QGIS Data Cleaning, Normalization, and Aggregation

*This document outlines the steps required in QGIS to prepare data ahead of visualizations and machine learning. These steps, as well as the Python scripts, have been tested for QGIS version 3.4.4-Madeira, with Python version 3.7.*

*These instructions use a method that divided the farm field into 20 meter by 20 meter squares. If desired, the user may use the alternative 40 meter by 40 meter division, in which case they will run the “clean\_normalize\_40mete\_stdDev.py” script file instead.*

Prerequisites:

1. QGIS 3.4.4-Madeira must be installed. Version 3.6.4-“Noosa” may include code-breaking changes, though this has not been tested.
2. “clean\_normalize\_20mete\_stdDev.py” must be saved on your computer.
3. Yield data must be stored locally on your computer in ESRI Shapefile format. Additionally, the file must have field attributes named exactly as follows:

Product, Yld\_Vol\_Dr, Elevation\_, Moisture\_\_,

The order doesn’t not matter, and other fields may be included in the attribute table.

Step 1: Open QGIS, and click on the “Processing” option on the top tab. Click “Toolbox”. The Processing Toolbox pane will appear.

A screenshot of a social media post

Description automatically generated

Step 2: Click the Python icon at the top of the toolbox, and click “Open Existing Script”. Open “clean\_normalize\_20mete\_stdDev.py”. The script editor will appear.

A screenshot of a social media post

Description automatically generated

Step 3: Beginning at line 10 in the code, you must define the absolute path to each of the .shp files in the “inputLayersPaths” dictionary array. Use the year in quotations as the key to each entry.

A screenshot of a cell phone

Description automatically generated

Step 4: Directly below the “inputLayersPaths” dictionary definition, you must define the “finalGriddedFolder” variable to the path name of the folder you want the program to write the final cleaned, normalized, and aggregated data layer to.

A screenshot of a social media post

Description automatically generated

Step 5: Save your changes if you wish, and then press the green button at the top of the script editor to run the code. This processing can take quite a while to run. On a ten year-old computer, it takes approximately 45 minutes to perform all the processes for 13 years-worth of harvest data. While the code is running, the entire QGIS program becomes responsive, so do not be alarmed by this.

A screenshot of a cell phone

Description automatically generated

Step 6: After the code has finished running, go to the folder defined by the “finalGriddedFolder” variable. In this folder, you will find several .gpkg files, as well as one .csv file. This .csv file contains all cleaned, normalized, and aggregated data, for all years.

A screenshot of a social media post

Description automatically generated

This .csv file contains statistical information on the dry yield, elevation, and soil moisture.

\*\*NOTE\*\*

The fid field is automatically created by the program, and has no geospatial significance. However, the “id” field does have geospatial significance. Each grid square is labelled starting from the top left down, then returning back to the top to start a new column. Also, only the grid square that contained any yield points are represented in the file, so there may be “gaps” in the indices.

Before this data can be used in any subsequent steps, this data will need to be exported as a .csv file. To do this, right click the “FINAL ALL YEARS BINNED FINAL” layer in the layers panel. Hover over the “Export” option, and click “Save features as” .

Additionally, the attribute table contains many columns that are unneeded in further steps of our data pipeline, as well as unneeded. The documentation for each step (PowerBI and Azure Machine Learning) will need to be consulted, and the proper columns renamed, before these steps can be carried out.