

## 2. Testing for differences between match/mismatch questions and plotting phylogenies using ggrepel

Sicily Fiennes

19 September 2021

### How to plot a phylogeny

Before you run or try out this code type ‘Ctrl+Shift+H’ to set your working directory, where these files can be stored.

In terms of plotting a phylogeny, you can learn how to here: <http://brettkvo.com/how-to-create-a-ultrametric-dichotomous-phylogeny-in-r-using-vertlife-org-or-birdtree-org/>

Another option, which was the one we followed : <https://daijiang.name/en/2017/08/25/r-phylomatic/>

The following libraries are required

```
library(ape)
library(phytools)

## Loading required package: maps

library(picante)

## Loading required package: vegan

## Loading required package: permute

## Loading required package: lattice

## This is vegan 2.5-7

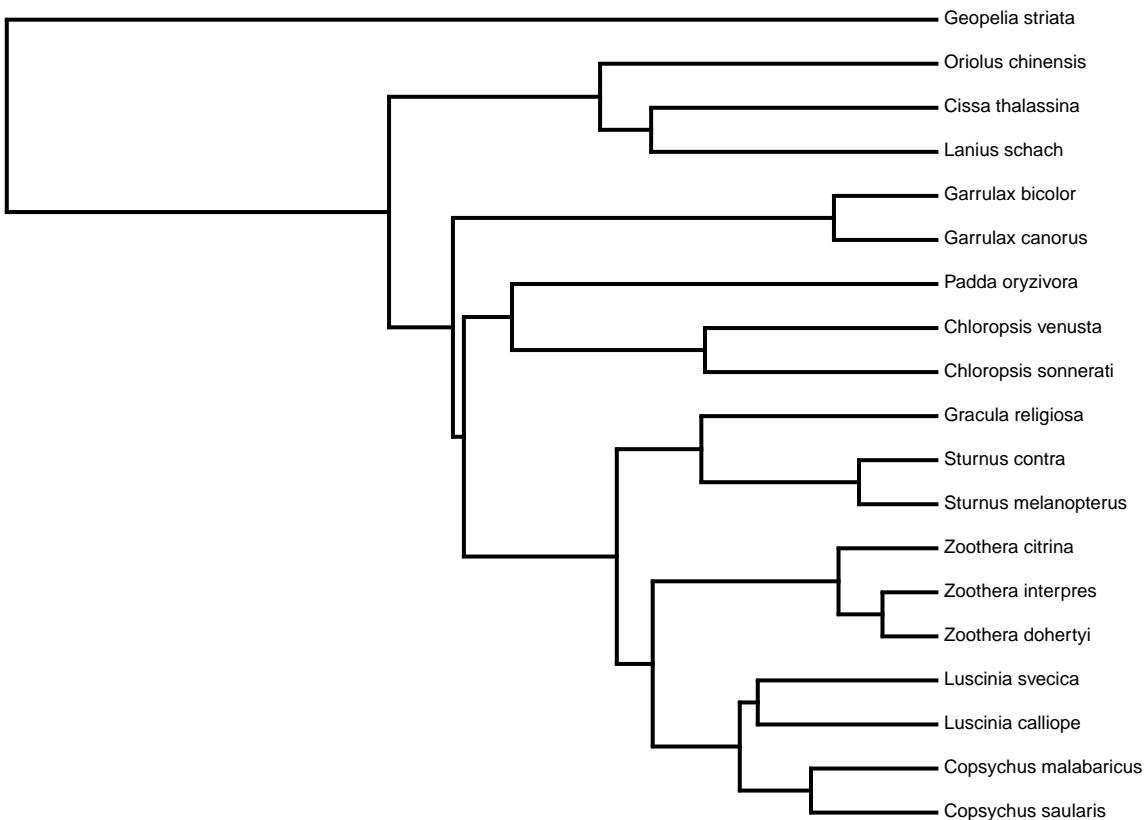
##
## Attaching package: 'vegan'

## The following object is masked from 'package:phytools':
## 
##     scores

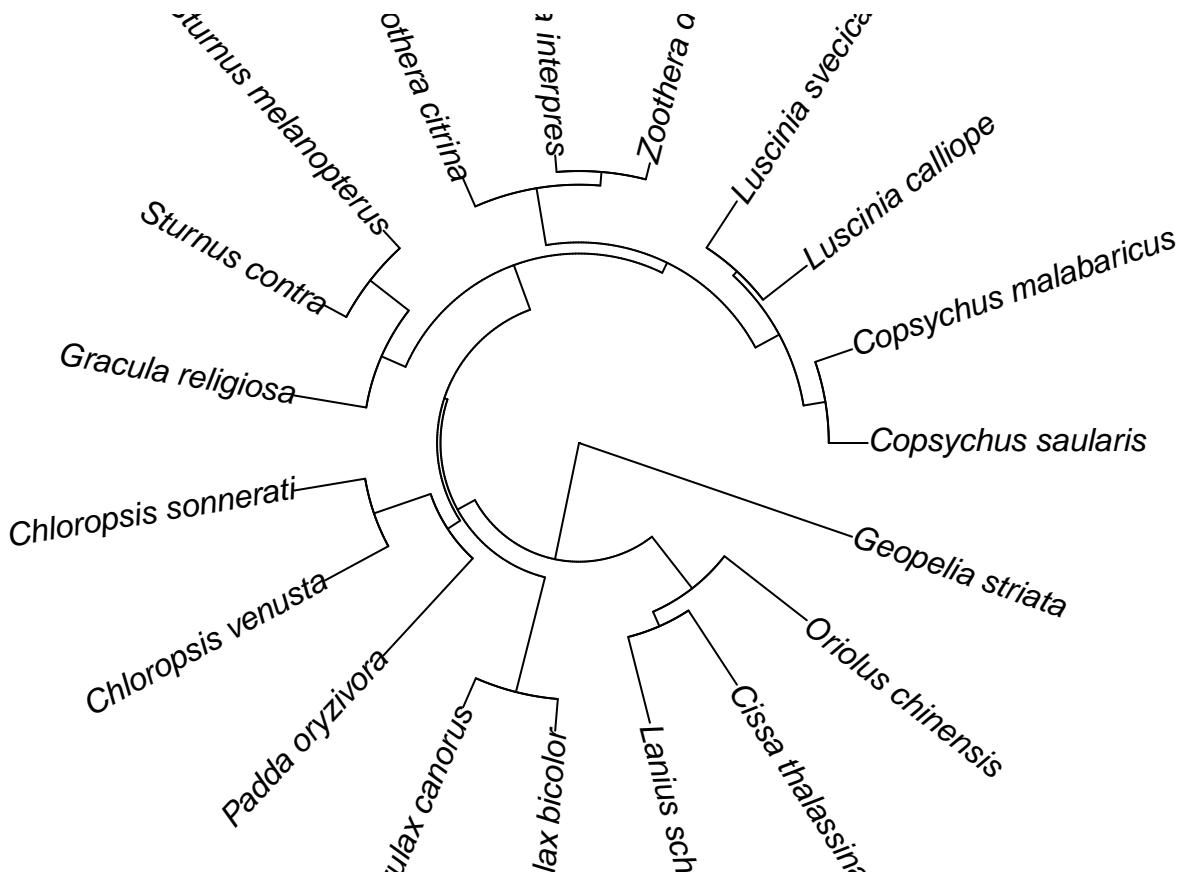
## Loading required package: nlme
```

```
#setwd("D:/Sicily/Documents/R/All R")
treefile <-read.nexus('output.nex')
consensus<-consensus.edges(treefile, consensus.tree=consensus(treefile,p=0.5))

plotTree(consensus, fsize=0.6)
```



```
plot(consensus, type="fan")
```



The file can be written and saved as either a newick, ‘nwk’ file or as a ‘txt’ file. Using the write.nexus command it can be saved as a .nex file.

```
write.tree(consensus, file="newick bird tree.txt")
write.tree(consensus, file="newick bird tree.nwk")
write.nexus(consensus, file="nexus bird tree.nex")
```

We want to make a colour tree, main source of inspiration from here: <https://yulab-smu.top/treedata-book/chapter4.html>, using the ggtree package. However, we found that this package was fairly difficult to use, so we sought another solution. There are ways to decorate a phyloeny with colours, based on a variable. You can also add local images in the ggtree package <https://guangchuangyu.github.io/2015/08/ggtree-annotate-phylogenetic-tree-with-local-images/>

Instead we used the ggfree instead, written by Art Poon: <https://github.com/ArtPoon/ggfree>

```
require(devtools)

## Loading required package: devtools

## Warning: package 'devtools' was built under R version 4.0.5

## Loading required package: usethis

##
## Attaching package: 'devtools'
```

```

## The following object is masked from 'package:permute':
##
##      check

#devtools::install_github("ArtPoon/ggfree")
# We did not run the above line in hthe console, since we already installed ggfree
library(ggfree)

##
## Attaching package: 'ggfree'

## The following object is masked from 'package:ape':
##
##      unroot

require(RColorBrewer)

## Loading required package: RColorBrewer

```

In our case we wanted to decorate the phylogeny with lower accuracy as red and higher accuracy as yellow. We tried the Red-Yellow colour palette. You can view other colour palettes here: <https://www.r-graph-gallery.com/38-rcolorbrewers-palettes.html>. The ‘acc’ variable represents the accuracy per species from our match-mismatch survey. There were 19 species in our phylogeny, so there is an accuracy value for each species. The order of the variables in the vector are according to the order of species in the ‘bird\_tree’ file. You can so print(bird\_tree) to see the order of species in the file, in terms of how closely related each species is.

```

bird_tree <- read.tree("newick bird tree.nwk") # this should be in newick format

acc <- c(0.92777778, 0.9438059, 0.9272353, 0.8306878, 0.8876118, 0.9082126,
       0.9476373, 0.9053498, 0.9333333, 0.9449735, 0.9487179, 0.9656085,
       0.9317739, 0.9272353, 0.9444444, 0.9225589, 0.9382716, 0.9469136,
       0.9583333)

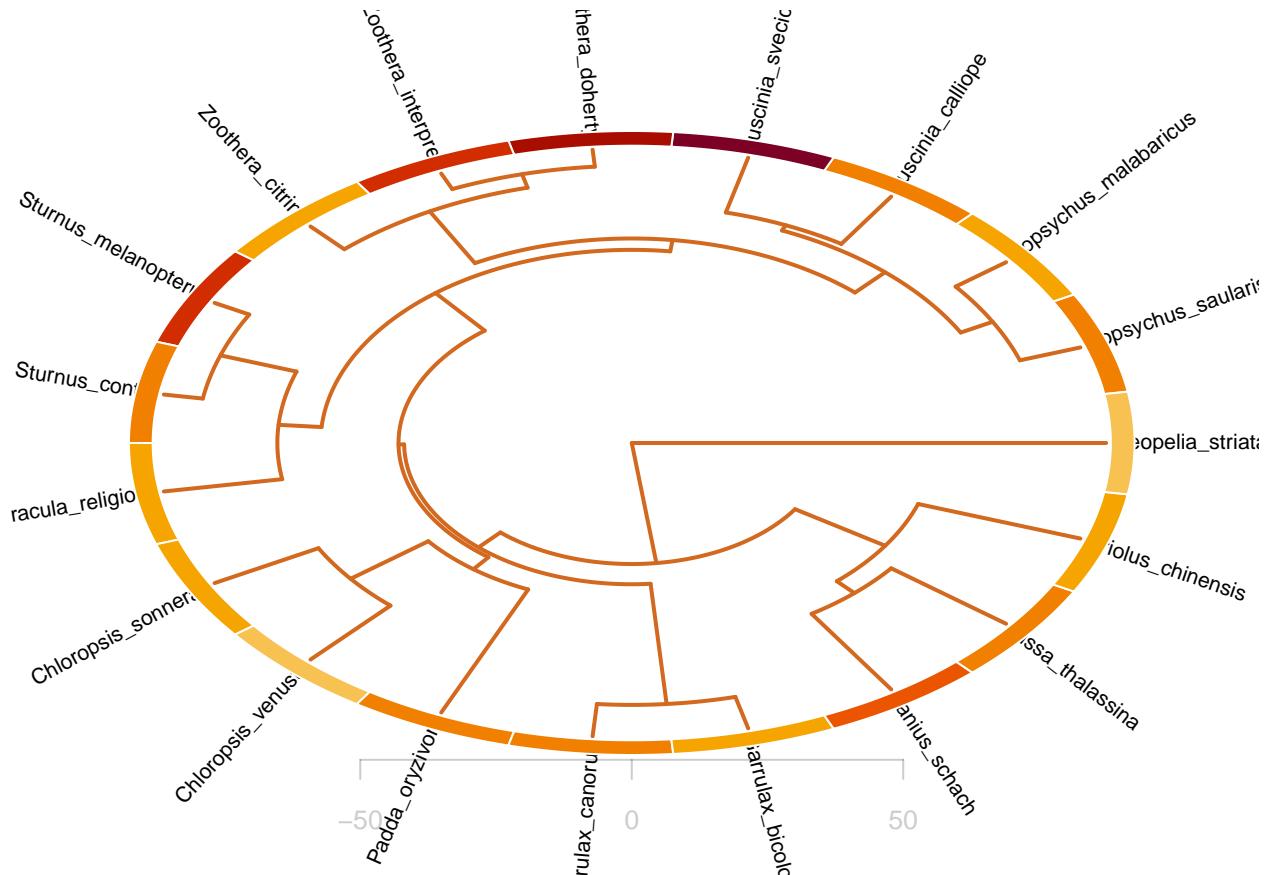
pal <- brewer.pal(9, 'OrRd')[2:9] # was Blues before
bins <- as.integer(acc)

# draw the tree, offsetting the labels for our image
L <- tree.layout(bird_tree, type='o')

pal <- hcl.colors(10, 'YlOrR') # adding ',rev=TRUE' after 'YlOrR' reverses the direction
# of the colour palette, if you need it

col.map <- as.integer(cut(acc, 10))
plot(L, cex.lab=0.7, offset=0, mar=rep(3,4), col='chocolate')
axis(side=1, col=rgb(0,0,0,0.2), col.axis=rgb(0,0,0,0.2), cex.axis=0.8)
image(L, z=as.matrix(col.map), col=pal)

```



```
# If for some reason your accuracy does not line up with the correct species, you can
# use this commented piece of code before you run the col.map
```

```
#cbind(bird_tree$tip.label, acc)
```

This plot shows us what we want through the labels are cut off. However, we saved the plot as a .pdf in the plot panel and then edited the species names and added in photos for each species, using the free editing program Inkscape.

The resultant plot looks like this...

