Package: FloraIberica (via r-universe)

August 17, 2024

Title Taxonomic and distribution data for all the vascular plants present in the Iberian Peninsula and Balearic Islands

Version 0.0.3

Description Check which plant taxa are present and/or endemic, and get detailed (10x10 km) distribution maps for all the vascular plants present in the Iberian Peninsula and Balearic Islands, from genus to subspecies level, based on the AFLIBER database (<doi:10.1111/geb.13363>).

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

URL https://github.com/Pakillo/FloraIberica

BugReports https://github.com/Pakillo/FloraIberica/issues

Suggests dplyr, testthat (>= 3.0.0)

Config/testthat/edition 3

Imports ggplot2, nngeo, sf

Depends R (>= 2.10)

LazyData true

LazyDataCompression bzip2

Repository https://pakillo.r-universe.dev

RemoteUrl https://github.com/Pakillo/FloraIberica

RemoteRef HEAD

RemoteSha 25b8bfa2f20bd6de9aeae67547b515ac09646b1d

Contents

Distributions   *Taxa distributions*

Description

Taxa distributions

Usage

```
Distributions
```

Format

`Distributions`:

A data frame with 1,824,549 rows and 6 columns:

Genus genus

Species species

Subspecies subspecies

UTM.cell Name of the 10x10 km UTM.cell

lng longitude in decimal degrees

lat latitude in decimal degrees

Source

---

get_checklist · · · · · · · · · · · *Get a zonal checklist*

Description

Get a checklist of the vascular plants growing near a given point or polygon.

Usage

```
get_checklist(zone = NULL, sf = FALSE)
```

Arguments

zone
: A vector of two numbers giving the point longitude and latitude (in that order), an sf object with a single point coordinate, or a polygon sf object.

sf
: Logical. If FALSE (default) return a dataframe with the checklist. If TRUE, return an sf object with the coordinates where each taxa is found. Note these point coordinates represent the centre of 10 km resolution UTM grid cells, not the actual location of these plants.

Value

A dataframe or sf object

Note

As the original data (AFLIBER database) have 10-km resolution, the resulting checklist may include taxa present within 10 km distance of the point or polygon.

Examples

```
sitio <- c(-5, 40)
head(get_checklist(sitio))
```

---

get_distribution · · · · · · · · *Get the distribution of one or more plant taxa*

Description

Get the distribution of taxa on a 10 x 10 km UTM grid. Point coordinates represent the center of each grid cell, not actual plant locations.

Usage

```
get_distribution(genus = NULL, species = NULL, subspecies = NULL, sf = TRUE)
```

Arguments

genus	character. One or more genera to check for presence. Required, unless `gbif.id` is provided.
species	character. Optional. One or more species names to check for presence. The length of genus must equal that of `species`, unless length(genus) == 1, in which case it will be assumed that all species belong to that same genus.
subspecies	character. Optional. One or more subspecies names to check for presence. The length of `species` must equal that of `subspecies`, unless length(species) == 1, in which case it will be assumed that all subspecies belong to that same species.
sf	Logical. Return a spatial (sf) or a plain dataframe?

Value

An sf object if sf = TRUE, a plain dataframe otherwise.

Examples

```
abies <- get_distribution("Abies")
abies
unique(abies$Species)  # including all species in the genus

pinsapo <- get_distribution("Abies", "pinsapo")
pinsapo
```

IberianPeninsula   *Iberian Peninsula contour*

Description

Iberian Peninsula contour

Usage

```
IberianPeninsula
```

Format

`IberianPeninsula`:

A sfc_MULTIPOLYGON object with the contour of Portugal, Peninsular Spain and Balearic Islands

Source

<https://www.naturalearthdata.com/>

`is_endemic`   *Is taxon endemic?*

Description

Is taxon endemic of the Iberian Peninsula and/or Balearic Islands? That is, not present naturally elsewhere in the world.

Usage

```
is_endemic(genus = NULL, species = NULL, subspecies = NULL, gbif.id = NULL)
```

Arguments

genus
: character. One or more genera to check for presence. Required, unless `gbif.id` is provided.

species
: character. Optional. One or more species names to check for presence. The length of genus must equal that of `species`, unless length(genus) == 1, in which case it will be assumed that all species belong to that same genus.

subspecies
: character. Optional. One or more subspecies names to check for presence. The length of `species` must equal that of `subspecies`, unless length(species) == 1, in which case it will be assumed that all subspecies belong to that same species.

gbif.id
: character. Optional.

Details

For a genus to be considered endemic, all its species must be endemic. Likewise, for a species to be considered endemic, all its subspecies must be endemic.

Value

A logical vector

Examples

```
is_endemic("Laurus")
is_endemic("Aconitum")
is_endemic("Aconitum", "variegatum")
is_endemic("Aconitum", "napellus")
is_endemic("Aconitum", "napellus", c("vulgare", "castellanum", "lusitanicum"))
```

---

is_present   *Is taxon present in the Iberian Peninsula and/or Balearic Islands?*

Description

Is taxon present in the Iberian Peninsula and/or Balearic Islands?

Usage

```
is_present(genus = NULL, species = NULL, subspecies = NULL, gbif.id = NULL)
```

Arguments

genus
: character. One or more genera to check for presence. Required, unless gbif.id is provided.

species
: character. Optional. One or more species names to check for presence. The length of genus must equal that of species, unless length(genus) == 1, in which case it will be assumed that all species belong to that same genus.

subspecies
: character. Optional. One or more subspecies names to check for presence. The length of species must equal that of subspecies, unless length(species) == 1, in which case it will be assumed that all subspecies belong to that same species.

gbif.id
: character. Optional.

Value

A logical vector

Examples

```
is_present("Laurus")
is_present("Laurus", "nobilis")
is_present("Laurus", "azorica")
is_present("Laurus", c("nobilis", "azorica"))
is_present(gbif.id = "3034015")
```

---

map_distribution   *Map taxa distributions*

Description

Get a map with the distribution of one or more taxa. Must provide either an sf object (as returned by [get_distribution()](#)) or genus, species, subspecies...

Usage

```
map_distribution(
  distrib.sf = NULL,
  genus = NULL,
  species = NULL,
  subspecies = NULL,
  taxo.level = "species",
  facet = FALSE,
  colour = "medium sea green",
  include.name = TRUE,
  ...
)
```

Arguments

distrib.sf	An sf object as returned by [get_distribution()]().
genus	character. One or more genera to check for presence. Required, unless gbif.id is provided.
species	character. Optional. One or more species names to check for presence. The length of genus must equal that of species, unless length(genus) == 1, in which case it will be assumed that all species belong to that same genus.
subspecies	character. Optional. One or more subspecies names to check for presence. The length of species must equal that of subspecies, unless length(species) == 1, in which case it will be assumed that all subspecies belong to that same species.
taxo.level	character Taxonomic level to show in the map. Either 'genus', 'species' (default) or 'subspecies'. If 'subspecies' argument is provided, taxo.level is automatically changed to 'subspecies'.
facet	Logical. For multiple taxa, make a single map with all taxa together, or make a multipanel (facetted) figure with one panel per taxa?
colour	character. When there is >1 taxon, only used if facet = TRUE.
include.name	Logical. When there is a single taxon to map, use taxon name as title?
...	Further params to be passed to [ggplot2::facet_wrap()]() if facet = TRUE, or to [ggplot2::geom_sf()]() if facet = FALSE.

Value

A map and ggplot2 object

Examples

```
laurus <- get_distribution("Laurus", "nobilis")
map_distribution(laurus)

map_distribution(genus = "Laurus", species = "nobilis")

abies <- get_distribution("Abies")
map_distribution(abies)
```

```
map_distribution(abies, facet = TRUE, ncol = 1)
map_distribution(abies, taxo.level = "genus")

# Map all the subspecies of a species
map_distribution(genus = "Berberis", species = "vulgaris", taxo.level = "subspecies")
map_distribution(genus = "Berberis", species = "vulgaris", subspecies = "seroi")
```

Taxa	*Plant taxa*

Description

Plant taxa

Usage

```
Taxa
```

Format

`Taxa`:

A data frame with 6,456 rows and 11 columns:

Taxon Full taxon name

Scientific_Name Scientific name

Endemic Endemic in the Iberian Peninsula and/or Balearic Islands (TRUE/FALSE)

Genus genus

Species species

Subspecies subspecies

Class class

Order order

Family family

GBIF_id GBIF unique ID for that taxon

POW_Name Name of the taxon in Plants of the World database

Source

https://doi.org/10.1111/geb.13363

Index