1 Warm up: Linear Classifiers and Boolean Functions

1. It is linearly separable. The linear threshold unit could be:

$$y = \begin{cases} 1 & if \quad x_1 + x_3 - x_2 \geq 0 \\ 0 & otherwise \end{cases} \tag{1}$$

2. It is linearly separable. The linear threshold unit could be:

$$y = \begin{cases} 1 & if \quad x_1 + x_3 + 2x_2 \geq 2 \\ 0 & otherwise \end{cases} \tag{2}$$

3. It is not linearly separable.

4. It is not linearly separable.

5. It is linearly separable. The linear threshold unit could be:

$$y = \begin{cases} 1 & if \quad -x_1 + x_2 - x_3 \geq 1 \\ 0 & otherwise \end{cases} \tag{3}$$

2 Mistake Bound Model of Learning

1. (a) There are only 80 possible values for l. And each l corresponds to a unique function in the concept class. So we can se that

$$|C| = 80 \tag{4}$$

   (b) Define a function

$$g(x_1, x_2) = f_l(x_1, x_2) - y^t \tag{5}$$

   if $g()$ equals zero, no mistake is made here. Otherwise, it makes a mistake.

   (c) i. If $g(x_1, x_2)$ is greater than zero, showing that function f think it is positive but the true label isn't. In this case, the range of f is bigger than we want. So remove all the functions with length no smaller than l.

   ii. If $g(x_1, x_2)$ is smaller than zero, showing that function f thinks that it is negative but the true label is positive. In this case, the range of l is smaller than what we want. So remove all the functions whose length is no bigger than l in the concept class.

Algorithm 1 Mistake-driven Learning Algorithm

(d)

```
Initialize C_0 = C.

for sample (x_1^i, x_2^i) do
    Find function f_k whose length k is the middle number among the length of all the functions in current concept class C_i.
    Check if the function f_k made a mistake.

    if f_k doesn't make a mistake then
        Assign C_{i+1} = C_i
        Continue to the next sample
    else
        Calculate g = f_k(x_1^i, x_2^i) - y^i
        if g > 0 then
            Remove all the functions in C_i with length ≥ k.
        else
            Remove all the functions in C_i with length ≤ k.
        end if
    end if
    if |C_i| = 1 then
        Break the loop.
    end if
    Assign C_{i+1} = C_i
end for
return the only remaining function.
```

2. *Proof.* For the Halving algorithm, the algorithm will stop when only M functions in the concept class. Because the remaining M functions are all perfect experts since they will never be removed from the concept class. Suppose the algorithm made k mistakes before it stops. Then we have:

$$M \times 2^k = N \tag{6}$$

So we know that $k = \log \frac{N}{M}$. It means that the mistakes the algorithm will make is no bigger than $\log \frac{N}{M}$. That is, the mistake bound of it is $O(\log \frac{N}{M}$. $\square$

3 The perceptron Algorithm and its Variants