Reproducibility review of " Spatial Disaggregation of Population Subgroups Leveraging Self-Trained Multi-Output Gradient Boosted Regression Trees"

Author: F.O. Ostermann ORCiD

To cite the report use

Ostermann, F. O. (2022). Reproducibility review of “Spatial Disaggregation of Population Subgroups Leveraging Self-Trained Multi-Output Gradient Boosted Regression Trees”. https://doi.org/10.17605/OSF.IO/CDFAH

This report is part of the reproducibility review at the AGILE conference. For more information see https://reproducible-agile.github.io/

Reviewed paper

Georgati, M., Monteiro, J., Martins, B., and Keßler, C.: Spatial Disaggregation of Population Subgroups Leveraging Self-Trained Multi-Output Gradient Boosted Regression Trees, AGILE GIScience Ser., 3, 5, https://doi.org/10.5194/agile-giss-3-5-2022

Summary

The paper presents an extensive quantitative study that consists of numerous (pre-)processing steps involving multiple input data sets of different types (e.g., CSV, remotely sensed imagery, and geographic vector data). As such, it is a stongly computational analysis that would benefit greatly from being reproducible and/or replicable.

Fortunately, the authors have made the effort to provide all code in a public GitHub repository. The input data is not provided but sufficiently documented to be recreatable or retrievable from other sources. Unfortunately, despite great support from the corresponding author, the time constraints of this review, coupled with the need to organize the data and the complexity of the workflow, allowed only a partial reproduction of the processing pipeline: the initial dasymetric mapping to generate the first inputs of disaggregated population density, and one of the multiple analysis on that data for the city of Amsterdam.

However, a careful evaluation of the available code led this reviewer to the conclusion that with more time, a successful reproduction of the entire workflow is highly likely. In any case, there is sufficient information to replicate the study for a different geographic area or with different methods or parameters.

Reproducibility reviewer notes

The processing pipelines consists of the following main steps for two cities (Copenhagen and Amsterdam), all implemented in multiple scripts using the Python programming language:

1. Create initial estimates based on aggregated population data, using dasymetric mapping or pycnophylactic interpolation

2. Train a regression model (random forest and gradient tree boosting) using additional, ancillary open data such as the global human settlement layer, the European settlement map, and various measures such as proximity to railway stations and other important infrastructure, using OpenStreetMap and official data sources.

3. Retrain the model a specified number of times.

Since there was no data pre-packaged with the code in the repository, the reviewer asked the corresponding author for help. This was provided very quickly in the form of initial statistical and ancillary data for Amsterdam.

After creating the Python virtual environment from the provided YAML file, an initial run hit a few minor issues caused by the fact the analysis pipeline is long and complex, and minor changes (in this case a new branch in the repository with small code modifications) can easily lead to issues further down the road.

The provided data and code produced the expected output in the form of TIF files that contained the dasymetric mapping of population density for different demographic subgroups (e.g., elderly, etc.).

Next, several analysis runs with varying parameters for the city of Amsterdam were completed without errors and expected output. However, this output is in relatively “raw” form, and the reproduction of the published figures and tables require further work that was beyond the scope of this review.

A personal remark from the reviewer is that this study and the attempted reproduction are a very good example why we need additional incentives and rewards for making research reproducible: The study is a prime example for an effort that would itself benefit greatly from replication, but that could also spawn several useful replications elsewhere (or with different methods for the same areas). The authors have done a laudable effort to provide all necessary code and some of the data, as well as basic initial information on how to use the provided repository, and followed up by quick responses to reviewer requests. However, the analysis is complex enough to require substantially more documentation to prevent potential reproducers to get bogged down in minor issues. At the same time, this reviewer would (could) not expect the authors to invest more effort that they have done so far.