Engn 395 Project 01: *Smart as a Rock*

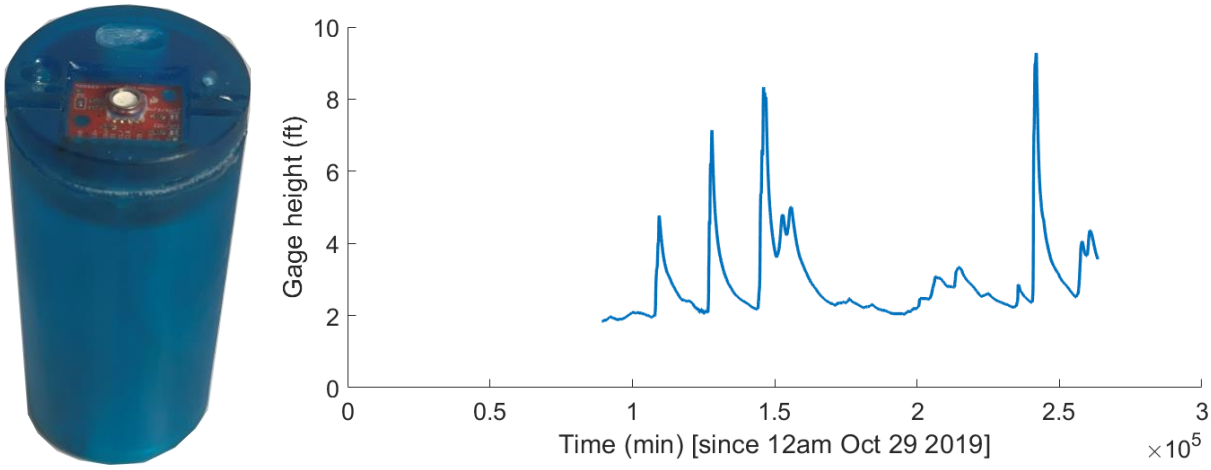

Figure 1. Left: SmartRock capsule (courtesy of Laws Smith and Stevan Kriss: https://github.com/skriss28/SmartRock). Right: Cowpasture River gage height vs. time. Data obtained from: https://waterdata.usgs.gov/nwis/uv?site_no=02016000

Background/Introductcion:

Rock plucking is a fundamental geological event. We currently know very little about the fluid dynamics driving this process. In order to gain a deeper understanding of rock plucking, the SmartRock capsule will be inserted into a rock in the Cowpasture River, near Clifton Forge, VA USA. The SmartRock capsule is equipped with 2 pressure sensors (top and bottom) and a 9 axis inertial measurement unit. In order to maximize battery life and thus the duration of the experiment, the SmartRock capsule must remain in a low-power state waking up periodically (say every 1 min) to measure the pressure at the top sensor, which is a proxy for the river depth. When the river starts to rise fast enough during a rain event, SmartRock must "wake up" and start sampling pressure and acceleration data at a much higher sampling rate (f_s = 40 Hz). This increases the probability of capturing data during a rock plucking event. Of course, the higher sampling rate drains the battery at a faster rate, so it should go back to sleep (low power mode) when the rain event is over.

Problem Statement:

Design and implement a signal processing technique (aka algorithm) that uses pressure readings from a single sensor to detect when the river is rising "fast enough" to the SmartRock capsule to transition from the low to high power states. Historical data for river gage height vs. time for the past 6 months is illustrated in Figure 1. Remember, in the real world scenario, you only get current and retrospective data at the sensors—not the entire data set into the future!

To help get you started:

1. In view of the data for river height vs. time provided in Figure 1, first write in "simple, but no simpler" terms possible what your plan is to detect river rising events. Be mathematically precise in your terminology.

2. Write matlab code to implement your signal processing technique. Helpful functions may include: plot, diff, mean, std, load. Consult the Mathworks website documentation as needed: https://www.mathworks.com/help/matlab/

3. Accessing and using data in Matlab:

   Data files are available in this box drive folder:
   https://wlu.box.com/s/1slscz3grmnapeu9896f13xoom8mvbl7

   The data is provided in a convenient to use matlab format in: *Cowpasture0201600_Data.mat*.

   You can load the data as follows. The variables are:

   - discharge: discharge rate of the river vs time in cubic feet per second (cfs)
   - height: height of the river vs. time (ft)
   - t: time vector at which samples were taken. The time t= 0 refers to midnight Oct 29, 2019.

```
>> y =load('Cowpasture0201600_Data.mat');

y =

  struct with fields:

    discharge: [17570×1 double]
       height: [17570×1 double]
            t: [1×17570 double]
>> figure; plot(y.t, y.height); xlabel('Time (s)'); ylabel('Gage height
(ft)')
```

Beware of NaN values (Not-A-Number) in the data. These torpedo many math operations, such as computing the mean, standard deviation, etc., because just a single NaN will ruin say 20,000 other valid values. To avoid the NaNs you can use the matlab function isfinite(), like so:

```
>> validIndex = isfinite(y.height);  %returns integer indices of all
samples that are finite, ie not NaN

>> meanHeight = mean(y.height(validIndex)) %compute mean on just the
subset of valid data
```

4. Visualize the results of your signal processing technique tagging SmartRock "turn on" and "turn off". This means you should plot the 'trigger on' times on top of the height vs. t plot. Very carefully review the results by eye. Use the zoom functions in matlab as necessary.

5. Evaluate the successes (and failures) of your first algorithm. In what specific instances did it correctly tag a time to trigger on (true positive = TP)? In what specific instances did you observe the algorithm turn on smartrock to high power mode when it probably shouldn't have (false positive = FP), or when it missed an actual river rising event (false negative = FN)?

6. Now revise your methodology to optimize (minimize FPs and FNs, and maximize TPs).

7. Are your results generalizable—i.e., will your signal processing technique work well for river height vs. time data that isn't exactly like the data set you worked with.

8. Report your findings with sufficient illustrative figures and explanatory text. 4 pages max! Also, submit your matlab code .m file and copy and paste text thereof into an appendix.