Large-scale models and simulation methods

Course Project, Spring 2024

The goal of the course project is to perform an agent-based transport simulation in a French city. The simulation has not the ambition to be correct, but rather to show that the underlying tools can be used and that analyses can be performed based on the outputs.

To pass the course project, provide a report that answers the questions indicated in the bullet points. Describe the analysis steps that you have taken, and make sure to document and discuss any problems that you encounter to obtain partial points in case the final results do not match. You may work in groups of **up to 4 people**. Make sure to indicate all names and e-mail addresses clearly on the report.

For the analysis, you may use any tools for visualization and processing. When maps are asked for, you may combine the information from multiple exercises in one map, as long as the required information is visible.

Send your report to **sebastian.horl@irt-systemx.fr** at latest **31/03/2024**.

Link to the course Material on Github

Link to the course slides

Updates Spring 2024

- 18/01/2024: Starting from today the TDs will be in 2B180
- Resulting data from Exercise 01 Exploration: https://drive.google.com/drive/folders/1ut28N-SBsaDlWXudfgfQjgOxZfCnEd8C?usp=sharing

Exercise 1.1: Study area (2 points)

To perform the exercise, please choose a study area, which will be centred on one municipality in France. You are free to choose any municipality that you like to explore, under the following conditions:

- The municipality should be found in the national census data set `[INSEE CENSUS]` (i.e., have more than 10,000 inhabitants).

- The municipality should have at least 2 neighbouring municipalities (with a direct border).

- At least two of the neighbouring municipalities should be included in `[INSEE CENSUS]` (with more than 10,000 inhabitants). Often, there may be others that are not covered by the census data, but which should be taken into account in the analyses.

The analyses of the project will be based on the centre municipality and all neighbouring municipalities. We define this area as the study area.

- Explain which centre municipality you have chosen and where it is located (department and region) (1 point)

- Prepare a map (1 point)
  - that shows the department in which the municipality is located
  - that shows all municipalities in the department
  - that highlights the “study area”, i.e., the selected municipality and all neighbouring municipalities
  - that highlights the selected municipality

For creating the maps, make use of the 2021 IRIS data set from [IGN] which is compatible with the census data.

`[INSEE CENSUS]` https://www.insee.fr/fr/statistiques/6544333
`[IGN]` https://geoservices.ign.fr/contoursiris

Hint: Center municipalities with the indicated requirements are often smaller cities. Avoid using very large cities in case your computational resources are limited.

Exercise 1.2: Territorial analysis I (3 points)

For your study area, perform a territorial analysis to understand who lives in this area.

- Provide a list of INSEE municipality identifiers for all the municipalities included in the study area. In a table, provide the number of samples (data points) in the `[INSEE CENSUS]` for each municipality along with the weighted population count (including zero). (1 point)

- Plot the age distribution of the study area and at least three municipalities in the study area, and report whether you see any differences or not. (1 point)

- Plot the distribution of social-professional categories of the study area and of at least three municipalities in the study area, and report whether you see any differences. (1 point)

Hint: Make sure to use weighted data!

Exercise 1.3: Territorial analysis II (3 points)

For your study area, perform an analysis of population and employment in the area.

- Make use of the aggregated census data `[INSEE AGGREGATED]` to create a bar plot indicating the number of inhabitants in each municipality of the study area. Also, provide this information on a map. (1 point)

- Make use of the URSSAF employment data `[URSSAF]` to create a bar plot indicating the number of employees in each municipality of the study area. Also, provide this information on a map. (1 point)

- Report the total number of inhabitants of the study area and the total number of employees in the study area (1 point)

`[INSEE AGGREGATED]` https://www.insee.fr/fr/statistiques/6543200
`[URSSAF]`https://open.urssaf.fr/explore/dataset/etablissements-et-effectifs-salaries-au-niveau-commune-x-ape-last/information/

Exercise 2.1: Trip production (2 points)

Setting up a full trip generation model is rather complex, hence, only a simple approach shall be used here to prepare data for the simulation. To generate the trips, make use of the aggregated census data [2]. The following model should be used for trip generation:

$$O_i = max(\beta_0 + \sum_s \beta_s \cdot n_s , 0)$$

With O_i indicating the number of originating trips in zone i, s indicating the socio-professional category (SPC), β_i the growth factor for an SPC, and n_i indicating the size of the population older than 15 years with SPC s. The growth factors from Table 1 shall be used.

Apply the model to your study area and solve the following tasks:

- Report how many trips have been generated in total (1 point)
- Report using a plot how many trips have been generated for each municipality in the study area. Also, show this information on a map. (1 point)

Parameter	**CSP**	**Value**
β_0	*Model offset*	27.244
β_1	Agriculteurs exploitants	0.319
β_2	Artisans, commerçants et chefs d'entreprise	0.994
β_3	Cadres et professions intellectuelles supérieures	0.863
β_4	Professions Intermédiaires	0.990
β_5	Employés	0.780
β_6	Ouvriers	0.708
β_7	Retraités	0.120
β_8	Autres personnes sans activité professionnelle	-0.073

Table 1: Trip generation model parameters

Exercise 2.2: Trip attraction (2 points)

To determine how many trips end in each municipality, we make use of a simple model as follows:

$$D_j = \frac{W_j}{\sum_{j'} W_{j'}} \cdot N$$

With D_j indicating the share of trips arriving in zone j, and W_j indicating the employment in zone j. N indicates the total number of trips within the study area. We define the latter variable to indicate that not all commuters generated in Exercise 2.1 stay within the study area and not all employees generated in Exercise 2.2 come from within the study area. For simplicity, assume that the total number of commuters in the zone is the minimum between 70% of the total employment and 70% of the originating commuters, i.e.,

$$N = min(\ 0.7 \cdot \sum_j W_j \ , \ 0.7 \cdot \sum_i O_i \ )$$

Apply the model to the study area and perform the following analysis:

- Report using a plot how many arriving trips have been generated for each municipality in the study area. Also, show this information on a map. (2 points)

<u>In preparation for the next task,</u> proportionately scale your demand per zone (O_i) from Exercise 2.1 so that the total is the same as the number of trips N. In a balanced model, the number of originating flows must match the number of arriving flows over all zones. For that, scale your demand following:

$$O'_i = N \cdot \frac{O_i}{\sum_i O_i}$$

Exercise 2.3: Trip distribution (2 points)

Now that it is established how many trips originate from each zone and how many trips arrive in each zone, a flow matrix F_{ij} indicating the movements from zone i to zone j can be established. The following double-constrained gravity model shall be used:

$$F_{ij} = \frac{O_i}{\sum_j a_j \cdot \rho_{ij}} \cdot \frac{D_j}{\sum_i p_i \cdot \rho_{ij}} \cdot \rho_{ij}$$

With a_j and p_i representing weighting factors (attraction and production) that allow the flow matrix to match the marginal distribution of origins and destinations in the matrix. ρ_{ij} describes the friction term that defines how difficult it is to get from zone i to zone j.

To obtain the weighting factors, the following expressions shall be evaluated iteratively:

$$p_i = O_i \,/\, \sum_j a_j \cdot \rho_{ij}$$

$$a_j = D_j \,/\, \sum_i p_i \cdot \rho_{ij}$$

Make use of the following friction model

$$\rho_{ij} = \exp(\beta \cdot d_{ij} + \alpha)$$

With d_{ij} indicating the distances between zone i and j (in meters) and the parameters in Table 2.

Parameter	Value
β	-1.1e-4
α	-0.4

Table 2: Gravity model parameters

After applying the model, solve the following tasks:

- Show the distance matrix as a table or in a plot. (1 point)
- Report the resulting flows F_{ij} in a table or plot. (1 point)
  - Which pair of zones has the highest flow?
  - Which one has the lowest?

Hints:

- First, calculate the distance matrix D_{ij} between the centroids of the municipalities in the study area. You may use the code developed during the course.
- Next, calculate the friction matrix ρ_{ij} based on D_{ij}.
- Initialize p_i and a_j to 1 and run the formulas iteratively until the values stabilize.
- Calculate the resulting flow using the given formula.

Verification: To verify that your implementation is correct, you may reproduce the following toy example. Based on distances D_{ij}, and the given values for O_i and D_j, you should obtain the flow matrix indicated below.

Zone i/j	**1**	**2**	**3**	**4**
Origins O_i	3396	5442	43196	5681
Destinations D_j	9462	2294	48377	6621

Table 3: Demand and destinations for the toy example

D_{ij}	**Destination** j			
Origin i	**1**	**2**	**3**	**4**
1	0	4066	6595	9462
2	4066	0	4340	10299
3	6595	4340	0	7287
4	9462	10299	7287	0

Table 4: Distance matrix for the toy example

F_{ij}	**Destination** j				
Origin i	1	2	3	4	∑
1	74	220	2661	439	**3396**
2	61	445	4415	518	**5442**
3	247	1463	37659	3825	**43196**
4	38	163	3640	1837	**5681**
∑	**9462**	**2294**	**48377**	**6621**	**57715**

Table 5: Resulting flow matrix of the toy example (values rounded down)

Exercise 3.1: Disaggregation (3 points)

The generated demand shall be simulated in an agent-based simulation later on. For that, the data needs to be disaggregated. Generate at least 1,000 individual trips based on your flow matrix F_{ij} and assign randomized coordinates from within the respective zones. Also, add a randomized departure time T to each generated trip based on the following normal distribution:

$T \sim N(\mu = 8 \cdot 3600,\ \sigma = 3600)$

If departure times smaller than zero are sampled, reset them to zero. Provide the following information in the report:

- Plot the generated trip pairs on a map using a line between origin and destination. (2 points)
- Plot the distribution of the departure times as a histogram or CDF. (1 point)

Hints:

- First, convert your flow matrix F_{ij} into a probability matrix P_{ij} with $\sum_{ij} P_{ij} = 1$ so you can sample individual pairs of origin and destination. Based on the probabilities, sample as many combinations of origin and destination as you required.
- For each pair, sample a random origin coordinate, and a destination coordinate from within their respective zones.

Bonus Exercise 3.2: Routing (4 points)

We now want to find out which roads are used by the generated trips. For that, network data from OpenStreetMap shall be used and each trip shall be routed on that network. After, we can aggregate the total travelers on each link:

- Convert a road network from OpenStreetMap data such that you can use it for routing. Plot the network on a map of your study area. (2 points)
- Perform a routing of all generated trips on the network. Count the number of trips traversing each edge of your network and indicate the flow on each link on a map. (2 points)

Hints:

- Download a cut-out of your study area's department from Geofabrik `[GEOFABRIK]` in PBF format and save it, for instance, as `auvergne.pbf`. Alternatively, you may use existing tools online that allow you to draw a perimeter on a map and download the data.
- Download the Make Polygon.ipynb notebook from the course material on Github. Have a look in the code comment to see how it works. You can use it to convert an output of your study area (select the municipalities that belong to your study area and save the geometry as `GPKG`) to the `poly` format, which we will need later on.
- Use osmosis `[osmosis]` to cut out the study area defined in poly format from the OSM data using the shape defined in the `poly` file. At the same time, we already filter out everything that is not a road ("highway" in OSM terms) to reduce the file size. The following command line can be used for that:

```
osmosis --read-pbf /path/to/your/cutout.osm.pbf --tag-filter accept-ways highway=*
--bounding-polygon file=/path/to/your/study_area.poly completeWays=yes --used-node --write-xml
/path/to/the/output.osm.xml
```

- You may load the network data using the osmnx library in Python and further process the data. Make sure to find the *nearest node* for the origins and destinations of each trip and then perform a *shortest path* routing. Follow the logic of the Exercise 05 notebook.

`[GEOFABRIK]` http://download.geofabrik.de/europe/france.html

`[OSMOSIS]` https://github.com/openstreetmap/osmosis/releases

Instructions for using conda at UGE

- Download miniforge
  https://github.com/conda-forge/miniforge

OS	Architecture	Download
Linux	x86_64 (amd64)	Miniforge3-Linux-x86_64

- Open a terminal and call `chmod u+x Miniforge3-Linux-x86_64.sh` to make the file executable

- The call `./Miniforge3-Linux-x86_64.sh` which will guide you through the installation

- Answer the last question (conda install) with yes

- Once you open a new Terminal, you are already in a conda session

- You can call `conda install ABC` to install package ABC

- Run `conda install jupyter`

- You can then start a jupyter session by calling `jupyter` from the terminal