LiDAR Data Science Assessment

Greetings budding data scientist. This document describes the data science assessment to see where your strengths in data science lie, so that we can assess your fit for an internship.

In this scenario, we have been hired by an electricity network to help process Drone based LiDAR data of their electricity lines. Our task is to create a python package that can identify how many wires are present in a lidar point cloud and generate 3D catenary models of these wires.

1.1. Project tasks and deliverables

For this project, you are given LiDAR point cloud datasets (in .parquet files found here) and be tasked to generate best fit Catenary models on each wire within the dataset. As a reminder, a Catenary is a curve that describes the shape of a flexible cable subject to its own weight. It is described by the following equation:

$$y(x) = y_0 + c \times \left[\cosh\left(\frac{x - x_0}{c}\right) - 1\right]$$

Where c is a curvature parameter, x is the distance along the wire x_0 is the x value of the trough, y is the elevation of the wire, and y_0 is the lowest elevation of the wire.

Note that each of the lidar files present different issues to work through to improve your algorithm. Also note that the above equation is 2-D whereas the point clouds are 3-D, additional parameters will be required to adapt the above.

We encourage you to use your creativity for this task but here are a few suggested steps to get you started:

- Cluster the point clouds per wire, you can use libraries such as scikit-learn to do so.
- Find a plane of best fit for each wire cluster.
- Find the best fitting catenary within the above plane.

This code must be shared via a github, gitlab, or bitbucket repository. Along with the code, we would like you to write a short summary in the README of your repo documenting the steps of your algorithm.

1.2. How you will be assessed

We will be focusing on your coding abilities as well as your knowledge of the steps in a data science pipeline. Notably we are looking for:

- **Readability:** Can we easily tell what is going on in your code via appropriate module organisation?
- **Reproducibility:** Can we pull and run your code without any issues?
- **Best Practices:** Are you using best practices to make your code easier to share and maintain including:
    - Organising changes to code sensibly in a git commit, and giving that commit an informative commit message
    - Implementing “don’t repeat yourself” practices, and choosing the right paradigms for the coding problem (i.e. functional vs object oriented)
    - *Desired*: Including automated testing and documentation

- **Approach:** You will also be assessed on how you approached this technical problem, and the logical structure of the analysis.