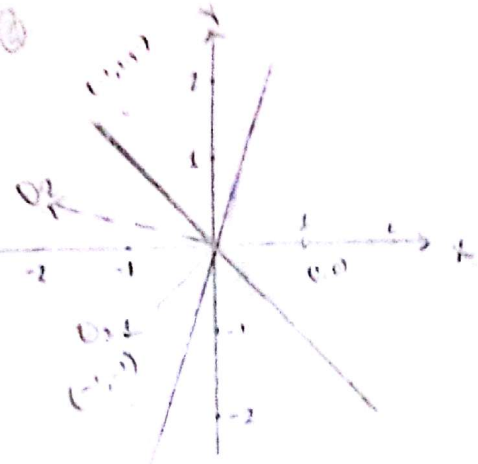
PERCEPTRON MISTAKES.



Makes 2 mistakes - including the first one.

$\theta_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

1. $y^{(1)}\theta_0 x^{(1)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}\begin{bmatrix} -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \le 0.$

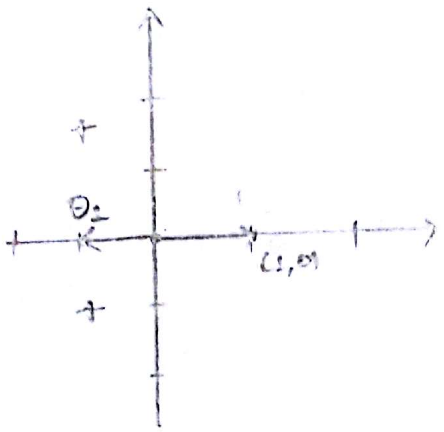
$\Rightarrow \theta_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -1 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$

2. $y_2 \theta_1 x^{(2)} = (-1)\begin{bmatrix} -1 \\ -1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = +1 > 0$

$\Rightarrow \theta_2 = \theta_1$

3. $y_3 \theta_2 x^{(3)} = +1\begin{bmatrix} -1 \\ -1 \end{bmatrix} \cdot \begin{bmatrix} -1 \\ 1.5 \end{bmatrix} = 1 - 1.5 \le 0.$

$\Rightarrow \theta_3 = \theta_2 + x^{(3)} = \begin{bmatrix} -1 & -1 \\ -1 & +1.5 \end{bmatrix} = \begin{bmatrix} -2 \\ 0.5 \end{bmatrix}$



$\theta_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

1. $y^{(2)}\theta_0 x_2 = (-1)\begin{bmatrix} 0 \\ 0 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = 0$

$\Rightarrow \theta_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$

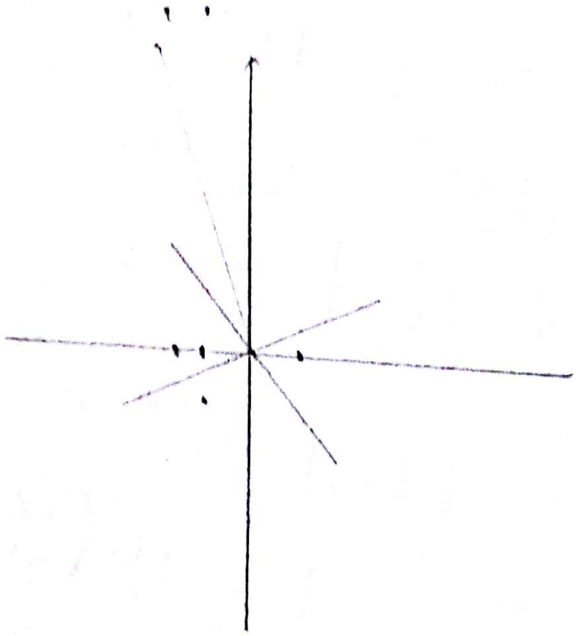
2. $y^{(1)}\theta_1 x_1 = +1 \cdot \begin{bmatrix} -1 \\ 0 \end{bmatrix}\begin{bmatrix} -1 \\ -1 \end{bmatrix} = 1 > 0.$

3. $y^{(3)}\theta_1 \cdot x_3 = +1\begin{bmatrix} -1 \\ 1.5 \end{bmatrix}\begin{bmatrix} -1 \\ 0 \end{bmatrix} = 1 > 0.$

Makes 1 mistake; the very first one.

Classifier = x-axis.

$\theta = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$



$\theta_0 = \begin{bmatrix}0\\0\end{bmatrix}$;

For x_1 : $y^{(1)}\theta_0 x^{(1)} = +1\cdot\begin{bmatrix}0\\0\end{bmatrix}\begin{bmatrix}-1\\-1\end{bmatrix} = 0 \le 0$

$\Rightarrow \theta_1 = \begin{bmatrix}0\\0\end{bmatrix} + \begin{bmatrix}-1\\-1\end{bmatrix} = \begin{bmatrix}-1\\-1\end{bmatrix}$

For x_2 : $y^{(2)}\theta_1 x^2 = -1\begin{bmatrix}-1\\-1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = -1(-1) > 0$

For x_3 : $y^{(3)}\theta_1 x^3 = +1\begin{bmatrix}-1\\-1\end{bmatrix}\begin{bmatrix}-1\\10\end{bmatrix} = 1-10 < 0.$

$\Rightarrow \theta_3 = \begin{bmatrix}-1\\-1\end{bmatrix} + \begin{bmatrix}-1\\10\end{bmatrix} = \begin{bmatrix}-2\\10\end{bmatrix}$

For x_1 $y^{(1)}\theta_3 x_1 = +1\begin{bmatrix}-2\\10\end{bmatrix}\begin{bmatrix}-1\\-1\end{bmatrix} = 2-10 < 0.$

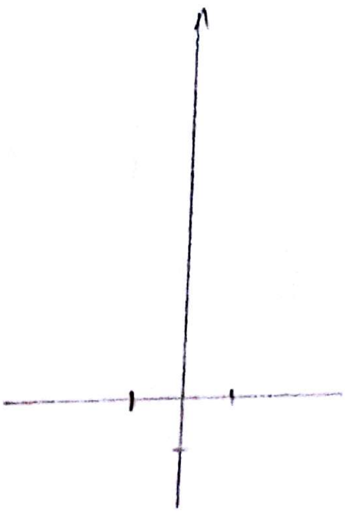
$\Rightarrow \theta_4 = \begin{bmatrix}-2-1\\10-1\end{bmatrix} = \begin{bmatrix}-3\\9\end{bmatrix}$

$\theta_5 = \begin{bmatrix}-4\\8\end{bmatrix}$; $\theta_6 = \begin{bmatrix}-5\\7\end{bmatrix}$; $\theta_7 = \begin{bmatrix}-6\\6\end{bmatrix}$

$\theta_8 = \begin{bmatrix}-7\\5\end{bmatrix}$

$\Rightarrow \boxed{\theta_8 \cdot x_1 y_1 > 0}$

7 Mistakes including the first one.



$\theta_0 = \begin{bmatrix}0\\0\end{bmatrix}$

For x_2 : $y^{(2)}\theta_0 x^{(2)} = 0 \le 0.$

$\Rightarrow \theta_1 = y^2 x^{(2)} + \theta_0 = (-1)\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}-1\\0\end{bmatrix}$

For x_1 : $y^{(1)}\theta_1 x^{(1)} = +1\begin{bmatrix}-1\\0\end{bmatrix}\begin{bmatrix}-1\\-1\end{bmatrix} > 0.$

For x_3 : $y^{(3)}\theta_1 x^{(3)} = +1\begin{bmatrix}-1\\0\end{bmatrix}\begin{bmatrix}-1\\10\end{bmatrix}$

$= +1 > 0.$

Only 1 mistake.

· Decision Boundry

(a). (i) $f(x_1, x_2, x_3) = 1$ when $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$

But, if $\theta_0 = 0$, $\theta \cdot x + \theta_1 \neq 0$.

So, it's not posible to learn such θ.

(ii) Yes, its possible to learn the pair θ & θ_0

(b).

(i). No. since three points lies on boundry.

(ii). Yes.

(iii). Yes. 2 points lie on boundry so still can classify.

(iv). Yes.

(c). (iii) & (iv) are linear.

Scanned by CamScanner

Q1.

(a) $A = \begin{bmatrix} +1/6 & +1/6 & +1/6 & +1/6 & +1/6 & +1/6 \\ +1/3 & +1/3 & +1/3 & -1/3 & -1/3 & -1/3 \end{bmatrix}$.

(b) Yes, we can classify the associated x_i if z_i can be classified. Since x_i has higher dimensions than x_i', new dimensions can simply be ignored by the classifier to classify x_i.

(c). No, we cannot always classify z_i's. Transformation into lower dimensions can reduce the necessary characteristics for classification.

(d) From (b), we know that it's always possible to find a equally good classifier in higher dimension. in training data.

However in lower dimension, we can generalize the data more since we are less likely to over fit the to data than in higher dimension.

(c) The difference in mistakes made by algorithm in (a) & (b) is due to the presence of large outlier (-1, 10) in part (b). Due to large magnitude of outlier, the resulting mistakes, the θ didn't get rotated fast enough to converge and hence resulted in greater number of errors in (b).

(d). As we see in (b), choosing a vector of large magnitude resulted in greater number of mistakes. So, if we start with vector with largest magnitude, we can maximize the # of mistakes as subsequent updates won't change θ significantly.

$x^{(1)} = \{\max \|x_i\|, i = 1 \ldots n\}$

2. <u>PERCEPTRON PERFORMANCE</u>

(a) post-training $\theta = \sum_{\text{mistakes}} y x = +1\begin{bmatrix}-4\\2\end{bmatrix} - 2\begin{bmatrix}-1\\-1\end{bmatrix} - 1\begin{bmatrix}2\\2\end{bmatrix}$

$$= \begin{bmatrix}-4+2-2\\2+2-2\end{bmatrix} = \begin{bmatrix}-4\\2\end{bmatrix}$$

(b). You could start with some random ~~vector~~ with greater length than any of them eg. $\begin{bmatrix}15\\20\end{bmatrix}$

(c) $\dfrac{\theta^{k}\theta^{*}}{|\theta^{*}|} \geq \dfrac{\theta^{k-1}\theta^{*}}{\theta^{*}} + \gamma$

$\Rightarrow \dfrac{\theta^{k}\theta^{*}}{|\theta^{*}|} \geq \dfrac{\theta^{0}\theta^{*}}{|\theta^{*}|} + k\gamma.$

$\geq 0 + k\gamma$

Also,

$|\theta^{k}|^{2} \leq |\theta^{0}|^{2} + kR^{2}$

... the presence of ... resulting mistakes ; the θ

5. MISTAKE BOUNDS

(a)

$d = n = 2 : i$

$x^{(1)} : [\cos \pi, 0] = [-1, 0]$

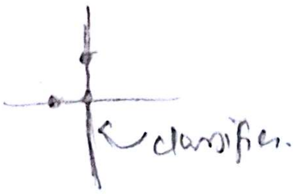
$x^{(2)} : [0, \cos(2\pi)] = [0, 1]$

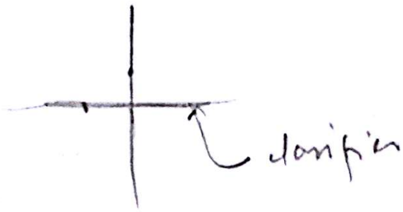
for $x^{(2)}$

$\theta = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$x^{(1)}$ always mistake.

$\theta = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$





In both case classifier can't completely seperate two points as one of them lies in the classifier.

So, After two updates,

$\theta = \begin{bmatrix} -1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$

$\theta = \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 \\ 0 \end{bmatrix}$

$\therefore \begin{bmatrix} -1 \\ 1 \end{bmatrix}$

(b) Because all the terms of vectors are zero except $\theta^{(i)th}$ position it takes 'd' mistakes to converge the vector. Since, taking the dot product always gives zero before n^{th} update, the algorithm only converges after 'n' mistakes.

No. theta does depends on the labeling but doesn't depend on ordering.

$\theta = [y_{(i)} x^{(i)} \ldots]$

(c) The ...

... SUPPORT VECTOR MACHINES.

$$\cancel{\theta^{k+1} = \theta^{k} + y x}$$

$$\Rightarrow \quad |\theta^{k}| \le \sqrt{|\theta^{0}|^{2} + kR^{2}} \le |\theta^{0}| + \sqrt{k}\,R$$

Now,

$$1 \ge \frac{\theta^{k}\,\theta^{*}}{|\theta^{k}|\,|\theta^{*}|} \ge \frac{a + kr}{|\theta^{0}_{0}| + \sqrt{k}\,.\,R}$$

$$\Rightarrow \quad a + k r - |\theta^{0}| \le \sqrt{k}\,.\,R$$

$$\Rightarrow \quad k \le \frac{\text{Quadratic in } (R)}{r^{2}}$$

(d). No it doesn't mean that resulting θ is same.

6. LINEAR SUPPORT VECTOR MACHINES

(a).

$$f(\theta) = \text{Loss}_h(y\theta \cdot x) + \frac{\lambda}{2} \|\theta\|^2$$

$$\Rightarrow f(\theta) = \begin{cases} 0 & \text{for } 1 - y\theta \cdot x < 0 \\ 1 - y\theta \cdot x & \text{for } 1 - y\theta \cdot x \geq 0 \end{cases} + \frac{\lambda}{2} \|\theta\|^2$$

$$\Rightarrow \nabla f(\theta) = -yx + 2 \cdot \frac{\lambda}{2} \cdot \theta$$

For $\nabla f(\theta) = 0$, $\boxed{\theta = \frac{1}{\lambda} \cdot y x}$

(b).

(c).

No, it is not possible that for some value of λ the training example (x, y) remains miss-classified by $\hat{\theta}(\lambda)$. since only the margin boundries is pushed outwoard not the classifier itself.

(d). Since, the margin boundries may cross some points, it is entirely possible that resulting marging extend past training point.