# Preparing Elevation Data Using "aggregate.py"

Andrew Kenneth Melkonian

May 25, 2015

---

**aggregate.py**

---

**Description:**

"aggregate.py" reads five-column ASCII text files (longitude, latitude, elevation, date in decimal years, uncertainty), with each file corresponding to a DEM, with the path to the reference DEM specified separately. It creates five-column ASCII text files as output, with the columns the same as the input DEMs. The data in the output files is organized on a pixel-by-pixel basis, with pixels separated by a single line containing the greater than character (">", e.g., GMT polygon file format). This script prepares the input for the "weightedRegression.py" script.

**Dependencies:**

**Python**

**Usage:**

python aggregate.py reference_dem_txt_path input_dem_txts_directory identifier increments

**Example:**

python /path/to/ref_dem.txt /path/to/inputs/ app 5

**Input Parameters:**

**reference_dem_txt_path:** Path to 5-column ASCII text file of longitudes, latitudes, elevations, date in decimal years, and uncertainties for reference DEM.

**input_dem_txts_directory:** Path to directory containing input DEMs. The ASCII text file for each input DEM must have the same format as the ASCII text file for the reference DEM.

**identifier:** String value, input DEMs in input_dem_txts_directory must end in "identifier.txt" to be read in.

**increments:** Must be a number; the output will be split into this many files (e.g., "glacier_ice_values_1.txt", "glacier_ice_values_2.txt" if increments is "2").

**Output:**

ASCII text file(s) with all of the data organized by pixel (pixels separated by single lines containing ">" character). The "increments" input parameter determines how many of these will be made. Each file has five columns:
1) Longitude (floating point)
2) Latitude (floating point)
3) Elevation (floating point)
4) Decimal year (floating point)
5) Uncertainty (floating point)