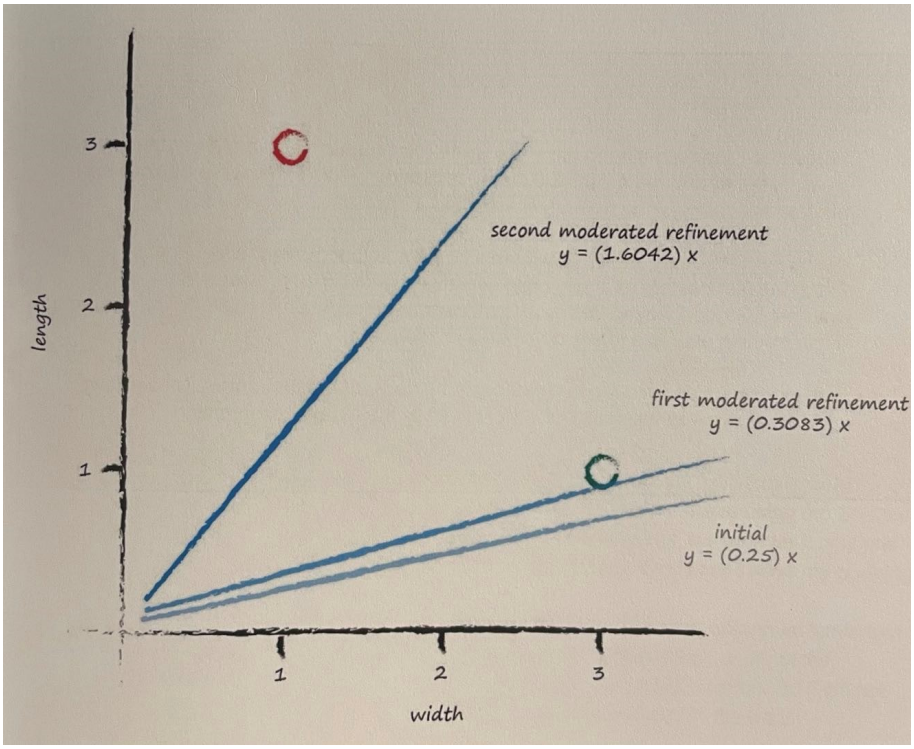
Make Your Own NN p1-p34

2021.1.25



$t = (A + \Delta A)x$	t represents the correct desired value
$E = \Delta A x$	E represent the error, $E = t - y$
$\Delta A = E/x$	we want to know how much to adjust A by to improve the slope of line so it is a better classifier, being informed by the error E

If we keep doing this, updating for each training data example, all we get is that the final update simply matches the last example closely. Therefore, we change it to

$$\Delta A = L * (E/x) \text{ where } L \text{ represents learning rate}$$

Let's run through again, we have an initial $A = 0.25$. The first training example gives us $y = 0.25 * 3.0 = 0.75$, note that the first point is $(3, 1)$ and desired y-value is 1.1. A desired value of 1.1 gives us an erroe of 0.35. The $\Delta A = L(E/x) = 0.5 * (0.35/3.0) = 0.0583$. The update A is $0.25 + 0.0583 = 0.3083$.

Trying out this new A on the training example at $x = 3.0$ gives $y = 0.3083 * 3.0 = 0.9250$. The line now falls on the wrong side of the training example because it is it is below 1.1 but it is not a bad result if you consider it a first refinement step of many to come. It did move in the right direction away from the initial line.

Let's press on to the second training data example at $x = 1.0$. Using A = 0.3083 we have $y = 0.3083 * 1.0 = 0.3083$. The desired value was 2.9, $(1, 3)$ originally, so the error is $2.9 - 0.3083 = 2.5917$. The $\Delta A = L(E/x) = 0.5 * (2.5917/1) = 1.2958$. The even newer A is now $0.3083 + 1.2958 = 1.6042$.

Key Points

⋆ We can use simple maths to understand the relationship between the output error of a linear classifier and the adjustable slop parameter. This is the same as knowing how much to adjust the slop to remove that output error.

⋆ A problem with doing these adjustments naively, is that the model is updated to best match the last training example only, effectively ignoring all previous training examples. A good way to fix this is to moderate the updates with a learning rate so no single training example totally dominates the learning.

⋆ Training examples form the real world can be noisy or contain errors. Moderating updates in this way helpfully limits the impact of these false example.