HONR 39900 – Homework 8

Justin A. Gould
gould29@purdue.edu

April 8, 2021

DUE DATE: 2021/11/08 23:59 EDT

Homework Instructions
To receive credit for the assignment, do the following:

1. Create an `.ipynb` file, and name it: `purduealias_honr39900_homework_number.ipynb` (e.g., `gould29_honr39900_homework_1.ipynb`)

2. Show all your work and follow the instructions below very carefully.

3. Please submit any requested `.gpx` files and OSM maps, too.

4. Submit a printout (e.g., as PDF) of your `.ipynb` file–and the file itself–to Brightspace by the due date.

5. You must show **all** your work and provide comments in your code explaining what you are doing.

For grading this assignment, I will not leverage unit tests. I will look at the printout of your `.ipynb` file. When in doubt, please show and comment all your work.

Problem 1

Map Matching GPS Traces with OSM #1 – 10 points
In the homework directory, there is a folder called `question_1_resources`. Within this folder, you will find GPS traces (`gps_1.gpx`) and a map (`osm_1.xml`). Please do the following:

- (8 points) Map match (`gps_1.gpx`) to (`osm_1.xml`) with non-emitting states. You can use any set of matching parameters. You must `print()` the states.
- (2 points) Visualize the map-matched GPS route overlaying the provided OSM map (`osm_1.xml`).

Problem 2

Map Matching GPS Traces with OSM #2 – 15 points
You will follow steps very similar to question 1. However, this time, you will generate your own data! This tests your ability to use OSM and other open source map tools, which is very common in industry. Please see the below procedures.

Procedures for Question 2:

1. Create a GPS trace consisting of at least 20 GPS points that are slightly off the road using https://gpxstudio.github.io/. The exported file should be `.gpx`, as in our lecture notebook.
2. Using the Overpass API (i.e., `overpass-api.de` from our in-class notebook...see the documentation for more information), get the OSM road networks for the region of your GPS trace (i.e., you should have two (2) files similar to `sample_osm.xml` from class). **TIP:** To reduce the compute requirements to run map matching on your map, set a bounding box for the Overpass API. For more information on bounding box values, see the OSM website.
3. Map match your GPS trace to the OSM map you generated.
4. Visualize the map using the OSM overlay.

Please do the following:

- (6 points) Map match your generated GPS trace to your generated OSM map with non-emitting states. You can use any set of matching parameters. You must `print()` the states.
- (2 points) Visualize the map-matched GPS route overlaying the provided OSM map (`osm_1.xml`).
- (4 points) Generate your GPS trace data and submit the `.gpx`
- (3 points) Generate the OSM map and submit the `.xml`