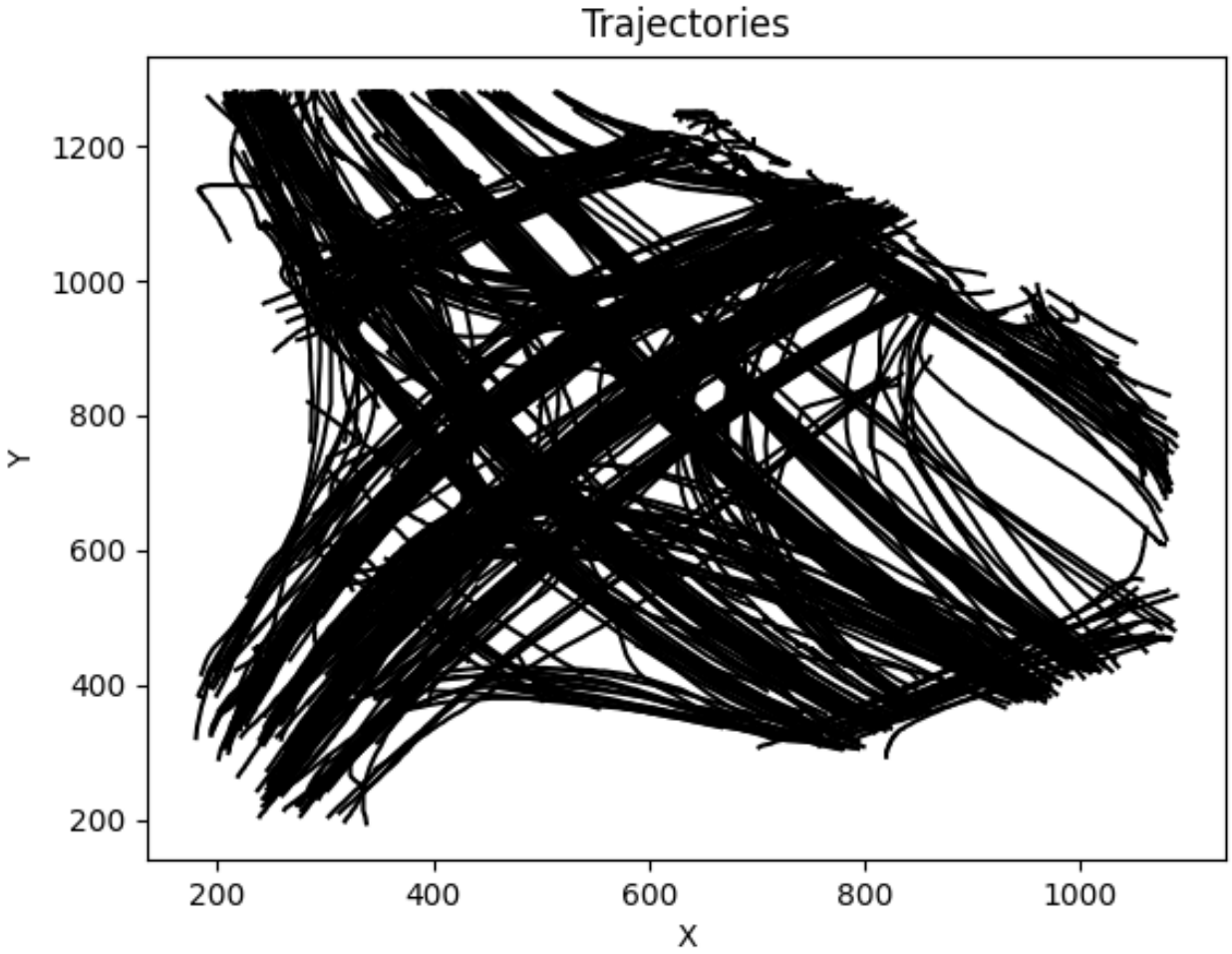

TRAJECTORY ANALYSIS

A trajectory is typically defined by a series of points, where each point is represented by a pair of coordinates (latitude and longitude in geographical terms). These points collectively form a track that illustrates the movement of the object over time.

In your task, the objective is to process a JSON file (see link below) containing trajectory data. Each trajectory is composed of a sequence of coordinate pairs, you'll need to group trajectories with similar directions, and assign colours to these groups.

Your task is to develop a robust Python project that can process trajectory data from a given JSON file (see link

below). The project should accomplish the following objectives:

- Read and interpret the JSON data from the specified file (i.e., "trajectories.json"), organising it into appropriate data structures that facilitate further processing.

- Incorporate any suitable approach to ascertain the direction of each trajectory, and subsequently group trajectories demonstrating similar directions, assigning unique colours to each distinct group.

- Provide visualisation by grouping the trajectories by colour.

Your submission should include a link to your Git Repository including a README.MD file to document your approach, reasoning, implementation details along with the final result.

To achieve this, you're expected to adhere to best practices of coding, must follow clean code, type hints, doc-strings, and the principles of object-oriented programming, utilise encapsulation and design patterns as required. You should also design and conduct unit tests to verify the correctness of your solution and strive for code quality suitable for production. To aid clarity, you should use comments to explain key implementation decisions.

Your solution should also showcase your ability to manage data processing, computations, and visualisation effectively. Feel free to incorporate hyper-parameters as required, each equipped with default values. Additional bonus points will be awarded based on the accuracy of implementation and its adaptability across various use cases.

trajectories.json
https://drive.google.com/file/d/1hBUfx3LTajUR-nAHXbb2LdTQ7MnhHyJS/view?usp=share_link