The story of why there are so many derivatives in this course

Part 1: Why do we need to update the weights?

So let's start our story here. We have a neural network model. We feed it training data and calculate the loss that shows us how wrong the model is. We aim to make this loss as small as possible because **big loss = bad model**, **small loss = good model**. We can change the loss by changing the weights of the model. Input is fixed, and the weights are what change the model's goodness. We want to find the best set of weights. We calculate the loss and if that is bad, we should change the weights. But how do we do that?

For the sake of simplicity and understanding let's think in one dimension and only have one weight w. We want to know how to change the w based on the loss function, let's call it $l(w)$.

So basically we want to have an update rule like this

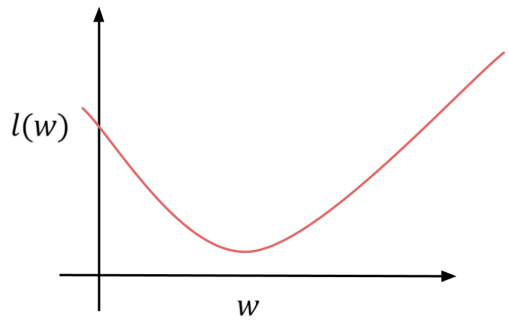
$w = w - ?$

(The $-$ sign is relevant and we will later see why.)

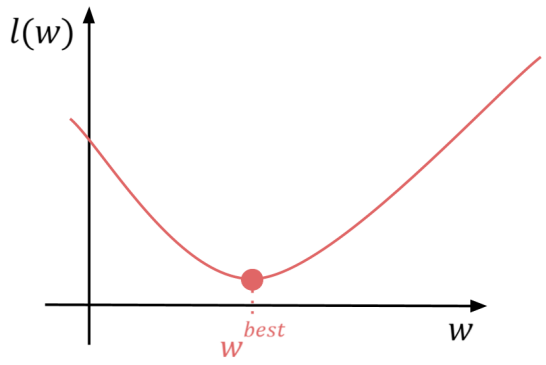
But what to write instead of the question mark?

Part 2: Intuition of how to change the weights?

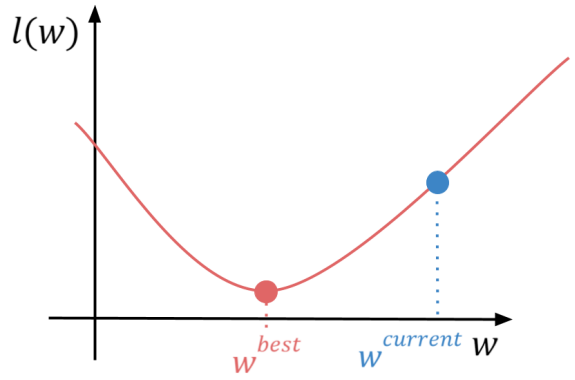
Let's draw the loss function in order to understand what we need. We will make it a random shape for the moment, just to emphasize our ideas, but in principle $l(w)$ is just some function that is dependant on w (e.g. $l(w) = max(0, wx + 2)$).



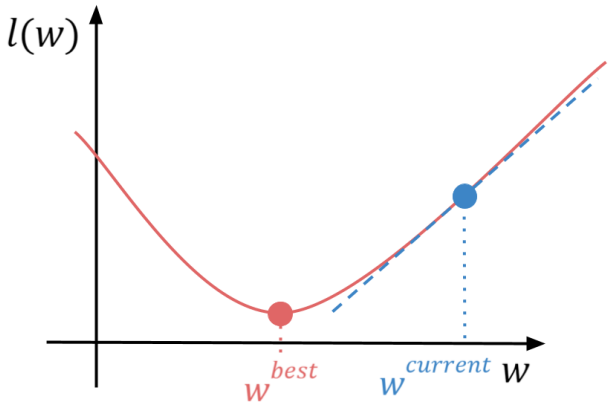
Now let's think about this function for a moment. What does it show? It shows our loss function - it shows for which w values it's higher and for which it is lower. We want the lowest possible loss function, so we have to find the w where the loss is lowest, or in other words, we need to minimize $l(w)$. *(In practice we usually are not able to find the actual minimum point for complex functions, but we still try to find the best result we are able to find.)*



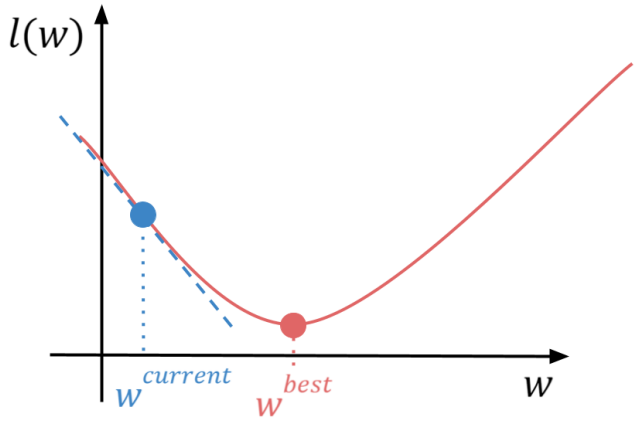
Now that we have set our goal we need to come up with the plan of how to get there. We have a point on the line for our current w and we know we need to somehow get closer to the point where the loss is the smallest.



Let's first use our intuition to figure out what we need. Since $l(w^{current})$ is worse than $l(w^{best})$ we need to change $w^{current}$ to be more like w^{best}. We can clearly see that this means that we need to make $w^{current}$ smaller. Our update rule was selected as $w = w - ?$ (or $w^{current} = w^{current} - ?$ for now). So in order for this rule to make $w^{current}$ smaller we need some positive value for the "?" part. Now let's try to draw a line at the blue point that shows how fast the function is changing at that point. And let's focus on the slope of that line. Slope on the $l(w^{current})$ line is positive (the function is growing at that point). So we could use it to change $w^{current}$ to the correct direction.



So can we use $w = w - slope$ as our update rule? Let's check what happens on the other side. On the other side we need to increase $w^{current}$ to make the loss smaller. Our slope for the line at that point is negative (function is decreasing) so using the update rule $w = w - slope$ we are actually making $w^{current}$ larger as we wanted. Cool! So we can use slopes.



(As you saw we used the minus sign in the update rule because we need to minimize the function.)

The actual value of the slope shows us how big a step towards the minimum we are taking. If the slope is small (line is flatter) then step is smaller and if slope is big (line is more vertical) the change is bigger (function changing more rapidly). Of course the learning rate will be modifying this and in the course you will see other methods to the update rule that change the exact value of this update, but the most important thing that does not change is the sign of the slope (weather the line is going up or down) - based on this we know to either make w smaller or larger to get to a smaller loss value.

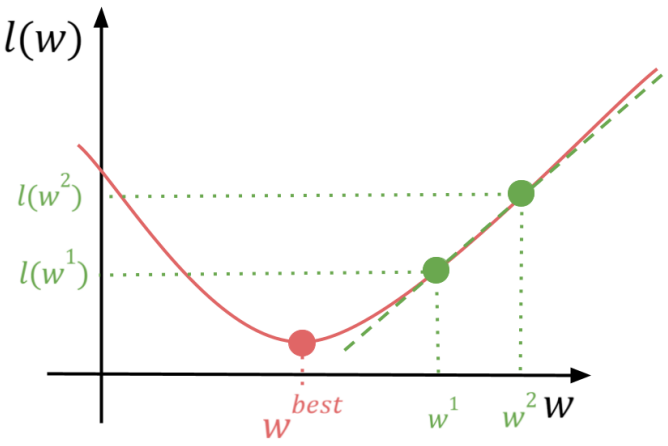
So now we know why we need to change the weights and the intuition behind how to do it. But how do we find these slopes?

Part 3: The derivative to the rescue!

We left off wanting to define the slope and coming up with how to calculate it. Now let's see what we know. We know that when we find the slope for the line of the function at point w it will show us the right direction of changing the weight w and thus minimizing the loss function. This is what we want! Now how do we find that slope? And what is it exactly?

It is super trivial when we have to find a slope based on 2 points. For example let's imagine we have two values: w^1 and w^2. In this case we can draw a line through them and then the slope is the change in $l(w)$ divided by the change in w.

$$slope = \frac{l(w^2) - l(w^1)}{w^2 - w^1}$$



Ok, cool, but we don't have two points w^1 and w^2, we just have one point $w^{current}$. So what can we do? I know! We can make a hack! We could take $w^1 = w^{current}$ an and w^2 a tiny-tiny bit larger than $w^{current}$, let's say $w^2 = w^{current} + h$ where h is a really-really small value. Then we have w^1 and w^2 so close together that they are practically the same and we can find the slope of the line in that one point *(instantaneous rate of change of the function in that point)*. Well my friends, if we make this $h \to 0$ then we are actually finding the derivative. :) The derivative is defined exactly like this:

$$\frac{dl(w)}{dw} = \lim_{h \to 0} \frac{l(w+h) - l(w)}{h}$$

If we rewrite it a bit we will see that this is the same thing we discussed above (substitute $w^1 = w^{current}$ and $w^2 = w^{current} + h$, $w^2 - w^1 = w^{current} + h - w^{current} = h$ yourself, and at the moment $w = w^{current}$).

$$\frac{dl(w)}{dw} = \lim_{h \to 0} \frac{l(w+h) - l(w)}{h} = \frac{l(w^2) - l(w^1)}{w^2 - w^1}$$

So here we have it, we have shown that the derivative of the function $l(w)$ with respect to w gives us the slope of the line at that point so if we want to know how to change w we have to just find the derivative $\frac{dl(w)}{dw}$. And most importantly, now we know why!

So finally we can write our update rule as

$$w = w - \lambda \frac{dl(w)}{dw}$$

(We will talk about the learning rate λ more during the course).

So that's it. The derivative is just conveniently giving us the correct direction of changing our weights. And this works out in larger dimensions as well.

(Also important: when you read about the derivative in the background material, lecture or other materials, usually x is used instead of w and $f(x)$ is used instead of $l(w)$ but we changed the notation in this material to emphasize that our function is the loss function and the variables we are working with are the weights.)

Part 4: Preparation for what's next.

Hopefully now you understand why we need derivatives for neural networks and for the next parts of the course you can just use derivative rules for calculating them. You don't have to remember why these rules are the way they are and prove them. Now that you know why derivatives are what we need you can just use the rules as black box to calculate stuff.

There will be more concepts introduced in this course regarding derivatives. More precisely, how to calculate them when we have many weights and a very complex loss function consisting of many layers. This is where concepts like chain rule and derivatives with matrices come into play. But we will talk about it shortly in the course.

Finally I want to talk about some terms. Sometimes terms like derivative, gradient and partial derivative can get mixed up. So let's clarify. Let's imagine that we don't have a one dimensional w, but a larger one, let's say a matrix W. Then in order to update the weights, we have to update each weight W_{ij} of the matrix separately. In order to do that we can find partial derivatives. For example, to find $\frac{dl(W)}{dW_{ij}}$ we will treat all other elements except for W_{ij} as constants. And then we will find the derivative with respect to W_{ij} and this shows us how to change W_{ij} to make the loss smaller. We can do that for each W_{ij}. All these partial

derivatives form a matrix of the same size as W (one partial derivative for each W_{ij}). And we call this matrix of partial derivatives the gradient. That's that!