Package ‘faunabr’

October 20, 2025

Title Explore Catálogo Taxônomico da Fauna do Brasil Database

Version 1.0.0

Description A collection of functions designed to retrieve, filter and spatialize data from the Catálogo Taxônomico da Fauna do Brasil. For more information about the dataset, please visit <`http://fauna.jbrj.gov.br/fauna/listaBrasil/`>.

Imports XML (>= 3.99.0.14), data.table (>= 1.14.8), httr (>= 1.4.6), terra (>= 1.7.39), stats (>= 4.2.3), utils(>= 4.2.3),

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 2.10)

LazyData true

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

BugReports `https://github.com/wevertonbio/faunabr/issues`

URL `https://wevertonbio.github.io/faunabr/`

NeedsCompilation no

Author Weverton Trindade [aut, cre] (ORCID: <`https://orcid.org/0000-0003-2045-4555`>)

Maintainer Weverton Trindade <`wevertonf1993@gmail.com`>

Repository CRAN

Date/Publication 2025-10-20 19:30:09 UTC

Contents

`check_fauna_names` *Check species names*

Description

check_fauna_names checks if the species names are correct and searches for suggestions if the name is misspelled or not found in the Fauna do Brasil database

Usage

```
check_fauna_names(data, species, max_distance = 0.1,
                  include_subspecies = TRUE)
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function.
`species`	(character) names of the species to be checked.
`max_distance`	(numeric) Maximum distance (as a fraction) allowed for searching suggestions when the name is misspelled. It can be any value between 0 and 1. The higher the value, the more suggestions are returned. For more details, see `agrep`. Default = 0.1.
`include_subspecies`	whether include subspecies when checking names. Default = TRUE.

Value

a data.frame with the following columns:

- input_name: the species names informed in species argument
- Spelling: indicates if the species name is Correct (a perfect match with a species name in the Flora e Funga do Brasil), Probably_incorrect (partial match), or Not_found (no match with any species).
- Suggested name: If Spelling is Correct, it is the same as the input_name. If Spelling is Probably_correct, one or more suggested names are listed, found according to the maximum distance. If Spelling is "Not_found", the value is NA.
- Distance: The integer Levenshtein edit distance. It represents the number of single-character edits (insertions, deletions, or substitutions) required to transform the input_name into the Suggested_name.
- taxonomicStatus: the taxonomic status of the species name ("valid" or "synonym").
- nomenclaturalStatus: the nomenclatural status of the species name. This information is not available for all species.
- validName: If the species name is not valid or incorrect, the valid name of the specie. If the species name is valid and correct, the same as input_name and Suggested_name.
- family: the family of the specie.

References

Brazilian Zoology Group. Catálogo Taxonômico da Fauna do Brasil. Available at: https://ipt.jbrj.gov.br/jbrj/resource?r=catal

Examples

```
data("fauna_data")
spp <- c("Pantera onça", "Mazama bororo", "Mazama jucunda",
         "Araucaria angustifolia")
check_fauna_names(data = fauna_data, species = spp)
```

`country_codes` *Country Codes and Names*

Description

A dataset containing country codes used in the Catálogo Taxonômico da Fauna do Brasil along with their corresponding country names, as defined in `faunabr::world_fauna`.

Usage

```
data(country_codes)
```

Format

A `data.frame` with 244 rows and 2 variables:

map_name Country names as defined in `faunabr::world_fauna`.

country_code Country codes used in the Catálogo Taxonômico da Fauna do Brasil.

`extract_binomial` *Extract the binomial name (Genus + specific epithet) from a Scientific Name*

Description

Extract the binomial name (Genus + specific epithet) from a Scientific Name

Usage

```
extract_binomial(species_names)
```

Arguments

`species_names`	(character) Scientific names to be converted to binomial names

Value

A vector with the binomial names (Genus + specific epithet).

Examples

```
spp <- c("Panthera onca (Linnaeus, 1758)",
"Zonotrichia capensis subtorquata Swainson, 1837",
"Paraganaspis egeria Díaz & Gallardo, 1996",
"Arrenurus tumulosus intercursor")
spp_new <- extract_binomial(species_names = spp)
spp_new
```

`fauna_attributes` *Get available attributes to filter species*

Description

This function displays all the options available to filter species by its characteristics

Usage

```
fauna_attributes(data, attribute)
```

Arguments

`data`	(data.frame) a data.frame imported with the `load_faunabr` function or a data.frame generated with the `select_fauna` function.
`attribute`	(character) the type of characteristic. Accept more than one option. See detail to see the options.

Details

The attribute argument accepts the following options: phylum, class, family, genus, lifeForm, habitat, states, country, origin, and taxonomicstatus. These options represent different characteristics of species that can be used for filtering.

Value

a list of data.frames with the available options to use in the `select_fauna` function.

Examples

```
data("fauna_data") #Load data example
# Get available states, countries and lifeForms to filter species
d <- fauna_attributes(data = fauna_data,
                      attribute = c("country", "lifeform", "states"))
```

`fauna_by_vernacular` *Search for taxa using vernacular names*

Description

Search for taxa using vernacular names

Usage

```
fauna_by_vernacular(data, names, exact = FALSE)
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function or generated with the function `select_fauna`.
`names`	(character) vernacular name ("Nome comum") of the species to be searched
`exact`	(logic) if TRUE, the function will search only for exact matches. For example, if names = "veado-mateiro" and exact = TRUE, the function will return only the species popularly known as "veado-mateiro". On the other hand, if names = "veado-mateiro" and exact = FALSE, the function will return other results as "Veado-mateiro-pequeno". Default = FALSE

Value

a data.frame with the species with vernacular names that match the input names

Examples

```
data("fauna_data") #Load Fauna do Brasil data
#Search for species whose vernacular name is 'veado-mateiro'
veado_exact <- fauna_by_vernacular(data = fauna_data,
                                   names = "veado-mateiro",
                                   exact = TRUE)
veado_exact
#Search for species whose vernacular name is 'veado_mateiro', allowing non-exact
#matches
veado_not_exact <- fauna_by_vernacular(data = fauna_data,
                                       names = "veado-mateiro",
                                      exact = FALSE)
```

`fauna_data` *Catálogo Taxonômico da Fauna do Brasil database - Version 1.17*

Description

A dataset containing a subset of the Catálogo Taxonômico da Fauna do Brasil database (version 1.17)

Usage

```
data(fauna_data)
```

Format

A `data.frame` with 9558 rows and 19 variables:

species Species names

subspecies Subspecies names

scientificName Complete scientific name of the species

validName Valid name of the species (NA when the name in species is already a valid name)

kingdom Kingdom to which species belongs (Animalia)

phylum Phylum to which species belongs

class Class to which species belongs

order Order to which species belongs

family Family to which species belongs

genus Genus to which species belongs

lifeForm Life form of the species (e.g: free_living_individual, colonial, sessile, etc.)

habitat Habitat type of the species (e.g., terrestrial, arboreal, freshwater, etc.)

states Federal states with confirmed occurrences of the species

countryCode Countries with confirmed occurrences of the species

origin Indicates whether the species is native, introduced, domesticated, cryptogenic or invasive

taxonomicStatus Indicates the level of recognition and acceptance of the species (valid or synonym)

nomenclaturalStatus Indicates the legitimacy and validity of the species name (original_combination, changed_combination, etc.)

vernacularName Locally or culturally used name for the species

taxonRank Taxonomic rank (Species, Genus, Family, Order, etc). This data contains only Species

References

Brazilian Zoology Group. Catálogo Taxonômico da Fauna do Brasil. Available at: https://ipt.jbrj.gov.br/jbrj/resource?r=catal

---

`fauna_discrepancies` *Resolve discrepancies between species and subspecies information*

Description

Resolve discrepancies between species and subspecies information

Usage

```
fauna_discrepancies(data)
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function.

Details

In the original dataset, discrepancies may exist between species and subspecies information. An example of a discrepancy is when species occurs only in one state (e.g., SP), but a subspecies or variety of the same species occurs in another states (e.g., SP and RJ). This function rectifies such discrepancies by considering distribution (states and countries) life form, and habitat. For instance, if a subspecies is recorded in a specific state, it implies that the species also occurs in that state

Value

a data.frame with the discrepancies solved

Examples

```
data("fauna_data") #Load fauna e Funga do Brasil data
#Check if discrepancies were solved in the dataset
attr(fauna_data, "solved_discrepancies")
#Solve discrepancies
fauna_solved <- fauna_discrepancies(fauna_data)
#Check if discrepancies were solved in the dataset
attr(fauna_solved, "solved_discrepancies")
```

`fauna_pam` *Get a presence-absence matrix*

Description

Get a presence-absence matrix of species based on its distribution (brazilian states and/or countries) according to Fauna do Brasil.

Usage

```
fauna_pam(data, by_state = TRUE, by_country= FALSE,
                remove_empty_sites = TRUE,
                return_richness_summary = TRUE,
                return_spatial_richness = TRUE,
                return_plot = TRUE)
```

Arguments

`data`	(data.frame) a data.frame imported with the `load_faunabr` function or generated by either `select_fauna` or `subset_fauna` functions
`by_state`	(logical) get occurrences by State. Default = TRUE
`by_country`	(logical) get occurrences by countries. Default = FALSE
`remove_empty_sites`	(logical) remove empty sites (sites without any species) from final presence-absence matrix. Default = TRUE

`return_richness_summary`	(logical) return a data.frame with the number of species in each site. Default = TRUE
`return_spatial_richness`	(logical) return a SpatVector with the number of species in each site. Default = TRUE
`return_plot`	(logical) plot map with the number of species in each site. Only works if return_spatial_richness = TRUE. Default = TRUE

Value

If return_richness_summary and/or return_spatial_richness is set to TRUE, return a list with:

- PAM: the presence-absence matrix (PAM)
- Richness_summary: a data.frame with the number of species in each site
- Spatial_richness: a SpatVector with the number of species in each site (by State and/or country)

If return_richness_summary and return_spatial_richness is set to FALSE, return a presence-absence matrix

Examples

```
#Test function
data("fauna_data") #Load fauna e Funga do Brasil data
#Select native species of mammals with occurrence only in Brazil
br_mammals <- select_fauna(data = fauna_data,
                           include_subspecies = FALSE, phylum = "all",
                           class = "Mammalia",
                           order = "all", family = "all",
                           genus = "all",
                           lifeForm = "all", filter_lifeForm = "in",
                           habitat = "all", filter_habitat = "in",
                           states = "all", filter_states = "in",
                           country = "BR", filter_country = "only",
                           origin = "all", taxonomicStatus = "valid")
#Get presence-absence matrix in states
pam_mammals <- fauna_pam(data = br_mammals, by_state = TRUE,
                         by_country = FALSE,
                         remove_empty_sites = TRUE,
                         return_richness_summary = TRUE,
                         return_spatial_richness = TRUE,
                         return_plot = TRUE)
```

`fauna_spat_occ` *Get Spatial polygons (SpatVectors) of species based on its distribution (states and countrys) according to Fauna do Brasil*

Description

Get Spatial polygons (SpatVectors) of species based on its distribution (states and countrys) according to Fauna do Brasil

Usage

```
fauna_spat_occ(data, species, state = TRUE, country = TRUE,
                        spat_state = NULL, spat_country = NULL, verbose = TRUE)
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function.
`species`	(character) one or more species names (only genus and specific epithet, eg. "Panthera onca")
`state`	(logical) get SpatVector of states with occurrence of the species? Default = TRUE
`country`	(logical) get SpatVector of countrys with occurrence of the species? Default = TRUE
`spat_state`	(SpatVector) a SpatVector of the Brazilian states. By default, it uses the SpatVector provided by geobr::read_state(). It can be another Spatvector, but the structure must be identical to 'faunabr::states', with a column called "abbrev_state" identifying the states codes.
`spat_country`	(SpatVector) a SpatVector of the world countries. By default, it uses the SpatVector provided by rnaturalearth::ne_countries. It can be another Spatvector, but the structure must be identical to 'faunabr::world_fauna', with a column called "country_code" identifying the country codes.
`verbose`	(logical) Whether to display species being filtered during function execution. Set to TRUE to enable display, or FALSE to run silently. Default = TRUE.

Value

A list with SpatVectors of states and/or countrys for each specie.

Examples

```
library(terra)
data("fauna_data")
spp <- c("Panthera onca", "Mazama jucunda")
#Get states, countrys and intersection states-countrys of species
spp_spt <- fauna_spat_occ(data = fauna_data, species = spp, state = TRUE,
                          country = TRUE, verbose = TRUE)
```

```
#Plot states with confirmed occurrence of Panthera onca and Mazama jucunda
plot(spp_spt$`Panthera onca`$states)
plot(spp_spt$`Mazama jucunda`$states)
#Plot countries with confirmed occurrence of Panthera onca and Mazama jucunda
plot(spp_spt$`Panthera onca`$countries)
plot(spp_spt$`Mazama jucunda`$countries)
```

`fauna_synonym` *Retrieve synonyms for species*

Description

Retrieve synonyms for species

Usage

```
fauna_synonym(data, species,
                    include_subspecies = TRUE)
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function
`species`	(character) names of the species
`include_subspecies`	(logical) include subspecies that are synonyms of the species? Default = TRUE

Value

A data.frame containing unique synonyms of the specified species along with relevant information on taxonomic status.

Examples

```
data("fauna_data") #Load Flora e Funga do Brasil data
#Species to extract synonyms
spp <- c("Panthera onca", "Mazama jucunda", "Subulo gouzoubira")
spp_synonyms <- fauna_synonym(data = fauna_data, species = spp,
                              include_subspecies = FALSE)
spp_synonyms
```

`fauna_version` *Check if you have the latest version of Fauna do Brasil data available*

Description

This function checks if you have the latest version of the Fauna do Brasil data available in a specified directory.

Usage

```
fauna_version(data_dir)
```

Arguments

`data_dir`	the directory where the data should be located.

Value

A message informing whether you have the latest version of Fauna do Brasil available in the data_dir

Examples

```
#Check if there is a version of Fauna do Brasil data available in the
#current directory
fauna_version(data_dir = getwd())
```

`filter_faunabr` *Identify records outside natural ranges according to Fauna do Brasil*

Description

This function removes or flags records outside of the species' natural ranges according to information provided by the Fauna do Brasil database

Usage

```
filter_faunabr(data, occ, species = "species", long = "x", lat = "y",
                      by_state = TRUE, buffer_state = 20, by_country = TRUE,
                      buffer_country = 20, value = "flag&clean",
                      keep_columns = TRUE, spat_state = NULL,
                      spat_country = NULL, verbose = TRUE)
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function.
`occ`	(data.frame) a data.frame with the records of the species.
`species`	(character) column name in occ with species names. Default = "species"
`long`	(character) column name in occ with longitude data. Default = "x"
`lat`	(character) column name in occ with latitude data. Default = "y"
`by_state`	(logical) filter records by state? Default = TRUE
`buffer_state`	(numeric) buffer (in km) around the polygons of the states of occurrence of the specie. Default = 20.
`by_country`	(logical) filter records by country? Default = TRUE
`buffer_country`	(numeric) buffer (in km) around the polygons of the countries of occurrence of the specie. Default = 20.
`value`	(character) Defines output values. See Value section. Default = "flag&clean".
`keep_columns`	(logical) if TRUE, keep all the original columns of the input occ. If False, keep only the columns species, long and lat. Default = TRUE
`spat_state`	(SpatVector) a SpatVector of the Brazilian states. By default, it uses the SpatVector provided by geobr::read_state(). It can be another Spatvector, but the structure must be identical to 'faunabr::states', with a column called "abbrev_state" identifying the states codes.
`spat_country`	(SpatVector) a SpatVector of the world countries. By default, it uses the SpatVector provided by rnaturalearth::ne_countries. It can be another Spatvector, but the structure must be identical to 'faunabr::world_fauna', with a column called "country_code" identifying the country codes.
`verbose`	(logical) Whether to display species being filtered during function execution. Set to TRUE to enable display, or FALSE to run silently. Default = TRUE.

Details

If by_state = TRUE and/or by_country = TRUE, the function takes polygons representing the states and/or countrys with confirmed occurrences of the specie, draws a buffer around the polygons, and tests if the records of the species fall inside it.

Value

Depending on the 'value' argument. If value = "flag", it returns the same data.frame provided in data with additional columns indicating if the record falls inside the natural range of the specie (TRUE) or outside (FALSE). If value = "clean", it returns a data.frame with only the records that passes all the tests (TRUE for all the filters). If value = "flag&clean" (Default), it returns a list with two data.frames: one with the flagged records and one with the cleaned records.

Examples

```
data("fauna_data") #Load fauna e Funga do Brasil data
data("occurrences") #Load occurrences
pts <- subset(occurrences, species == "Panthera onca")
```

```
fd <- filter_faunabr(data = fauna_data,
                     occ = pts, long = "x", lat = "y", species = "species",
                     by_state = TRUE, buffer_state = 20,
                     by_country = TRUE, buffer_country = 20,
                     value = "flag&clean", keep_columns = TRUE,
                     verbose = FALSE)
```

`get_faunabr` *Download the latest version of Catálogo Taxonômico da Fauna do Brasil*

Description

This function downloads the latest or an older version of Catálogo Taxonômico da Fauna do Brasil database, merges the information into a single data.frame, and saves this data.frame in the specified directory.

Usage

```
get_faunabr(output_dir, data_version = "latest",
                   solve_discrepancies = TRUE, translate = TRUE,
                   overwrite = TRUE, verbose = TRUE)
```

Arguments

`output_dir`	(character) a directory to save the data downloaded from Fauna do Brasil
`data_version`	(character) Version of the Fauna do Brasil database to download. Use "latest" to get the most recent version, which is updated frequently. Alternatively, specify an older version (e.g., data_version = "1.2").Default value is "latest".
`solve_discrepancies`	Resolve inconsistencies between species and subspecies information. When set to TRUE (default), species information is updated based on unique data from subspecies. For example, if a subspecies occurs in a certain state, it implies that the species also occurs in that state.
`translate`	(logical) whether to translate the original dataset ("lifeForm", "origin", "habitat", and "taxonRank") from Portuguese to English. Default is TRUE.
`overwrite`	(logical) If TRUE, data is overwritten. Default = TRUE.
`verbose`	(logical) Whether to display messages during function execution. Set to TRUE to enable display, or FALSE to run silently. Default = TRUE.

Value

The function downloads the latest version of the Catálogo Taxonômico da Fauna do Brasil database from the official source. It then merges the information into a single data.frame, containing details on species, taxonomy, occurrence, and other relevant data. The merged data.frame is then saved as a file in the specified output directory. The data is saved in a format that allows easy loading using the `load_faunabr` function for further analysis in R.

References

Brazilian Zoology Group. Catálogo Taxonômico da Fauna do Brasil. Available at: https://ipt.jbrj.gov.br/jbrj/resource?r=catal

Examples

```
## Not run:
#Creating a folder in a temporary directory
#Replace 'file.path(tempdir(), "faunaabr")' by a path folder to be create in
#your computer
my_dir <- file.path(file.path(tempdir(), "faunabr"))
dir.create(my_dir)
#Download, merge and save data
get_faunabr(output_dir = my_dir)

## End(Not run)
```

`load_faunabr` *Load Brazilian Fauna database*

Description

Load Brazilian Fauna database

Usage

```
load_faunabr(data_dir, data_version = "latest",
                       type = "short", verbose = TRUE, encoding = "UTF-8")
```

Arguments

`data_dir`	(character) the same directory used to save the data downloaded from Brazilian Fauna using the get_faunabr function.
`data_version`	(character) the version of Brazilian Fauna database to be loaded. It can be "latest", which will load the latest version available; or another specified version, for example "1.2". Default = "latest".
`type`	(character) it determines the number of columns that will be loaded. It can be "short" or "complete". Default = "short". See details.
`verbose`	(logical) Whether to display messages during function execution. Set to TRUE to enable display, or FALSE to run silently. Default = TRUE.
`encoding`	(character) the declared encodings for special characters. Character strings in R can be declared to be encoded in "latin1" or "UTF-8". Default: "UTF-8".

Details

The parameter type accepts two arguments. If type = short, it will load a data.frame with the 20 columns needed to run the other functions of the package: species, subspecies, scientificName, validName, kingdom, phylum, class, order, family, genus, lifeForm, habitat, states, countryCode, origin, taxonomicStatus, nomenclaturalStatus, vernacularName, and taxonRank. If type = complete, it will load a data.frame with all 31 variables available in Brazilian Fauna database.

Value

A data.frame with the specified version (Default is the latest available) of the Brazilian Fauna database. This data.frame is necessary to run most of the functions of the package.

References

Brazilian Zoology Group. Catálogo Taxonômico da Fauna do Brasil. Available at: https://ipt.jbrj.gov.br/jbrj/resource?r=catal

Examples

```
## Not run:
#Creating a folder in a temporary directory
#Replace 'file.path(tempdir(), "faunabr")' by a path folder to be create in
#your computer
my_dir <- file.path(file.path(tempdir(), "faunabr"))
dir.create(my_dir)
#Download, merge and save data
get_fauna(output_dir = my_dir, data_version = "latest", overwrite = TRUE,
            verbose = TRUE)
#Load data
df <- load_faunabr(data_dir = my_dir, data_version = "latest",
                   type = "short")

## End(Not run)
```

`map_translation` *Helpers for translating data*

Description

A list of data.frames used by `faunabr::translate_faunabr()` function. `faunabr::map_translation`.

Usage

```
data(map_translation)
```

Format

A `list` with 5 data.frames ("lifeForm", "origin", "habitat", "taxonRank", and "taxonomicStatus"). Each data.frame has 2 columns:

pt_br The attribute in Brazilian Portuguese.

en The attribute in English.

`occurrences` *Records of animal species*

Description

A dataset containing records of 2 species downloaded from GBIF, with additional fake data. The records were obtained with `plantR::rgbif2`

Usage

```
data(occurrences)
```

Format

A `data.frame` with 2798 rows and 3 variables:

species Species names (Panthera onca and Chaetomys subspinosus)

x Longitude

y Latitude

source record downloaded from GBIF or fake data

References

GBIF, 2024. florabr R package: Records of plant species. https://doi.org/10.15468/DD.QPGEB7

`select_fauna` *Selection of species based on its characteristics and distribution*

Description

select_fauna allows filter species based on its characteristics and distribution available in Brazilian Fauna

Usage

```
select_fauna(data, include_subspecies = FALSE, phylum = "all",
                      class = "all", order = "all", family = "all",
                      genus = "all",
                      lifeForm = "all", filter_lifeForm = "in",
                      habitat = "all", filter_habitat = "in",
                      states = "all", filter_states = "in",
                      country = "all", filter_country = "in",
                      origin = "all", taxonomicStatus = "valid")
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function.
`include_subspecies`	(logical) include subspecies? Default = FALSE
`phylum`	(character) The phyla for filtering the dataset. It can be included more than one phylum. Default = "all".
`class`	(character) The classes for filtering the dataset. It can be included more than one class. Default = "all".
`order`	(character) The orders for filtering the dataset. It can be included more than one order. Default = "all".
`family`	(character) The families for filtering the dataset. It can be included more than one family. Default = "all".
`genus`	(character) The genus for filtering the dataset. It can be included more than one genus. Default = "all".
`lifeForm`	(character) The life forms for filtering the dataset. It can be included more than one lifeForm. Default = "all"
`filter_lifeForm`	(character) The type of filtering for life forms. It can be "in", "only", "not_in" and "and". See details for more about this argument.
`habitat`	(character) The life habitat for filtering the dataset. It can be included more than one habitat. Default = "all"
`filter_habitat`	(character) The type of filtering for habitat. It can be "in", "only", "not_in" and "and". See details for more about this argument.
`states`	(character) The states for filtering the dataset. It can be included more than one state. Default = "all".
`filter_states`	(character) The type of filtering for states. It can be "in", "only", "not_in" and "and". See Details for more about this argument.
`country`	(character) The country or countries with confirmed occurrences for filtering the dataset. It can be included more than one country. Default = "all".
`filter_country`	(character) The type of filtering for country. It can be "in", "only", "not_in" and "and". See details for more about this argument.
`origin`	(character) The origin for filtering the dataset. It can be "native", "introduced", "cryptogenic", "domesticaded" and "invasora". Default = "all".
`taxonomicStatus`	(character) The taxonomic status for filtering the dataset. It can be "valid", "synonym" or "all". Default = "valid".

Details

It's possible to choose 4 ways to filter by lifeform, by habitat, by states and by country: "in": selects species that have any occurrence of the determined values. It allows multiple matches. For example, if country = c("brazil", argentina") and filter_country = "in", it will select all species that occur in Brazil and/or Argentina, some of which may also occur in other countries.

"only": selects species that have only occurrence of the determined values. It allows only single matches. For example, if country = c("brazil", argentina") and filter_country = "in", it will select all species that occur exclusively in both countries, without any occurrences in other countries.

"not_in": selects species that don't have occurrence of the determined values. It allows single and multiple matches. For example, if country = c("brazil", argentina") and filter_country = "not_in", it will select all species without occurrences in Brazil and Argentina.

"and": selects species that have occurrence in all determined values. It allows single and multiple matches. For example, if country = c("brazil", argentina") and filter_country = "and", it will select all species that occurs only in both countries,including species that occurs in other countries too.

To get the complete list of arguments available for phylum, class, order, family, genus, lifeForm, habitat, states, country and origins, use the function `fauna_attributes`

Value

A new dataframe with the filtered species.

References

Brazilian Zoology Group. Catálogo Taxonômico da Fauna do Brasil. Available at: https://ipt.jbrj.gov.br/jbrj/resource?r=catal

Examples

```
data("fauna_data") #Load data example
#Select endemic and native species of birds (Aves) with confirmed occurrence
#in Brazil or Argentina
aves_br_ar <- select_fauna(data = fauna_data, include_subspecies = FALSE,
                           phylum = "all", class = "Aves",
                           order = "all",
                           family = "all",
                           genus = "all",
                           lifeForm = "all", filter_lifeForm = "in",
                           habitat = "all", filter_habitat = "in",
                           states = "all", filter_states = "in",
                           country = c("BR", "AR"),
                           filter_country = "in",
                           origin = "native",
                           taxonomicStatus = "valid")
```

`states` *SpatVector of the federal states of Brazil*

Description

A simplified and packed SpatVector of the polygons of the federal states of Brazil. The spatial data was originally obtained from `geobr::read_state`. Borders have been simplified by removing vertices of borders using `terra::simplifyGeom`. It's necessary unpack the Spatvectos using `terra::unwrap`

@usage data(states) states <- terra::unwrap(states)

Usage

```
states
```

Format

A `SpatVector` with 27 geometries and 3 attributes:

abbrev_state	State acronym
name_state	State's full name
name_region	The region to which the state belongs

`subset_fauna` *Extract a subset of species from Fauna do Brasil database*

Description

Returns a data.frame with a subset of species from Fauna do Brasil database

Usage

```
subset_fauna(data, species, include_subspecies = FALSE)
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function.
`species`	(character) names of the species to be extracted from Fauna do Brasil database.
`include_subspecies`	(logical) include subspecies? Default = FALSE

Value

A data.frame with the selected species.

Examples

```
data("fauna_data") #Load data example
#Species to extract from database
spp <- c("Panthera onca", "Mazama jucunda", "Subulo gouzoubira")
spp_subset <- subset_fauna(data = fauna_data, species = spp,
                           include_subspecies = FALSE)
spp_subset
```

`translate_faunabr` *Translate information in Brazilian Fauna database*

Description

This function translates information in the "lifeForm", "origin", "habitat", "taxonRank", and "taxonomicStatus" columns between Portuguese and English.

Usage

```
translate_faunabr(data, map_list = NULL, to = "en")
```

Arguments

`data`	(data.frame) the data.frame imported with the `load_faunabr` function.
`map_list`	(list) A list of data.frames used for translation. The default is NULL, which means it uses `faunabr::map_translation`. If not NULL, its structure (list names and data.frame column names) must be identical to `faunabr::map_translation`.
`to`	(character) The target language for translation. Available options are "en" to translate from Portuguese to English, and "pt_br" to translate from English to Portuguese. The default is "en".

Value

A data.frame with the values in the "lifeForm", "origin", "habitat", "taxonRank", and "taxonomicStatus" columns translated.

Examples

```
data("fauna_data") #Load data example (in English)
#Translate to Portuguese
fauna_portugues <- translate_faunabr(data = fauna_data, to = "pt_br")
# See attributes of lifeForm in Portuguese
fauna_attributes(fauna_portugues, attribute = "lifeForm")
```

`world_fauna` *SpatVector of the world countries*

Description

A simplified and packed SpatVector of the world country polygons. The spatial data was originally obtained from `rnaturalearth::ne_countries`. Borders have been simplified by removing vertices of borders using `terra::simplifyGeom`.It's necessary unpack the Spatvectos using `terra::unwrap`

@usage data(world_fauna) biomes <- terra::unwrap(world_fauna)

Usage

```
world_fauna
```

Format

A `SpatVector` with 258 geometries and 1 attribute:

name The name of the country (argentina, brazil, colombia, etc.)

Index