

# Lab 3 Bioinformatics

*Duc Doung, Martin Smelik, Raymond Sseguya*

*2018 M12 8*

## Assignment 1

### 1.1

In Chapter 3 we read the sequences and use clustal function for multiple sequence alignment and we use mafft function for alignment of the sequences. Then we just name the sequences and save the data.

In Chapter 5 we compute the distance matrices for according to six different models. With two of the obtained distances we compute nj trees. We use one of the leaves as root and perform bootstrapping to see how reliable the tree is. We did not manage to make work this line “`phyml.sylvia <- phymltest(“sylvia.txt“, execname = “~/phyml“)`” and therefore we could not perform analysis related to it. Using the function `chronopl` with cross validation we estimated the node ages and saved this as `sylvia.chrono`.

In Chapter 6 we are using the `ace` function to see the properties of geographical range. With this we can see the likelihoods of the range. With information from `geo_range` we can fill the leaves with corresponding colors and using the likelihood we can fill the nodes with the colors corresponding to the likelihood

### 1.2

```
gr_sym$rates
```

```
## [1] 3.983947 0.000000 11.884956
```

```
gr_sym$se
```

```
## [1] 2.319376      NaN 5.916586
```

```
gr_er$rates
```

```
## [1] 5.988848
```

```
gr_er$se
```

```
## [1] 2.230071
```

```
gr_ard$rates
```

```
## [1] 2.967695 0.000000 0.000000 0.000000 0.000000 7.202527
```

```
gr_ard$se
```

```
## [1] 2.331727 2.784387      NaN      NaN 9.761911 2.599815
```

We used 3 different models for analyzing the geographical range variable. In the results we can see the maximum likelihood estimates of the transition rates and also standard errors of these rates. First model corresponds to symmetrical model, the second one corresponds to the equal-rates model and the last one to all-rates-different model. Our goal is for sure minimize the standard errors. We can see that the lowest standard error we can get is with equal-rates model.

## Assignment 2

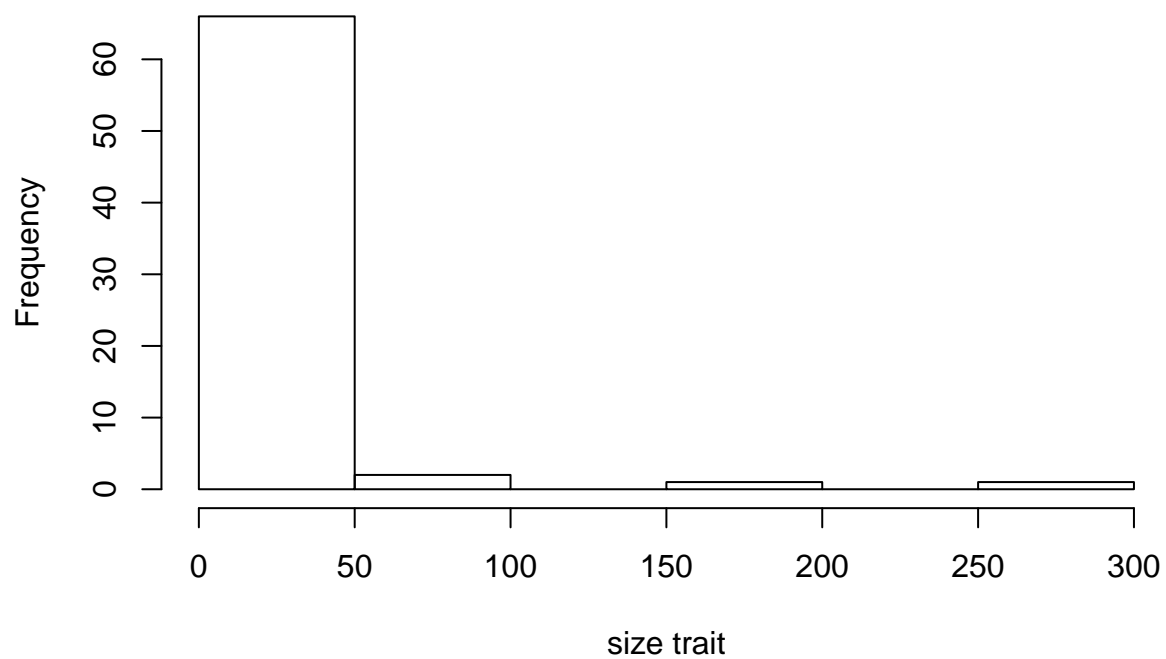
### 2.1

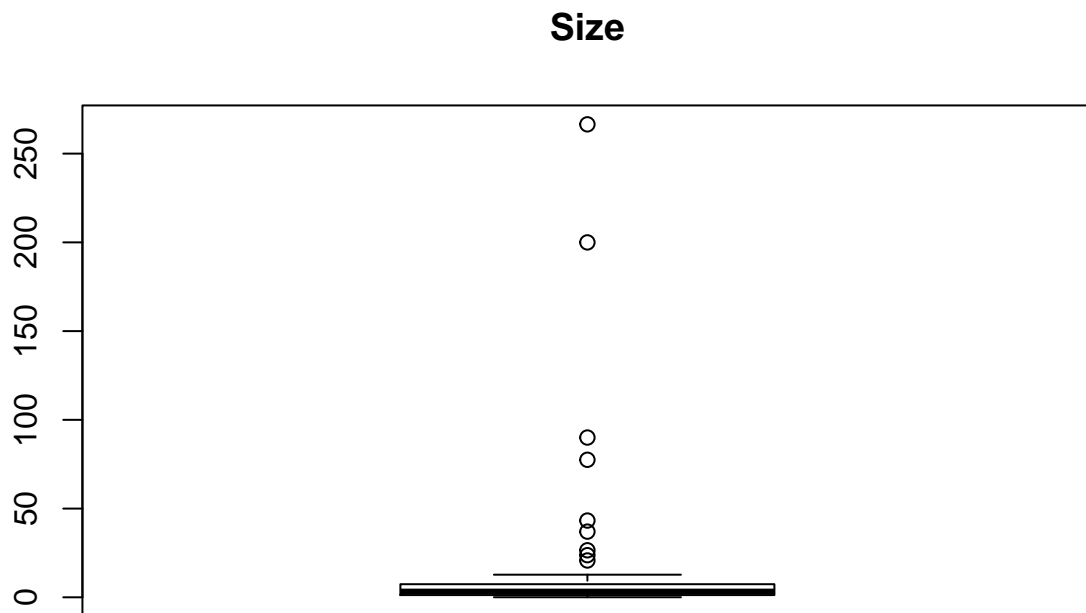
We analysed the carnivores dataset, `data(carni70)`, from the `ade4` R package. The data is about 70 species.

```
head(carni70$tab, 10)
```

```
##              size range
## Puma_concolor 37.01 36.00
## Herpailurus_yaguaroundi 2.59 24.00
## Leopardus_wiedii 3.20 21.75
## Leopardus_pardalis 7.90 24.50
## Oreailurus_jacobita 3.99 1.75
## Oncifelis_colocolo 2.99 6.25
## Oncifelis_guigna 2.23 1.25
## Oncifelis_geoffroyi 3.80 4.75
## Leopardus_tigrinus 2.19 20.25
## Lynx_rufus 11.20 13.50
```

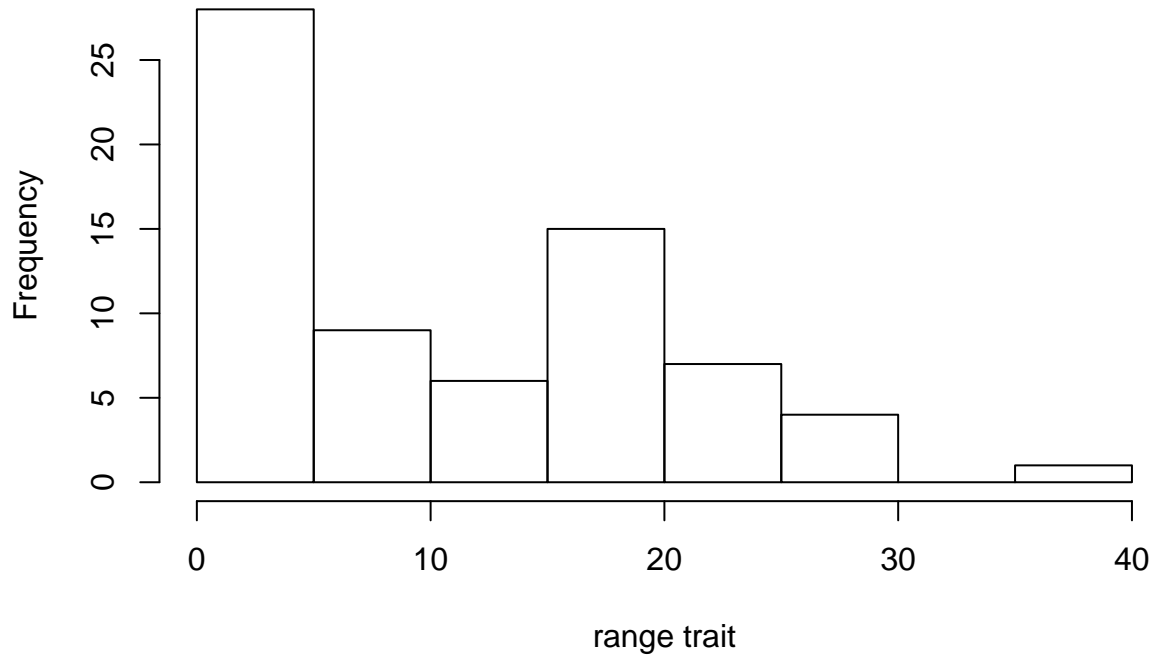
**Histogram of Size**

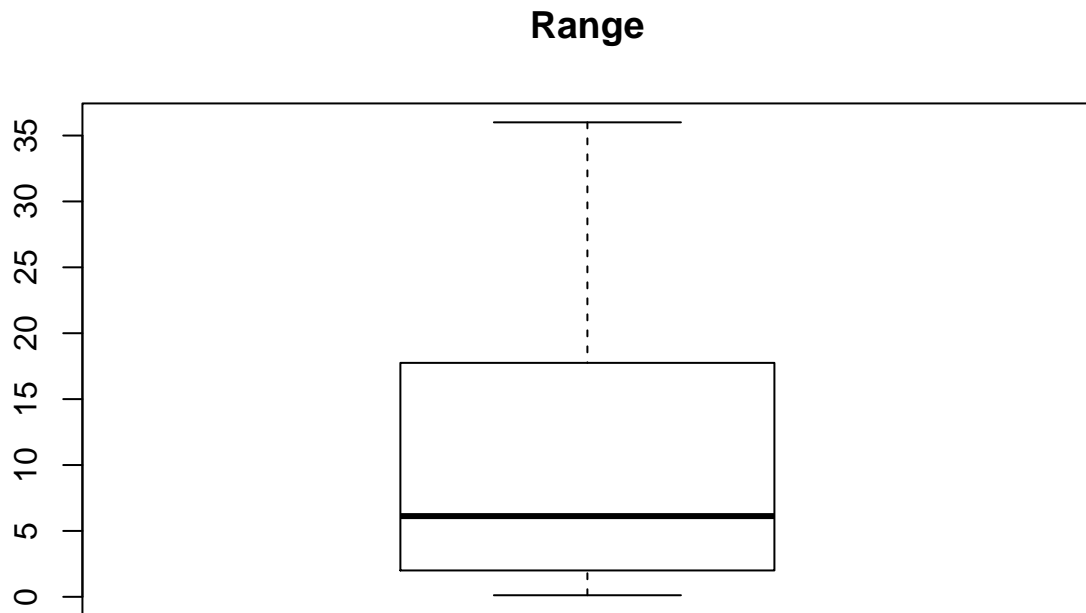




The distribution of the size trait is skewed to the left. In other words, the majority of the 70 species have small sizes. Some of the species have outlier sizes.

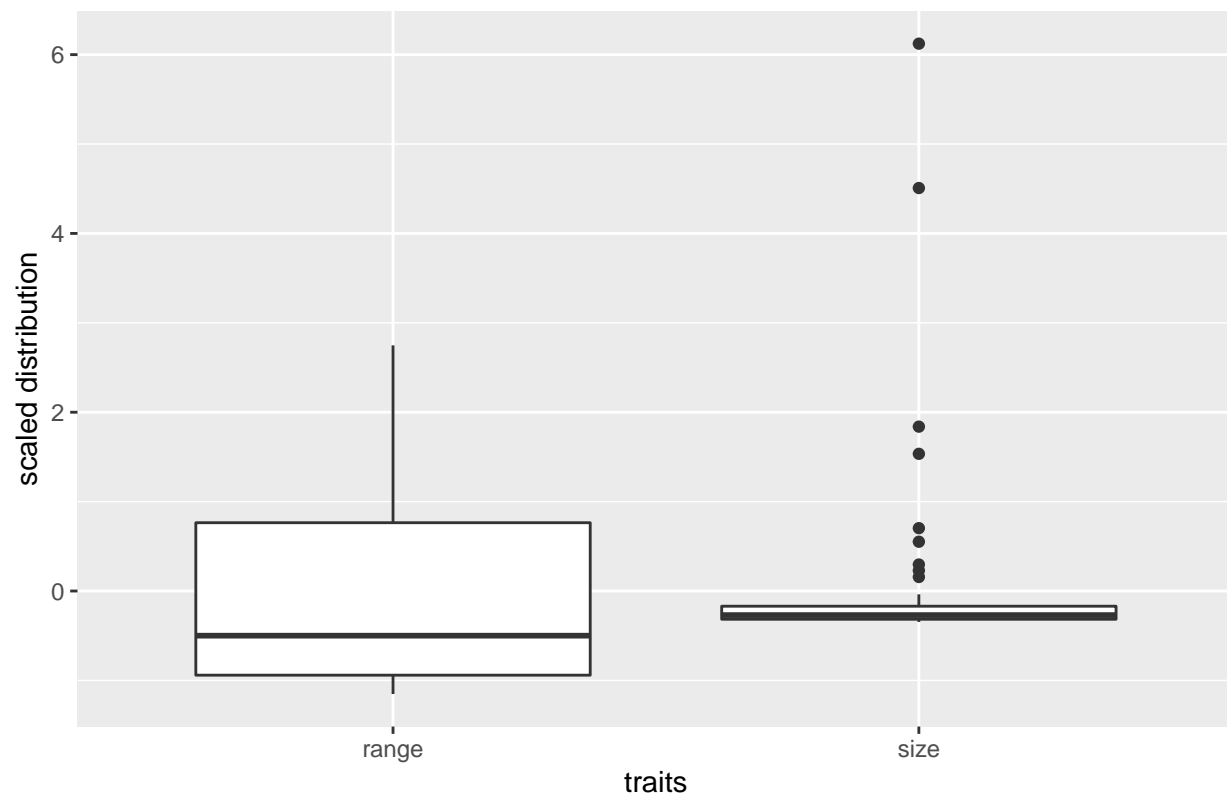
**Histogram of Range**





The distribution of the range trait is also generally more balanced however with evident skewness to the left. There are fewer and fewer species with longer ranges.

## Size and Range compared on the same scale



After scaling the data and comparing the distributions of size and range traits on the same scale with boxplots, it is clear that the size trait is more skewed towards low values than the range trait.

```
rownames(carni70_2[which.max(carni70_2[, "size"]), ])
```

```
## [1] "Ursus_arctos"
```

```
rownames(carni70_2[which.min(carni70_2[, "size"]), ])
```

```
## [1] "Mustela_nivalis"
```

```
rownames(carni70_2[which.max(carni70_2[, "range"]), ])
```

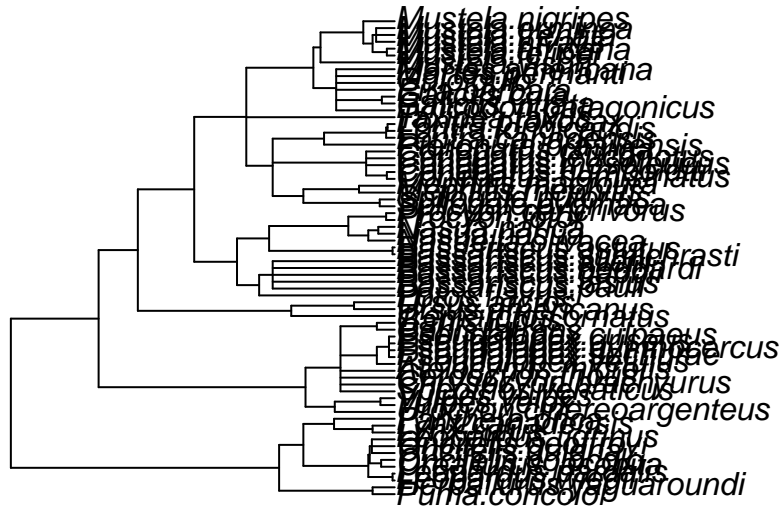
```
## [1] "Puma_concolor"
```

```
rownames(carni70_2[which.min(carni70_2[, "range"]), ])
```

```
## [1] "Bassariscus_pauli"
```

The species “Ursus\_arctos” has the biggest size while “Mustela\_nivalis” the smallest size. The species “Puma\_concolor” has the biggest range while the species “Bassariscus\_pauli” has the smallest range

```
tree_phylo <- ape::read.tree(text=carni70_1)
plot(tree_phylo)
```



Above is a plot of the phylogenetic tree.

## 2.2

### 2.2.1 Both traits evolve as independent Brownian motions

```
mvMORPH::mvBM(tree_phylo, carni70$tab, model="BM1",
  param = list(constraint = "diagonal"))
```

```
## row names of the data matrix must match tip names of your phylogeny!
## successful convergence of the optimizer
## a reliable solution has been reached
##
## -- Summary results for constrained rate BM1 model --
## LogLikelihood:   -589.2373
## AIC:             1186.475
## AICc:            1186.771
## 4 parameters
##
## Estimated rate matrix
## -----
```



```
##           size    range
## size  73.70335  0.00000
## range  0.00000 13.54524
##
## Estimated root state
## -----
##           size    range
## theta: 38.43947 13.78439
```

Because of the assumption that the two traits follow independent processes, the off diagonal elements of the estimated rate matrix are zero. From the rate matrix, we see that the changes in the size trait along the phylogenetic tree are more frequent (at 73) than the changes of the range trait on the same tree. That explains why the histogram of the size trait is more left skewed than that of the range trait (13). This model also suggests that the ancestor(root) had a size of 38.43947 and a range of 13.78439.

## 2.2.2 The traits evolve as a correlated Brownian motion

```
library(ouch)
tree_ouch <- ouch::ape2ouch(tree_phylo, branch.lengths = tree_phylo$edge.length)
library(mvSLOUCH)
mvSLOUCH::BrownianMotionModel(tree_ouch, data = as.matrix(carni70$tab))
```

```
## $ParamsInModel
## $ParamsInModel$Sxx
##           size    range
## size  171.415584  0.00000
## range   1.327048 26.01045
##
## $ParamsInModel$vX0
##           [,1]
## size    8.434762
## range 11.084473
##
##
## $ParamSummary
## $ParamSummary$StS
##           size    range
## size  29383.3025 227.4768
## range   227.4768 678.3047
##
## $ParamSummary$LogLik
## [1] -656.7365
##
## $ParamSummary$dof
## [1] 5
##
## $ParamSummary$m2loglik
## [1] 1313.473
##
## $ParamSummary$aic
## [1] 1323.473
```

```
##
## $ParamSummary$aic.c
## [1] 1323.921
##
## $ParamSummary$sic
## [1] 1338.181
##
## $ParamSummary$bic
## [1] 1338.181
##
## $ParamSummary$RSS
## [1] 140
```

Under the assumption that the two traits follow correlated Brownian motion processes, the ancestor (root) had a size of 8.434762 and a range of 11.084473. Even under this model, the size trait changes more frequently than the range trait, with 29383.3025 for size versus 678.3047 for range. The difference between size and range is clearly indicated in the skewnesses of histograms.

### 2.2.3 independent Ornstein Uhlenbeck processes

```
mvMORPH::mvOU(tree_phylo, data=carni70$tab$size, model = c("OU1"),
  diagnostic = FALSE, echo = TRUE)
```

```
## species in the matrix are assumed to be in the same order as in the phylogeny, otherwise specify row
##
## -- Summary results --
## LogLikelihood: -324.2639
## AIC: 654.5278
## AICc: 654.8914
## 3 parameters
##
## Estimated theta values
## -----
## OU1 38.43947
##
## ML alpha values
## -----
## 8.737191e-10
##
## ML sigma values
## -----
## 73.70246
```

```
mvMORPH::mvOU(tree_phylo, data=carni70$tab$range, model = c("OU1"),
  diagnostic = FALSE, echo = TRUE)
```

```
## species in the matrix are assumed to be in the same order as in the phylogeny, otherwise specify row
```

```
##
## -- Summary results --
## LogLikelihood:    -248.9895
## AIC:              503.979
## AICc:             504.3427
## 3 parameters
##
## Estimated theta values
## -----
## OU1 11.08609
##
## ML alpha values
## -----
## 0.2295954
##
## ML sigma values
## -----
## 38.05329
```

The following is one of the mathematical representations of a Ornstein-Uhlenbeck processes sourced from the Wikipedia page on Ornstein–Uhlenbeck process

## Representation via a stochastic differential equation [\[ edit \]](#)

An Ornstein–Uhlenbeck process,  $x_t$ , satisfies the following [stochastic differential equation](#):

$$dx_t = \theta(\mu - x_t) dt + \sigma dW_t$$

where  $\theta > 0$ ,  $\mu$ , and  $\sigma > 0$  are parameters and  $W_t$  denotes the [Wiener process](#).

This shows that the OU process comprises a Wiener process component (Brownian motion component.) When we modeled the changes in both the size and range traits, we obtained theta and sigma values. The theta and sigma values help us compare the two traits. The size trait has a theta value of 38.43947 which is bigger than the theta value for range of 11.08609. Given the above shown differential question, this implies that the Ornstein-Uhlenbeck change process of size trait is bigger than that of the range trait. Additionally concerning the sigma values, the size trait has a bigger sigma value of 73.70246 as compared to that of the range trait of 38.05329. The sigma value is the Wiener process drift term. That means that the change in the size trait takes longer to reach equilibrium (or stabilise) as compared to the change in the range trait.

### 2.2.4 traits evolve as a bivariate Ornstein-Uhlenbeck process

```
mvMORPH::mvOU(tree_phylo, data=carni70$tab, model = c("OU1"),
              diagnostic = TRUE, echo = TRUE)
```

## row names of the data matrix must match tip names of your phylogeny!

```
## successful convergence of the optimizer
## a reliable solution has been reached
##
## -- Summary results --
## LogLikelihood:    -572.6063
## AIC:              1161.213
## AICc:             1162.312
## 8 parameters
##
## Estimated theta values
## -----
##           size      range
## OU1 37.55833 11.42602
##
## ML alpha values
## -----
##           size      range
## size  8.345549e-05 -0.003859286
## range -3.859286e-03  0.233577603
##
## ML sigma values
## -----
##           size      range
## size  73.766441  5.859783
## range  5.859783 38.247492
```

This model does not give us notable new information from the earlier approaches. However, it is worth noting that the model gives us a very low log likelihood of -572.6063 which is very good. This means it is one of the good models to study further

## 2.2.5 size evolves as a Brownian motion and range as an Ornstein Uhlenbeck

```
mvslouchModel(tree_ouch, kY=1,
  data = as.matrix(cbind(carni70$tab$range, carn170$tab$size)))
```

The model, where the size changes as a Brownian motion and range as an Ornstein Uhlenbeck process is perhaps the best model because it has the lowest log likelihood of -641.8002. The objective is the minimise the log likelihood therefore this fifth model gives the best result in that regard.

Below is the output from the R Console.

```
# [1] "Starting point of heuristic search procedure : "
#           A           Syy
# -0.3038808 -1.1454798
# $`FinalFound`
# $`FinalFound`$`HeuristicSearchPointFinalFind`
#           A           Syy           LogLik
#    2.339259    3.712662 -641.801747
#
# $`FinalFound`$ParamsInModel
# $`FinalFound`$ParamsInModel$`A`
```

```

#           V1
# V1 10.37354
#
# $`FinalFound`$ParamsInModel$B
#           V2
# V1 -0.6884966
#
# $`FinalFound`$ParamsInModel$mPsi
#       reg.1
# V1 10.17529
#
# $`FinalFound`$ParamsInModel$mPsi0
#       [,1]
# V1      0
#
# $`FinalFound`$ParamsInModel$uY0
#       [,1]
# V1 10.73511
#
# $`FinalFound`$ParamsInModel$uX0
#       [,1]
# V2 8.434762
#
# $`FinalFound`$ParamsInModel$Syy
#           V1
# V1 40.96271
#
# $`FinalFound`$ParamsInModel$Syyx
#       V2
# V1 0
#
# $`FinalFound`$ParamsInModel$Sxy
#       V1
# V2 0
#
# $`FinalFound`$ParamsInModel$Sxx
#           V2
# V2 171.4156
#
#
# $`FinalFound`$ParamSummary
# $`FinalFound`$ParamSummary$`phyl.halflife`
# $`FinalFound`$ParamSummary$`phyl.halflife`$`directions`
#       [,1]
# [1,]      1
#
# $`FinalFound`$ParamSummary$`phyl.halflife`$halflives
#           [,1]
# eigenvalues 10.37354269
# halflife      0.06681875
# %treeheight  6.68187524
#
# $`FinalFound`$ParamSummary$`phyl.halflife`$halflifeLowerbounds

```

```

# [1] 0.06681875
#
#
# `$FinalFound`$ParamSummary$expmtA
#           V1
# V1 3.124839e-05
#
# `$FinalFound`$ParamSummary$optimal.regression
#           V2
# V1 0.06637044
#
# `$FinalFound`$ParamSummary$mPsi.rotated
#           reg.1
# V1 10.17497
#
# `$FinalFound`$ParamSummary$mPsi0.rotated
#           [,1]
# V1      0
#
# `$FinalFound`$ParamSummary$cov.matrix
#           V1           V2
# V1 191.5953 1762.193
# V2 1762.1927 29383.302
#
# `$FinalFound`$ParamSummary$corr.matrix
#           V1           V2
# V1 1.0000000 0.7426952
# V2 0.7426952 1.0000000
#
# `$FinalFound`$ParamSummary$conditional.cov.matrix
#           V1
# V1 85.91207
#
# `$FinalFound`$ParamSummary$conditional.corr.matrix
#           V1
# V1 1
#
# `$FinalFound`$ParamSummary$stationary.cov.matrix
#           V1
# V1 87.1148
#
# `$FinalFound`$ParamSummary$stationary.corr.matrix
#           V1
# V1 1
#
# `$FinalFound`$ParamSummary$optima.cov.matrix
#           V1
# V1 129.4345
#
# `$FinalFound`$ParamSummary$optima.corr.matrix
#           V1
# V1 1
#
#

```

```

# `$FinalFound`$ParamSummary$cov.with.optima
#           V1
# V1 116.9575
#
# `$FinalFound`$ParamSummary$corr.with.optima
#           V1
# V1 0.7426952
#
# `$FinalFound`$ParamSummary$evolutionary.regression
#           V2
# V1 0.05997259
#
# `$FinalFound`$ParamSummary$StS
#           V1           V2
# V1 1677.944         0.0
# V2    0.000 29383.3
#
# `$FinalFound`$ParamSummary$LogLik
# [1] -641.8017
#
# `$FinalFound`$ParamSummary$dof
# [1] 6
#
# `$FinalFound`$ParamSummary$m2loglik
# [1] 1283.603
#
# `$FinalFound`$ParamSummary$aic
# [1] 1295.603
#
# `$FinalFound`$ParamSummary$aic.c
# [1] 1296.235
#
# `$FinalFound`$ParamSummary$sic
# [1] 1313.253
#
# `$FinalFound`$ParamSummary$bic
# [1] 1313.253
#
# `$FinalFound`$ParamSummary$RSS
# `$FinalFound`$ParamSummary$RSS$`RSS`
#           [,1]
# [1,] 139.8423
#
# `$FinalFound`$ParamSummary$RSS$R2
#           [,1]
# [1,] 0.0177938
#
#
# `$FinalFound`$ParamSummary$trait.regression
# `$FinalFound`$ParamSummary$trait.regression[[1]]
#           V2
# V1 0.05997259
#

```

```

#
# `$FinalFound`$ParamSummary$confidence.interval
# `$FinalFound`$ParamSummary$confidence.interval`$regression.summary`
# `$FinalFound`$ParamSummary$confidence.interval`$regression.summary`$`B.regression.confidence.interval
#   Lower.end Estimated.Point Upper.end
# V1 -1.888154      -0.6884966 0.5111613
#
# `$FinalFound`$ParamSummary$confidence.interval`$regression.summary`$mPsi.regression.confidence.interv
#   Lower.end Estimated.Point Upper.end
# V1 6.944177      10.17529 13.40641
#
#
#
#
# `$FinalFound`$LogLik
# [1] -641.8017
#
#
# $MaxLikFound
# $MaxLikFound`$HeuristicSearchPointMaxLik`
#       A       Syy       LogLik
# 2.339259 3.712662 -641.800203
#
# $MaxLikFound$ParamsInModel
# $MaxLikFound$ParamsInModel`A`
#       V1
# V1 10.37354
#
# $MaxLikFound$ParamsInModel$B
#       V2
# V1 -0.6792388
#
# $MaxLikFound$ParamsInModel$mPsi
#       reg.1
# V1 10.15824
#
# $MaxLikFound$ParamsInModel$mPsi0
#       [,1]
# V1      0
#
# $MaxLikFound$ParamsInModel$vY0
#       [,1]
# V1 10.72571
#
# $MaxLikFound$ParamsInModel$vX0
#       [,1]
# V2 8.434762
#
# $MaxLikFound$ParamsInModel$Syy
#       V1
# V1 40.96271
#
# $MaxLikFound$ParamsInModel$Syx

```



```

#      V2
# V1  0
#
# $MaxLikFound$ParamsInModel$Sxy
#      V1
# V2  0
#
# $MaxLikFound$ParamsInModel$Sxx
#           V2
# V2 171.4156
#
#
# $MaxLikFound$ParamSummary
# $MaxLikFound$ParamSummary$`phyl.halflife`
# $MaxLikFound$ParamSummary$`phyl.halflife`$`directions`
#           [,1]
# [1,]      1
#
# $MaxLikFound$ParamSummary$`phyl.halflife`$halflives
#           [,1]
# eigenvalues 10.37354269
# halflife      0.06681875
# %treeheight  6.68187524
#
# $MaxLikFound$ParamSummary$`phyl.halflife`$halflifeLowerbounds
# [1] 0.06681875
#
#
# $MaxLikFound$ParamSummary$expmtA
#           V1
# V1 3.124839e-05
#
# $MaxLikFound$ParamSummary$optimal.regression
#           V2
# V1 0.065478
#
# $MaxLikFound$ParamSummary$mPsi.rotated
#           reg.1
# V1 10.15793
#
# $MaxLikFound$ParamSummary$mPsi0.rotated
#           [,1]
# V1      0
#
# $MaxLikFound$ParamSummary$cov.matrix
#           V1      V2
# V1 188.6378 1738.498
# V2 1738.4977 29383.302
#
# $MaxLikFound$ParamSummary$corr.matrix
#           V1      V2
# V1 1.0000000 0.7384302
# V2 0.7384302 1.0000000

```

```

#
# $MaxLikFound$ParamSummary$conditional.cov.matrix
#      V1
# V1 85.77755
#
# $MaxLikFound$ParamSummary$conditional.corr.matrix
#      V1
# V1 1
#
# $MaxLikFound$ParamSummary$stationary.cov.matrix
#      V1
# V1 86.94816
#
# $MaxLikFound$ParamSummary$stationary.corr.matrix
#      V1
# V1 1
#
# $MaxLikFound$ParamSummary$optima.cov.matrix
#      V1
# V1 125.977
#
# $MaxLikFound$ParamSummary$optima.corr.matrix
#      V1
# V1 1
#
# $MaxLikFound$ParamSummary$cov.with.optima
#      V1
# V1 113.8334
#
# $MaxLikFound$ParamSummary$corr.with.optima
#      V1
# V1 0.7384302
#
# $MaxLikFound$ParamSummary$evolutionary.regression
#      V2
# V1 0.05916618
#
# $MaxLikFound$ParamSummary$StS
#      V1      V2
# V1 1677.944    0.0
# V2    0.000 29383.3
#
# $MaxLikFound$ParamSummary$LogLik
# [1] -641.8002
#
# $MaxLikFound$ParamSummary$dof
# [1] 6
#
# $MaxLikFound$ParamSummary$m2loglik
# [1] 1283.6
#
# $MaxLikFound$ParamSummary$aic
# [1] 1295.6

```

```

#
# $MaxLikFound$ParamSummary$aic.c
# [1] 1296.232
#
# $MaxLikFound$ParamSummary$sic
# [1] 1313.25
#
# $MaxLikFound$ParamSummary$bic
# [1] 1313.25
#
# $MaxLikFound$ParamSummary$RSS
# $MaxLikFound$ParamSummary$RSS$`RSS`
#           [,1]
# [1,] 139.8697
#
# $MaxLikFound$ParamSummary$RSS$R2
#           [,1]
# [1,] 0.01778443
#
#
# $MaxLikFound$ParamSummary$trait.regression
# $MaxLikFound$ParamSummary$trait.regression[[1]]
#           V2
# V1 0.05916618
#
#
# $MaxLikFound$ParamSummary$confidence.interval
# $MaxLikFound$ParamSummary$confidence.interval$`regression.summary`
# $MaxLikFound$ParamSummary$confidence.interval$`regression.summary`$`B.regression.confidence.interval`
#   Lower.end Estimated.Point Upper.end
# V1 -1.867116      -0.6792388 0.5086384
#
# $MaxLikFound$ParamSummary$confidence.interval$`regression.summary`$mPsi.regression.confidence.interva
#   Lower.end Estimated.Point Upper.end
# V1  6.883083      10.15824  13.4334
#
#
#
# $MaxLikFound$LogLik
# [1] -641.8002
#
#

```

## Appendix

```

library(ape)

###
### Chapter 3
###

```

```

x <- paste("AJ5345", 26:49, sep = "")
x <- c("Z73494", x)
sylvia.seq <- read.GenBank(x)

sylvia.clus <- clustal(sylvia.seq) ## storing to the matrix
library(phyloch)
sylvia.mafft <- mafft(sylvia.seq, path = "C:/Users/smelo/Downloads/mafft/mafft-win/mafft")
#sylvia.mafft <- mafft(sylvia.seq)
identical(sylvia.clus[x, ], sylvia.mafft[x, ])

taxa.sylvia <- attr(sylvia.seq, "species")
names(taxa.sylvia) <- names(sylvia.seq)
rm(sylvia.seq)
taxa.sylvia[1] <- "Sylvia_atricapilla"
taxa.sylvia[24] <- "Sylvia_abyssinica"

sylvia.eco <- read.table("sylvia_data.txt")
str(sylvia.eco)
rownames(sylvia.eco)
save(sylvia.clus, taxa.sylvia, sylvia.eco,
     file = "sylvia.RData")

###
### Chapter 5
###

syl.K80 <- dist.dna(sylvia.clus, pairwise.deletion = TRUE)
syl.F84 <- dist.dna(sylvia.clus, model = "F84", p = TRUE)
syl.TN93 <- dist.dna(sylvia.clus, model = "TN93", p = TRUE)
syl.GG95 <- dist.dna(sylvia.clus, model = "GG95", p = TRUE)

round(cor(cbind(syl.K80, syl.F84, syl.TN93, syl.GG95)), 3)

syl.JC69 <- dist.dna(sylvia.clus, model = "JC69", p = TRUE)
syl.raw <- dist.dna(sylvia.clus, model = "raw", p = TRUE)
layout(matrix(1:2, 1))
plot(syl.JC69, syl.raw)
abline(b = 1, a = 0) # draw x = y line
plot(syl.K80, syl.JC69)
abline(b = 1, a = 0)

layout(matrix(1:3, 1))
for (i in 1:3) {
  s <- logical(3); s[i] <- TRUE
  x <- sylvia.clus[, s]
  d <- dist.dna(x, p = TRUE)
  ts <- dist.dna(x, "Ts", p = TRUE)
  tv <- dist.dna(x, "Tv", p = TRUE)
  plot(ts, d, xlab = "Number of Ts or Tv", col = "blue",
       ylab = "K80 distance", xlim = range(c(ts, tv)),
       main = paste("Position", i))
  points(tv, d, col = "red")
}

```

```

y <- numeric()
for (i in 1:3) {
  s <- logical(3); s[i] <- TRUE
  y <- c(y, dist.dna(sylvia.clus[, s], p = TRUE))
}
g <- gl(3, length(y) / 3)
library(lattice)
histogram(~ y | g, breaks = 20)

nj.sylvia.K80 <- nj(syl.K80)
nj.sylvia.GG95 <- nj(syl.GG95)
dist.topo(nj.sylvia.K80, nj.sylvia.GG95)

grep("Chamaea", taxa.sylvia, value = TRUE)
f <- function(xx) root(nj(dist.dna(xx, p=TRUE)), "AJ534526")
#tr <- f(sylvia.clus)
## same than:
tr <- root(nj.sylvia.K80, "AJ534526")
nj.boot.sylvia <- boot.phylo(tr, sylvia.clus, f, 200,
                           rooted = TRUE)
nj.boot.codon <- boot.phylo(tr, sylvia.clus, f, 200, 3,
                           rooted = TRUE)

nj.est <- tr
nj.est$tip.label <- taxa.sylvia[tr$tip.label]
plot(nj.est, no.margin = TRUE)
nodelabels(round(nj.boot.sylvia / 200, 2), bg = "white")
add.scale.bar(length = 0.01)
write.tree(nj.est, "sylvia_nj_k80.tre")

write.dna(sylvia.clus, "sylvia.txt")
phyml.sylvia <- phymltest("sylvia.txt", execname = "~/phym1") ##ignore
summary(phyml.sylvia) ##ignore
plot(phyml.sylvia, col = "black") ##ignore
TR <- read.tree("sylvia.txt_phym1_tree.txt") ##ignore
mltr.sylvia <- TR[[28]] ##ignore
mltr.sylvia$tip.label <- taxa.sylvia[mltr.sylvia$tip.label] ##ignore
mltr.sylvia <- root(mltr.sylvia, "Chamaea_fasciata") ##ignore
plot(mltr.sylvia, no.margin = TRUE) ##ignore
add.scale.bar(length = 0.01) ##ignore

tr.ml <- drop.tip(mltr.sylvia, "Chamaea_fasciata")
tr.ml <- drop.tip(nj.est, "Chamaea_fasciata")
res <- vector("list", 9)
for (L in -4:4)
  res[[L + 5]] <- chronopl(tr.ml, 10^L, 12, 16, CV = TRUE)
Lambda <- 10^(-4:4)
CV <- sapply(res, function(x) sum(attr(x, "D2"))))
plot(Lambda, CV / 1e5, log = "x")

sylvia.chrono <- res[[2]]
rts <- attr(sylvia.chrono, "rates")
summary(rts)

```

```

par(mar = c(2, 0, 0, 0))
plot(sylvia.chrono, edge.width = 100*rts, label.offset = .15)
axisPhylo()
write.tree(sylvia.chrono, "sylvia.chrono.tre")

###
### Chapter 6
###

load("sylvia.RData")
nj.est <- read.tree("sylvia_nj_k80.tre")
nj.est <- drop.tip(nj.est, "Chamaea_fasciata")
DF <- sylvia.eco[nj.est$tip.label, ]
table(DF$geo.range, DF$mig.behav)

syl.er <- ace(DF$geo.range, nj.est, type = "d")
syl.sym <- ace(DF$geo.range, nj.est, type="d", model="SYM")
anova(syl.er, syl.sym)

mod <- matrix(0, 3, 3)
mod[2, 1] <- mod[1, 2] <- 1
mod[2, 3] <- mod[3, 2] <- 2
syl.mod <- ace(DF$geo.range, nj.est, type="d", model=mod)

sapply(list(syl.er, syl.sym, syl.mod), AIC)

Q <- syl.mod$index.matrix
diag(Q) <- 0
Q[1, 2] <- Q[2, 1] <- syl.mod$rates[1]
Q[2, 3] <- Q[3, 2] <- syl.mod$rates[2]

#Q[] <- c(0, syl.mod$rates)[Q + 1]
Q[] <- c(0, syl.mod$rates)*(Q + 1)

diag(Q) <- -rowSums(Q)

P <- matexpo(0.05 * Q)
rownames(P) <- c("temp", "temptrop", "trop")
colnames(P) <- rownames(P)

co <- rep("grey", 24)
co[DF$geo.range == "temp"] <- "black"
co[DF$geo.range == "trop"] <- "white"
plot(nj.est, "c", FALSE, no.margin = TRUE, label.offset = 1)
tiplabels(pch = 22, bg = co, cex = 2, adj = 1)
nodelabels(thermo = syl.mod$lik.anc, cex = 0.8,
           piecol = c("black", "grey", "white"))

sylvia.chrono <- read.tree("sylvia.chrono.tre")
yule(sylvia.chrono)
birthdeath(sylvia.chrono)
1 - pchisq(2*(-1.034112 - -1.113822), 1)

```

```

x <- sylvia.eco[sylvia.chrono$tip.label, "geo.range"]
ANC <- ace(x, sylvia.chrono, type = "d", model = mod)
ANC$lik.anc[1:3, ]
anc <- apply(ANC$lik.anc, 1, which.max)
X <- factor(c(x, anc))
yule.cov(sylvia.chrono, ~ X) ## IGNORE
1 / (1 + exp(-(-0.0535529)))
1 / (1 + exp(-(-0.0535529 -1.4608019)))
1 / (1 + exp(-(-0.0535529 -0.9775966)))

fsamp <- function(x) sample(length(x), size = 1, prob = x)
nrep <- 1e3
Pvls <- numeric(nrep)
for (i in 1:nrep) {
  anc <- apply(ANC$lik.anc, 1, fsamp)
  X <- factor(c(x, anc))
  Pvls[i] <- yule.cov(sylvia.chrono, ~ X) ##IGNORE
}
hist(Pvls, freq = FALSE, main = "")
lines(density(Pvls))
#####
gr_sym <- ace(DF$geo.range, nj.est, type="d", model="SYM")
gr_er <- ace(DF$geo.range, nj.est, type="d", model="ER")
gr_ard <- ace(DF$geo.range, nj.est, type="d", model="ARD")

gr_sym$se
gr_sym$rates
gr_er$se
gr_er$rates
gr_ard$se
gr_ard$rates

```