

# Midterm Exam

Zach Stecher

Due: 10/13/16

**Problem 1: Using the given data set and generator, try these three approaches and note observations.**

**(a)The Pocket algorithm, starting from  $w = 0$ .**

I tried running the Pocket algorithm multiple times at different amounts of iterations to see how it would perform. While the algorithm took a heavy computational toll, it seemed to perform well classifying a blob of data points that lacked a clearly defined linear solution(or any linear solution, really). At 1,000 iterations, the algorithm converged anywhere between 12-14%. Interestingly, when I increased the iterations to 3,000, the program still converged at 12%, although more consistently than before where it would vary by as much as 2%.

**(b)Linear Regression (applied as a classification method).**

Linear Regression took almost no computational resources compared to the Pocket algorithm. The trade-off, however, was that  $E_{out}$  for Linear Regression was much, much higher. Just by looking at the plot, it's clear that while it excelled at finding an acceptable slope for linear classification, the actual placement of the line was way off.

**(c)The Pocket algorithm, starting from the solution given by Linear Regression.**

By initializing the weights used in the Pocket algorithm to the solution found by Linear Regression, the program ended up converging on an answer with less updates to the pocket itself. Interestingly, the program still converged at 12%  $E_{out}$ , leading me to believe that there are a significant number of outliers in the data we were given. Doing this did seem to lessen the amount of computational power required by the Pocket algorithm to converge, though not significantly.

**Problem 2: Write a Python program that solves Problem 2.12 in an iterative manner.**

```
import math

N = 1000
n = 0
it = 0

#While the difference is more than 1, we haven't yet
converged.
while((N - n) > 1):
    it += 1
    n = N
    N = (8/(0.05**2))*math.log((4*((2*N)**10 + 1)/0.05))

#Just in case something goes wrong, to prevent an
infinite loop.
    if(it == 20):
        break

print N
```