**Lab 5: Model Building**

*Geog2011: Introduction to GIScience*

While many GIS users rely on the point and click interface of ArcGIS or QGIS, more complicated analysis tasks often rely on code or models that allow for easy re-analysis or reproducibility. This allows users to redo the analysis if new observations are added or to share code for others to use on their datasets.

In this lab, you’ll be creating your own analysis model in QGIS using the Graphical Modeler. Your model creates distance-based buffers around amenities (such as parks) downloaded from OpenStreetMap. The goal of this analysis is to identify the percentage of Milwaukee residents that live within easy access to park space. You’ll use block-level data from the 2010 access to calculate the percentage of residents within the buffers you create.

In addition to learning how to use the Graphical Modeler, you’ll also get experience with common spatial and vector operations: reproject, clip, filter, and buffer. An analysis of proximity using spatial buffers is common in GIScience, so this lab also gives you a sense of how this kind of project works.

**Workflow**

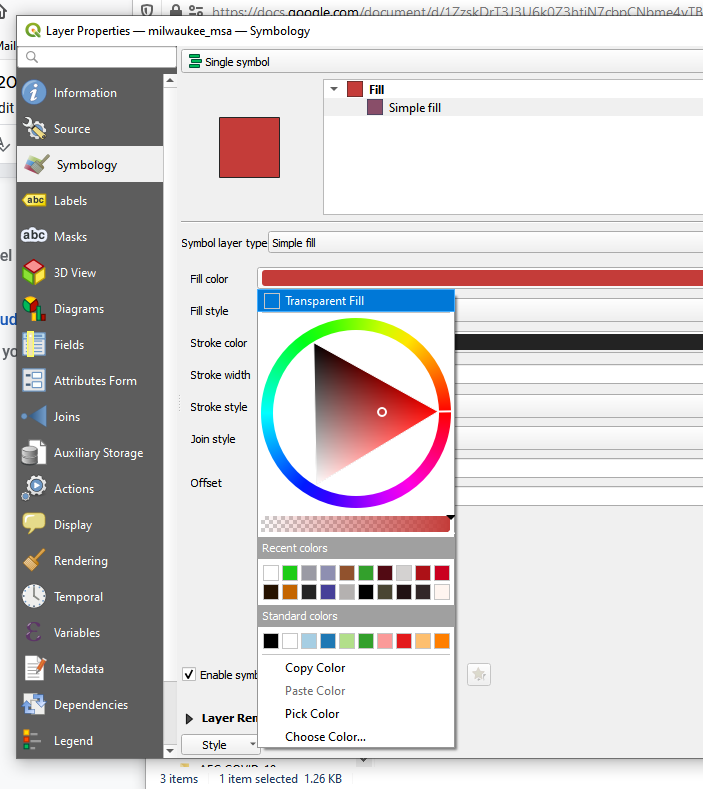
There are four data files to download for this lab, the boundaries of the Milwaukee and Seattle metropolitan statistical areas (MSA) and census blocks within each area. The latter includes the recorded population in 2010. We will be working with Milwaukee data for this lab, but the Seattle data are there for extra credit (see the end of the directions). These are uploaded to ELC as a single zipped folder. Create a working folder on your local computer or flash drive, and then download and extract those files there.

The rest of the lab will follow the following steps:

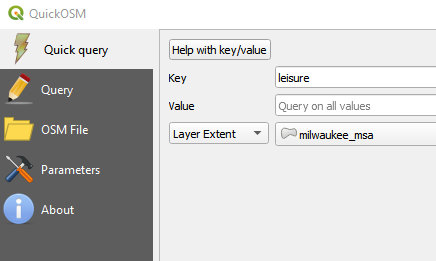
1. Load the study area data and download OSM polygons
2. Create your model parameters (inputs) in the Graphical Modeler.
3. Load the functions to process your data
4. Test the model and map the results
5. Examine the Python code
6. Calculate population accessibility

***Load the study area data and download OSM polygons***

Load the Milwaukee MSA into QGIS. To have some context, you can use the QuickMapServices plugin (see Lab 4) to add a basemap from OpenStreetMap. You can also change the fill of the MSA polygon to transparent so that only the boundaries are visible. Just click on the “Simple fill” for Symbology and click the box for “Transparent fill” under Fill color.



Now you can download data from OpenStreetMap using the QuickOSM plugin (also from Lab 4). Your settings should look like this:



This will download all amenities tagged as “leisure” within the Milwaukee MSA polygon. For this lab, you’ll only be interested in the polygons from the downloaded data.

***Create your model parameters (inputs) in the Graphical Modeler***

Now you’re ready to get started with the graphical modeler. The main purpose of the modeler is to automate a series of steps so that you don’t have to click the same buttons every time. In one sense, it’s just a time saver if you’re lazy. But it also creates a reproducible process, allowing you to remember what steps you followed and share your process with others.

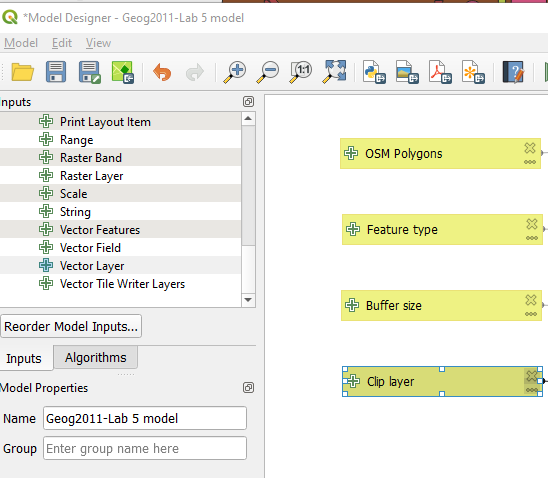
You can open the modeler under Processing > Graphical Modeler. From there, this video walks you through the process of setting up your parameters and doing an initial buffer of your data. Since OSM data sometimes has topology errors (basically the person who entered the data clicked in a bad spot), creating a buffer of 0 allows QGIS to automatically correct any problems so that there’s no errors later on in your process.

* <https://youtu.be/bA-BGOqH2wk>
* (Second tutorial that has a similar workflow: <https://www.youtube.com/watch?v=eZb5VLTc9-o>)

For this model you’ll need the following Inputs as parameters:

* Vector Layer: for the OSM polygons
* String: for the type of feature you’re selecting (park, pavilion, swimming pool, etc.)
* Number: for the size of the buffer you’re creating (will be in meters)
* Vector layer: for the MSA layer you’d like to clip the results to

You should have something that looks like this when you’re done (note that I’ve named all the Inputs based on their function in the model):

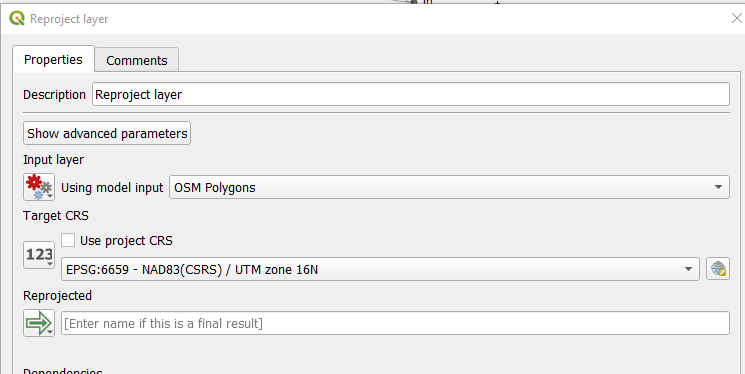


***Load the functions to process your data***

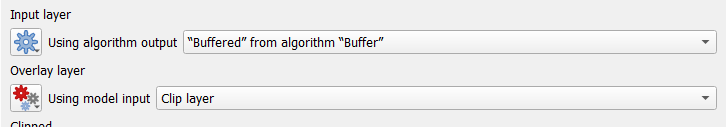
Now you can get to work adding functions. Here’s the “recipe” you will follow:

* ***Reproject your data:*** OSM downloads in WGS84, which uses decimal degrees as a distance measure. You will need to reproject to UTM 16N, which uses meters.
* ***Buffer*** your data with a distance of 0 to remove topology errors (see the video above)
* ***Filter*** the data so that only one subtype of features is selected (Extract by attribute is the function you will use for this).
* ***Buffer*** those features to a set distance, dissolving to a single polygon (requires the buffer size input as a parameter)
* ***Clip*** those features to the MSA boundaries (see the video above, adjusting the clip layer to the appropriate vector layer input parameter).

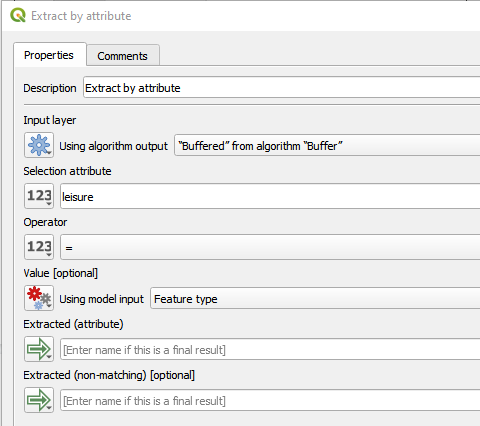
To reproject the layer, search for the ***Reproject Layer*** algorithm. You want the target CRS to be EPSG: 6659 - NAD83(CCSRS)/UTM zone 16N. Note that this won’t change the way the map itself is displayed, just how the underlying data are stored. This will allow you to use meters to determine your buffer size later on.



Both buffer and clip are covered in the walkthrough video. Make sure for the clip that you use the input item you created for the Overlay layer.

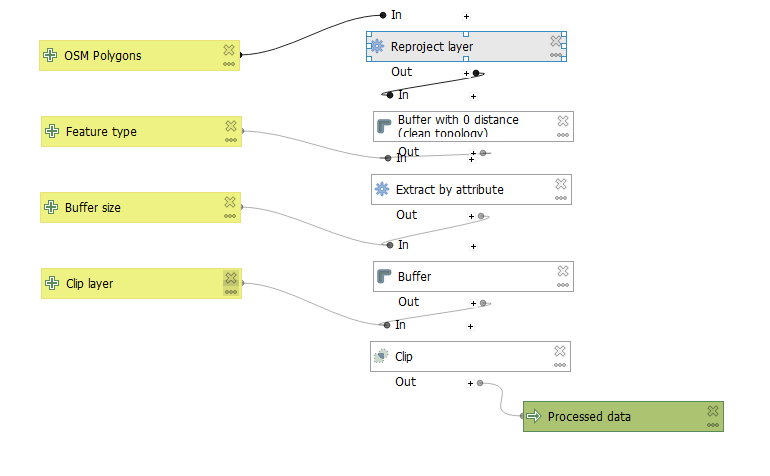


Filtering is a bit more complicated. You want to choose ***Extract by attribute*** from the list of Algorithms. Set it up as you see below. This will select only those features that have the string you enter as the feature type as their value in the leisure column. So if you type “pool” you will only get the observations that are listed as a pool in the leisure column. You can open the attribute table of the OSM polygons to see what different feature types are listed.



For the second buffer, you’ll want to use the “Buffer size” input for the buffer size. Also choose “Yes” for Dissolve result. This will give you one large polygon rather than many individual features.

When you’re done, you should have a model that looks like this:

****

***Run the model***

Now you’re ready to try this out. Click the green triangle button (looks like a Play button) at the top of the window and you’ll get a new window asking for all your inputs. Type in “park” as your feature type and pick a buffer size in meters that makes sense to you. Make sure your OSM leisure polygons are selected as the starting features and the Milwaukee MSA as the clip layer.

You can also test each step as you go, and this is encouraged for a complex model. When you’re done, make sure you save your model. To do so, you’ll also have to enter a model name in the Model Properties window on the left hand side.

Once your model appears to be working, create buffers for three distances: 500 meters, 1,000 meters, and 2,000 meters. Save each to your working folder so you have them if QGIS crashes or you want to come back to them in another session.

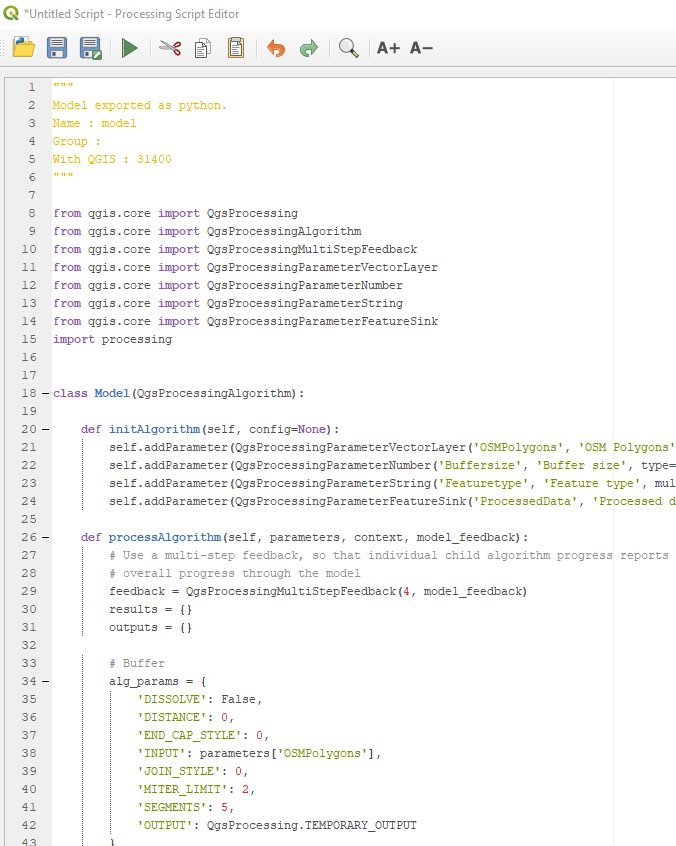
Lastly, ***take three screenshots*** of the map showing each of the park buffers you created, the MSA boundaries, and the basemap you added at the beginning of the lab.

***Examine the Python code***

The graphical modeler (and really all of the functions in QGIS) are point and click interfaces for underlying code written in Python, which is the basis for both QGIS and ArcGIS processing. You can see this code in the Graphical Modeler by clicking on the Export as Script button, which looks like this:



If you aren’t familiar with Python, this is probably hard to read, but notice that there are comments (marked with a hashtag) throughout that walk through the same series of functions as your graphical model, such as the first buffer that starts on line 33 below:



Lines 35-42 simply provide the same set of parameters as you saw in the Algorithm box when you set it up earlier. This is basically looking “under the hood” of the graphical interface of QGIS to see what actual functions are being run.

Find the section of your exported script that handles the **Clip** function. Take a screenshot of it and include it in your lab submission.

*(Note: Some earlier versions of QGIS do not have the Export to Script button. If it is not showing up for you, send us an email with a screenshot of the Graphical Modeler window, and we will work out an alternative.)*

***Calculate population accessibility***

Lastly, you need to calculate the percentage of residents living within each of the three buffers you created. You can do so using the Select by Location tool (under Vector > Research Tools). You want all census blocks that are **within** each set of buffers. You can then calculate the count of people living within those selected blocks by summing the “totpop” field.

Here’s a video that walks through the process:

* <https://youtu.be/XwDIAOkIyto>

You’ll need to calculate four numbers: the total population in all blocks and then the population in blocks within the three buffer areas (500, 1000, and 2000 meters). From that, you can calculate the percentage of the population living within those areas.

**Submitting your assignment**

Your final submission should include these screenshots:

* Your final model (10 points)
* The three map screenshots, showing each of the buffers you created (500m, 1,000m, and 2,000m) (15 points)
* A screenshot of the section of your exported code that does the Clip function (5 points)

In addition, you should submit a ~3 minute video in which you:

* Explain what each algorithm in your model does and why it’s necessary. Pretend you are explaining this to a friend who has some knowledge of GIS but doesn’t know anything about this assignment.
* Provide the results of your accessibility analysis--the percentage of the population living within 500, 1,000, and 2,000 meters of parks as listed on OpenStreetMap.

If you prefer to write this out, you can do that instead.

This will receive up to 10 points based on how clearly you explain the purpose of each algorithm (8 points) and clearly defined criteria for where a new park should be located (2 points).

**For three points extra credit**: Blocks and boundaries for the Seattle MSA are also included on ELC in a zipped folder. Reproduce the analysis you did in Milwaukee using 1,000 meters as the buffer distance. Include a screenshot of the buffered map for Seattle with your submission. In your video or written response, briefly state how Seattle compares to Milwaukee’s park accessibility rate for that distance.