The functionality of RCompadre and Rage

Pol Capdevila, John Jackson & Owen Jones

2021-04-12

## Introduction

The aim of this vignette is to give users a walk through of the functionality of the Rcompadre and Rage R packages.

The Rcompadre package allows users to access and manipulate data from the COMPADRE and COMADRE Plant and Animal Matrix Databases (Salguero-Gomez et al. 2015, 2016). These two databases contain matrix population models (MPMs) and associated metadata obtained from the published and “grey” literature.

The Rage package provides tools for conducting aging- and life history-focussed analysis of MPMs. Rage includes a wide variety of functions the calculation of life history traits (e.g. longevity, generation time), vital rates (e.g. growth, reproduction) and to run perturbation analyses (e.g. vital rate elasticities). Furthermore, the package includes functions to manipulate and transform MPMs and life tables.

## Research question (start of manuscript section)

In the following example we illustrate the use of Rcompadre to carry out typical data download and preparation tasks for an analysis relevant to comparative population dynamics research and implemented using Rage.

Let’s imagine that we are interested in exploring variation in mammal longevity across different ecoregions. The demographic information contained in MPMs can be used to estimate longevity (see SI in Jones et al, 2014) and we can therefore leverage the large taxonomic and geographic coverage of COMADRE to make interesting comparative inferences.

# [RCOMPADRE SECTION BEGINS]

The first task is to install the latest version of Rcompadre. RCompadre is developed on the GitHub platform and can be installed using remotes::install\_github.

install.packages("remotes")  
remotes::install\_github("jonesor/Rcompadre")

After ensuring that the package is installed we load it, along with other useful packages.

library(Rcompadre)  
library(tidyverse)

After that, we can download the latest version of the COMPADRE or COMADRE databases from the website using cdb\_fetch. We could also load local files with the same function (x <- cdb\_fetch("PATH/TO/FILE/COMADRE\_v.4.21.1.0.RData")). After loading the data, a message is displayed stating the database version number, release date, and links to the user agreement and citation information.

comadre <- cdb\_fetch("comadre")  
#> This is COMADRE version 4.21.1.0 (release date Jan\_25\_2021)  
#> See user agreement at https://compadre-db.org/Help/UserAgreement  
#> See how to cite at https://compadre-db.org/Help/HowToCite

### Data management

The COMADRE database includes metadata associated with each matrix including taxonomic information, geolocation, and details of the source publication (see the User Guide at [www.compadre-db.org](http://www.compadre-db.org) or Salguero-Gomez et al. 2015, 2016 for details). We can get an idea of the richness of the metadata with the names function.

names(comadre)  
#> [1] "mat" "MatrixID"   
#> [3] "SpeciesAuthor" "SpeciesAccepted"   
#> [5] "CommonName" "Kingdom"   
#> [7] "Phylum" "Class"   
#> [9] "Order" "Family"   
#> [11] "Genus" "Species"   
#> [13] "Infraspecies" "InfraspeciesType"   
#> [15] "OrganismType" "DicotMonoc"   
#> [17] "AngioGymno" "Authors"   
#> [19] "Journal" "SourceType"   
#> [21] "OtherType" "YearPublication"   
#> [23] "DOI\_ISBN" "AdditionalSource"   
#> [25] "StudyDuration" "StudyStart"   
#> [27] "StudyEnd" "ProjectionInterval"   
#> [29] "MatrixCriteriaSize" "MatrixCriteriaOntogeny"  
#> [31] "MatrixCriteriaAge" "MatrixPopulation"   
#> [33] "NumberPopulations" "Lat"   
#> [35] "Lon" "Altitude"   
#> [37] "Country" "Continent"   
#> [39] "Ecoregion" "StudiedSex"   
#> [41] "MatrixComposite" "MatrixSeasonal"   
#> [43] "MatrixTreatment" "MatrixCaptivity"   
#> [45] "MatrixStartYear" "MatrixStartSeason"   
#> [47] "MatrixStartMonth" "MatrixEndYear"   
#> [49] "MatrixEndSeason" "MatrixEndMonth"   
#> [51] "CensusType" "MatrixSplit"   
#> [53] "MatrixFec" "Observations"   
#> [55] "MatrixDimension" "SurvivalIssue"   
#> [57] "\_Database" "\_PopulationStatus"   
#> [59] "\_PublicationStatus"

The MPMs are contained in a list column called mat where each element contains a list of the four matrices: **A** and the submatrices **U**, **F** and **C** (see above). The list also includes information on matrix stage definitions. All other columns of the COMADRE database object are ordinary vectors.

We first explore taxonomic groups available in the dataset.

table(comadre$Class)  
#>   
#> Actinopterygii Adenophorea   
#> 196 4   
#> Amphibia Anthozoa   
#> 24 102   
#> Arachnida Ascidiacea   
#> 5 18   
#> Aves Bivalvia   
#> 409 30   
#> Branchiopoda Cephalaspidomorphi   
#> 7 9   
#> Clitellata Demospongiae   
#> 3 13   
#> Diplopoda Echinoidea   
#> 1 11   
#> Elasmobranchii Gastropoda   
#> 4 20   
#> Gymnolaemata Insecta   
#> 10 73   
#> Malacostraca Mammalia   
#> 30 2181   
#> Maxillopoda Polychaeta   
#> 11 19   
#> Reptilia Revtraviricetes   
#> 168 2   
#> Scyphozoa Secernentea   
#> 2 4   
#> Spirochaetes Thaliacea   
#> 9 1

We see that COMADRE contains information for many groups. In this case, we are only interested in mammals.Rcompadre allows us to subset the database easily using the function filter from dplyr as if our CompadreDB object was an ordinary data frame.

mammals <- comadre %>%  
 filter(Class == "Mammalia")  
  
table(mammals$Order)  
#>   
#> Artiodactyla Carnivora Cetacea   
#> 137 223 58   
#> Chiroptera Cingulata Dasyuromorphia   
#> 1 3 3   
#> Didelphimorphia Diprotodontia Lagomorpha   
#> 1 28 10   
#> Perissodactyla Primates Proboscidea   
#> 1 1557 3   
#> Rodentia Sirenia   
#> 153 3

### Data checking

Although COMPADRE and COMADRE data undergo rigorous checks before being made available, that doesn’t necessarily mean that all data will meet the exact requirements of any particular analysis. Rcompadre includes several functions for checking the data.

For example, for a minority of studies, it is not possible to decompose **A** matrices into the **U**, **F** and **C** submatrices. This is potentially important because the calculation of some life history traits requires one or more of these submatrices. For instance, in our example analysis focussing on longevity we require a **U** submatrix with no missing values.

To check, and filter out un-usable matrices, we use the function cdb\_flag and, in this case, specify checks = c("check\_NA\_U","check\_zero\_U") (see ?cdb\_flag for all available data checks). This specifies that we want to check whether the studies contain **U** matrices (check\_NA\_U) and whether these **U** matrices are all zeros (check\_zero\_U).

mammals <- cdb\_flag(mammals, checks = c("check\_NA\_U", "check\_zero\_U"))

The cdb\_flag function returns a version of the data with additional logical columns for each of the checks, which we can summarise using table.

table(mammals$check\_NA\_U)  
#>   
#> FALSE TRUE   
#> 2168 13  
table(mammals$check\_zero\_U)  
#>   
#> FALSE TRUE   
#> 2172 9

Now we can filter our data to include only data with suitable **U** matrices.

mammals <- mammals %>%  
 filter(check\_NA\_U == FALSE, check\_zero\_U == FALSE)

Finally, most COM(P)ADRE data includes geolocation information. In addition to Country and Continent, precise latitude and longitude (Lat and Lon) make it easy to visualise the spatial patterns in our data set using tools in the ggplot2 and maps packages. We illustrate this in the next section.

Now that we are sure that our data set contains the right information, we are ready to calculate longevity using functions in the Rage package.

# [RCOMPADRE SECTION ENDS]

# [RAGE SECTION BEGINS]

In the following section we demonstrate the use of Rage, focussing on the longevity analysis introduced above. We first install the package from GitHub.

install.packages("remotes")  
remotes::install\_github("jonesor/Rage")

Then load it.

library(Rage)

Our goal is to calculate the longevity of mammals from across the world using the dataset we generated above. We can do this using the longevity function, which simulates the ages at death of a synthetic cohort of individuals following the demographic pattern described by the MPM. The function documentation (?longevity) tells us that it requires the **U** matrix, which we can obtain using the matU function from Rcompadre.

The function also requires us to define which stage we consider to be the beginning of life. This is fairly clear for most mammals, but may be contentious in some groups, such as plants with a seed bank. Although it may be desirable to do this manually in some cases, the Rcompadre function mpm\_first\_active automates the task by returning an integer index for the first active stage class, as noted by the original study author.

The longevity function also the argument lx\_crit, which defines the critical threshold of survivorship that defines longevity. The default is 0.01, meaning that longevity is defined as the age at which only 1% of the synthetic cohort remain alive. Finally, the function requires us to set a maximum age to consider (xmax, which defaults to 1000).

Thus, our first steps in this analysis are to first extract the **U** matrices with matU, then add using mutate the first stage with mpm\_first\_active before applying the longevity function over our dataset using mapply.

mammals <- mammals %>%  
 mutate(  
 matU = matU(.),  
 start\_life = mpm\_first\_active(.)  
 ) %>%  
 mutate(longevity = mapply(longevity, matU, start\_life))

Now that we have the longevity data for all of our mammals we can explore its variation across the globe and across the mammal orders.

First, we will examine how longevity varies spatially (Fig. 2).

globalDist\_longevity <- ggplot(mammals, aes(Lon, Lat)) +  
 borders(database = "world", fill = "grey80", col = NA) +  
 geom\_point(aes(colour = log10(longevity)), alpha = 0.4) +  
 scale\_colour\_viridis\_c(option = "C",name = expression(log[10]\*"(longevity)")) +  
 labs(x = NULL, y = NULL) +  
 theme\_minimal()   
  
globalDist\_longevity

Next we examine the association between longevity and latitude and ecoregion. A visual inspection of the data shows an apparent decline in longevity at intermediate latitudes (Fig. 3A), and variation among ecoregions (Fig. 3B). These patterns could well be driven by data biases, but would make an interesting topic for future exploration.

longevity\_lat <- ggplot(mammals, aes(x = abs(Lat), y = log10(longevity))) +  
 geom\_point(alpha = 0.4) +  
 labs(x = "Absolute Latitude", y = expression(log[10]\*"(longevity)")) +  
 theme\_minimal() +  
 geom\_smooth()  
  
longevity\_lat  
  
longevity\_ecoregion <- ggplot(mammals, aes(  
 x = Ecoregion, y = log10(longevity),  
 fill = Ecoregion  
)) +  
 geom\_boxplot(show.legend = FALSE) +  
 geom\_jitter(alpha = 0.5, width = 0.1, show.legend = FALSE) +  
 labs(x = "Ecoregion", y = expression(log[10]\*"(longevity)")) +  
 coord\_flip() +  
 theme\_minimal()  
  
longevity\_ecoregion

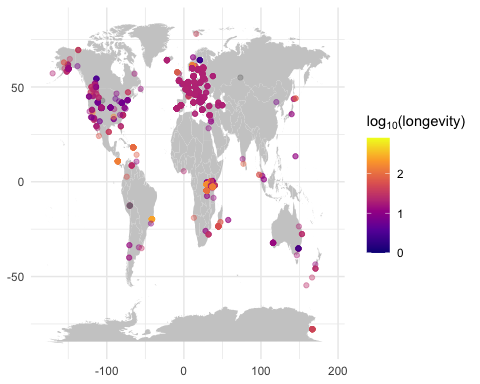
With the broad range of tools at our fingertips in Rcompadre and Rage, we can thus begin to address such detailed comparative questions with increased ease.

# Figure captions

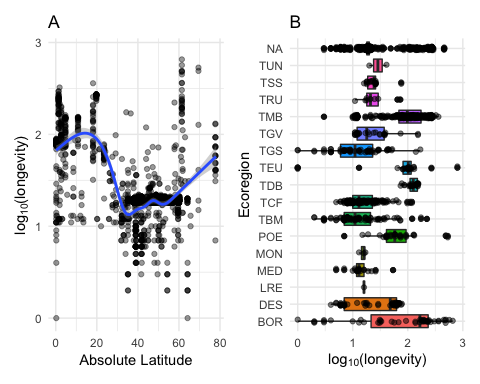
*Figure 2. The spatial extent of data, colour-coded by estimated longevity. Note that 269 of matrices in our set (~12%) do not have associated spatial information.*

*Figure 3. The relationship between estimated longevity and (A) absolute latitude and (B) ecoregion. The line in 3A represents a smoothed spline fitted using ggplot’s default settings.*

# Figures



#> `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'



#> `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'