Calculate Perceptron Update

Exercise 1

Consider the following two-point 2D data set:

- Positive class (+1): $(1,3)$
- Negative class (-1): $(-1,4)$

Starting with $\mathbf{w}_0=(0,0)$, how many updates will you have to perform to $\mathbf{w}$ until convergence? Write down the sequence of each updated $\mathbf{w}_t$ ($[\mathbf{w}_1,\mathbf{w}_2,\dots,\mathbf{w}_n]$) by iterating the data points in the order:$[(1,3),(-1,4),(1,3),(-1,4),\dots]$.

When you think you know the answers, click the images to reveal the solutions

Click to reveal the answer

Answer: The perceptron algorithm stops when for all points $(\mathbf{x}_i,y_i)$ in the dataset $D, y_i\left(\mathbf{w}^\top\mathbf{x}_i\right)>0$.

Iteration t	$\mathbf{w}_{t-1}$	Value of $y_i\left(\mathbf{w}_{t-1}^\top\mathbf{x}_i\right)$	Misclassified?	If yes, update $\mathbf{w}_t\leftarrow\mathbf{w}_{t-1}+y_i\mathbf{x}_i$
1	$(0,0)$	$1\cdot\left((0,0)^\top(1,3)\right)=0\leq 0$	Yes	$\mathbf{w}_1\leftarrow(0,0)+(1,3)=(1,3)$
2	$(1,3)$	$-1\cdot\left((1,3)^\top(-1,4)\right)=-11\leq 0$	Yes	$\mathbf{w}_2\leftarrow(1,3)-(-1,4)=(2,-1)$
3	$(2,-1)$	$1\cdot\left((2,-1)^\top(1,3)\right)=-1\leq 0$	Yes	$\mathbf{w}_3\leftarrow(2,-1)+(1,3)=(3,2)$
4	$(3,2)$	$-1\cdot\left((3,2)^\top(-1,4)\right)=-5\leq 0$	Yes	$\mathbf{w}_4\leftarrow(3,2)-(-1,4)=(4,-2)$
5	$(4,-2)$	$1\cdot\left((4,-2)^\top(1,3)\right)=-2\leq 0$	Yes	$\mathbf{w}_5\leftarrow(4,-2)+(1,3)=(5,1)$
6	$(5,1)$	$-1\cdot\left((5,1)^\top(-1,4)\right)=1>0$	No	
7	$(5,1)$	$1\cdot\left((5,1)^\top(1,3)\right)=8>0$	No	

Therefore, there were 5 updates to $\mathbf{w}$: $[(0,0),(1,3),(2,-1),(3,2),(4,-2),(5,1)]$

Exercise 2

Your friend Cüneyt comes to you, desperate for your perceptron expertise. His data set is massive, with more than 10 trillion training examples. After hours of training his perceptron until convergence, his code malfunctioned and did not save the final weight vector.

Thankfully, at every training iteration, the code saved which example was used for the update step. Surprisingly, only five of the more than 10 trillion training examples were ever misclassified. They are listed below, along with the number of times they were used in an update step.

Training Example	Times Used in an Update Step
$(0,0,0,0,4),+1$	2
$(0,0,6,5,0),+1$	1
$(3,0,0,0,0),-1$	1
$(0,9,3,6,0),-1$	1
$(0,1,0,2,5),-1$	1

What is the final weight vector of this perception (i.e. the weight vector that would have been saved if the code had not malfunctioned)?

Click to reveal the answer

Answer:

Recall that the perceptron algorithm only updates $\mathbf{w}$ when it misclassifies a point. Since each update is $\mathbf{w}\leftarrow\mathbf{w}+y_i\mathbf{x}_i$, the final vector $\mathbf{w}$ is equal to $\sum_{(\mathbf{x}_i,y_i)\in M}y_i\mathbf{x}_i$, where M is the list of points that were misclassified (repetition allowed) in the training process. Hence, the final weight vector is:

$$\mathbf{w}=(0,0,0,0,4)\cdot 2+(0,0,6,5,0)-(3,0,0,0,0)-(0,9,3,6,0)-(0,1,0,2,5)=(-3,-10,3,-3,3)$$

‹ Previous

Next ›