



Geo-X training course

Hands-on 3

Please use the following citation for:

- **This exercise**

Miguel Sanchez-Lopez, Michaela Rychetska, 2026, Hands-on 3: Data Inputs – Geo-X Training Course (Version 1.0.). Climate Compatible Growth. DOI: [to be assigned].

- **Geo-X software**

Sanchez-Lopez, M., Naqvi, S., Leonard, A., Schirren, L., & Hirmer, S. (2026). Geo-X: Generalised geospatial software for production of green commodities. *SoftwareX*, 34, 102712. <https://doi.org/10.1016/j.softx.2026.102712>

Learning outcomes

By the end of this exercise, you will be able to:

- Understand the three types of input data Geo-X requires: spatial hexagons, techno-economic parameters, and the configuration file.
- Fork, clone and set up Geo-X-data-prep, and know where Ghana's prepared hexagons belong in your Geo-X fork.
- Understand the hydrogen and ammonia production pathways, and how they map to Geo-X's parameter structure.
- Read and interpret Ghana's demand definitions for hydrogen and ammonia.
- Configure config.yaml for your first Ghana scenario.

Activity 1 – The three inputs Geo-X needs

Before Geo-X can run an optimisation, it needs three types of input:

- **Spatial hexagons:** the geographic units covering the study area, produced by Geo-X-data-prep.



- **Parameter files:** techno-economic assumptions – costs, efficiencies, capacities, and more.
- **config.yaml:** instructions telling Geo-X what to run – which country, which commodity, which year.

Note: parameters are your assumptions. Geo-X does not prescribe values – it analyses whatever you give it. The quality of your outputs depends on the quality of your inputs.

Activity 2 – Fork, clone and set up Geo-X-data-prep

Geo-X-data-prep is the second repository in the Geo-X ecosystem. It turns raw GIS layers (land cover, slope, water bodies, roads, and more) into the H3 hexagon file Geo-X needs, through three steps: `prep_before_spider.py` applies land exclusion criteria, SPIDER tiles the eligible land into hexagons, and `prep_after_spider.py` finalises the file as `hex_final_[ISO].geojson`.

Try it: set up Geo-X-data-prep the same way you set up Geo-X in Class 2.

- Go to <https://github.com/ClimateCompatibleGrowth/Geo-X-data-prep> and fork it.
- Clone your fork, including its submodules (Slope-Exclusion, GLAES, SPIDER):
`git clone --recurse-submodules https://github.com/your-username/Geo-X-data-prep.git`
- Create and activate the environments: `mamba env create -f environment.yaml` then `mamba activate prep`. Separately, for SPIDER: `cd ccg-spider/prep, mamba env create -n spider, mamba activate spider, mamba install pip gdal, pip install -e`.

Note: you will not run the raw GIS pipeline yourself in this class. We are providing Ghana's prepared hexagon file directly. If you want to see how it is built end to end, the Geo-X-data-prep README walks through the full process; treat it as optional further reading, not required for this exercise.

Try it: add Ghana's hexagons to your Geo-X fork.

- Copy the provided `hex_final_GA.geojson` file into your Geo-X fork, at `data/GA/hex_final_GA.geojson`.



- Confirm it is in place: from the root of your Geo-X folder, in the geox environment, run `ls data/GA/hex_final_GA.geojson`

Note: Ghana's hexagon file contains 177 hexagons. Each one carries land-use exclusion distances (ocean_dist, waterbody_dist, waterway_dist, road_dist) and theoretical renewable potential (theo_pv, theo_turbines). This is what Geo-X will optimise a plant design over, from Class 4 onwards.

Activity 3 – Hydrogen and ammonia pathways

To configure Geo-X correctly, it helps to understand what the model is actually doing. Each device in a pathway is characterised by techno-economic parameters, investment cost, efficiency, and capacity.

- **Hydrogen pathway:** onshore wind and solar generation feed an electrolyser, backed by a battery; hydrogen is compressed, stored, transported, and delivered to meet an inflexible demand.
- **Ammonia pathway:** builds on the hydrogen pathway and adds Haber-Bosch synthesis, a fuel cell, and ammonia storage. Because it combines three interacting sub-pathways, it is the more complex of the two.

Note: the basic modelling shown here is what you will use to build your first scenarios. More sophisticated technical alternatives, demand-side flexibility, power purchase agreements, salt cavern storage, are covered in Class 5 and Class 6.

Activity 4 – Explore the parameter file structure for Ghana

You have been given read-only reference copies of Ghana's parameter files, organised by commodity: `parameters/GA/hydrogen/` and `parameters/GA/ammonia/`.

Try it:

- Open both folders. In `hydrogen/`, you will find `country_parameters.xlsx`, `demand_parameters.xlsx`, `technology_parameters.xlsx`, `transport_parameters.xlsx`, and `conversion_parameters.xlsx`.



- In `ammonia/`, you will find the same core files, but `pipeline_parameters.xlsx` in place of `conversion_parameters.xlsx`.
- Open one or two files in each folder and note what each sheet defines. We will use these in detail from Class 4 onwards.

Note: these are shared reference files. Please, do not edit them directly. When you start building your own case, copy the files you need into your own Geo-X fork, under its `parameters/GA/` folder.

Activity 5 – Read Ghana's demand definitions

A demand centre defines where a commodity needs to be delivered: its location, the annual volume required, and the state it must be delivered in. Choosing a demand centre is an analytical decision – for example, a port for export, a border crossing, or an industrial hub.

Try it:

- Open `demand_parameters.xlsx` in `parameters/GA/hydrogen/`. You will find one demand centre: Accra (5.5, 0.19), 10,000,000 kg/a, demand state NH3.
- Now open the equivalent file in `parameters/GA/ammonia/`. Same location, but 1,000,000,000 kg/a.

Note: both files express demand in NH3 terms, but the quantities are very different. This is not a data error. Geo-X applies a conversion step whenever hydrogen is delivered as ammonia, and because a kilogram of ammonia carries much less usable hydrogen than a kilogram of hydrogen itself, matching the same underlying demand requires a much larger mass of ammonia. Keep this in mind when comparing scenarios later in the course.

Activity 6 – Set `config.yaml` for your first scenario

`config.yaml` is the scenario dial: it tells Geo-X which country, commodity, weather year, generators, and solver to use for a run.

Try it: on your `ghana-h2-nh3` branch from Class 2, open `config.yaml` and set:

- `country: ['GA']`



- `weather_year: [2023]` (or the year agreed in class)
- `plant_type: 'hydrogen'`
- `generators_dict: {'solar':[], 'wind':[]}`
- `solver: 'cbc'`

Save your changes, then commit and push: `git add config.yaml`, `git commit -m "Configure Ghana hydrogen scenario"`, `git push origin ghana-h2-nh3`.

Note: we are starting with `plant_type: 'hydrogen'`. You will set up the ammonia scenario as a second run once we start running the model in Class 4.

Wrap-up

You have completed this exercise. Before moving on, check that:

- Geo-X-data-prep is forked, cloned, and its environments are created.
- `hex_final_GA.geojson` is in place at `data/GA/` in your Geo-X fork.
- You can explain the difference between the hydrogen and ammonia pathways.
- You have located and reviewed Ghana's parameter files (read-only reference).
- You understand why the hydrogen and ammonia demand quantities differ.
- `config.yaml` is set for your first Ghana hydrogen scenario, committed and pushed.

Well done! In Class 4, you will run the model for real, using everything you have set up here.