Final Project

Emerson C. Barbosa Júnior

2020/12/01

# Final Project of R discipline

## Chapter I - Introduction to Agent-Based Model (INCOMPLETE)

In this phase, I am carrying packages and data to use in next steps.

install.packages("Rcpp")  
install.packages("raster")  
install.packages("plyr")  
install.packages("stringr")  
install.packages("wallace")  
install.packages("spocc")  
install.packages("spThin")  
install.packages("dismo")  
install.packages("rgeos")  
install.packages("ENMeval")  
install.packages("dplyr")

library(Rcpp)  
library(raster)  
library(plyr)  
library(stringr)  
library(wallace)  
library(spocc)  
library(spThin)  
library(dismo)  
library(rgeos)  
library(ENMeval)  
library(dplyr)  
  
mydata <- read.csv("C:/Users/Emerson Júnior/Dropbox/Coisas minhas/Mestrado e doutorado em Ecologia - UFBA/Doutorado/final\_project\_R/Database.csv", header = TRUE, sep = ",")  
#View(mydata)

Below, I began to create a agent to my agent-based model, but this process is incomplete. I had focused more a chapter two because there are much steps to reach the basic of expected in model.

##################### SETUP #########################  
Species\_1 <- data.frame(AgentN = 1,  
 color = "red",  
 size = sqrt(rnorm(1, 2, 0.5) ^ 2),  
 shape = "type\_animal",  
 energy = runif(1, 1, 10),  
 xcor = runif(1, 1, 180),  
 ycor = runif(1, 1, 180),   
 age = runif(1, 1, 50))  
  
N\_pop <- function(number = 10) {  
 return(number)  
}  
  
for (i in 2:N\_pop()) {  
 Specie\_1\_ind <- data.frame(AgentN = i,  
 color = "red",  
 size = sqrt(rnorm(1, 2, 0.5) ^ 2),  
 shape = "type\_animal",  
 energy = runif(1, 1, 10),  
 xcor = runif(1, 1, 180),  
 ycor = runif(1, 1, 180),   
 age = runif(1, 1, 50))  
 Species\_1 <- rbind(Species\_1, Specie\_1\_ind)  
   
}  
#View(Species\_1)  
Species\_1

## AgentN color size shape energy xcor ycor age  
## 1 1 red 1.546580 type\_animal 7.776997 105.47737 16.331024 4.855724  
## 2 2 red 1.015290 type\_animal 7.864603 135.18826 123.019459 4.173824  
## 3 3 red 1.953647 type\_animal 6.052484 106.74817 148.343189 23.071368  
## 4 4 red 2.291241 type\_animal 5.377221 157.78839 27.571804 28.880213  
## 5 5 red 2.139053 type\_animal 2.348171 162.63986 81.200457 30.464669  
## 6 6 red 1.437473 type\_animal 3.201026 67.43827 133.135675 47.748202  
## 7 7 red 1.710015 type\_animal 7.142505 153.07144 169.885048 17.606831  
## 8 8 red 1.692088 type\_animal 4.223519 87.94383 64.185095 3.386124  
## 9 9 red 1.550257 type\_animal 5.773320 51.71306 85.781022 35.079741  
## 10 10 red 2.028868 type\_animal 9.755212 12.39986 5.384995 36.427733

Species\_2 <- data.frame(AgentN = 1,  
 color = "blue",  
 size = sqrt(rnorm(1, 5, 0.5) ^ 2),  
 shape = "type\_animal",  
 energy = runif(1, 1, 10),  
 xcor = runif(1, 1, 180),  
 ycor = runif(1, 1, 180),   
 age = runif(1, 1, 50))  
  
  
for (i in 2:N\_pop()) {  
 Specie\_2\_ind <- data.frame(AgentN = i,  
 color = "blue",  
 size = sqrt(rnorm(1, 5, 0.5) ^ 2),  
 shape = "type\_animal",  
 energy = runif(1, 1, 10),  
 xcor = runif(1, 1, 180),  
 ycor = runif(1, 1, 180),   
 age = runif(1, 1, 50))  
 Species\_2 <- rbind(Species\_2, Specie\_2\_ind)  
}  
#View(Species\_2)  
Species\_2

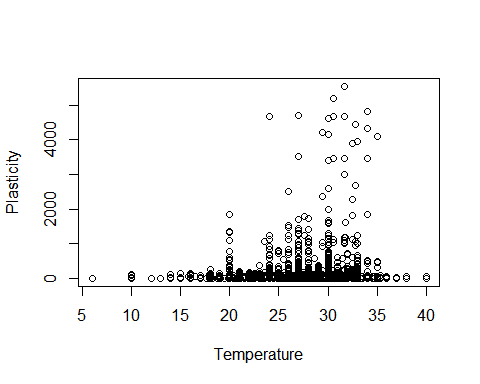
## AgentN color size shape energy xcor ycor age  
## 1 1 blue 4.917744 type\_animal 7.931092 164.28191 138.759835 32.261416  
## 2 2 blue 6.011658 type\_animal 5.518768 124.30037 17.109025 20.878448  
## 3 3 blue 4.517960 type\_animal 6.944041 133.49019 33.980388 3.063523  
## 4 4 blue 5.198809 type\_animal 2.764971 58.15848 2.638907 4.235987  
## 5 5 blue 5.171307 type\_animal 2.352959 73.14082 33.163225 7.880869  
## 6 6 blue 4.825896 type\_animal 1.260499 87.50348 14.055082 6.697176  
## 7 7 blue 5.325055 type\_animal 1.601965 111.76982 33.616398 24.773089  
## 8 8 blue 5.238589 type\_animal 6.829879 34.69927 119.322637 7.115471  
## 9 9 blue 4.608004 type\_animal 7.587858 64.06463 97.038416 19.159458  
## 10 10 blue 5.046002 type\_animal 9.683817 41.51654 138.441856 34.048304

#### GO ####

## Chapter II - Thermal Plasticity Database Exploration

Here, I am going to present a plot and analysis to test if is there relationship between temperature and plasticity.

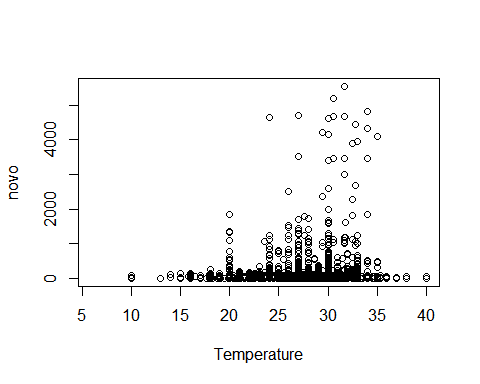
#mydata\_2 <- na.omit(my.data)  
Plasticity <- as.numeric(mydata$mean)  
Temperature <- as.numeric(mydata$T)  
plot(Plasticity ~ Temperature)



model <- lm(Plasticity ~ Temperature)  
summary(model)

##   
## Call:  
## lm(formula = Plasticity ~ Temperature)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -86.5 -50.3 -36.7 -12.9 5483.2   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -32.160 17.453 -1.843 0.0654 .   
## Temperature 2.976 0.629 4.731 2.27e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 239.6 on 9765 degrees of freedom  
## (6 observations deleted due to missingness)  
## Multiple R-squared: 0.002287, Adjusted R-squared: 0.002185   
## F-statistic: 22.38 on 1 and 9765 DF, p-value: 2.266e-06

Plasticity <- ceiling(Plasticity)  
  
### New plot removing a little values from graph ###  
n <- length(Plasticity)  
novo <- numeric(n)  
for (i in 1:n) {  
 ifelse(Plasticity[i] < 1, novo[i] <- NA, novo[i] <- Plasticity[i])  
}   
  
plot(novo ~ Temperature)



In this phase of database exploration, I am going to do a function to select more frequent data or species from my database.

### Identifying more frequent species ####  
  
frequent <- function(column) {  
 unique(column)  
 caracterizando <- table(column)  
 caracterizando2 <- as.data.frame(caracterizando)  
 ordenado <- sort(caracterizando2$Freq, decreasing = TRUE)  
 mais\_frequente <- ordenado[1]  
 n <- length(caracterizando2$Freq)  
 novo\_de\_novo <- numeric(n)  
 for (i in 1:n) {  
 if (caracterizando2$Freq[i] == mais\_frequente) {  
 novo\_de\_novo[i] <- TRUE  
 }  
 }  
   
 posicao <- which(grepl(1, novo\_de\_novo))  
 names(caracterizando2) <- c("Var1", "Freq")  
 resultado <- caracterizando2$Var1[posicao]  
 resultado2 <- as.character(resultado)  
 return(resultado2)  
}  
frequent(mydata$species)

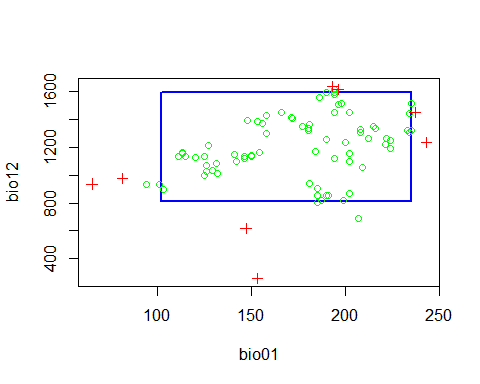
## [1] "serpentina"

especie\_nome\_final <- frequent(mydata$species)  
  
posicao\_sepentina <- which(grepl("serpentina", mydata$species))  
posicao\_sepentina2 <- posicao\_sepentina[1]  
genero <- mydata$genus[posicao\_sepentina2]  
  
tmp <- cbind(genero, especie\_nome\_final)  
  
specie\_more\_frequent <- str\_c(tmp, collapse = " ")  
specie\_more\_frequent

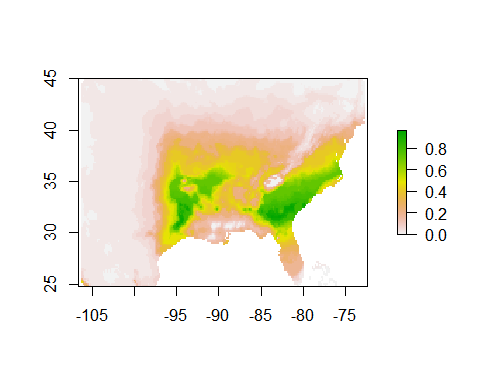
## [1] "Chelydra serpentina"

Below, I am going to present a niche model code withdraw of model done utilizing Wallace package. I did it to my more frequent species in previous database.

source(system.file('shiny/funcs', 'functions.R', package = 'wallace'))  
  
results <- spocc::occ(query = "Chelydra serpentina", from = "gbif", limit = 100, has\_coords = TRUE)  
results.data <- results[["gbif"]]$data[[formatSpName("Chelydra serpentina")]]  
occs.dups <- duplicated(results.data[c('longitude', 'latitude')])  
occs <- results.data[!occs.dups,]  
occs$latitude <- as.numeric(occs$latitude)  
occs$longitude <- as.numeric(occs$longitude)  
occs$occID <- row.names(occs)  
occs <- occs %>% filter(!(occID %in% 40))  
output <- spThin::thin(occs, 'latitude', 'longitude', 'name', thin.par = 1, reps = 100, locs.thinned.list.return = TRUE, write.files = FALSE, verbose = FALSE)  
maxThin <- which(sapply(output, nrow) == max(sapply(output, nrow)))  
maxThin <- output[[ifelse(length(maxThin) > 1, maxThin[1], maxThin)]]   
occs <- occs[as.numeric(rownames(maxThin)),]   
envs <- raster::getData(name = "worldclim", var = "bio", res = 10, lat = , lon = )  
envRes <- 10  
if (envRes == 0.5) {  
 i <- grep('\_', names(envs))  
 editNames <- sapply(strsplit(names(envs)[i], '\_'), function(x) x[1])  
 names(envs)[i] <- editNames  
}  
i <- grep('bio[0-9]$', names(envs))  
editNames <- paste('bio', sapply(strsplit(names(envs)[i], 'bio'), function(x) x[2]), sep='0')  
names(envs)[i] <- editNames  
envs <- envs[[c('bio01', 'bio12')]]  
locs.vals <- raster::extract(envs[[1]], occs[, c('longitude', 'latitude')])  
occs <- occs[!is.na(locs.vals), ]   
xmin <- min(occs$longitude)  
xmax <- max(occs$longitude)  
ymin <- min(occs$latitude)  
ymax <- max(occs$latitude)  
bb <- matrix(c(xmin, xmin, xmax, xmax, xmin, ymin, ymax, ymax, ymin, ymin), ncol=2)  
bgExt <- sp::SpatialPolygons(list(sp::Polygons(list(sp::Polygon(bb)), 1)))  
bgExt <- rgeos::gBuffer(bgExt, width = 0.5)  
envsBgCrop <- raster::crop(envs, bgExt)  
envsBgMsk <- raster::mask(envsBgCrop, bgExt)  
bg.xy <- dismo::randomPoints(envsBgMsk, 10000)  
bg.xy <- as.data.frame(bg.xy)   
occs.xy <- occs[c('longitude', 'latitude')]  
group.data <- ENMeval::get.jackknife(occ=occs.xy, bg.coords=bg.xy)  
occs.grp <- group.data[[1]]  
bg.grp <- group.data[[2]]  
e <- BioClim\_eval(as.data.frame(occs.xy), bg.xy, occs.grp, bg.grp, envsBgMsk)  
evalTbl <- e$results  
evalMods <- e$models  
names(e$predictions) <- "BIOCLIM"  
evalPreds <- e$predictions  
plot(evalMods[['BIOCLIM']], a = 1, b = 2, p = 0.9)



mod <- evalMods[["BIOCLIM"]]  
pred <- evalPreds[["BIOCLIM"]]  
plot(pred)



projCoords <- data.frame(x = c(-115.9918, -95.9603, -65.7374, -66.6159, -93.676, -115.9918), y = c(42.1634, 18.1459, 33.8704, 47.9899, 50.5134, 42.1634))  
projPoly <- sp::SpatialPolygons(list(sp::Polygons(list(sp::Polygon(projCoords)), ID=1)))  
envsFuture <- raster::getData("CMIP5", var = "bio", res = 10, rcp = 45, model = "CE", year = 50)  
predsProj <- raster::crop(envsFuture, projPoly)  
predsProj <- raster::mask(predsProj, projPoly)  
names(predsProj) <- paste0('bio', sprintf("%02d", 1:19))  
predsProj <- raster::subset(predsProj, names(envs))  
proj <- dismo::predict(mod, predsProj)  
plot(proj)

