

Package ‘florabr’

March 14, 2024

Title Explore Brazilian Flora 2020 Database

Version 1.1.0

Description A collection of functions designed to retrieve, filter and spatialize data from the Brazilian Flora 2020 dataset. For more information about the dataset, please visit <https://floradobrasil.jbrj.gov.br/consulta/>.

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), grDevices (>= 4.2.3)

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.3

Depends R (>= 2.10)

LazyData true

Suggests knitr, rmarkdown

VignetteBuilder knitr

URL <https://wevertonbio.github.io/florabr/>

BugReports <https://github.com/wevertonbio/florabr/issues>

NeedsCompilation no

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

Maintainer Weverton Trindade <wevertonf1993@gmail.com>

Repository CRAN

Date/Publication 2024-03-14 01:10:10 UTC

R topics documented:

Attributes	2
bf_data	2
biomes	3
brazil	4

check_names	4
check_version	5
filter_florabr	6
get_attributes	8
get_binomial	9
get_florabr	10
get_pam	11
get_spat_occ	12
load_florabr	14
occurrences	15
select_by_vernacular	16
select_species	17
states	20
subset_species	20

Index 22

Attributes	<i>Available attributes/parameters to filter and select species</i>
------------	---

Description

A list of dataset containing the available attributes to filter and select species. The dataset is used internally by `get_attributes`

Usage

```
data(Attributes)
```

Format

A list with 11 elements: States, Biome, vegetationType, lifeForm, habitat, nomenclaturalStatus, taxonomicStatus, Endemism, Origin, Group, and Subgroup

bf_data	<i>Brazilian Flora 2020 database - Version 393.387</i>
---------	--

Description

A dataset containing a subset of the Brazilian Flora database (version 393.387)

```
@usage data(bf_data)
```

Usage

```
bf_data
```

Format

A data.frame with 110250 rows and 18 variables:

species Species names

acceptedName Accepted name of the species (NA when the name in species is already an accepted name)

kingdom Kingdom to which species belongs (Plantae or Fungi)

Group Major group to which species belongs (Angiosperms, Gymnosperms, Ferns and Lycophytes, Bryophytes, and Algae)

Subgroup Subgroup to which species belongs. Only available for Bryophytes (Mosses, Hornworts, and Liverworts)

family Family to which species belongs

genus Genus to which species belongs

lifeForm Life form of the species (e.g: Tree, Herb, Shrub, etc.)

habitat Habitat type of the species (e.g., Terrestrial, Rupicolous, Epiphytic, etc.)

Biome Biomes with confirmed occurrences of the species

States Federal states with confirmed occurrences of the species

vegetationType Vegetation types with confirmed occurrences of the species

Origin Indicates whether the species is Native, Naturalized, or Cultivated in Brazil

Endemism Indicates whether the species is Endemic or Non-endemic to Brazil

taxonomicStatus Indicates the level of recognition and acceptance of the species (Accepted or Synonym)

nomenclaturalStatus Indicates the legitimacy and validity of the species name (Correct, Illegitimate, Uncertain_Application, etc.)

vernacularName Locally or culturally used name for the species

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

biomes

SpatVector of the biomes of Brazil

Description

A simplified and packed `SpatVector` of the polygons of the biomes present in Brazilian territory. The spatial data was originally obtained from `geobr::read_biomes`. Borders have been simplified by removing vertices of borders using `terra::simplifyGeom`. It's necessary unpack the `Spatvectos` using `terra::unwrap`

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

Usage

```
biomes
```

Format

A `SpatVector` with 6 geometries and 1 attribute:

name_biome The name of the biome (Amazon, Caatinga, Cerrado, Atlantic_Forest, Pampa, and Pantanal)

brazil	<i>SpatVector of the Brazil's national borders</i>
--------	--

Description

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

```
@usage data(brazil) brazil <- terra::unwrap(brazil)
```

Usage

```
brazil
```

Format

A `SpatVector` with 1 geometry and 0 attribute

check_names	<i>Check species names</i>
-------------	----------------------------

Description

`check_names` checks if the species names are correct and searches for suggestions if the name is misspelled or not found in the Brazilian Flora 2020 database.

Usage

```
check_names(data, species, max_distance = 0.1, Kingdom = "Plantae")
```

Arguments

<code>data</code>	(<code>data.frame</code>) the <code>data.frame</code> imported with the <code>load_florabr</code> function.
<code>species</code>	(<code>character</code>) names of the species to be checked.
<code>max_distance</code>	(<code>numeric</code>) 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.
<code>Kingdom</code>	(<code>character</code>) the kingdom to which the species belong. It can be "Plantae" or "Fungi". Default = "Plantae".

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 Brazilian Flora 2020), 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 ("Accepted" or "Synonym").
- `nomenclaturalStatus`: the nomenclatural status of the species name. This information is not available for all species.
- `acceptedName`: If the species name is not accepted or incorrect, the accepted name of the specie. If the species name is accepted and correct, the same as `input_name` and `Suggested_name`.
- `family`: the family of the specie.

References

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Examples

```
data("bf_data", package = "florabr")
spp <- c("Butia cattarinensis", "Araucaria angustifolia")
check_names(data = bf_data, species = spp)
```

check_version

Check if you have the latest version of Brazilian Flora data available

Description

This function checks if you have the latest version of the Brazilian Flora data available in a specified directory.

Usage

```
check_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 Brazilian Flora data available in the data_dir

Examples

```
#Check if there is a version of Brazilian Flora data available in the current
#directory
check_version(data_dir = getwd())
```

filter_florabr	<i>Identify records outside natural ranges according to Brazilian Flora 2020</i>
----------------	--

Description

This function removes or flags records outside of the species' natural ranges according to information provided by the Brazilian Flora 2020 database.

Usage

```
filter_florabr(data, occ, Species = "species", Long = "x", Lat = "y",
               by_State = TRUE, buffer_State = 20, by_Biome = TRUE,
               buffer_Biome = 20, by_Endemism = TRUE,
               Buffer_Brazil = 20, State_vect = NULL,
               state_column = NULL, Biome_vect = NULL,
               biome_column = NULL, BR_vect = NULL,
               value = "flag&clean", keep_columns = TRUE,
               verbose = TRUE)
```

Arguments

data	(data.frame) the data.frame imported with the load_florabr 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_Biome	(logical) filter records by Biome? Default = TRUE
buffer_Biome	(numeric) buffer (in km) around the polygons of the biomes of occurrence of the specie. Default = 20.
by_Endemism	(logical) filter records by endemism? Default = TRUE

Buffer_Brazil	(numeric) buffer (in km) around the polygons of the Brazil. Default = 20.
State_vect	(SpatVector) a SpatVector of the Brazilian states. By default, it uses the SpatVector provided by <code>geobr::read_state()</code> . It can be another Spatvector, but the structure must be identical to <code>geobr::read_state()</code> .
state_column	(character) name of the column in <code>State_vect</code> containing state abbreviations. Only use if <code>Biome_vect</code> is not null.
Biome_vect	(SpatVector) a SpatVector of the Brazilian biomes. By default, it uses the SpatVector provided by <code>geobr::read_biomes()</code> . It can be another SpatVector, but the structure must be identical to <code>geobr::read_biomes()</code> with biome names in English.
biome_column	(character) name of the column in <code>Biome_vect</code> containing names of brazilian biomes (in English: "Amazon", "Atlantic_Forest", "Caatinga", "Cerrado", "Pampa" and "Pantanal". Only use if <code>Biome_vect</code> is not null.
BR_vect	(SpatVector) a SpatVector of Brazil. By default, it uses the SpatVector provided by <code>geobr::read_state()</code> after being aggregated/dissolved,
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
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_Biome = TRUE`, the function takes polygons representing the states and/or Biomes with confirmed occurrences of the specie, draws a buffer around the polygons, and tests if the records of the species fall inside it. If `by_Endemism = TRUE`, the function checks if the species is endemic to Brazil. If it is endemic, the function tests if the records of the specie fall inside a polygon representing the boundaries of Brazil (with a buffer).

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.

References

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Examples

```
data("bf_data") #Load Brazilian Flora data
data("occurrences") #Load occurrences
pts <- subset(occurrences, species == "Myrcia hatschbachii")
fd <- filter_florabr(data = bf_data, occ = pts,
```

```

by_State = TRUE, buffer_State = 20,
by_Biome = TRUE, buffer_Biome = 20,
by_Endemism = TRUE, Buffer_Brazil = 20,
State_vect = NULL,
Biome_vect = NULL, BR_vect = NULL,
value = "flag&clean", keep_columns = TRUE,
verbose = FALSE)

```

get_attributes

Get available attributes to filter species

Description

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

Usage

```
get_attributes(data, attribute, Kingdom = "Plantae")
```

Arguments

data	(data.frame) a data.frame imported with the load_florabr function or a data.frame generated with the select_species function.
attribute	(character) the type of characteristic. See detail to see the options.
Kingdom	(character) the kingdom to which the species belong. It can be "Plantae" or "Fungi". Default = "Plantae".

Details

The attribute argument accepts the following options: Group, SubGroup, family, lifeForm, habitat, vegetationType, Origin, Endemism, Biome, States, taxonomicStatus or nomenclaturalStatus". These options represent different characteristics of species that can be used for filtering.

Value

a data.frame with two columns. The first column provides the available options in English. Use this options in the [select_species](#) function. The second columns provides the options in Portuguese.

References

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Examples

```
data("bf_data") #Load Brazilian Flora data
# Get available biomes to filter species
get_attributes(data = bf_data, Kingdom = "Plantae", attribute = "Biome")
# Get available life forms to filter species
get_attributes(data = bf_data, Kingdom = "Plantae", attribute = "lifeForm")
# Get available states to filter species
get_attributes(data = bf_data, Kingdom = "Plantae", attribute = "States")
```

get_binomial	<i>Extract the binomial name (Genus + specific epithet) from a Scientific Name</i>
--------------	--

Description

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

Usage

```
get_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("Araucaria angustifolia (Bertol.) Kuntze",
        "Butia catarinensis Noblick & Lorenzi",
        "Adesmia paranensis Burkart")
spp_new <- get_binomial(species_names = spp)
spp_new
```

get_florabr	<i>Download the latest version of Brazilian Flora 2020 database</i>
-------------	---

Description

This function downloads the latest or an older version of Brazilian Flora 2020 database, merges the information into a single data.frame, and saves this data.frame in the specified directory.

Usage

```
get_florabr(output_dir, data_version = "latest",
            solve_incongruences = TRUE, overwrite = TRUE,
            verbose = TRUE)
```

Arguments

output_dir	(character) a directory to save the data downloaded from Brazilian Flora 2020
data_version	(character) Version of the Brazilian Flora database to download. Use "latest" to get the most recent version, updated weekly. Alternatively, specify an older version (e.g., data_version = "393.319"). Default value is "latest".
solve_incongruences	Resolve inconsistencies between species and subspecies/varieties information. When set to TRUE (default), species information is updated based on unique data from varieties and subspecies. For example, if a subspecies occurs in a certain biome, it implies that the species also occurs in that biome.
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 Brazilian Flora 2020 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_florabr](#) function for further analysis in R.

References

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Examples

```
## Not run:
#Creating a folder in a temporary directory
#Replace 'file.path(tempdir(), "florabr")' by a path folder to be create in
#your computer
```

```

my_dir <- file.path(file.path(tempdir(), "florabr"))
dir.create(my_dir)
#Download, merge and save data
get_florabr(output_dir = my_dir, data_version = "latest",
            solve_incongruences = TRUE, overwrite = TRUE, verbose = TRUE)

## End(Not run)

```

get_pam

Get a presence-absence matrix

Description

Get a presence-absence matrix of species based on its distribution (States, Biomes and vegetation types) according to Brazilian Flora 2020

Usage

```

get_pam(data, by_Biome = TRUE, by_State = TRUE,
        by_vegetationType = 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_florabr function or generated by either select_species or subset_species functions
by_Biome	(logical) get occurrences by Biome. Default = TRUE
by_State	(logical) get occurrences by State. Default = TRUE
by_vegetationType	(logical) get occurrences by vegetation type. 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 (only by State and Biome)

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

References

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Examples

```
data("bf_data") #Load Brazilian Flora data
#Select endemic and native species of trees with occurrence only in Amazon
am_trees <- select_species(data = bf_data,
                           include_subspecies = FALSE,
                           include_variety = FALSE,
                           Kingdom = "Plantae",
                           Group = "All", Subgroup = "All",
                           Family = "All", Genus = "All",
                           LifeForm = "Tree", filter_LifeForm = "only",
                           Habitat = "All", filter_Habitat = "in",
                           Biome = "Amazon",
                           filter_Biome = "only",
                           State = "All", filter_State = "and",
                           VegetationType = "All",
                           filter_Vegetation = "in",
                           Endemism = "Endemic", Origin = "Native",
                           TaxonomicStatus = "Accepted",
                           NomenclaturalStatus = "All")
#Get presence-absence matrix
pam_am <- get_pam(data = am_trees, by_Biome = TRUE, by_State = TRUE,
                  by_vegetationType = FALSE, remove_empty_sites = TRUE,
                  return_richness_summary = TRUE,
                  return_spatial_richness = TRUE,
                  return_plot = TRUE)
```

Description

Get Spatial polygons (SpatVectors) of species based on its distribution (States and Biomes) according to Brazilian Flora 2020

Usage

```
get_spat_occ(
  data,
  species,
  State = TRUE,
  Biome = TRUE,
  intersection = TRUE,
  State_vect = NULL,
  state_column = NULL,
  Biome_vect = NULL,
  biome_column = NULL,
  verbose = TRUE
)
```

Arguments

data	(data.frame) the data.frame imported with the load_florabr function.
species	(character) one or more species names (only genus and specific epithet, eg. "Araucaria angustifolia")
State	(logical) get SpatVector of states with occurrence of the species? Default = TRUE
Biome	(logical) get SpatVector of biomes with occurrence of the species? Default = TRUE
intersection	(character) get a Spatvector representing the intersection between States and Biomes with occurrence of the specie? To use intersection = TRUE, you must define State = TRUE and Biome = TRUE". Default = TRUE
State_vect	(SpatVector) a SpatVector of the Brazilian states. By default, it uses the SpatVector provided by <code>geobr::read_state()</code> . It can be another Spatvector, but the structure must be identical to <code>geobr::read_state()</code> .
state_column	(character) name of the column in State_vect containing state abbreviations. Only use if Biome_vect is not null.
Biome_vect	(SpatVector) a SpatVector of the Brazilian biomes. By default, it uses the SpatVector provided by <code>geobr::read_biomes()</code> . It can be another SpatVector, but the structure must be identical to <code>geobr::read_biomes()</code> .
biome_column	(character) name of the column in Biome_vect containing names of brazilian biomes (in English: "Amazon", "Atlantic_Forest", "Caatinga", "Cerrado", "Pampa" and "Pantanal". Only use if Biome_vect is not null.
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 Biomes and/or Intersections for each specie.

References

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Examples

```
library(terra)
data("bf_data") #Load Brazilian Flora data
spp <- c("Araucaria angustifolia", "Adesmia paranensis") #Example species
#Get states, biomes and intersection states-biomes of species
spp_spt <- get_spat_occ(data = bf_data, species = spp, State = TRUE,
                      Biome = TRUE, intersection = TRUE, State_vect = NULL,
                      Biome_vect = NULL, verbose = TRUE)

#Plot states of occurrence of Araucaria angustifolia
plot(spp_spt[[1]]$States, main = names(spp_spt)[[1]])
#Plot biomes of occurrence of Araucaria angustifolia
plot(spp_spt[[2]]$Biomes, main = names(spp_spt)[[2]])
#Plot intersection between states and biomes of occurrence of
#Araucaria angustifolia
plot(spp_spt[[1]]$States_Biomes)
```

load_florabr

Load Brazilian Flora database

Description

Load Brazilian Flora database

Usage

```
load_florabr(data_dir, data_version = "Latest_available",
             type = "short", verbose = TRUE)
```

Arguments

<code>data_dir</code>	(character) the same directory used to save the data downloaded from Brazilian Flora 2020 using the get_florabr function.
<code>data_version</code>	(character) the version of Brazilian Flora database to be loaded. It can be "Latest_available", which will load the latest version available; or another specified version, for example "393.364". Default = "Latest_available".
<code>type</code>	(character) it determines the number of columns that will be loaded. It can be "short" or "complete". Default = "short". See details.
<code>verbose</code>	(logical) Whether to display messages during function execution. Set to TRUE to enable display, or FALSE to run silently. Default = TRUE.

Details

The parameter type accepts two arguments. If type = short, it will load a data.frame with the 19 columns needed to run the other functions of the package: species, scientificName, acceptedName, kingdom, Group, Subgroup, family, genus, lifeForm, habitat, Biome, States, vegetationType, Origin, Endemism, taxonomicStatus, nomenclaturalStatus, vernacularName, and taxonRank. If type = complete, it will load a data.frame with all 39 variables available in Brazilian Flora database.

Value

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

References

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Examples

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

## End(Not run)
```

occurrences

Records of plant species

Description

A dataset containing records of 7 plant species downloaded from GBIF. The records were obtained with `plantR::rgbif2`

Usage

```
data(occurrences)
```

Format

A data.frame with 1521 rows and 3 variables:

species Species names (Araucaria angustifolia, Abatia americana, Passiflora edmundoi, Myrcia hatschbachii, Serjania pernambucensis, Inga virescens, and Solanum restingae)

x Longitude

y Latitude

select_by_vernacular *Search for taxa using vernacular names*

Description

Search for taxa using vernacular names

Usage

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

Arguments

data	(data.frame) the data.frame imported with the <code>load_florabr</code> function or generated with the function <code>select_species</code> .
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 = "pinheiro" and exact = TRUE, the function will return only the species popularly known as "pinheiro". On the other hand, if names = "pinheiro" and exact = FALSE, the function will return other results as "pinheiro-do-parana". Default = FALSE

Value

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

References

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Brazilian Flora 2020. Jardim Botânico do Rio de Janeiro. Available at: <http://floradobrasil.jbrj.gov.br/>

Examples

```

data("bf_data") #Load Brazilian Flora data
#Search for species whose vernacular name is 'pinheiro'
pinheiro_exact <- select_by_vernacular(data = bf_data,
                                     names = "pinheiro",
                                     exact = TRUE)

pinheiro_exact
#Search for species whose vernacular name is 'pinheiro', allowing non-exact
#matches
pinheiro_not_exact <- select_by_vernacular(data = bf_data,
                                           names = "pinheiro",
                                           exact = FALSE)

head(pinheiro_not_exact)

```

select_species

Selection of species based on its characteristics and distribution

Description

select_species allows filter species based on its characteristics and distribution available in Brazilian Flora 2020

Usage

```

select_species(data,
               include_subspecies = FALSE, include_variety = FALSE,
               Kingdom = "Plantae", Group = "All", Subgroup = "All",
               Family = "All", Genus = "All",
               LifeForm = "All", filter_LifeForm = "in",
               Habitat = "All", filter_Habitat = "in",
               Biome = "All", filter_Biome = "in",
               State = "All", filter_State = "in",
               VegetationType = "All", filter_Vegetation = "in",
               Endemism = "All", Origin = "All",
               TaxonomicStatus = "Accepted",
               NomenclaturalStatus = "All")

```

Arguments

data (data.frame) the data.frame imported with the [load_florabr](#) function.

include_subspecies (logical) include subspecies? Default = FALSE

include_variety (logical) include varieties of the species? Default = FALSE

Kingdom (character) The Kingdom for filtering the dataset. It can be "Plantae" or "Fungi". Default = "Plantae". To include both, use c("Plantae", "Fungi")

Group	(character) The groups for filtering the datasets. It can be "Fungi", "Angiosperms", "Gymnosperms", "Ferns and Lycophytes", "Bryophytes" and "Algae". To use more than one group, put the available items in a vector, for example: Group = c("Angiosperms", "Gymnosperms"). Default = "All".
Subgroup	(character) The subgroups for filtering the dataset. Only available if the Group is "Fungi" or "Bryophytes". For Fungi, it can be "stricto sensu" or "lato sensu". For Bryophytes, it can be "Mosses", "Hornworts" and "Liverworts". To use more than one group, put the available items in a vector, for example: Subgroup = c("Mosses", "Hornworts"). 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.
Biome	(character) The biomes for filtering the dataset. It can be included more than one biome. Default = "All"
filter_Biome	(character) The type of filtering for biome. It can be "in", "only", "not_in" and "and". See details for more about this argument.
State	(character) The States for filtering the dataset. It can be included more than one state. Default = "All".
filter_State	(character) The type of filtering for state. It can be "in", "only", "not_in" and "and". See Details for more about this argument.
VegetationType	(character) The vegetation types for filtering the dataset. It can be included more than one vegetation type. Default = "All".
filter_Vegetation	(character) The type of filtering for vegetation type. It can be "in", "only", "not_in" and "and". See details for more about this argument.
Endemism	(character) The endemism (endemic or non-endemic to Brazil) for filtering the dataset. It can be "All", "Endemic" or "Non-endemic". Default = "All".
Origin	(character) The origin for filtering the dataset. It can be "All", "Native", "Cultivated" and "Naturalized". Default = "All".
TaxonomicStatus	(character) The taxonomic status for filtering the dataset. It can be "All", "Accepted" or "Synonym". Default = "Accepted".
NomenclaturalStatus	(character) The nomenclatural status for filtering the dataset. Default = "Accepted"

states	<i>SpatVector of the federal states of Brazil</i>
--------	---

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_species	<i>Extract a subset of species from Brazilian Flora 2020 database</i>
----------------	---

Description

Returns a `data.frame` with a subset of species from Brazilian Flora 2020 database

Usage

```
subset_species(data, species,
               include_subspecies = FALSE,
               include_variety = FALSE,
               Kingdom = "Plantae")
```

Arguments

data	(<code>data.frame</code>) the <code>data.frame</code> imported with the <code>load_florabr</code> function.
species	(character) names of the species to be extracted from Brazilian Flora database.
include_subspecies	(logical) include subspecies? Default = FALSE
include_variety	(logical) include varieties of the species? Default = FALSE
Kingdom	(character) The Kingdom for filtering the dataset. It can be "Plantae" or "Fungi". Default = "Plantae". To include both, use <code>c("Plantae", "Fungi")</code>

Index

* datasets

- Attributes, [2](#)
- bf_data, [2](#)
- biomes, [3](#)
- brazil, [4](#)
- occurrences, [15](#)
- states, [20](#)

agrep, [4](#)

Attributes, [2](#)

bf_data, [2](#)

biomes, [3](#)

brazil, [4](#)

check_names, [4](#)

check_version, [5](#)

filter_florabr, [6](#)

get_attributes, [8](#), [19](#)

get_binomial, [9](#)

get_florabr, [10](#), [14](#)

get_pam, [11](#)

get_spat_occ, [12](#)

load_florabr, [4](#), [6](#), [8](#), [10](#), [11](#), [13](#), [14](#), [16](#), [17](#),
[20](#)

occurrences, [15](#)

select_by_vernacular, [16](#)

select_species, [8](#), [11](#), [16](#), [17](#)

states, [20](#)

subset_species, [11](#), [20](#)