ArchaeoAstroInsight

User Manual

Installation and Use Guide

Version 0.1.0 – October 2021

Roxana Ionescu, Marc Frincu

Table of Contents

Overview

ArchaeoAstroInsight (A2i) is a Python script developed by Roxana Ionescu and currently maintained by Marc Frincu. It was initially a BSc final year project proposed and supervised by Marc at the West University of Timisoara, Romania. The icons and logo have been designed by Teodora Marisescu. It is designed to be used in the **QGIS platform** to **automate the initial archaeoastronomy analysis researchers perform** on archaeological sites by using Google Earth. The script relies on www.HeyWhatsThat.com and SkyScapeR to calculate the altitude of the horizon in a given azimuth direction. In addition, it uses the Yale Bright Star Catalog to identify potential stars in the direction of the orientation.

After configuring the parameters and entering location of the site, the user can draw a line by selecting 2 points. A2i will automatically compute the **azimuth**, **altitude**, and **declination** of the bearing. In addition, it **identifies if the bearing points to major lunar and solar points** (solstices, lunastices, equinoxes) **and if any bright stars rise from its direction**. A2i assumes an error of ±0.5°, which is roughly the same as that of a visual observer.

Disclaimer

A2i is currently under tests. Theoretically its accuracy is equal to that of the tools it relies on (QGIS for azimuth and HeyWhatsThat.com for altitude). The software is not guaranteed to be error free.

Installation Steps

Install required software (QGIS and R)

1. Install **QGIS** (Script tested for version 3.18) https://qgis.org/en/site/forusers/download.html
   a. There are two options to installing QGIS: **standalone** or through the **OSGeo4W** package
   b. Pick **Standalone** if you only wish to install QGIS

Latest release (richest on features):

⬇ QGIS Standalone Installer Version 3.18 (64 bit) ↗
sha256 ↗

⬇ QGIS Standalone Installer Version 3.18 (32 bit) ↗
sha256 ↗

c. Pick the **OSGeo4W** if you want to install a whole package of GIS applications (check their website for more info https://www.osgeo.org/projects/osgeo4w/)

2. Install **R** (any version) https://cran.r-project.org/bin/windows/base/

Download R 4.1.0 for Windows (86 megabytes, 32/64 bit)

Installation and other instructions
New features in this version

Install A2i

There are currently two options to install/run A2i: as a QGIS plugin or directly be using the provided scripts. Below we explain each step (notice there are two archives a zip one for the plugin another for the script version; this is to avoid confusion for people not familiarized with the process):

Install A2i as a plugin

3. Download the **a2i.zip** (https://github.com/marcfrincu/ArchaeoAstroInsight) and **unzip it** to where the QGIS Python plugins are located. You can find their location from Settings → User Profiles → Open Active Profile Folder. In the window that opens you will see a folder called *python* containing a subfolder *plugins*. It is there that you must unzip the archive. You should have a folder called a2i containing all the required files. Restart now QGIS. You should see the A2i plugin should appear under Plugins.
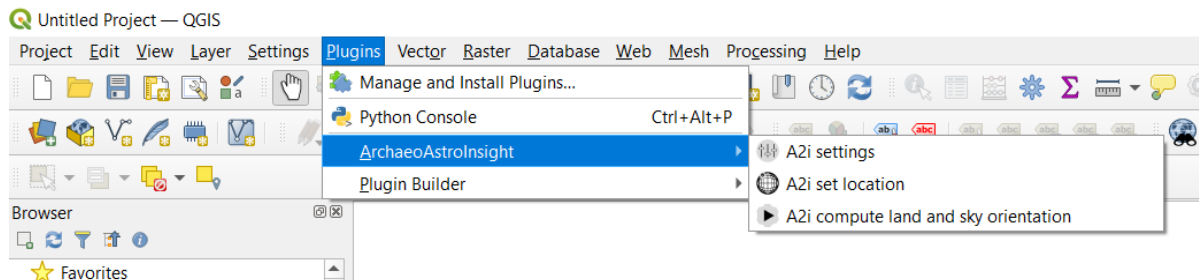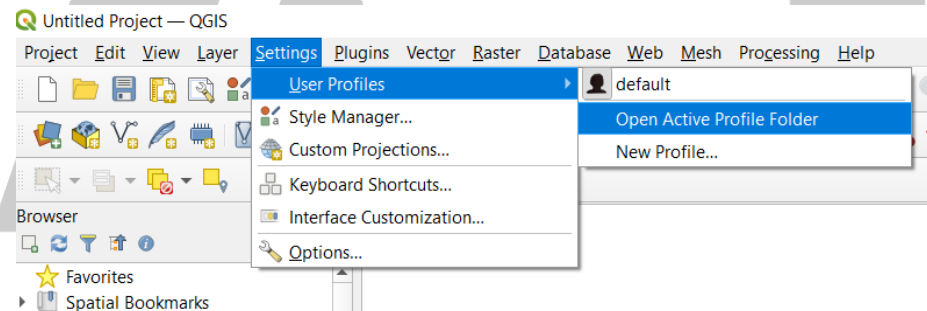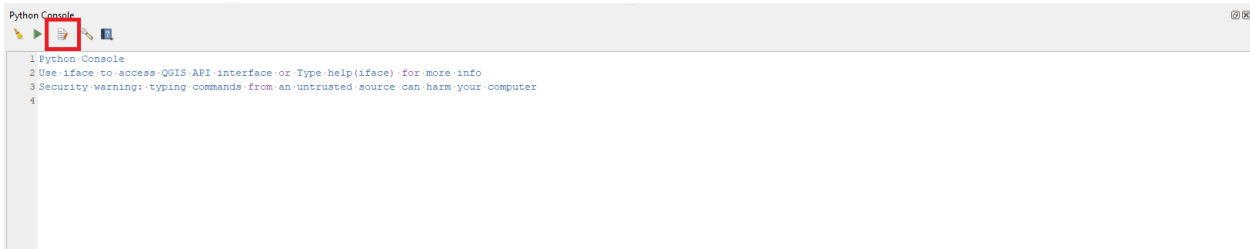
4. Extract the files from **ArchaeoAstroInsight.zip**
(https://github.com/marcfrincu/ArchaeoAstroInsight) to a folder of your choice. right-click on
the file and click **Extract to ArchaeoAstroInsight/**
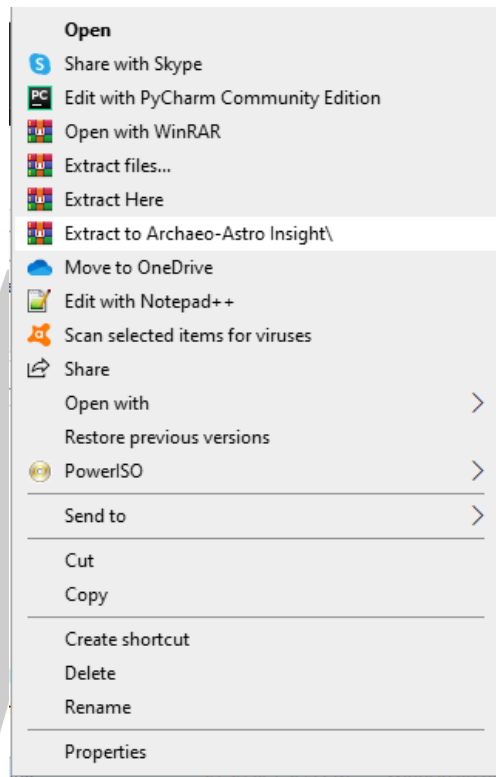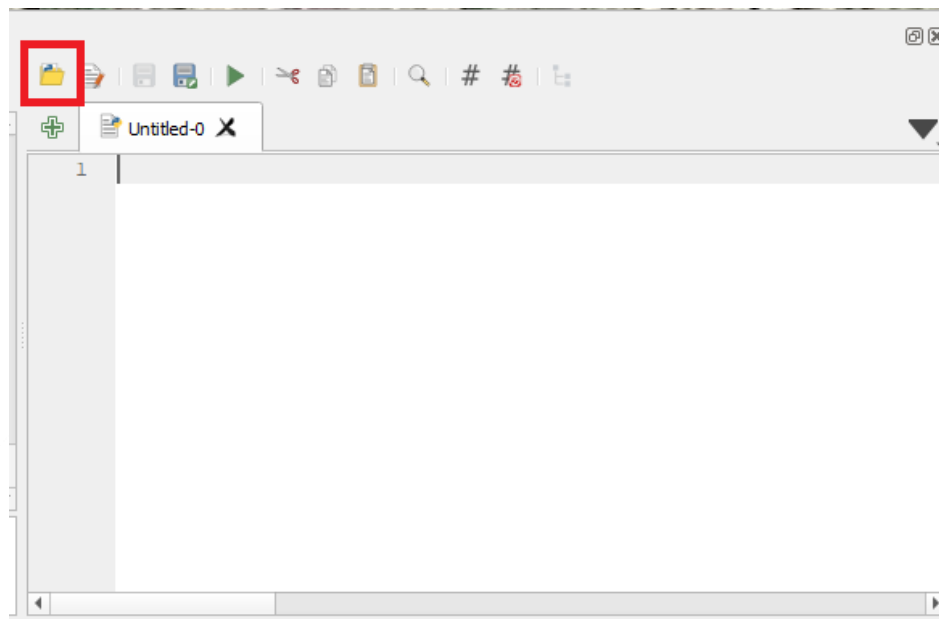

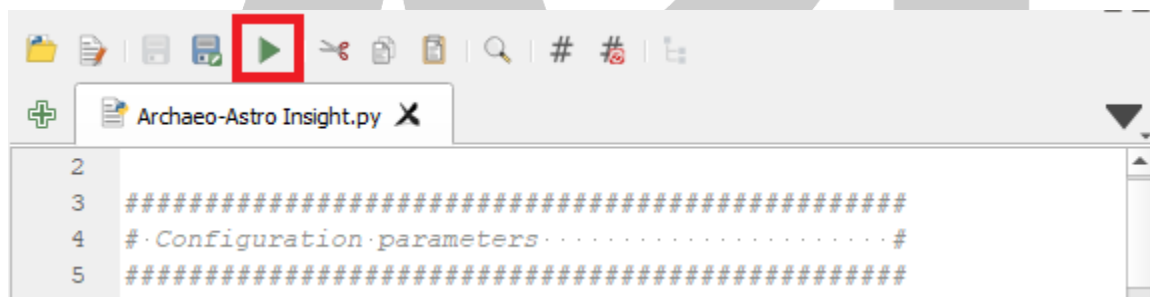5. Open **QGIS** and open the script
   a. Open the **QGIS Python Console**

   b. Click the **Show Editor** button

   c. Click the **Open Script...** button

   d. Go to the folder where you saved the script and select **get_path.py**
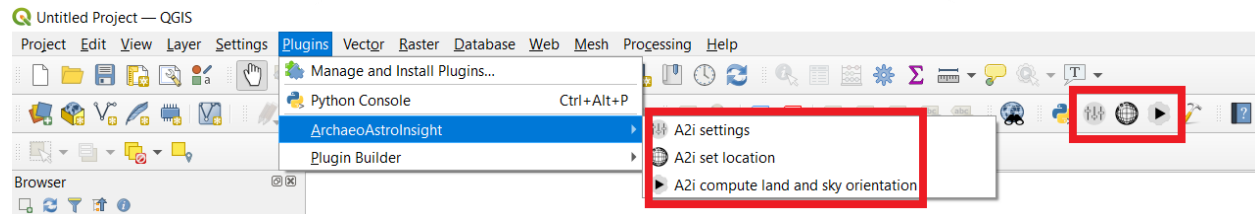
6. Now press the **Run Script** button

7. Repeat step 5 but for **Archaeo-Astro Insight.py**

User Guide

If installed as a plugin, A2i should appear both in the top menu under Plugins and in the toolbar. If it was installed as a script, only the toolbar options will be available. It contains three options:

Plugin/script settings	
Location setup	
Computation of land and sky orientation	

We must first **perform the initial configuration**. On the toolbar, you will see 3 buttons. Press the **Settings** button. A window will pop up. Following is an explanation of all the parameters:
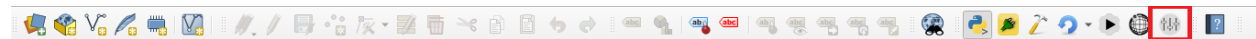
a. **Results Path** – this is the default location where you save the data output (can be any existing folder you want).
b. **Rscript Path** – this is the path to Rscript, it comes with the R installation. Examples of paths depending on your Operating System (the R version differs based on your installation):
    i.   **For Windows:** C:/Program Files/R/R-4.0.5/bin/x64/
    ii. **For MacOS:** /Library/Frameworks/R.framework/Versions/4.1/Resources/ or /usr/local/bin/
    iii. **For Linux:** usr/local/bin/ or /usr/bin/

c. **Script Sleep** – this value is used to determine the amount of time the script must wait before it can access the generated data. If the value is too small, it might fail. Check **Possible Errors** section for more details.

d. **Download Map** – this specifies if you want to download a Google Maps satellite image to your project. By default, this is checked, if you don't want a new layer when you run the script, uncheck it and delete the downloaded layer.

e. **Line Width –** this specifies the thickness of the line you draw on the map to be (more on this on the user guide).

f. **Map Type –** the type of satellite image to download.

This setup is only necessary on the first use, you can use the script without this step on future occasions and only update the fields if you wish to change something.

There are two functionalities to this script/plugin: zooming to specific geographic coordinates (latitude, longitude) on the map, and computing the bearing of a given orientation.

If you click **Location setup** a window will open. There you can enter the geographical coordinates for a point on the map. The order of the coordinates is **latitude,longitude**, without any spaces. Coordinates with up to 4 decimal points can be introduced.


Example: Coordinates for Stonehenge

1. The simplest way to **obtain coordinates** is **from Google Maps** (https://www.google.com/maps).
2. Enter the name of the location, in our case Stonehenge and then click on a point on the map close to the desired location.
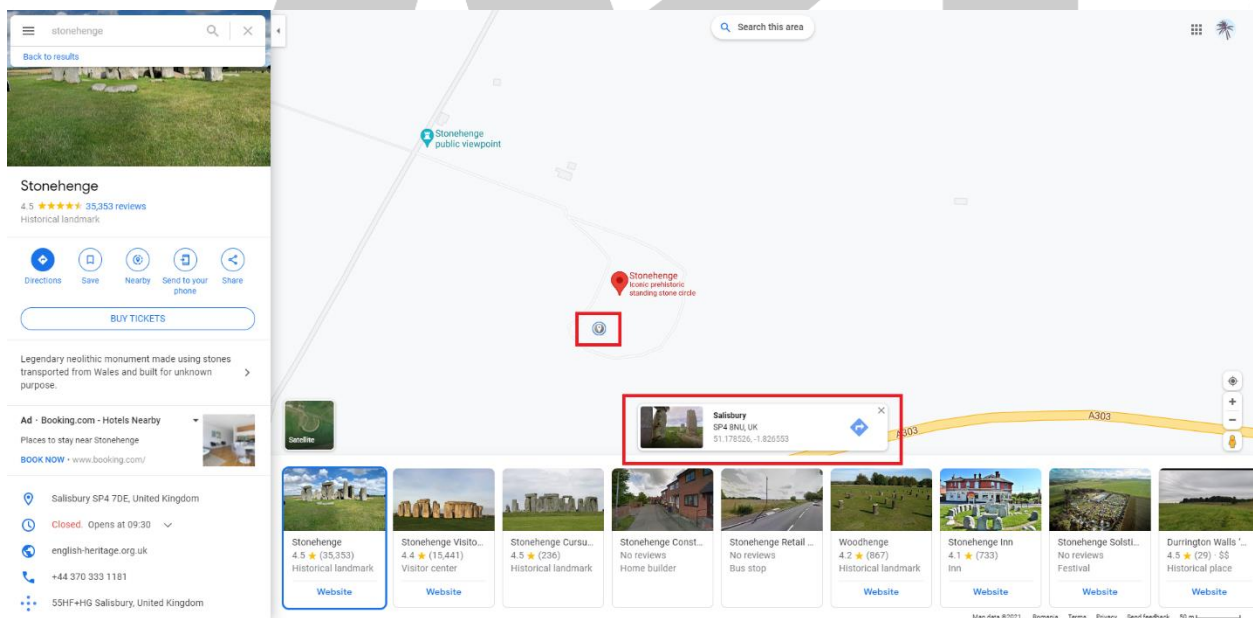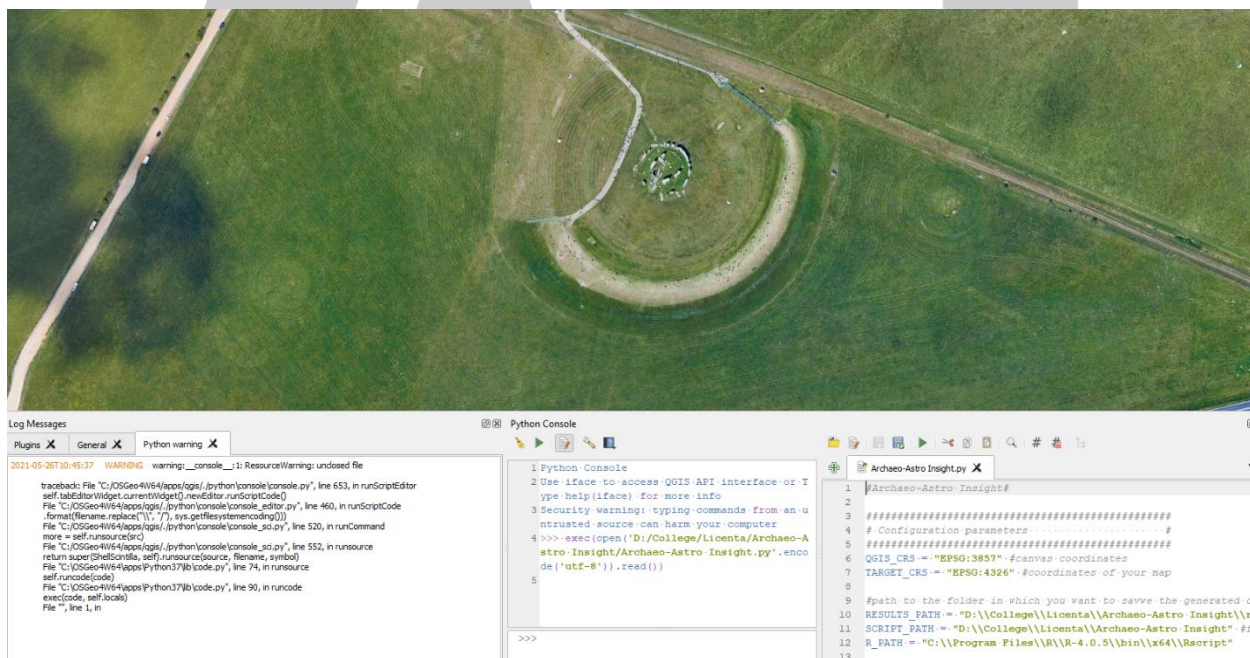

3. Click the coordinates in the rectangle at the bottom of the screen (see previous picture) and then copy the coordinates from the left side of the screen (use the decimal ones, not the ones in degrees, minutes, seconds).
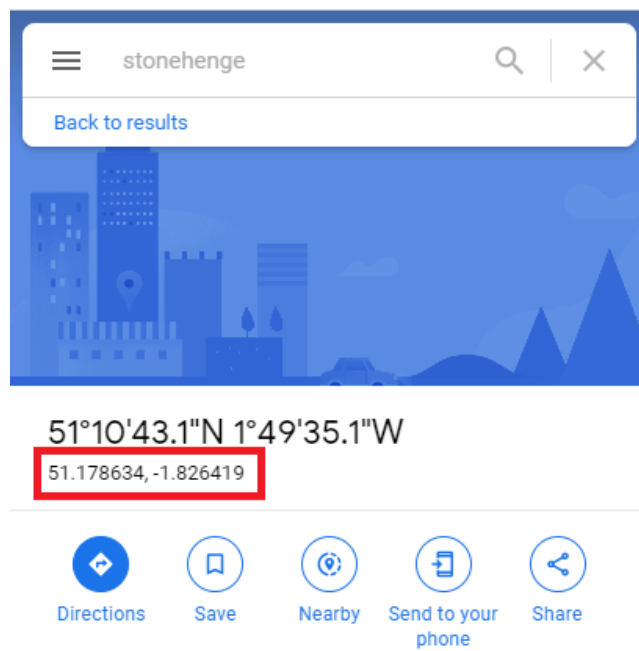
4. Copy these coordinates and paste them into the Enter Coordinates window in QGIS
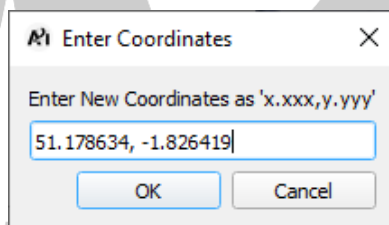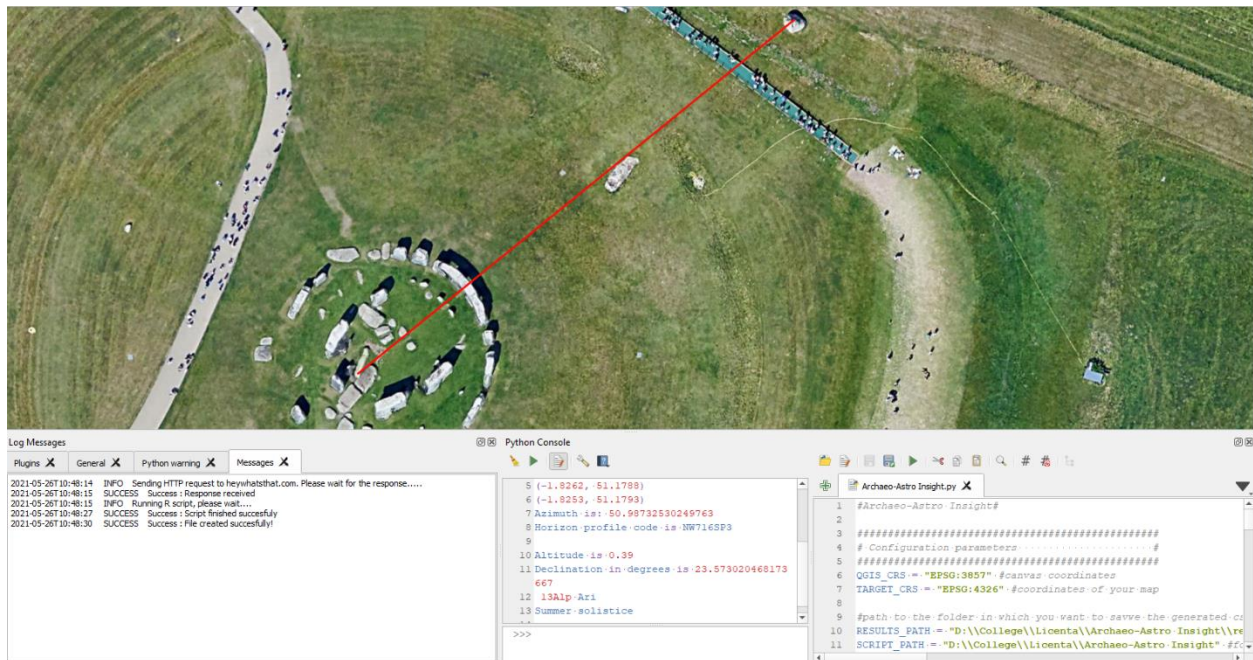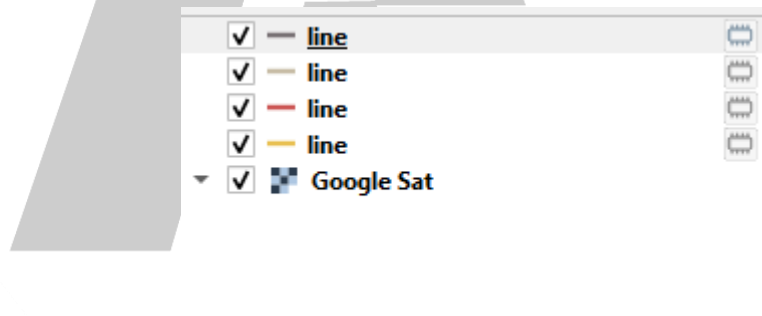

5. You're at Stonehenge now!

If you click **Compute land and sky orientation**, you will be able to draw a line and compute the azimuth, altitude, and declination of its bearing. Simply click on two points on the map and wait for the values to be computed. The line between the two will appear automatically.

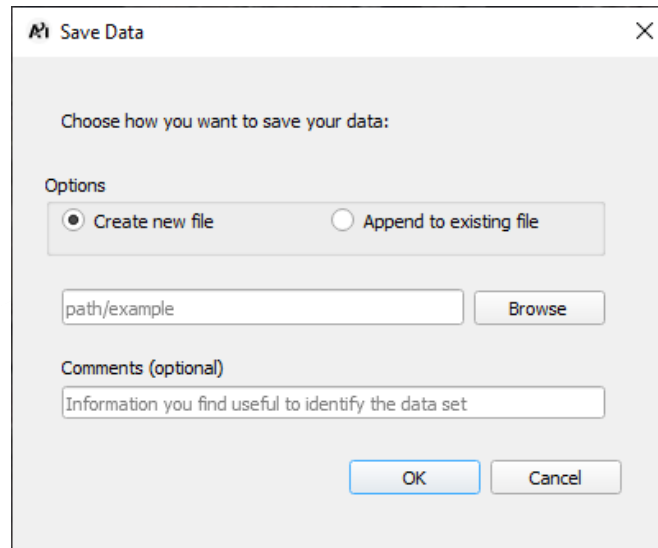Lines drawn on the map are saved as separate layers:

If you hover over a line for a few seconds, you can see the coordinates of the points. You can also delete any line without affecting the other lines.
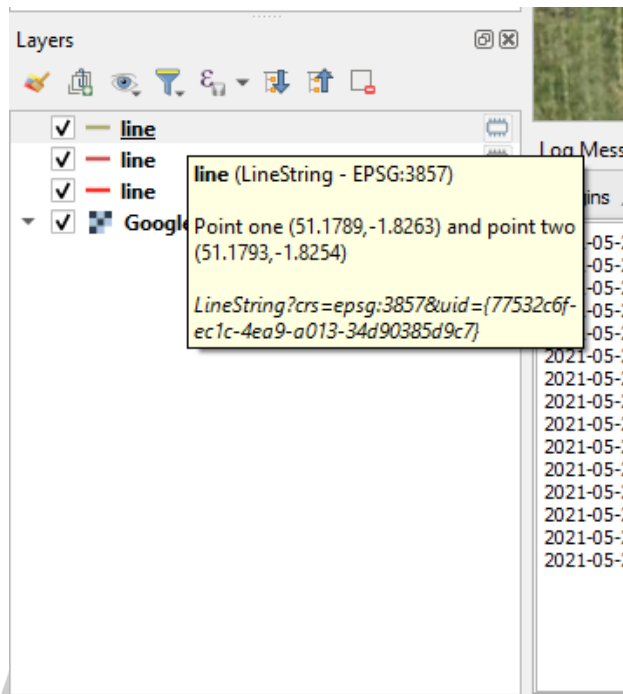
If you want to change the line width, change the value of the **Line Width** parameter from the **Settings** button.

At the end, you will be given the option to save the results as a .csv file. The results are also visible in the console as they are generated and as a message in the GQIS interface.
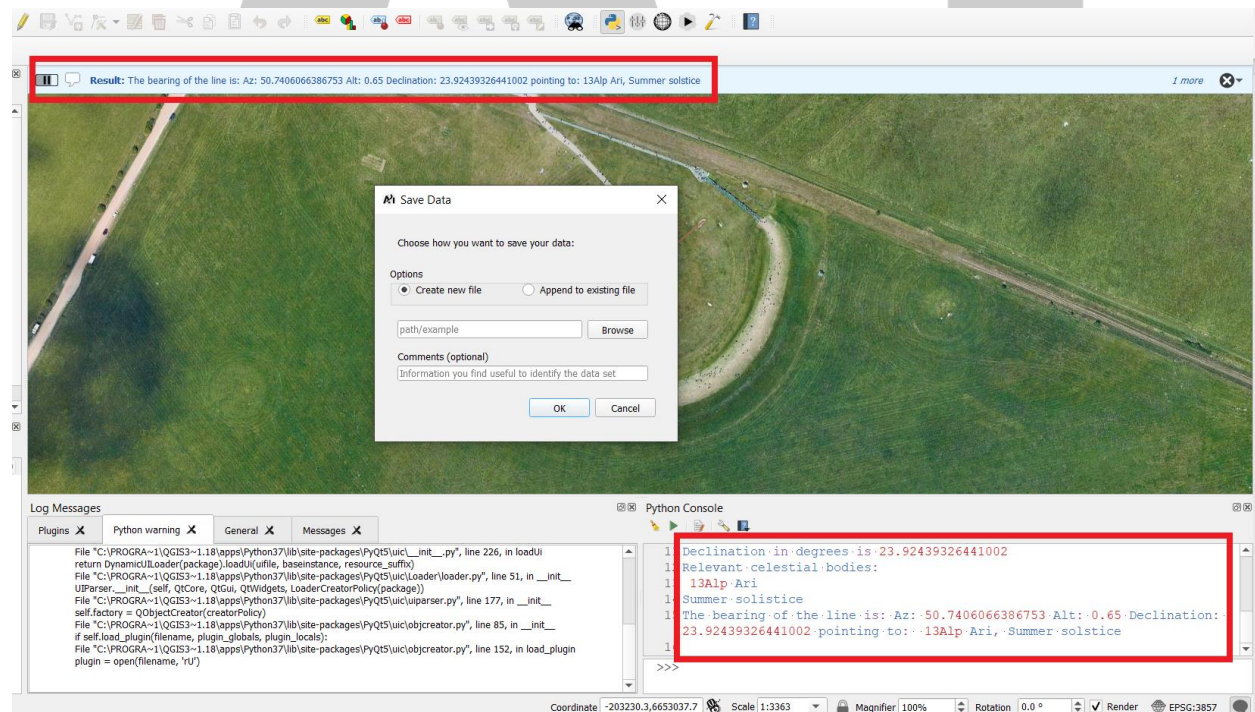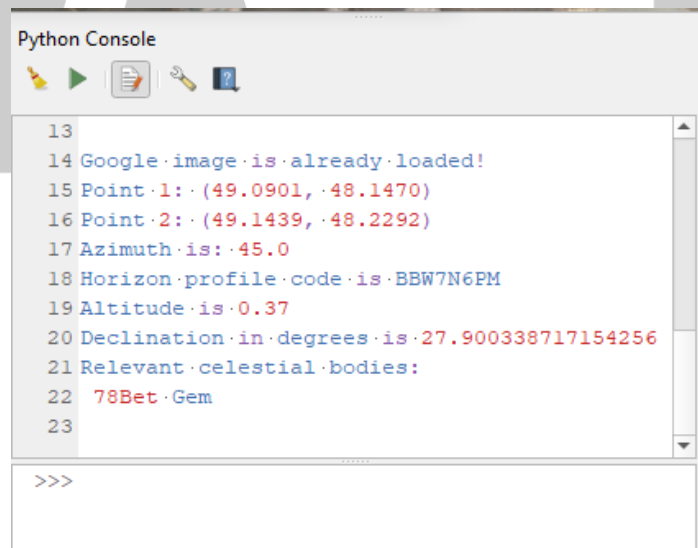
The Save Data window allows you to:

    a. **Create new file** – if this is checked, you will select the path where you wish to save your data and a **new .csv** file will be created. The default saving path is the one set during **Settings**

    b. **Append to existing file** – if this is checked, you will select an already existing .csv file and add the newest data at the end of that file. **You can only select .csv files!**

    c. **Comments** – you can add comments specific to the data you wish to save. You can leave this blank, as it is optional.

If you click cancel, the data will be discarded, and you will have to draw the line again or copy the data from the console yourself.

Known Issues

When generating the altitude of the point, the script gets the information from HeyWhatsThat.com automatically. Unfortunately, sometimes the data is not ready in time and the script cannot compute the value. This will cause the script to fail. In this case, you can try and increase the **Script Sleep** value from **Set Params** and **rerun the script** (as described in the installation steps). This will give more time for the data to be generated.

If you enter coordinates in the **Location setup** screen and your map turns white, try zooming out using your mouse wheel. If the coordinates you have chosen happen to be in the ocean, the zoom distance of Google Satellite is smaller so you will have to zoom out in order to see your location.

Sometimes if errors occur, the first point will not be reset. In this case only Point 2 will appear in the Python console and no line will be drawn. If this happens click again the **Compute land and sky orientation** button.