Ecological niche models and species distributions

Jesús N. Pinto-Ledezma and Jeannine Cavender-Bares

In this lab, we will explore some Correlative models use data at species level (occurrence data points) and environmental data at large spatial scales. We will use data occurrence data of the Hyacinth macaw (*Anodorhynchus hyacinthinus*), the largest macaw in the world and that is distributed in the Pantanal of Bolivia and Brazil and the Cerrado en Brazil. The environmental was obtained from **Ecoclimate** (http://ecoclimate.org). Notice that you can obtain environmental data from WorldClim.

Part of the explanation of algorithms was extracted from the vignette of the dismo package.

Set up your data and your working directory

Set up a working directory and put the two data files in that directory. Tell R that this is the directory you will be using, and read in your data:

```
setwd("path/for/your/directory")
```

Install and load the following packages.

```
library(vegan)
library(raster)
library(sp)
library(psych)
library(maps)
library(maptools)
library(kernlab)
library(dismo)
```

Prepare environmental data

Read the environmental variables (bioclimatic variables in WorldClim - http://www.worldclim.org/bioclim)

```
datoG0 <- read.table("Data/Environment/bio_var_CCSM_0k_global.txt", h = T)
```

Lets explore some details of the raw data

```
head(datoG0)
datoG0[1:5, 1:5]
dim(datoG0)
str(datoG0)
summary(datoG0)
class(datoG0)
names(datoG0)
```

Note that the environmental data is a data frame, but we need a raster. To transfor the data frame to raster, we just need to use the function **gridded**. Lets try!

```
gridded(datoG0) <- ~long+lat
```

```
class(datoG0)
```

Now use the function **stack** to merge all environmental variables into a single raster

```
climaOk <- stack(datoGO)
```

Lets plot one of the variable to check if everithing is ok.

```
plot(climaOk$bio.1)
map(add = T)
```

This map correspond to the environmental varibale Annual Mean Temperature or BIO-1 accross the world.

However, we do not need all environmental information across the world, as our focal species is ditributed only in South America (if the species is in a Zoo in US or Europe doesn't mean that the species occur in those places, dah!).

```
# Set a geographical extension (Sur America long = -90, -30, lat = -60, 15)
e <- extent(c(-90, -30, -60, 15))
```

Using the extent cut the information only for South America

```
climaOk.SA <- crop(climaOk, e)
```

Check if everything is ok.

```
plot(climaOk.SA$bio.9)
map(add = T)
```

```
ncell(climaOk.SA)
```

The next step is to extract the information from the raster of South America.

```
SAOk <- values(climaOk.SA)
```

```
class(SAOk)
```

Obtain the the geographical coordinates for each cell in South America.

```
coord.SA <- xyFromCell(climaOk.SA, 1:ncell(climaOk.SA))
```

Now, merge the environmental values with the geographical coordinates.

```
SAOk <- cbind(coord.SA, SAOk)
```

Check the values on specific rows and columns.

```
# observar valores de lineas y columnas especificas
SAOk[1:5,]
SAOk[1:5, 1:5]
```

Now, we need to decide which variables are needed to model the niche (Grinnellian niche or Fundamental niche) for the Hyacinth macaw. Here we can rely the selection to the specialist or use statistical tools or both.

Lets use statistical tools. In this case we will use a factorial analysis that is usually used in psicology.

```
# Variable selection (exceute a factorial analysis [FA])
fa.parallel(SAOk[, -c(1:3)])
```

Why we are not using the columns 1 to 3? lets inspect.

```
SAOk[1:5, 1:3]
```

The parallel analysis suggests that the number of factors = 5 and the number of components = 5 are sufficient for further analysis.

```
SAOk.fa <- fa(SAOk[, -c(1:3)], nfactors = 5, rotate = "varimax")
SAOk.fa
```

```
loadings(SAOk.fa)
```

Based on the results of the FA and the specialist knowledge we can select the next variables: bio1, bio2, bio15, bio4, bio2.

```
#6. Guardar las variables seleccionadas
write.table(SAOk[, c("x", "y", "bio.1", "bio.2", "bio.4", "bio.12", "bio.15")],
            row.names = FALSE, "Data/Environment/climaSAOk.txt", sep = "\t")
```

We can also save each environmental variable as raster files in our computer, but we do not need to do that.

```
writeRaster(climaOk.SA, "Data/Environment/climaSAOk.asc", format = "ascii", bylayer = T)
```

Now we have the environmental information needed for modeling the fundamental niche of the Hyacinth macaw. So, we can clean our R environment.

```
rm(list = ls())
```

Prepare species data

Get some occurrence data for our species from GBIF, directly within R. This may take some time, given the number of occurrences for the selected species.

NOTE: we need to have an internet connection.

```
hyacinth <- gbif("anodorhynchus", "hyacinthinus*", geo = FALSE)
```

Inspect the results.

```
# how many rows and colums?
dim(hyacinth)
## [1] 3986   130
# select the records that have longitude and latitude data
colnames(hyacinth)
```

Wow, there are a bunch of columns, lets clean our data.

First, removing all NA data.

```
hyacinth <- subset(hyacinth, !is.na(lon) & !is.na(lat))
```

```
dim(hyacinth)
## [1] 3703   130
# show some values
hyacinth[1:4, c(1:5, 7:10)]
```

Second, select only the columns that are necessary for our modeling porpuses. Notice that using colnames(hyacinth) you can obtain the position of the columns, and the columns 114, 84 and 77 (in that order) are the columns that correspond to species name, longitude and latitude, respectively and that is all what we need.

```
hyacinth_coords <- hyacinth[, c(114, 84, 77)]
```

Lets inspect the results.

```
head(hyacinth_coords)
```

Now lets save the data needed for further analyses.

```
dir.create("Data/OCC")
write.csv(hyacinth_coords, "Data/OCC/hyacinth_data.csv")
```

We can clean our R environment.

```
rm(list = ls())
```

Merge both information

First load the occurrence data for our species.

```
hyacinth <- read.csv("Data/OCC/hyacinth_data.csv")
```

As we are working only with one species, we just need the lon/lat columns.

```
head(hyacinth)
```

The columns 3 and 4 correspond to the geographical coordinates.

```
hyacinth_coords <- hyacinth[, c(3, 4)]
```

```
head(hyacinth_coords)
```

Now, load the environmental information saved in the first part of the this lab.

```
climSA0k <- read.table("Data/Environment/climaSA0k.txt", h = T)
class(climSA0k)
```

```
gridded(climSA0k) <- ~x + y
climSA0k <- stack(climSA0k)
climSA0k
```

Inspect the both data.

```
plot(climSA0k$bio.1)
points(hyacinth_coords[, "lon"], hyacinth_coords[, "lat"])
```

Great, it seems that everything is good!

Now, we will extract the environmental information using the hyacinth occurrences.

```
hyacinth_var <- extract(climSA0k, hyacinth_coords, cellnumbers = T)
head(hyacinth_var)
```

Merge the environmental data with the species occurrences.

```
hyacinth_var <- cbind(hyacinth_coords, hyacinth_var)
hyacinth_var[1:5, ]
```

Remove of rows with NA values.

```
hyacinth_var <- na.omit(hyacinth_var) # remove NA's
dim(hyacinth_var)
```

Now, lets remove all duplicate values based on the cells position.

```
duplicated(hyacinth_var[,"cells"])
```

Wow, there are a lot of duplicate cells.

```
a <- which(duplicated(hyacinth_var[, "cells"]) == T)
a
```

Using the object a, we can delete the duplicate cells.

```
hyacinth_var <- hyacinth_var[-a, ]
dim(hyacinth_var)
```

Interesting, only 123 occurrences without duplicates. We can save this information in our OCC subfolder.

```
write.table(hyacinth_var, row.names = FALSE, "Data/OCC/hyacinth_var.txt", sep = "\t")
```

Inspect how the 123 looks in the map.

```
plot(climSAOk$bio.1)
points(hyacinth_var[, "lon"], hyacinth_var[, "lat"])
```

The last step to build the Hyacinth macaw environmental niche is to establish a background (pseudoabsences). To do that we will use the environmental data.

```
climaOk <- read.table("Data/Environment/climaSAOk.txt", h = T)
dim(climaOk)
```

Set the background.

```
id.back <- sample(1:nrow(climaOk), 123) #123 is the same number of TRUE occurrences
length(id.back)
```

```
background <- climaOk[id.back, ]
dim(background)
names(background)
```

```
write.table(background, "Data/Environment/background.txt", row.names = F, sep = "\t")
```

Well, we are almost there.

Building ecological niche models

The first step is to set data for training and test. To do that, we will use the TRUE occurrences and the background (pseudoabsences).

```
id.ocur <- sample(1:nrow(hyacinth_var), round(0.75*nrow(hyacinth_var)))
length(id.ocur)
```

```
id.back <- sample(1:nrow(background), round(0.75*nrow(background)))
length(id.back)
```

Set data for training.

```
training <- prepareData(x = climSAOk,
                              p = hyacinth_var[id.ocur, 1:2],
                              b = background[id.back, 1:2], xy = T)
```

Set data for testing.

```
test <- prepareData(x = climSAOk,
                          p = hyacinth_var[-id.ocur, 1:2],
                          b = background[-id.back,1:2], xy = T)
```

Now, we will fit some models using different algorithms. Bioclim, DomianGower, SVM, GLMz.

Fit models

Bioclim model

The BIOCLIM algorithm computes the similarity of a location by comparing the values of environmental variables at any location to a percentile distribution of the values at known locations of occurrence ('training sites').

```r
Bioclim.model <- bioclim(x = training[training[, "pb"] == 1, -c(1:3)])
```

```r
Bioclim.model
```

```r
plot(Bioclim.model)
```

Lets plot the response variables, or single variable response curves for a model.

We can used this model to explore how the species responses to these particular variables (based on the particular algorithm that we used and the relationship between the geographic and environmental space), using a response function that creates response plots for each variable, with the other variables at their median value.

```r
response(Bioclim.model)
```

Gower model

The Domain algorithm computes the Gower distance between environmental variables at any location and those at any of the known locations of occurrence ('training sites'). For each variable the minimum distance between a site and any of the training points is taken.

```r
Gower.model <- domain(x = training[training[, "pb"] == 1, -c(1:3)])
Gower.model
```

We can also explore the response for each variable.

```r
response(Gower.model)
```

Support Vector Machines or SVM model

Support Vector Machines are an excellent tool for classification, novelty detection, and regression.

Support Vector Machines (SVMs; Vapnik, 1998) apply a simple linear method to the data but in a high-dimensional feature space non-linearly related to the input space, but in practice, it does not involve any computations in that high-dimensional space. This simplicity combined with state of the art performance on many learning problems (classification, regression, and novelty detection) has contributed to the popularity of the SVM (Karatzoglou et al., 2006).

```r
svm.model <- ksvm(pb ~ bio.1 + bio.2 + bio.4 + bio.12 + bio.15, data = training)
```

Generalized Linear Models

A generalized linear model (GLM) is a generalization of ordinary least squares regression. Models are fit using maximum likelihood and by allowing the linear model to be related to the response variable via a link function and by allowing the magnitude of the variance of each measurement to be a function of its predicted value. Depending on how a GLM is specified it can be equivalent to (multiple) linear regression, logistic regression or Poisson regression.

```
glm.model <- glm(pb ~ bio.1 + bio.2 + bio.4 + bio.12 + bio.15,
                  data = training, family = binomial(link = "logit"))
```

Until now, we just fitted different different models for the Hyacinth macaw using four different algorithms. The next step is use these fitted models to make spatial predictions.

Predictions

Notice that in here we will use the fitted models to predict the fundamental niche (Grinnellian) for our species in South America.

Bioclim

```
BioclimOk <- predict(object = Bioclim.model, x = climSAOk)
```

Lets see how the bioclim model looks like! and after that repeat for the other three algorithms.

```
plot(BioclimOk)
```

What about if we plot the prediction and the TRUE occurrences used for fit the model.

```
plot(BioclimOk)
points(training[training[, "pb"]==1,"x"], training[training[, "pb"]==1,"y"])
```

points(entrenamiento[entrenamiento[, “pb”]==0,“x”], entrenamiento[entrenamiento[, “pb”]==0,“y”], pch = 20, col = “red”)

Gower

```
GowerOk <- predict(climSAOk, Gower.model)
```

Inspect the Gower model.

```
plot(GowerOk)
```

SVM

```
svmOk <- predict(climSAOk, svm.model)
```

```
plot(svmOk)
```

GLMz

```
GLMOk <- predict(climSAOk, glm.model)
```

```
plot(GLMOk)
```

Now lets plot all four models # Ver los resultados

```
par(mfrow = c(2, 2))
plot(BioclimOk, main = "bioclim")
plot(GowerOk, main = "Gower")
plot(svmOk, main = "svm")
plot(GLMOk, main = "glm")
```

Model evaluation

Now we will evaluate the performance of each model. To evaluate the performance of our models, we will use the testing data. Repeat for all four models.

Cross-validation of models with presence/absence data. Given a vector of presence and a vector of absence values (or a model and presence and absence points and predictors), confusion matrices are computed (for varying thresholds), and model evaluation statistics are computed for each confusion matrix / threshold.

Bioclim

```
Bioclim.eval <- evaluate(p = test[test[, "pb"] == 1, 1:2],
                         a = test[test[, "pb"] == 0, 1:2],
                         model = Bioclim.model,
                         x = climSAOk)
```

```
Bioclim.eval
str(Bioclim.eval)
```

Gower

```
Gower.eval <- evaluate(p = test[test[, "pb"] == 1, 1:2],
                       a = test[test[, "pb"] == 0, 1:2],
                       model = Gower.model,
                       x = climSAOk)
```

SVM

```
svm.eval <- evaluate(p = test[test[, "pb"] == 1, 1:2],
                     a = test[test[, "pb"] == 0, 1:2],
                     model = svm.model,
                     x = climSAOk)
```

GLMz

```
glm.eval <- evaluate(p = test[test[, "pb"] == 1, 1:2],
                     a = test[test[, "pb"] == 0, 1:2],
                     model = glm.model,
                     x = climSAOk)
```

Now, lets plot the performance of the models.

```
# validate results
par(mfrow = c(2, 2))
plot(Bioclim.eval, "ROC")
```

```r
plot(Gower.eval, "ROC")
plot(svm.eval, "ROC")
plot(glm.eval, "ROC")
```

Finally, lets produce “shadows” from “ghosts”.

Species geographical distributions

First we need to set threshold for each model, the threshold (cut-off) is used to transform model predictions (probabilities, distances, or similar values) to a binary score (presence or absence).

Bioclim

```r
Bioclim.thr <- threshold(Bioclim.eval)
```

Inspect the “thr” object and then repeat for the other three models.

```r
Bioclim.thr
```

There are different columns in the “thr” object and each one is a measure of threshold, in this example we will use the “spec_sens” column or species sensibility. In the spec_sens, the threshold at which the sum of the sensitivity (true positive rate) and specificity (true negative rate) is highest.

```r
bio <- Bioclim.thr$spec_sens
```

Gower

```r
Gower.thr <- threshold(Gower.eval)
```

```r
gow <- Gower.thr$spec_sens
```

SVM

```r
svm.thr <- threshold(svm.eval)
```

```r
s <- svm.thr$spec_sens
```

GLMz

```r
glm.thr <- threshold(glm.eval)
```

```r
g <- glm.thr$spec_sens
```

Lets see what happen…

```r
par(mfrow = c(2, 2))
plot(BioclimOk > bio, main = "Bioclim")
plot(GLMOk > g, main = "GLM")
plot(GowerOk > gow, main = "Gower")
plot(svmOk > s, main = "SVM")
```

Looks like the different algortihms produce different shadows… lets see what happen if we combine all four models into a single one "ensemble framework".

```r
# Combine all thresholds
thrs <- (bio + gow + s + g)
```

Now combine all predictions into a single prediction.

```r
tmp <- stack(BioclimOk, GowerOk, GLMOk, svmOk)
```

First, lets sum all predictions and see what happen…

```r
map.sum <- sum(tmp) # sum
```

Plot the result.

```r
par(mfrow = c(2, 2))
plot(map.sum)
plot(map.sum > thrs)
plot(map.sum > 2)
plot(map.sum > 3)
```

Now, what about the mean…

```r
map.mean <- mean(tmp) # mean
```

```r
map.sd <- calc(tmp, sd) # sd
```

Plot the results…

```r
plot(map.mean)
```

```r
par(mfrow = c(2, 2))
plot(map.mean)
plot(map.mean > thrs)
plot(map.mean > 0.2)
plot(map.mean > 0.3)
```

However, this is a problematic approach as the values predicted by the models are not all on the same (between 0 and 1) scale; so you may want to fix that first. Another concern could be weighting. Let's combine the four models weighted by their AUC scores. Here, to create the weights, we substract 0.5 (the random expectation) and square the result to give further weight to higher AUC values.

```r
auc <- sapply(list(Bioclim.eval, Gower.eval, svm.eval, glm.eval), function(x) x@auc)
```

```r
w <- (auc-0.5)^2
```

```r
map.mean.weight <- weighted.mean(tmp[[c("layer.1", "layer.2", "layer.3", "layer.4")]], w)
```

```r
plot(map.mean.weight)
```

Finally, lets plot the four predictions

```r
par(mfrow = c(2, 2))
plot(map.sum, main = "Sum of all models")
plot(map.mean, main = "Mean of all models")
plot(map.mean.weight, main = "Weighted mean of all models")
plot(map.sd, main = "Standard deviation of all models")
```

That's it, we modeled environmental niches (ghosts) to produce geographical distributions (shadows)…

Excercices

Please respond each question based on the practice.

1. Summarize the data: how many records are there, how many have coordinates, how many records without coordinates have a textual georeference (locality description)?
2. Do you think the observations are a reasonable representation of the distribution (and ecological niche) of the species?
3. There is a best model? Explain your answer.
4. Can we use ENM/SDM to aid the conservation of the Hyacinth macaw? Explain your answer.
5. Based on the lecture and the practice, what can we conclude?

References

Karatzoglou, A., D. Meyer & K. Hornik, 2006. Support Vector Machines in R. Journal of statistical software 15(9).

Lima-Ribeiro, M.S., S. Varela, J. González-Hernández, G. Oliveira, J.A.F. Diniz-Filho & L.C. Terribile. 2015. ecoClimate: a database of climate data from multiple models for past, present, and the future for macroecologistis and biogeographers. Biodiversity Informatics, 10: 1-21.

Vapnik, V., 1998. Statistical Learning Theory. Wiley, New York.