

Package: biomes (via r-universe)

October 5, 2026

Title Reproducible Occurrence-to-Biome Classification Using 31 Global Biome Schemes

Version 0.9.5

Description Reproducibly classifies occurrence records into biomes using 31 published global biome schemes compiled by Fischer and colleagues (2022) <[doi:10.1111/geb.13574](https://doi.org/10.1111/geb.13574)>, provided as harmonised raster layers at 10x10 km resolution globally. Includes functions to choose the most suitable biome scheme for a dataset by a data-driven ranking, to classify occurrence records, and to tabulate and visualise the result. Works with user-provided occurrences or a taxon name, in which case occurrences are downloaded from GBIF (<<https://www.gbif.org>>) and cleaned automatically.

URL <https://azizka.github.io/biomes/>, <https://github.com/azizka/biomes>

BugReports <https://github.com/azizka/biomes/issues>

Encoding UTF-8

Language en-GB

Roxygen list(markdown = TRUE)

RoxygenNote 8.0.0

Depends R (>= 4.1.0), terra

Imports readr, checkmate, rlang, ggplot2, sf, viridis, tidyterra, utils

VignetteBuilder knitr

Suggests knitr, rmarkdown, testthat (>= 3.0.0), dplyr, tidyr, rgbif, CoordinateCleaner, cowplot, ggforce, rstudioapi

Config/testthat/edition 3

License CC BY 4.0

LazyData true

Config/Needs/website rmarkdown

Config/roxygen2/version 8.0.0

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

Repository <https://azizka.r-universe.dev>

Date/Publication 2026-10-05 12:29:08 UTC

RemoteUrl <https://github.com/azizka/biomes>

RemoteRef HEAD

RemoteSha dfc30fd508667aa8bfcfb69083d73c51924709236

Contents

biomes_classify	2
biomes_download	4
biomes_full	5
biomes_get	7
biomes_info	8
biomes_information	9
biomes_legend	10
biomes_occ	11
biomes_rank	13
biomes_tab	15
biomes_visualise	16
bombacoideae_occurrences	18

Index 20

biomes_classify	<i>Classify occurrences into biomes</i>
-----------------	---

Description

For each occurrence record, assigns a biome label based on its spatial position and one or more biome raster layers. One row of the returned data frame corresponds to one row (one occurrence record) of x.

Usage

```
biomes_classify(  
  x,  
  scheme = NULL,  
  biome = NULL,  
  lon = "decimalLongitude",  
  lat = "decimalLatitude",  
  value = "name",  
  append = TRUE,  
  na = "no_biome",  
  raster_file = NULL  
)
```

Arguments

<code>x</code>	A data frame (with longitude and latitude columns), an <code>sf</code> spatial object, or a <code>terra::SpatVector</code> of point geometries.
<code>scheme</code>	Integer vector in 1:31 selecting one or more biome schemes (biome scheme numbers) from the packaged stack (e.g. <code>scheme = 1</code> or <code>scheme = c(1, 25)</code>). Ignored when <code>biome</code> is supplied. Defaults to <code>NULL</code> , meaning <i>all</i> 31 schemes.
<code>biome</code>	Optional <code>terra::SpatRaster</code> with one or more biome schemes. Use this for custom rasters; for the packaged stack prefer <code>scheme = <int></code> .
<code>lon</code>	Column name of longitude in <code>x</code> (only used if <code>x</code> is a non-spatial data frame). Default: <code>"decimalLongitude"</code> .
<code>lat</code>	Column name of latitude in <code>x</code> (only used if <code>x</code> is a non-spatial data frame). Default: <code>"decimalLatitude"</code> .
<code>value</code>	Character. One of <code>"ID"</code> , <code>"name"</code> , or <code>"both"</code> . Controls whether the returned data frame contains the raw raster value (<code>"ID"</code>), the biome name (<code>"name"</code>), or both (<code>"both"</code>).
<code>append</code>	Logical. If <code>TRUE</code> (default), the classification columns are appended to the input <code>x</code> . If <code>FALSE</code> , only the classification columns are returned (one column per layer for <code>"ID"</code> / <code>"name"</code> , two columns per layer for <code>"both"</code>).
<code>na</code>	Character or <code>NA</code> . Label used for records that fall outside every biome polygon (<code>NA</code> in the raw extraction) in the <code>_name</code> columns. Defaults to <code>"no_biome"</code> , so such records are counted by <code>biomes_tab()</code> under that label. Pass <code>na = NA</code> to keep the missing values as <code>NA</code> .
<code>raster_file</code>	Optional path to a custom biome raster stack file or a <code>SpatRaster</code> . Only used if <code>biome</code> is <code>NULL</code> .

Value

A data frame with one row per record in `x`. By default (`append = TRUE`) the original columns of `x` are kept and the classification columns are added on the right. With `append = FALSE` only the classification columns are returned. Classification columns are named after the input layers, with the suffix `_value` for the raster value and `_name` for the biome name. Raster values without a name in the legend (typically azonal classes encoded with high values) fall back to `"azonal (raster value: X)"`.

Examples

```
# Load example occurrence data
data("bombacoideae_occurrences")

# The biome raster (~36 MB) is downloaded and cached on first use.

# Default: classify against all 31 layers and append the result to x
biomes_classify(bombacoideae_occurrences)

# Single scheme, both raster value and biome name
biomes_classify(bombacoideae_occurrences, scheme = 1, value = "both")
```

```
# Multiple schemes
biomes_classify(bombacoideae_occurrences, scheme = c(1, 25))

# Return only the classification columns (old default behaviour)
biomes_classify(bombacoideae_occurrences, scheme = 1, append = FALSE)
```

biomes_download	<i>Download the packaged biome raster stack</i>
-----------------	---

Description

The 31-layer biome raster stack (`Biomes_Inventory_RasterStack.tif`, ~36 MB) is too large to ship inside the package on CRAN. It is hosted as a release asset on GitHub instead. `biomes_download()` fetches it once and reuses the local copy on every later call, including every internal use by `biomes_get()` or `biomes_classify()`.

Usage

```
biomes_download(path = NULL, overwrite = FALSE, quiet = FALSE)
```

Arguments

path	Optional character string: directory in which to store the raster. Default NULL: use the persistent per-user cache directory if the user has agreed to it, otherwise <code>tempdir()</code> (see Details).
overwrite	Logical flag; if TRUE, re-download the raster even when a local copy already exists. Defaults to FALSE.
quiet	Logical flag; if TRUE, suppress the informational message and the download progress bar. Defaults to FALSE.

Details

The storage location is chosen as follows:

- If path is supplied, the raster is written to that directory and no other location is touched.
- If path is NULL and a copy already exists in the persistent per-user cache directory (see `tools::R_user_dir()`), that copy is reused.
- Otherwise, in interactive sessions, the function asks once for permission to store the raster in the persistent per-user cache directory, so that the download happens only once across R sessions.
- If permission is declined, or in non-interactive sessions, the raster is stored under `tempdir()` and is removed automatically when the R session ends.

The package therefore never writes outside `tempdir()` without the user's explicit consent (an interactive confirmation or an explicit path).

Value

The local file path to the raster, invisibly.

See Also

`biomes_get()` to load the raster as a `terra::SpatRaster`.

Examples

```
# Downloads ~36 MB into the session's temporary directory.
raster_path <- biomes_download(path = tempdir())
raster_path
```

biomes_full	<i>One-call workflow: from taxon (or dataset) to table (and optional figure)</i>
-------------	--

Description

Convenience wrapper that runs the full biomes workflow in a single call. There are two entry paths:

Usage

```
biomes_full(
  x = NULL,
  taxon = NULL,
  scheme = "best",
  lon = "decimalLongitude",
  lat = "decimalLatitude",
  value = "name",
  plot = "none",
  show = FALSE,
  ...
)
```

Arguments

x	Optional. A data frame with longitude/latitude columns, an <code>sf</code> spatial object, or a <code>terra::SpatVector</code> . Mutually exclusive with <code>taxon</code> .
taxon	Optional scientific name (species, genus, family, ...). Mutually exclusive with <code>x</code> .
scheme	One of: an integer in 1:31 (biome scheme number) to force a specific scheme; "best" (default) to pick the best-fitting scheme across all 31 via <code>biomes_rank()</code> ; or a biome definition ("climate", "vegetation", "land_cover", "ecoregion", "integrative", "anthropogenic") to pick the best-fitting scheme within that biome definition.

lon, lat	Column names of longitude / latitude in x (data frame only). Defaults "decimalLongitude"/"decimalLatitude".
value	Passed to <code>biomes_classify()</code> : "name" (default), "ID", or "both".
plot	Which figure(s) <code>biomes_visualise()</code> should build. "none" (default): no figure (the fastest option). "all": the combined, lettered figure (rank + map + barplot) in \$plot. A subset of <code>c("rank", "map", "barplot")</code> : the requested panels are returned individually (no panel letters) in \$rank, \$map and \$barplot – e.g. <code>plot = c("rank", "map", "barplot")</code> fills all three, <code>plot = "map"</code> fills only \$map. NULL is accepted as an alias for "none".
show	Logical. If TRUE, print the figure (if any) and the tabulation to the console as a side effect. The function always returns its result invisibly. Default: FALSE.
...	Further arguments passed to <code>biomes_occ()</code> when taxon is given (e.g. limit, year_min, year_max, use_download, GBIF credentials).

Details

1. **From a taxon name.** Pass a scientific name as taxon (`x = NULL`). `biomes_full()` calls `biomes_occ()` to download cleaned GBIF occurrences for the taxon and then proceeds as below.
2. **From an occurrence dataset.** Pass a data frame, sf object or `terra::SpatVector` as x (`taxon = NULL`).

Once occurrences are available the function:

- picks the biome scheme (either `scheme = <integer>` or, with the default `scheme = "best"`, by running `biomes_rank()` and selecting the top-1 scheme);
- classifies the records with `biomes_classify()`;
- tabulates them with `biomes_tab()`;
- optionally builds a figure with `biomes_visualise()` (controlled by plot; skipped by default for speed).

Value

Invisibly, a `biomes_full` list with elements:

`occ` The occurrence data frame (downloaded or provided).

`scheme` The chosen biome scheme number.

`ranking` The ranking data frame (only when `scheme = "best"`), otherwise NULL.

`classified` The output of `biomes_classify()`.

`table` The biome occurrence table from `biomes_tab()`.

`plot` The combined, lettered figure (only when `plot = "all"`), otherwise NULL.

`rank, map, barplot` The individual panels (no letters), each present only when requested via `plot = c(...)`, otherwise NULL.

Examples

```
## Not run:
# Path 1: from a taxon name. Queries the GBIF web service and may
# prompt for the download workflow (GBIF credentials), so it is not
# run here.
res <- biomes_full(taxon = "Fagus sylvatica", limit = 2000)
res$table

## End(Not run)

# Path 2: from an existing data frame, pick the best scheme.
# Uses the biome raster (~36 MB), downloaded on first use.
data("bombacoideae_occurrences")
res <- biomes_full(x = bombacoideae_occurrences, scheme = "best")

# Path 2 with a fixed scheme
res <- biomes_full(x = bombacoideae_occurrences, scheme = 1)

# Path 2, best-fitting scheme within the vegetation group,
# and build the full figure
res <- biomes_full(x = bombacoideae_occurrences, scheme = "vegetation", plot = "all")
res$plot

# individual panels (no a-c letters) in $rank / $map / $barplot
res <- biomes_full(x = bombacoideae_occurrences, plot = c("map", "barplot"))
res$map
res$barplot
```

biomes_get

Load the packaged biome raster stack

Description

Loads the 31 biome layers shipped with the package as a `terra::SpatRaster` stack.

Usage

```
biomes_get(...)
```

Arguments

... Reserved for future use. Currently no arguments are accepted; passing any will raise an error.

Value

A `terra::SpatRaster` with 31 layers, one per biome classification (in the same order as the rows of [biomes_information](#)).

Examples

```
# Load the default biome raster stack (downloads ~36 MB on first use)
biomes_raster <- biomes_get()
biomes_raster
```

biomes_info	<i>Print metadata for selected biome definitions</i>
-------------	--

Description

Prints a human-readable summary of the biome schemes shipped with the package. For each requested classification the function prints the publication, the criteria and methodology used to define the biomes, a short description, the number of biomes, the biome scheme number, and a list of biome names with their raster values.

Usage

```
biomes_info(x = NULL)
```

Arguments

x	Integer vector of biome scheme numbers between 1 and 31. If NULL (the default), information for all 31 schemes is printed.
---	--

Details

This is the *interactive* sibling of the `biomes_information` data set: use [biomes_information](#) when you want the raw metadata table (e.g. to subset, filter, or join programmatically), and `biomes_info()` when you want a quick read of the most relevant fields for a specific biome scheme.

Value

Invisibly returns the integer vector of biome scheme numbers that was printed. The function is called for its side effect of printing to the console.

See Also

[biomes_information](#) for the underlying metadata table and [biomes_legend](#) for the mapping from raster values to biome names.

Examples

```
# Print information for all biome definitions
biomes_info()

# Print information for the first three biomes
biomes_info(1:3)
```

biomes_information	<i>Metadata for the 31 biome schemes</i>
--------------------	--

Description

A data frame containing descriptive metadata for each of the 31 biome classifications shipped with the package. Each row corresponds to one biome layer in the raster stack returned by `biomes_get()`, in the same order. The metadata is derived from the inventory compiled by Fischer et al. (2022).

Usage

```
biomes_information
```

Format

A data frame with 31 rows and 12 columns:

publication Original publication of the biome scheme.

name_of_classification Full name of the biome scheme.

criteria_for_biome_assignment Criteria used to assign biomes.

methodology Methodology used to derive the biome classification.

scheme_number Biome scheme number (1-31); index of the corresponding layer in the raster stack returned by `biomes_get()`.

background_and_specifications Free-text background information about the classification scheme.

number_of_biomes_zonal_azonal Total number of biomes in the classification, with the split between zonal and azonal biomes in parentheses.

cover_deviation_percent Deviation of the total area covered by this classification from the mean area of all 31 classifications, in percent.

original_file_format File format of the original data source (e.g. raster, shapefile).

source URL or citation of the original data source.

access_date Date on which the original data source was accessed.

biome_definition The concept on which the scheme delimits its biomes, one of "climate", "vegetation", "land_cover", "ecoregion", "integrative" (a synthesis of several criteria or data sources), or "anthropogenic". Used by `biomes_rank()` to rank schemes within the group sharing one biome definition.

Details

This is the raw metadata table. For an interactive, human-readable summary of one or more classifications, see `biomes_info()`.

Source

Fischer J-C, Walentowitz A, Beierkuhnlein C (2022) The biome inventory - Standardizing global biogeographical units. Global Ecology and Biogeography 31(11): 2172-2183. doi:10.1111/geb.13574

biomes_legend	Legend (biome names) for the 31 biome schemes
---------------	---

Description

A data frame mapping the raster values used in each of the 31 biome layers to human-readable biome names. Each row corresponds to one layer in the raster stack returned by `biomes_get()`, in the same order. Columns `id_1`, `id_2`, ... give the biome names for raster values 1, 2, ..., respectively. Cells are NA for classifications with fewer biomes than the maximum across all classifications.

Usage

`biomes_legend`

Format

A data frame with 31 rows and 41 columns:

layer Index of the layer in the raster stack returned by `biomes_get()`.

source Short reference to the publication that defines the classification.

id_1, id_2, id_3, id_4, id_5, id_6, id_7, id_8, id_9, id_10, id_11, id_12, id_13, id_14, id_15, id_16, id_17, id_18, id_19, id_20, id_21, id_22, id_23, id_24, id_25, id_26, id_27, id_28, id_29, id_30, id_31 Biome names for raster values 1 through 39. NA if the classification has fewer biomes.

Source

Fischer J-C, Walentowitz A, Beierkuhnlein C (2022) The biome inventory - Standardizing global biogeographical units. Global Ecology and Biogeography 31(11): 2172-2183. doi:10.1111/geb.13574

biomes_occ

*Download and clean GBIF occurrences for a taxon***Description**

Retrieves occurrence records for a given taxon (species, genus, family, ...) from GBIF and, optionally, runs standard coordinate cleaning with [CoordinateCleaner::clean_coordinates\(\)](#).

Usage

```
biomes_occ(
  taxon,
  use_download = FALSE,
  username = NULL,
  pwd = NULL,
  email = NULL,
  save_dir = NULL,
  filter_clean = TRUE,
  filter_sea = FALSE,
  year_min = NULL,
  year_max = NULL,
  country = NULL,
  limit = NULL,
  slim = TRUE
)
```

Arguments

taxon	Scientific name(s) to query (species, genus, family, ...). Accepts a single name or a character vector of names. All matching keys are bundled into one <code>occ_download()</code> job; the <code>occ_search()</code> path loops over the taxa and splits the user-requested number of records evenly across them.
use_download	Logical. Force the GBIF download workflow even if the total record count is below 100,000. Default: FALSE.
username	GBIF username (used when the download workflow is triggered, either via <code>use_download = TRUE</code> or by the interactive prompt). If NULL, the user is asked at the console.
pwd	GBIF password (same logic as username).
email	GBIF account email (same logic as username).
save_dir	Directory used for outputs when <code>occ_download()</code> is triggered (both the data CSV and a <code>*_citation.txt</code>). If NULL (default), the user is asked at the console; an empty answer and non-interactive sessions fall back to the session's temporary directory (tempdir()).
filter_clean	Logical. If TRUE, run basic spatial cleaning on coordinates with CoordinateCleaner::clean_coordinates() . Default: TRUE.

filter_sea	Logical. If TRUE, also flag occurrences in the sea ("seas" test). Default: FALSE.
year_min	Optional integer. If supplied, only records with year >= year_min are kept on the GBIF side. NULL (default) means no year filter.
year_max	Optional integer. Same as year_min but for the upper bound.
country	Optional character vector. One or more ISO 3166-1 alpha-2 country codes (only used in the download workflow).
limit	Optional integer. Number of records to download. When NULL (default), the user is asked interactively in the occ_search() path; the occ_download() path always returns all available records.
slim	Logical. If TRUE (default), the result is trimmed to family, genus, species, year, countryCode, decimalLongitude, decimalLatitude. Set to FALSE to keep all GBIF columns.

Details

By default, biomes_occ() first asks GBIF how many records exist for the taxon and then prompts the user (in interactive sessions):

1. Use rgbif::occ_search()? (no login required, capped at 100,000 records.) If yes, the user is then asked for the number of records. If no, the function switches to rgbif::occ_download(), which needs a save directory and GBIF credentials and downloads everything (you get a DOI for citation).

The only GBIF predicate applied is hasCoordinate = TRUE. The result is slim by default: family, genus, species, year, countryCode, decimalLongitude, decimalLatitude. Set slim = FALSE to keep every GBIF column. With occ_download() the downloaded data and the GBIF citation are written to save_dir.

Value

A data frame of GBIF occurrence records, optionally cleaned. Always contains decimalLongitude and decimalLatitude columns (when records are returned), so the result can be passed directly to biomes_classify(), biomes_rank() or biomes_full().

Examples

```
## Not run:
# interactive: prompted for occ_search vs occ_download
occ <- biomes_occ(taxon = "Solemyida")

# force the GBIF download workflow up front (requires credentials)
occ <- biomes_occ(
  taxon      = "Fagus sylvatica",
  use_download = TRUE,
  username   = "xxx",
  pwd        = "xxx",
  email      = "you@example.org",
  save_dir   = file.path(tempdir(), "GBIF")
)

## End(Not run)
```

biomes_rank

Rank biome schemes for a given occurrence dataset

Description

Compares the biome schemes for a user-supplied set of occurrence records and proposes a single "best" scheme for that dataset. Each scheme is scored on several data-driven criteria that are combined into one composite_score, which drives the ranking.

Usage

```
biomes_rank(
  x,
  scheme = NULL,
  biome = NULL,
  lon = "decimalLongitude",
  lat = "decimalLatitude",
  definition = "all",
  criteria = c("coverage", "effective_biomes", "granularity"),
  tiebreaker = c("year", "biomes", "none"),
  verbose = TRUE
)
```

Arguments

x	A data frame with longitude / latitude columns, an sf spatial object, or a terra::SpatVector of point geometries.
scheme	Optional integer vector in 1:31 (biome scheme numbers) to restrict the ranking to a subset of the packaged schemes (e.g. scheme = c(1, 5, 25)). NULL (default) ranks all 31 schemes. Ignored when biome is supplied.
biome	Optional terra::SpatRaster stack of biome schemes. Use this for custom rasters; for the packaged stack prefer scheme = <int> instead.
lon	Column name of longitude in x (only used if x is a non-spatial data frame). Default "decimalLongitude".
lat	Column name of latitude in x (only used if x is a non-spatial data frame). Default "decimalLatitude".
definition	Character. Restrict the ranking to the schemes that share one biome definition: one of "all" (default; rank all 31 schemes), "climate", "vegetation", "land_cover", "ecoregion", "integrative", or "anthropogenic". The grouping is taken from the biome_definition column of biomes_information . When a specific definition is chosen, only the schemes of that definition are classified, scored and returned, so the scaled scores and the best scheme are determined within that group. Ignored when biome is supplied.

criteria	Character vector with one or more of "coverage", "effective_biomes", "granularity". Default: all three.
tiebreaker	How tied composite_scores are resolved: "year" (default, more recent publication ranks higher), "biomes" (more biomes ranks higher), or "none" (do not break ties; tied schemes share a rank, dense ranking). With "year" and "biomes" the other key serves as a further fallback, alphabetical scheme_name resolves any remaining ties, and ranks are strict 1..N. With "none" multiple schemes may carry is_best = TRUE.
verbose	Logical. Print progress messages? Default TRUE.

Details

Three equally weighted criteria are used:

1. **coverage**: fraction of records that the scheme places in a biome at all (the rest fall on unclassified, NA cells).
2. **effective_biomes**: $\exp(H')$ (Hill number of order 1), i.e. the effective number of biomes the records spread across, weighted by evenness.
3. **granularity**: biomes actually used, divided by the biomes available in the scheme.

Value

A data frame of classes `biomes_rank` and `data.frame`, with one row per compared biome scheme. Columns: `scheme` (the biome scheme number, 1-31), `scheme_name`, `year` (publication year of the scheme), `n_total`, `n_hit` and `n_na` (number of records in total, classified, and unclassified), `pct_na` (percentage of unclassified records), then one `*_raw` and one `*_scaled` column per requested criterion (the raw score and its rescaled version), `composite_score` (mean of the scaled criteria, drives the ranking), `rank` (1 = best), and `is_best` (TRUE for the top-ranked scheme). The result carries the attributes `criteria` (requested), `criteria_used` (those that entered the composite), `tiebreaker`, `definition`, and `best_scheme` (the biome scheme number of the top-ranked scheme, ready to be used as the `scheme` argument of `biomes_classify()` or `biomes_full()`).

Scaling and the composite score

Every criterion is min-max rescaled to $[0, 1]$ across the compared schemes: the scheme with the lowest value gets 0, the one with the highest gets 1. The `composite_score` is the equal-weight mean of the rescaled criteria. Because the rescaling is relative to the compared set, composite scores are comparable only among schemes that were ranked together, and a rescaled 0 means "lowest among the compared schemes", not zero. Two edge cases: a criterion that does not vary among the compared schemes carries no information and is left out of the composite (its `*_scaled` column is NA; see the attribute `criteria_used`), and a single compared scheme gets no composite score (NA) but is still returned as `best_scheme`.

Schemes are ordered by `composite_score` and ties resolved according to `tiebreaker`.

Note

`biomes_rank()` gives a data-driven ranking, not an authoritative "best" classification. The criteria favour schemes that cover your records and split them into many, evenly-used biomes, but the top-ranked scheme is not necessarily the most suitable one for your question. For best results, narrow

the comparison to one biome definition via definition, and treat the ranking as a shortlist rather than a verdict: inspect the per-criterion columns in the result and use `biomes_info()` to choose the scheme whose concept and resolution actually match your data.

Examples

```
data("bombacoideae_occurrences")

# Ranks the schemes of the biome raster (~36 MB), downloaded on first use.

# Default call: coverage + effective_biomes + granularity, equally weighted
r <- biomes_rank(bombacoideae_occurrences, verbose = FALSE)
head(r)
attr(r, "best_scheme")

# Restrict to a subset of criteria
r2 <- biomes_rank(
  bombacoideae_occurrences,
  criteria = c("coverage", "effective_biomes"),
  verbose = FALSE
)
```

biomes_tab	<i>Tabulate the number of occurrences per biome</i>
------------	---

Description

Summarizes the number of **occurrence records** (one row of x = one occurrence) in each biome, for one or more biome schemes. The output is a long-format table with one row per (scheme, biome) pair.

Usage

```
biomes_tab(x, value = "names")
```

Arguments

<code>x</code>	A data frame returned by <code>biomes_classify()</code> .
<code>value</code>	Character. "names" (default) tabulates the <code>_name</code> columns from <code>biomes_classify()</code> ; "ID" tabulates the <code>_value</code> columns.

Details

This function counts occurrences, not species. To count unique species per biome, deduplicate by species before tabulating (e.g. `dplyr::distinct(species, biome)` after combining classifications with the original data).

Value

A data frame with columns `scheme`, `biome`, and `n` (the number of occurrence records in that biome on that scheme).

Examples

```
# Load example occurrence data
data("bombacoideae_occurrences")

# biomes_classify() downloads and caches the biome raster (~36 MB).

# Tabulate by biome name
classified_names <- biomes_classify(
  x      = bombacoideae_occurrences,
  value = "name"
)
biomes_tab(classified_names, value = "names")

# Tabulate by raster value
classified_ids <- biomes_classify(
  x      = bombacoideae_occurrences,
  value = "ID"
)
biomes_tab(classified_ids, value = "ID")
```

biomes_visualise

Visualise the biomes workflow (ranking, map and biome composition)

Description

Produces the publication figure of the *biomes* workflow for a set of occurrence records. Up to three panels are drawn and combined:

Usage

```
biomes_visualise(
  x,
  scheme = NULL,
  definition = "all",
  biome = NULL,
  lon = "decimalLongitude",
  lat = "decimalLatitude",
  panels = c("rank", "map", "barplot"),
  top_n = NULL,
  titles = TRUE,
```

```

    legend_counts = FALSE,
    legend = TRUE,
    point_color = "#B20000",
    point_size = 0.25,
    combine = TRUE,
    verbose = FALSE
  )

```

Arguments

x	A data frame with longitude/latitude columns, an sf spatial object, or a terra::SpatVector of point geometries.
scheme	Integer in 1:31 (biome scheme number). If NULL (default), the best-fitting scheme is chosen by <code>biomes_rank()</code> (within definition).
definition	Character. Biome definition to rank within when scheme is NULL; passed to <code>biomes_rank()</code> . Default "all".
biome	Optional single-layer terra::SpatRaster. If supplied it is mapped directly and only the map panel is available (no ranking).
lon, lat	Column names of longitude / latitude in x (data frame only). Defaults "decimalLongitude"/"decimalLatitude".
panels	Character vector, any subset of c("rank", "map", "barplot") (default all three). Panels are drawn and lettered in this order.
top_n	Integer or NULL (default). If given, only the top_n top-ranked schemes are shown in the rank panel; NULL shows all compared schemes.
titles	Logical. If TRUE (default), each panel carries a left-aligned title: "Ranked biome schemes" (or "Top n ranked biome schemes" with top_n), "Spatial projection for <reference>" and "Occurrence and species number for <reference>", where the reference is the source of the chosen scheme, e.g. "Ramankutty & Foley (1999)". With FALSE no titles are drawn and the panel letters are placed in the top-left corners instead.
legend_counts	Logical. If TRUE, append the number of records per biome to the map legend labels. Default FALSE (biome names only).
legend	Logical. If TRUE (default), draw the biome colour legend on the map panel.
point_color	Colour of the occurrence points. Default "#B20000".
point_size	Numeric size of the occurrence points. Default 0.25.
combine	Logical. When more than one panel is drawn: TRUE (default) combines them into one lettered figure (a, b, c); FALSE returns a named list of the individual panels (no letters). Ignored for a single panel (always returned as a bare plot).
verbose	Logical. Passed to <code>biomes_rank()</code> . Default FALSE.

Details

- **rank**: the data-driven ranking of the biome schemes (`biomes_rank()`): the composite score per scheme next to the criterion values it averages (coverage, effective number of biomes, granularity), all on the common 0 to 1 scale that enters the composite (the min-max rescaled values). Schemes are labelled by their biome scheme number and source, e.g. 25 (Ramankutty

& Foley, 1999); the composite bar of the best scheme and, in each criterion panel, the bar with the best value of that criterion are outlined in red. All compared schemes are shown unless `top_n` cuts the list.

- **map:** the occurrence records (points) mapped over the chosen biome scheme, with the number of records per biome optionally appended to the legend labels.
- **barplot:** the number of occurrence records (left) and species (right) per biome, with the biome names in the centre. Bars use the same colour per biome as the map; off-map records ("no biome") are grey.

Which panels are drawn is controlled by `panels`. When several panels are combined into one figure, the panel letters (a, b, c) are assigned in drawing order and written into the panel titles ("a: ..."), so selecting only rank and barplot labels them (a) and (b).

Value

For a single panel, a ggplot object (map) or a cowplot object (rank, barplot). For several panels: a combined cowplot object when `combine = TRUE` (default), or a named list of the individual panels (rank, map, barplot) when `combine = FALSE`. Print to display or save with `ggplot2::ggsave()`.

Examples

```
data("bombacoideae_occurrences")
# full figure (rank + map + barplot), best scheme chosen automatically
biomes_visualise(bombacoideae_occurrences)

# only the map, for a fixed scheme
biomes_visualise(bombacoideae_occurrences, scheme = 1, panels = "map")

# map + barplot for the best vegetation scheme
biomes_visualise(bombacoideae_occurrences, definition = "vegetation",
                 panels = c("map", "barplot"))
```

bombacoideae_occurrences

Example occurrence dataset: Bombacoideae

Description

Cleaned occurrence records of the plant subfamily Bombacoideae (Malvaceae): the example dataset used in the README, the vignettes and the function examples, and the worked example of the *biomes* publication. The subfamily is a well-studied case of biome conservatism at the savanna-forest interface. The records were compiled and cleaned by Zizka et al. (2020) from GBIF, BIEN, speciesLink, RAINBIO and further sources.

Usage

bombacoideae_occurrences

Format

A data frame with 17,030 rows and 4 columns:

species Scientific species name (185 species).

decimalLongitude Decimal longitude in WGS84.

decimalLatitude Decimal latitude in WGS84.

countryCode ISO 3166-1 alpha-3 country code of the record.

Source

Zizka A, Carvalho-Sobrinho JG, Pennington RT, Queiroz LP, Alcantara S, Baum DA, Bacon CD, Antonelli A (2020) Transitions between biomes are common and directional in Bombacoideae (Malvaceae). *Journal of Biogeography* 47(6): 1310-1321. doi:[10.1111/jbi.13815](https://doi.org/10.1111/jbi.13815)

Examples

```
data("bombacoideae_occurrences")
head(bombacoideae_occurrences)
length(unique(bombacoideae_occurrences$species))
```

Index

* datasets

- biomes_information, 9
- biomes_legend, 10
- bombacoideae_occurrences, 18

- biomes_classify, 2
- biomes_classify(), 4, 6, 12, 14, 15
- biomes_download, 4
- biomes_full, 5
- biomes_full(), 12, 14
- biomes_get, 7
- biomes_get(), 4, 5, 9, 10
- biomes_info, 8
- biomes_info(), 10, 15
- biomes_information, 8, 9, 13
- biomes_legend, 8, 10
- biomes_occ, 11
- biomes_occ(), 6
- biomes_rank, 13
- biomes_rank(), 5, 6, 9, 12, 17
- biomes_tab, 15
- biomes_tab(), 3, 6
- biomes_visualise, 16
- biomes_visualise(), 6
- bombacoideae_occurrences, 18

- CoordinateCleaner::clean_coordinates(), 11

- ggplot2::ggsave(), 18

- tempdir(), 4, 11

- tools::R_user_dir(), 4