Traits and trees

Rutger Vos (@rvosa)

1-11-2018

Analyzing trees and traits

In this example we are going to explore whether domesticated ungulates are closer to each other in trait values than they are in evolutionary relatedness. Only some ungulates have ever been domesticated, and there are indications that the succesful ones have been domesticated multiple times (cows, pigs), whereas others have been failures. Apparently there are combinations of trait values among ungulates that determine whether domestication can ever succeed, and only some of them have the right combination (Jared Diamond's Anna Karenina argument). Thus, average pairwise distance among domesticated ungulates clustered by Gower distance should be shorter than everage pairwise evolutionary distance.

To assess this, we are going to build on the occurrences/trees tutorial. First, we need a tree for the Ungulates.

```r
library(taxize)
library(DBI)
```

```
## Warning: package 'DBI' was built under R version 3.4.4
```

```r
library(ape)
source('../R/expand_taxon.R')

# 'ungulates' is a common name, not an accepted higher taxon.
# So what accepted taxa fall under this name?
ungulates <- unique(comm2sci("Ungulates",db="itis")$Ungulates)

# locations of data files
db_file <- '../data/sql/tgd.db'
msw3_file <- '../data/taxa/msw3-all.csv'
supertree_file <- '../data/phylogeny/Bininda-emonds_2007_mammals.tsv'

# we need the tree_id of the msw3_file
db <- dbConnect(RSQLite::SQLite(), db_file)
query <- 'select tree_id from trees where tree_name="%s"'
tree_id <- dbGetQuery(db, sprintf(query, msw3_file))$tree_id

# iterate over the top level 'ungulates' and expand them to species
names <- list()
for ( name in ungulates ) {

    # the actual call to the file we sourced above
    expanded <- expand.taxon(
        db_file     = db_file,
        taxon_name  = toupper(name),
        db          = db,
        tree_id     = tree_id,
        taxon_level = 'SPECIES'
    )

    # post process to get a flat list
```

```
    nnames <- length(expanded)
    expanded_names <- vector(mode="list", length=nnames)
    for ( i in 1:nnames ) {
        expanded_names[[i]] <- expanded[[i]]$taxon_name
    }
    names <- c(names,expanded_names)
}

# now switch to the mammal supertree and extract its subtree
source('../R/make_phylo.R')
tree_id <- dbGetQuery(db, sprintf(query, supertree_file))$tree_id
tree <- make.phylo(db_file, tree_id, names)
plot(tree, show.tip.label=F)
axisPhylo(side = 1)
```
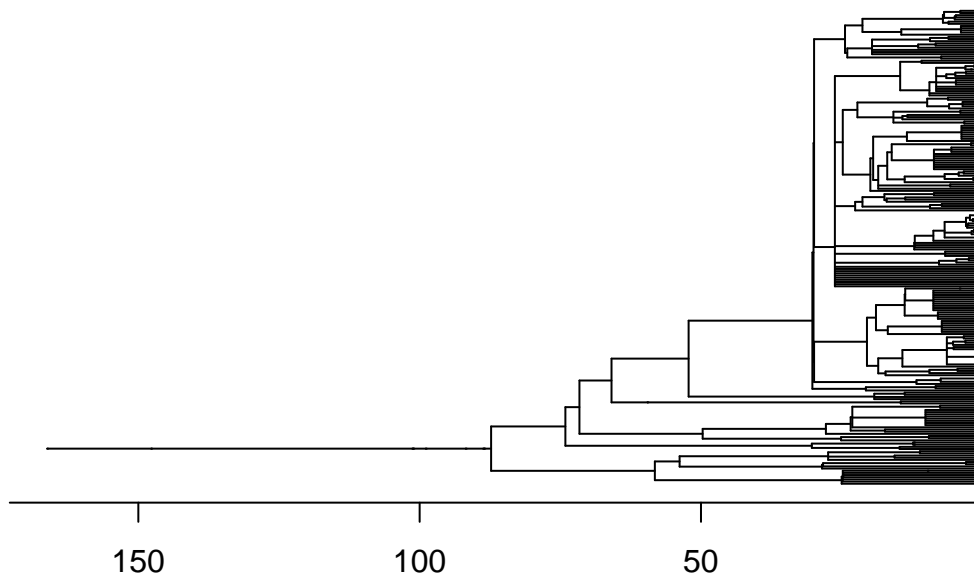

Now that we have a tree, we can look for character state values for the tips in in the tree. As follows:

```
source('../R/make_traitset.R')

# we use the traits from PanTHERIA
df <- make.traitset(db_file, tree$tip.label, 'PanTHERIA_1-0_WR05_Aug2008')
```

Now let's do an analysis:

```
library(FD)
```

```
## Loading required package: ade4

## Loading required package: geometry

## Loading required package: magic

## Loading required package: abind

## Loading required package: vegan

## Loading required package: permute

## Loading required package: lattice

## This is vegan 2.4-4
```

```r
# in this case all values are numeric, but
# come out of the database as character,
# so we coerce them here
for (i in 1:50) {
    df[,i] <- as.numeric(df[,i])
}
```

```
## Warning: NAs introduced by coercion
```

```r
# here we compute the Gower distance from FD,
# then compute a dendrogram using neighbor-joining
dist <- gowdis(df)
```

```
## Warning in min(x): no non-missing arguments to min; returning Inf
```

```
## Warning in max(x): no non-missing arguments to max; returning -Inf
```

```
## Warning in min(x): no non-missing arguments to min; returning Inf
```

```
## Warning in max(x): no non-missing arguments to max; returning -Inf
```

```
## Warning in max(v, na.rm = T): no non-missing arguments to max; returning -
## Inf
```

```
## Warning in max(v, na.rm = T): no non-missing arguments to max; returning -
## Inf
```

```
## Warning in min(v, na.rm = T): no non-missing arguments to min; returning
## Inf
```

```
## Warning in min(v, na.rm = T): no non-missing arguments to min; returning
## Inf
```

```r
plot(njs(dist), type = "unrooted")
```