# HW 5 ~ Species Distribution Modeling

Due Wednesday, Oct. 19, 2022 at 11:00AM via GitHub

---

***Assignment:*** Like many analyses in spatial ecology, working with SDMs requires that you have skills to prepare the necessary spatial data sets. In this HW assignment, we will be fitting SDMs to our good friend the Austral grass tree (*Xanthorrhoea australis*). To do so, we will need to assemble two data sets: (1) species occurrence records and (2) environmental (bioclimatic) rasters. You will first prepare the bioclimatic rasters (from Worldclim) as in HW #2 (but using a different subset of variables as outlined below). You will then download (from GBIF) and prepare a *thoroughly cleaned* set of occurrence records for *X. australis*. We will divide the occurrence data into training and testing sets for model fitting (training) and evaluation (testing) and will check our candidate variables for potential issues with collinearity, removing any variables that are problematic before fitting models.

Data 'cleaning' is always critical and especially so for data downloaded from online biodiversity databases such as GBIF. Data cleaning typically involves removing duplicates, erroneous records (i.e., records with wrong geographic coordinates, outside of the native range of the species, low spatial precision, etc.). The goal is to produce a data set that is appropriate for (1) fitting SDMs (i.e., to avoid the old modeling adage: 'garbage in, garbage out') and (2) your particular research objective. One important step in the data cleaning process is plotting your data to make sure everything overlaps correctly in geographic space and generally makes sense. The necessary steps from HW #2 to prepare the data are repeated below, *some with minor modifications, so read carefully.* Refer to the HW #2 solution to make sure your code works correctly.

As always, you will be graded on your ability to produce clean, well commented R code that performs the tasks listed below without error. When you are done, push your code to GitHub, following the instructions provided in the document: `mees698C.submittingHW.pdf`.

Let's get started!

Figure 1: A stand of Austral grass trees in Warrumbungle National Park, New South Wales.

1. To reduce computational demands, we will be using rasters with a coarser resolution (grain) than we used in HW #2. Use functions in the `geodata` package to download the Worldclim bioclimatic variables at *5 arc-minute resolution*. Note that the resulting object will be of class `terra::SpatRaster` and I found it easier to convert it to class `raster::stack` and to also rename the layers. See HW #2 solution for how to do this if you are unsure.

2. In HW #2, we worked with four of the bioclimatic variables. Here, we want to consider more candidate variables:

   - Modify the raster stack of the 19 bioclim variables downloaded in step #1 to produce a new stack that contains all variables EXCEPT `bio1`, `bio8`, `bio9`, `bio12`, `bio13`, `bio14`, `bio16`, and `bio17` (in other words, retain `bio2-7`, `10`, `11`, `15`, `18`, and `19`).
   - Crop the resulting raster stack to the *outline of Australia* (not the extent) using the shapefile provided with HW #2. Rename the cropped rasters so they have the correct names (`bio2`, `bio3`, etc.). Depending on how you go about masking / cropping the rasters, you may end up with a raster with a global extent & many ros and columns of NA values. You may need to use the `raster::trim` function to remove these extra rows / columns. The `trim` function can be slow, so be patient.

3. Next, obtain species occurrence records for the Austral grass tree (*Xanthorrhoea australis*). As in HW #2, use the `gbif` function in the `dismo` package. Once downloaded from GBIF, clean the resulting data frame by removing records that:

   - Do not have geographic coordinates
   - Fall outside the native range of the species (southeast corner of Australia **only**)
   - Do not overlap the bioclimatic rasters
   - Have coordinate uncertainty greater than 10 km
   - Are duplicated
   - Are *spatial* duplicates

What are spatial duplicates? Spatial duplicates are observations that are close enough in geographic space such that they fall in the same raster grid cell, so spatial duplication depends on the resolution of the raster data. We generally do not want to fit a model using spatial duplicates.

**HW QUESTION: In general terms, how would you expect the resolution of a raster to influence the number of spatial duplicates?**

Removing spatial duplicates will be easier after you have converted the GBIF data into a `SpatialPointsDataFrame`. At the end of step #3, you should have a cleaned point occurrence data set with the correct CRS and containing only these attributes: `acceptedScientificName`, `lon`, `lat`, `coordinateUncertaintyInMeters`. Be sure to plot your data to check if everything seems OK. I ended up with about 730 points post-cleaning (Fig. 2). You may have a few more or less points, but your result should be close to this number.

After Step #3, you should have (1) a prepared set of bioclimatic rasters and (2) cleaned *presence-only* occurrence records. We need to do a few more things before we are ready to fit and evaluate models. First, we need to remove highly correlated variables and then divide the occurrence data set into training and testing sets. We will be fitting two presence-only SDM methods (Mahalanobis and Maxent). However, our evaluation metrics require absence data and so we will need to generate background points (pseudo-absences) for the model evaluation / testing.

Let's start with checking variable correlations. To do so, you will need to extract the values of the rasters at the occurrence records first (same as in HW #2 - see solution if you are unsure).

4. Use your cleaned point occurrence data to extract the bioclimatic variables from the raster stack. You should end up with a table similar to this (only first few rows printed):

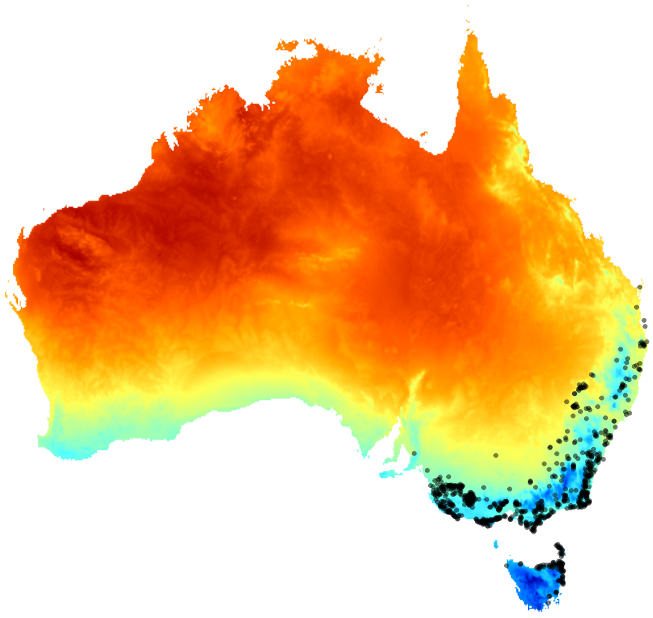

Figure 2: Cleaned grass tree occurrence records.

```
head(envDat)
```

```
##       bio2     bio3     bio4     bio5     bio6     bio7    bio10     bio11
## 1 9.106083 50.55845 322.4719 19.80600 1.795000 18.01100 14.24567  6.283166
## 2 9.411750 51.53734 305.2852 23.92200 5.660000 18.26200 17.50267  9.899834
## 3 7.420213 43.37956 332.3106 23.31170 6.206383 17.10532 17.84238  9.541135
## 4 8.849607 48.69031 321.6593 22.65059 4.475294 18.17529 17.10373  9.132941
## 5 8.520417 49.55459 315.2919 23.23600 6.042000 17.19400 17.71283  9.859000
## 6 8.564394 49.13722 320.9536 23.52159 6.092045 17.42955 18.01970 10.021401
##      bio15 bio18 bio19
## 1 15.68661   166   223
## 2 37.53335   107   295
## 3 21.24436   154   259
## 4 23.46360   115   208
## 5 22.90671   122   199
## 6 21.65250   120   185
```

5. Use the `vifstep` function in the `usdm` library to remove highly correlated variables from the raster stack. Create a new raster stack containing only the variables retained by `vifstep`. You will use this new raster stack to fit SDMs. I ended up with 6 uncorrelated bioclimatic variables - 3 temperature variables and 3 precipitation variables.

Now we need to (1) divide the occurrence data into training-testing sets and (2) create a set of background points for model evaluation.

6. Divide the presence-only data table into 80% training and 20% testing data sets (see `?kfold`). Make a plot showing the training and testing data as different symbols.

7. Create a `spatialPoints` object containing 10,000 random background (pseudo-absence) points. See `randomPoints` in the `dismo` package.

8. Use your training data and the uncorrelated set of bioclimatic rasters to fit and predict a Mahalanobis model (using the `mahal` function in `dismo`).

9. Make a map of the prediction (note that `mahal` predictions are slow, so it may take a few minutes to complete this step). The predictions from `mahal` are `1-distance`, so we will need to convert the distance predictions from the `mahal` function to a probability.

Here is some R code to do that - it takes as input (1) a raster of the raw prediction (called `mahalPred` in the code below) from the `predict` function and the stack of rasters used to fit the model (called `bioRastsKeep` in the code below).

```
# Convert distances to a p-value
# Mahal distances (D^2) are Chi-square distributed
probMap <- (1-mahalPred)
dists <- as.numeric(na.omit(getValues(probMap)))
p.value <- 1-as.numeric(pchisq(dists, df=nlayers(bioRastsKeep)))
probMap[!is.na(probMap[])] <- p.value
```

10. Use your training occurrence data and the uncorrelated set of bioclimatic rasters to fit and predict a MaxEnt model (using the `maxent` function in `dismo`). Use `jackknife` to assess variable importance. Make a map of the prediction.

**HW QUESTION: What are the top two most important variables associated with the distribution of the Austral grass tree? Which variable is least important?**

**HW QUESTION: Compare the predicted distributions from the two SDMs. How are they similar / different? Where do the models over- or under-predict the distribution? What might account for these model errors?**

11. Evaluate the `mahal` and `maxent` models using the testing data and background data.

**HW QUESTION: Briefly discuss the model evaluation metrics. Which model performed best? My AUC values for these models were quite similar even though their predictions were not. If you were a conservation manager and were provided output from these two models, how might you handle this seeming contradiction between the differences in the spatial predictions, but similarity in AUC?**

**HW QUESTION: How might you improve these models?**