COMP47670 Assignment 2

Overview:

The objective of this assignment is to scrape a collection of GPX files from the http://mlg.ucd.ie/modules/python/EVdata web pages and perform some analysis. GPX is an XML-based format for GPS data[1]. These GPX files have been copied from eurovelo.com.

The EuroVelo website provides GPX files for 17 long distance cycle routes. These routes are divided into stages and the focus of this assignment is to generate summary statistics (distances and elevation) for these stages. The idea being that these statistics would help people identify stages that would meet their interests and abilities.

Your submission should include two notebooks, one for the scraping (Task 1) and another for the analysis (Tasks 2 & 3).

Task 1. Data Collection

1. Scrape the GPX files from http://mlg.ucd.ie/modules/python/EVdata. There are 17 Eurovelo routes so you should get 17 GPX files. You may find BeautifulSoup to be useful for this task. The 17 GPX files should be stored in a data folder/directory where they can be accessed for Tasks 2 and 3.

   Hint: The urllib.request library has a handy function called urlretrieve for moving files from one location to another:
   https://docs.python.org/3/library/urllib.request.html#urllib.request.urlretrieve

Task 2. Generate summary statistics

1. Each route is presented in a single GPX file made up of thousands of data points, e.g. :

   ```
   <trkpt lat="48.577254277197" lon="-3.8284097146243">
            <ele>17.1</ele>
    </trkpt>
   ```

   These are grouped into stages marked by the <trkseg> tag.

   Use the gpxpy library (https://pypi.org/project/gpxpy/) to load the GPX files into the second notebook.

   Hint: The example at https://pypi.org/project/gpxpy/ shows how to use the gpxpy library to parse the gpx data. Identify the gpx objects representing individual tracks; what methods are available with these objects?

[1] https://en.wikipedia.org/wiki/GPS_Exchange_Format

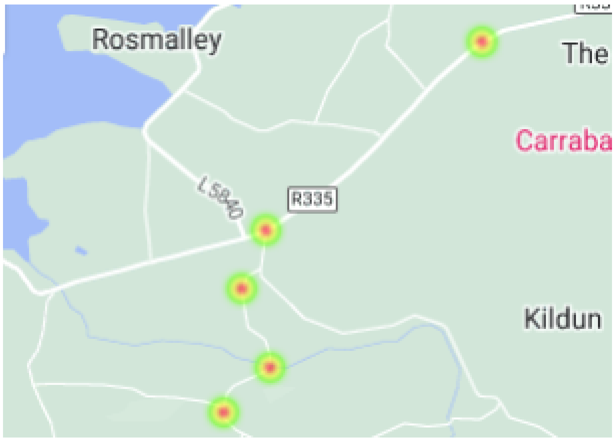
2. For a given route (say EuroVelo 6) build a dataframe with summary statistics for each stage, i.e. name, length in km, total uphill and downhill.

   see: https://pypi.org/project/gpxpy/

3. Write functions to find the longest and hilliest stages in a given route.
   1. What is the longest stage in EuroVelo 6?
   2. What is the stage in EuroVelo 1 with the most uphill?
4. A typical requirement in exploring the EuroVelo data would be to find a sequence of flat stages for a short holiday. Write a function to meet this requirement.
   1. What are the three flattest contiguous stages in EuroVelo 2?
   2. Find the five hilliest (most uphill) contiguous stages in EuroVelo 1.

Task 3. Test the accuracy of the distance estimates

The figure below shows sample GPX track points shown in green/orange. The **`length_3d`** method in the **`gpxpy`** library effectively calculates straight-line distances between these points so the distance estimates are underestimates.



The extent of these underestimations can be estimated using a mapping service API, e.g.

- MapBox https://www.mapbox.com
- Googlemaps https://pypi.org/project/googlemaps/ (requires credit card details)
- TomTom https://developer.tomtom.com

The free tier options of these services will be adequate for this exercise. The navigation facilities provided by these APIs will provide distance estimates between GPX points that should be more accurate than the straight line distances.

1. Focusing on some of the shorter routes (e.g. EuroVelo routes 14 and 19), provide an estimate of the error (underestimate) in the **`gpxpy`** track lengths obtained using the **`length_3d`** method.
2. Discuss the merits of this error estimation, are all the discrepancies on the GPX side?

Hints:

- The MapBox and TomTom APIs are rate limited so your process will need to *sleep* between API calls.
- Sometimes the calls to the directions APIs return empty JSON. It is ok to use try/except error handling to skip a few samples.

Guidelines:

- The assignment should be completed individually. Any evidence of plagiarism will result in a 0 grade.
- The grade awarded will depend on the complexity of the analysis and level of detail, i.e., data preprocessing, underestimation assessment etc.
- Submit your assignment via the COMP47670 Brightspace page. Your submission should be in the form of a single ZIP file containing the two notebooks (i.e. IPYNB files) and your data as stored in Task 1.
- Hard deadline:
  - 1-5 days late: 1 grade point deduction, e.g. B to B-
  - 6-10 days late: 2 grade point deduction, e.g. B to C+
  - Assignments will not be accepted after 10 working days without Extenuating Circumstances formally approved by UCD.