

Sampling in QGIS

This assignment guides you on how to create training areas in the form of polygons, which we save in a shapefile in QGIS for further analysis in R.

Collecting training areas is essential when working with supervised classifiers and significantly influences the classification outputs. You should make some preliminary considerations and approach the sampling very carefully! In the following, we will focus on the most important basics to consider.

Preliminary thoughts about sampling

Probably the most frequently asked questions are how many polygons should be created by class and how big should they be?

– Good questions!

Unfortunately, those just cannot be answered directly. The amount of training data you need, i.e., polygon count and size, depends both on the

- complexity of your classification problem (number and similarity of target classes, ...) &
- complexity of your classification algorithm (number of parameters or weights, RF, SVM, ANN, ML, ...).

Sampling data in machine learning is a science in itself, which is why there is a wealth of scientific publications about it (CURRAN & WILLIAMSON 1986, FIGUEROA et al. 2012) and even entire books (MARCHETTI et al. 2006, HASTIE et al. 2017).

Fine, so far that is not much of a help...

To keep it very simple: You need a sample of your data that representatively describes the problem you want to solve. Keep in mind, a classifier learns a mathematical function, which maps input data (e.g., spectral bands) to output data (e.g., class labels). In order to achieve this, you should provide enough training data to capture the relationships between input and output. **Training data** will optimally meet the following requirements:

- **independent of test data:**
A training dataset must be independent of the test dataset used for a validation, but can follow the same probability distribution. No training sample may be used to test (validate) the performance of the classifier! In the context of remote sensing data, it is also important that train and test data are spatially maximally distant to avoid spatial autocorrelation (Morans I).
- **mostly identical distributed:**
Each target class should be equally represented in the training data set. Most datasets do not have an exactly equal number of instances in each class. Small differences often do not matter. However, if there is a strong imbalance, e.g., 90% of all training data represent class 1 and only 10% class 2, most algorithms very quickly over classify the more-prevalent classes. Some simple options here: Collect more samples of the low-represented classes, use data augmentation to synthetically create new samples for under-represented classes, or use an

under sampling method. The simplest under sampling method is to delete samples from the over-represented classes during classifier training. We will use this latter method for the RF and SVM implementations later on.

- **representative for target classes:**

Training data should cover as many intra-class variations as possible, e.g., all spectral classes of a thematic target class, such as deciduous trees and conifers for the target class “forest”. Especially with more complex, non-linear classifiers, such as RF and SVM, it is important to include near-border training samples to map the class transitions more accurately. For example, water bodies should also be sampled in the shore area rather than just creating polygons in deep water areas.

- **available in sufficient quantity:**

There are statistical heuristic methods available to calculate a suitable sample size. Often a factor of the number of classes, the number of input features or the model parameters are used (e.g., 5 features – 25 training samples per class, THEODORIDIS et al. 2008) or the minimum number of samples necessary to perform the power calculation is searched (DELL et al. 2002). However, these rules are not universally applicable! Anyway, if you have many features, e.g., hundreds of spectral channels in hyperspectral images, it is important to collect even more samples to avoid the curse of dimensionality, i.e., Hughes phenomenon (HUGHES 1968). This curse occurs when the samples cannot reflect the possible parameter combinations in such a high dimensional feature space. As a result, the classification accuracy decreases as more features are included in the algorithm.

The best way to find out if the training samples are sufficiently set is to plot a learning curve. A learning curve plots the model performance on the y-axis versus the size of the training dataset on the x-axis as a line. On this way, you may be able to evaluate the amount of data that is required for a solid model performance, or perhaps how little data you actually need before the learning curve stagnates or even drops again. This plot can be generated during training, as shown in the next sections.

Before you start sampling the training data in QGIS, here are some general tips for digitizing your polygons, if you want to perform a monotemporal classification based on spectral features:

- evenly distribute the polygons for each class over the entire scene to best cover any atmospheric variations that may exist within the image,
- for each class, try to digitize an area of approximately the same size (sum of all polygons),
- keep in mind: each raster pixel under your polygons is a training sample,
- avoid huge polygons(!), e.g., creating a huge polygon over a homogeneous lake does not add much value in terms of characterization of the spectral properties of a lake. – create several small polygons covering different lakes instead,
- take your time! Sampling is an essential processing step and will largely determine your further analysis,

Enough theory, time to collect training data.

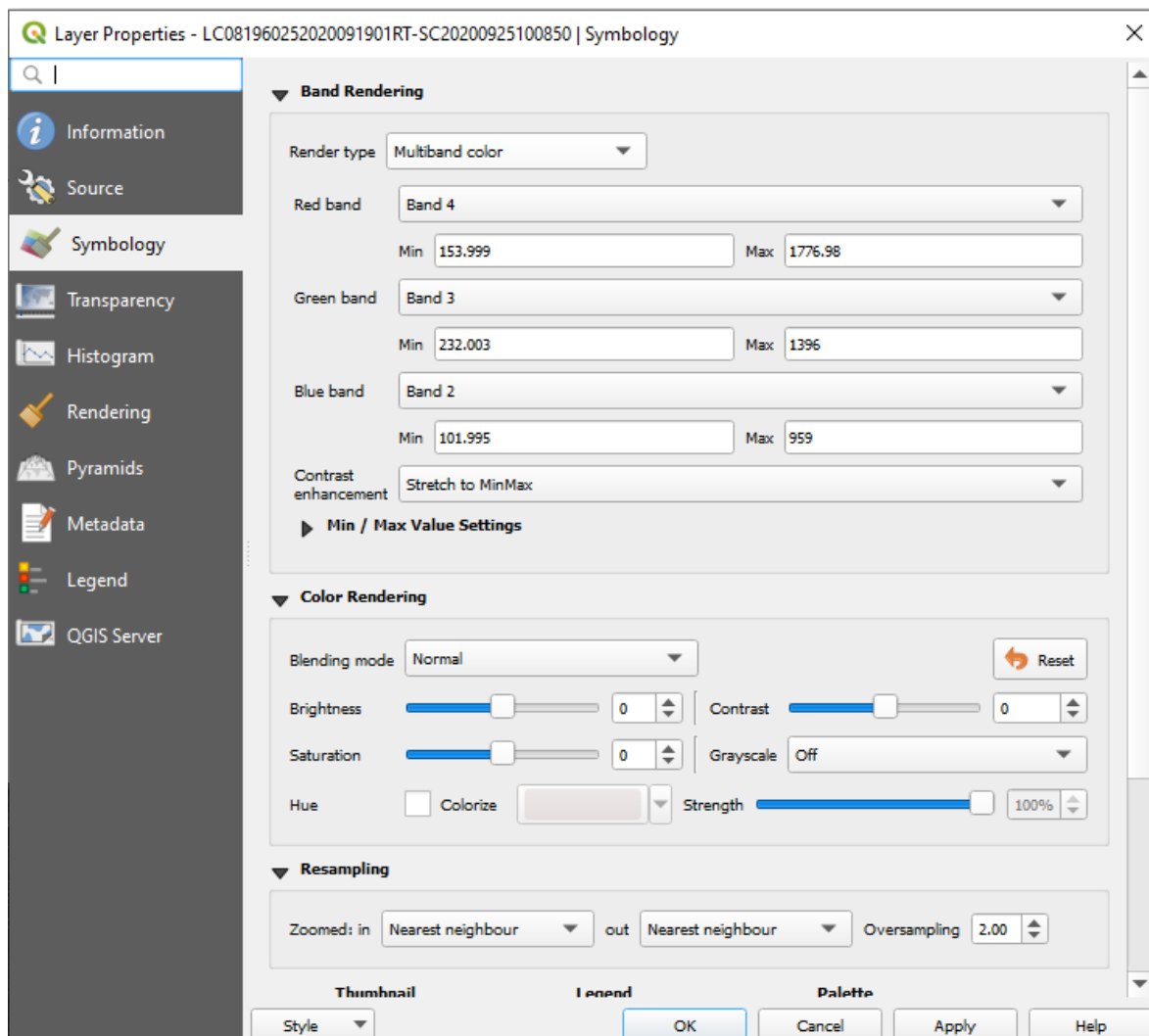
Import a Raster Dataset

The training polygons should define relevant areas for the differentiation of the desired target classes (*field, water, grassland, forest, urban* in our example). To know where these surfaces are located, we need corresponding image data as a basis. So let us import an image dataset!

First of all, open QGIS.

We are going to use one of our raster stacks from the previous preprocessing assignment. Import your file via Layer -> Add Layer -> Add Raster Layer... In this example, we use "LC081960252020091901RT-SC20200925100850.tif"

First, we are changing the appearance of our image to natural colors (RGB). Right click on the file, go to properties and Symbology. As render type, choose Multiband Color and set the following options: red (band 4), green (band 3) and blue (band 2).



In the next step we would like to clip our data to the area of Bonn, as the current image is way too huge. Accordingly, you need to find Bonn first. Doing so, we can use the built-in OpenStreetMap Plugin via XYZ Tiles in the browser.



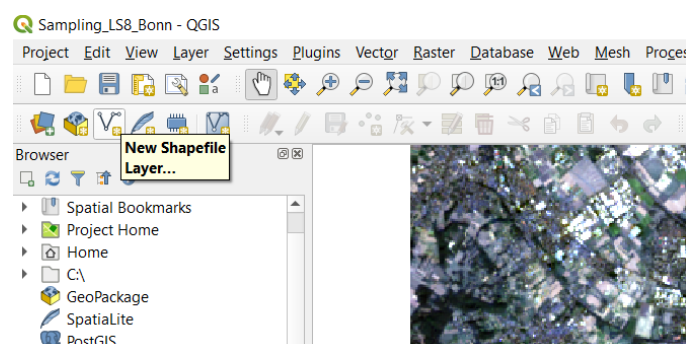
Now, it should be fairly easy to find Bonn. If you like, change the transparency of your LS image via properties -> Transparency -> Global Opacity.

Clipping the raster layer works as follows: go to Raster -> Extraction => Clip Raster By Extent... The Input layer should already be chosen correctly (your LS8 image). Choose Select Extent on Canvas as Clipping extent. You should be able to create a rectangular canvas on your LS image. Approximately find the outer areas of Bonn. We would like to save the result to a file, so give it a proper name, such as "LS8_Bonn_SC20200925100850.tif". Run the process. Afterwards, the file should appear in your layer-window.

If you have started a new QGIS project (or just opened QGIS), the projection of the entire project will be based on the first dataset you load – in this case the raster file. You can see the current projection of the project in the lower right corner of QGIS. If you use our example data set, you should now see "EPSG:32632" there. Click on this entry to get more detailed information about coordinate system of our raster dataset ("WGS84 / UTM ZONE 32 N"). Alternatively, you can double-click the dataset in the Layer Panel and view the Coordinate Reference System (CRS) in the General-tab. We want to generate a new shapefile, which shares exactly this georeference system. This is the best way to ensure that the polygons are geographically correctly located in the end.

Create a New Polygon Shapefile

Click on the New Shapefile Layer icon in the toolbar. If you cannot find this icon, right-click in the toolbar area and make sure there is a check mark next to "Manage Layer Toolbar", which should reveal this icon among others.



Once clicked, the “New Shapefile Layer” dialog will be displayed. Browse in your working directory and give a proper filename, such as “training_data.shp”. Choose “polygon” as the Type. Click on the Coordinate System icon. A new window will pop up, allowing you to choose the CRS of your new shapefile. Choose the same CRS as your raster data (you can use the filter function at the top). On the Fields list, select “id”, and click the “Remove Field” button at the bottom of the list. Under “New field”, type “classes” in the Name box, click on “Add to Fields List”. Finally, this should look like this:

New Shapefile Layer

File name: ments\Studium_Geographie_M.Sc\RSRG\Teaching_material\RESEDA_Tutorial\Analyse\Sampling_QGIS\training_data.shp

File encoding: System

Geometry type: Polygon

Additional dimensions: ☒ None ☐ Z (+ M values) ☐ M values

Project CRS: EPSG:32632 - WGS 84 / UTM zone 32N

New Field

Name: class

Type: abc Text data

Length: 80 Precision:





Add to Fields List

Fields List

Name	Type	Length	Precision
class	String	80	

Remove Field

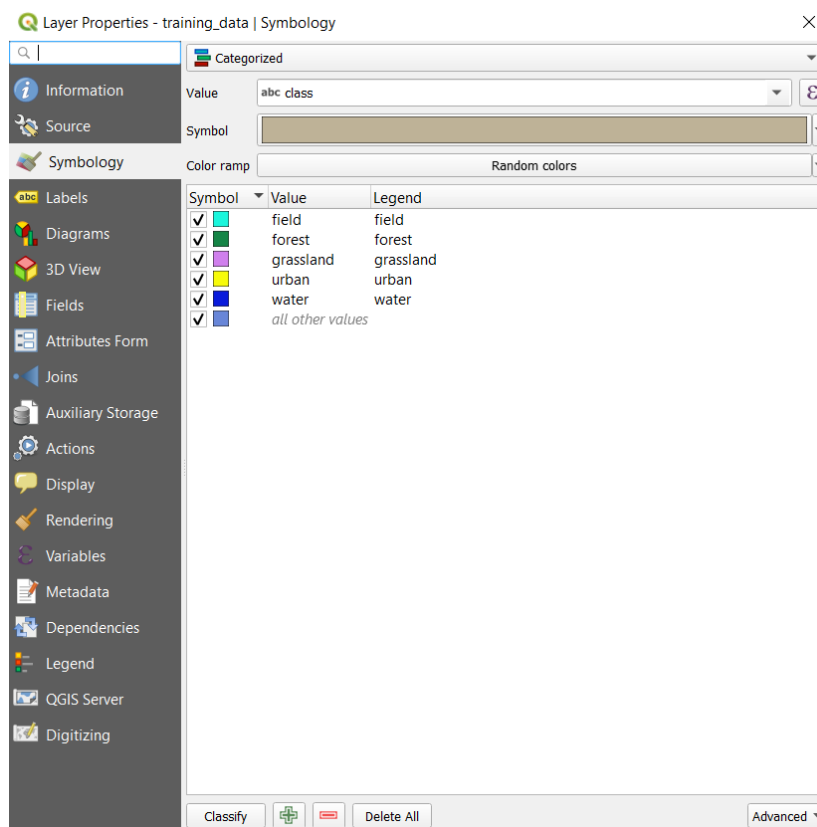
OK Cancel Help

You will be able to see the new shapefile in the Layers Panel of QGIS. Keep in mind our desired classes for the classification: *field, water, grassland, forest, urban*. Select it and press the Toggle Editing  icon in order to activate editing functionalities. Note that a little pencil symbol will show up on top of the layer, indicating that the layer is now editable. Now click on the Add-Polygon-Feature icon. The mouse cursor will now look like a crosshairs. Left-click on the map in the Map View to create the first point of your new feature. Keep on left-clicking for each additional point you wish to include in your polygon. When you have finished adding your points, right-click anywhere on the map area to confirm your polygon geometry. An attribute window will appear immediately, asking for your class label. Input the appropriate class label for your polygon and click OK. Click on the Toggle Editing  icon again in order to end editing and to save your changes by choosing Save. You can edit the shape of a polygon with the Node tool . Delete any unwanted polygons by clicking on the tool called “Select Features by Area or Single Click” . Once activated you can left-click on polygons you want to delete, causing them to turn yellow. Then, press the delete key on your keyboard to remove the polygons (only in editing mode).

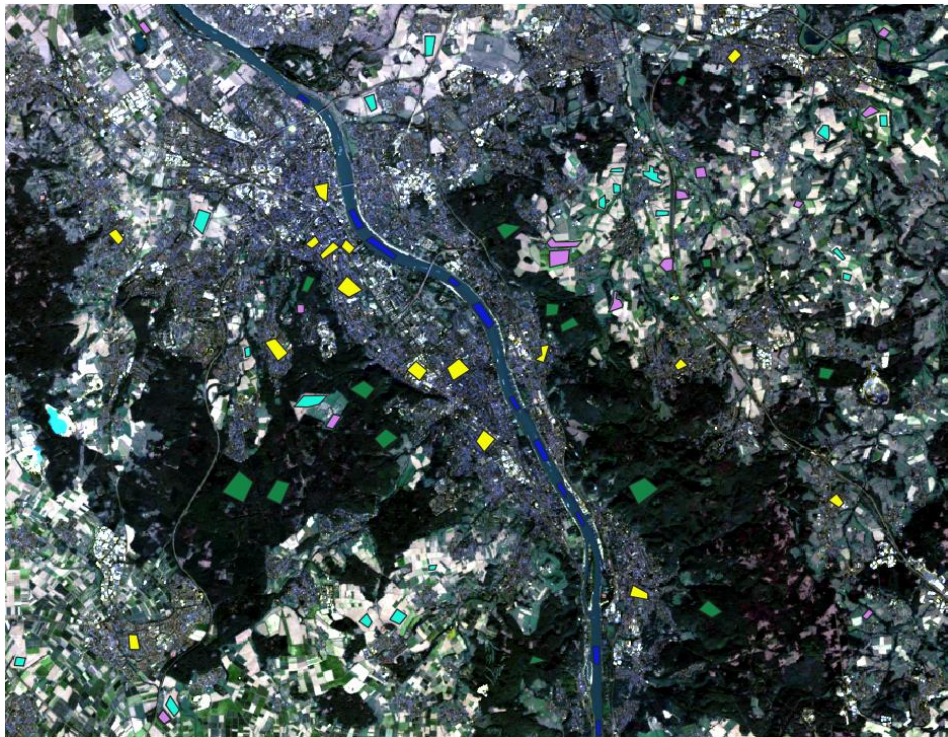
After some time, you should have collected some training areas:



It is recommendable to color the polygons during editing based on the “classes” attribute, which makes it easier for you to estimate the class distribution. Go to the properties of the shapefile in the Layers Panel and navigate to Symbology. Choose “Categorized” at the top field. Ensure that your attribute “classes” is selected in Value. Click Classify once to apply an individual color to each class (click on the colored boxes in order to change the colors) and confirm everything by pressing OK:



This should result in something like this:



If you think you have collected enough samples, save everything by clicking on the Toggle Editing icon again and choose to Save.



We do not need QGIS anymore, so close it.

Questions / prove your knowledge:

- Sum up the requirements to optimally generate training samples!
- Briefly summarize how to approach the amount of training data you need!