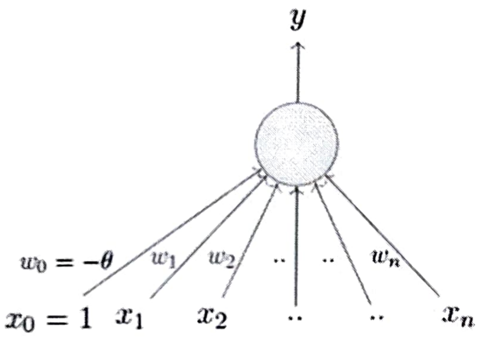
Perceptron Learning Rule

We are going to use a perceptron to estimate if I will be watching a movie based on historical data with the above-mentioned inputs. The data has positive and negative examples, positive being the movies I watched i.e., 1.



$$x_1 = isActorDamon$$
$$x_2 = isGenreThriller$$
$$x_3 = isDirectorNolan$$
$$x_4 = imdbRating(scaled\ to\ 0\ to\ 1)$$
$$\ldots \qquad \ldots$$
$$x_n = criticsRating(scaled\ to\ 0\ to\ 1)$$

Perceptron Learning Algorithm:

Our goal is to find the **w** vector that can perfectly classify positive inputs and negative inputs in our data.

Initialize w with some random vector.

We then iterate over all the examples in the data, (P U N) both positive and negative examples.

Now if an input x belongs to P, ideally what should the dot product w.x be? I'd say greater than or equal to 0 because that's the only thing what our perceptron wants at the end of the day so let's give it that. And if x belongs to N, the dot product must be less than 0.

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$
Initialize **w** randomly;
while !*convergence* **do**
    Pick random $\mathbf{x} \in P \cup N$;
    if $\mathbf{x} \in P$ *and* $\mathbf{w}.\mathbf{x} < 0$ **then**
        $\mathbf{w} = \mathbf{w} + \mathbf{x}$;
    end
    if $\mathbf{x} \in N$ *and* $\mathbf{w}.\mathbf{x} \geq 0$ **then**
        $\mathbf{w} = \mathbf{w} - \mathbf{x}$;
    end
end
//the algorithm converges when all the inputs are classified correctly

Case 1: When **x** belongs to *P* and its dot product $\mathbf{w}.\mathbf{x} < 0$

Case 2: When **x** belongs to *N* and its dot product $\mathbf{w}.\mathbf{x} \geq 0$

Only for these cases, we are updating our randomly initialized **w**. Otherwise, we don't touch **w** at all because Case 1 and Case 2 are violating the very rule of a perceptron.

When x belongs to P, we want w.x > 0, basic perceptron rule. What we also mean by that is that when x belongs to P, the angle between w and x should be less than 90 degrees.

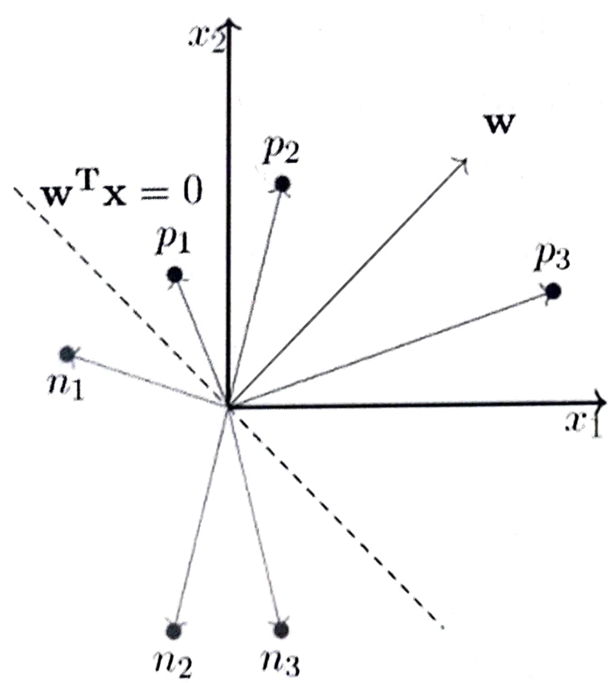
$$\cos\alpha = \frac{\mathbf{w}^T\mathbf{x}}{||\mathbf{w}||\,||\mathbf{x}||} \quad \Bigg| \quad \cos\alpha \propto \mathbf{w}^T\mathbf{x}$$

$$\text{So if } \mathbf{w}^T\mathbf{x} > 0 \;\Rightarrow \cos\alpha > 0 \;\Rightarrow \alpha < 90$$

$$\text{Similarly, if } \mathbf{w}^T\mathbf{x} < 0 \;\Rightarrow \cos\alpha < 0 \;\Rightarrow \alpha > 90$$

So whatever the w vector may be, as long as it makes an angle less than 90 degrees with the positive example data vectors (x E P) and an angle more than 90 degrees with the negative example data vectors (x E N), we are cool.

So ideally, it should look something like this:



So we now strongly believe that the angle between w and x should be less than 90 when x belongs to P class and the angle between them should be more than 90 when x belongs to N class.

Weight Updation :

For positive examples -

$$W_{new} = w + x$$

$$\therefore W_{new} = W_{old} + x$$

$$B_{new} = B_{old} + x$$

For Negative examples -

$$W_{new} = w - x$$

$$W_{new} = W_{old} - x$$

$$B_{new} = B_{old} - x$$