

# Perceptron algorithm

Recall that the perceptron algorithm works in the following way.

You have some data points like the blue and red, and we want to find a perfect line that splits them.

A perfect line has the equation  $Wx+b=0$ , where  $W$  and  $x$  are vectors and  $b$  is a scalar. So it looks more like this,  $w_1x_1+w_2x_2+b=0$ .

As a quick example, let's say the equation is  $3x_1+4x_2+5=0$ .

So we want to 'punish' this line with some error function, and the error function roughly measures how many points are misclassified. It actually measures something closer to the sum of the distances of these misclassified points to the boundary.

We can split this into two sets, the red points and the blue points, and our error function is going to punish those two red points on the left that are on top of the line and those two blue points on the right that are below the line, but it's going to punish them according to their distance to the main line. It's going to be the blue area that is over the line.

The more a red point is into the blue area, the more it gets punished. This means that the point that is close to the line gets punished a little bit, and the point that is far from the line gets punished more. And the other part of the error does the complete opposite. It punishes the blue points that are in the red area. The points that are close to the line don't get punished much, and the ones that are far from the line get punished a lot more. But let's actually put some math here. Let's ask how much the error is at every point.

So we have that our equation  $Wx+b=0$ , which is a linear equation.  $Wx+b$  takes all the values, so in particular, it is one of the lines that is parallel to the main line and a little bit above and then it takes the value 2, 3, 4 successively. And the same thing on the negative side, it takes the values -1, -2, -3, -4, etc. Basically, this is going to be the error. It's going to be the value of  $Wx+b$ . Since we have negative numbers, it's going to be  $|Wx+b|$

### Quiz Question

The error calculated in the Perceptron Algorithm uses the Absolute value of  $(Wx+b)$ . Why is the Absolute value needed?

- a. There are misclassified points both below and above the line
- b. Sometimes the red points or blue points are below the x-axis