

## SVM algorithm

Let's do the same thing we did for the perceptron algorithm, except with the SVM algorithm.

Recall that the equation of the line is  $Wx+b=0$ .

Now we don't just want a single line, we want the line with two *extra* lines that create the margin.

And the equations for these lines are going to be

- $Wx+b=1$
- $Wx+b=-1$

We still want to 'punish' the points that are incorrectly classified. But now, we're going to think of the **region** as points that are incorrectly classified since we don't want anything in between the two lines.

Let's split the error in two. In order to punish the points that are within the margin

- The blue error will now start from the bottom line
- The red error is going to now start from the top line

We look at the values of  $Wx+b$ . As we go up, it's going to be 1, 2, 3, 4, etc. And as we go down, it's going to be -1, -2, -3, etc. In order to build the error, we take the absolute value of those errors and translate it into one.

- The blue error starts at zero on the bottom boundary line and increases by one with each step
- Similarly, the red error is going to be zero on the top boundary line and then increase linearly in the opposite direction.

### Quiz Question

As we saw with the Perceptron Algorithm, the error calculated in the SVM Algorithm also uses the Absolute value of  $(Wx+b)$ . Why is the Absolute value needed?

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

### Quiz Question

Which of the following statements is true about SVM classification errors

- a. The classification error of each miscalculated point is based on the distance that it is from the main boundary line.
- b. Points that are within the margin contribute more to the overall classification error than other misclassified points.

- c. The classification error of each miscalculated point is based on the distance that it is from the extra lines that form the margin around the boundary line.