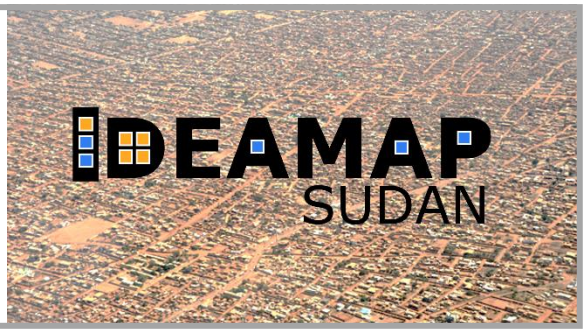
Image Classification

Practical Exercise



General Information

Goal:

- Apply a Random Forest classifier to a remote sensing image

Content:

- Generate / import training samples
- Apply a Random Forest for classifying land use in satellite images

In Part II we will assess the results and try to improve the model.

Software Requirements:

- QGIS
- The "dzetsaka" QGIS plugin (see instructions below)
- The data for this practical which can be downloaded from Canvas.

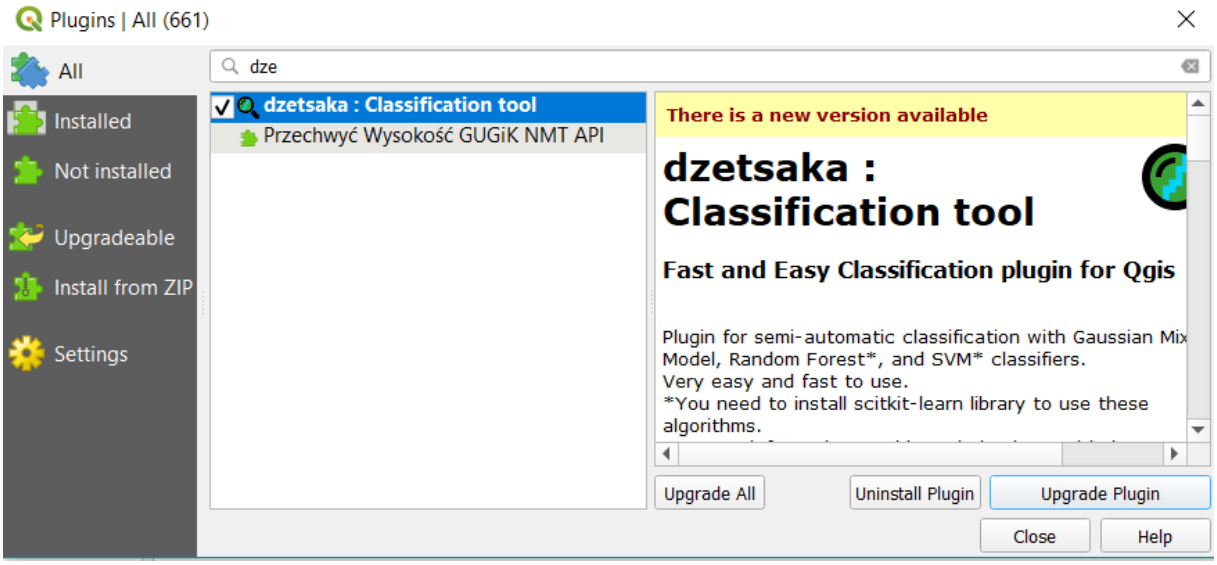
Data to be used:

- **ESACCI-LC10-Map-20m-2016-MLPractical.tif** – Land cover dataset raster
- **ESACII-LC-colormap.clr** – QGIS style for the Land Cover dataset
- **S2_20170101_Year_20m-MLPractical.tif** – Satellite imagery
- **MLPractical_Training_Points.geojson** – Training points
- **CCI_L10_Training_Points_Style** – QGIS style for the training points

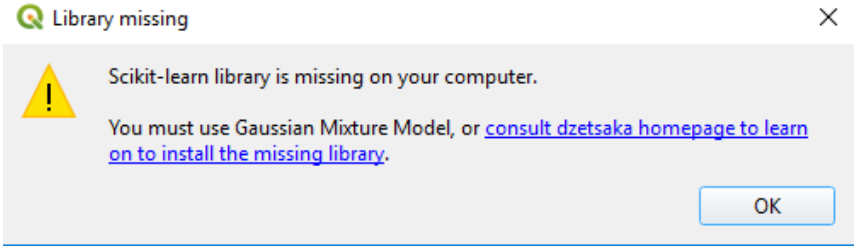
Software Installation

We will be using QGIS as you installed in Topic 2. You will also need to install the dzetsaka plug-in in QGIS.

In the top menu of QGIS, select "Plugins" → "Manage and Install Plugins". Then type **dzetsaka** in the search bar. Select it and then install (or upgrade) the plugin. After it is correctly installed you should see it under the list of installed plug-ins (left column in screenshot below).



To run this plugin, you need to install the Scikit-learn library. If this library is missing, the following message will appear:



If the library is missing, open the program OSGeo4W Shell on your computer (it is installed automatically when you install QGIS). In the interface, type the following command, enter to execute, then type the second.

```
py3_env.bat
python -m pip install scikit-learn --user
```

Alternatively, you can install miniconda: https://docs.conda.io/en/latest/miniconda.html. Once you have installed it, you need to open Anaconda Prompt and write the following command:

```
conda install scikit-learn
```

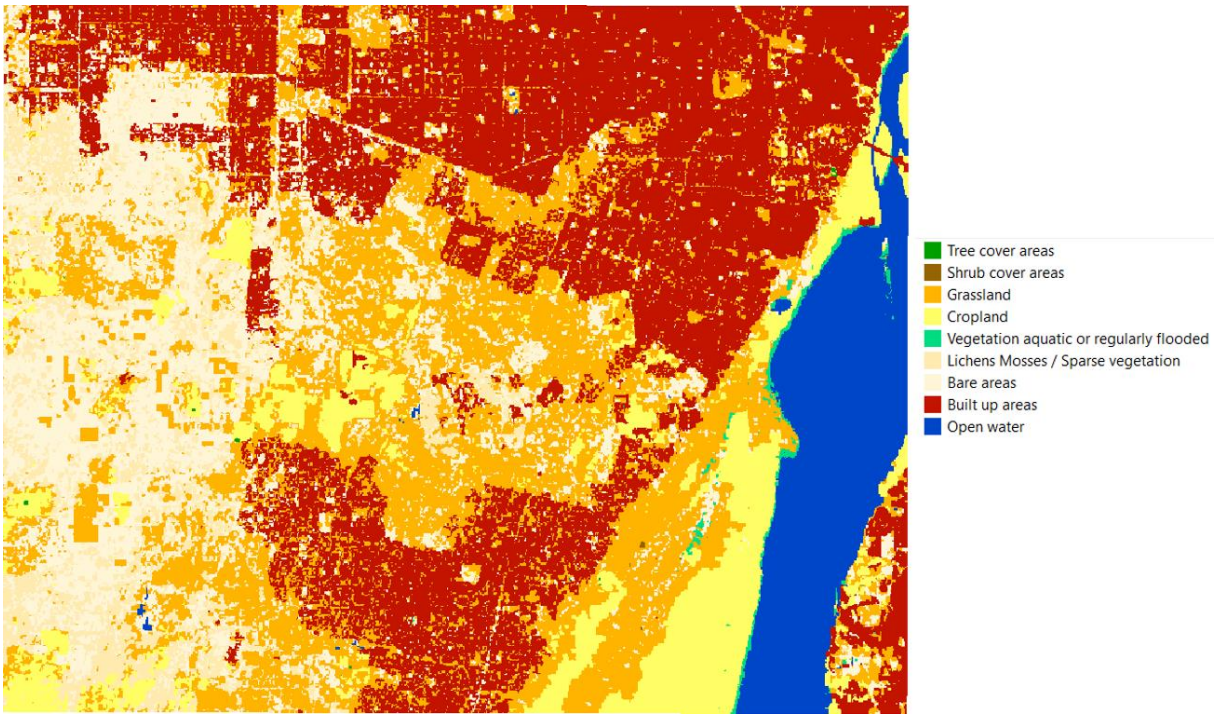
Once the plugin has been installed, you can add the classification dock to your QGIS project. More information about this plugin can be found by accessing the following links:

- https://github.com/lennepkade/dzetsaka
- https://www.karasiak.net/dzetsaka-how-to-make-your-first-classification-in-qgis/

Datasets

Reference data

We will be using Land Cover data from the **CCI Land Cover dataset**. It has a spatial resolution of 20m and was generated from one year of Sentinel 2A observations from December 2015 to December 2016. More information can be found here: http://2016africalandcover20m.esrin.esa.int/ . A part of Khartoum was subsetted for this practical exercise.



Input data

The classification will use a **Sentinel 2a image** mosaic of 2017. The input bands 2 – 8A are available:

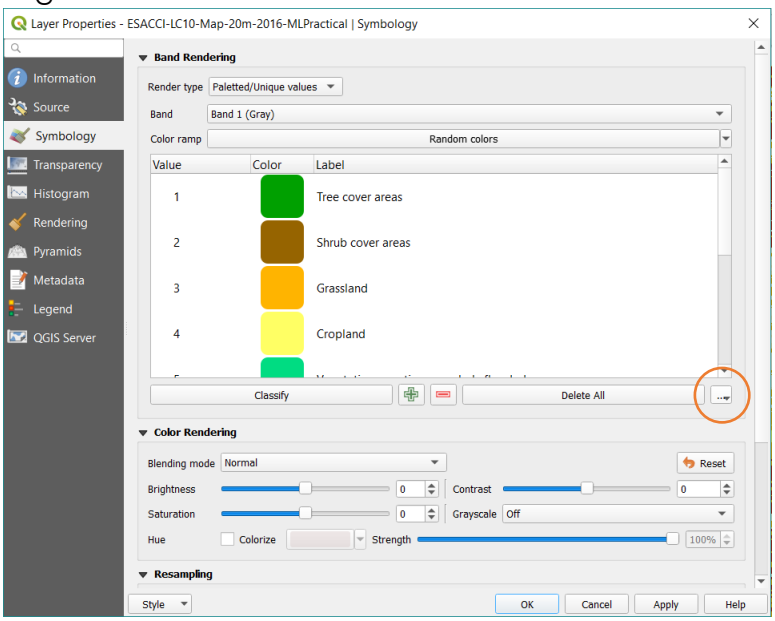
Sentinel-2 Band	Central Wavelength (µm)
Band 2 – Blue	0.490
Band 3 – Green	0.560
Band 4 – Red	0.665
Band 5 – Vegetation Red Edge	0.705
Band 6 – Vegetation Red Edge	0.740
Band 7 – Vegetation Red Edge	0.783
Band 8 – NIR	0.842
Band 8A – Vegetation Red Edge	0.865

Training samples:

To train the classifier, 150 points were randomly selected and labeled according to the CCI land cover layer.

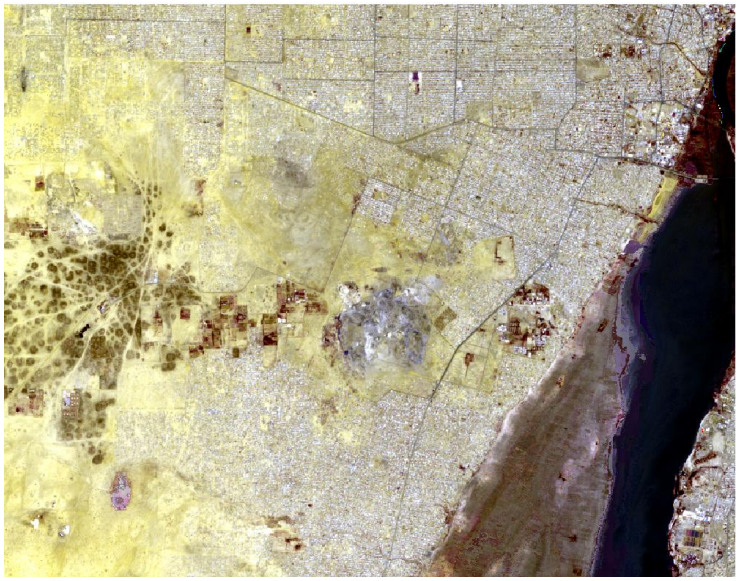
Instructions

1. Add the land cover map (**ESACCI_LC10-Map-20m-2016-MLPractical.tif**) to a new QGIS project. Then change the symbology of the layer to "Paletted/Unique values" and import the color scheme (**ESACII-LC-colormap.clr**) to visualize the land cover map by clicking on the "…" icon circled in orange below.



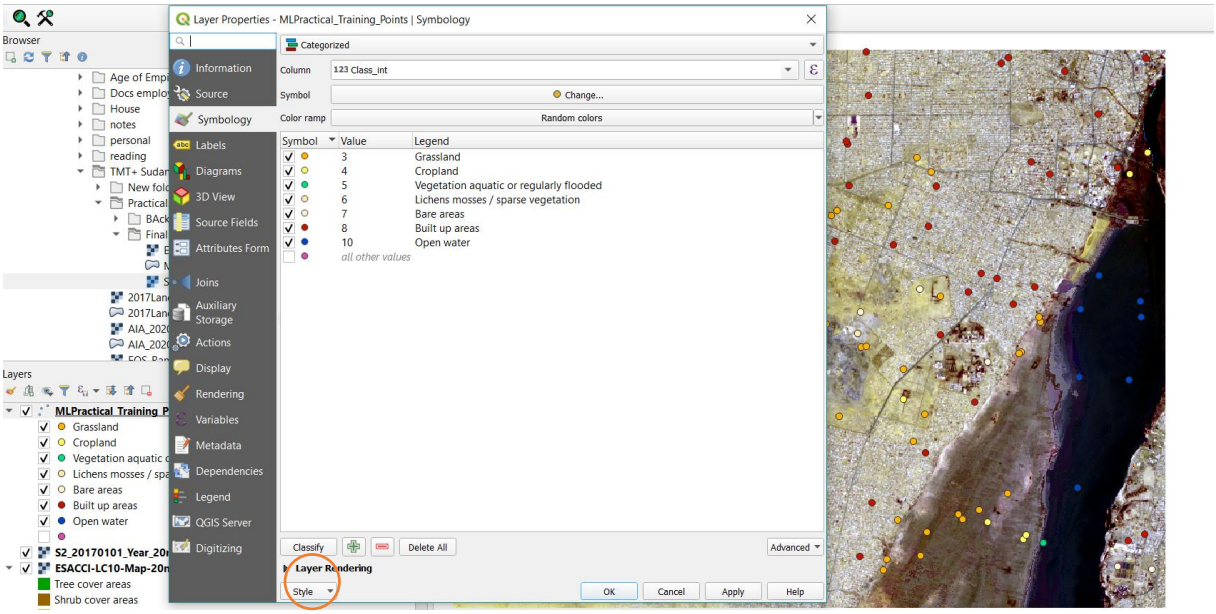
2. Now add the satellite image (**S2_20170101_Year_20m-MLPractical.tif**).

> ***ADVANCED:*** *You can change the visualization to B4, B3, B2 if you want a true-color image. Based on your knowledge of image spectra and reflectance, you can play around with other combinations to visualize the different types of land cover.*



3. Add the training samples to the map (**MLPractical_Training_Points.geojson**). Go to "Layer properties → Symbology" and make sure you select "categorized". At the very bottom, indicated by the orange circle below, you can select "Style → Load Style" and select the style file (**CCI_L10_Training_Points_Style**). Make sure the column "Class_int" is selected. You should then see the training points colored and labeled in the same way as the CCI landcover raster.

> ***ADVANCED:*** *You can also generate your own training data. In the top menu, select "Vector → Research tools → Random points in Extent"* to generate randomly distributed points. Then attach the value of the CCI input raster to each point by installing the QGIS plugin "Point Sampling Tool" and then going to "Plugins → Analyses → Point Sampling Tool". You may then need to convert the field to integer using the "Field Calculator" in the attribute table for the next step.



4. Perform the Random Forest classification using the dzetsaka plug-in. Open the classification tool window. Select the Sentinel 2 image mosaic as input and the training points with the label field "Class_int". Run the classification algorithm and look at your results.

> ***ADVANCED: (1)*** *Go back and add more training points (automatically and/or manually). Run the algorithm again. Do the results improve?* ***(2)*** *Change the settings to run a different classification model. Which method gets the best results?*

Analysis questions

- Do you think this is a good distribution of training points? *Hint: consider both the spatial distribution and the class distribution*.
- Compare your classification map to the CCI Land cover map. Where do you see similarities and where do you see differences?
- What are possible causes of these differences? Which are due to the input data and which are due to the algorithm?
- What is different between this machine learning workflow and the "reclassification" introduced in Lesson 8?