Species Distribution Modeling in R

Sode A.I., Olajide A.Y., Nakhwala L., Opara A., Opoku M., Barasa C.W.

2024-09-25

Introduction

This tutorial presents the workflow to build a Species Distribution Model (SDM) for *Bidens bipinnata*, the most abundant introduced species in Fogo Island in Cabo Verde (West Africa).

We used **RMarkdown**, a simple formatting syntax for authoring HTML, MS Word and PDF documents (see http://rmarkdown.rstudio.com). So, when you open the RMarkdown file and click the **Knit** button, a document will be generated that includes both content and the output of any embedded R code chunks within the document.

The tutorial is built on the blog publication of Olivier 2024. However, we made some major changes to the environmental data processing section as these data have different resolutions and projection systems.

In the next section, we will describe the data we used to build the distribution model for *Bidens bipinnata* on Fogo Island.

Description of the data

Two data types have been used to build the SDM for *B. bipinnata*: occurrence and environmental data. The **occurrence data** are constituted of species' geographical coordinates (longitude and latitude). These data are the subset of the Fogo species data we used in the second Module of our learning materials. The **environmental data** stand for descriptors of the environment. They can include abiotic measurements such as temperature, precipitation, soil types, and land cover as well as biotic factors, such as the presence or absence of other species (like predators, competitors, or food sources). In this tutorial, we will focus on climate data and land cover.

Data preparation

Loading required R packages

```
library(terra)

## terra 1.7.78

library(geodata)
library(predicts)
```

Occurrence data

First, we load the occurrence data of *B. bipinnata* observed in Fogo Island.

```
obs_data <- read.csv(file = "occ_data/Bidens_bipinnata.csv")
head(obs_data)
```

```
##            species longitude latitude
## 1 Bidens bipinnata  781245.0  1643415
## 2 Bidens bipinnata  781200.6  1643406
## 3 Bidens bipinnata  781107.8  1643438
## 4 Bidens bipinnata  781091.3  1643488
## 5 Bidens bipinnata  781051.6  1643561
## 6 Bidens bipinnata  781117.0  1643610
```

After loading the data, we get an overview of it. We notice 132 records for the species occurrence.

```
summary(obs_data)
```

```
##    species            longitude         latitude
##  Length:132         Min.   :771175   Min.   :1642888
##  Class :character   1st Qu.:778956   1st Qu.:1649060
##  Mode  :character   Median :781090   Median :1658847
##                     Mean   :781810   Mean   :1655529
##                     3rd Qu.:784493   3rd Qu.:1661021
##                     Max.   :791290   Max.   :1662071
```

We drop NAs (if applicable) and make sure they went way before proceeding. We can notice there is no NA.

```
obs_data <- obs_data[!is.na(obs_data$latitude), ]
summary(obs_data)
```

```
##    species            longitude         latitude
##  Length:132         Min.   :771175   Min.   :1642888
##  Class :character   1st Qu.:778956   1st Qu.:1649060
##  Mode  :character   Median :781090   Median :1658847
##                     Mean   :781810   Mean   :1655529
##                     3rd Qu.:784493   3rd Qu.:1661021
##                     Max.   :791290   Max.   :1662071
```

Now, we create a spatial vector object using the UTM coordinate system. This will help us later when we need to overlay the species locations onto Fogo Island map.

```
obs_points <- vect(obs_data, geom = c("longitude", "latitude"),
                   crs = "+proj=utm +zone=26 +datum=WGS84 +units=m")
```

Then we project the vector data into *long/lat* reference system to match it with the reference system of Fogo Island map and bioclimatic layers. Note that bioclimatic data available in *long/lat* projection system will be downloaded in the next section.

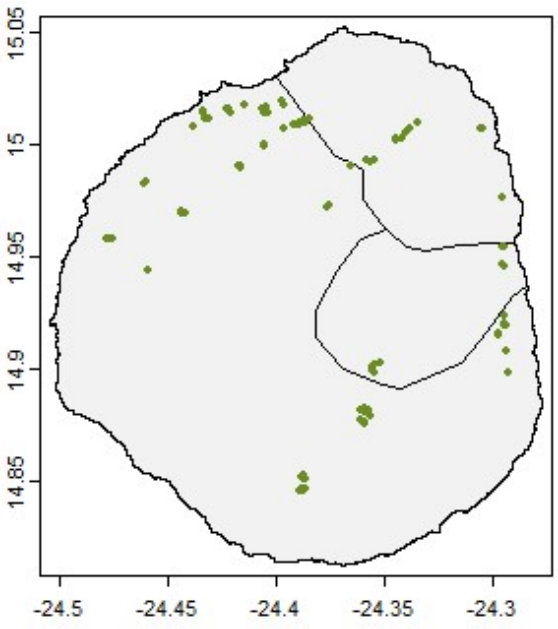
```
obs_points <- project(obs_points, "+proj=longlat +datum=WGS84")
```

Before proceeding, we should load Fogo Island map to visualize the observed species locations.

```
# Load the map of Cabo Verde available as a shapefile.
map_fogo <- vect("gadm_cpv/fogo_island.shp")
```

We can then plot the observed points on the map and see how they are distributed across the region.

```
plot(map_fogo, axes = TRUE, col = "grey95")
points(x = geom(obs_points)[,"x"],
       y = geom(obs_points)[,"y"],
       col = "olivedrab",
       pch = 20,
       cex = 0.8)
```



Though our focus in this module is not mapping, you are free to add to the map above a legend, title and any other formatting elements relevant for its understanding.

Environmental data

Bioclimatic data

After we processed the field data, we are going to download the climate data from the `Worldclim` website https://www.worldclim.org/.

```r
# the path where the data should be stored
data_path <- "env_data/climate/wc2.1_country"

# Check if the data already exists
if (!file.exists(data_path)) {
  # data does not exist, download it
  message("Climate data not found, downloading...")
  bioclim_data <- worldclim_country(country = "cabo verde", var = "bio",
                                    res = 0.5,
                                    path = "env_data/")
} else {
  # data exists, load it and proceed
  message("Climate data already exists, proceeding...")
  bioclim_data <- rast(list.files(data_path, pattern = ".tif", full.names =
TRUE))
}
```

```
## Climate data already exists, proceeding...
```

For the sake of simplicity, we consider only three bioclimatic variables: annual temperature (bio1), temperature seasonality (bio4) and annual precipitation (bio12). However, we recommend you to explore variables selection techniques or use expert knowledge to come up with the potential environmental variables that could influence the distribution of the species of interest.

```r
bioclim_data <- c(bioclim_data$wc2.1_30s_bio_1, bioclim_data$wc2.1_30s_bio_4,
bioclim_data$wc2.1_30s_bio_12)
```

Then, we crop bioclimatic variables using the geographic extent of Fogo Island. For other applications in which environmental variables are available beyond the study region, a study extent slightly larger than the study region is recommended. In this specific application, bioclimatic data are not available in the ocean around Fogo Island. So there no need to include this area in the analysis.

```r
bioclim_crop <- crop(bioclim_data, map_fogo)
bioclim_crop
```

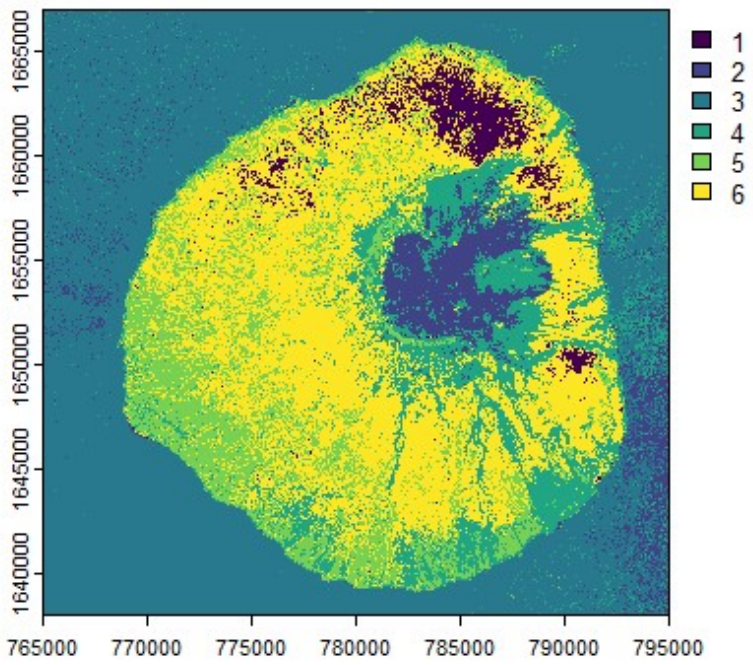
```
## class       : SpatRaster
## dimensions  : 29, 28, 3  (nrow, ncol, nlyr)
## resolution  : 0.008333333, 0.008333333  (x, y)
## extent      : -24.50833, -24.275, 14.80833, 15.05  (xmin, xmax, ymin,
ymax)
## coord. ref. : lon/lat WGS 84 (EPSG:4326)
## source(s)   : memory
## varname     : CPV_wc2.1_30s_bio
## names       : wc2.1_30s_bio_1, wc2.1_30s_bio_4, wc2.1_30s_bio_12
## min values  :        11.03333,        138.8017,              229
## max values  :        24.40417,        177.9364,              598
```

We can see that the resolution of the bioclimatic data is `0.008333333` (approximately 1 kilometer at the Ecuador).

Land cover data

Temperature and precipitation are well known to influence species distribution at large scale. However, at small scale like Fogo Island, other abiotic measurements like land cover may influence the species distribution. So, we have to load our land cover layer we created from Module 2 to use it as a covariate in the species distribution model we are building for *B. bipinnata*.

```
landcov <- rast("env_data/landcover_model2.tif")
plot(landcov)
```



Since all the environmental variables should have the same projection system and resolution, we project the land cover variable into *long/lat* system and crop it to Fogo Island extent. However, you should be aware that the good practice is to project vector layers instead of raster layers due to the lack of precision associated with raster projection. So we recommend projecting a raster layer in another reference system if it is really necessary.

In the next chunk of code, we project the land cover map and crop its extent from Fogo Island map so that all environmental data have the same extent. Note that the `mask = TRUE` argument helps remove the Ocean around Fogo Island as there is no bioclimatic data available in that region.

```
landcov_ll <- project(landcov, "+proj=longlat +datum=WGS84", method ="near")
landcov_ll <- crop(landcov_ll, map_fogo, mask = TRUE)
landcov_ll
```

```
## class       : SpatRaster
## dimensions  : 2619, 2483, 1  (nrow, ncol, nlyr)
## resolution  : 9.163383e-05, 9.163383e-05  (x, y)
## extent      : -24.50488, -24.27735, 14.81239, 15.05238  (xmin, xmax, ymin,
ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs
## source(s)   : memory
## name        : class
## min value   :     1
## max value   :     6
```

As we can see, the land cover resolution is different from that of bioclimatic data. So, we have to *resample* the land cover raster to the same resolution as the climatic data using the *near* interpolation method recommended for *categorical variable*. Note that the resolution of bioclimatic data is lower than the original resolution of the land cover data. However, 30 arc-second resolution used in this study is the highest resolution available for bioclimatic data on the Worldclim website.

```
landcov_res <- resample(landcov_ll, bioclim_crop$wc2.1_30s_bio_1, method =
"near")
landcov_res
```

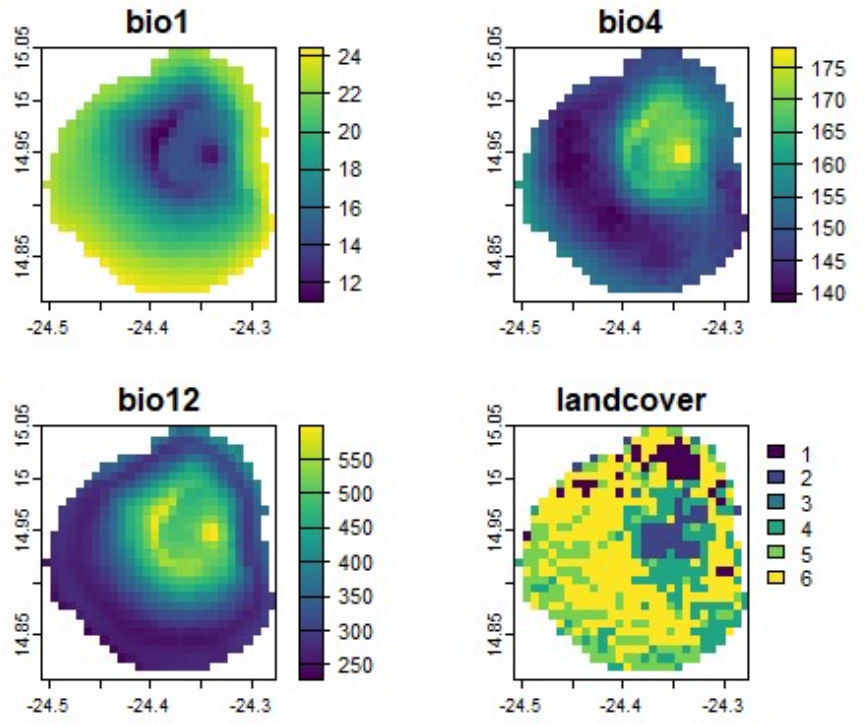
```
## class       : SpatRaster
## dimensions  : 29, 28, 1  (nrow, ncol, nlyr)
## resolution  : 0.008333333, 0.008333333  (x, y)
## extent      : -24.50833, -24.275, 14.80833, 15.05  (xmin, xmax, ymin,
ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs
## source(s)   : memory
## varname     : CPV_wc2.1_30s_bio
## name        : class
## min value   :     1
## max value   :     6
```

After resampling the land cover map, we convert the raster into a factor before proceeding as we are dealing with a categorical variable.

```
landcov_res <- as.factor(landcov_res)
```

Then, we merge the four environmental variables into a raster stack and visualize them on a quick map.

```
bioclim_kept        <- c(bioclim_crop, landcov_res)
names(bioclim_kept) <- c("bio1", "bio4", "bio12", "landcover")
plot(bioclim_kept)
```



Creating pseudo-absence points

To evaluate species distribution models with presence-only data, and really understand the factors influencing where *B.bipinnata* occur, we need to include some absence or "background" points for coercing presence-only data for use with presence/absence approaches.

we then create a set of 200 background points (i.e. pseudo-absences) at random, and add them to our data. For a large study extent and depending on the size of observed points, one can use 1,000 or even 5,000 pseudo-absence points. We encourage you to play with different numbers of background points and compare the results.

```
# Set seed for the random number generator to ensure results are similar
across users.
set.seed(12354)

# Randomly sample points
background <- spatSample(x = bioclim_kept,
                         size = 200,      # 200 pseudo-absence points
                         values = FALSE, # don't need values
                         na.rm = TRUE,   # no sample from ocean
                         xy = TRUE)      # coordinates

# Look at the first rows
head(background)
```

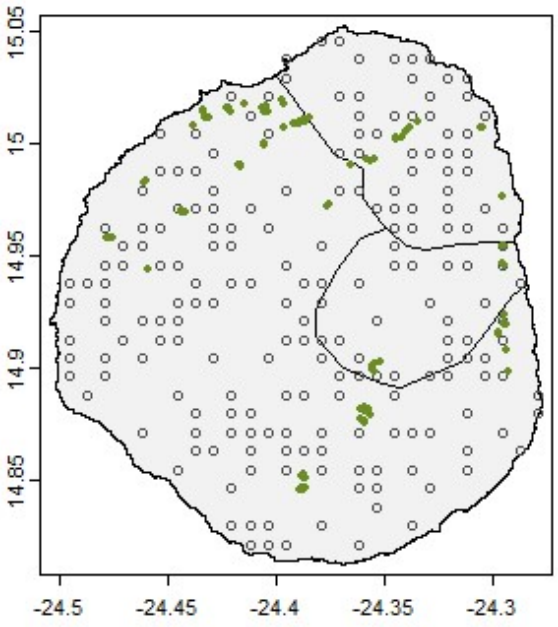
```
##              x        y
## [1,] -24.37917 14.91250
## [2,] -24.41250 14.82917
## [3,] -24.30417 15.01250
## [4,] -24.47917 14.90417
## [5,] -24.31250 14.99583
## [6,] -24.35417 14.84583
```

After creating the background points, we can map them together with the observed species locations. On the resulting map, the background points are highlighted in grey color while the species occurrence locations are shown in *oliverdrab* color.

```
# Plot the base map
plot(map_fogo,
     axes = TRUE,
     col = "grey95")

# Add the background points
points(background,
       col = "grey30",
       pch = 1,
       cex = 0.75)

# Add the points for individual observations
points(x = geom(obs_points)[,"x"],
       y = geom(obs_points)[,"y"],
       col = "olivedrab",
       pch = 20,
       cex = 0.75)
```



Now, we can create a single dataset for both occurrence and pseudo-absence data. We create an additional column `pa` to indicate each type of points.

```
# Presence-only data
presence <- as.data.frame(geom(obs_points)[, c("x", "y")])
colnames(presence) <- c("longitude", "latitude")
# Add column indicating presence
presence$pa <- 1

# Convert background data to a data frame
absence <- as.data.frame(background)
colnames(absence) <- c("longitude", "latitude")
# Add column indicating absence
absence$pa <- 0

# Join data into single data frame
all_points <- rbind(presence, absence)

# check the results
head(all_points)
```

```
##   longitude latitude pa
## 1 -24.38655 14.85031  1
## 2 -24.38697 14.85024  1
## 3 -24.38782 14.85054  1
## 4 -24.38797 14.85099  1
```

```
## 5 -24.38833 14.85165  1
## 6 -24.38772 14.85209  1
```

Adding climate data

We use the `extract()` function, which takes geographic coordinates and raster layers as input, and extract values in the raster data for each of the geographic coordinates.

```
bioclim_extract <- extract(x = bioclim_kept,
                           y = all_points[, c("longitude", "latitude")],
                           ID = FALSE)
```

Now, we need to join the extracted data with points and drop out the longitude/latitude columns which are no longer relevant for the SDM implementation.

```
# Add the point and climate datasets together
points_climate <- cbind(all_points, bioclim_extract)

# Identify columns that are latitude & longitude
drop_cols <- which(colnames(points_climate) %in% c("longitude", "latitude"))
drop_cols

## [1] 1 2

# Remove the geographic coordinates from the data frame
points_climate <- points_climate[, -drop_cols]
```

Note that before proceeding, one can standardize numeric covariates to have the same scale, especially when the model includes many covariates with different scales. We encourage learners to think about this aspect in their future projects. In the next section, we will generate the training and test data for our SDM.

Training and testing data

After preparing our data for model building, we are going to split it into training and test samples. So, we will use 80% of the data for training the model and 20% for testing it.

```
# Create vector indicating fold
fold <- folds(x = points_climate,
              k = 5,
              by = points_climate$pa)
```

Take a look at each split

```
table(fold)

## fold
##  1  2  3  4  5
## 66 67 66 67 66
```

We can use any observations in fold 1 as a test sample and the remaining folds as the training set. A more robust approach is the *K-fold cross-validation* used in Module 2 for land

cover classification. We encourage you to test this approach and compare results with those we obtained here.

```
testing  <- points_climate[fold == 1, ]
training <- points_climate[fold != 1, ]
```

Model building

Now, it is time to build our SDM. Several SDM approaches are available to handle presence-absence or presence-background data including generalized linear models (GLMs) and its variants, Maximum Entropy (Maxent), tree-based methods (e.g. Random Forest), etc.

In this study, we use a generalized linear model with `binomial()` family which is also known as logistic regression model, a popular modeling approach used in machine learning. The column `pa` is the binary response variable while `"."` indicates to the `glm()` function that all the the remaining columns should be considered as covariates in the model (i.e. bio1, bio4, bio12 and land cover).

```
# Build a model using training data
glm_model <- glm(pa ~ ., data = training, family = binomial())
```

After building the model, we can now view the results and look at the significance of covariates. So we run the analysis of variance (ANOVA) on the model object.

```
anova(glm_model)
```

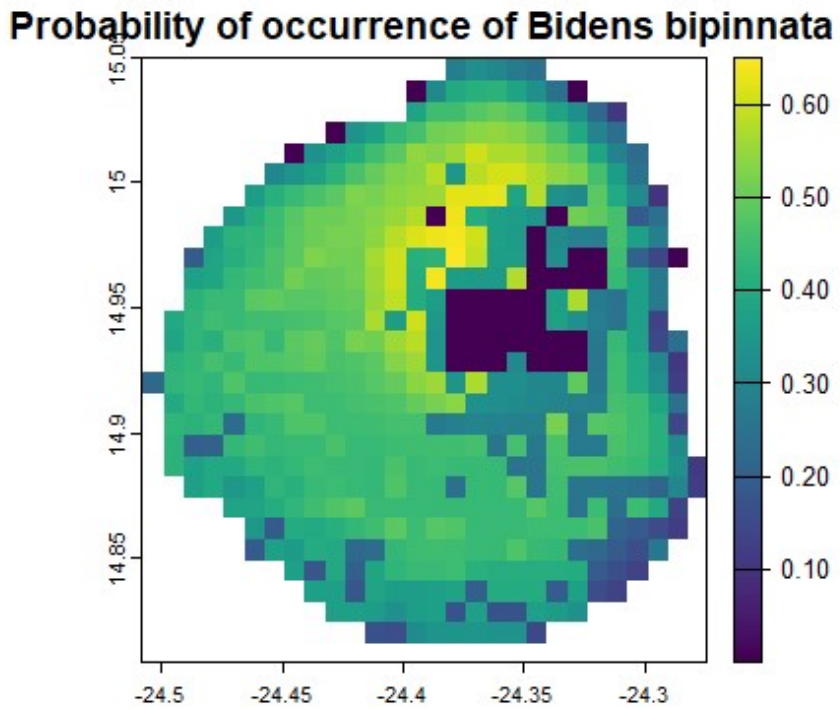
```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: pa
##
## Terms added sequentially (first to last)
##
##
##           Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                        265     357.72
## bio1       1   0.8278       264     356.89 0.362921
## bio4       1   4.3698       263     352.52 0.036581 *
## bio12      1   5.0356       262     347.48 0.024832 *
## landcover  5  17.1837       257     330.30 0.004164 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ANOVA table shows that temperature seasonality, precipitation, and land cover classes have a significant effect on the probability of presence of *B. bipinnata* in Fogo Island. This result aligns with the one obtained from the analyses carried out in the Module 2 on land cover classification where we notice a significant variation of the amount of introduced species among land cover classes.

After we have built our model, we can use it to predict the habitat suitability across the entire Fogo Island map.

```
# Get predicted values from the model
glm_predict <- predict(bioclim_kept, glm_model, type = "response")

# Print predicted values
plot(glm_predict, main = "Probability of occurrence of Bidens bipinnata")
```



We can also forecast the species distribution in the future using future climatic data. For more details see Oliver (2024).

Model evaluation

We now take that model, and evaluate it using the observation data and the pseudo-absence points we reserved for model testing. We then use this test to establish a cutoff of occurrence probability to determine the boundaries of the *B. bipinnata* range. In the following code, p argument stands for presence data while astands for absence/background data.

```
# Use testing data for model evaluation
glm_eval <- pa_evaluate(p = testing[testing$pa == 1, ],
                        a = testing[testing$pa == 0, ],
                        model = glm_model,
                        type = "response")
```

We determine a minimum threshold as the cutoff for converting the habitat suitability map predicted by the model into presence and absence.

```
# Determine minimum threshold for "presence"
glm_threshold <- glm_eval@thresholds$max_spec_sens
```

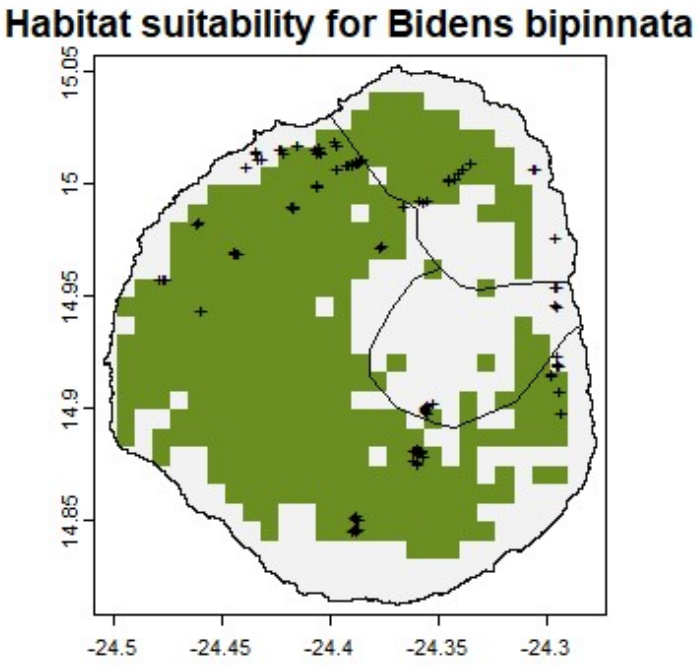
Finally, we can use that threshold to paint a map with sites predicted to be suitable for *Bidens bipinnata* in Fogo Island. Raster cells with 0 are set to NA while those with 1 are colored on the final map. After overlaying the observed points, we can see a few cells where the species was observed, but the model predicts them as unsuitable. This is known as *omission error* in machine learning.

```
# Plot base map
plot(map_fogo,
     axes = TRUE,
     col = "grey95",
     main = "Habitat suitability for Bidens bipinnata")

# Only plot areas where probability of occurrence is greater than the
threshold
plot(glm_predict > glm_threshold, # this generates a raster with 0 and 1
     add = TRUE,
     legend = FALSE,
     col = c(NA, "olivedrab")) # we provide different colors: 0 (NA) and 1
("olivedrab"):

# And add those observations
points(x = geom(obs_points)[,"x"],
       y = geom(obs_points)[,"y"],
       col = "black",
       pch = "+",
       cex = 0.75)

# Redraw the Fogo Island borders
plot(map_fogo, add = TRUE, border = "grey5")
```



Conclusion and perspectives

This tutorial presented a step-by-step workflow for building and evaluating the Species Distribution Model in R using *Bidens bipinnata* plant species as a case study. It implemented the generalized linear model (GLM) to predict the *probability of species occurrence* in Fogo Island. The tutorial used environmental data with different resolutions and projection systems to show you how to handle such complexity within spatial data used for building SDM.

However, other algorithms including Generalized Additive Models (GAMs), Maxent and tree-based approaches are well known in the literature to implement SDMs using background points as absence data. We encourage you to test different algorithms and select the best one based on performance metrics like the *area under the ROC curve (AUC), accuracy, precision, recall, etc*. You can also use the *k-fold cross-validation* technique for building and testing your models.

Note that recent developments of SDM suggested modeling a species distribution as an *Inhomogeneous Poisson Process (IPP)* which is implemented in the recent version of Maxent software. Technical aspects related to this modeling framework are beyond the scope of this tutorial and we recommend you to read the paper of Phillips et al. (2017) and the references therein to have an idea about the IPP framework.

References

Phillips, S.J., Anderson, R.P., Dudík, M., Schapire, R.E., Blair, M.E., 2017. Opening the black box: an open-source release of Maxent. Ecography 40, 887–893.https://doi.org/10.1111/ecog.03049.

Oliver, J., 2024. learn-r - A very brief introduction to species distribution models in R [WWW Document]. URL https://jcoliver.github.io/learn-r/011-species-distribution-models.html9 (accessed 9.25.24).