An Introduction to **plantR**

Renato A. F. de Lima[*], Sara R. Mortara[†]and Andrea Sánchez-Tapia[‡]

05 October 2021

Contents

1 Introduction

The databases of biological collections are becoming increasingly available online, providing an unprecedented amount of species records for biodiversity-related studies. Managing the information associated with species records is an important but difficult task. The notation of collectors' names, numbers, and dates varies between collections, and sometimes within them. In addition, it is often difficult to validate the localities, geographical coordinates and identifications associated with individual species records, especially when working with thousands or millions of them. Thus, having tools to process and validate large amounts of records can be quite handy.

plantR is an R package that was developed to manage, standardize, and validate the information associated with species records from biological collections (e.g., herbaria). It can be used for data coming from a single collection or different biodiversity databases, such as GBIF. Moreover, **plantR** can be used by collection curators to manage their databases and by final users of species records (e.g., taxonomists, ecologists, and conservationists), allowing the comparison of data across collections.

1.1 Main features and workflow

The package **plantR** provides tools to standardize the information from typical fields associated with species records, such as collectors' and species names. In addition, **plantR** proposes a comprehensive and reproducible workflow to apply those tools while handling records from biological collections, which includes the following steps:

[*] Naturalis Biodiversity Center and Universidade de São Paulo, https://github.com/LimaRAF
[†] Jardim Botânico do Rio de Janeiro, https://github.com/saramortara
[‡] Jardim Botânico do Rio de Janeiro,https://github.com/AndreaSanchezTapia

1. import or download of species records for a list of species names, collections codes or other search fields;
2. batch standardization of typical fields (e.g., collector name);
3. validation of the locality and geographical coordinates of the records, based on maps and gazetteers;
4. spell-checking and validation of botanical families and species names using different taxonomic backbones (e.g. Flora do Brasil);
5. assessment of the confidence level of species identifications, based on a global list of plant taxonomists;
6. retrieval of duplicated specimens across collections, including the homogenization of the information within duplicates;
7. summary of species data and validation steps, and (fast) export of the validated records by groups (e.g. families or countries).

1.2 Basic assumptions and limitations

The tools provided by **plantR** do not edit the columns with the original information. All the outputs of each editing or validation step are stored in new, separated columns added to the original data. This is important for the collection curation process because it allows the comparison between the original and edited information. However, it increases the number of columns in the dataset, which may become a problem while managing and saving big datasets.

plantR was initially developed to manage plant records from herbaria. Therefore, some of the tools offered by the package are exclusive to plants, particularly the checking of species names. However, some of its main features are expected to work for other groups of organisms as well, as long as the data structure is similar.

Currently, the download of records is available for the Global Biodiversity Information Facility (GBIF), and speciesLink, but the user can also provide their own dataset as an input. Future versions of the package may include the download from data stored in JABOT.

Name editing and standardization cover most of the typical variation in the notation of people's names, trying to provide standardized outputs in the TDWG format. The same applies to collection codes, collector numbers and dates. However, **plantR** does not handle all possibilities of notation. So, some double-checking and corrections may be needed depending on the user's goals.

Regarding the validation of geographical coordinates. In the case of invalid or missing coordinates, we assume that the locality information associated with the record (e.g. country, state, county) is correct (i.e. locality prevails over coordinates), and so working coordinates are taken from a gazetteer. It is important to note that if the locality information is indeed mistaken (e.g., wrong county name), then even if the original coordinates are good, they will not be validated (record locality and coordinate locality don't match) and may be replaced by coordinates taken from the gazetteer.

Currently, geographical validation can be performed at the county level for Latin American countries and at the country level for the rest of the world. We provide a gazetteer to retrieve and check localities and geographical coordinates, which is currently biased towards Latin American countries, particularly Brazil. Therefore, the validation of geographical coordinates provided by other R packages (e.g. **CoordinateCleaner**) may be more appropriate for studies extending outside Latin America.

Taxonomic validation is performed based on (i) the correction of plant family and species names (i.e. synonyms, typos) and (ii) the confidence level on the species identification, based on a dictionary of plant taxonomist names from all over the world. For (i), names are currently checked against the Flora do Brasil project and The Plant List, using the R packages **flora** and **Taxonstand**. Future versions may include comparisons against Tropicos and GBIF backbones.

During the assessment of the taxonomic confidence level of the identifications, we did not attempt to set priorities for different specialists within a given family. That is, all species names determined by a specialist within their family of expertise are taken as being correct. Although we recognize that there are specialists for genera within a family, the validation process is currently performed only at the family level. In the case of conflicting species identification among family specialists for duplicates across collections, we assume the most recent identification as being the valid one.

plantR provide tools for searching for duplicated records across collections. This search makes more sense when data from different collections are combined and it performs well even when using relatively large datasets (i.e., millions of records).

However, the retrieval of duplicates greatly depends on the completeness of the input information, the notation standards and if **plantR** is able to handle those differences in notation across collections. In addition, true duplicates may not be found due to typos and false duplicates may be returned if the duplicate search fields are too flexible.

2 Using plantR

2.1 Installation

The package can be installed and loaded from GitHub with:

```r
install.packages("remotes")
library("remotes")
install_github("LimaRAF/plantR")
library("plantR")
```

2.2 Main features

2.2.1 Data entry

Users can provide their own dataset, import it from a GBIF DwC-A zip file (function `readData()`) or download data directly from R using one of **plantR** download functions. They include the function `rspeciesLink()`:

```r
occs_splink <- rspeciesLink(species = "Euterpe edulis")
## Making request to speciesLink...
## Please make sure that the restrictions and citation indicated by
##   each speciesLink/CRIA data provider are observed and respected.
```

This function can also be used to search from records based on localities, collections, and other options (see `?rspeciesLink` for details).

plantR also provides the function `rgbif2()`, which is a wrapper of the function `rgbif()` of the **rgbif** package, with a standardized output:

```r
occs_gbif <- rgbif2(species = "Euterpe edulis")
## Making request to GBIF for Euterpe edulis...
## Please make sure that the restrictions and citation indicated by
##   each GBIF data provider are observed and respected.
```

2.2.1.1 Field names It is important to make sure that the field names of the input data follow the DarwinCore format. In **plantR** this is performed using the function `formatDwc()`, which joins data from different sources (e.g. GBIF and speciesLink) and standardizes their field names:

```r
occs <- formatDwc(splink_data = occs_splink,
                  gbif_data = occs_gbif)
```

2.2.2 Data editing

2.2.2.1 Collection codes, people names, collector number and dates The names of the collections, collectors, and identifiers, as well as the collection numbers and dates, can be edited using the function `formatOcc()`:

```
occs <- formatOcc(occs)
```

2.2.2.2 Locality information The locality information associated with the occurrence data (e.g., country or city names) can be standardized using the function `formatLoc()`:

```
occs <- formatLoc(occs)
```

2.2.2.3 Geographical coordinates The geographical coordinates are prepared using function `formatCoord()`, which guarantees that they are in a good format for validation (i.e., decimal degrees). This function also retrieves missing coordinates from a gazetteer based on the locality information:

```
occs <- formatCoord(occs)
```

2.2.2.4 Species and family names In this example, although we have downloaded data for a single species (i.e., *Euterpe edulis* Mart.), there are differences in the notation of botanical family and species names, some of them being synonyms. To obtain only valid names, we use the function `formatTax()`:

```
occs <- formatTax(occs)
```

```
## The following family names were automatically replaced:
##
## |Genus   |Old fam. |New fam.  |
## |:-------|:--------|:---------|
## |Euterpe |Palmae   |Arecaceae |
```

2.2.3 Data validation

2.2.3.1 Locality information Once the new columns with the edited and standardized information are available, the records can be validated. The first validation step regards the locality information, which is done using the function `validateLoc()`:

```
occs <- validateLoc(occs)
```

```
## [1] "Locality resolution in the original data vs. edited data:"
##                original
## edited          country locality municipality no_info stateProvince
##   country           186        2            1       0             1
##   locality            4      303            0       0             0
##   municipality        0      736          207       0             0
##   no_info             0        0            0      29             0
##   stateProvince       0      153            5       0            89
```

2.2.3.2 Geographical coordinates The second validation step regards the geographical coordinates of the records, which is performed using the function `validateCoord()`:

```
occs <- validateCoord(occs)
```

2.2.3.3 Species taxonomy and identification The next validation step regards the confidence level in the species identification, which is one of the main **plantR** features and executed by function `validateTax()`:

```
occs <- validateTax(occs)
```

```
## Top people with many determinations but not in the taxonomist list:
##
## |Identifier       | Records|
## |:----------------|-------:|
## |Caxambu, M.G.    |      66|
## |Verdi, M.        |      22|
## |Reis, A.         |      19|
## |Wandekoken, D.T. |      18|
## |Guedes, M.L.     |      15|
## |Silva, E.F.L.P.  |      15|
## |Rossato, M.      |      14|
## |Medeiros-Costa   |      12|
## |Ribeiro, M.      |      12|
## |Thomas, W.W.     |      11|
```

Note that the function returns up to 10 names of determiners not taken as specialists of the family. The argument `miss.taxonomist` can be used to include missing names of taxonomists (e.g., `miss.taxonomist = c("Arecaceae_Reis, A.")`).

2.2.3.4 Duplicate specimens Another main feature of **plantR** is the search for duplicates across herbaria (i.e., same biological specimen with accession numbers in two or more collections). It uses different combinations of search strings to find direct and indirect links between records. Besides the search itself, the user can also homogenize information within groups of duplicates, such as species names or geographical coordinates. This tool is performed using the function `validateDup()`:

```
occs <- validateDup(occs)
## 608 truly duplicate records (same record in different sources) were removed from the data
```

2.2.4 Data summary and export

Once the editing and validation steps are finished, **plantR** provides tools for summarizing the occurrence data, using the function `summaryData()`. In this example, the taxonomic summary is quite uninformative, since we have only one species.

```
summ <- summaryData(occs)
```

```
## =========
##  RECORDS
## =========
## |Type                     | Records|
## |:------------------------|-------:|
## |Unicates                 |     103|
## |Duplicates               |     546|
## |Unknown                  |     459|
## |Total without duplicates |     894|
## |Total with duplicates    |    1108|
##
## =============
##  COLLECTIONS
## =============
## Number of biological collections: 128
## Number of collectors' names: 391
## Collection years: 1816-2021 (>90% and >50% after 1967 and 2001)
##
## Top collections in numbers of records:
## |Collection   | Records|
## |:------------|-------:|
## |RB           |      89|
## |Observations |      67|
## |HCF          |      59|
## |SINBIOTA     |      57|
## |MBML         |      50|
##
## Top collectors in numbers of records:
## |Collector         | Records|
## |:-----------------|-------:|
## |Fernandes, H.Q.B. |      66|
## |Caxambu, M.G.     |      35|
## |Lima, H.C.        |      27|
## |Glaziou, A.       |      21|
## |Noblick, L.R.     |      21|
##
## ==========
##  TAXONOMY
## ==========
## Number of families: 1
## Number of genera: 1
## Number of species: 1
##
## Top richest families:
## |family.new |    N|  S|
## |:----------|----:|--:|
## |Arecaceae  | 1108|  1|
##
## Top richest genera:
## |genus.new |    N|  S|
## |:---------|----:|--:|
## |Euterpe   | 1108|  1|
##
## ===========
##  COUNTRIES
## ===========
## Number of countries: 17
##
## Top countries in numbers of records:
## |Country   | Records| Species|
## |:---------|-------:|-------:|
## |Brazil    |    1017|       1|
## |Argentina |      28|       1|
## |[Unknown] |      26|       1|
## |Paraguay  |      19|       1|
## |Guyana    |       4|       1|
```

plantR also provides an overview of the validation results (function `summaryFlags()`):

```
flags <- summaryFlags(occs)
```

```
## ==================
##  DUPLICATE SEARCH
## ==================
## Records per strength of duplicate indication:
##
## |Strenght                | Records|
## |:-----------------------|-------:|
## |0%                      |     103|
## |25%                    |      37|
## |50%                    |       7|
## |100%                   |     502|
## |Cannot check (no info) |     459|
##
## =====================
##  LOCALITY VALIDATION
## =====================
## Results of the locality validation:
##
## |Validation           | Records|
## |:--------------------|-------:|
## |ok (same resolution) |     566|
## |probably ok          |     437|
## |check (downgraded)   |     101|
## |ok (upgraded)        |       4|
##
## Details of the validation (original vs. validated localities):
##
## |original.resolution | no_info| country| stateProvince| municipality| locality|
## |:-------------------|-------:|-------:|-------------:|------------:|--------:|
## |no_info             |      26|       0|             0|            0|        0|
## |country             |       0|     169|             0|            0|        4|
## |stateProvince       |       0|       1|            84|            0|        0|
## |municipality        |       0|       1|             3|          151|        0|
## |locality            |       0|       2|            93|          409|      165|
##
## =======================
##  COORDINATE VALIDATION
## =======================
## Valid coordinates per origin:
##
## |Validated |Origin       | Records|
## |:---------|:------------|-------:|
## |yes       |original     |     823|
## |yes       |gazetter     |     259|
## |no        |cannot_check |      26|
##
## Valid coordinates per resolution:
##
## |Validated |Resolution          | Records|
## |:---------|:-------------------|-------:|
## |yes       |ok_county           |     648|
## |yes       |ok_state            |     228|
## |yes       |ok_country          |     175|
## |no        |no_cannot_check     |      26|
## |yes       |ok_locality         |      26|
## |yes       |shore               |       4|
## |yes       |bad_country[border] |       1|
##
## ======================
##  CULTIVATED SPECIMENS
## ======================
## Number of specimens from cultivated individuals:
```

```
##
## |Cultivated   | Records|
## |:------------|-------:|
## |probably not |    1096|
## |probably yes |      11|
## |yes          |       1|
##
## ======================
##  TAXONOMIC CONFIDENCE
## ======================
## Confidence level of the taxonomic identifications:
##
## |Confidence | Records|
## |:----------|-------:|
## |unknown    |     467|
## |low        |     428|
## |high       |     213|
```

The package **plantR** can also build species checklists with vouchers using the function `checkList()`:

```
checkList(occs, n.vouch = 3, type = "short")
```

```
##   family.new scientificName.new records tax.CL geo.CL
## 1  Arecaceae     Euterpe edulis    1108  19.22  60.83
##
## 1 Fernandes, H.Q.B., 2519 (MBML 5289) [paratype]; Fernandes, H.Q.B., 2543 (MBML 5288, R-TIPOS 174930)
```

Finally, **plantR** exports data into a local folder, using function `saveData()`, which can be used to save compressed '.csv' files based on different grouping fields (e.g., botanical family, country, biological collection). The export is performed using function `fwrite()` from package **data.table** which is quite fast even for large datasets.

3 Citation

If you use this package, please cite it as:

Lima, R.A.F., Sánchez-Tapia, A., Mortara, S.R., ter Steege, H., Siqueira, M.F. (2021). *plantR*: An R package and workflow for managing species records from biological collections. bioRxiv: 2021.04.06.437754. https://doi.org/10.1101/2021.04.06.437754

If you use the function `prepSpecies()`, please also cite the following packages (depending on the database used):

Carvalho, G. (2020) flora: Tools for Interacting with the Brazilian Flora 2020. R package version 0.3.4. https://CRAN.R-project.org/package=flora

Cayuela, L., Macarro, I., Stein, A. and Oksanen, J. (2021) Taxonstand: Taxonomic Standardization of Plant Species Names. R package version 2.3. https://CRAN.R-project.org/package=Taxonstand

If you use the function `rgbif2()`, please also cite the following package:

Chamberlain, S. et al. (2021) rgbif: Interface to the Global Biodiversity Information Facility API. R package version 3.5.2. https://CRAN.R-project.org/package=rgbif>.