Data Science II

White Paper

Madison Harris

Introduction & Background

The desert dace (Eremichthys acros) is a thermophilic cyprinid endemic to a small hot spring system in Soldier Meadows, Northwestern Nevada, Great Basin physiographic province. Due to their long history of isolation with no exact time frame known, desert dace are considered to be a relict species (Hubbs and Miller 1948). The species was listed under the ESA in 1983 (Federal Register on June 14, 1983 (48 FR 27273) and the primary threats include habitat alteration leading to habitat loss and degradation from livestock grazing, water diversions and invasive species. Desert dace have the highest temperature tolerance of any minnow in the western North America (Nyquist 1963) and are found in waters ranging from 40.5°C to 18°C; individuals will behaviorally thermo-regulate by moving away from spring heads when temperatures exceed this comfortable range (Duncan 2010). Therefore, cooler waters downstream of springheads represent critical habitat. Climate change leading to decreasing and/or more variable surface water resources is likely to further impact long term persistence probability of the desert dace. Here I propose to characterize the population genetic structure, gene flow, genetic diversity, and effective population size of the extant desert dace populations. My main objectives are to assess: 1) whether there is evidence of movement among springs, and 2) characterize the environmental correlates of observed patterns. Little is known about this species

To begin to accomplish this, I aim to use USGS streamflow data downloaded from the National Water Information System: Web Interface to build predictor models and to aid in later study. This will help guide me later to create temperature gradients of the hot spring systems for conservation purposes and predict locations of other viable habitats.

Methods

Using the USGS website and integrated mapping tool, I have located approximately 30 stream and spring gauges with relevant information, ranging from water temperature, air temperature, streamflow velocity, discharge, and chemical concentrations, to name a few. Site name, site number, latitude and longitude location, status, elevation, and date range have been pulled and placed into a .csv file for ease of access, which has already been cleaned and properly formatted using OpenRefine; see OpenRefine homework in my Bio792\_Shared GitHub. Data for each gauge will be pulled from USGS via the Surface-water Daily Data for Nevada site via site numbers and the R package dataRetrieval. OpenRefine will be utilized to clean up and organize the data as needed. Pandas via Jupyter Notebook will be used to create a for-loop code to organize and combine the multiple .csv files into one or more files, such as separating water temperature from all the USBLM DESERT DACE SW TEMP gauges into one file. Commands used will include pd.read\_csv(), pd.concat(), usecols= , sort\_values(), pd.DataFrame.to\_csv(), and others.

A stretch goal, which may or may not be included in the final paper will be to repeat the above steps with niche climate data pulled from various sites and sources that will eventually be put into ArcGIS for modeling purposes. I would be using Pandas and OpenRefine to clean up and organize this data into a format that can be ported easily into GIS.

Results

Issues were encountered with the USGS website portal for downloading the data and in the R package. The R package dataRetrieval allows for instant download of a single parameter from a single gauge directly into R for graphing and analysis purposes, and I used the readNWISuv command multiple times to do so and get figures such as Fig 1.

OpenRefine was used to clean up and organize the file USGS\_streamflow\_gages.csv, by standardizing the dates taken from the website and formatting the site\_number column to the Number format. Transform commands

value.replace(/[^\u0020-\u007F]/,""); value.replace("'", ""); value.replace('"', "")

Chart

Description automatically generatedsuccessfully removed special characters from the Latitude and Longitude columns, turning them into a string of numbers. These columns were separated by field lengths and the python code lat\_long\_decimal\_conversion.ipynb converted these separated strings of Degree/Minute/Second into decimal points to standardize.

Python scripts merge\_files, separate\_files, and isolate\_sample\_location.ipynb were written and used to merge separated location files made in Google Earth into one file and pull apart this new file according to location for later usage, respectively.

Figure 1: Water temperature in \*C, from Oct 2020 to 2021. Measured by USBLM Desert Dace SW Temp 12, one of 20 streamflow gauges operated by the USGS in Northern Nevada, Soldier Meadows. Site number: 412044119121701. Parameter code: 00010. Acquired with package dataRetrieval.

Discussion

While retrieving the data I have is critical to my research down the line, using these methods highlighted their shortcomings and where I can grow and lean more in the future. The R package dataRetrieval is a great boon to my time, by cutting out the middleman as it were and eliminating the need to clean up and pull values from the .txt files downloaded from the USGS portal. It is, however, lacking in a few glaring ways. There is no clear, simple way to fetch multiple batches of data at once, from multiple sites within a specific area at once. According to a few people I have spoken to about this issue, there is a workaround for this by creating a vector of the site numbers or site names and using that in the code. I have not attempted this yet, but it is certainly something to try in the future. For now, I am content with the current code.

As for the work with my sample files, my primary goal was to standardize the files I already had, to make it easier on myself when sampling more in the future. There are websites for changing DMS coordinates to degree coordinates but using those in the long run would require more work than running an OpenRefine saved edit code and a python script to change over the data. And for the Jupyter Notebook scripts that pull and separate files, I plan to change them over into actual python .py scripts that can be run from the terminal and format the commands into for-loops. When I collect more samples over the summer, and need those files formatted and separated for usage in ArcGIS and RStudio –and later in future genomic analysis— having that code that will separate samples by site and then by subsite with a few keystrokes will be very helpful.

Citations

Duncan, Doug. (2010). Desert Pupfish (Cyprinodon macularius) 5-Year Review: Summary and Evaluation. 10.13140/2.1.1563.9364.

Hubbs, C.L and R.R. Miller. 1948. Two new, relict genera of cyprinid fishes from Nevada. Univ. Michigan. Mus. Zool. Occ. Pap. 507:1-30.

Nyquist, D. 1963. The ecology of Eremichthys acros, an endemic thermal species of cyprinid fish from northwestern Nevada. M.S. Thesis, University of Nevada. Reno.