**Temporal analysis data pre-processing procedure**

Mark A. Hayes

8/25/2017

- Download the data for a species from the EDDMapS dataset. Keep a copy of this dataset with the name: "species_eddmaps_master_date.csv". For example: "pymo_eddmaps_master_04042016.csv"
- Save this file as a new csv titled "species_noDups.csv". e.g. "pymo_noDups.csv"
- Remove all extraneous columns, but keep objectid, year, species, verification_method, identification_credibility, and lat and lon coordinates. Note, the location coordinates are lat then lon. Separate date into multiple columns and save year. Also separate lat and lon into multiple columns.
- Sort by identification credibility and remove all but credible and verified.
- Sort by lat and lon and remove records without lat and lon values.
- Add columns as necessary and calculate difference between one row's lat and lon and the next row's lat and lon, and calculate the distance in decimal degrees between points using
    - Distance = sqrt(lat_diff^2 + lon_diff^2)
- Assume that 111 km is very approximately equal to 1 decimal degree. This is the case in Florida. For lat: 1 DD at equator = 110.567 km, and 1 DD at pole = 111.699 km. Thus 1 km ~ 0.01 DD.
- Add a logical operator column and ask if the dist_diff_x_y cell is < 0.01, using, e.g., "=B4<0.01".
- Using a filter, select all TRUE. Make these cells bold and a background of yellow to make easier to see.
- For each yellow group including the row above it, delete all rows but the earliest row. This may take a while. If growing impatient, relax and make some tea and/or go for a trundle.
- Clear all bold and yellow background.
- Repeat the "TRUE" logic query. Use a filter to find any TRUEs.
- Delete the appropriate row for each true.
- Sort by year.
- Remove 2017 records, because this is not yet a full year.
- This process, if done carefully, effectively removed duplicates and thins the data to occurrence records that are > 1 km apart. This can likely be done in R, so in the future keep an ear to the ground and/or look into automating this procedure.
- To the right side of the data list the years from first occurrence through 2016.
- Create a "countif" function for each year to count the number of items in each year.
    - =COUNTIF($B$2:B$196,J20)
- You are now ready to create a csv data frame with the number of occurrences per year and proceed with polynomial regression. Name the csv using the following approach, such as for pymo: "pymo_uniqueLocs.csv".
- For each species, upload two files to GitHub, as csv's: the master file and the uniqueLocs file. Also upload this procedure to the project.