Cleaning species occurrence records

Ian Ondo

2024-02-05

Contents

Package notes:
We need to install the following set of R packages to successfully run the codes in this vignette:

- **CoordinateCleaner** (Zizka et al. 2019)
- **TDWG** (Ondo 2024)
- **dplyr** (Wickham et al. 2023)

Introduction

Cleaning and filtering out data that look spurious or do not fit the purpose of a study is probably the most important step in any data science project. It is generally a good idea to start your data curation by applying general filters that would exclude observations that do not fit the scope of your study, such as records out of their expected spatial and temporal range. Then, apply more specific filters to refine your selection.

In this vignette, we are going to use the World Checklist of Vascular Plants (WCVP) (Govaerts et al. 2021) distribution data and the R packages **TDWG** and **CoordinateCleaner**, to exclude occurrence locations outside of species known range and common georeferencing errors.

I. *Cleaning records outside of species range* (for vascular plants only !)

Let's assume that you loaded your occurrence records in the dataframe `mySpecies_occurrences`. We are going to use the package **TDWG** (Ondo 2024) to filter out records sampled outside of the known range of your species according to the WCVP database and the World Geographical Scheme for Recording Plant Distributions (WGSRPD).

```
library(TDWG)

# check the manual help for more information about the function `coordinatesCleaner`
clean0 <- TDWG::coordinatesCleaner(
                  # your species occurrence dataframe
                   mySpecies_occurrences,
```

```
                    # indicate the species_name
                    species_name = "Abies spectabilis",
                    # indicate the establishment range you want to filter against
                    # both <=> combined native+introduced range
                    status='both',
                    # geographic level at which the filter must be applied
                    initial_level=2,
                    # do not run in parallel
                    do.parallel=FALSE,
                    # return the full dataset (i.e. all columns) with filtered occurrences
                    full_data = TRUE,
                    # run quietly
                    verbose=FALSE)
```

If everything runs smoothly, `clean0` should be your initial dataset but with the occurrence records sampled outside of your species range removed.

II. *Cleaning unfit and mis-georefenced records*

We use metadata information kept from our primary downloads (see vignette *Data gathering and pre-processing*) to exclude locations:

- collected before 1945
- with coordinates out of their expected range (i.e. with Lat $\notin$ [-90,90] and Lon $\notin$ [-180,180])
- with coordinates uncertainty <= 20km
- with no decimal place (i.e. rounded)

Then, **CoordinateCleaner** to exclude locations:

- with zero coordinates
- with identical latitude/longitude
- within 10km buffer distance from centroids of countries, provinces, capitals or biodiversity institutions (museums, zoos, botanical gardens, universities)
- within 1km buffer distance from the headquarters of GBIF

```
library(dplyr)
library(CoordinateCleaner)

#-------------------------------------------------------------
##= 1. Remove records beyond a certain date
#-------------------------------------------------------------
clean <- clean0 %>%
  dplyr::filter(is.na(year) | year >= 1945)

if(nrow(clean)==0) stop('No more records available in the dataset')

#-------------------------------------------------------------
##= 2. Remove cultivated specimens
#-------------------------------------------------------------
# clean <- clean %>%
#
#   dplyr::filter(is.na(is_cultivated_observation) | is_cultivated_observation == "Yes"
#                 & is.na(basisOfRecord) | basisOfRecord!= "Cultivated habitat")
#
# if(nrow(clean)==0) return(NULL)
```

```
#-------------------------------------------------------------
##= 3. Remove (un/mis-)georeferenced specimens
#-------------------------------------------------------------
clean <- clean %>%
  dplyr::filter(!is.na(decimalLongitude)) %>%
  dplyr::filter(!is.na(decimalLatitude)) %>%

  # Occurences with impossible lat-long coordinates
  dplyr::filter(decimalLatitude < 90 | decimalLatitude > -90 |
                  decimalLongitude < 180 | decimalLongitude > -180)

if(nrow(clean)==0) stop('No more records available in the dataset')

#-------------------------------------------------------------
##= 4. Remove coordinates with large uncertainty in m
#-------------------------------------------------------------
clean <- clean %>%
  dplyr::filter(coordinateUncertaintyInMeters <= 20000 |
                  is.na(coordinateUncertaintyInMeters))

if(nrow(clean)==0) stop('No more records available in the dataset')

#-------------------------------------------------------------
##= 5. Remove coordinates with no decimal place (rounded)
#-------------------------------------------------------------
clean <- clean %>%
  dplyr::filter((decimalLongitude*10) %% 10 != 0) %>%
  dplyr::filter((decimalLatitude*10) %% 10 != 0)

if(nrow(clean)==0) stop('No more records available in the dataset')

#-------------------------------------------------------------
##= 6. Clean records whose individual counts are 0s
#-------------------------------------------------------------
clean <- clean %>%
  dplyr::filter(individualCount > 0 | is.na(individualCount)) %>%
  dplyr::filter(individualCount < 99 | is.na(individualCount))

if(nrow(clean)==0) stop('No more records available information in the dataset')

#=============================================================
##
##=  Start CoordinateCleaner cleaning process
##
#=============================================================
# Rename lon/lat fields
clean <- clean %>%
  dplyr::rename(decimallongitude = 'decimalLongitude',
                decimallatitude = 'decimalLatitude')

#-------------------------------------------------------------
##= 7. Clean records with 0 long and lat
#-------------------------------------------------------------
```

```
clean <- clean %>%
  CoordinateCleaner::cc_zero(buffer = 0.5, verbose=FALSE)

if(nrow(clean)==0) stop('No more records available in the dataset')

#-----------------------------------------------------------
##= 8. Clean records with equal latitude and longitude
#      coordinates, either exact or absolute
#-----------------------------------------------------------
clean <- clean %>%
  CoordinateCleaner::cc_equ(test = "absolute", verbose=FALSE)

if(nrow(clean)==0) stop('No more records available in the dataset')

#-----------------------------------------------------------
##= 9. Clean records outside reported country
#-----------------------------------------------------------
# clean <- clean %>%
#
#   CoordinateCleaner::cc_coun(iso3='countryCode', ref=NULL, verbose=FALSE)
#
# if(nrow(clean)==0) stop('No more records available in the dataset')

#-----------------------------------------------------------
##= 10. Clean records with country and province centroids
#-----------------------------------------------------------
clean <- clean %>%
  CoordinateCleaner::cc_cen(buffer = 10000,
                            geod=TRUE,
                            test='both',
                            species='species',
                            ref=NULL,
                            verify=FALSE,
                            verbose=FALSE)

if(nrow(clean)==0) stop('No more records available in the dataset')

#-----------------------------------------------------------
##= 11. Clean records in country capitals
#-----------------------------------------------------------
clean <- clean %>%
  CoordinateCleaner::cc_cap(buffer = 10000,
                            geod=TRUE,
                            species='species',
                            ref=NULL,
                            verify=FALSE,
                            verbose=FALSE)

if(nrow(clean)==0) stop('No more records available in the dataset')

#-----------------------------------------------------------
##= 12. Clean records with institutional coordinates
#-----------------------------------------------------------
```

```
clean <- clean %>%
  CoordinateCleaner::cc_inst(buffer = 10000,
                             geod=TRUE,
                             species='species',
                             verify=FALSE,
                             verify_mltpl = 10,
                             verbose=FALSE)

if(nrow(clean)==0) stop('No more records available in the dataset')

#-------------------------------------------------------------
##= 13. Clean records with GBIF HQ coordinates
#-------------------------------------------------------------
clean <- clean %>%
  CoordinateCleaner::cc_gbif(buffer = 1000,
                             geod=TRUE,
                             species='species',
                             verify=FALSE,
                             verbose=FALSE)

if(nrow(clean)==0) stop('No more records available in the dataset')

#=============================================================
##
##=  End CoordinateCleaner cleaning process
##
#=============================================================
# Rename lon/lat fields
clean <- clean %>%
  dplyr::rename(decimalLongitude = 'decimallongitude',
                decimalLatitude = 'decimallatitude')
```

References

Govaerts, R., E. Nic Lughadha, N. Black, R. Turner, and A. Paton. 2021. "The World Checklist of Vascular Plants, a Continuously Updated Resource for Exploring Global Plant Diversity." Journal Article. *Sci Data* 8 (1): 215. https://doi.org/10.1038/s41597-021-00997-6.

Ondo, Ian. 2024. *TDWG: Cleaning Occurrence Records Using the World Geographical Scheme for Recording Plant Distributions (WGSRPD)*. https://github.com/IanOndo/TDWG.

Wickham, Hadley, Romain François, Lionel Henry, Kirill Müller, and Davis Vaughan. 2023. *Dplyr: A Grammar of Data Manipulation*. https://CRAN.R-project.org/package=dplyr.

Zizka, Alexander, Daniele Silvestro, Tobias Andermann, Josue Azevedo, Camila Duarte Ritter, Daniel Edler, Harith Farooq, et al. 2019. "CoordinateCleaner: Standardized Cleaning of Occurrence Records from Biological Collection Databases." *Methods in Ecology and Evolution*, no. 10: –7. https://doi.org/10.1111/2041-210X.13152.