SENTINEL 2 IMAGE CLASSIFICATION: BELIZE

Date: 22-03-2021

By

Stephen Carpenter,

National Oceanography Centre,

Southampton

DOCUMENT RELEASE SHEET

	Name	Organisation	Signature
Authors	Stephen Carpenter	NOC	

CHANGE RECORD

Version	Purpose of Update	Date
1.0	First copy of instructions	15/02/2020

Contents

DESCRIPTION

These instructions were produced in association with the *Virtual Workshop CMEP: Advanced scientific techniques to inform integrated coastal zone management Sessions* as part of session 5):

Marine Survey Tools part 2: How do you map the shallow water environments in the coastal zone? Estimating bathymetry using satellite and in-situ data within ArcMap and Google Earth Engine

Session Description

This session will showcase supervised classification of benthic habitats using the freely available Google Earth Engine: a powerful cloud computing tool to which holds a multi-petabyte catalogue of satellite imagery ready for rapid analysis. We will cover how to visualise, explore, process and download data within a simple supervised classification using Sentinel 2 imagery. Training and validation data will be created to carry out an accuracy assessment and quantify the performance of the classification.

Requirements

- Google Earth Engine account (free) - Sign up to Earth Engine using the following link: **https://accounts.google.com/signin/v2/identifier?service=ah&passive=true&continue=https%3A%2F%2Fuc.appengine.google.com%2F_ah%2Fconflogin%3Fcontinue%3Dhttps%3A%2F%2Fsignup.earthengine.google.com%2F&flowName=GlifWebSignIn&flowEntry=ServiceLogin**
- Bathymetry data – CSV or shapefile of points

The document explains the following scripts developed in Google Earth Engine accessible via the following link (https://code.earthengine.google.com/?accept_repo=users/stcarp/CME_MappingWorkshop):

1. Visualise Sentinel 2
2. Sentinel 2 difference
3. Create Land mask
4. Estimate Bathymetry Lyzenga / Stumpf
5. Create Training Data
6. Classification

Layout of Google Earth Engine and scripts

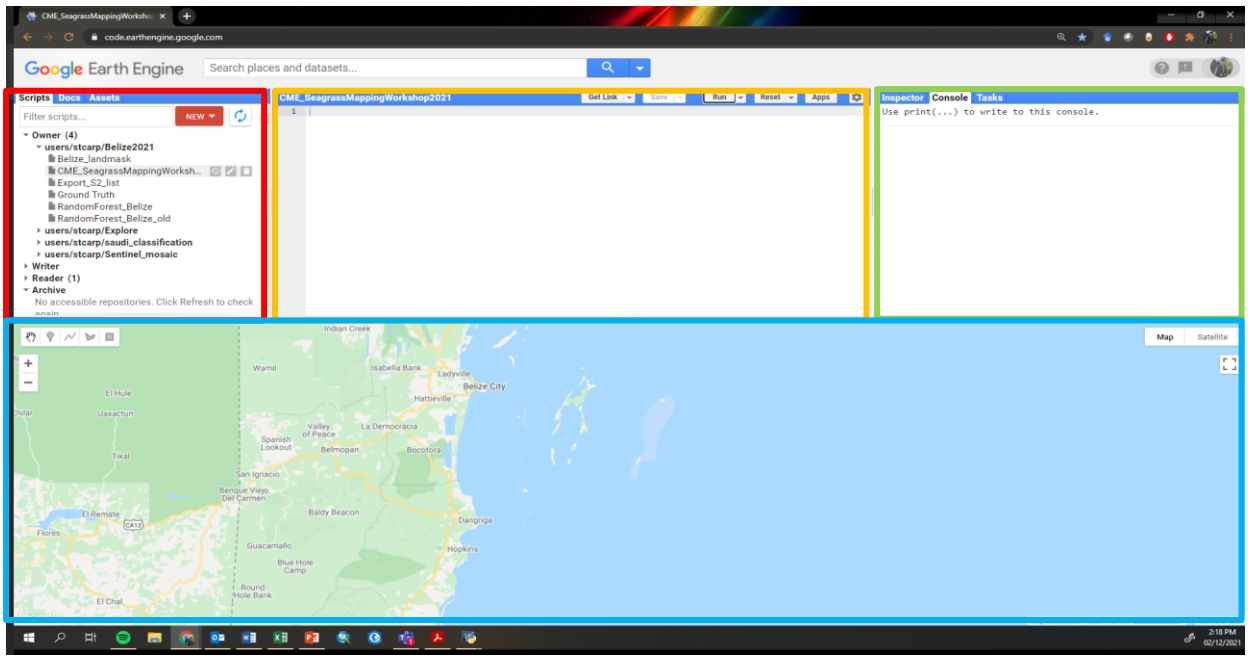
The google earth Engine code editor has the following layout. There are four main windows which divide the internet browser:

1. Scripts/Docs/Assets – This window emulates file-explorer-style interface.
2. Script – Where you create, run and share scripts to complete actions
3. Inspector/Console/Tasks – This is where you can explore data, print outputs and complete tasks (e.g. export data)
4. Map window – Visualise data

Acts as online file

Script (where to run tools)

Inspect data, outputs



Map view: Visualise

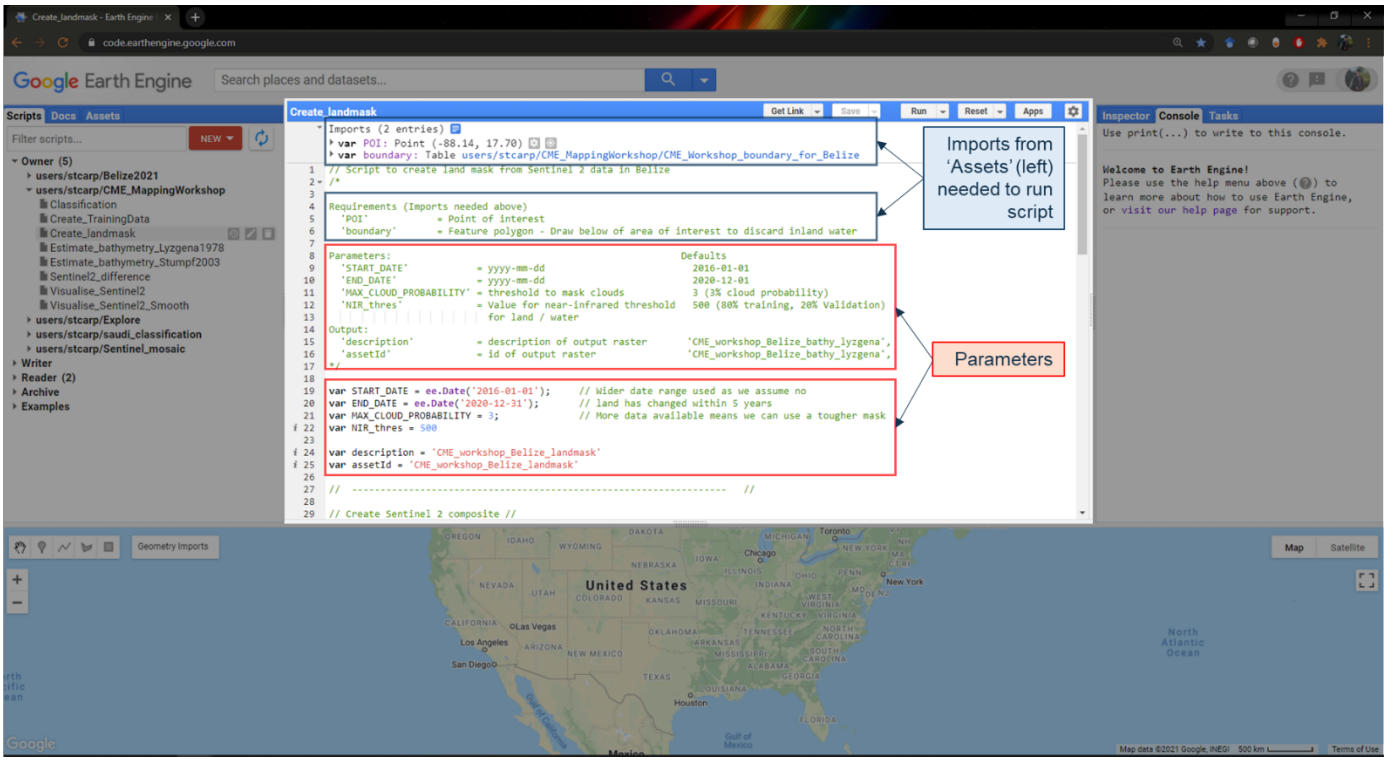
To simplify the process of editing and running scripts, the scripts are split into sections. The first of which is the main editor for imports, requirements and variables which sits above the main script (as below) and is where the majority of the user-customisation is.

As long as there are variables and imports correctly formatted above the line which looks like this:

// ---------- //

is script will run.

When estimating bathymetry. The script requires an import of bathymetry points to create the bathymetry layer.



PRACTICAL INSTRUCTIONS

Exploring Sentinel 2 imagery

The following scripts are given to help explain why masking clouds is essential when creating a cloud-free composite for a given area. The retrieval and processing of the images is repeated in the other scripts given. **Please skip this section if you understand why a cloud mask is needed and know how to run a script in Earth Engine.**

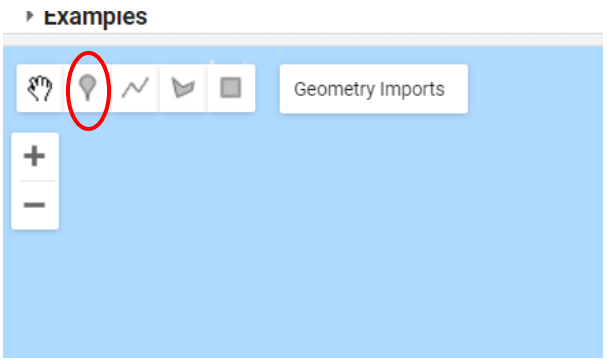
1. **Script: Visualise Sentinel2**

This script is used to create median composite of Sentinel 2 data from images less than a threshold.

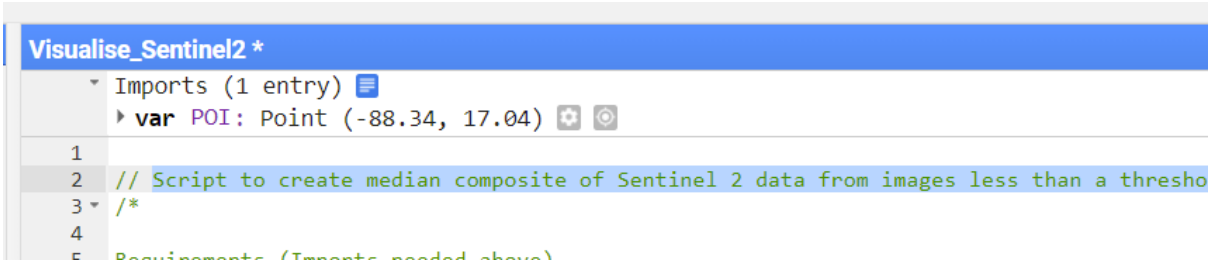
a. Follow the link for the workshop and sign in, if necessary.

b. Open the new script by clicking on 'Visualise_Sentinel2' in the 'scripts' window on the left.

c. Using the tools in the map view, click on the marker and click a point on the map where you would like to visualise satellite data.

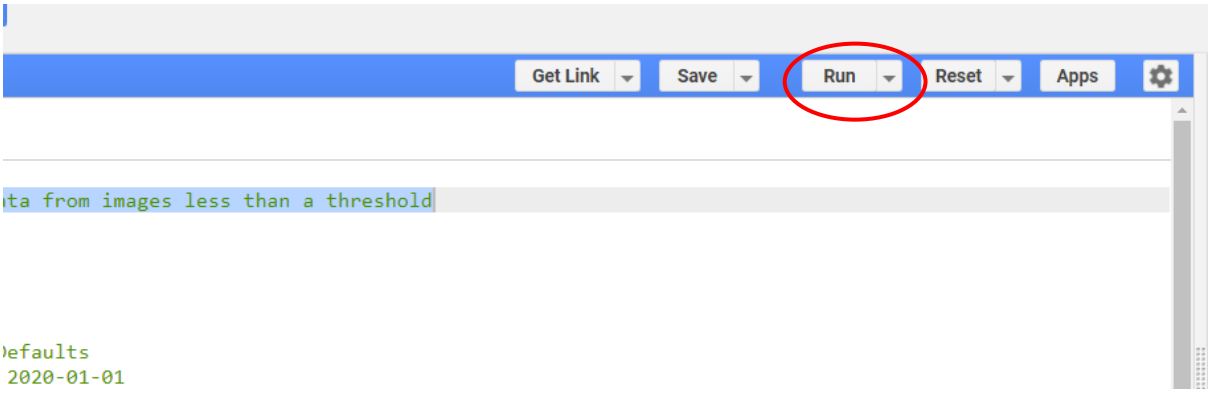


d. Within the scripts window, you should now see 'geometry: Point (xx.xx, xx.xx)'. Rename 'geometry' to POI, as below



The script collects data based on the START_DATE and END_DATE and defaults to make an annual composite from 2020. Change if necessary. A cloud threshold is also used to filter out very cloud images in the composite.

e. Now run the tool by clicking 'run' in the script window.



2. Script: Sentinel 2 difference

This script is similar to the previous, a 'POI' point is required to run the script. This script outputs two images in a split panel. The left shows the image create in the previous script, the right a median composite using a cloud mask every pixel that is likely to be cloud. Here, we use a cloud probability threshold of 5%. Depending on where you are in the world, the number of satellite images available and the cloud coverage changes. In areas where there are less images available for the median, a tougher cloud mask may produce areas with no data available. The high reflectance of near-infrared in urban areas can also result in a high cloud probability in the layer and give no data in the composite. If you are mapping land areas then you may need to increase the threshold.

a) Open the new script by clicking on 'Sentinel 2 difference' in the 'scripts' window on the left.

a) Using the marker in the map window, place a marker in your area of interest. Again, at the top the script, there should now be a POI. If there are more than one point then delete the import (If you put the cursor over the import, there should be a bin marker – click this to remove an import). Re-draw the point and rename 'POI'. Change the dates and max cloud probability threshold parameters, if necessary.

b) Now run the tool by clicking 'run' in the script window. The right image may take a minute or so to load (but this is because we are masking 134 individual satellite images [in the example] and merging them as one!)

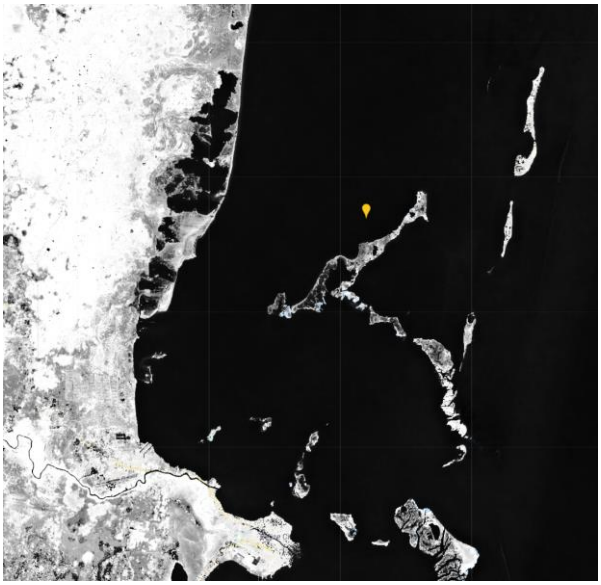
c) You'll notice in the console window on the right, the number of images used in the median has increased. By using a cloud mask, we can use all of the imagery available in the image collection, rather than having to filter out images with greater than 40% cloud.

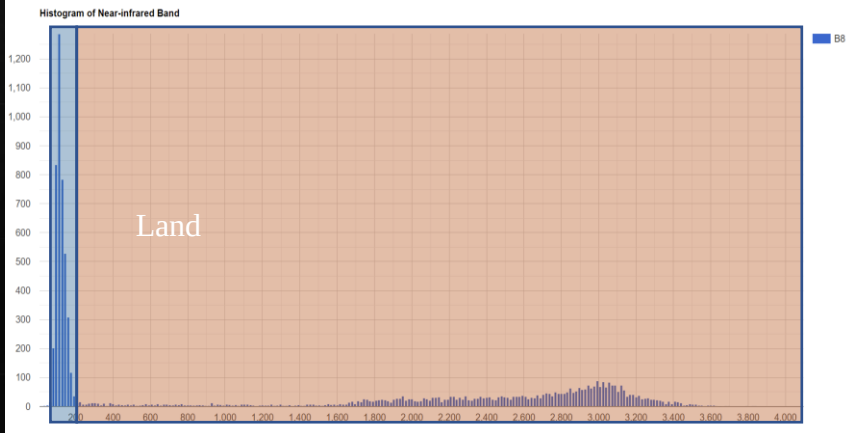
d) By using the slider, you should see that the image on the right is clearer than the left. There should be more contrast between the brightness of features on the ground and little to no clouds in the image. We will use this image to map our benthic habitats for the next stage of processing.

Creating a Land Mask

1. **Script: Create_landmask**

The focus of this classification is on shallow water environments; therefore, we need to distinguish between land and water. To do this, we use the near-infrared (NIR) band of the satellite, which almost entirely gets absorbed by the water. The image below shows a NIR image of Belize and the corresponding values in a histogram. There is a clear difference between the low absorbing pixels on the left of the histogram which represent the water pixels and those pixels with varying brightness on land. A threshold between the two can be chosen to define water and land.





Water Pixels

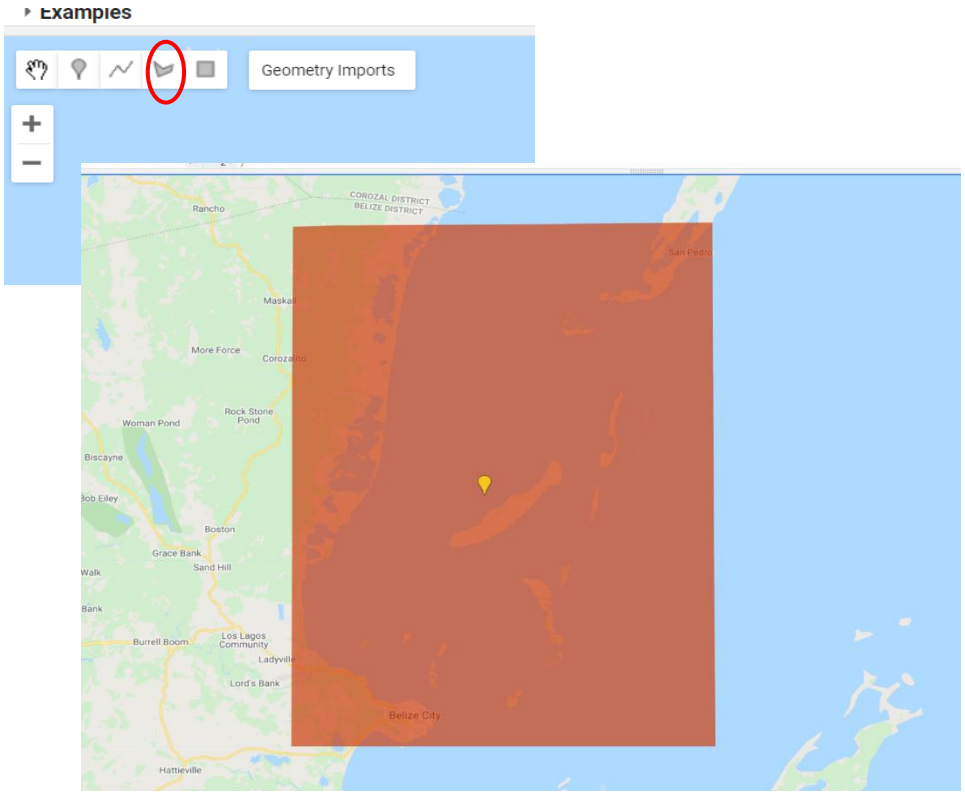
Near Infrared Band – Belize City

a) Open the new script by clicking on 'Create_landmask' in the 'scripts' window on the left.

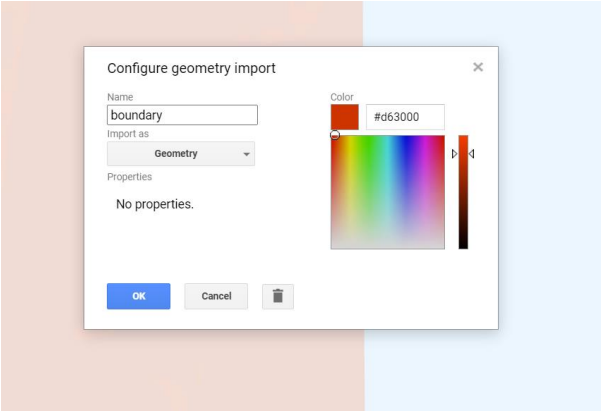
b) Using the marker in the map window, place a marker in your area of interest. Again, at the top the script, there should now be a POI. Change the dates and max cloud probability threshold parameters, if necessary.

b) In the script window, you'll notice that another import is required to run the script. This is a boundary of the area we are interested in to create this boundary, we use the polygon drawer in the map view (see below)

c) Click on the polygon drawer and draw a box around the area of interest



d) Now in the map view, you should see 'geometry imports' next to the polygon drawer tool on the left. Move the cursor over it and see 'POI' and 'geometry'. Move cursor over 'geometry' and a settings wheel should appear – click on this and change the Name from 'geometry' to 'boundary' As below:



e) Leave the 'import as' = geometry and press OK. POI and boundary should now be name in the imports at the top of the script window.

f) Three more parameters are given in this script, the final two are output variables we use to name the land mask for when it is exported to an asset (in the window on the left), rename output names to something more appropriate to your data or region:

i) NIR_thres = the near-infrared threshold used to define land and water

ii) Description = A short text describing the data

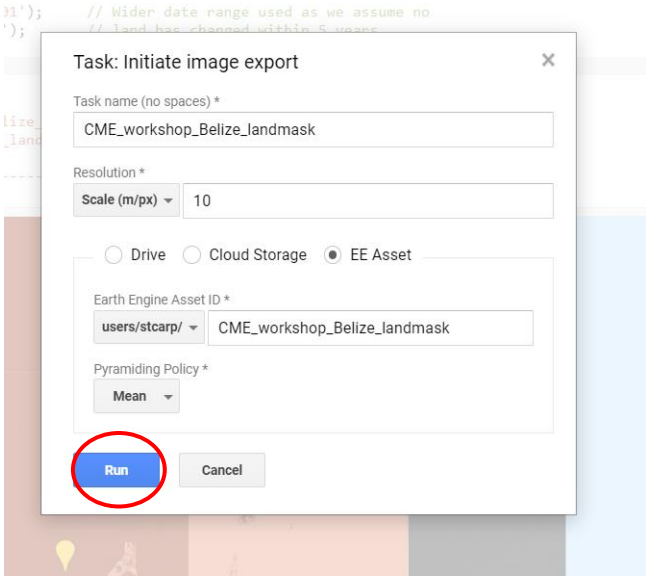
iii) assetId = The name of the output data

g) The date range has also been changed in this script to incorporate data from a wider range (2016-2020) as we try to create the smoothest median composites possible and we expect there to be little change in land area within this 5-year period.

h) Now run the tool by clicking 'run' in the script window.

i) You'll notice the 'Tasks' window on the right is now orange, meaning that a task is pending. This is the export of the land mask to an asset. Click on the tasks window and you shall notice 'CME_workshop_Belize_landmask' has appeared.

j) We need to run the export to begin the tasks. This brings up another window with some further parameters, but keep these the same and press 'run' again. The land mask will now be exporting to the asset window. You may need to create a home folder if you haven't already.



Estimate Bathymetry

A key variable in habitat mapping is bathymetry which can be estimate correlating in-situ data with log transformed satellite bands. Here, two of the most popular methods to derived bathymetry are given:

Stumpf et al. 2003

Uses empirical relationship between the ratio of the log-transformed green band to the log-transformed blue band and water depth.

Lyzenga et al., 1985

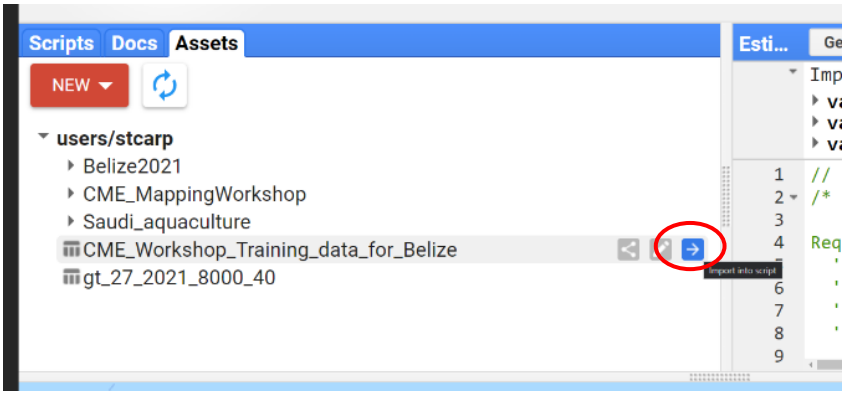
Assumes a linear relationship between the log-transformed bands and known depth via (multiple) linear regression

The Lyzenga model has been shown to be quantitatively the most accurate in both tropical and temperate waters (Traganos et al., 2018) and has a better prediction capacity beyond the extent of the training data, both horizontally and vertically so we will proceed with this model. Both scripts have the same requirements and parameters.

c) Open the new script by clicking on 'Estimate_bathymetry_Lyzgena1985' in the 'scripts' window on the left.

d) Using the marker in the map window, place a marker in your area of interest. Again, at the top the script, there should now be a POI. Change the dates and max cloud probability threshold parameters, if necessary.

e) In the script window, you'll notice that a further two imports are required to run the script. This is the land mask we just created in the previous script. To import this, click on the 'Assets' tab on the left. You should see your land mask data, in this case our data is called 'CME_workshop_Belize_landmask'. Place your cursor over the data and press the right arrow to import the data into the current script.



f) The second import is bathymetry point data, here named 'bathy_points'. When applying this method to a new area, you will need to import your own data – See *'Importing your data'* (below)

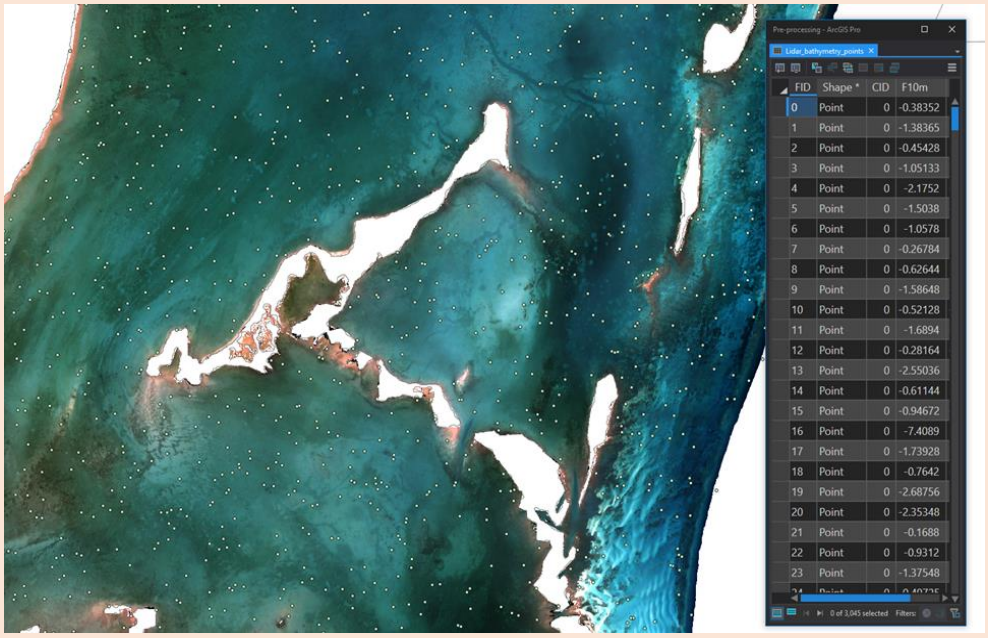
Importing your data

Within Earth Engine you can upload tables, shapefiles and rasters (via geoTIFFs or TFRecords) to analyse data not in the Earth Engine catalogue. For a detailed explanation on each upload method see:

- Table/shapefiles - https://developers.google.com/earth-engine/guides/importing
- Raster - https://developers.google.com/earth-engine/guides/image_upload

For more information on managing assets, see https://developers.google.com/earth-engine/guides/asset_manager.

In the context of this work, we want to upload a shapefile of bathymetry points, which includes an attribute table of depths created from LiDAR data aggregated to 10m.



For shapefiles - In the 'Assets' menu, click 'new' > shapefile. Click 'select', navigate to your shapefile and click upload

For tables - In the 'Assets' menu, click 'new' > csv. Click 'select', navigate to your table and click upload

You'll notice the 'Tasks' window on the right is now orange, meaning that a task is pending. This is the import of the data to an asset. Click on the tasks window and you shall notice your filename has appeared. Run the task

Once completed, the data should appear in your assets window.

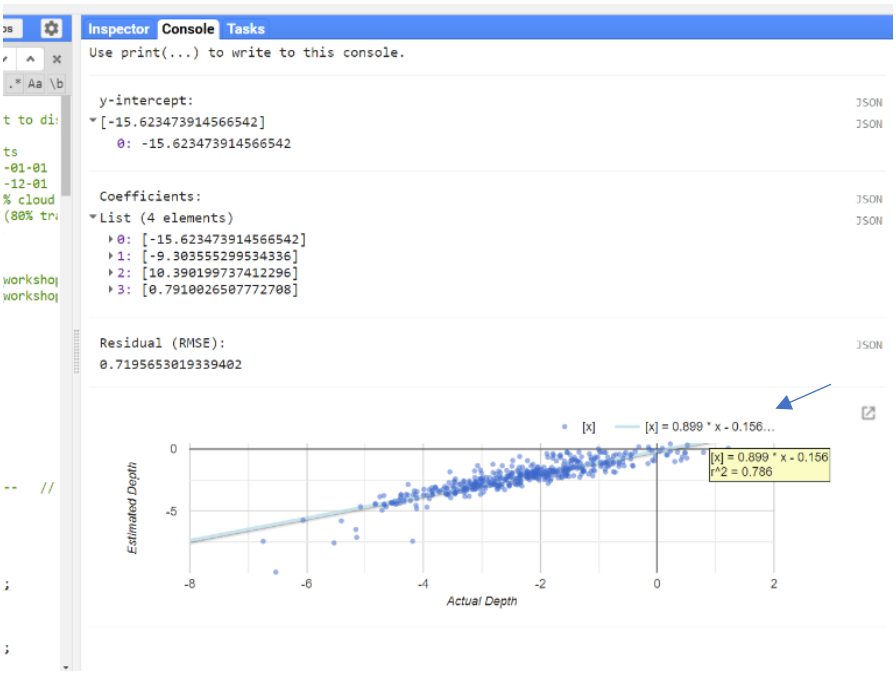
g) Once your data is uploaded, add it to the script using the right arrow on the asset.

h) Similar to the land mask, we expect there to not be much change in the bathymetry over 5 years, to we use a 5-year median composite of data from 2016-2020. A longer time period means we can utilise a lower cloud probability value for a really tough mask to capture even more erroneous data.

i) Two other parameters are included; 'split' which is a value between 0-1 that divides the bathymetry data between training and validation. Here we use an 80/20 split. The variable 'depthName' is the name of the depth field in the bathymetry table uploaded. In this case the field is F10m.

j) Finally, rename the output names and description if necessary and press Run.

k) Both the 'Console' and 'Tasks' on the right should now be orange. First, let's look at the Console. A few values are displayed, which give the y-intercept and coefficients for the blue, green and red bands. Used in a multiple linear regression to find depth (see Lyzgena, 1985 for more details). The root mean square error is also given. An r-squared value can be seen by moving the cursor over the equation of the trendline, in this case we see a value of 0.786. A value closer to 1 means the estimated and actual bathymetry is highly correlated, so 0.786 is a respectable value for the layer to use as a proxy for bathymetry. This type of bathymetry can be highly skewed by water quality and turbidity; therefore, interpretation of the raw depths should be done cautiously.



l) Now select the 'Tasks' menu and click Run to export the data to your assets. This script includes lots of processing which is why the bathymetry is not added to the map view at the end. To see the product, export the data to assets and import into the script. Once you have done this, move the cursor over the image import and at the end of the line you'll see an eye symbol. Click this to see the bathymetry in map view.

Build Training

In the script, we build training and validation data using expert labelling. Usually this is completed by visualising some high-resolution data and drawing areas of difference classes to train a model which uses these zones to group together similar pixels and classify them into the classes. There are a variety of methods which can be used to model and group the data; Random Forests machine learning algorithm (Breiman, 2001), classification and regression tree (CART; Breiman et al., 2017) and support vector machine (SVM; Zhang et al., 2001) are available within Earth Engine. Here, we use Random Forest as it has shown to be the most accurate in coastal regions according to Poursanidis et al., 2020.

Once drawn, each polygon is randomly sampled to create points for training and validation used for the next script.

a) Open the new script by clicking on 'Create_TrainingData' in the 'scripts' window on the left.

b) Using the marker in the map window, place a marker in your area of interest. Again, at the top the script, there should now be a POI.

c) Import your bathymetry layer from the assets window by clicking on the right arrow when the cursor is on the layer. Change the dates and max cloud probability threshold parameters, if necessary.

d) In the workshop, a high-resolution aerial photography image was used to assist the drawing of the classes for training, you may want to upload other high-resolution datasets such as drone data. If so, see the following section, otherwise continue to e).

    i) Once you have uploaded the data into Earth Engine, import the layer from the assets into the script and rename the layer accordingly.

    ii) Scroll down the script to the section below outputs, named 'Importing High-resolution data?'. If necessary, delete the two forward slashes before the line 'Map.addLayer'. This will change the words from green (comments) to a variety of colours.

    iii) Run the script to visualise the high-resolution layer.

    If you already have ground truth polygons, these can also be uploading individually and then import into the script.

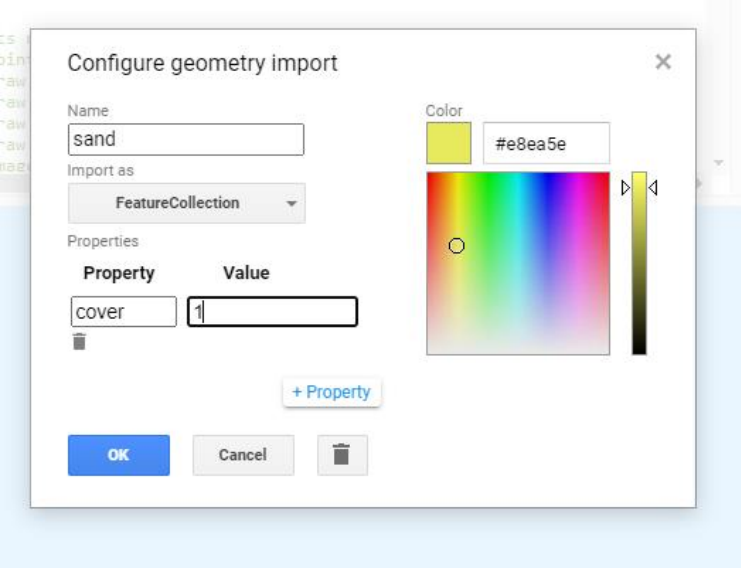
e) Run the script to visualise your satellite/high resolution imagery. The other imports; *seagrass, coral, sand and deep-water* are given, but remove these if looking at a new area by moving the cursor over the import and clicking the bin in the scripts window. To create a new polygon class, put your cursor over 'geometry imports' next to the polygon drawer tool on the left. Move the cursor over it and click on '+ new layer' to create a new layer. Move cursor over 'geometry' and a settings wheel should appear – click on this.

i) Change the Name from 'geometry' to your chosen class e.g. 'sand'.

ii) Under 'import as' ensure the option list equals *FeatureCollection*

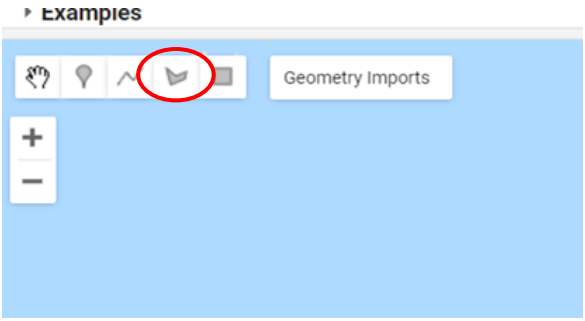
iii) Add a property, name it 'cover' and give it a unique value, e.g. 1. Make a note of this number. Each class needs to have the property 'cover' and a **unique value**. When we repeated this for another class, you will have to add the property 'cover' again, and give it another value, e.g. 2.

iv) Change the colour of the class to something you will remember by clicking in the colour window and changing the brightness. Here, use the code '#e8ea5e' for sand.

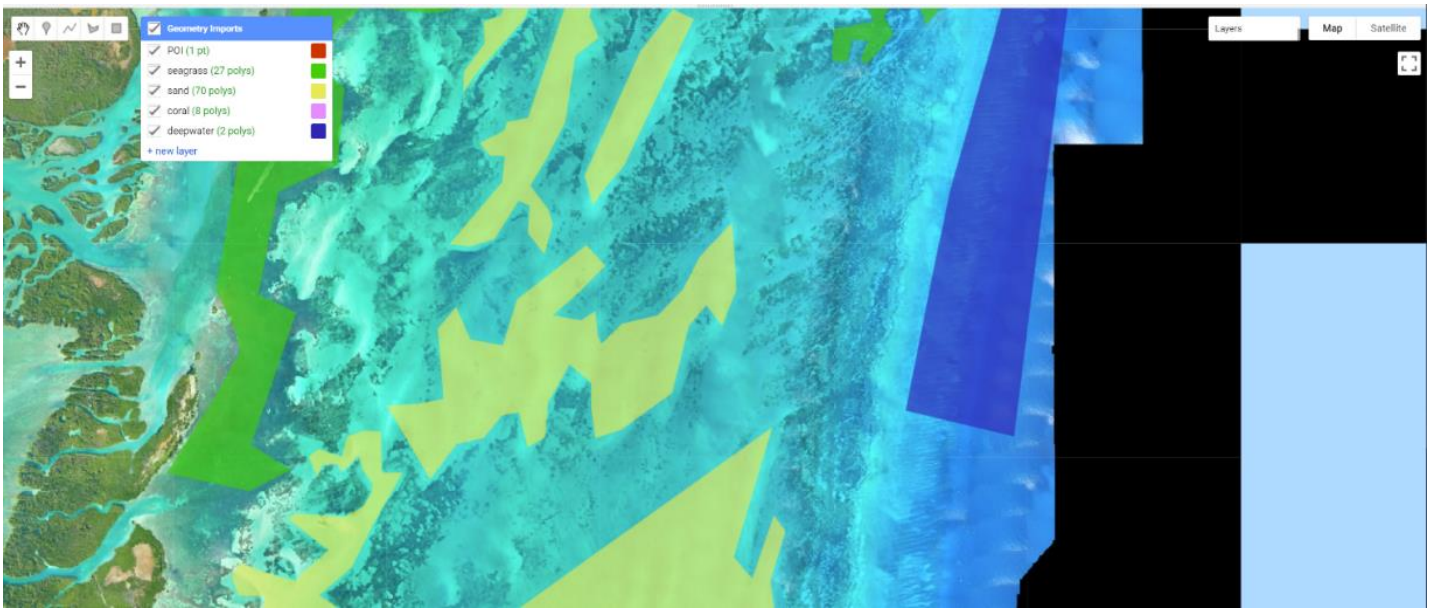


v) Click Ok

f) The cover class will now appear in the imports at the top of the script. Now use the polygon drawer tool to draw over zones of interest



g) Repeat steps e) to f) and create classes for each bottom type. Here, a four-class training dataset is used; seagrass, sand, coral and deep-water. Below exemplifies an area of Belize with polygons created in each bottom type.



h) Once all of the polygons have been created, we can set the number of points for each class. This number may change based on how much spectral variation these is within a single class, or the size of the class itself. Here, there are large areas of sand and seagrass, therefore they have more training points.

i) Rename your output files to something more appropriate.

j) Run the script again. Some results have been printed in the console:

   i) A list of cover class numbers, these are the numbers you entered when numbering each cover class. Check this match the numbers you noted down in step e) iii).

   ii) Number of training points (handy for reporting)

   iii) Number of validation points (handy for reporting)

k) The 'Tasks' window is also orange implying you have the training and validation datasets ready to be exported to your assets. Click run on both and wait for them to transfer to assets.

Classification

a) Open the new script by clicking on 'Classification' in the 'scripts' window on the left.

b) Using the marker in the map window, place a marker in your area of interest. Again, at the top the script, there should now be a POI.

c) Import the following layers from the asset window:

   i) Land mask

   ii) Training

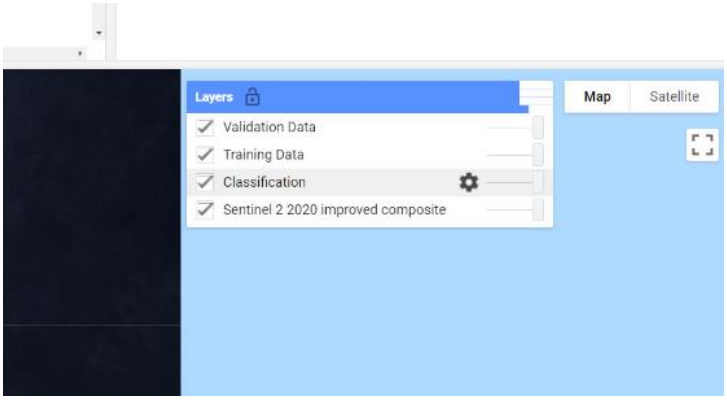
   iii) Validation

   iv) Bathymetry

d) Change the dates and max cloud probability threshold if necessary. An additional variable is given here, the number of trees. This represents the number of decision trees in the random forest classifier. More trees will generally result in more accurate model prediction, but the model will take longer to calculate.

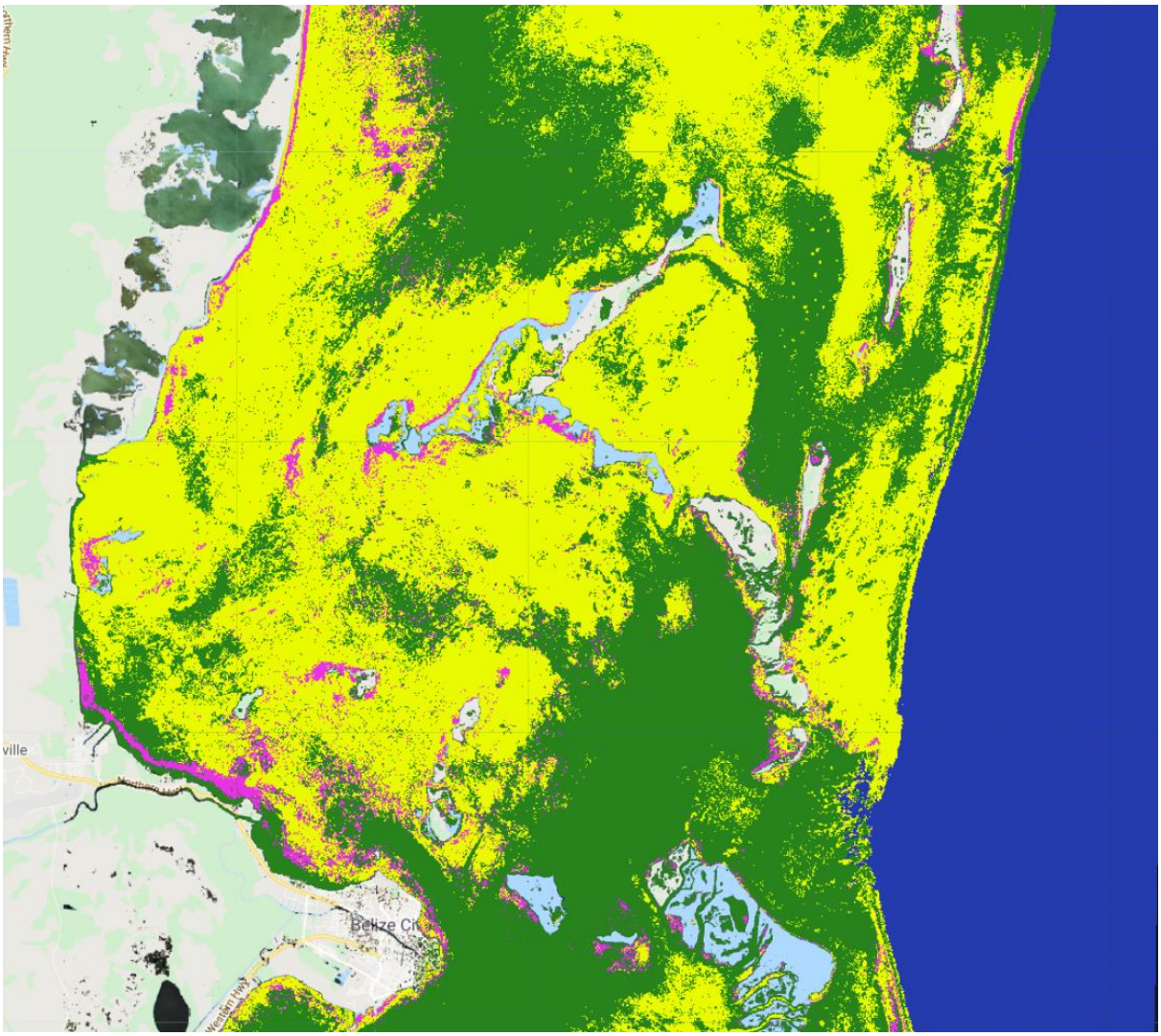
e) Run the script. Four layers are displayed in the map view. The Sentinel 2 composite, classification, training and validation data.

f) Whilst in the map view, move the cursor over the layers tab on the right and see these layers (below). Each layer has a tick box to turn on the layer in the map view, a settings wheel (when cursor is over the layer) and a transparency bar. The classification may take a long a time to load so turn it off for now.



g) Zoom to an area of training/validation points (red/blue zones). Here you will see how the training data has sampled the cover classes from the previous script. If you are happy with the distribution of the points and number, continue, otherwise you can go back to the training script and and change the number of points used or add new polygons.

h) Now turn on the classification and use the transparency bar to compare between it and the original Sentinel 2 composite to see how the classification performs. In this example the end classification looks



like this:

As you can see, there is some confusion within the turbid waters above Belize City and mainland as a result of an over-simplified classification. Coral areas are challenging to map using Sentinel 2, where they are often mixed with seagrass and sand within the 10m pixel. The reflectance of these bottom types is combining to produce a similar signature to turbid waters. This may prompt the addition of a turbid water class to describe these areas.

Elsewhere, the seagrass and sand distribution appear reasonable.

i) In the Console on the right, you'll see some statistics of the classification– we will go into these in more detail in the next section:

i) An **overall accuracy** is given and a **kappa statistic** index of agreement. These accuracy rates range from 0 to 1, where 1 represents 100 percent accuracy. The kappa statistic is a percent agreement between the classification and what's on the ground and takes in the element of chance. A value closer to 1 shows a closer agreement and a result that is less down to chance.

ii) A **Confusion Matrix**, where axis 1 (the rows) of the matrix correspond to the actual values, and axis 0 (the columns) to the predicted values

iii) **Producer Accuracy**, a list of values which correlate with the cover class numbers.

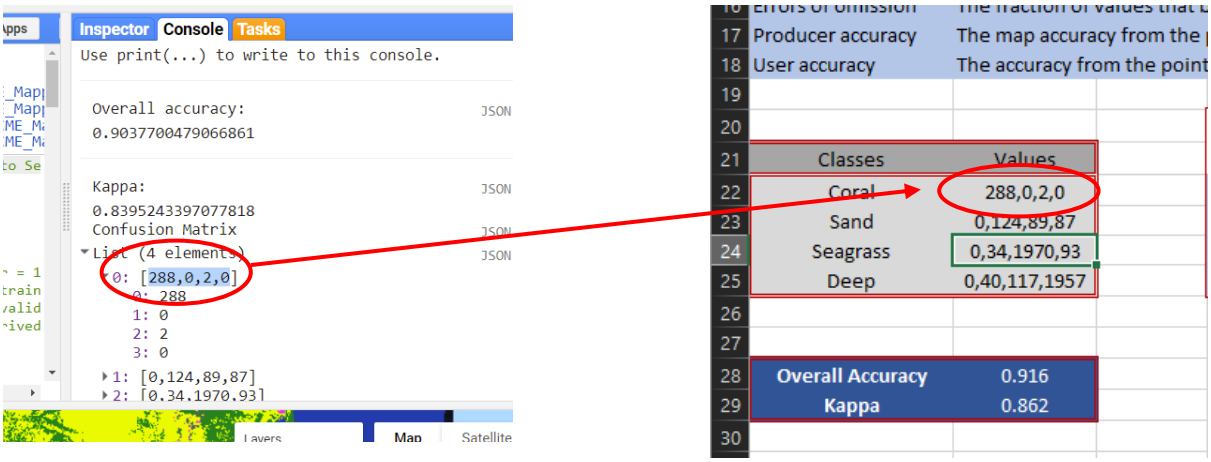
iv) **User Accuracy**, another list of values which correlate with the cover class numbers.

It is important to note that these statistics may be skewed by the validation used to assess the accuracy of the classification. The training and validation data are taken from the same data source (the cover class polygons) drawn from high confident areas. It is recommended that two different sets of polygons are used to reduce spatial autocorrelation[1].

[1] Spatial autocorrelation is the term used to describe the presence of systematic spatial variation in a variable and positive spatial autocorrelation, which is most often encountered in practical situations, is the tendency for areas or sites that are close together to have similar values (Haining, 2001).

Accuracy Assessment

As seen in the Console after running the classification, the format of the confusion matrix and statistics can be difficult to interpret. Here, this document should be provided with an excel spreadsheet 'Confusion_Matrix_EarthEngine'. Open this spreadsheet and follow the instructions at the top of the page. You can copy the comma delimited values from the Earth Engine Console and paste each line into the appropriate cell.



After following the instructions here, you will be able to interpret the behaviour of the classification more easily.

Extra Resources:

Geo for Good 2020 Summit.

- YouTube Playlist of all talks - https://www.youtube.com/playlist?list=PLLW-qoCMKQsze8jjRsfbXurFm3wUyOerb

Earth Engine Community Tutorials - https://developers.google.com/earth-engine/tutorials

- Forest cover change
- High quality cloud masking
- Vegetation condition modelling

Easy browser for S2 Surface Reflectance images - https://showcase.earthengine.app/view/s2-sr-browser-s2cloudless-nb

References

Breiman, L., 2001. Random forests. *Machine learning*, *45*(1), pp.5-32.

Breiman, L., Friedman, J., Stone, C.J. and Olshen, R.A., 1984. *Classification and regression trees*. CRC press.

Haining., 2001. Spatial Sampling. International Encyclopedia of the Social & Behavioral Sciences. (pp. 14822-14827)

Lyzenga, D. R. 1985. “Shallow-water Bathymetry Using Combined LiDAR and Passive Multispectral Scanner Data.” International Journal of Remote Sensing 6: 115–125. doi:10.1080/01431168508948428.

Poursanidis, D., Traganos, D., Teixeira, L., Shapiro, A. and Muaves, L., 2020. Cloud-native Seascape Mapping of Mozambique’s Quirimbas National Park with Sentinel-2. *Remote Sensing in Ecology and Conservation*.

Stumpf, R. P., K. Holderied, and M. Sinclair. 2003.“Determination of Water Depth with High-resolution Satellite Imagery over Variable Bottom Types.” Limnology and Oceanography 48: 547–556. doi: 10.4319/lo.2003.48.1_part_2.0547.

Tong, S. and Chang, E., 2001, October. Support vector machine active learning for image retrieval. In *Proceedings of the ninth ACM international conference on Multimedia* (pp. 107-118).