BINF*6210—Bioinformatics Software Tools

Karl Cottenie

October 25, 2024

Ulli Bodnar—1351935

Assignment 2

Introduction

Phenotypic variations offer organisms the ability to survive in a catalogue of environments. Pigment patterns can elicit effective camouflage; a simple change in body mass—increasing overall volume, thus reducing the surface area to volume ratio—can allow an organism to survive in a harsh winter environment (McQueen et al., 2022). Bergmann's rule (Bergmann, C. 1847) explains that "within species and amongst closely related species of homeothermic animals a larger size is often achieved in colder climates than in warmer ones" (Salewski et al., 2017). As explained by Bodnar (2024), "Bergmann's rule can be observed within a combination of the Pantheria and BOLD System's databases among species containing the *Lepus* (hare) genus designation." To continue exploring this interesting topic, this project will build on previous work (Bodnar, 2024) and determine whether related species that are genetically close (within the Lepus genus) share similar body mass traits, i.e., within the Lepus genus, does Bergmann's rule follow evolutionary relationships or random chance?

Code Setup

```r
# Setup ------------------------------------------------------------------

# Import the libraries so that their functions can be used
# Just a heads up, I find R kinda unpredictible with the cache and holding libraries loaded so if this doesn't work by just the ones i've uncommented, please just uncomment all of these below and run it again. It will for sure work with all loaded, but I am pretty sure I only need the ones that are labelled 'in use'
# MIGHT NEED LIBRARIES:
# library(stats)
# library(sf)
# library(randomForest)
# library(viridis)
# library(seqinr)

# Libraries in use
library(tidyverse)
library(rentrez)
library(Biostrings)
library(fmsb)
library(ape)
library(dendextend)
library(phytools)
library(DECIPHER)
library(muscle)


# Notes:
# Write confirmation checks (that the NAs were removed, etc.)
# Write comments that state what the returned information is and even results and interpretation
# How is this helping me address my main question

# Set working directory to utilize data from datasets
setwd("/Users/ullibodnar/Documents/School/Guelph Masters/Bioinformatics Software Tools/Assignment 2/code")


```

```r

# Global variables ----------------------------------------------------------

#### CHECK ONLINE IF THESE ARE THE CORRECT CHOICES FOR WHAT I'M DOING ####

missingData <- 0.01
lengthVar <- 50
chosenModel <- "K80" # K2P
clusteringMethod <- "ML"



# Global functions ----------------------------------------------------------

# Use the species name from BOLD dataset to obtain corresponding mass (in grams) from Pantheria dataset
getAverageMass <- function (df, speciesName) {
  averageMass <- df |>
    filter(MSW05_Binomial == speciesName) |>
    pull("5-1_AdultBodyMass_g")

  # Substitute -999 masses
  ifelse(averageMass |>
           length() == 0
         , 0, averageMass)
}



```

```r
# Importing data ------------------------------------------------------------
# BOLD DB
# API CALL TO SHOW I KNOW HOW TO DO IT
# Lepus <- read_tsv("http://v3.boldsystems.org/index.php/API_Public/combined?taxon=Lepus&format=tsv")
# write_tsv(Lepus, "lepus_bold_data.txt")

# Import lepus data from BOLD database to have larger selection of COI-5P sequences
rawBoldLepus <- read_tsv(file = "../data/lepus_bold_data.txt")

# Import Pantheria DB for body mass in grams data
pantheriaData <- read_tsv(file = "../data/Pantheria.tsv")

# NCBI's nucleotide ---
# Determine possible database search locations
entrez_dbs()

# Sequence length range chosen becuase NCBI returned 671bp length for this sequence of interest when inspecting the database. Range to catch ones with missing data
# Query NCBI's nucleotide database to test the waters of possible data
lepusSearch <- entrez_search(db = "nucleotide", term = "Lepus[ORGN] AND COI AND 600:800[SLEN]", retmax = 100)

# Determine result count and change entrez_search to provide unique IDs for each possible returned result
lepusRetmax <- lepusSearch$count
lepusSearch <- entrez_search(db = "nucleotide", term = "Lepus[ORGN] AND COI AND 600:800[SLEN]", retmax = lepusRetmax)

# Fetch the data as fasta
lepusFetch <- entrez_fetch(db = "nucleotide", id = lepusSearch$ids, rettype = "fasta")

# Write the data to fasta file; commented out because the data have already been imported
# write(lepusFetch, "lepus_fetch.fasta", sep = "\n") --- Done on Oct. 18

# Import already created NCBI data file
nucleotideLepusStringSet <- readDNAStringSet("../data/lepus_fetch.fasta")

# Convert to a dataframe object for easy manipulation
nucleotideLepus <- data.frame(title = names(nucleotideLepusStringSet), nucleotides = paste(nucleotideLepusStringSet))

```

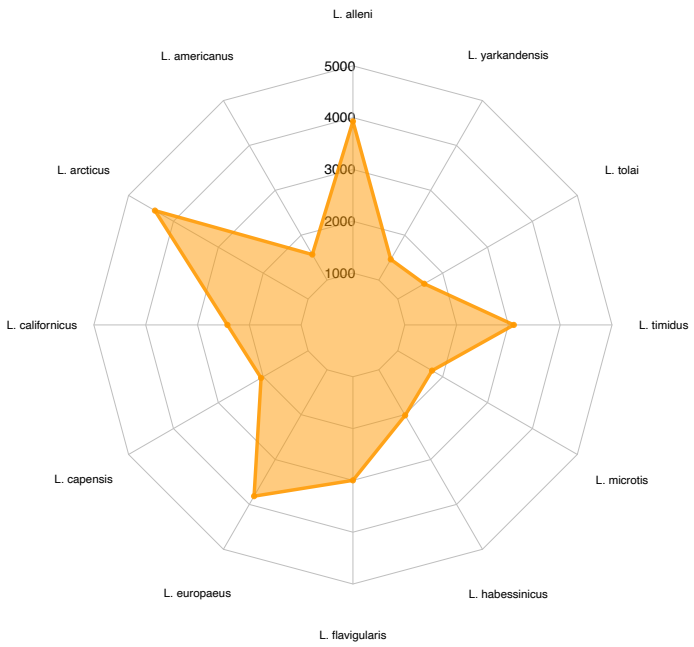
```r
# Formatting data ----------------------------------------------------------
# Subset bold lepus for only sections needed to reduce cognitive load when viewing dataframe
# remove ones that don't contain nucleotide data and ones that arent COI
boldLepus <- rawBoldLepus[, c("processid", "species_name", "markercode", "nucleotides")] |>
  filter(!is.na(nucleotides)) |>
  filter(markercode == "COI-5P")

# Change BOLD's processid column name to "id" to match NCBI database
names(boldLepus)[names(boldLepus) == "processid"] <- "id"

# Manipulate the NCBI dataframe to have correct column names for later merge with boldLepus
nucleotideLepus$id <- word(nucleotideLepus$title, 1L)
nucleotideLepus$species_name <- word(nucleotideLepus$title, 2L, 3L)

# filter out any non cytochrome sequences
nucleotideLepus <- filter(nucleotideLepus, grepl("cytochrome oxidase subunit", title))

# Add marker code column and rearrange columns to allow for clean merge with bold
nucleotideLepus$markercode <- "COI-5P"
nucleotideLepus <- nucleotideLepus[ , c("id", "species_name", "markercode", "nucleotides")]


# MERGE DATAFRAMES
lepusSeq <- merge(nucleotideLepus, boldLepus, all = T)

# Map body masses from Pantheria to the subsetted column for downstream analysis of body masses
lepusSeq$mass_g <- purrr::map(lepusSeq$species_name, function (x) {getAverageMass(pantheriaData, x)}) |>
  as.numeric()

# Remove duplicates, trim Ns from the ends and remove gaps, remove entries with NA species_name, and remove entries with no mass data
lepusSeq <- lepusSeq[!duplicated(lepusSeq$nucleotides), ]

lepusSeq <- lepusSeq |>
  filter(!is.na(species_name)) |>
  filter(mass_g > 0) |>
  mutate(nucleotides2 = str_remove_all(nucleotides, "^N+|N+$|-")) |>
  filter(str_count(nucleotides2, "N") <= (missingData * str_count(nucleotides))) |>
  filter(str_count(nucleotides2) >= median(str_count(nucleotides2)) - lengthVar & str_count(nucleotides2) <= median(str_count(nucleotides2)) + lengthVar)

# Create a subset dataframe containing a random sequence from each species; this allows for comparison between just one sample per species
set.seed(1234) # so that we get the same result

lepusSeqSubset <- lepusSeq |>
  group_by(species_name) |>
  slice_sample(n = 1) |> # Randomly selects one row per species, because Karl told me to do it :)
  ungroup() |>
  as.data.frame()
```

Figures

Figure 1

```r

# View data in radar chart ------------------------------------------------------
# The following chart idea taken from https://r-graph-gallery.com/142-basic-radar-chart.html

# Map the data as names and mass to visualize in a radar chart
massAndNames <- as.data.frame(matrix(lepusSeqSubset$mass_g , ncol=12))
colnames(massAndNames) <- paste("L.", word(lepusSeqSubset$species_name, 2L), sep = " ")

# Add in the upper and lower limits to comply to the formatting requirements for fmsb library
massAndNames <- rbind(rep(5000,12), rep(1000,12), massAndNames)

# check data formatted properly
# head(massAndNames)

# Remove styling from other graph so it doesn't display weird when running
par(mar=c(1,1,1,1))

# Render the radar chart
radarchart( massAndNames, axistype=1,
            #customize the polygon, grid, and labels
            pcol=rgb(1,0.6,0,0.9) , pfcol=rgb(1,0.6,0,0.5) , plwd=4 ,
            cglcol="grey", cglty=1, axislabcol="black", caxislabels=seq(1000,5000, 1000), cglwd=0.8,
            vlcex=0.8
)


```

Body Mass (in Grams) of Each Lepus Species



Explanation for the radar chart:

Since body mass was the only trait being explored (absent are extra traits such as latitude), a radar chart was employed to effectively depict the body mass (in grams) of each of the *Lepus* species to better understand the distribution of weights and compare across species.

Alignment (Preliminary to Figure 2)

```
174
175 # Aligning sequences ──────────────────────────────────────────────
176 # put entire lepus subset into DNAStringSet to work with the library
177 lepusSeqSubset$nucleotides2 ← DNAStringSet(lepusSeqSubset$nucleotides2)
178
179 # Map the names, substituting L. for Lepus for readability.
180 names(lepusSeqSubset$nucleotides2) ← paste("L.", word(lepusSeqSubset$species_name, 2L), sep = " ")
181 # Check that it worked
182 names(lepusSeqSubset$nucleotides2)
183
184 # Conduct alignment with muscle
185 lepusSeqSubsetAlignment ← DNAStringSet(muscle::muscle(lepusSeqSubset$nucleotides2))
186 # Check it out in the browser to see if anything is out of place ⟶ originally, I saw that one of the names was NA
    and I had to go back to remove NA species_name entries
187 #BrowseSeqs(lepusSeqSubsetAlignment)
188
189
190
191 # Clustering ──────────────────────────────────────────────────────
192 # Convert to a dataclass used by Ape for distance clustering
193 lepusSeqBin ← as.DNAbin(lepusSeqSubsetAlignment)
194 # Check conversion is correct
195 class(lepusSeqBin)
196
197 # Create distance matrix
198 distanceMatrix ← dist.dna(lepusSeqBin, model = chosenModel, as.matrix = TRUE, pairwise.deletion = TRUE)
199 # Check out distance matrix worked
200 head(distanceMatrix)
201
202 # PHYLOGENETIC TREE
203 clustersLepusCOI ← DECIPHER::TreeLine(lepusSeqSubsetAlignment,
204                                       myDistMatrix = distanceMatrix,
205                                       method = clusteringMethod,
206                                       model = chosenModel,
207                                       reconstruct = TRUE,
208                                       maxTime = 0.01)
209
```

```
33
34 # Global variables ──────────────────────────────────────────────────
35
36 #### CHECK ONLINE IF THESE ARE THE CORRECT CHOICES FOR WHAT I'M DOING ####
37
38 missingData ← 0.01
39 lengthVar ← 50
40 chosenModel ← "K80" # K2P
41 clusteringMethod ← "ML"
42
```

Explanation for the chosen method:

The maximum likelihood (ML) method was chosen for evolutionary tree generation because of its “statistical consistency, robustness,...and ability to...make full use of original data within a statistical framework” (Zou, Yue, et al., 2024).

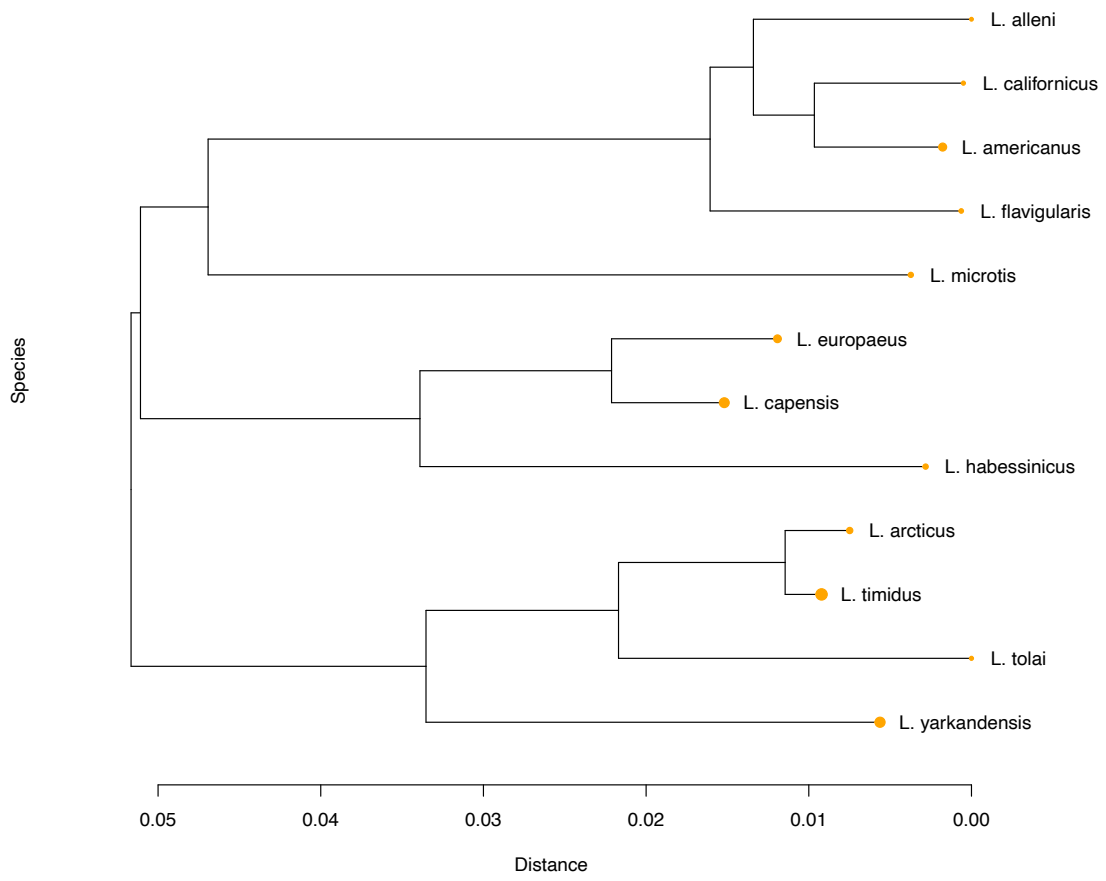
Explanation for using the K80 model:

The Kimura 2-parameter (K2P/K80) model is typically used early in phylogenetic analysis for constructing trees (Salemi et al., 2009, p. 163), which fits handsomely within the context and scope of this analysis.

Figure 2

```
210
211  # Bottom, left, top, right margins
212  par(mar=c(5,5,1,10))
213
214  # Create a vector of masses scaled relative to the max mass
215  maxMass ← max(lepusSeqSubset$mass_g)
216  scaledMass ← lepusSeqSubset$mass_g / maxMass * 2
217
218  # Plot the phylogenetic tree
219  clustersLepusCOI ▷
220    set("leaves_pch", 20)  ▷
221    set("leaves_cex", scaledMass) ▷
222    set("nodes_col", "orange") ▷
223    set("labels_col", "black") ▷
224    plot(horiz = TRUE, xlab = "Distance", ylab = "Species")
225
```

Phylogenetic Tree Depicting the Genetic Relationship Among Lepus Species Based on COI Gene Sequences, with Relative Body Mass Represented by Tip Sizes



● = Relative body mass of *Lepus sp.*

Table 1

```
# Test for phylogenetic conservatism ──────────────────────────────────────
# Export tree as a format used by ape package
tree_phylo ← as.phylo(clustersLepusCOI)

# Lambda estimation will test the extent to which body mass shows phylogenetic signal.
# If it is close to 1, the trait follows the phylogeny
# If it is close to 0, the trait does not really follow the phylogenetic signal and does not exhibit phylogenetic
conservatism
lambda_estimation ← phylosig(tree_phylo, lepusSeqSubset$mass_g, method = "lambda")

print(lambda_estimation)

```

Summary of Important Phylogenetic Signal Outputs

Phylogenetic Signal Lambda	LogL (lambda)
4.787×10^{-5}	-99.78

Results and Discussion

A phylogenetic tree (Figure 2) and phylogenetic signal lambda (Table 1) were created and calculated with data of *Lepus* species from the BOLD, NCBI, and Pantheria databases. The data were aligned with the muscle library's muscle function. A distance matrix was created by utilizing ape's dist.dna function and a phylogenetic tree was created using DECIPHER's TreeLine function. The phylogenetic signal was computed via phytool's phylosig function. The output of the phylogenetic signal lambda test demonstrated that there is no significant evidence that the trait (body mass) follows the phylogenetic signal. Within the scope of this assessment, the trait does not exhibit phylogenetic conservatism, i.e., related species that are genetically close (within the *Lepus* genus) do not share similar body mass traits. The results of this analysis can be confirmed following visual inspection of the phylogenetic tree (Figure 2): closely related species clustered within the same clade do not consistently appear to have a similar body mass, e.g., *L. arcticus* and *L. timidus*.

There are several limitations to this work that should be considered. First, the data were filtered to remove species entries with missing data. While this step was necessary to perform an analysis, i.e., sequence alignment cannot be conducted on a species that has no sequence data, it may have removed key species that are necessary for a thorough analysis. The absence of these species may have adversely impacted the tree and, more importantly, the phylogenetic signal test. In a future study, 47S ribosomal genes may be better suited for analysis than COI genes. Even though COI genes are robust in their ability to determine phylogenetic relationships, the 47S ribosomal genes may produce a more reliable analysis when determining whether the body mass trait follows phylogeny (Law et al., 2024) among *Lepus* species.

Acknowledgements

The author wishes to thank Sophia Teng for the moral support and the conversations that prompted him to question the strength of his analysis and adjust where suitable. The author also wishes to acknowledge that he could not have created the figures without the help of The R Graph Gallery.

References

Bergmann, C. (1847). Über die verhältnisse der wärmeökonomie der thiere zu ihrer grösse. Gött. *Stud*, *1*, 595-708.

Bodnar, U. (2024). *Assignment 1*. [Unpublished manuscript]. University of Guelph.

Law, Pui Pik, et al. "Ribosomal DNA Copy Number Is Associated with Body Mass in Humans and Other Mammals." *Nature Communications*, vol. 15, no. 1, June 2024, p. 5006. *DOI.org (Crossref)*, https://doi.org/10.1038/s41467-024-49397-5.=

McQueen, Alexandra, et al. "Thermal Adaptation Best Explains Bergmann's and Allen's Rules across Ecologically Diverse Shorebirds." *Nature Communications*, vol. 13, no. 1, Aug. 2022, p. 4727. *DOI.org (Crossref)*, https://doi.org/10.1038/s41467-022-32108-3.

Salemi, M., Vandamme, A.-M., & Lemey, P. (Eds.). (2009). *The phylogenetic handbook : a practical approach to phylogenetic analysis and hypothesis testing* (2nd ed.). Cambridge University Press.

Salewski, Volker, and Cortney Watt. "Bergmann's Rule: A Biophysiological Rule Examined in Birds." *Oikos*, vol. 126, no. 2, Feb. 2017, p. oik.03698. *DOI.org (Crossref)*, https://doi.org/10.1111/oik.03698.

Zou, Yue, et al. "Common Methods for Phylogenetic Tree Construction and Their Implementation in R." *Bioengineering*, vol. 11, no. 5, May 2024, p. 480. *DOI.org (Crossref)*, https://doi.org/10.3390/bioengineering11050480.