'eco.fresh' (for freshwater species; Abell & al. 2008). For marine species, eco.earth may also be used if the user wants to represent the terrestrial range of species that also partially settle on mainland. For fresh water species, same may be done if the user considers that terrestrial ecoregions should be more representtaive of the species ecology.

`Bioreg_name` Character. How is the slot containing the ecoregion names called? Default is the very detailed level of 'eco.earth' (aka 'ECO_NAME'). Note that 'EcoRegion' must always be used when using a make_ecoregion() output.

`degrees_outlier` Numeric. Distance threshold (degrees) for outlier classification. If the nearest minimal distance to the next point is larger than this threshold, it will be considered as an outlier.

`clustered_points_outlier` Numeric. Maximum number of points which are closer to each other than the degrees_outlier, but should still be considered as outliers.

`buffer_width_point` Numeric. Buffer (in degrees) which will be applied around single observations.

`buffer_increment_point_line` Numeric. How much should the buffer be increased for each point on a line.

`buffer_width_polygon` Numeric. Buffer (in degrees) which will be applied around distribution polygons (for each ecoregion).

`dir_temp` Character. Where should the temporary text file for the convex hull be saved? (text file will be deleted again).

raster	Logical. Should the output be a unified raster? Default is TRUE
res	Numeric. If raster = TRUE, which resolution? Final resolution in ° = 1°/res e.g., = 0.1° if res = 10. Default is 10.

Details

Ecoregions cover relatively large areas of land or water, and contain characteristic, geographically distinct assemblages of natural communities sharing a large majority of species, dynamics, and environmental conditions. The biodiversity of flora, fauna and ecosystems that characterise an ecoregion tends to be distinct from that of other ecoregions (https://en.wikipedia.org/wiki/Ecoregion).

Value

A Shapefile or a SpatRaster.

References

Oskar Hagen, Lisa Vaterlaus, Camille Albouy, Andrew Brown, Flurin Leugger, Renske E. Onstein, Charles Novaes de Santana, Christopher R. Scotese, Loïc Pellissier. (2019) Mountain building, climate cooling and the richness of cold-adapted plants in the Northern Hemisphere. Journal of Biogeography. doi: 10.1111/jbi.13653

Lyu, L., Leugger, F., Hagen, O., Fopp, F., Boschman, L. M., Strijk, J. S., ... & Pellissier, L. (2022). An integrated high resolution mapping shows congruent biodiversity patterns of Fagales and Pinales. New Phytologist, 235(2), 759-772 10.1111/nph.18158

Olson, D. M., Dinerstein, E., Wikramanayake, E. D., Burgess, N. D., Powell, G. V. N., Underwood, E. C., D'Amico, J. A., Itoua, I., Strand, H. E., Morrison, J. C., Loucks, C. J., Allnutt, T. F., Ricketts, T. H., Kura, Y., Lamoreux, J. F., Wettengel, W. W., Hedao, P., Kassem, K. R. 2001. Terrestrial ecoregions of the world: a new map of life on Earth. Bioscience 51(11):933-938. doi: 10.1641/0006-3568(2001)051

The Nature Conservancy (2009). Global Ecoregions, Major Habitat Types, Biogeographical Realms and The Nature Conservancy Terrestrial Assessment Units. GIS layers developed by The Nature Conservancy with multiple partners, combined from Olson et al. (2001), Bailey 1995 and Wiken 1986. Cambridge (UK): The Nature Conservancy.

Mark D. Spalding, Helen E. Fox, Gerald R. Allen, Nick Davidson, Zach A. Ferdaña, Max Finlayson, Benjamin S. Halpern, Miguel A. Jorge, Al Lombana, Sara A. Lourie, Kirsten D. Martin, Edmund McManus, Jennifer Molnar, Cheri A. Recchia, James Robertson, Marine Ecoregions of the World: A Bioregionalization of Coastal and Shelf Areas, BioScience, Volume 57, Issue 7, July 2007, Pages 573–583. doi: 10.1641/B570707

Spalding, M. D., Agostini, V. N., Rice, J., & Grant, S. M. (2012). Pelagic provinces of the world: a biogeographic classification of the world's surface pelagic waters. Ocean & Coastal Management, 60, 19-30. doi: 10.1016/j.ocecoaman.2011.12.016

The Nature Conservancy (2012). Marine Ecoregions and Pelagic Provinces of the World. GIS layers developed by The Nature Conservancy with multiple partners, combined from Spalding et al. (2007) and Spalding et al. (2012). Cambridge (UK): The Nature Conservancy.

Robin Abell, Michele L. Thieme, Carmen Revenga, Mark Bryer, Maurice Kottelat, Nina Bogutskaya, Brian Coad, Nick Mandrak, Salvador Contreras Balderas, William Bussing, Melanie L. J. Stiassny, Paul Skelton, Gerald R. Allen, Peter Unmack, Alexander Naseka, Rebecca Ng, Nikolai Sindorf, James Robertson, Eric Armijo, Jonathan V. Higgins, Thomas J. Heibel, Eric Wikramanayake, David Olson, Hugo L. López, Roberto E. Reis, John G. Lundberg, Mark H. Sabaj Pérez, Paulo Petry, Freshwater Ecoregions of the World: A New Map of Biogeographic Units for Freshwater Biodiversity Conservation, BioScience, Volume 58, Issue 5, May 2008, Pages 403–414. doi: 10.1641/B580507

Hijmans, Robert J. "terra: Spatial Data Analysis. R Package Version 1.6-7." (2022). Terra - CRAN

See Also

For more information on the original code and methods, check Hagen, Oskar et al. (2019), Data from: Mountain building, climate cooling and the richness of cold-adapted plants in the northern hemisphere, Dryad, Dataset, https://doi.org/10.5061/dryad.0ff6b04.

Examples

```
# Load available ecoregions
data(ecoregions)

# First download the worldwide observations of Panthera tigris and convert to SpatialPoints
obs.pt = get_gbif("Panthera tigris",basis=c("OBSERVATION","HUMAN_OBSERVATION","MACHINE_OBSERVATION"))

# Plot
plot(eco.earth)
plot(sp.shp,pch=20,col="#238b4550",cex=4,add=TRUE)

# Generate the distributional range map of Panthera tigris for the finest terrestrial ecoregions
range.tiger = get_range("Panthera tigris",sp.shp,eco.earth,"ECO_NAME")

# Plotting
plot(eco.earth)
plot(range.tiger,col="#238b45",add=TRUE)
```

---

get_taxonomy *Retrieve from GBIF all scientific names of a specific Taxa*

Description

Generates, based on a given species name, a list of all its scientific names (accepted, synonyms) found in the GBIF backbone taxonomy and used to download the data in get_gbif(). Children and related doubtful names not used to download the data may also be extracted.

Usage

```
get_taxonomy(sp_name = NULL, conf_match = 90, all = FALSE)
```

Arguments

sp_name	Character. Species name from which the user wants to retrieve all existing GBIF names.
conf_match	Numeric. From 0 to 100. Determine the confidence threshold of match of 'sp_name' with the GBIF backbone taxonomy. Default is 90.
all	Logical. Default is FALSE. Should all species names be retrieved or only the accepted name and its synonyms?

Value

Data.frame with three columns: (1) GBIF taxonomic key, (2) scientificName and (3) Backbone Taxonomy Status.

References

Chamberlain, S., Oldoni, D., & Waller, J. (2022). rgbif: interface to the global biodiversity information facility API. 10.5281/zenodo.6023735

See Also

The rgbif package for additional and more general approaches on how to retrieve scientific names from the GBIF backbone taxonomy.

Examples

```
get_taxonomy("Cypripedium calceolus",all=FALSE)
get_taxonomy("Cypripedium calceolus",all=TRUE)
```

---

make_ecoregion — *Make an ecoregion map based on input environmental variables*

Description

This function may be used if the in-house ecoregion shapefiles are too coarse for a given geographic region (e.g., for local studies) or a shapefile of finer environmental details is needed. Based on several environmental layers (e.g. climate, soil and land cover), this function can generate a map of environmental regions containing n categories/classes. The classes are calculated with the 'clustering large applications' method (CLARA), which recognize patterns and relationships existing in spatial data, and classify it in clusters.

Usage

```
make_ecoregion(
  env = NULL,
  nclass = NULL,
  path = "",
  name = "",
  raster = FALSE,
  ...
)
```

Arguments

`env`	Object of class SpatRaster, RasterBrick or RasterStack of desired resolution and extent defining the study area. Used to generate the map of clusters needed to summarize the environmental space of the study area.
`nclass`	Numeric, How many number of environmental classes should have the output?
`path`	Character. Folder path where the output should be saved. Default is none.

`name`	Character. If 'path' is used, should include the name of the output file (without file extension)
`raster`	Logical. Whether the output should be a raster layer. Default is FALSE.
`...`	Additonnal parameters for the function clara() of the clutser R package.

Value

A TIFF or SHP file

References

Chauvier, Y., Zimmermann, N. E., Poggiato, G., Bystrova, D., Brun, P., & Thuiller, W. (2021). Novel methods to correct for observer and sampling bias in presence-only species distribution models. Global Ecology and Biogeography, 30(11), 2312-2325.

Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., & Hornik, K. (2021). cluster: Cluster Analysis Basics and Extensions. R package version 2.1.2 — For new features, see the 'Changelog' file (in the package source). https://CRAN.R-project.org/package=cluster

Reynolds, A. P., Richards, G., de la Iglesia, B., & Rayward-Smith, V. J. (2006). Clustering rules: A comparison of partitioning and hierarchical clustering algorithms. Journal of Mathematical Modelling and Algorithms, 5(4), 475–504. doi: 10.1007/s10852-005-9022-1

Schubert, E., & Rousseeuw, P. J. (2019). Faster k-Medoids clustering: Improving the PAM, CLARA, and CLARANS algorithms. In G. Amato, C. Gennaro, V. Oria, & M. Radovanović (Eds.), Similarity search and applications. SISAP 2019. Lecture Notes in Computer Science (Vol. 11807, pp. 171–187). Springer.

Examples

```
# Load the European Alps extent and a raster of a random resolution
data(geo_dat)
data(exrst)

# Apply the function by infering 50 classes of environments
my.eco = make_ecoregion(rst,50)
plot(my.eco)

# Downloading in the European Alps the observations of one plant species
obs.arcto = get_gbif("Arctostaphylos alpinus",geo=shp.lonlat)

# Create the range map based on our custom ecoregion
range.arcto = get_range("Arctostaphylos alpinus",obs.arcto,my.eco,"EcoRegion",res=20)

# Plot
plot(vect(shp.lonlat))
plot(range.arcto,add=TRUE,col="darkgreen")
points(obs.arcto[,c("decimalLongitude","decimalLatitude")],pch=20,col="#238b4550",cex=1)
```

make_tiles *Create a specific number of tiles based on a raster extent*

Description

Based on a specific extent, one or several tiles are generated. Tiles can be smaller raster extents or geometry arguments POLYGON(). The original extent is therefore either converted into a POLYGON() argument, or divided into Ntiles of regular fragments which are converted into POLYGON() arguments and smaller SpatExtent.

Usage

```
make_tiles(geo, Ntiles, sext = TRUE)
```

Arguments

geo	Object of class Extent, SpatExtent, SpatialPolygon, SpatialPolygonDataframe, or SpaVector (WGS84 or planar) to define the study's area extent. Default is NULL i.e. the whole globe.
Ntiles	Numeric. In how many tiles/fragments should geo be divided approximately?
sext	Logical. Should a list of SpatExtent also be returned for each generated POLYGON()?

Value

A list of geometry arguments POLYGON() of length Ntiles (and of SpatExtent if sext=TRUE).

References

Chauvier, Y., Thuiller, W., Brun, P., Lavergne, S., Descombes, P., Karger, D. N., ... & Zimmermann, N. E. (2021). Influence of climate, soil, and land cover on plant species distribution in the European Alps. Ecological monographs, 91(2), e01433. 10.1002/ecm.1433

Examples

```
# Load the European Alps Extent
data(geo_dat)

# Apply the function to divide the extent in ~20 fragments
mt = make_tiles(geo=shp.lonlat,Ntiles=20,sext=TRUE); mt
```

obs_filter *Filter a set of GBIF observations according to a defined grain*

Description

Whereas the 'grain' parameter in get_gbif() allows GBIF observations to be filtered according to a certain spatial precision, obs_filter() accepts as input a get_gbif() output (one or several species) and filter the observations according to a given grid resolution (one observation per pixel grid kept). This function allows the user to refine the density of GBIF observations according to a defined analysis/study's resolution.

Usage

```
obs_filter(get.gbif, grid)
```

Arguments

get.gbif	One get_gbif() output including one or several species. Note that if GBIF absences are kept in the output(s), the function should be used distinctively for observations and absences.
grid	Object of class SpatRaster, RasterLayer, RasterBrick or RasterStack of desired resolution and extent (WGS84).

Value

Data frame with two columns named 'x' and 'y' comprising the new set of observations filtered at grid resolution.

Examples

```
# Load the European Alps extent and a raster of a random resolution
data(geo_dat)
data(exrst)

# Downloading in the European Alps the observations of two plant species
obs.arcto = get_gbif("Arctostaphylos alpinus",geo=shp.lonlat)
obs.saxi = get_gbif("Saxifraga cernua",geo=shp.lonlat)
plot(vect(shp.lonlat))
points(obs.arcto[,c("decimalLongitude","decimalLatitude")],pch=20,col="#238b4550",cex=1)
points(obs.saxi[,c("decimalLongitude","decimalLatitude")],pch=20,col="#99000d50",cex=1)

# rbind both datasets
both.sp = rbind(obs.arcto,obs.saxi)

# Run function
obs.filt = obs_filter(both.sp,rst)

# Check new points
x11();plot(vect(shp.lonlat))
points(obs.filt[obs.filt$Species%in%"Arctostaphylos alpinus",c("x","y")],pch=20,col="#238b4550",cex=1)
points(obs.filt[obs.filt$Species%in%"Saxifraga cernua",c("x","y")],pch=20,col="#99000d50",cex=1)
```

Index