National liveability indicator calculation

October 2018

Contents

1. Overall project structure

A project folder, for historical reasons called 'D:/ntnl_li_2018_template' has been set up for the 2018 liveability indicator calculation.

The project directory contents are as follows:

admin
The 'admin' folder contains key project documentation (meeting notes, literature searches, etc)

data
The 'data' folder contains raw and derived data sources, as well as the project intermediary data files and back up of current SQL database.

maps
Any map files created in the course of script running could be stored here; however, these may be more appropriately located under the relevant study region folder in *data/studyregions*

process
The 'process' folder contains the scripts required to create the project database and index.

- Study regions now share a common ind_study_regions_matrix.xlsx which define all the parameters for running the project, including data sources, parameters and indicator variations for each region
- Study regions are named in the database like 'li_shortname_2018'
    - 'li' is just a prefix to make clear all study regions relate to same project
    - 'shortname' is a lower case version of the study region name, with any spaces replaced using underscores, optionally shortened e.g. 'bris' or 'albury_wodonga'. Study region short names are defined in ind_study_region_matrix.xlsx
    - 2018 is now used to distinguish aim to have 'most current' data
    - The year suffix used was previously '2016' due to intent to targeti in this case, the ABS census year of 2016
- However, to process them we use just their short stub name at the command prompt
  e.g. for li_bris_2016, we instead do: `python 00_create_database.py bris`
- Check your study regions by typing 'python who.py' at the command prompt
- There are / will be some batch files to aid processing. For example, once the first script zero (00_create_database.py) has been run for your study regions, a user can launch a batch script to process script 01 through 06 under the assumption that the configuration file and local environment has been set up correctly.
    - Bec:
      `process_scripts_01_06.bat adelaide ballarat bendigo darwin geelong hobart melb mitchell western_sydney wollongong`
    - Julianna:
      `process_scripts_01_06.bat bris canberra goldcoast_tweedheads mackay sunshine_coast syd toowoomba townsville`
    - Carl:
      `process_scripts_01_06.bat albury_wodonga cairns launceston newcastle_maitland perth`

In general, scripts should be run sequentially. However, script 10 (sausage buffers) should be fine run after script 6 without problems. Script 07 through 09 specifically relate to Origin-Destination processing, and we may prepare a similar batch process for these to run them sequentially for single study regions.

2. Data folder structure

The 'data' folder for the should contain the following folders and subfolders:

ABS

Original source data and derived versions used in processing are respectively stored in the folders 'downloads' and 'derived'. Any downloaded data should be accompanied by a note indicating when and where it was downloaded, what date its current for and who downloaded it.

address_points

Currently the project uses G-NAF address points as the sample units of analysis. Source G-NAF files may be located here, and a geodatabase containing the correctly project files for the project spatial reference system is also located here (GDA2020_GA_LCC.gdb).

destinations

Destination data sources are located in folders here, along with ArcGIS geodatabases for destinations (destinations_2016.gdb), public open space (pos_2018.gdb) and inclusion regions for public open space (pos_includedareas_2018.gdb).

elevation

At a later date, elevation data may be incorporated into the liveability indicator calculation process (e.g. for evaluating accessibility of walking paths based on gradient). An example set of such data for Melbourne is currently included here sourced from the Japan Aerospace Exploration Agency (JAXA) ALOS Global Digital Surface Model "ALOS World 3D - 30m (AW3D30)" .

observatory

Output data files for use in the observatory are located here.

roads

Historically this has included road and intersection data for study regions. However, the current approach employs only the intersections geodatabase and osmnx_nd_template.xml from this folder. For all other study region, road data is stored as .shp files in their respective study region folders. This road data is extracted from OpenStreetMap using OSMnx in a pre-processing step (see Jupyter Notebook "OSMnx - 21 Cities.ipynb") and interpreted into a network dataset using the osmnx_nd_template.xml

study_region

Study region specific input and output data, including database backups, will be located here. Any input resources located in these folders can be pointed to using the shared ind_study_region_matrix.xlsx file.

3. Data acquisition principles

- Keep a provenance.txt file noting all data sources in the same folder where they are stored
- If there are multiple resources in a folder, this file will contain information on all resources
- Retain original data sources, or make a note of where these are located and can be accessed offline in the provenance file
- If data sources are modified, make clear and detailed notes of any processing and who did this so this can be replicated if required

4. Project software environment

This project assumes that you have set things up for GDA2020 use as per the pdf file:
./ntnl_li_2018_template/admin/Updating software for GDA2020.pdf

Specifically:

- Postgresql 9.6 or higher installed w/ PostGIS extension (version 2.4, at time of writing).
    - https://www.enterprisedb.com/downloads/postgres-postgresql-downloads
- ArcGIS 10.6 or higher installed, with 64-bit background geoprocessing
- You have Python 2.7 installed (currently advice is to use the 64-bit version with ArcGIS 10.6.x)
    - *Its assumed that this is the primary python installation you are using! Confirm that it is the first entry when you enter in 'where python' at the command prompt*
- Install psycopg2 (the python library facilitating connection with PostgreSQL) into the ArcGIS 64-bit Python 2.7 installation
    - open a console window to C:\Python27\ArcGISx6410.6\Scripts
    - type: pip install psycopg2
    - change directory to C:\Python27\ArcGISx6410.6
    - type: python
    - type: import psycopg2
    - if an error message appears, something has gone wrong. Ask Carl as there are some quick fixes if it doesn't work
- Install sqlalchemy (for more postgresql integration using python - e.g. pandas dataframe to_sql function)
    - as per instructions for psycopg2, except 'pip install sqlalchemy'

Current recommendation is that you run the scripts from Notepad++ (npp). To do this:

- install
- press F5
- in shortcut location write: C:\Python27\ArcGISx6410.6\python.exe -i "$(FULL_CURRENT_PATH)"
- choose a shortcut like Ctrl Alt X
- Now, when I open a script in the process folder and press ctrl+alt+x this script will run with the correct python version and aware of relative paths from config file

However, it is proposed that it may be more optimal to run scripts from command line using study region as an argument, rather than relying on the current git repository branch to determine study region.

Data sources should be pre-processed as GDA2020 GA LLC using the NTv2 transformation in ArcGIS or QGIS --- the latter may be more reliable (e.g. in ArcGIS the correct transformation options for some GDA94 projected files don't automatically get identified, but in such cases I noticed they were in QGIS -- example is the trial 2018 Vic Open GNAF data from JA.

So --- tranform files to correct CRS (EPSG 7845) first.

5. Configuration

All scripts within the process folder draw on the sources, parameters and modules specified in the file ./process/ind_study_region_matrix.xlsx to source and output resources. It is the best definition of where resources are sourced from and how the methods used have been parameterised.

If you are starting a new project, you can set up the global parameters which (pending overrides) should be applied for each study region in the detailed_explanation' folder.

If you are adding a new study region to an existing project, this study region will be entered as a row in the 'study_regions' worksheet; the corresponding column fields must be completed as required. See the worksheet 'detailed explanation' for a description of what is expected for each field.

If you are running a project on a specific computer that requires some kind of override of the parameters set up above, you can **in theory** use the 'local_environments' worksheet to do this. In practice this hasn't been implemented yet, and the sheet is just a placeholder for the event that such overrides are required.

The file which draws on the project, study region, destination and local settings specificied in the ind_study_region_matrix.xlsx file and implements these across scripts is config_ntnl_li_process.py

The benefit of using the Excel file approach, is that multiple worksheets full of settings for different project aspects can be retained together in the one file, with formatting and additional commentary guiding a user as to what parameters refer to.