1

ECE 657 Assignment 1

Hammad Rehman
ID: 20293340

1) - Let there be ~~2 vectors~~ a set of vectors w_i, where $i \in [0, N]$, and a vector y. w_i represents the set of inputs to the perceptron, and y represents the set of correctly assigned weights.

Assume:

$$\alpha \leftarrow w_i \cdot y > \theta \qquad (1)$$

where θ is the threshold above which the output is correct.

Let there be an infinite sequence of inputs such that every w_i occurs an infinite number of times. Now assume a set of vectors $V_0 \dots V_n$, where V_0 is arbitrary and V_n:

Total error correction $\leftarrow$

$$V_n = \begin{cases} V_{n-1} \rightarrow \alpha \geq \theta \\ V_{n-1} + w_{i_n} \rightarrow \alpha \leq \theta \end{cases} \qquad (2)$$

The theorem says that for some large index m, $V_m = V_{m+1} \dots\dots$, i.e. no more error correction will be required.

So we remove all situations where $V_n = V_{n-1}$:

$$(3) \qquad V_n = V_{n-1} + w_{i_n} \; ; \quad w_{i_n} \cdot V_{n-1} \leq \theta \Rightarrow \text{error correction required}$$

However, it is not possible for (1) and (3) to be true indefinitely. So to ~~prove a co~~ provide a contradiction, ~~start to~~ first we observe that eq. (1) implies:

→ for some large n

$$\|V_n\|^2 > C n^2 \qquad (4)$$

Since $V_n = \sum_{i=0}^{n-1} V_i$, $V_n \cdot y > (V_0 \cdot y + n\theta)$. So, using the Cauchy Schwarz inequality:

$$\|V_n\|^2 \ge \frac{(V_n \cdot y)^2}{\|y\|^2} > \frac{[(V_0 \cdot y) + n\theta]^2}{\|y\|^2}$$

$$\frac{\left[\left(\frac{V_0 \cdot y}{\theta} + n\right)\theta\right]^2}{\|y\|^2} \to \frac{\theta^2}{\|y\|^2}\left[n + \frac{V_0 \cdot y}{\theta}\right]^2$$

If $V_0 \cdot y \ge 0$, $C = \theta^2/\|y\|^2$ → for all n, otherwise $C = \left(\frac{1}{4}\right)\left(\frac{\theta^2}{\|y\|^2}\right)$, for $n > -2\left(\frac{V_0 \cdot y}{\theta}\right)$. So now we've set a minimum bound on V_n.

(3)

From (3), we extrapolate:

$$||V_k||^2 = ||V_{k-1}||^2 + 2w_i \cdot V_{k-1} + ||w_i||^2 \quad \text{for each } k$$

$$||V_k||^2 - ||V_{k-1}||^2 = \underbrace{2w_i \cdot V_{k-1}}_{\leq 0} + ||w_i||^2 \Rightarrow (4a)$$

$$\hookrightarrow 2w_i \cdot V_{k-1} + ||w_i||^2 \leq 2\theta + M$$

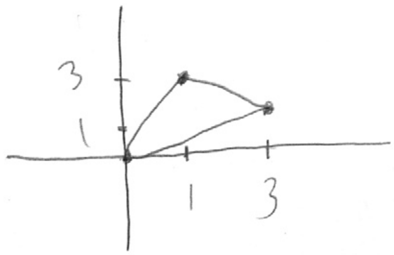
M is implicitly $\max_{i=1 \ldots N} ||w_i||^2$

If you sum eq. 4a for all $k = 1 \rightarrow n$, you get

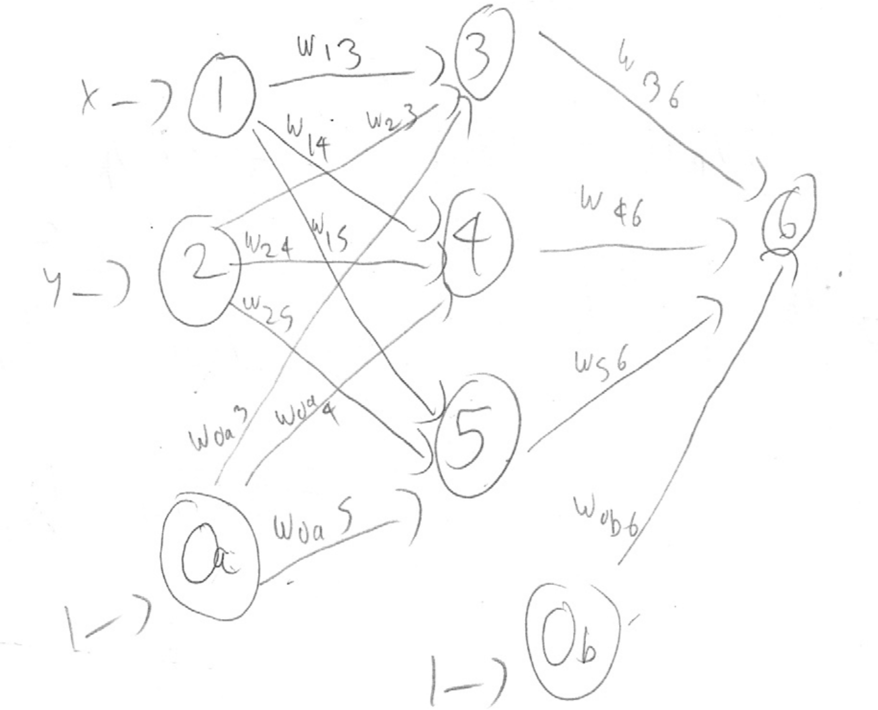
$$||V_n||^2 \leq ||v_0||^2 + (2\theta + M)n$$

and you establish an upper bound on V_n. Thus for n sufficiently large, the perceptrons converge.

2)



Redrawn network



- Assume for point to be inside the Δ, P3, P4, P5 must all output +1 (assuming sgn activation function). So positive case: $P6 \geq 3$, negative $P6 < 3$. With this in mind, I produced the following model using an Excel speadsheet, first by calculation the line equations:

(1) $y - 3x = 0$	(2) $3y - x = 0$	(3) $y + x - 4 = 0$
$\Downarrow$	$\Downarrow$	$\Downarrow$
$w_{13} = -3$	$w_{14} = -1$	$w_{15} = 1$
$w_{23} = 1$	$w_{24} = 3$	$w_{25} = 1$
$w_{0a_3} = 0$	$w_{0a_4} = 0$	$w_{0a_5} = -4$
activation$(O_3) = -$SIGN	activation$(O_4) =$ SIGN	activation$(O_5) = -$ SIGN

⑤

Above, the activation are + SIGN if the point needs to be above the line and −SIGN if below; & then set the following weights:

$$w_{36} = w_{46} = w_{56} = 1$$

$$w_{0_B 6} = 0$$

$$\text{activation } (O_6) = f(x) = \begin{cases} 1 & \text{if } x \geq 3; \\ 0 & \text{otherwise} \end{cases} \Rightarrow \text{Call it } G$$

Note that this classifies on the triangle boundary as not inside the triangle. Example:

Point $(1,2)$: $\rightarrow$ inside

$$O_3 = 1(-3) + 2(1) + 1(0) = -1$$

$$O_4 = 1(-1) + 2(3) + 1(0) = 5$$

$$O_5 = 1(1) + 2(1) + 1(-4) = -1$$

$$O_6 = G\left[\left[-SIGN(-1) + SIGN(5) + -SIGN(-1) + 0\right]\right] = 1 \checkmark$$

$(3,0)$: $\rightarrow$ out

$$O_3 = 3(-3) + 0(1) + 1(0) = -9$$

$$O_4 = 3(-1) + 0(3) + 1(0) = -3$$

$$O_5 = 3(1) + 0(1) + 1(-4) = -1$$

$$O_6 = G\{[-SIGN(-9) + SIGN(-3) + -SIGN(-1)]\} = 0 \checkmark$$

6

3) $\Delta w^{(k+1)} = \eta\left(\underbrace{t^{(k+1)}}_{=t^{(k)} \text{ since inputs are the same}} - w^{(k+1)} x^{(k)}\right) \dfrac{x^{(k)}}{\|x^{(k)}\|^2}$

Note $w^{(k+1)} = \Delta w^{(k)} + w^{(k)}$

$$= \eta\left(t^{(k)} - \left(\Delta w^{(k)} + w^{(k)}\right) x^{(k)}\right) \frac{x^{(k)}}{\|x^{(k)}\|^2}$$

$$= \eta\left[\frac{t^{(k)} x^{(k)}}{\|x^{(k)}\|^2} - \Delta w^{(k)} - w^{(k)}\right]$$

$$= \frac{\eta t^{(k)} x^{(k)}}{\|x^{(k)}\|^2} - \eta \Delta w^{(k)} - \frac{\eta w^{(k)} \|x^{(k)}\|}{\|x^{(k)}\|^2}$$

$$= \boxed{\frac{\eta\left(t^{(k)} - w^{(k)} x^{(k)}\right) x^{(k)}}{\|x^{(k)}\|^2}} - \eta \Delta w^{k}$$

$$= \Delta w^{(k)} - \eta \Delta w^{(k)}$$

$$= (1-\eta) \Delta w^{(k)}$$

4) I settled on 45 neurons in the hidden layers, and softmax activation functions for both hidden + output layers, from ~~p~~ online research, I used a 75%/25% training/testing split. When training (≈250 – 300 epochs), I converged on an ~~loss of~~ accuracy of ≈ 80%, and with the test set it was 81%.

I tried a larger number of neurons (≈400), but above 70 neurons my losses increased massively.