Package: b3gbi (via r-universe)

February 20, 2025

Type Package

Title General Biodiversity Indicators for Biodiversity Data Cubes

Version 0.4.2

Description Calculate general biodiversity indicators from GBIF data cubes. Includes many common indicators such as species richness and evenness, which can be calculated over time (trends) or space (maps).

License MIT + file LICENSE

URL https://github.com/b-cubed-eu/b3gbi, https://b-cubed-eu.github.io/b3gbi/

Depends R (>= 3.5.0),

Imports boot, dplyr, furrr, future, ggplot2, iNEXT, labeling, magrittr, mgrs (>= 0.2.4), patchwork, permute, purrr, readr, rlang, rnaturalearth, sf, stringr, tibble, tidyr, units

Suggests bold (>= 1.3.0), knitr, rmarkdown, rnaturalearthdata, taxize (>= 0.9.99), testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

VignetteBuilder knitr, rmarkdown

Additional_repositories https://b-cubed-eu.r-universe.dev, https://ropensci.r-universe.dev

Remotes github::hrbrmstr/mgrs, github::ropensci/bold, github::ropensci/taxize

Config/pak/sysreqs libgdal-dev gdal-bin libgeos-dev libicu-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev libx11-dev

Repository https://b-cubed-eu.r-universe.dev

RemoteUrl https://github.com/b-cubed-eu/b3gbi

RemoteRef HEAD

RemoteSha bb3e83543d122157060ebeed56eef3b24a58ebc4

Contents

`ab_rarity_map` *Calculate Abundance-Based Rarity Over Space*

Description

This function calculates abundance-based rarity over a gridded map.

Usage

```
ab_rarity_map(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow`
	`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	`region` The region of interest (e.g., "Europe"). (Default: "Europe")
	`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	`first_year` Exclude data before this year. (Uses all data in the cube by default.)
	`last_year` Exclude data after this year. (Uses all data in the cube by default.)

spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'ab_rarity' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
abr_map <- ab_rarity_map(example_cube_1, level = "country", region = "Denmark")
plot(abr_map)

## End(Not run)
```

---

ab_rarity_ts *Calculate Abundance-Based Rarity Over Time*

Description

This function calculates abundance-based rarity over time.

Usage

```
ab_rarity_ts(data, ...)
```

Arguments

data A data cube object (class 'processed_cube').

... Arguments passed on to compute_indicator_workflow

ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)

cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)

level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')

region The region of interest (e.g., "Europe"). (Default: "Europe")

`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")

`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")

`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)

`first_year` Exclude data before this year. (Uses all data in the cube by default.)

`last_year` Exclude data after this year. (Uses all data in the cube by default.)

`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'ab_rarity' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
abr_ts <- ab_rarity_ts(example_cube_1, first_year = 1985)
plot(abr_ts)

## End(Not run)
```

`area_rarity_map` *Calculate Area-Based Rarity Over Space*

Description

This function calculates area-based rarity over a gridded map.

Usage

```
area_rarity_map(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'area_rarity' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
arr_map <- area_rarity_map(example_cube_1, level = "country", region = "Denmark")
plot(arr_map)

## End(Not run)
```

area_rarity_ts *Calculate Area-Based Rarity Over Time*

Description

This function calculates area-based rarity over time.

Usage

```
area_rarity_ts(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'area_rarity' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
arr_ts <- area_rarity_ts(example_cube_1, first_year = 1985)
plot(arr_ts)

## End(Not run)
```

`available_indicators` *Indicators Available for Use in the Package*

Description

A list of all biodiversity indicators available within the package, along with the dimensions they can be calculated across, the functions to access them, and any special arguments

Usage

```
available_indicators
```

Format

A special object of class 'available_indicators' containing a list of indicators and six fields with information about them

indicator_class class of the indicator

indicator_name name of the indicator

plot_title title to be used when plotting with automated title generation

legend_label title to be used when plotting with automated legend title generation

legend_transformation any transformation to perform on the legend when plotting, to improve visualization of maps

map_wrapper wrapper function to use when calculating indicator as a map

ts_wrapper wrapper function to use when calculating indicator as a time series

map_function_arguments any special arguments to consider when using the function to calculate an indicator map

ts_function_arguments any special arguments to consider when using the function to calculate an indicator time series

Source

N/A

calc_ci *Calculate Confidence Intervals for a Biodiversity Indicator*

Description

This function calculates bootstrap confidence intervals for a biodiversity indicator. It is called automatically when calculating a biodiversity indicator over time unless you choose 'none' for ci_type.

Usage

```
calc_ci(x, indicator, ...)

## S3 method for class 'total_occ'
calc_ci(x, indicator, num_bootstrap = 1000, ci_type = ci_type, ...)

## S3 method for class 'occ_density'
calc_ci(x, indicator, num_bootstrap = 1000, ci_type = ci_type, ...)

## S3 method for class 'newness'
calc_ci(x, indicator, num_bootstrap = 1000, ci_type = ci_type, ...)

## S3 method for class 'williams_evenness'
calc_ci(x, ...)

## S3 method for class 'pielou_evenness'
calc_ci(x, ...)

## S3 method for class 'ab_rarity'
calc_ci(x, indicator, num_bootstrap = 1000, ci_type = ci_type, ...)

## S3 method for class 'area_rarity'
calc_ci(x, indicator, num_bootstrap = 1000, ci_type = ci_type, ...)

## S3 method for class 'spec_occ'
calc_ci(x, indicator, num_bootstrap = 1000, ci_type = ci_type, ...)

## S3 method for class 'spec_range'
calc_ci(x, indicator, num_bootstrap = 1000, ci_type = ci_type, ...)

## S3 method for class 'tax_distinct'
calc_ci(
  x,
  indicator,
  num_bootstrap = 1000,
  ci_type = ci_type,
  set_rows = 1,
  ...
```

```
)
```

Arguments

x	A data cube object
indicator	An indicator calculated over time, in the form of a data frame. *Note: this should NOT be an 'indicator_ts' object as it is meant to be called by the 'compute_indicator_workflow' function.
...	Additional arguments passed to specific indicator calculation functions.
num_bootstrap	Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)
ci_type	Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
set_rows	Automatically select which taxonomic information to keep when there are multiple options. Default value of 1 keeps the first option, which is usually the best.

---

calc_map.obs_richness *Calculate Biodiversity Indicators Over Space*

Description

This function provides a flexible framework for calculating various biodiversity indicators on a spatial grid or as a time series. It prepares the data, creates a grid, calculates indicators, and formats the output into an appropriate S3 object ('indicator_map'). Specific implementations for different indicator types are provided using the appropriate wrappers.

Usage

```
## S3 method for class 'obs_richness'
calc_map(x, ...)

## S3 method for class 'total_occ'
calc_map(x, ...)

## S3 method for class 'newness'
calc_map(x, newness_min_year = NULL, ...)

## S3 method for class 'occ_density'
calc_map(x, ...)

## S3 method for class 'williams_evenness'
calc_map(x, ...)

## S3 method for class 'pielou_evenness'
calc_map(x, ...)
```

```
## S3 method for class 'ab_rarity'
calc_map(x, ...)

## S3 method for class 'area_rarity'
calc_map(x, ...)

## S3 method for class 'spec_occ'
calc_map(x, ...)

## S3 method for class 'spec_range'
calc_map(x, ...)

## S3 method for class 'tax_distinct'
calc_map(x, ...)

calc_map(x, ...)
```

Arguments

x	A data cube object ('processed_cube').
...	Additional arguments passed to specific indicator calculation functions.
newness_min_year	If set, only shows values above this (e.g. 1970). Values below the minimum will be replaced with NA. This can be useful e.g. if you have outlier cells where the data is very old causing the legend gradient to stretch in a way that makes other cell values difficult to discern.

Value

An S3 object of the class 'indicator_map' containing the calculated indicator values and metadata.

Examples

```
observed_richness_map <- obs_richness_map(example_cube_1, level = "country", region = "Denmark")
plot(observed_richness_map)
```

---

calc_ts.obs_richness *Calculate Biodiversity Indicators Over Time*

Description

This function provides a flexible framework for calculating various biodiversity indicators over time. It prepares the data, creates a grid, calculates indicators, and formats the output into an appropriate S3 object ('indicator_ts'). Specific implementations for different indicator types are provided using the appropriate wrappers.

Usage

```
## S3 method for class 'obs_richness'
calc_ts(x, ...)

## S3 method for class 'cum_richness'
calc_ts(x, ...)

## S3 method for class 'total_occ'
calc_ts(x, ...)

## S3 method for class 'occ_density'
calc_ts(x, ...)

## S3 method for class 'newness'
calc_ts(x, ...)

## S3 method for class 'williams_evenness'
calc_ts(x, ...)

## S3 method for class 'pielou_evenness'
calc_ts(x, ...)

## S3 method for class 'ab_rarity'
calc_ts(x, ...)

## S3 method for class 'area_rarity'
calc_ts(x, ...)

## S3 method for class 'spec_occ'
calc_ts(x, ...)

## S3 method for class 'spec_range'
calc_ts(x, ...)

## S3 method for class 'tax_distinct'
calc_ts(x, set_rows = 1, ...)

## S3 method for class 'occ_turnover'
calc_ts(x, ...)

calc_ts(x, ...)
```

Arguments

x	A data cube object ('processed_cube').
...	Additional arguments passed to specific indicator calculation functions.
set_rows	Automatically select which taxonomic information to keep when there are multiple options. Default value of 1 keeps the first option, which is usually the best.

Value

An S3 object of the class 'indicator_ts' containing the calculated indicator values and metadata.

Examples

```
occurrence_density_trend <- occ_density_ts(example_cube_1)
plot(occurrence_density_trend, min_year=1980)
```

`compute_indicator_workflow` *Calculate Biodiversity Indicators Over Space or Time*

Description

This function provides a flexible framework for calculating various biodiversity indicators on a spatial grid or as a time series. It prepares the data, creates a grid, calculates indicators, and formats the output into an appropriate S3 object ('indicator_map' or 'indicator_ts').

Usage

```
compute_indicator_workflow(
  data,
  type,
  dim_type = c("map", "ts"),
  ci_type = c("norm", "basic", "perc", "bca", "none"),
  cell_size = NULL,
  level = c("cube", "continent", "country", "world", "sovereignty", "geounit"),
  region = "Europe",
  ne_type = c("countries", "map_units", "sovereignty", "tiny_countries"),
  ne_scale = c("medium", "small", "large"),
  output_crs = NULL,
  first_year = NULL,
  last_year = NULL,
  spherical_geometry = TRUE,
  make_valid = FALSE,
  num_bootstrap = 1000,
  ...
)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`type`	The indicator to calculate. Supported options include: • 'obs_richness': Observed species richness. • 'total_occ': Total number of occurrences.

- 'newness': Mean year of occurrence.
- 'occ_density': Density of occurrences.
- 'williams_evenness', 'pielou_evenness': Evenness measures.
- 'ab_rarity', 'area_rarity': Abundance-based and area-based rarity scores.
- 'cum_richness': Cumulative species richness.
- 'occ_turnover': Occupancy turnover.
- 'spec_range': Species range size.
- 'spec_occ': Species occurrences.
- 'tax_distinct': Taxonomic distinctness.
- 'hill0': Species richness (estimated by coverage-based rarefaction).
- 'hill1': Hill-Shannon diversity (estimated by coverage-based rarefaction).
- 'hill2': Hill-Simpson diversity (estimated by coverage-based rarefaction).

`dim_type` Dimension to calculate indicator over (time: 'ts', or space: 'map')

`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)

`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)

`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')

`region` The region of interest (e.g., "Europe"). (Default: "Europe")

`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")

`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")

`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)

`first_year` Exclude data before this year. (Uses all data in the cube by default.)

`last_year` Exclude data after this year. (Uses all data in the cube by default.)

`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

`...` Additional arguments passed to specific indicator calculation functions.

Value

An S3 object containing the calculated indicator values and metadata.

Examples

```
diversity_map <- compute_indicator_workflow(example_cube_1,
                                            type = "obs_richness",
                                            dim_type = "map",
                                            level = "country",
                                            region = "Denmark")
diversity_map
```

`cum_richness_ts` *Calculate Cumulative Species Richness*

Description

This function calculates cumulative species richness as a time series.

Usage

```
cum_richness_ts(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow`
	`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	`region` The region of interest (e.g., "Europe"). (Default: "Europe")
	`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	`first_year` Exclude data before this year. (Uses all data in the cube by default.)
	`last_year` Exclude data after this year. (Uses all data in the cube by default.)
	`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'cum_richness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
cr_ts <- cum_richness_ts(example_cube_1, first_year = 1985)
plot(cr_ts)

## End(Not run)
```

---

example_cube_1 *Cube of GBIF Mammal Occurrences in Denmark*

Description

Small example cube containing the mammal occurrences in Denmark available on GBIF as of 16.03.2024.

Usage

```
example_cube_1
```

Format

A 'processed_cube' object containing a tibble with 276,950 rows and 10 variables, as well as metadata

year year occurrence was recorded

eea_cell_code code containing the cell resolution and coordinates on the EEA grid

taxonKey taxonomic key associated with the species on GBIF

obs number of individuals observed

scientificName scientific species name

rank taxonomic rank

kingdom kingdom

xcoord East-West coordinate on the EEA grid

ycoord North-South coordinate on the EEA grid

resolution grid cell size

Source

https://doi.org/10.15468/dl.5mb887

Examples

```
{
  ## Not run:
    # Basic example of how to use the dataset
    denmark_mammals_or_map <- obs_richness_map(example_cube_1,
                                               level = "country",
                                               region = "Denmark")
    plot(denmark_mammals_or_map,
         title = "Mammals in Denmark (1751-2023): Observed Species Richness")

## End(Not run)
}
```

---

example_indicator_ts1 *Time Series of Observed Species Richness for Mammals in Denmark*

Description

Example indicator containing a time series of observed species richness for mammal occurrences in Denmark (occurrences from GBIF: 16.03.2024).

Usage

```
example_indicator_ts1
```

Format

An 'indicator_ts' object containing a tibble with 54 rows and 2 variables, as well as metadata

year a year the indicator was calculated for

diversity_val calculated richness value for the year

Source

https://doi.org/10.15468/dl.qk4f2z

example_indicator_ts2 *Time Series of Cumulative Species Richness for Insects in Europe*

Description

Example indicator containing a time series of cumulative species richness for insect occurrences in Europe (occurrences from GBIF: 16.03.2024).

Usage

```
example_indicator_ts2
```

Format

An 'indicator_ts' object containing a tibble with 89 rows and 2 variables, as well as metadata

year a year the indicator was calculated for

diversity_val calculated richness value for the year

Source

https://doi.org/10.15468/dl.qk4f2z

get_bootstrap_ci *Calculate confidence intervals for list of* boot *objects*

Description

This function calculates confidence intervals for a list of objects of class "boot" per year into a dataframe containing all required summaries.

Usage

```
get_bootstrap_ci(bootstrap_list, ..., temporal_list_name = "year")
```

Arguments

bootstrap_list	A list of objects of class "boot" per year.
...	Additional argument to be passed to the boot::boot.ci() function.
temporal_list_name	The temporal list names of bootstrap_list (e.g., year, month ...) containing time point values. Default year.

Value

The returned value is a dataframe containing the time point, the type of interval (int_type), the lower limit of the confidence interval (ll), the upper limit of the confidence interval (ul), and the confidence level of the intervals (conf_level).

get_observed_years *Extract Years With Observations from an Indicator Map*

Description

Takes an "indicator_map", "indicator_ts", or "processed_cube" object and determines the years for which observation data exists.

Usage

```
get_observed_years(x)
```

Arguments

x	An "indicator_map" object containing calculated indicator values associated with grid cells.

Value

A data frame with two columns:

- years: A sequence of years covering the range of observations.
- occurrences: A logical vector indicating if observations exist for each year (TRUE if present, FALSE if absent).

Examples

```
total_occ_mammals_denmark <- total_occ_map(example_cube_1, level = "country", region = "Denmark")
get_observed_years(total_occ_mammals_denmark)
```

hill0_map *Calculate Estimated Species Richness Over Space*

Description

This function uses coverage-based methods to estimate species richness over a gridded map.

Usage

```
hill0_map(data, coverage = 0.95, cutoff_length = 5, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`coverage`	The sample coverage value for the estimator. Default is 0.95.
`cutoff_length`	The minimum number of data points for each grid cell. Grid cells with fewer data points will be removed before calculations to avoid errors. Default is 5.
`...`	Arguments passed on to `compute_indicator_workflow` `ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.) `cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world) `level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube') `region` The region of interest (e.g., "Europe"). (Default: "Europe") `ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries") `ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium") `output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system) `first_year` Exclude data before this year. (Uses all data in the cube by default.) `last_year` Exclude data after this year. (Uses all data in the cube by default.) `spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE) `make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE). `num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'hill0' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
h0_map <- hill0_map(example_cube_1, level = "country", region = "Denmark")
plot(h0_map)

## End(Not run)
```

`hill0_ts` *Calculate Estimated Species Richness Over Time*

Description

This function uses coverage-based methods to estimate species richness over time.

Usage

```
hill0_ts(data, coverage = 0.95, cutoff_length = 5, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`coverage`	The sample coverage value for the estimator. Default is 0.95.
`cutoff_length`	The minimum number of data points for each year. Years with fewer data points will be removed before calculations to avoid errors. Default is 5.
`...`	Arguments passed on to `compute_indicator_workflow` `ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.) `cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world) `level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube') `region` The region of interest (e.g., "Europe"). (Default: "Europe") `ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries") `ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium") `output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system) `first_year` Exclude data before this year. (Uses all data in the cube by default.) `last_year` Exclude data after this year. (Uses all data in the cube by default.) `spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE) `make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE). `num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'hill0' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
h0_ts <- hill0_ts(example_cube_1, first_year = 1985)
plot(h0_ts)

## End(Not run)
```

`hill1_map` *Calculate Hill-Shannon Diversity Over Space*

Description

This function uses coverage-based methods to estimate Hill-Shannon Diversity over a gridded map.

Usage

```
hill1_map(data, cutoff_length = 5, coverage = 0.95, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`cutoff_length`	The minimum number of data points for each grid cell. Grid cells with fewer data points will be removed before calculations to avoid errors. Default is 5.
`coverage`	The sample coverage value for the estimator. Default is 0.95.
`...`	Arguments passed on to `compute_indicator_workflow` `ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.) `cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world) `level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube') `region` The region of interest (e.g., "Europe"). (Default: "Europe") `ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries") `ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium") `output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system) `first_year` Exclude data before this year. (Uses all data in the cube by default.) `last_year` Exclude data after this year. (Uses all data in the cube by default.)

spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'hill1' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
h1_map <- hill1_map(example_cube_1, level = "country", region = "Denmark")
plot(h1_map)

## End(Not run)
```

---

hill1_ts *Calculate Hill-Shannon Diversity Over Time*

Description

This function uses coverage-based methods to estimate Hill-Shannon Diversity over time.

Usage

```
hill1_ts(data, cutoff_length = 5, coverage = 0.95, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
cutoff_length	The minimum number of data points for each year. Years with fewer data points will be removed before calculations to avoid errors. Default is 5.
coverage	The sample coverage value for the estimator. Default is 0.95.
...	Arguments passed on to compute_indicator_workflow

ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)

cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)

`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')

`region` The region of interest (e.g., "Europe"). (Default: "Europe")

`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")

`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")

`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)

`first_year` Exclude data before this year. (Uses all data in the cube by default.)

`last_year` Exclude data after this year. (Uses all data in the cube by default.)

`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'hill1' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
h1_ts <- hill1_ts(example_cube_1, first_year = 1985)
plot(h1_ts)

## End(Not run)
```

`hill2_map` *Calculate Hill-Simpson Diversity Over Space*

Description

This function uses coverage-based methods to estimate Hill-Simpson Diversity over a gridded map.

Usage

```
hill2_map(data, cutoff_length = 5, coverage = 0.95, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`cutoff_length`	The minimum number of data points for each grid cell. Grid cells with fewer data points will be removed before calculations to avoid errors. Default is 5.
`coverage`	The sample coverage value for the estimator. Default is 0.95.
`...`	Arguments passed on to `compute_indicator_workflow` `ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.) `cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world) `level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube') `region` The region of interest (e.g., "Europe"). (Default: "Europe") `ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries") `ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium") `output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system) `first_year` Exclude data before this year. (Uses all data in the cube by default.) `last_year` Exclude data after this year. (Uses all data in the cube by default.) `spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE) `make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE). `num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'hill2' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
h2_map <- hill2_map(example_cube_1, level = "country", region = "Denmark")
plot(h2_map)

## End(Not run)
```

`hill2_ts` *Calculate Hill-Simpson Diversity Over Time*

Description

This function uses coverage-based methods to estimate Hill-Simpson Diversity over time.

Usage

```
hill2_ts(data, cutoff_length = 5, coverage = 0.95, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`cutoff_length`	The minimum number of data points for each year. Years with fewer data points will be removed before calculations to avoid errors. Default is 5.
`coverage`	The sample coverage value for the estimator. Default is 0.95.
`...`	Arguments passed on to `compute_indicator_workflow` `ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.) `cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world) `level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube') `region` The region of interest (e.g., "Europe"). (Default: "Europe") `ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries") `ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium") `output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system) `first_year` Exclude data before this year. (Uses all data in the cube by default.) `last_year` Exclude data after this year. (Uses all data in the cube by default.) `spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE) `make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE). `num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'hill2' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
h2_ts <- hill2_ts(example_cube_1, first_year = 1985)
plot(h2_ts)

## End(Not run)
```

`list_species` *Extract Species Names.*

Description

Retrieves a list of species names from a biodiversity data object ('processed_cube', 'indicator_map' or 'indicator_ts').

Usage

```
list_species(object)
```

Arguments

`object`	A biodiversity data object containing species names, either as a separate vector called species_names or as a column called scientificName.

Value

Either a character vector of species names (if directly available) or a data frame with columns:

- `taxonKey`: A unique identifier for each species.
- `scientificName`: The scientific name of each species.

Examples

```
list_species(example_cube_1)
```

`newness_map` *Calculate Mean Year of Occurrence Over Space*

Description

This function estimates the relative newness of records in a data cube by calculating the mean year of occurrence over a gridded map.

Usage

```
newness_map(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow`
	`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	`region` The region of interest (e.g., "Europe"). (Default: "Europe")
	`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	`first_year` Exclude data before this year. (Uses all data in the cube by default.)
	`last_year` Exclude data after this year. (Uses all data in the cube by default.)
	`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'newness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
n_map <- newness_map(example_cube_1, level = "country", region = "Denmark")
plot(n_map)

## End(Not run)
```

`newness_ts` *Calculate Mean Year of Occurrence Over Time*

Description

This function estimates the change in relative newness of records in a data cube over time by calculating the mean year of occurrence as a time series.

Usage

```
newness_ts(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow`
	`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	`region` The region of interest (e.g., "Europe"). (Default: "Europe")
	`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	`first_year` Exclude data before this year. (Uses all data in the cube by default.)
	`last_year` Exclude data after this year. (Uses all data in the cube by default.)
	`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'newness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
n_ts <- newness_ts(example_cube_1, first_year = 1985)
plot(n_ts)

## End(Not run)
```

---

obs_richness_map *Calculate Observed Species Richness Over Space*

Description

This function calculates observed species richness over a gridded map.

Usage

```
obs_richness_map(data, ...)
```

Arguments

data A data cube object (class 'processed_cube').

... Arguments passed on to compute_indicator_workflow

ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)

cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)

level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')

region The region of interest (e.g., "Europe"). (Default: "Europe")

ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")

`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")

`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)

`first_year` Exclude data before this year. (Uses all data in the cube by default.)

`last_year` Exclude data after this year. (Uses all data in the cube by default.)

`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'obs_richness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
or_map <- obs_richness_map(example_cube_1, level = "country", region = "Denmark")
plot(or_map)

## End(Not run)
```

`obs_richness_ts` *Calculate Observed Species Richness Over Time*

Description

This function calculates observed species richness as a time series.

Usage

```
obs_richness_ts(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'obs_richness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
or_ts <- obs_richness_ts(example_cube_1, first_year = 1985)
plot(or_ts)

## End(Not run)
```

occ_density_map *Calculate Occurrence Density Over Space*

Description

This function calculates the density of records over a gridded map.

Usage

```
occ_density_map(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'occ_density' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
od_map <- occ_density_map(example_cube_1, level = "country", region = "Denmark")
plot(od_map)

## End(Not run)
```

`occ_density_ts` *Calculate Occurrence Density Over Time*

Description

This function calculates density of records as a time series.

Usage

```
occ_density_ts(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow`
	`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	`region` The region of interest (e.g., "Europe"). (Default: "Europe")
	`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	`first_year` Exclude data before this year. (Uses all data in the cube by default.)
	`last_year` Exclude data after this year. (Uses all data in the cube by default.)
	`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'occ_density' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
od_ts <- occ_density_ts(example_cube_1, first_year = 1985)
plot(od_ts)

## End(Not run)
```

---

occ_turnover_ts *Calculate Occupancy Turnover Over Time*

Description

This function calculates occupancy turnover as a time series.

Usage

```
occ_turnover_ts(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow

ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)

cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)

level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')

region The region of interest (e.g., "Europe"). (Default: "Europe")

ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")

`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")

`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)

`first_year` Exclude data before this year. (Uses all data in the cube by default.)

`last_year` Exclude data after this year. (Uses all data in the cube by default.)

`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'occ_turnover' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
ot_ts <- occ_turnover_ts(example_cube_1, first_year = 1985)
plot(ot_ts)
```

`pielou_evenness_map` *Calculate Pielou's Evenness Over Space*

Description

This function calculates Pielou's evenness over a gridded map.

Usage

```
pielou_evenness_map(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'pielou_evenness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
pe_map <- pielou_evenness_map(example_cube_1, level = "country", region = "Denmark")
plot(pe_map)

## End(Not run)
```

`pielou_evenness_ts` *Calculate Pielou's Evenness Over Time*

Description

This function calculates Pielou's evenness over time.

Usage

```
pielou_evenness_ts(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow`
	`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	`region` The region of interest (e.g., "Europe"). (Default: "Europe")
	`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	`first_year` Exclude data before this year. (Uses all data in the cube by default.)
	`last_year` Exclude data after this year. (Uses all data in the cube by default.)
	`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'pielou_evenness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
pe_ts <- pielou_evenness_ts(example_cube_1, first_year = 1985)
plot(pe_ts)

## End(Not run)
```

---

plot_map *Plot Biodiversity Indicator Map*

Description

Creates a map visualization of a calculated biodiversity indicator, providing customization options.

Usage

```
plot_map(
  x,
  title = "auto",
  auto_title = NULL,
  leg_label_default = NULL,
  xlims = NULL,
  ylims = NULL,
  trans = NULL,
  breaks = NULL,
  labels = NULL,
  Europe_crop_EEA = TRUE,
  crop_to_grid = FALSE,
  surround = TRUE,
  panel_bg = NULL,
  land_fill_colour = NULL,
  legend_title = NULL,
  legend_limits = NULL,
  legend_title_wrap_length = 10,
  title_wrap_length = 60
)
```

Arguments

x	An 'indicator_map' object containing indicator values associated with map grid cells.
title	Plot title. Replace "auto" with your own title if you want a custom title or if calling the function manually.

auto_title Text for automatic title generation, provided by an appropriate S3 method (if calling the function manually, leave as NULL).

leg_label_default
Default label for the legend, provided by an appropriate S3 method (if calling the function manually, leave as NULL).

xlims (Optional) Custom x-axis limits.

ylims (Optional) Custom y-axis limits.

trans (Optional) Scale transformation for the fill gradient (e.g., 'log').

breaks (Optional) Break points for the legend scale.

labels (Optional) Labels for legend scale break points.

Europe_crop_EEA
If TRUE, crops maps of Europe using the EPSG:3035 CRS to exclude far-lying islands (default is TRUE, but does not affect other maps or projections). Will not work if crop_to_grid is set to TRUE.

crop_to_grid If TRUE, the grid will determine the edges of the map.Overrides Europe_crop_EEA. Default is FALSE.

surround If TRUE, includes surrounding land area in gray when plotting at the country or continent level. If FALSE, all surrounding area will be colored ocean blue (or whatever colour you set manually using panel_bg). Default is TRUE.

panel_bg (Optional) Background colour for the map panel.

land_fill_colour
(Optional) Colour for the land area outside of the grid (if surround = TRUE). Default is "grey85".

legend_title (Optional) Title for the plot legend.

legend_limits (Optional) Limits for the legend scale.

legend_title_wrap_length
Maximum legend title length before wrapping to a new line.

title_wrap_length
Maximum title length before wrapping to a new line.

Value

A ggplot object representing the biodiversity indicator map. Can be customized using ggplot2 functions.

Examples

```
evenness_map <- pielou_evenness_map(example_cube_1,
                                    level = "country",
                                    region = "Denmark")
plot_map(x = evenness_map,
         title = "Map of Species Evenness in Denmark",
         legend_title = "Evenness")
```

plot_species_map *Plot Occurrence Map or Range Map of Individual Species*

Description

Creates map visualizations of species ranges or species occurrences, providing customization options.

Usage

```
plot_species_map(
  x,
  species = NULL,
  leg_label_default = NULL,
  auto_title = NULL,
  suppress_legend = FALSE,
  title = "auto",
  xlims = NULL,
  ylims = NULL,
  trans = NULL,
  breaks = NULL,
  labels = NULL,
  Europe_crop_EEA = TRUE,
  crop_to_grid = FALSE,
  surround = TRUE,
  single_plot = TRUE,
  panel_bg = NULL,
  land_fill_colour = NULL,
  legend_title = NULL,
  legend_limits = NULL,
  legend_title_wrap_length = 10,
  title_wrap_length = 60
)
```

Arguments

x	An 'indicator_map' object containing indicator values associated with map grid cells.
species	Species you want to map occurrences for. Can be either numerical taxonKeys or species names. Partial species names can be used (the function will try to match them).
leg_label_default	Default label for the legend, provided by an appropriate S3 method (if calling the function manually, leave as NULL).
auto_title	Text for automatic title generation, provided by an appropriate S3 method (if calling the function manually, leave as NULL).

suppress_legend
: Do not show legend. This should be set to true when plotting species ranges, as all cell values are 1.

title
: Plot title. Replace "auto" with your own title if you want a custom title or if calling the function manually.

xlims
: (Optional) Custom x-axis limits.

ylims
: (Optional) Custom y-axis limits.

trans
: (Optional) Scale transformation for the fill gradient (e.g., 'log').

breaks
: (Optional) Break points for the legend scale.

labels
: (Optional) Labels for legend scale break points.

Europe_crop_EEA
: If TRUE, crops maps of Europe using the EPSG:3035 CRS to exclude far-lying islands (default is TRUE, but does not affect other maps or projections).

crop_to_grid
: If TRUE, the grid will determine the edges of the map.Overrides Europe_crop_EEA. Default is FALSE.

surround
: If TRUE, includes surrounding land area in gray when plotting at the country or continent level. If FALSE, all surrounding area will be coloured ocean blue (or whatever colour you set manually using panel_bg). Default is TRUE.

single_plot
: By default all species occurrence time series will be combined into a single multi-panel plot. Set this to FALSE to plot each species separately.

panel_bg
: (Optional) Background colour for the map panel.

land_fill_colour
: (Optional) Colour for the land area outside of the grid (if surround = TRUE). Default is "grey85".

legend_title
: (Optional) Title for the plot legend.

legend_limits
: (Optional) Limits for the legend scale.

legend_title_wrap_length
: Maximum legend title length before wrapping to a new line.

title_wrap_length
: Maximum title length before wrapping to a new line.

Value

A ggplot object representing the map of species range or occurrences. Can be customized using ggplot2 functions.

Examples

```
spec_occ_mammals_denmark <- spec_occ_map(example_cube_1,
                                  level = "country",
                                  region = "Denmark")
plot_species_map(x = spec_occ_mammals_denmark, c(2440728, 4265185))
```

plot_species_ts *Plot Occurrence Trends or Range Size Trends for Individual Species*

Description

Creates time series plots of species occurrences or species range sizes, with an optional smoothed trendline, and visualizes uncertainty.

Usage

```
plot_species_ts(
  x,
  species = NULL,
  single_plot = TRUE,
  min_year = NULL,
  max_year = NULL,
  title = "auto",
  auto_title = NULL,
  y_label_default = NULL,
  suppress_y = FALSE,
  smoothed_trend = TRUE,
  linecolour = NULL,
  linealpha = 0.8,
  ribboncolour = NULL,
  ribbonalpha = 0.2,
  error_alpha = 1,
  trendlinecolour = NULL,
  trendlinealpha = 0.5,
  envelopecolour = NULL,
  envelopealpha = 0.2,
  smooth_cialpha = 1,
  point_line = c("point", "line"),
  pointsize = 2,
  linewidth = 1,
  ci_type = c("error_bars", "ribbon"),
  error_width = 1,
  error_thickness = 1,
 smooth_linetype = c("solid", "dashed", "dotted", "dotdash", "longdash", "twodash"),
  smooth_linewidth = 1,
  smooth_cilinewidth = 1,
  gridoff = FALSE,
  x_label = NULL,
  y_label = NULL,
  x_expand = NULL,
  y_expand = NULL,
  x_breaks = 10,
  y_breaks = 6,
```

```
  title_wrap_length = 60
)
```

Arguments

x	An 'indicator_ts' object containing time series of indicator values matched to species names and/or taxon keys.
species	Species you want to map occurrences for. Can be either numerical taxonKeys or species names. Partial species names can be used (the function will try to match them).
single_plot	By default all species occurrence time series will be combined into a single multi-panel plot. Set this to FALSE to plot each species separately.
min_year	(Optional) Earliest year to include in the plot.
max_year	(Optional) Latest year to include in the plot.
title	Plot title. Replace "auto" with your own title if you want a custom title or if calling the function manually.
auto_title	Text for automatic title generation, provided by an appropriate S3 method (if calling the function manually, leave as NULL).
y_label_default	Default label for the y-axis, provided by an appropriate S3 method (if calling the function manually, leave as NULL).
suppress_y	If TRUE, suppresses y-axis labels.
smoothed_trend	If TRUE, plot a smoothed trendline over time (`stats::loess()`).
linecolour	(Optional) Colour for the indicator line or points. Default is darkorange.
linealpha	Transparency for indicator line or points. Default is 0.8.
ribboncolour	(Optional) Colour for the bootstrapped confidence intervals. Default is goldenrod1. Set to "NA" if you don't want to plot the CIs.
ribbonalpha	Transparency for indicator confidence interval ribbon (if ci_type = "ribbon"). Default is 0.2.
error_alpha	Transparency for indicator error bars (if ci_type = "error_bar"). Default is 1.
trendlinecolour	(Optional) Colour for the smoothed trendline. Default is blue.
trendlinealpha	Transparency for the smoothed trendline. Default is 0.5.
envelopecolour	(Optional) Colour for the uncertainty envelope. Default is lightsteelblue.
envelopealpha	Transparency for the smoothed trendline envelope. Default is 0.2.
smooth_cialpha	Transparency for the smoothed lines forming the edges of the trendline envelope. Default is 1.
point_line	Whether to plot the indicator as a line or a series of points. Options are "line" or "point". Default is "point".
pointsize	Size of the points if point_line = "point". Default is 2.
linewidth	Width of the line if point_line = "line". Default is 1.

ci_type Whether to plot bootstrapped confidence intervals as a "ribbon" or "error_bars". Default is "error_bars".

error_width Width of error bars if ci_type = "error_bars". Default is 1. Note that unlike the default 'width' parameter in geom_errorbar, 'error_width' is NOT dependent on the number of data points in the plot. It is automatically scaled to account for this. Therefore the width you select will be consistent relative to the plot width even if you change 'min_year' and 'max_year'.

error_thickness
Thickness of error bars if ci_type = "error_bars". Default is 1.

smooth_linetype
Type of line to plot for smoothed trendline. Default is "solid".

smooth_linewidth
Line width for smoothed trendline. Default is 1.

smooth_cilinewidth
Line width for smoothed trendline confidence intervals. Default is 1.

gridoff If TRUE, hides gridlines.

x_label Label for the x-axis.

y_label Label for the y-axis.

x_expand (Optional) Expansion factor to expand the x-axis beyond the data. Left and right values are required in the form of c(0.1, 0.2). Default is c(0.05,0.05).

y_expand (Optional) Expansion factor to expand the y-axis beyond the data. Lower and upper values are required in the form of c(0.1, 0.2). Default is c(0.05,0.05).

x_breaks Integer giving desired number of breaks for x axis. (May not return exactly the number requested.)

y_breaks Integer giving desired number of breaks for y axis. (May not return exactly the number requested.)

title_wrap_length
Maximum title length before wrapping to a new line.

Value

A ggplot object representing species range or occurrence time series plot(s). Can be customized using ggplot2 functions.

Examples

```
spec_occ_ts_mammals_denmark <- spec_occ_ts(example_cube_1,
                                           level = "country",
                                           region = "Denmark")
# default colours:
plot_species_ts(spec_occ_ts_mammals_denmark, c(2440728, 4265185))

# custom colours:
plot_species_ts(spec_occ_ts_mammals_denmark, c(2440728, 4265185),
        linecolour = "thistle",
        trendlinecolour = "forestgreen",
        envelopecolour = "lightgreen")
```

plot_ts *Plot Biodiversity Indicator Trend*

Description

Creates a time series plot of a calculated biodiversity indicator, with an optional smoothed trendline, and visualizes uncertainty.

Usage

```
plot_ts(
  x,
  min_year = NULL,
  max_year = NULL,
  title = "auto",
  auto_title = NULL,
  y_label_default = NULL,
  suppress_y = FALSE,
  smoothed_trend = TRUE,
  linecolour = NULL,
  linealpha = 0.8,
  ribboncolour = NULL,
  ribbonalpha = 0.2,
  error_alpha = 1,
  trendlinecolour = NULL,
  trendlinealpha = 0.5,
  envelopecolour = NULL,
  envelopealpha = 0.2,
  smooth_cialpha = 1,
  point_line = c("point", "line"),
  pointsize = 2,
  linewidth = 1,
  ci_type = c("error_bars", "ribbon"),
  error_width = 1,
  error_thickness = 1,
 smooth_linetype = c("solid", "dashed", "dotted", "dotdash", "longdash", "twodash"),
  smooth_linewidth = 1,
  smooth_cilinewidth = 1,
  gridoff = FALSE,
  x_label = NULL,
  y_label = NULL,
  x_expand = NULL,
  y_expand = NULL,
  x_breaks = 10,
  y_breaks = 6,
  wrap_length = 60
)
```

Arguments

x	An 'indicator_ts' object containing a time series of indicator values.
min_year	(Optional) Earliest year to include in the plot.
max_year	(Optional) Latest year to include in the plot.
title	Plot title. Replace "auto" with your own title if you want a custom title or if calling the function manually.
auto_title	Text for automatic title generation, provided by an appropriate S3 method (if calling the function manually, leave as NULL).
y_label_default	Default label for the y-axis, provided by an appropriate S3 method (if calling the function manually, leave as NULL).
suppress_y	If TRUE, suppresses y-axis labels.
smoothed_trend	If TRUE, plot a smoothed trendline over time (`stats::loess()`).
linecolour	(Optional) Colour for the indicator line or points. Default is darkorange.
linealpha	Transparency for indicator line or points. Default is 0.8.
ribboncolour	(Optional) Colour for the bootstrapped confidence intervals. Default is goldenrod1. Set to "NA" if you don't want to plot the CIs.
ribbonalpha	Transparency for indicator confidence interval ribbon (if ci_type = "ribbon"). Default is 0.2.
error_alpha	Transparency for indicator error bars (if ci_type = "error_bar"). Default is 1.
trendlinecolour	(Optional) Colour for the smoothed trendline. Default is blue.
trendlinealpha	Transparency for the smoothed trendline. Default is 0.5.
envelopecolour	(Optional) Colour for the uncertainty envelope. Default is lightsteelblue.
envelopealpha	Transparency for the smoothed trendline envelope. Default is 0.2.
smooth_cialpha	Transparency for the smoothed lines forming the edges of the trendline envelope. Default is 1.
point_line	Whether to plot the indicator as a line or a series of points. Options are "line" or "point". Default is "point".
pointsize	Size of the points if point_line = "point". Default is 2.
linewidth	Width of the line if point_line = "line". Default is 1.
ci_type	Whether to plot bootstrapped confidence intervals as a "ribbon" or "error_bars". Default is "error_bars".
error_width	Width of error bars if ci_type = "error_bars". Default is 1. Note that unlike the default 'width' parameter in geom_errorbar, 'error_width' is NOT dependent on the number of data points in the plot. It is automatically scaled to account for this. Therefore the width you select will be consistent relative to the plot width even if you change 'min_year' and 'max_year'.
error_thickness	Thickness of error bars if ci_type = "error_bars". Default is 1.

smooth_linetype
: Type of line to plot for smoothed trendline. Default is "solid".

smooth_linewidth
: Line width for smoothed trendline. Default is 1.

smooth_cilinewidth
: Line width for smoothed trendline confidence intervals. Default is 1.

gridoff
: If TRUE, hides gridlines.

x_label
: Label for the x-axis.

y_label
: Label for the y-axis.

x_expand
: (Optional) Expansion factor to expand the x-axis beyond the data. Left and right values are required in the form of c(0.1, 0.2). Default is c(0,0).

y_expand
: (Optional) Expansion factor to expand the y-axis beyond the data. Lower and upper values are required in the form of c(0.1, 0.2). Default is c(0,0).

x_breaks
: Integer giving desired number of breaks for x axis. (May not return exactly the number requested.)

y_breaks
: Integer giving desired number of breaks for y axis. (May not return exactly the number requested.)

wrap_length
: Maximum title length before wrapping to a new line.

Value

A ggplot object representing the biodiversity indicator time series plot. Can be customized using ggplot2 functions.

Examples

```
# default colours:
plot_ts(example_indicator_ts1,
        y_label = "Species Richness",
        title = "Observed Species Richness: Mammals in Denmark")

# custom colours:
plot_ts(example_indicator_ts1,
        y_label = "Species Richness",
        title = "Observed Species Richness: Mammals in Denmark",
        linecolour = "thistle",
        trendlinecolour = "forestgreen",
        envelopecolour = "lightgreen")
```

---

print.available_indicators

Print Available Indicators

Description

Provide a summary of indicators registered in the package.

Usage

```
## S3 method for class 'available_indicators'
print(x, n = 30, ...)
```

Arguments

x	Object of class available_indicators
n	Integer specifying the number of rows of data to display.
...	Additional arguments.

---

print.indicator_map *Print an Indicator Map Object*

Description

Provides a summary representation of an indicator_map object, designed for user-friendly display in the console.

Usage

```
## S3 method for class 'indicator_map'
print(x, n = 10, ...)
```

Arguments

x	An indicator_map object.
n	Integer specifying the number of rows of data to display.
...	Additional arguments.

---

print.indicator_ts *Print an Indicator Time Series Object*

Description

Provides a summary representation of an indicator_ts object, designed for user-friendly display in the console.

Usage

```
## S3 method for class 'indicator_ts'
print(x, n = 10, ...)
```

Arguments

x	An indicator_ts object.
n	Integer specifying the number of rows of data to display.
...	Additional arguments.

Examples

```
print(example_indicator_ts1)
```

---

print.processed_cube *Print a Processed Data Cube Object*

Description

Provides a summary representation of a processed_cube object, designed for user-friendly display in the console.

Usage

```
## S3 method for class 'processed_cube'
print(x, n = 10, ...)
```

Arguments

x	A processed_cube object.
n	Integer specifying the number of rows of cube data to display.
...	Additional arguments.

Examples

```
print(example_cube_1)
```

print.processed_cube_dsinfo *Print a Processed Data Cube Object*

Description

Provides a summary representation of a processed_cube_dsinfo object, designed for user-friendly display in the console.

Usage

```
## S3 method for class 'processed_cube_dsinfo'
print(x, n = 10, ...)
```

Arguments

x	A processed_cube_dsinfo object.
n	Integer specifying the number of rows of data to display.
...	Additional arguments.

print.sim_cube *Print a Simulated Data Cube Object*

Description

Provides a summary representation of a sim_cube object, designed for user-friendly display in the console.

Usage

```
## S3 method for class 'sim_cube'
print(x, n = 10, ...)
```

Arguments

x	A sim_cube object.
n	Integer specifying the number of rows of cube data to display.
...	Additional arguments.

process_cube *Process GBIF Data Cubes*

Description

Processes a GBIF data cube and (if applicable) an associated taxonomic information file. If your cube includes a taxonomic info file it is likely a previous generation cube and should be processed using 'process_cube_old'. The taxonomic info file must reside in the same directory as your cube and share a base file name (e.g., 'cubes/my_mammals_cube.csv', 'cubes/my_mammals_info.csv'). If your cube does NOT include a taxonomic info file then it is likely a current generation cube and should be processed using the standard process_cube function The API used to generate the current generation cubes is very flexible and allows user-specified column names. Therefore, please check that the column names of your cube match the Darwin Core standard expected by the process_cube function. If they do not, you may need to enter them manually. The function will return an error if it cannot find all required columns.

Usage

```
process_cube(
  cube_name,
  grid_type = c("automatic", "eea", "mgrs", "eqdgc", "custom", "none"),
  first_year = NULL,
  last_year = NULL,
  force_gridcode = FALSE,
  cols_year = NULL,
  cols_yearMonth = NULL,
  cols_cellCode = NULL,
  cols_occurrences = NULL,
  cols_scientificName = NULL,
  cols_minCoordinateUncertaintyInMeters = NULL,
  cols_minTemporalUncertainty = NULL,
  cols_kingdom = NULL,
  cols_family = NULL,
  cols_species = NULL,
  cols_kingdomKey = NULL,
  cols_familyKey = NULL,
  cols_speciesKey = NULL,
  cols_familyCount = NULL,
  cols_sex = NULL,
  cols_lifeStage = NULL
)

process_cube_old(
  cube_name,
  tax_info = NULL,
  datasets_info = NULL,
  first_year = 1600,
```

```
  last_year = NULL
)
```

Arguments

cube_name	The location and name of a data cube file (e.g., 'inst/extdata/europe_species_cube.csv').
grid_type	Specify which grid reference system your cube uses. By default the function will attempt to determine this automatically and return an error if it fails. If you want to perform analysis on a cube with custom grid codes (e.g. output from the gcube package) or a cube without grid codes, select 'custom' or 'none', respectively.
first_year	(Optional) The first year of occurrences to include. If not specified, uses a default of 1600 to prevent false records (e.g. with year = 0).
last_year	(Optional) The final year of occurrences to include. If not specified, uses the latest year present in the cube.
force_gridcode	Force the function to assume a specific grid reference system. This may cause unexpected downstream issues, so it is not recommended. If you are getting errors related to grid cell codes, check to make sure they are valid.
cols_year	The name of the column containing the year of occurrence (if something other than 'year'). This column is required unless you have a yearMonth column.
cols_yearMonth	The name of the column containing the year and month of occurrence (if present and if other than 'yearMonth'). Use this if only if you do not have a year column. The b3gbi package does not use month data, so the function will convert your yearMonth column to a year column.
cols_cellCode	The name of the column containing the grid reference codes (if other than 'cellCode'). This column is required.
cols_occurrences	The name of the column containing the number of occurrence (if other than 'occurrences'). This column is required.
cols_scientificName	The name of the column containing the scientific name of the species (if other than 'scientificName'). Note that it is not necessary to have both a species column and a scientificName column. One or the other is sufficient.
cols_minCoordinateUncertaintyInMeters	The name of the column containing the minimum coordinate uncertainty of the occurrences (if other than 'minCoordinateUncertaintyinMeters').
cols_minTemporalUncertainty	The name of the column containing the minimum temporal uncertainty of the occurrences (if other than 'minTemporalUncertainty').
cols_kingdom	The name of the column containing the kingdom the occurring species belongs to (if other than 'kingdom'). This column is optional.
cols_family	The name of the column containing the family the occurring species belongs to (if other than 'family'). This column is optional.
cols_species	The name of the column containing the name of the occurring species (if other than 'species'). Note that it is not necessary to have both a species column and a scientificName column. One or the other is sufficient.

cols_kingdomKey
: The name of the column containing the kingdom key of the occurring species (if other than 'kingdomKey'). This column is optinal.

cols_familyKey
: The name of the column containing the family key of the occurring species (if other than 'familykey'). This column is optional.

cols_speciesKey
: The name of the column containing the species key of the occurring species (if other than 'speciesKey'). This column is required, but note that if you have a 'taxonKey' column you can provide it as the speciesKey.

cols_familyCount
: The name of the column containing the occurrence count by family. This column is optional.

cols_sex
: The name of the column containing the sex of the observed individuals. This column is optional.

cols_lifeStage
: the name of the column containing the life stage of the observed individuals. This column is optional.

tax_info
: The location and name of an associated taxonomic info file (e.g., 'inst/extdata/europe_species_info.csv').

datasets_info
: The location and name of an associated dataset info file (e.g., 'inst/extdata/europe_species_datasets.csv').

Value

A tibble containing the processed GBIF occurrence data.

Examples

```
## Not run:
cube_name <- system.file("extdata", "europe_species_cube.csv", package = "b3gbi")
tax_info <- system.file("extdata", "europe_species_info.csv", package = "b3gbi")
europe_example_cube <- process_cube(cube_name, tax_info)
europe_example_cube

## End(Not run)
```

---

spec_occ_map *Calculate Species Occurrences Over Space*

Description

This function calculates the number of occurrences for individual species over a gridded map.

Usage

```
spec_occ_map(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'spec_occ' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
so_map <- spec_occ_map(example_cube_1, level = "country", region = "Denmark")
plot(so_map, c(2440728, 4265185))

## End(Not run)
```

spec_occ_ts *Calculate Number of Occurrences for One or More Species Over Time*

Description

This function calculates number of occurrences for individual species as time series.

Usage

```
spec_occ_ts(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'spec_occ' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
so_ts <- spec_occ_ts(example_cube_1, first_year = 1985)
plot(so_ts, c(2440728, 4265185))
```

`spec_range_map` *Plot Species Ranges Over Space*

Description

This function plots the cells occupied for individual species over a gridded map.

Usage

```
spec_range_map(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow`
	`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	`region` The region of interest (e.g., "Europe"). (Default: "Europe")
	`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	`first_year` Exclude data before this year. (Uses all data in the cube by default.)
	`last_year` Exclude data after this year. (Uses all data in the cube by default.)
	`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'spec_range' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
sr_map <- spec_range_map(example_cube_1, level = "country", region = "Denmark")
plot(sr_map, c(2440728, 4265185))
```

---

spec_range_ts *Calculate Range Size (Number of Cells Occupied) for One or More Species Over Time*

Description

This function calculates number of cells occupied for individual species as time series.

Usage

```
spec_range_ts(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)

`last_year` Exclude data after this year. (Uses all data in the cube by default.)

`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'spec_range' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
sr_ts <- spec_range_ts(example_cube_1, first_year = 1985)
plot(sr_ts, c(2440728, 4265185))

## End(Not run)
```

`tax_distinct_map` *Calculate Taxonomic Distinctness Over Space*

Description

This function calculates the taxonomic distinctness index over a gridded map.

Usage

```
tax_distinct_map(data, rows = 1, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`rows`	Choose which row to select if there are multiple matches when retrieving taxonomic information from GBIF. (Default is 1. Use NA for interactive mode.)
`...`	Arguments passed on to `compute_indicator_workflow`

`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)

`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)

`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')

`region` The region of interest (e.g., "Europe"). (Default: "Europe")

`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")

`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")

`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)

`first_year` Exclude data before this year. (Uses all data in the cube by default.)

`last_year` Exclude data after this year. (Uses all data in the cube by default.)

`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'tax_distinct' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
td_map <- tax_distinct_map(example_cube_1, level = "country", region = "Denmark")
plot(td_map)

## End(Not run)
```

---

tax_distinct_ts *Calculate Taxonomic Distinctness Over Time*

Description

This function calculates the taxonomic distinctness index as a time series.

Usage

```
tax_distinct_ts(data, rows = 1, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
rows	Choose which row to select if there are multiple matches when retrieving taxonomic information from GBIF. (Default is 1. Use NA for interactive mode.)
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'tax_distinct' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
td_ts <- tax_distinct_ts(example_cube_1, level = "country", region = "Denmark")
plot(td_ts)

## End(Not run)
```

`total_occ_map` *Calculate Total Occurrences Over Space*

Description

This function calculates the total number of species occurrence records over a gridded map.

Usage

```
total_occ_map(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow`
	`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	`region` The region of interest (e.g., "Europe"). (Default: "Europe")
	`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	`ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	`output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	`first_year` Exclude data before this year. (Uses all data in the cube by default.)
	`last_year` Exclude data after this year. (Uses all data in the cube by default.)
	`spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'total_occ' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
to_map <- total_occ_map(example_cube_1, level = "country", region = "Denmark")
plot(to_map)

## End(Not run)
```

`total_occ_ts` *Calculate Total Occurrences Over Time*

Description

This function calculates the total number of species occurrence records as a time series.

Usage

```
total_occ_ts(data, ...)
```

Arguments

`data`	A data cube object (class 'processed_cube').
`...`	Arguments passed on to `compute_indicator_workflow` `ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.) `cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world) `level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube') `region` The region of interest (e.g., "Europe"). (Default: "Europe") `ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries") `ne_scale` The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium") `output_crs` The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system) `first_year` Exclude data before this year. (Uses all data in the cube by default.) `last_year` Exclude data after this year. (Uses all data in the cube by default.) `spherical_geometry` If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

`make_valid` Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

`num_bootstrap` Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'total_occ' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
to_ts <- total_occ_ts(example_cube_1, first_year = 1985)
plot(to_ts)

## End(Not run)
```

`williams_evenness_map` *Calculate Williams' Evenness Over Space*

Description

This function calculates Williams' evenness over a gridded map.

Usage

```
williams_evenness_map(data, ...)
```

Arguments

`data` A data cube object (class 'processed_cube').

`...` Arguments passed on to `compute_indicator_workflow`

`ci_type` Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)

`cell_size` Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)

`level` Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')

`region` The region of interest (e.g., "Europe"). (Default: "Europe")

`ne_type` The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")

ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")

output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)

first_year Exclude data before this year. (Uses all data in the cube by default.)

last_year Exclude data after this year. (Uses all data in the cube by default.)

spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)

make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).

num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_map' and 'williams_evenness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
we_map <- williams_evenness_map(example_cube_1, level = "country", region = "Denmark")
plot(we_map)

## End(Not run)
```

---

williams_evenness_ts *Calculate Williams' Evenness Over Time*

Description

This function calculates Williams' evenness over time.

Usage

```
williams_evenness_ts(data, ...)
```

Arguments

data	A data cube object (class 'processed_cube').
...	Arguments passed on to compute_indicator_workflow
	ci_type Type of bootstrap confidence intervals to calculate. (Default: "norm". Select "none" to avoid calculating bootstrap CIs.)
	cell_size Length of grid cell sides, in km. (Default: 10 for country, 100 for continent or world)
	level Spatial level: 'cube', 'continent', 'country', 'world', 'sovereignty', or 'geounit'. (Default: 'cube')
	region The region of interest (e.g., "Europe"). (Default: "Europe")
	ne_type The type of Natural Earth data to download: 'countries', 'map_units', 'sovereignty', or 'tiny_countries'. (Default: "countries")
	ne_scale The scale of Natural Earth data to download: 'small' - 110m, 'medium' - 50m, or 'large' - 10m. (Default: "medium")
	output_crs The CRS you want for your calculated indicator. (Leave blank to let the function choose a default based on grid reference system)
	first_year Exclude data before this year. (Uses all data in the cube by default.)
	last_year Exclude data after this year. (Uses all data in the cube by default.)
	spherical_geometry If set to FALSE, will temporarily disable spherical geometry while the function runs. Should only be used to solve specific issues. (Default is TRUE)
	make_valid Calls st_make_valid() from the sf package. Increases processing time but may help if you are getting polygon errors. (Default is FALSE).
	num_bootstrap Set the number of bootstraps to calculate for generating confidence intervals. (Default: 1000)

Value

An S3 object with the classes 'indicator_ts' and 'williams_evenness' containing the calculated indicator values and metadata.

See Also

compute_indicator_workflow

Examples

```
## Not run:
we_ts <- williams_evenness_ts(example_cube_1, first_year = 1985)
plot(we_ts)

## End(Not run)
```

Index