**Uwescience: Datasci\_course\_materials**

**Introduction:**

In this assignment, you will be working with data from the SeaFlow environmental flow cytometry instrument. A flow cytometer delivers a flow of particles through capilliary. By shining lasers of different wavelengths and measuring the absorption and refraction patterns, you can determine how large the particle is and some information about its color and other properties, allowing you to detect it. The technology was developed for medical applciations, where the particles were potential pathogens in, say, serum, and the goal was to give a diagnosis. But the technology was adapted for use in environmental science to understand microbial population profiles.

The SeaFlow instrument, developed by the Armbrust Lab at the University of Washington, is unique in that it is deployed on research vessels and takes continuous measurements of population profiles in the open ocean. The scale of the data can be quite large, and is expected to grow significantly: A two-week cruise from one vessel can generate hundreds of gigabytes per day, and the vision is to deploy one of these instruments on not only research vessels but the commercial shipping fleet as well. While there are a number of challenging analytics tasks associated with this data, a central task is classification of particles. Based on the optical measurements of the particle, it can be identified as one of several populations.

**Data:**

You are provided a dataset that represents a 21 minute sample from the vessel. This sample has been pre-processed to remove the calibration "beads" that are passed through the system for monitoring, as well as some other particle types. The columns of this dataset are as follows:

* **file\_id:** The data arrives in files, where each file represents a three-minute window; this field represents which file the data came from. The number is ordered by time, but is otherwise not significant.
* **time:** This is an integer representing the time the particle passed through the instrument. Many particles may arrive at the same time; time is not a key for this relation.
* **cell\_id:** A unique identifier for each cell WITHIN a file. (file\_id, cell\_id) is a key for this relation.
* **d1, d2:** Intensity of light at the two main sensors, oriented perpindicularly. These sensors are primarily used to determine whether the particles are properly centered in the stream. Used primarily in preprocesssing; they are unlikely to be useful for classification.
* **fsc\_small, fsc\_perp, fsc\_big:** Forward scatter small, perpendicular, and big. These values help distingish different sizes of particles.
* **pe:** A measurement of phycoerythrin fluorescence, which is related to the wavelength associated with an orange color in microorganisms
* **chl\_small, chl\_big:** Measurements related to the wavelength of light corresponding to chlorophyll.
* **pop:** This is the class label assigned by the clustering mechanism used in the production system. It can be considered "ground truth" for the purposes of the assignment, but note that there are particles that cannot be unambiguously classified, so you should not aim for 100% accuracy. The values in this column are crypto, nano, pico, synecho, and ultra

**Steps:**

**Step 1: Read and summarize the data**

* Using Python, read the file seaflow\_21min.csv and get the overall counts for each category of particle.

**Step 2: Sanity check the data**

* As a data scientist, you should never trust the data, especially if you did not collect it yourself. There is no such thing as clean data. You should always be trying to prove your results wrong by finding problems with the data. Richard Feynman calls it "bending over backwards to show how you're maybe wrong." This is even more critical in data science, because almost by definition you are using someone else's data that was collected for some other purpose rather than the experiment you want to do. So of course it's going to have problems.
* The measurements in this dataset are all supposed to be continuous (fsc\_small, fsc\_perp, fsc\_big, pe, chl\_small, chl\_big), but one is not.
* There is more subtle issue with data as well. Plot time vs. chl\_big, and you will notice a band of the data looks out of place. This band corresponds to data from a particular file for which the sensor may have been miscalibrated. Remove this data from the dataset by filtering out all data associated with file\_id 208.

**Step 3: Split the data into test and training sets**

* Divide the data into two equal subsets, one for training and one for testing. Make sure to divide the data in an unbiased manner.

**Step 4: Plot the data**

* Plot pe against chl\_small and color by pop

**Step 5: Train a decision tree.**

* Train a tree as a function of the sensor measurements: fsc\_small, fsc\_perp, chl\_small, pe, chl\_big, chl\_small

**Step 5: Evaluate the decision tree on the test data.**

* Use the predict function to generate predictions on your test data. Then, compare these predictions with the class labels in the test data itself.

**Step 6: Build and evaluate a random forest.**

* Evaluate this model on the test data the same way you did for the tree. Random forests can automatically generate an estimate of variable importance during training by permuting the values in a given variable and measuring the effect on classification. If scrambling the values has little effect on the model's ability to make predictions, then the variable must not be very important.

**Step 7: Train a gradient boosting tree and compare results.**

**Step 8: Construct confusion matrices**

* Use the table function to generate a confusion matrix for each of your three methods. Generate predictions using the predict function.