Visualization
Prof. Bernhard Schmitzer, Uni Göttingen, summer term 2024

Exam 2024-09-16

General remarks.

- The exam is built on three datasets provided in `data/problem.X/`. Each dataset is briefly described in a corresponding meta file.
- Work with Python and jupyter notebooks, e.g. via the GWDG jupyter server at `https://jupyter-cloud.gwdg.de/`.
- Your submission should consist of (multiple) notebook files (optionally in text format, e.g. via jupytext), combined into a single zip/tar archive.
- Do not include the data files in your submission.
- Submit via StudIP, similar to the problem sheets. Submission deadline: 2024-09-27 23:59
- Indicate your enrolment numbers clearly at the beginning of each notebook file.
- Groups of two students do not have to complete problem 2.
- To judge the expected amount of work in the assignments, an approximate number of suggested charts is given in parentheses. There is no single right solutions to the assignments. Sometimes more or even less charts may be appropriate, depending on your approach.
- If you do not understand a problem description, do not hesitate to contact `schmitzer@cs.uni-goettingen.de`.

Problem 1: NASA/JPL Small-body database.

In this problem we work with data on asteroids in the solar system.

1. Import the data file `sbdb.csv`. Filter out rows for which `a` is negative or nan. Convert the columns `first_obs` and `class` to appropriate dtypes.
2. Show the distribution of the parameter `a` in the range from 0.1 AU to 10^5 AU. In addition, show the joint distribution of parameters `a` and `e` (same range for `a`). (2 charts)
3. Show the distribution of the parameter `a` in the range from 1 to 6 AU. Show the parts of the distribution that belong the the asteroid classes `IMB`, `MBA`, `OMB`, and `TJN`. Also highlight the Kirkwood gaps associated with resonances 5:1, 4:1, 3:1, 5:2, and 7:3 that appear approximately at values $\mathtt{a} \in \{1.78\,\mathrm{AU}, 2.07\,\mathrm{AU}, 2.50\,\mathrm{AU}, 2.83\,\mathrm{AU}, 2.96\,\mathrm{AU}\}$. (1 chart)
4. Select now the subset of the rows for which valid values are given for the columns `diameter`, `albedo`, and `H`.
5. Analyze and visualize how `H` is approximately a function of `diameter` and `albedo`. (2 charts)
6. Analyze and visualize how `first_obs` is approximately a function of `H`. What is the influence of `a`? Highlight the asteroids belonging to class `TJN` in this relation. (2 charts)

Problem 2: Voyager space probe trajectories.
In this problem we visualize the trajectories of the two voyager space probes through the solar system.

1. Import all 11 data files for 8 planets, the sun, and the two voyager space probes.
2. Create two static overview plots of the space probes' trajectories in the XY-plane, once for the region up to Jupiter's orbit, and once for the region up to Uranus' orbit. Find a way to visualize the substantial divergence of the probes from the XY-plane after their last gravity assists. (2 charts)
3. Show the absolute speed (magnitude of the V-vector) of the space probes over time, to emphasize the effect of the gravity assists. (1 chart)
4. Generate a dynamic animation of the probes' and planets' trajectories. (1 chart/animation)

Problem 3: Public transport in Schleswig-Holstein.
In this problem we visualize the public transport network in Schleswig-Holstein (and slightly beyond). *Hint:* The data is given as a relational database and the problem will require several merges/joins between the tables. Make sure you have a rough understanding of the `pd.merge` function.

1. Import all provided tables.
2. For all agencies, show how many routes, and of which type, they provide. (1-2 charts)
3. Compute the length of all paths represented by the `shapes` table. For this you can simply compute the Euclidean length of each line sequence (see comment in meta file). Next, associate to each row in `routes` a length, via the `trips` and `shapes` tables. I.e. use the `trips` table to find shapes (and their lengths) that are linked with a given route. When multiple lengths are linked, pick the maximal one. Some routes will be dropped, since no shape is available for them. Finally, show the distribution of the remaining routes with respect to type and lengths. (1-2 charts)
4. Similar as above, for each row in `stops`, find the number of distinct routes, that service this stop. Do this separately for bus and train routes. For this, use the `trips` and `stop_times` tables. A route is assumed to service a stop, if there is at least one trip for this route that stops at the stop. Show the distribution of routes per stop, and the 'hubs' of the transport network, i.e. the stops to which most routes (bus, train, and both) are associated. (2-4 charts)
5. Create a simplified map of the traffic network. It should contain all routes to which a shape was associated. Use the position coordinates associated with shapes and stops. For orientation you can use the polygons contained in vg2500.npz. See data meta file for some more details. Drawing all stops is probably not practical, so you must choose some selection criterion. It might be appropriate to make the map interactive, to be able to use zoom, pan, and hover effects. (1 chart)