OpenStreetMap data

--- Query, downloading, handling

OpenStreetMap provides tremendous amount of information. While it is becoming indispensable in urban GIS (e.g. routing), it also provides invaluable labels for supervised machine learning methods development (particularly currently burgeoning deep neural networks) in many applications, such automatic road and building delineation from very high resolution (VHR) satellite imagery, which arise hopes in high-resolution global-scale mapping. For examples, for air pollution mapping, we need global road infrastructure, but the transportation network provided by OSM is incomplete in many countries, such as in China and African countries. VHR satellite imagery (e.g. worldview2) and machine learning (particularly deep learning neural networks) are promising techniques to complement the OSM, and evaluate the consequences of directly using the OSM with missing roads to predict air pollution. This is the note of how to download and process OSM.

*Method 1: Download everything and query*

* Use a mirror to download:

wget https://ftp.fau.de/osm-planet/pbf/planet-200413.osm.pbfusing Osmium

* Filter roads: (With osmium):

cmd = "{0} tags-filter {1}.osm.pbf nwr/{2}={3} -o gap\_{4}.osm.pbf".format(osmium, fname, keyword, value, out\_fname)

* Convert to gpkg: (with GDAL)

cmd = "ogr2ogr -f GPKG gap\_{}.gpkg gap\_{}.osm.pbf".format(out\_fname, out\_fname)

Advantage: store everything on hard disk, if you know exactly what you want, this maybe the most straight-forward option.

Limitation: cannot explore data before downloading, also need big data storage

*Method 2: Interactive query, download, and analysis (using OSMnx):*

*Two ways of installing: conda or pip (additional packages needed beforehand with pip), and docker*

1. You can create a conda environment to install the OSMnx, as indicated in the manuscript:

conda config --prepend channels conda-forge

conda create -n ox --strict-channel-priority osmnx

1. run jupyter from docker: <https://hub.docker.com/r/gboeing/osmnx>

*Method 3: Query (using QGIS) osm package, with SQL*