1 Data available:

1. Accelerometer data: measures all forces (include gravity) in three dimensions. Note that this is auto-processed into orientation data. I don't think this will be useful because hand motions will dominate.

2. Gyroscope data: measures rate of rotation in all three physical axes. Highly worth looking into.

3. Barometer: one-dimensional, determines device deformation, may determine radial location of input

2 Possible models

1. Use pressure sensor to determine when an input is pressed. (I think using the height of the pressure reading to determine radial distance will not be feasible; low SNR and too much variance in signal. No two taps are identically the same.)

2. Given above model to determine when an input is pressed, we can detect the peak in the gyroscope at the time. Can I assume that the closest gyro point is near the top of the peak? Can I then detect the peak by catching the zeros on the left/right side? Or maybe instead of using zero as a threshold for left/right peak detection, I can isolate the peak by using —p— ¿ 0.2 or something.

   I think model 1 is going to be most challenging since it will require peak detection in fairly sparse data. Visually, touch inputs are succeeded by an increase/decrease in pressure. This makes sense, because the touch input is registered initially, when the finger makes first contact with the screen; the device pressure spike would occur subsequently, after first contact when the finger applies pressure.

model 2 sounds doable with basic numerical analysis.

3 Procedure

1. Create application where accelerometer/barometer data can be recorded and touch events can be logged
   (a) start/stop for logging accelerometer/barometer.
      i. Make a button with text that changes when pressed.
      ii. Figure out how to log data in external files.
      iii. Log it.
      iv.
   (b) within the start/stop, record touches
      i. Figure out how to record touch information. Do I need some sort of canvas?
2. Log some things, upload to computer.
   (a) Start by placing a dot at a random location that moves when a tap is detected.
3. From the sample datastream, figure out how to isolate peaks.
4. Assign a normed score for each peak.
   (a) Perhaps the normed score could be the height, or the ratio of peak height to width (width determined by half height). Maybe weight this ratio, let it be a parameter.
5. Create a heatmap of peak norms and pixels, and determine viability of project. Is there a working polar-coordinate proxy? Or will we have to do more abstract machine learning stuff?

6. Once a peak detection algorithm has been identified, implement it on the device so that it logs only the peaks instead of all noise (for data reduction)
7. Create application where digits can be inputted and recorded.
8. Include accelerometer/barometer write-to-disk.
9. Figure out peak detection; need to log tuples
    (a) Accelerometer curve
    (b) Barometer curve
    (c)
10. Dump it all into SQL on desktop
11. Do the machine learning
    (a) Start by creating a heatmap of
12. With the model, attempt to identify digits from accelerometer/barometer
    (a) Create application with grid
    (b) User inputs data for SOME PERIOD OF TIME
    (c)