CISC452/CMPE452/COGS 400
Perceptron Convergence Theorem

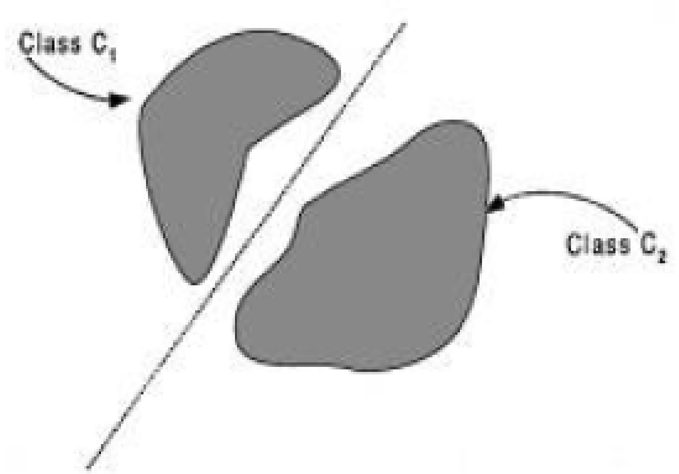
Farhana Zulkernine

Perceptron Convergence Theorem

- The theorem states that for any data set which is linearly separable, the perceptron learning rule is guaranteed to find a solution in a finite number of iterations.
- Idea behind the proof: Find upper & lower bounds on the length of the weight vector to show finite number of iterations.

Perceptron Convergence Theorem

Let's assume that the input variables come from two linearly separable classes C_1 & C_2.



Let T_1 & T_2 be subsets of training vectors which belong to the classes C_1 & C_2 respectively. Then $T1 \cup T2$ is the complete training set.

Perceptron Convergence Theorem

As we have seen, the learning algorithms try to find a weight vector *w* such that

$$w \cdot x > 0 \quad \forall x \in C_1$$

$$w \cdot x \leq 0 \quad \forall x \in C_2$$

(***x*** *is an input vector*)

If the *k*th member of the training set, *x(k)*, is correctly classified by the weight vector *w(k)* computed at the *k*th iteration of the algorithm, then we do not adjust the weight vector.

However, if it is incorrectly classified, we use the modifier

$$w(k+1) = w(k) + \eta d(k) x(k)$$

Perceptron Convergence Theorem

So we get

$$w(k+1) = w(k) - \eta x(k) \quad if \quad w(k)\cdot x(k) > 0,\ x(k) \in C_2$$

$$w(k+1) = w(k) + \eta x(k) \quad if \quad w(k)\cdot x(k) \leq 0,\ x(k) \in C_1$$

We can set $\eta = 1$, as for $\eta \neq 1$ *(>0)* just scales the vectors.

We can also set the initial condition *w(0) = 0*, as any non-zero value will still converge, just decrease or increase the number of iterations.

Perceptron Convergence Theorem

Suppose that $w(k) \cdot x(k) < 0$ *for* $k = 1, 2, \ldots$ where $x(k) \in T_1$, so with an incorrect classification we get

$$w(k+1) = w(k) + x(k) \qquad x(k) \in C_1$$

By expanding iteratively, we get

$$\begin{aligned} w(k+1) &= x(k) + w(k) \\ &= x(k) + x(k-1) + w(k-1) \\ &\;\vdots \\ &= x(k) + \ldots + x(1) + w(0) \end{aligned}$$

Perceptron Convergence Theorem

As we assume linear separability, ∃ a solution w^* where $w \cdot x(k) > 0$, $x(1)...x(k) \in T_1$. Multiply both sides by the solution w^* to get

$$w^* \cdot w(k+1) = w^* \cdot x(1) + ... + w^* \cdot x(k)$$

These are all > 0, hence all >= α, where

$$\alpha = \min \ w^* \cdot x(k)$$

Thus we get

$$w^* \cdot w(k+1) \geq k\alpha$$

Perceptron Convergence Theorem

Now we make use of the Cauchy-Schwarz inequality which states that for any two vectors A, B

$$\|A\|^2 \|B\|^2 = (A \cdot B)^2$$

Applying this we get

$$\|w^*\|^2 \|w(k+1)\|^2 \geq (w^* \cdot w(k+1))^2$$

From the previous slide we know $w^* \cdot w(k+1) \geq k\alpha$

Thus, it follow that

$$\|w(k+1)\|^2 \geq \frac{k^2 \alpha^2}{\|w^*\|^2}$$

Perceptron Convergence Theorem

We continue the proof by going down another route.

$$w(j+1) = w(j) + x(j) \quad for\ j = 1, \ldots, k \quad with\ x(j) \in T_1$$

We square the Euclidean norm on both sides

$$\|w(j+1)\|^2 = \|w(j) + x(j)\|^2$$
$$= \|w(j)\|^2 + \|x(j)\|^2 + 2\mathrm{w}(j) \cdot x(j)$$

incorrectly classified, so < 0

Thus we get

$$\|w(j+1)\|^2 - \|w(j)\|^2 \leq \|x(j)\|^2$$

Perceptron Convergence Theorem

Summing both sides for all *j*

$$\|w(j+1)\|^2 - \|w(j)\|^2 \le \|x(j)\|^2$$

$$\|w(j)\|^2 - \|w(j-1)\|^2 \le \|x(j-1)\|^2$$

$$\vdots$$

$$\|w(1)\|^2 - \|w(0)\|^2 \le \|x(1)\|^2$$

We get

$$\|w(k+1)\|^2 \le \sum_{j=1}^{k} \|x(j)\|^2$$

$$\le k\beta \qquad \beta = max \ \|x(j)\|^2$$

Perceptron Convergence Theorem

But now we have a conflict between the equations, for sufficiently large values of *k*

$$\|w(k+1)\|^2 \leq k\beta \qquad \|w(k+1)\|^2 \geq \frac{k^2\alpha^2}{\|w^*\|^2}$$

So, we can state that *k* cannot be larger than some value k_{max} for which the two equations are both satisfied.

$$k_{max}\beta = \frac{k_{max}^2\alpha^2}{\|w^*\|^2} \quad \Rightarrow \quad k_{max} = \frac{\beta\|w^*\|^2}{\alpha^2}$$

Perceptron Convergence Theorem

Thus it is proved that for $\eta_k = 1$, $\forall\, k$, $w(0) = 0$, given that a solution vector w^* exists, the perceptron learning rule will terminate after at most k_{max} iterations.