

Bioinformatics Lab 3, Group 3

Roshni Sundaramurthy, Prudhvi Peddmallu, Jiawei Wu, Zijie Feng

2 December 2018

Assignment 1

1.1

We create the mode charachers “x” with the accession numbers. And then we use the mode charachers to read sequences from GenBank.

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

By *clustal*, we make an alignment operation to the sylvia.seq and make them into same length. And check the sequences by doing the same operation by *MAFFT*. And as the result shows two method lead to the same result.

```
#Compare seqs and compare two method
sylvia.clus <- clustal(sylvia.seq)
library(phyloch)
sylvia.mafft <- mafft(sylvia.seq)
identical(sylvia.clus[x, ], sylvia.mafft[x, ])
```

```
## [1] TRUE
```

We keep the original sequences’ species name and the accession number. And erase the original sequences. Then we correct five of their names, because some of them are having wrong names and the others are not matching with the ecological data.

```
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"
taxa.sylvia[5] <- "Sylvia_subcaeruleum"
taxa.sylvia[15] <- "Sylvia_conspicillata"
taxa.sylvia[20] <- "Sylvia_melanocephala"
```

We read the ecological data and show the row name of it. Then we save the “sylvia.clus” “taxa.sylvia” and “sylvia.eco” into a RData file.

```
sylvia.eco <- read.table("sylvia_data.txt")
str(sylvia.eco)
```

```
## 'data.frame': 26 obs. of 3 variables:
## $ mig.dist : int 0 5000 7500 5900 5500 3400 2600 0 0 0 ...
## $ mig.behav: Factor w/ 3 levels "long","resid",...: 2 3 1 1 1 1 1 2 2 2 ...
## $ geo.range: Factor w/ 3 levels "temp","temptrop",...: 3 2 2 2 2 2 2 3 3 3 ...
rownames(sylvia.eco)
```

```
## [1] "Sylvia_abyssinica" "Sylvia_atricapilla" "Sylvia_borin"
```

```
## [4] "Sylvia_nisoria"      "Sylvia_curruca"      "Sylvia_hortensis"
## [7] "Sylvia_crassirostris" "Sylvia_leucomelaena"  "Sylvia_buryi"
## [10] "Sylvia_lugens"       "Sylvia_layardi"      "Sylvia_subcaeruleum"
## [13] "Sylvia_boehmi"       "Sylvia_nana"         "Sylvia_deserti"
## [16] "Sylvia_communis"     "Sylvia_conspicillata" "Sylvia_deserticola"
## [19] "Sylvia_undata"       "Sylvia_sarda"        "Sylvia_balearica"
## [22] "Sylvia_cantillans"   "Sylvia_mystacea"     "Sylvia_melanocephala"
## [25] "Sylvia_rueppelli"    "Sylvia_melanothorax"

save(sylvia.clus, taxa.sylvia, sylvia.eco,
     file = "sylvia.RData")
```

Then we build a function to estimate the NJ tree with the *root* function. And we use *boot.phylo* to get 200 times bootstrap.

```
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)
```

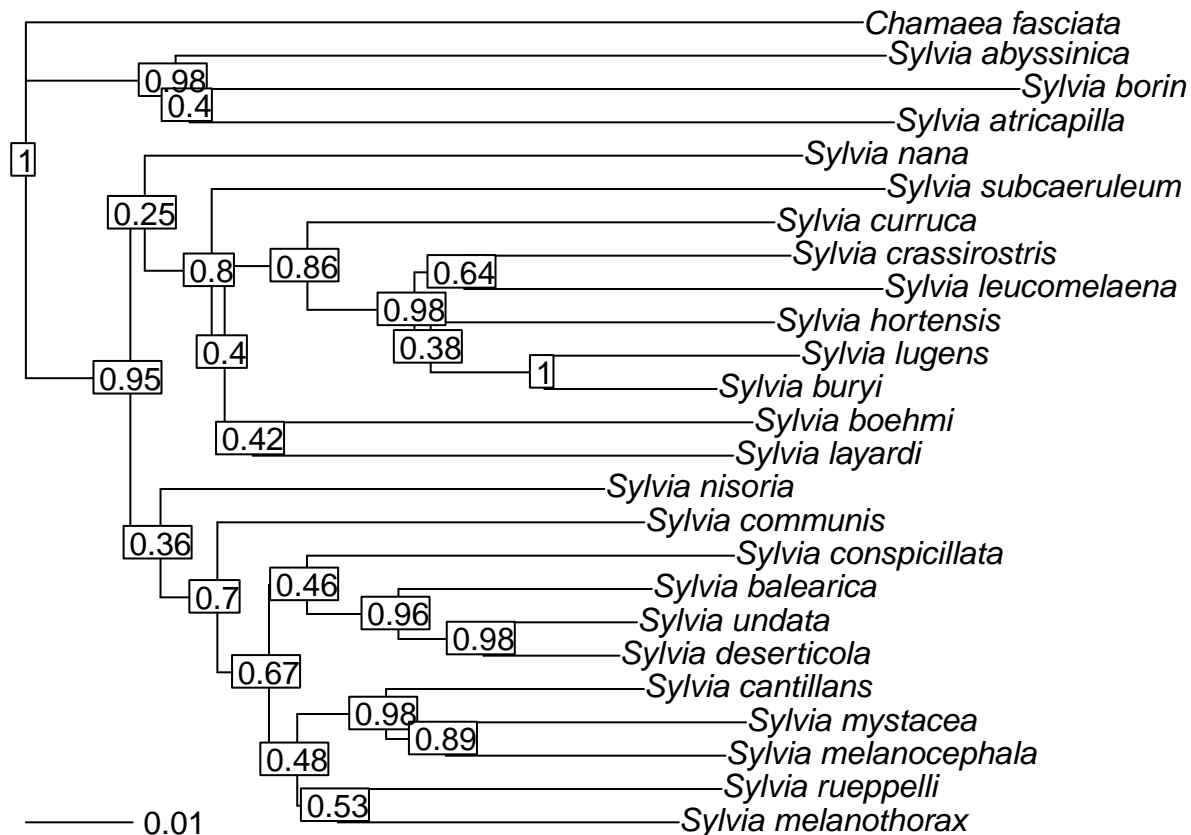
```
##
Running bootstraps:      100 / 200
Running bootstraps:      200 / 200
## Calculating bootstrap values... done.
```

```
nj.boot.sylvia
```

```
## [1] 200 190 50 73 140 134 97 92 106 159 79 85 172 196 193 178 197
## [18] 76 128 195 79 197 199
```

We plot the tree with the bootstrap result on the nodes and the species names with it. The tree are plotted with *plot* and bootstrap result added with *nodeLabels*. And we save the Sylvia warblers phylogeny with fuction *write.tree*.

```
nj.est <- tr
nj.est$tip.label <- taxa.sylvia[tr$tip.label]
layout(matrix(1, 1))
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")
```

Task 2

We read the tree data and remove the “Chamaea_fasciata” from the ecological data, because there is no record in the ecological data. And we also sort the ecological data with the order of tree label. With the `ape::ace` we get the analysis results with three model “ER”(equal-rates model), “SYM”(symmetrical model) and “ARD”(all-rates-different model).

```
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)
```

```
##
##           long resid short
##   temp         0     4     0
##   temptrop      9     0     4
##   trop          0     7     0
```

The result of the three model is shown below.

```
syl.er <- ace(DF$geo.range,nj.est,type="discrete",model = "ER" )
syl.er
```

```
##
##   Ancestral Character Estimation
##
```

```
## Call: ace(x = DF$geo.range, phy = nj.est, type = "discrete", model = "ER")
##
##      Log-likelihood: -22.16276
##
## Rate index matrix:
##           temp temptrop trop
## temp      .         1     1
## temptrop   1         .     1
## trop       1         1     .
##
## Parameter estimates:
##   rate index estimate std-err
##           1   5.4806  1.8448
##
## Scaled likelihoods at the root (type '...$lik.anc' to get them for all nodes):
##           temp temptrop trop
## 0.01809245 0.90972802 0.07217953

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

```
##
##      Ancestral Character Estimation
##
## Call: ace(x = DF$geo.range, phy = nj.est, type = "d", model = "SYM")
##
##      Log-likelihood: -20.38674
##
## Rate index matrix:
##           temp temptrop trop
## temp      .         1     2
## temptrop   1         .     3
## trop       2         3     .
##
## Parameter estimates:
##   rate index estimate std-err
##           1   3.4911  1.9330
##           2   0.0000   NaN
##           3  10.1228  4.3518
##
## Scaled likelihoods at the root (type '...$lik.anc' to get them for all nodes):
##           temp temptrop trop
## 0.001717209 0.777724754 0.220558038

syl.ard <- ace(DF$geo.range, nj.est, type="d", model="ARD")
syl.ard
```

```
##
##      Ancestral Character Estimation
##
## Call: ace(x = DF$geo.range, phy = nj.est, type = "d", model = "ARD")
##
##      Log-likelihood: -21.17252
##
## Rate index matrix:
```

```
##           temp temptrop trop
## temp           .           3    5
## temptrop       1           .    6
## trop           2           4    .
##
## Parameter estimates:
##   rate index estimate std-err
##           1  2.6135  1.9816
##           2  0.0000 11.0679
##           3  0.0000 29.3467
##           4  0.0000    NaN
##           5  0.0000 22.8236
##           6  7.6095  3.8195
##
## Scaled likelihoods at the root (type '...$lik.anc' to get them for all nodes):
##           temp temptrop      trop
##           0         1         0
```

And the estimate values and their stand error is shown as below:

```
syl.er.est<-cbind(syl.er$rates,syl.er$se)
colnames(syl.er.est) <- c("estimate", "standard error")
rownames(syl.er.est) <- c(1)

syl.sym.est<-cbind(syl.sym$rates,syl.sym$se)
colnames(syl.sym.est) <- c("estimate", "standard error")
rownames(syl.sym.est) <- c(1:3)

syl.ard.est<-cbind(syl.ard$rates,syl.ard$se)
colnames(syl.ard.est) <- c("estimate", "standard error")
rownames(syl.ard.est) <- c(1:6)

paste("the estimate value and stand error of er model is")
```

```
## [1] "the estimate value and stand error of er model is"
syl.er.est
```

```
##   estimate standard error
## 1 5.480571      1.844787
paste("the estimate value and stand error of sym model is")
```

```
## [1] "the estimate value and stand error of sym model is"
syl.sym.est
```

```
##   estimate standard error
## 1  3.49114      1.933031
## 2  0.00000           NaN
## 3 10.12285      4.351795
paste("the estimate value and stand error of ard model is")
```

```
## [1] "the estimate value and stand error of ard model is"
syl.ard.est
```

```
##   estimate standard error
```

```
## 1 2.613538      1.981600
## 2 0.000000      11.067898
## 3 0.000000      29.346710
## 4 0.000000      NaN
## 5 0.000000      22.823571
## 6 7.609460      3.819466
```

```
sapply(list(syl.er, syl.sym, syl.ard), AIC)
```

```
## [1] 46.32552 46.77348 54.34504
```

From the AIC values, we can see the “ard” model has the largest AIC which is not a good model in this case. From the index matrix, we can get the probabilities matrix showing as follow.

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

```
##           temp  temptrop    trop
## temp      0.75056430 0.1868287 0.06260704
## temptrop  0.18682866 0.6263427 0.18682866
## trop      0.06260704 0.1868287 0.75056430
```

In the “er” model, we can see the we can see the temptrop state is the least stable one, and the trop and temp state have the same stability. And the evolve between temp, temptrop and temptrop, trop are totally the same. And it is unlikely to evolve between trop and temp.

```
Q <- syl.sym$index.matrix
diag(Q) <- 0
Q[1, 2] <- Q[2, 1] <- syl.sym$rates[1]
Q[2, 3] <- Q[3, 2] <- syl.sym$rates[3]
Q[1, 3] <- Q[3, 1] <- syl.sym$rates[2]
# 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)
P
```

```
##           temp  temptrop    trop
## temp      0.85095138 0.1201851 0.02886357
## temptrop  0.12018505 0.5861572 0.29365773
## trop      0.02886357 0.2936577 0.67747871
```

In the “SYM” model, we can see the temptrop state is still the least stable one, and it is more likely to evolve to the trop state. And we can see the temp state are the most stable one and it are mostly evolve from the temptrop state. And the trop state species are more likely to evolve to temptrop state species.

```

Q <- syl.ard$index.matrix
diag(Q) <- 0
Q[1, 2] <- syl.ard$rates[1]
Q[1, 3] <- syl.ard$rates[2]
Q[2, 1] <- syl.ard$rates[3]
Q[2, 3] <- syl.ard$rates[4]
Q[3, 1] <- syl.ard$rates[5]
Q[3, 2] <- syl.ard$rates[6]
# Q[] <- c(0, syl.mod$rates)[Q + 1]
diag(Q) <- -rowSums(Q)

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

```

```

##           temp  temptrop   trop
## temp      0.8775013 0.1224987 0.000000
## temptrop  0.0000000 1.0000000 0.000000
## trop      0.0000000 0.3164620 0.683538

```

In the “ARD” model, we can see the temptrop one is the most stable one, and it won’t evolve to other states. The temp is the second stable one, which only evolve to temptrop state. And the trop state can only evolve to temptrop state with a higher propability. But trop and temp have no chance to evolve to each other.

Assignment 2: Exploring carnivores dataset (carni70)

2.1

```

library(ape)
library(ade4)

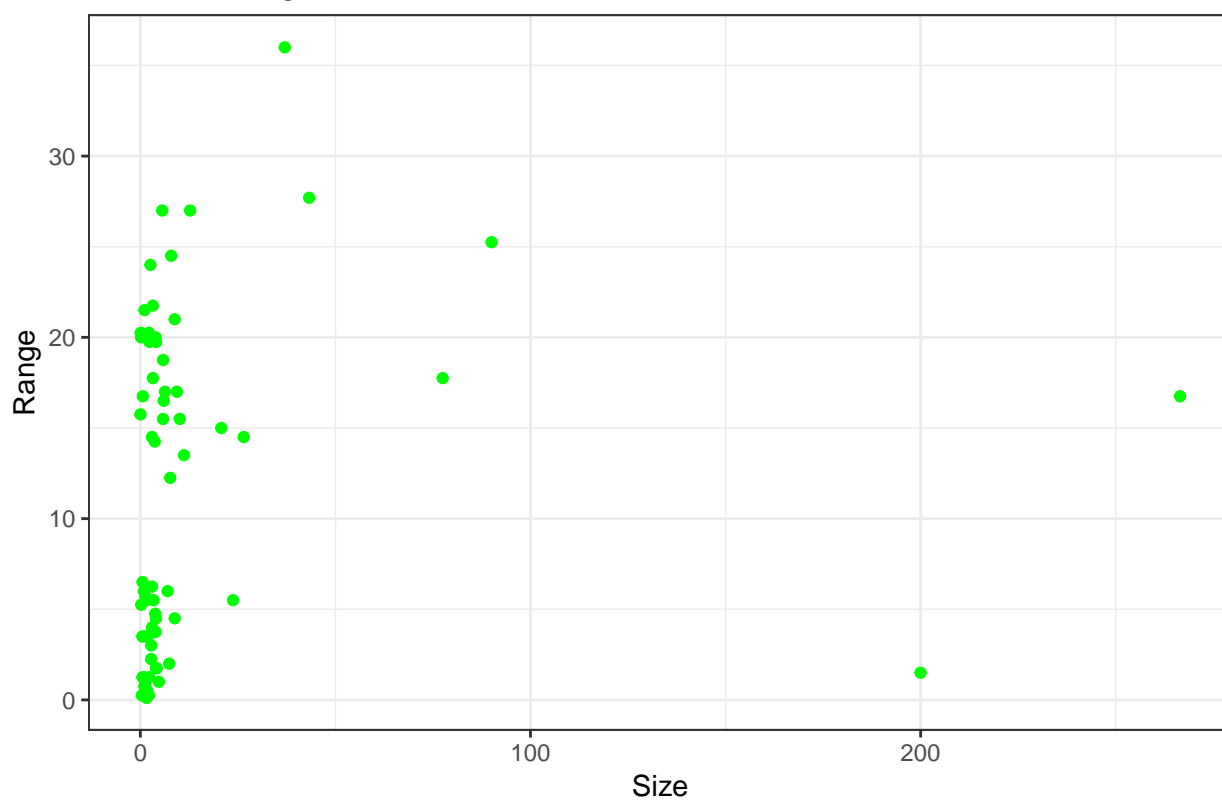
data(carni70) #Phylogeny and quantitative traits of carnivora
rownames(carni70$tab) <- gsub("_", ".", rownames(carni70$tab))

# look at two variables
size <- carni70$tab$size
range <- carni70$tab$range

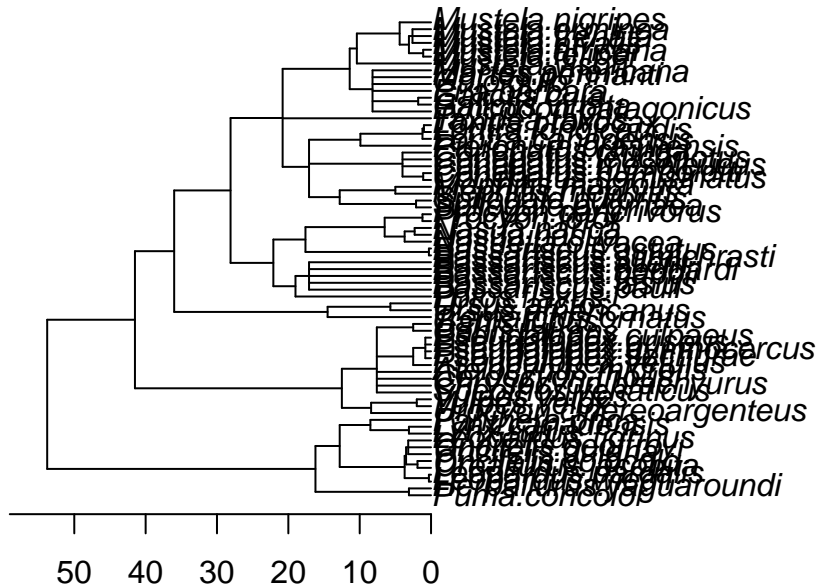
#plot the size and range
library(ggplot2)
ggdat <- data.frame(Size=size, Range=range)
ggplot(ggdat)+geom_point(aes(Size,Range),colour="green")+
  ggtitle("Size Vs Range of carni70 data")+theme_bw()

```

Size Vs Range of carni70 data



Phylo tree



Analysis:

The `carni70` data is a list containing the following 2 objects:

tree: It is a character string giving the phylogenetic tree in Newick format. Branch lengths are expressed as divergence times (millions of years).

tab: It is a data frame with 70 species and two traits: Size - body size in kg Range - geographic range size in km

It describes the phylogeny of 70 carnívora as reported by Diniz-Filho and Torres (2002). It also gives the geographic range size and body size corresponding to these 70 species.

The relationship between size and range is plotted. It is observed that the species with body size within 50 kg have geographic range size of 0-30 km. There exists some outliers in size of 200 and 280. And from the tree, we observed that some of the species get divided too early based on their phylogeny.

2.2

The two traits (size and range) with a number of different phylogenetic comparative models were analyzed using the following R packages mvMORPH, mvSLOUCH, ouch and slouch.

2.2.1 Both traits evolve as independent Brownian motions

```
library(mvMORPH)
BM1<-mvBM(tree=tre,data=ggdat,model="BM1",param = list(constraint="diagonal"))
```

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

The mvBM function allows the fitting of multivariate multiple rates of evolution under a Brownian Motion model. This function can also fit constrained models. Here a phylo object “tre” is used with the “BM1” model (for a unique rate of evolution per trait, i.e, for traits size and range). The “constraint” argument in the “param” list allows us to compute the joint likelihood for each trait by assuming they evolved independently by specifying “” constraint=“diagonal” “”.

2.2.2 The traits evolve as a correlated Brownian motion

```
BM2<-mvBM(tree=tre,data=ggdat,model="BM1",param = list(constraint="correlation",
                                                         diagnostic=FALSE, echo=FALSE))
```

```
## "shared","variance","correlation" and "proportional" can be used only with BMM model
## species in the matrix are assumed to be in the same order as in the phylogeny, otherwise specify row
## successful convergence of the optimizer
## a reliable solution has been reached
##
## -- Summary results for multiple rate BM1 model --
## LogLikelihood:    -588.7853
## AIC:              1187.571
## AICc:             1188.018
## 5 parameters
##
## Estimated rate matrix
## -----
##           Size      Range
## Size  73.703348  3.579341
## Range  3.579341  13.545242
##
## Estimated root state
## -----
##           Size      Range
## theta: 38.43947  13.78439
```

The same function is used as above. But a multiple rate matrix model is used for which the correlations are identical but with different variances between groups by specifying “param=list(constraint=“correlation”)” in the function.

Likelihood Ratio Test (LRT)

This function compares the fit of two models of trait evolution with a loglikelihood-ratio statistic.

```
LRT(BM1,BM2)
```

```
## -- Log-likelihood Ratio Test --
## Model BM1 versus BM1 diagonal
## Number of degrees of freedom : 1
## LRT statistic: 0.9041323 p-value: 0.3416761
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.2.3 Both traits evolve as independent Ornstein-Uhlenbeck processes

```
#library(slouch)
# Fitting the Ornstein Uhlenbeck with multiple optimums
OU<-mvOU(tre, data=ggdat, model="OUM",param = list(constraint="diagonal"))
```

```
## species in the matrix are assumed to be in the same order as in the phylogeny, otherwise specify row
## No selective regimes mapped on the tree, only a OU1 model could be estimated
## successful convergence of the optimizer
## a reliable solution has been reached
##
## -- Summary results --
## LogLikelihood:    -573.7985
## AIC:             1163.597
## AICc:            1164.696
## 8 parameters
##
## Estimated theta values
## -----
##           Size      Range
## OU1 37.13101 11.41749
##
## ML alpha values
## -----
##           Size      Range
## Size  0.006381174 -0.00373479
## Range -0.003734790  0.23496685
##
## ML sigma values
## -----
##           Size      Range
## Size  79.978292  6.125847
## Range  6.125847 38.498785
```

```
# to get the evolutionary correlations between traits
cov2cor(stationary(OU))
```

```
##           Size      Range
## Size  1.0000000 0.1710922
## Range 0.1710922 1.0000000
```

The OU model describes how a trait evolves away from its optimal value (h), and is pulled back towards the optimum with a strength corresponding to α (or ‘rubberband’ parameter). Species measurement errors or dispersions is included in this model. The phylo tree “tre” is used with ggdat dataframe consisting size and range values. Model OUM is specified for a multiple selective regime model. The estimated ancestral states with size is 37.1304 and for range 11.417.

2.2.4 The traits evolve as a bivariate Ornstein-Uhlenbeck process

```
library(mvSLOUCH)
#converting the phylo tree to ouch tree
newtree<-ape2ouch(tre, scale = TRUE, branch.lengths = tre$edge.length)

#Estimate parameters under a (multivariate) OU model of evolution
ouchModel(newtree,data=as.matrix(carni70$tab))
```

The ouchModel function uses maximum likelihood to fit parameters of a multivariate OU model evolving on the phylogeny. The runtime of this function is quite long (took almost 40 minutes). This function estimates the parameters of the following multivariate SDE,

$$dY(t) = -A(Y - \psi(t))dt + \sum dW(t), Y(0) = Y_0$$

on a phylogenetic tree. $Y(0)$ is assumed to be equal to the root value of Ψ .

```
[1] "Starting point of heuristic search procedure : "
      Astart      A_2      A_3      Aend      Syystart      Syy_2      Syyend
-1.2545811 -0.1577980 -0.5123168  0.4130209  0.5642563  0.7260660 -0.4151022

$`FinalFound`$`HeuristicSearchPointFinalFind`
      Astart      A_2      A_3      Aend
-1.7556069  2.8333436  0.7160068  1.1023159
      Syystart      Syy_2      Syyend      LogLik
10.5178266 118.0367837  3.7039063 -607.0847467

$`FinalFound`$ParamsInModel
$`FinalFound`$ParamsInModel$`A`
      size      range
size 408517.83181 -0.5004939
range 11.06714 9.7645770

$`FinalFound`$ParamsInModel$mPsi
      reg.1
size 14.28771
range 11.11701

$`FinalFound`$ParamsInModel$mPsi0
      [,1]
size 0
range 0

$`FinalFound`$ParamsInModel$vy0
      [,1]
size 14.28771
range 11.11701

$`FinalFound`$ParamSummary$`phyl.half.life`$half.lives
      [,1]      [,2]
eigenvalues 4.085178e+05 9.76459051
half.life 1.696737e-06 0.07098579
%treeheight 1.696737e-04 7.09857909

$`FinalFound`$ParamSummary$`phyl.half.life`$half.life.lowerbounds
[1] 1.696696e-06

$`FinalFound`$ParamSummary$cov.matrix
      size      range
size 1672.7519288 -0.0334796
range -0.0334796 84.4663755

$`FinalFound`$ParamSummary$corr.matrix
      size      range
size 1.000000e+00 -8.906819e-05
range -8.906819e-05 1.000000e+00
```

```

$`FinalFound`$ParamSummary$LogLik
[1] -607.0847

$`FinalFound`$ParamSummary$dof
[1] 9

$`FinalFound`$ParamSummary$m2loglik
[1] 1214.169

$`FinalFound`$ParamSummary$aic
[1] 1232.169

$`FinalFound`$ParamSummary$aic.c
[1] 1233.554

```

We used `ape2ouch` function to translate `ape`'s phylo representation of a phylogenetic tree into `ouch`'s `ouchtree` representation and the tree's depth is scaled to 1. The `newtree` (`ouchtree`) is used for `ouchmodel` with dataset of size and range as matrix. The final found values for `Y0` is 14.287 for body size and 11.1170 for range. The correlation matrix exhibits the same value ($-8.906819e^{-05}$) for both traits.

2.2.5 size evolves as a Brownian motion and range as an Ornstein-Uhlenbeck process adapting to it

```

#size&range ouchm
# Estimate parameters under a (multivariate) OUBM model of evolution
mvslouchModel(newtree, data = as.matrix(ggdat), kY=1)

```

The estimation function `mvslouchModel` uses maximum likelihood to fit parameters of a multivariate OUBM model evolving on the phylogeny. This function estimates the parameters of the following multivariate SDE,

$$dY(t) = -A(Y(t)) - (\psi(t) - A - 1BX(t))dt + \sum_{yy} dB(t), Y(0) = Y_0$$

$$dX(t) = \sum_{xx} dB(t), X(0) = X_0,$$

on a phylogenetic tree. $Y(0)$ is assumed to be equal to $- \text{solve}(A)BX(0)$ plus the root value of Ψ .

```

[1] "Starting point of heuristic search procedure : "
      A      Syy
0.1665597 0.2653246
$`FinalFound`
$`FinalFound`$`HeuristicSearchPointFinalFind`
      A      Syy      LogLik
-5.147928  5.144370 -656.808469

```

```

$`FinalFound`$ParamSummary$cov.matrix
      Size      Range
Size 37315.320 2028.6625
Range 2028.663 678.3047

```

```

$`FinalFound`$ParamSummary$corr.matrix
      Size      Range
Size 1.0000000 0.4032309
Range 0.4032309 1.0000000

```

```

$`FinalFound`$ParamSummary$LogLik
[1] -656.8085

```

```

$`FinalFound`$ParamSummary$dof
[1] 6

```

```

$`FinalFound`$ParamSummary$m2loglik
[1] 1313.617

```

```

$`FinalFound`$ParamSummary$aic
[1] 1325.617

```

```

$`FinalFound`$ParamSummary$aic.c
[1] 1326.249

```

```

$`FinalFound`$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
Size -30.39972      -5.99316   18.4134

$`FinalFound`$ParamSummary$confidence.interval$`regression.summary`$mPsi.regression.confidence.interval
  Lower.end Estimated.Point Upper.end
Size -57973.26      -11422.21  35128.84

```

The same ouchtree is used here for mvslouchModel with size and range dataset as matrix. KY denotes number of “Y” (response) variables. The MaxLikFound (point with the highest likelihood) in the result seems to be “Same as final found”. This means that the value is the same as found in final point (FinalFound - the point where the search procedure stopped).

We illustrated the usefulness of the method using the phylogeny of the carni70 data together with data on the body size-range relationship to estimate the effect of phylogenetic history.

The table below shows the summary of comparison of model fit to the body size and range.

Model	k	lnLik	AIC	AICc
Independent Brownian motions	4	-589.2373	1186.475	1186.771
Correlated Brownian motion	5	-588.7853	1187.571	1188.018
Independent Ornstein-Uhlenbeck processes	8	-573.7985	1163.597	1164.696
Bivariate Ornstein-Uhlenbeck process	9	-607.0847	1232.169	1233.554
Size evolves as a Brownian motion and Range as an Ornstein-Uhlenbeck process	6	-656.8085	1325.617	1326.249

where

k = Number of model parameters, lnLik = Natural logarithm of Maximum likelihood AICc = Sample size corrected Akaike Information Criterion

As expected, AIC criterion shows that the generating “*Independent Ornstein-Uhlenbeck processes*” we simulated is preferred over all other fitted models. The log likelihood value seems to be larger for this model. The lnLik value is -573.7985 with AIC 1163.597 (lowest than all other models). Hence, this model seems to fit better to the data. Other models may overestimate or underestimate the influence of phylogeny respectively.