# Species Identification/Selection

## Endries Aquatic Occurrence Data

Mark Endries, of the US Fish and Wildlife Service (Asheville Office) provided us with a shapefile of aquatic species occurrences. Each species was represented as a multipoint feature, tagged with the species common name, scientific name, and the number of occurrences in the dataset.

## Disaggregating the multipoint features and tagging them with catchment/HUC data

The Python script “EEP\_CreateHabitatModelInput.py” converts the multipoint features in Endries’ aquatic species occurrence dataset in to single point features, and then intersects these points with the NHD catchment polygons. It then adds new field, “Present”, and sets all values to “1”.

Next, the attribute table of this result is transformed so that each species scientific name becomes a column with the rows set to the NHD catchment FEATUREID. Values of ‘1’ indicate the species was observed in the catchment, values of ‘0’ indicates it was not.

*The file* [*Endries\_Data\_WithRank.xlsx*](https://github.com/Duke-NSOE/EEP/blob/SDMBranch1/Docs/Endries_Data_WithRank.xlsx) *(created by Audrey Archer) contains the a statewide listing of occurrence records compiled by Mark Endries, but with records tagged with NHD catchment ID (GRIDCODE), HUC IDs, Ecoregion, and NatureServe G-Rankings.*

## Identifying species with enough occurrence records to sustain modeling

The ideal species is one that is constrained geographically to a specific HUC6 or ecoregion, but that also has enough observed occurrences for adequate statistical analysis. To determine this, the script [SDM\_CountOccurrences.py](https://github.com/Duke-NSOE/EEP/blob/SDMBranch1/Scripts/SDM_CountOccurrences.py) creates a table that lists in how many HUC6s in which the species is found (to identify those found in just one) as well as how many occurrences occur within that HUC6.

# Generating model inputs

The inputs for the statistical analyses to see which habitat variables are statistically linked to species presence are derived from two primary datasets: the HabModel table listing in which catchments a given species occurs, and the EnvStats table listing the habitat attributes of each catchment within a HUC 6. To prepare these two files for analysis, they must be combined and thinned: thinned first to remove catchments well outside the likely habitat (e.g. to avoid searching for polar bears in the desert), and thinned again to remove habitat attributes that are either (1) not correlated with presence/absence of the species or (2) are strongly correlated, i.e. redundant, with other habitat variables.

## Selecting catchments to model and generating presence/absence data

The script [R\_CreateDataFile.py](https://github.com/Duke-NSOE/EEP/blob/master/Scripts/R_CreateDataFile.py) combines element occurrences for a given species with the environmental attributes at these locations. It also appends pseudo-absences comprised of all other catchments found within the HUC8s (within the HUC6) in which the species was observed.

The script first isolates all the catchment IDs in which the user selected species is found by selecting records from the HabModel table generated in the HUC 6 data assembly stage. It then creates a list of all the HUC 8s in which the species is found (within the HUC 6 being modeled). Then habitat attributes for all catchments within these selected HUC 8s are subset from the EnvStats table and a “Species” column is prepended to it. Values in this “Species” column are set to 1 in those HUCs where the species were observed and set to 0 where it was not (i.e., pseudo absences). (*Removing HUC8s where the species is not found narrows the statistical analysis to more homogenous regions, thus increasing the discriminatory power of the result.)*.

From this table, catchment attributes that have null or no-data values are removed. All others are included in a table written out as a CSV file. The result is a presence/absence table of NHD catchments and their attributes.

## Removing variables not correlated with presence/absence

The script [R\_SHCorrelate.py](https://github.com/Duke-NSOE/EEP/blob/master/Scripts/R_SHCorrelate.py) calculates Pearson product-moment correlations among Habitat variables included in the table created above and the binary presence/absence data. Those habitat variables found not to correlate at all (p > 0.05) are removed from the table.

## Identifying and removing redundant habitat variables

The script [R\_ScreenCor.py](https://github.com/Duke-NSOE/EEP/blob/master/Scripts/R_ScreenCor.py) computes the Pearson product moment correlations among pairs of habitat variables. Those with a coefficient > 0.7 are considered mutually redundant, and the script creates an HTML page displaying these mutually redundant variable pairs as nodes linked with a line (edge), with the thicker lines inferring a stronger correlation. Also, the size of the nodes reflect the strength of the correlation of the variable with the species presence absence.

The purpose of this html page is to allow the user to identify which of the redundant habitat characteristics to eliminate from the analysis. This can be based on a number of factors, for example the strength of correlation with presence/absence, or ability to affect the characteristic with management actions.

The end of this step should be a list of which habitat factors to include in modeling the species habitat preferences.

# MaxEnt Analysis

## SWD File

MaxEnt is run with a “Species With Data”, or SWD data file. This file is very similar to the CSV table generated above in that it lists all the habitat variables deemed relevant for all catchments. The key difference is that the speices column is recoded from 1’s and 0’s to the species name and “background”, respectively. Also, the “GRIDCODE” and “FeatureID” columns are renamed “X” and “Y”: the SWD format requires a locational identifier, usually an X-Y coordinate pair. However, as our analysis operates on a catchment - rather than pixel - scale, we can simply use the GRIDCODE-FeatureID pair as a unique identifier for each catchment.

## Projection Files

# MaxEnt Results

# GLM Analysis

* **R\_CreateDataFile.py** – Merges records for a species selected from the HabModel table and the EnvStats feature class into a single csv table ready for analysis in R. This table sets the species columns to “1” in the catchments where the species was observed, and “0” in all other catchments. The catchments included in the table is limited to only those in HUC8s where the species was observed.
  + Makes a list of the HUC8s in which the species was recorded
  + Extracts catchments records in those HUC8s from the EnvStats feature class
  + Joins species presences records to those extracted records
  + Writes all occurrence records to an output file, setting species = 1
  + Writes all non-occurrence records to an output file, setting species = 0
* Correlate spp presence/absence with habitat variables.

# GLM Results

Habitat Model Screening

The inputs for the statistical analyses to see which habitat variables are statistically linked to species presence are derived from two primary datasets: First is a table listing the NHD catchments in which the target species was found, and second is another table listing the ~120 different attributes of each catchment within a HUC 6. To prepare these two files for analysis, however, both the rows (catchments) and the columns (catchment attributes) must be screened so that statistical analysis is robust and meaningful.

First, we remove catchments from the analysis that fall well outside the likely range of the species. This improves the discriminatory power of the model by focusing on more subtle ranges in catchment attributes. (For example, if you avoid searching for polar bears in the desert, you can focus on the more meaningful landscape features). We do this by limiting our analysis only to catchments within the HUC8 in which the species was observed. (For example, if a species was found in only 13 of the 25 HUC8s within an ecoregion, we use the just the NHD catchments within those 18 HUC8s to determine which attributes are linked to species presence.).

Next, we screen the number of catchment attributes that go into the statistical model. First we eliminate any attribute not found to be statistically correlated with species presence. Those catchment attributes shown to have no significant correlation (p < 0.05) with whether a species is presence or absent in a catchment are dropped from the analysis. Then, we screen out redundant catchment attributes, i.e., those that are correlated to each other. Here, we calculate the correlation coefficients among attribute pairs and then manually examine correlated pairs (R2 > 0.7) to determine which of the two should be dropped from the analysis. The decision on which to drop can be based on a number of factors, for example the strength of correlation with presence/absence, or the ability to affect the characteristic with management actions.

In the end we have produced a list of relevant catchments, each tagged with whether the species was recorded present within it or not, and with a value for each of the pertinent, non-redundant catchment attributes. This list serves as the input for three statistical models – a generalized linear model (GLM, a maximum entropy model (MaxEnt), and a random forests (RF) model – which will be used in an ensemble approach to predict habitat likelihood of a given species.