# Maresme County Gloria analysis

Derek van de Ven
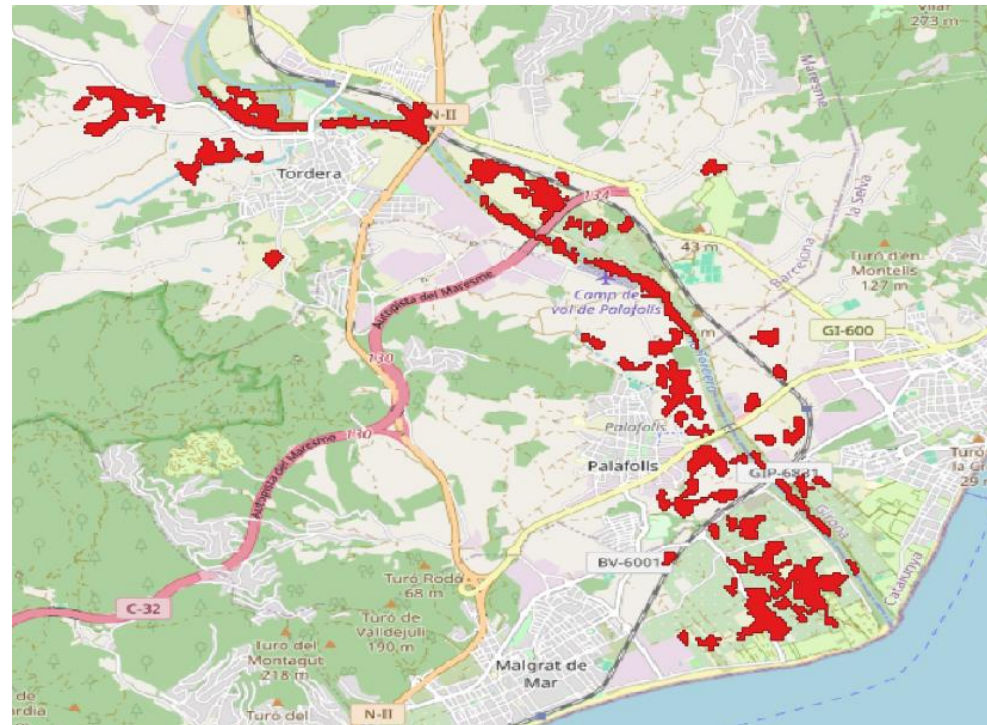
Methodology

- Finding suitable (satellite) data
  - I first wanted to investigate using visual data, but since the results were cloudy I opted for VV radar. I made composites of the radar imagery from VV before and during the flood in Google Earth Engine, and downloaded the data.
- Masking flooded areas
  - I then masked the part that had an increase in VV reflection, without creating many extra flooding areas not near the river (see figure below)

- Population and roads
  - For the population and roads I downloaded data from OpenStreetMap and humdata.org. Then I checked which parts spatially intersected with the mask and summed the length of road or amount of people in that pixel area. I removed tiny polygons so we wouldn't get population info from pixels far away from the river.
- Coding
  - I started with tests to first develop the code to do the processing and then applied this code in the API and Notebook.
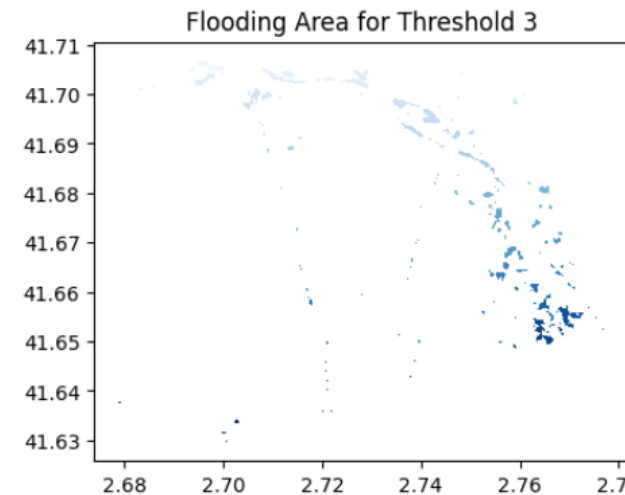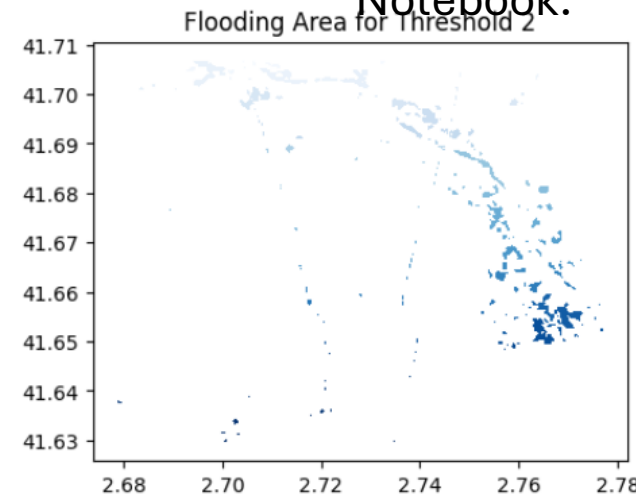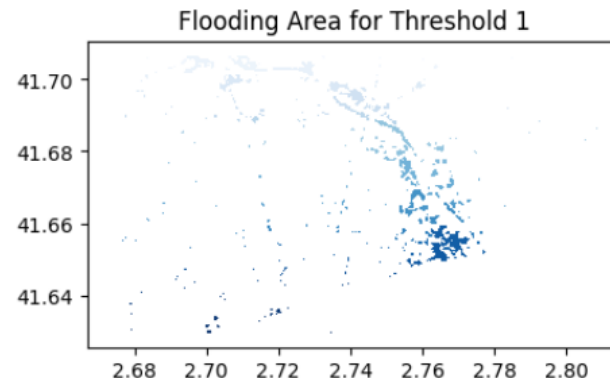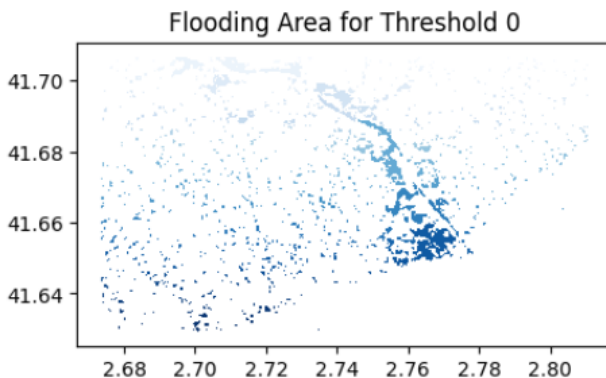
Key findings

- ## My findings indicate the following results:
  - population impacted: 40951 people
  - flooding area km2: 70.0326
  - road impacted: 76659m

API

- I created the two requested api endpoints.
- I opted to use Pydantic and a post endpoint for the /flood_analytics, since a lot of input is needed
- For the /flood_map I simply use query parameters

- Because it is an exercise, I haven't added a database or authentication

Some niceties

- I use uv for dependency management, it's really fast
- I use ruff and pre-commit to make sure my code is all aligned in a nice way
- I use typing to keep the code maintainable
- I test the code by writing the tests. The tests don't check the output but they check if the functionality does what it's supposed to do.

- A nice improvement would be to automate the gathering of data so you don't need to pass along the data in the endpoints.