

# ecoeko-hmsc

```
library(ecoeko)
library(here)
library(R.matlab)
library(grDevices)
library(knitr)
library(Hmsc)
library(beanplot)
```

## 1 Overview

The goal of this vignette is to illustrate how to use HMSC (Heirarchical Models of Species Communities; Ovaskainen et al. 2017; Tikhonov et al. 2020) to analyze eco-evolutionary community and metacommunity data. The eco-evolutionary data is generated using a multi-species model of growth and competition (a Leslie-Gower model, which is a discrete-time Lotka-Volterra model). In this model, we also include the possibility that the population growth rate can evolve (using a model of evolutionary rescue introduced by Gomulkiewicz & Holt 1995). The model is as follows:

$$N_{i,t+1} = \frac{\hat{W} e^{-\left(\frac{(w+(1-h^2)P)}{P+w}(E-x_t)\right)^2}}{1 + \alpha_{ii} N_{i,t} + \sum_j \alpha_{ij} N_{j,t}} N_{i,t}$$

where  $N_i, t$  is the population size of a species at time  $t$ ,

$$\hat{W}$$

is calculated as:

$$\hat{W} = W_{max} \sqrt{\left(\frac{w}{P+w}\right)}$$

,

$W_{max}$  is the species' maximum per-capita growth rate,  $w$  is the width of the Gaussian fitness function (which determines the strength of selection),  $P$  is the width of the distribution of the phenotype  $x$ ,  $h^2$  is the heritability of the trait  $x$ ,  $E$  is the local environmental optimum trait value,  $x_t$  is the trait value of the species at time  $t$ ,  $\alpha_{ii}$  is the intraspecific competition coefficient and  $\alpha_{ij}$  is the interspecific competition coefficient.

I consider the evolution of multiple species ( $j = 15$ ) in a landscape of patches ( $i = 50$ ). The patches have values of an environmental property  $E$  (drawn from a uniform distribution ranging from 0 to 1) that determines the local optimum phenotype  $E$ , i.e. where species experience their absolute fitness  $W_{max}$ , and patches also have spatial locations  $X$  and  $Y$  (both drawn from  $U(0, 1)$ ). The patches thus have a connectivity matrix  $\mathbf{D}$  (here given by their Euclidean distance), as well as a connectivity matrix  $\mathbf{C}$  that is a Gaussian function of  $\mathbf{D}$  and a species dispersal rate  $d_j$ .

## 2 Simulation

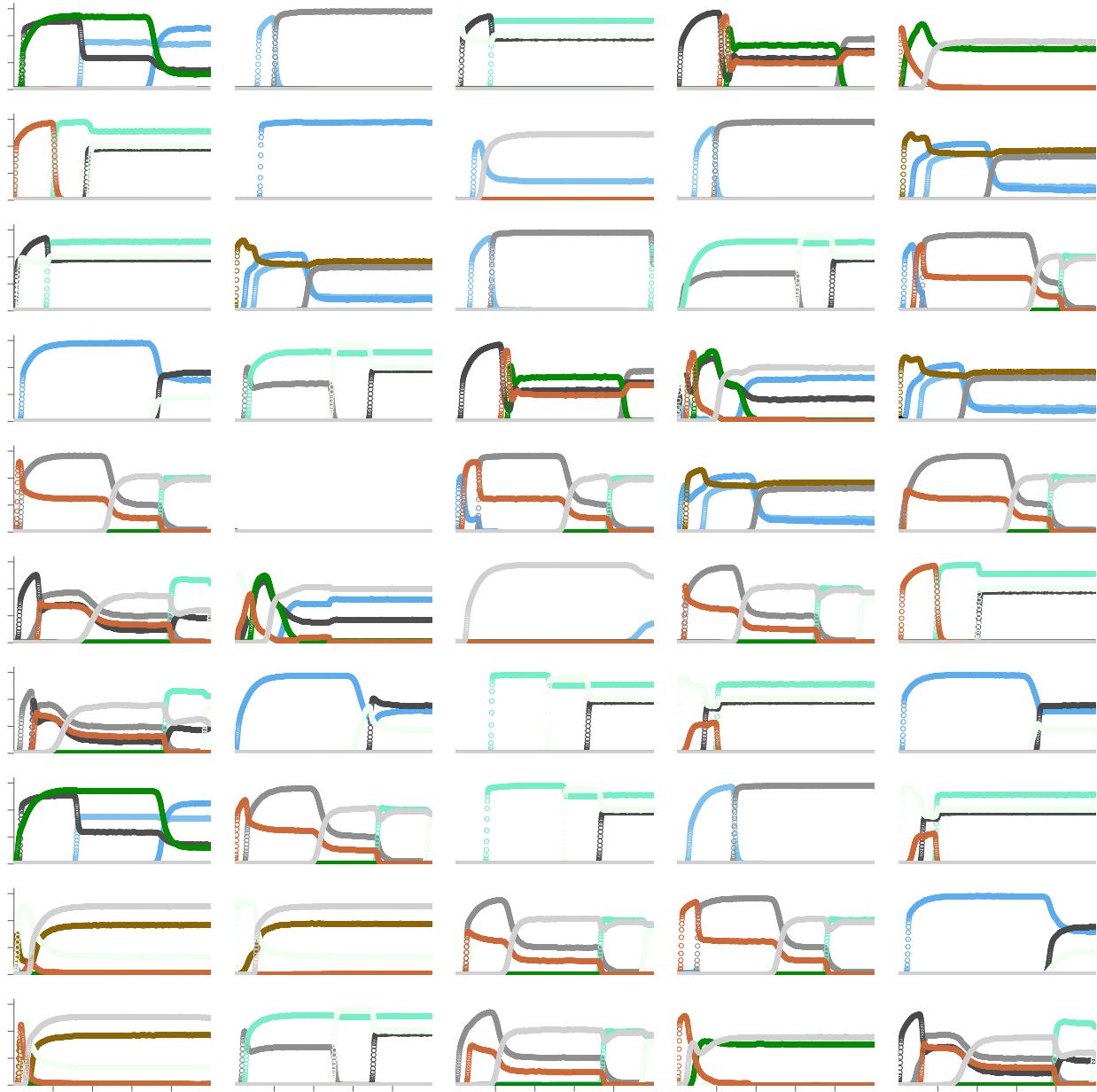
The simulation is run in MATLAB (the code for this can be found at [https://github.com/jhpantel/HMSC\\_ecoeco/code/simulation/metacom\\_evol\\_sim\\_same\\_init.m](https://github.com/jhpantel/HMSC_ecoeco/code/simulation/metacom_evol_sim_same_init.m)), and generates a  $50 \times 15 \times 1000$  array of abundance values (for 50 patches, 15 species, at 1000 time points). In this example, we consider  $h^2 = 0.1$  and  $d = 10^{-3}$  for all 15 species.

We can see the dynamics of the simulation here:

```
datapath <- here::here("inst", "extdata", "hmse")
result <- "h_01_d_minus3" # Enter desired simulation conditions here
res <- R.matlab::readMat(paste(datapath, "/", result, "_res.mat", sep=""))
N <- res$N

## Plot of Patch 1 community dynamics over time
cl <- grDevices::colors(distinct = TRUE)
mycols2 <- sample(cl, 15)

par(mfrow=(c(10,5)))
par(mar=c(1,1,1,1))
for(j in 1:50){
  plot(res$N[j,1,], col=mycols2[1], ylim=c(0,800), axes=FALSE, type="n", ylab=NA, xlab=NA, xaxs="i", yaxs="i")
  if (j %in% c(1,6,11,16,21,26,31,36,41,46))
    axis(2,col="grey40",col.axis="grey20",at = c(0,250,500,750,1000))
  if (j %in% c(46,47,48,49,50))
    axis(1,col="grey40",col.axis="grey20",at = c(0,200,400,600,800))
  for(i in 1:15){
    points(res$N[j,i,], col=mycols2[i])
  }
}
```



The plots are time series of population size (y-axis) over time (x-axis) for all species in all 50 patches. The plots are ordered by rows, from lowest to highest E values and each species is plotted with a unique color. Traits are evolving for all species as well, which we can visualize here:

```

par(mfrow=c(10,5))
par(mar=c(1,1,1,1))
for(j in 1:50){
  plot(res$xt[j,1],col=mycols2[1],ylim=c(-2,1),axes=FALSE,type="n",ylab=NA,xlab=NA,xaxs="i",yaxs="i")
  if (j %in% c(1,6,11,16,21,26,31,36,41,46))
    axis(2,col="grey40",col.axis="grey20",at = c(-2,-1.5,-1,-.5,0,.5,1))
  if (j %in% c(46,47,48,49,50))
    axis(1,col="grey40",col.axis="grey20",at = c(0,250,500,750,1000))
  for(i in 1:15){
    points(res$xt[j,i],col=mycols2[i])
  }
}

```



### 3 HMSC Analysis

The goal of our analysis is to estimate the relative effects of intrinsic population dynamics, environmental drivers, spatial structure, and trait evolution for driving community composition in the metacommunity. We use HMSC to build a linear Bayesian hierarchical model.

To analyze our data with HMSC, we first re-arrange our data to fit the structure needed for input into HMSC.

```

# Arrange all data for HMSC
## site-by-species abundance data as a time series
N <- res$N
Y <- array(NA,dim=c(50000,15),dimnames=list(NULL,c("y1","y2","y3","y4","y5","y6","y7","y8","y9","y10","y11","y12","y13","y14","y15")))
count <- seq(1,50000,by=50)
for(i in 1:1000){
  Y[(count[i]:(count[i]+49)),] <- N[,,i]
}

## site-by-species trait data as a time series
trait_x <- res$xt
trait_init <- trait_x[,1] # Eliminate my initial trait value of 9999 for absent species
trait_init[trait_init == 9999] <- NaN
trait_x[,1] <- trait_init
## site-by-species CHANGE in trait value as a time series
delta_x <- array(NA,c(50,15,999),dimnames=list(NULL,c("x1","x2","x3","x4","x5","x6","x7","x8","x9","x10","x11","x12","x13","x14","x15")))
count <- seq(1,50,by=50)
for(i in 2:1000){
  delta_x[,,i-1] <- trait_x[,,i] - trait_x[,,i-1]
}
delta_x_time <- array(NA,dim=c(49950,15),dimnames=list(NULL,c("x1","x2","x3","x4","x5","x6","x7","x8","x9","x10","x11","x12","x13","x14","x15")))
count <- seq(1,49950,by=50)
for(i in 1:999){
  delta_x_time[(count[i]:(count[i]+49)),] <- delta_x[,,i]
}
delta_x_time[is.nan(delta_x_time)] <- 0

## Random effects of site and year
Random <- array(NA,dim=c(50000,2),dimnames=list(NULL,c("site","year")))
Random[,1] <- rep(1:50,1000)
Random[,2] <- rep(1:1000,each=50)
Random <- as.data.frame(Random)

## Fixed effects: environment, site abundance the year before, species trait value the year before
X <- Y[Random$year != 1000,] # Use abundance in years 1-999 as fixed effects - the abundance the year before
Y <- Y[Random$year != 1,] # Trim Y to only include years 2-1000

Random <- Random[Random$year != 1,] # Trim Random to only include years 2-1000
Random$site <- as.factor(Random$site)
Random$year <- as.factor(Random$year)
Random$sample <- as.factor(1:nrow(Random))

## Random effects
xy <- res$xy
rownames(xy) <- 1:50
rL.spatial <- Hmsc::HmscRandomLevel(sData = xy)
rL.spatial = Hmsc::setPriors(rL.spatial,nfMin=1,nfMax=5)

rL.time <- Hmsc::HmscRandomLevel(units = unique(Random$year))
rL.time = Hmsc::setPriors(rL.time,nfMin=1,nfMax=5)

## Fixed effects: environment
env_hold <- array(NA,dim=c(dim(Y)[1],dim(res$E)[2]))

```

```

for(i in 1:dim(res$E)[2]) { # for each environmental variable
  env_hold[,i] <- rep(res$E[,i], 999)
  colnames(env_hold)[i] <- paste("env", i, sep="")
}

## Interaction effects: site abundance the year before X change in species trait value
X_delta_x <- X * delta_x_time
colnames(X_delta_x) <- paste("x", colnames(X_delta_x), sep="")

# When h2 = 0
#X <- cbind(X, env_hold)
#X <- scale(X)
#X <- cbind(X, MEM_hold)

# When h2 != 0
X <- cbind(X, env_hold, delta_x_time, X_delta_x)
X <- scale(X)
#X <- cbind(X, MEM_hold)
X <- as.data.frame(X)

m <- Hmsc::Hmsc(Y=Y, XData=X, studyDesign=Random, ranLevels=list("site"=rL.spatial, "year"=rL.time), distr="")
# The model takes a long time to run. The results included here are not reliable, but demonstrate how to
m.1 <- Hmsc::sampleMcmc(m, 10, transient=5, nChains=2, nParallel=2, verbose=5)

```

Now that the model is fit, we can analyze the results.

```

#Explanatory power
preds = Hmsc::computePredictedValues(m.1)
MF = Hmsc::evaluateModelFit(hM=m.1, predY=preds)

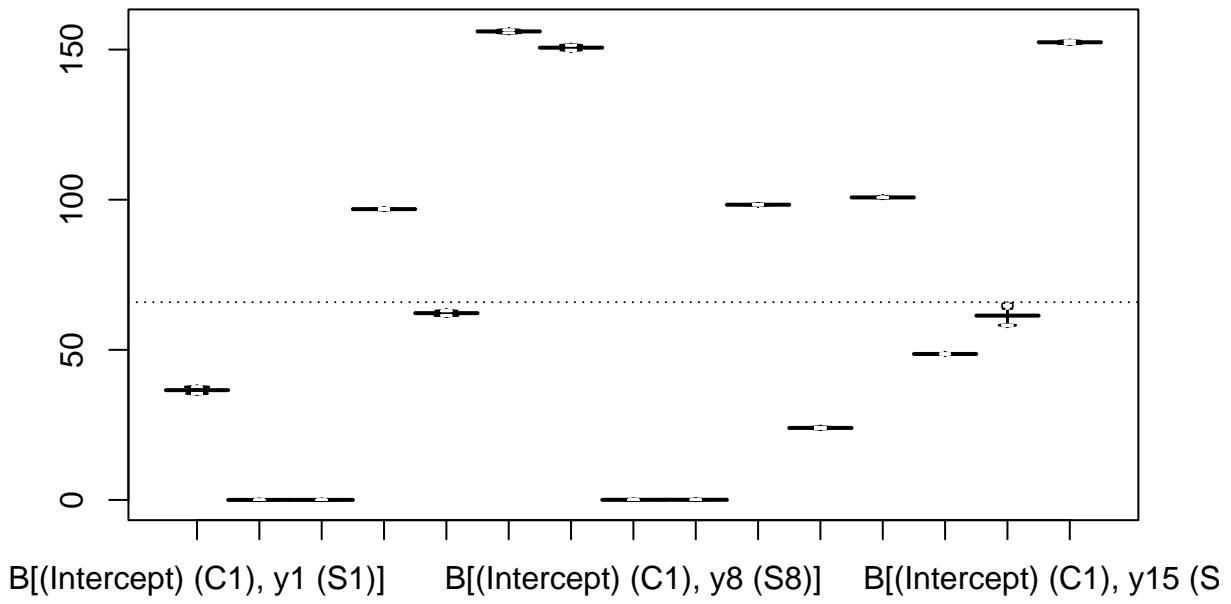
```

We should also evaluate the chains, for convergence. Then we will inspect the results.

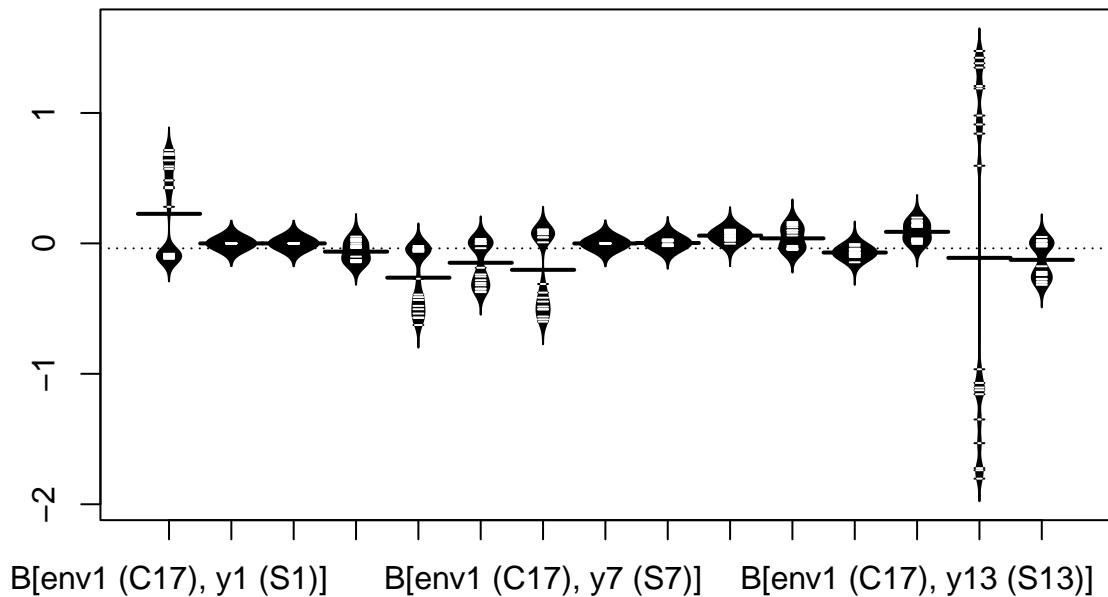
```

m.post = Hmsc::convertToCodaObject(m.1)
m.df <- as.data.frame(rbind(m.post$Beta[[1]], m.post$Beta[[2]]))
## intercept
B.I <- m.df[, seq(1, 705, by=47)]
beanplot::beanplot(B.I, log="")

```



```
## Fixed effects: environment
B.E <- m.df[,seq(17,705,by=47)]
beanplot::beanplot(B.E)
```



## References

- Gomulkiewicz, R. & Holt, R. D. When does evolution by natural selection prevent extinction? *Evolution* 49, 201–207 (1995)
- Ovaskainen, O., Tikhonov, G., Norberg, A., Guillaume Blanchet, F., Duan, L., Dunson, D., Roslin, T. and Abrego, N. (2017). How to make more out of community data? A conceptual framework and its implementation as models and software. *Ecology Letters*, 20(5), 561-576.
- Tikhonov, G., Opedal, Ø. H., Abrego, N., Lehikoinen, A., de Jonge, M. M., Oksanen, J., & Ovaskainen, O. (2020). Joint species distribution modelling with the R-package Hmsc. *Methods in Ecology and Evolution*, 11(3), 442-447.