Large datasets

Anonymous

2025-03-06

## Initialisation

# clearing the environment  
rm(list = ls())  
  
# loading packages  
library(Rcompadre)  
library(tidyverse)  
library(here)  
library(phytools)  
library(taxize)  
library(Rage)  
library(popbio)  
library(svglite)  
library(ggbeeswarm)  
library(FSA)  
library(dunn.test)  
  
# loading self made functions (file contents shown at the end of document)  
source(here("Data", "Functions.R"))

# Introduction

In this document we run an analysis of biological patterns within the Aves class using data from various online databases. We collect demographic data from comadre, species’ status from IUCN RedList and phylogenetic data from the Open Tree of Life project. Focusing on their growth rate, generation time, status and the relationships of these variables to eachother, and to their phylogeny.

This Document also functions as a template for this analysis to be used for any species, by changing the variable “class” to the name of the class (as used in the comadre database) you want to follow, and the code will run the same analysis on that data and change figures and figure titles accordingly, and the rest of the code is created so you can easily change the values being analysed. Some extra work may be needed to account for inconsistencies between databases.

On the anonomised github repository you can find all of the files used and a blank template, “Large datasets template.rmd”, containing the code without specific analysis and write up for usage in your own analysis (The 2025).

GitHub Link: <https://github.com/The-Nedstar/LargeDatasets>

## Enter your species of interest here:

class <- "Aves"

# Comadre

Comadre is a online database containing demographic data for many animals collected from almost a thousand studies hosted at the Max Planck institute (Salguero-Gómez et al. 2016). It includes: Their species identification and taxonomy; geographical data; population parameters such as generation time and growth rate and information on the studies the data is taken from. Population parameters are stored in matrix population models (MPMs) all accessible through R studio as well as various other programs. This database is particularly useful for ecologists, evolutionary biologists and conservationists

## Acessing Data

## creating class variable for titling of figures  
ClassName <- paste(class, "class")  
  
## fetching data from the comadre database  
Comadre <- cdb\_fetch("comadre")  
  
## Subsetting data  
# subsetting to wild, unmanipulated individuals from chosen class  
ComSub <- subset(Comadre,  
 Class == class &  
 MatrixTreatment == "Unmanipulated" &  
 MatrixCaptivity == "W")  
# flagging and subsetting to only data without NAs and which is ergodic  
ComFlag <- cdb\_flag(ComSub)  
ComSubFlag <- subset(ComFlag,  
 check\_NA\_A == FALSE &  
 check\_ergodic == TRUE)

## Distribution of growth rate and generation time (Q1)

We first Take a look at the distribution of both growth rate and generation time within our class using the data collected from comadre

## extraction of growth rate and generation time from the MPMs  
ComSubFlag$lambda <- unlist(lapply(matA(ComSubFlag),   
 popbio::lambda))  
ComSubFlag$generation\_time <- unlist(lapply(matA(ComSubFlag),  
 popbio::generation.time))  
  
## removing any observations with infinite or NA values in either variable  
GTvL <- as\_data\_frame(ComSubFlag) %>%  
 select(lambda, generation\_time) %>%   
 mutate(across(everything(), ~ na\_if(., Inf))) %>%   
 mutate(across(everything(), ~ na\_if(., -Inf))) %>%  
 drop\_na()

## using a self made function to save an svg of a histogram  
histogram(GTvL, GTvL$lambda,  
 "Population Growth Rate(λ)",   
 paste("Histogram of Growth Rate Within the",  
 ClassName), 0.15, TRUE, "LHist.svg")

## png   
## 2

## Printing summary statistics for growth rate  
cat("--- Summary statistics for Growth rate ---\n")

## --- Summary statistics for Growth rate ---

summary(GTvL$lambda)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.3261 0.8586 0.9949 0.9944 1.1652 1.7949

Figure 1: Histogram showing the distribution of the population growth rate within the Aves Clade.

It appears the populationg growth grate is roughly normally distributed around a mean of ~0.9944 so the majority of species are stable with a roughly even distribution between increasing and decreasing.

## using a self made function to save an svg of a histogram  
histogram(GTvL, GTvL$generation\_time,  
 "Generation Time (years)",   
 paste("Histogram of Generation Time Within the",  
 ClassName), 1, FALSE, "GTHist.svg")

## png   
## 2

## Printing summary statistics for generation time  
cat("--- Summary statistics for generation time ---\n")

## --- Summary statistics for generation time ---

summary(GTvL$generation\_time)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.493 4.487 5.944 7.494 9.576 53.210

Figure 2: Histogram showing the distribution of the population growth rate within the Aves Clade.

The distribution fo generation time appears to be assymetrically distributed with a right skew with a mean of ~7.494 but a median of ~5.944. So most of the species are shorter living (<10 years) with a fewer that surpass this.

## Relationship between Growth rate and generation time (Q2)

For the next part of the analysis, we compare growth rate and generation time to eachother and attempt to fit a model to explore potential correlations. We would expect there to be a negative correlation between the variables.

H0: There is no significant relationship between growth rate and generation time

H1: There is a significant relationship between growth rate and generation time

## creation of a explorative scatterplot comparing growth rate and generation time using a self made function  
scatterplot(GTvL, GTvL$generation\_time,   
 "Generation Time (years)",   
 GTvL$lambda, "Growth Rate (λ)",   
 paste("Generation time Vs Growth Rate in the",  
 ClassName), FALSE, "GTvL.svg")

## png   
## 2

Figure 3: a scatterplot comparing growth rate to generation time in the Aves class.

From first inspection there is potentially a positive relationship, but it is unlikely to be linear.

### **Linear model**

## creation of a version of the dataset without the outlier  
CGTvL <- subset(GTvL, generation\_time < 50)  
  
## creation of a linear model and testing violations of assumptions using a diagnostic plot  
# with all the data  
GTvLMod <- lm(lambda ~ generation\_time, GTvL)  
diagnostic\_plots("GTvLMod DP.svg", GTvLMod)

## png   
## 2

# with the outlier removed  
CGTvLMod <- lm(lambda ~ generation\_time, CGTvL)  
diagnostic\_plots("CGTvLMod DP.svg", CGTvLMod)

## png   
## 2

**Log transformed**

## creation of a log transformed linear model and testing violations of assumptions using a diagnostic plot  
# with all data  
LGTvLMod <- lm(log(lambda) ~ log(generation\_time), GTvL)  
diagnostic\_plots("LGTvLMod DP.svg", LGTvLMod)

## png   
## 2

# with the outlier removed  
LCGTvLMod <- lm(log(lambda) ~ log(generation\_time), CGTvL)  
diagnostic\_plots("LCGTvLMod DP.svg", LCGTvLMod)

## png   
## 2

**Polynomial**

## creation of a polynomial model and testing violations of assumptions using a diagnostic plot  
# with all the data  
PGTvLMod <- lm(lambda ~ generation\_time + I(generation\_time^2) + I(generation\_time^3), GTvL)  
diagnostic\_plots("PGTvLMod DP.svg", PGTvLMod)

## png   
## 2

# with the outlier removed  
PCGTvLMod <- lm(lambda ~ generation\_time + I(generation\_time^2) + I(generation\_time^3), CGTvL)  
diagnostic\_plots("PCGTvLMod DP.svg", PCGTvLMod)

## png   
## 2

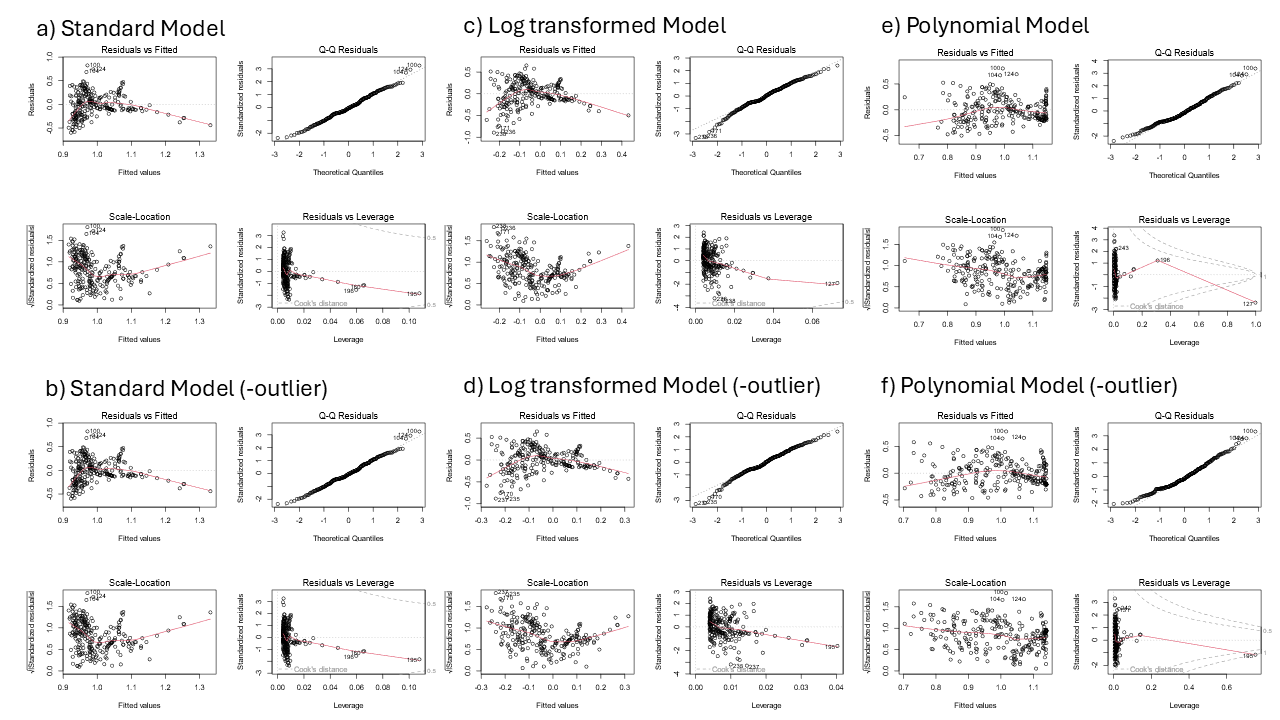


Figure 4: diagnostic plots for all attempted models

It appears that the data violates the assumptions of a generalized linear model under all attempted variations. For all models, both with and without the outlier removed the ‘residuals vs fitted’ and ‘scale vs location’ plots both show deviations from a flat horizontal relationship, indicating that the data hetereoscedastic and the relationship is non-linear.  
The polynomial has the least violation of these assumptions, however the outliers have a greater impact on the data, even when the most severe outlier is removed.

### Spearman’s rank correlation coefficient

Due to the data violating the assumptions of a linear model approach, we decided to move forward with a non parametric approach using a Spearman’s rank correlation coefficient. Our data does match the assumptions as it is monotonic with independent observations and as a ranked method it is robust to outliers.

cat("--- Outcome of the Spearman's rank correlation test ---\n")

## --- Outcome of the Spearman's rank correlation test ---

cor.test(GTvL$lambda, GTvL$generation\_time, method = "spearman")

##   
## Spearman's rank correlation rho  
##   
## data: GTvL$lambda and GTvL$generation\_time  
## S = 1720894, p-value = 8.774e-09  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.3547759

The results of this test show that there is sufficient data to reject our null hypotheses. There appears to be a relatively weak positive relationship between the data. Considering the diagnostic plots from earlier this relationship is likely non-linear.

# IUCN RedList

The International Union for the Conservation of Nature (IUCN) is a collection of organisations that work together to aid in conservation efforts (“The IUCN Red List of Threatened Species,” n.d.). the RedList is a database containing many species and describing their conservation status. Using this database, research and conservation efforts can be effectively distributed between the species that most need it.

## Acessing data (Q3)

Due to issues with accessing the online database, instead we have imported the data relevant to the comadre database from a csv (also accessible on the github repository, under the folder “Data” (The 2025)).

## loading the IUCNRedList data  
IUCNData <- read.csv(here("Data", "IUCN\_comadre\_compadre.csv"))  
# combining with our comadre dataset  
ComIUCN <- ComSubFlag %>%  
 left\_join(x = ., y = IUCNData, by = "SpeciesAccepted") %>%   
 mutate(IUCNstatus = replace\_na(IUCNstatus, "NE"))  
# producing a new column converting the two letter status codes into full words  
ComIUCN <- ComIUCN %>%   
 mutate(IUCNstatLong = case\_when(  
 IUCNstatus == "EN" ~ "Endangered",  
 IUCNstatus == "VU" ~ "Vulnerable",  
 IUCNstatus == "NT" ~ "Near Threatened",  
 IUCNstatus == "LC" ~ "Least Concern",  
 IUCNstatus == "DD" ~ "Data Deficient",  
 IUCNstatus == "CR" ~ "Critically Endangered",  
 IUCNstatus == "EW" ~ "Extinct in the wild",  
 IUCNstatus == "EX" ~ "Extinct",  
 IUCNstatus == "NE" ~ "Not Evaluated"),  
 IUCNstatLong = factor(IUCNstatLong,   
 levels = c("Extinct in the wild",  
 "Extinct","Endangered",  
 "Vulnerable",  
 "Near Threatened",  
 "Least Concern",  
 "Data Deficient",  
 "Not Evaluated")))  
## converting into a data frame  
ComIUCN <- as\_data\_frame(ComIUCN)

## analysis

IUCNVGTvL <- subset(as\_data\_frame(ComIUCN),generation\_time < 50) %>%  
 select(lambda, generation\_time, IUCNstatLong) %>%   
 mutate(across(c(lambda, generation\_time),   
 ~ na\_if(., Inf))) %>%   
 mutate(across(c(lambda, generation\_time),   
 ~ na\_if(., -Inf))) %>%  
 drop\_na()

**generation time**

## using a self made function to produce a boxplot comparing the varaibles  
boxplot(IUCNVGTvL, IUCNVGTvL$IUCNstatLong, "IUCN status", IUCNVGTvL$generation\_time, "Generation Time (years)", paste("IUCN status vs generation time in the", ClassName), "IUCNvGT.svg")

## png   
## 2

## running an ANOVA analysis  
cat("--- ANOVA analysis of differences in generation time between conservation statuses ---/n")

## --- ANOVA analysis of differences in generation time between conservation statuses ---/n

ANOVAGT <- aov(generation\_time ~ IUCNstatLong, data = IUCNVGTvL)  
summary(ANOVAGT)

## Df Sum Sq Mean Sq F value Pr(>F)   
## IUCNstatLong 4 547 136.76 9.375 4.61e-07 \*\*\*  
## Residuals 243 3545 14.59   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

cat("--- Dunn's test analysis of differences in generation time between conservation statuses ---/n")

## --- Dunn's test analysis of differences in generation time between conservation statuses ---/n

dunnTest(generation\_time ~ IUCNstatLong, data = IUCNVGTvL)

## Comparison Z P.unadj P.adj  
## 1 Endangered - Least Concern -1.0794234 2.803990e-01 8.411971e-01  
## 2 Endangered - Near Threatened -2.2512459 2.436997e-02 1.462198e-01  
## 3 Least Concern - Near Threatened -2.0339491 4.195672e-02 2.097836e-01  
## 4 Endangered - Not Evaluated -1.4535475 1.460718e-01 5.842872e-01  
## 5 Least Concern - Not Evaluated -0.8820864 3.777301e-01 7.554602e-01  
## 6 Near Threatened - Not Evaluated 0.6528047 5.138822e-01 5.138822e-01  
## 7 Endangered - Vulnerable -4.7704702 1.837964e-06 1.654168e-05  
## 8 Least Concern - Vulnerable -5.2591784 1.447005e-07 1.447005e-06  
## 9 Near Threatened - Vulnerable -3.0463839 2.316119e-03 1.621283e-02  
## 10 Not Evaluated - Vulnerable -3.3630788 7.707835e-04 6.166268e-03

Figure 5: Boxplot showing the generation time of species in each conservation status category

There is a significant difference between vulnerable and the rest of the categories, but none between the rest of the categories. This suggests that species with longer generations are more likely to be categorised as vulnerable. However, this relationship is not found with endangered species, making the overall effect of generation time on vulnerability confusing.

**Growth Rate**

## using a self made function to produce a boxplot comparing the varaibles  
boxplot(IUCNVGTvL, IUCNVGTvL$IUCNstatLong, "IUCN status", IUCNVGTvL$lambda, "Growth Rate (λ)", paste("IUCN status vs growth rate in the", ClassName), "IUCNvL.svg" )

## png   
## 2

## running an ANOVA analysis  
cat("--- ANOVA analysis of differences in growth rate between conservation statuses ---/n")

## --- ANOVA analysis of differences in growth rate between conservation statuses ---/n

ANOVALambda <- aov(lambda ~ IUCNstatLong, data = IUCNVGTvL)  
summary(ANOVALambda)

## Df Sum Sq Mean Sq F value Pr(>F)   
## IUCNstatLong 4 2.658 0.6644 11.29 2.03e-08 \*\*\*  
## Residuals 243 14.299 0.0588   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

cat("--- Dunn's test analysis of differences in growth rate between conservation statuses ---/n")

## --- Dunn's test analysis of differences in growth rate between conservation statuses ---/n

dunnTest(lambda ~ IUCNstatLong, data = IUCNVGTvL)

## Comparison Z P.unadj P.adj  
## 1 Endangered - Least Concern -3.6302589 2.831371e-04 1.981960e-03  
## 2 Endangered - Near Threatened -2.6289141 8.565800e-03 3.426320e-02  
## 3 Least Concern - Near Threatened 0.4884560 6.252269e-01 6.252269e-01  
## 4 Endangered - Not Evaluated -3.1939794 1.403261e-03 8.419568e-03  
## 5 Least Concern - Not Evaluated -0.6799008 4.965673e-01 9.931346e-01  
## 6 Near Threatened - Not Evaluated -0.8834502 3.769931e-01 1.000000e+00  
## 7 Endangered - Vulnerable -6.1665962 6.977565e-10 6.977565e-09  
## 8 Least Concern - Vulnerable -4.6878401 2.761037e-06 2.484933e-05  
## 9 Near Threatened - Vulnerable -4.2084282 2.571532e-05 2.057226e-04  
## 10 Not Evaluated - Vulnerable -3.0740389 2.111819e-03 1.055909e-02

Figure 6: Boxplot showing the population growth rate of species in each conservation status category

In the case of population growth rate, Near threatened and least concerned show no significant different from each other, or from species which have not been evaluated. Endangered species show a significantly decreased growth rate, as expected, whereas vulnerable species show an increased growth rate. This is potentially due to implementation of succsessful protective strategies increasing the growth rate of species which are described as vulnerable.

# open tree of life (Q4)

The open tree of life is a collaborative effort to create a dynamic phylogeny of the entire tree of life. It is openly accessible through an API, but in this case it is being accessed through a .tre file of all of the data relevant to the comadre dataset from the repository (Jones 2024) (also accessible on the github repository, under the folder “Data” (The 2025)).

## Creating a new dataframe with each species only present once  
ComSingle <- ComIUCN[which(  
 duplicated(ComIUCN$SpeciesAccepted) == FALSE),]  
  
## wrangling data using the open tree of life online API  
# Creating a list of names from our dataframe that matches an entry in the open tree of life  
ResNames <- rotl::tnrs\_match\_names(names = ComSingle$SpeciesAccepted)  
# adding the open tree of life id to our dataframe  
ComSingle$ott\_id <- ResNames$ott\_id  
# removing any values where there is no id  
ComSingle <- ComSingle[-which(is.na(ComSingle$ott\_id)),]  
# updating our list of mached names using only the ones with an id  
ResNames <- rotl::tnrs\_match\_names(names = ComSingle$SpeciesAccepted)  
# putting this names into a new column in our dataframe  
ComSingle$OTL\_unique\_name <- ResNames$unique\_name

## Loading in the tree data  
Tree <- read.tree(here("Data", "COMPADRE-COMADRE\_Phylo\_June\_16\_2019.tre"))  
  
## remove NA and infinite values from our dataframe   
ComSingle <- ComSingle %>%   
 mutate(across(c(lambda, generation\_time), ~ na\_if(., Inf))) %>%   
 mutate(across(c(lambda, generation\_time), ~ na\_if(., -Inf))) %>%  
 drop\_na(lambda, generation\_time)  
   
## Using the Open tree of life to create our own tree  
# Shortening the tip labels  
Tree$tip.label <- gsub("\_", " ", Tree$tip.label)  
# Removing species which aren't in our dataframe  
PrunedTree <- drop.tip(Tree, setdiff(Tree$tip.label, ComSingle$OTL\_unique\_name))  
# alterning the order of our dataframe to map the tree  
ComSingle <- ComSingle[match(PrunedTree$tip.label,ComSingle$OTL\_unique\_name),]  
# setting the species names as the row names  
row.names(ComSingle) <- ComSingle$OTL\_unique\_name  
  
## Plotting our Pruned tree  
svglite(here("Figures", "PrunedTree.svg"), width = 8, height = 8, scaling = 0.9)   
plot(PrunedTree, fsize = c(0.7,0.8))   
dev.off()

## png   
## 2

## Plotting phylogeny with population parameters

### **growth rate**

## Creating a list containing the Log of the growth rate attached to the species name  
LogL <- log(setNames(ComSingle$lambda, row.names(ComSingle)))  
  
## creating a phylogenetic tree coloured based on the population growth rate  
ContMapL <- contMap(PrunedTree, LogL,plot=FALSE,res=1000,  
 method="anc.ML")   
ContMapL <- setMap(ContMapL, c("white","#FFFFB2","#FECC5C",  
 "#FD8D3C","#E31A1C"))  
  
## Plotting the phylogenetic tree  
svglite(here("Figures", "ContMapLambda.svg"), width = 8, height = 8, scaling = 1.2)   
plot(ContMapL, fsize = c(0.7,0.8), leg.txt =   
 "log(population growth rate)")   
dev.off()

## png   
## 2

It doesn’t appear that the population growth rate does is not affected by phylogenetic inertia. There is no clear clustering or relation between individuals with either particularly high or low growth rates, with growth rates often being very different from closely related individuals.

### **Generation time**

## Creating a list containing the Log of the generation time attached to the species name  
LogGT <- log(setNames(ComSingle$generation\_time, row.names(ComSingle)))  
  
## creating a phylogenetic tree coloured based on the generation time  
ContMapGT <- contMap(PrunedTree, LogGT,plot=FALSE,res=200,   
 method="anc.ML")  
ContMapGT <- setMap(ContMapGT, c("white","light blue","blue",  
 "violet","purple"))  
  
## Plotting the phylogenetic tree  
svglite(here("Figures", "ContMapGT.svg"),   
 width = 8, height = 8, scaling = 1.2)  
plot(ContMapGT, fsize=c(0.7,0.8), leg.txt =   
 "log(generation time) (years)")  
dev.off()

## png   
## 2

Generation time on the other hand shows a substantial effect of phylogenetic inertia. It is generally more closely matched between closely related individuals than those which are distantly related and shows strong clustering.

## population performance (Q5)

This final phylogeny shows Whether each species is increasing (population growth rate > 1) or decreasing (population growth rate <1) as well as their conservation status.

using code from (“Exercise 15: Plotting Methods for Phylogenies & Comparative Data in r,” n.d.)

## Creating a list of either "Increase" if growth rate is greater than one, or "Decrease" if it is less  
PopPerf <- as\_data\_frame(ComSingle$lambda) %>%   
 mutate(category = factor(  
 case\_when(value < 1 ~ "Decrease",  
 value > 1 ~ "Increase")))  
  
## adding the conservation status to the tip labels  
# seperating creating a new column in our dataframe combining the status and the species name into one string  
ComSingle$TipName <- paste(ComSingle$IUCNstatus,  
 ComSingle$OTL\_unique\_name,   
 sep = " ")  
# replacing the tip labels with these new strings  
PrunedTree$tip.label <- ComSingle$TipName  
# replacing PopPerf with a list of factors as named by our new tip labels  
PopPerf <- as.factor(setNames(PopPerf$category,  
 ComSingle$TipName))  
  
## plotting our new tree  
svglite(here("Figures", "PerfTree.svg"), width = 8,  
 height = 8,  
 scaling = 0.9)  
# plotting a tree with dots coloured depending on whether the growth rate is greater than or less than 1  
dotTree(PrunedTree, PopPerf, colors = setNames(  
 c("#EA0000", "#1E88E5"),  
 c("Decrease","Increase")),  
 ftype="i", )  
dev.off()

## png   
## 2

# Appendix

## references

“Exercise 15: Plotting Methods for Phylogenies & Comparative Data in r.” n.d. <http://www.phytools.org/Cordoba2017/ex/15/Plotting-methods.html>.

Jones, Owen. 2024. *Jonesor/compadreDB*. <https://github.com/jonesor/compadreDB>.

Nichols, David. n.d. “Coloring for Colorblindness.” <http://www.davidmathlogic.com/colorblind/>.

Salguero-Gómez, Roberto, Owen R. Jones, C. Ruth Archer, Christoph Bein, Hendrik De Buhr, Claudia Farack, Fränce Gottschalk, et al. 2016. “COMADRE : A Global Data Base of Animal Demography.” Edited by Tim Coulson. *Journal of Animal Ecology* 85 (2): 371–84. <https://doi.org/10.1111/1365-2656.12482>.

“The IUCN Red List of Threatened Species.” n.d. <https://www.iucnredlist.org/en>.

The, Nedstar. 2025. “The-Nedstar/LargeDatasets.” <https://github.com/The-Nedstar/LargeDatasets>.

Whitlock, Michael C., and Dolph Schluter. 2020. *The Analysis of Biological Data*. 3rd ed. New York: Macmillan Learning.

## Contents of the Functions.R file

Can also be found on the anonymised github repository (The 2025).

###########################################  
# Functions for the Large Dataset assignment  
# author: anonymous  
###########################################  
  
### creation of a histogram  
histogram <- function(Data, Xaxis, Xtitle, Title, BW, Line, File){  
 ## defining the histogram  
 temp <- ggplot(Data, aes(x = Xaxis)) +  
 geom\_histogram(aes(fill = as.factor(floor(..x.. / BW) %% 2)), # alternative colours  
 binwidth = BW, boundary = 1, # ensuring it is split at the value 1  
 colour = "black", size = 0.4) +  
 (if (Line == TRUE) { # only carrying out if specified  
 geom\_vline(xintercept = 1, colour = "red",   
 size = 1.3, linetype = "21")}) + # creates a vertical line at the value 1  
 scale\_fill\_manual(values = c("#707070", "#555555")) + # specifying alternating colours  
 theme\_bw()+  
 theme(  
 legend.position = "none", # removing the legend  
 axis.title = element\_text(size = 14),  
 title = element\_text(size = 14)  
 ) +  
 xlab(Xtitle) +  
 ylab("Frequency") +  
 ggtitle(Title)  
 ## saving the graph as a .svg  
 svglite(here("Figures", File), width = 10,  
 height = 6,  
 scaling = 1.1)  
 print(temp)  
 dev.off()  
}  
  
### create a scatterplot  
scatterplot <- function(Data, Xaxis, Xtitle, Yaxis, Ytitle, Title, Line, File){  
 ## creating the scatterplot  
 temp <- ggplot(Data, aes(x = Xaxis, y = Yaxis)) +  
 geom\_point(colour = "darkblue")+  
 (if (Line == TRUE) { # Only running if specified  
 geom\_smooth(method = "lm", se = TRUE, linetype = "solid",   
 colour = "darkred")}) + # creating a line defined by a generalised linear model  
 theme\_bw()+  
 theme(  
 legend.position = "none", # removing the legend  
 axis.title = element\_text(size = 14),  
 title = element\_text(size = 14)  
 ) +  
 xlab(Xtitle) +  
 ylab(Ytitle) +  
 ggtitle(Title)  
 ## saving as a .svg  
 svglite(here("Figures", File), width = 10,  
 height = 6,  
 scaling = 1.1)  
 print(temp)  
 dev.off()  
}  
  
### Create Diagnostic plots  
diagnostic\_plots <- function(File, Model) {  
 ## saving plot as an SVG  
 svglite(here("Figures", File), width = 10,  
 height = 8,  
 scaling = 1.3)  
 ## creating a multi plot  
 # defining multi plot layout  
 par(mfrow=c(2,2))  
 # combining each diagnostic plot into one multi plot  
 DiaPlots <- (plot(Model,1) | plot(Model,2)) / (plot(Model,3) | plot(Model,5))  
 dev.off()  
}  
  
### create a boxplot  
boxplot <- function(Data, Xaxis, Xtitle, Yaxis, Ytitle, Title, File){  
 ## defining the boxplot  
 temp <- ggplot(Data, aes(x = Xaxis, y = Yaxis, fill = Xaxis)) +  
 geom\_boxplot() +  
 geom\_beeswarm(alpha = 0.6, size = 0.75, corral.width = 0.5) +  
 scale\_fill\_manual(values = c("#E89494", "#A4DFE4", "#8082E4",  
 "#F2D16F", "#EC89E4")) + # defining colours  
 theme\_bw() +  
 theme(  
 legend.position = "none", # removing the legend  
 axis.title = element\_text(size = 14),  
 title = element\_text(size = 14)  
 ) +  
 xlab(Xtitle) +  
 ylab(Ytitle) +  
 ggtitle(Title)  
 ## saving as a .svg  
 svglite(here("Figures", File), width = 10,  
 height = 6,  
 scaling = 1.3)  
 print(temp)  
 dev.off()  
}