**Article CAGEO\_2019\_443**

**Computers and Geoscience:** Applying Machine Learning to LIDAR Pulse Return Meta-Data to Improve Bathymetric Mapping in Shallow Water

**Computer Code Availability**

The following code was developed by the lead author (name and contact details have been provided with the article submission) from November 2018 to May 2019. It has been developed in Python using publically available software libraries. All programs are less than 50kb in size. Source code will be made available on github if the article is accepted.

|  |  |
| --- | --- |
| **Program Name** | **Function** |
| *getflightpath\_and\_edge\_scanangle.py* | Extract edge point clouds for each flight path from lidar point cloud. |
| *getflightpath\_edge\_and\_summarise.py* | Extract definitive edge points from the edge point clouds and summarize (e.g., calculate maximum deviation from a straight line) for each flight path. |
| *read\_subset\_SBET.py* | Identify and extract SBET data for individual flight paths. |
| *uncert\_to\_las\_points.py* | Add SBET uncertainty data to individual pulse returns. |
| *get\_scan\_direction\_scanangle.py* | Determine scan direction for each pulse return. |
| *LAS\_Make\_DEM\_extract\_topo\_for\_las.py* | Calculate topographic variables depth, slope, and aspect. (This is run once for each spatial resolution 1m and 5m.) |
| *join\_diff\_res.py* | Add topographic variables of both to each pulse return. |
| *LAS\_add\_horiz\_angle\_via\_chunking.py* | Add azimuthal information to each pulse return. |
| *final\_drop\_vars.py* | Eliminate superfluous variables from data set. |
| *LAS\_MLPClassifier\_use\_stderr.py* | Add topographic variables, fit one of three models (multi-layer perceptron (MLP), extreme gradient boosting (XGB), or regularized logistic regression (Logistic), and output goodness-of-fit summary information. |
| *LAS\_XGBoost\_use\_stderr.py* |
| *LAS\_Logisitic\_use\_stderr.py* |