Rectified Linear Unit (ReLU)

The ReLU function is a mathematical function defined as $h = \max(0, a)$ where a ($a = Wx + b$) is any real number. In simpler terms, if a is less than or equal to 0, the function returns 0. Otherwise, it returns a.

Continuity and Differentiability of the ReLU function

For a function to be differentiable, it must first be continuous. The ReLU function satisfies this requirement as it is continuous.

However, the derivative of the ReLU function "**does not exist at $a = 0$"**. This means that the ReLU function is **not differentiable at this point.**

ReLU function still used in deep learning?

Although the ReLU function is not differentiable at $a = 0$, we can still use it in deep learning with the help of Gradient Descent.

Gradient Descent is an optimization algorithm that is used to minimize the cost function in deep learning.

When the derivative of the ReLU function is not defined at $a = 0$, we set it to 0 **(or any arbitrary value)** and continue with the optimization process.

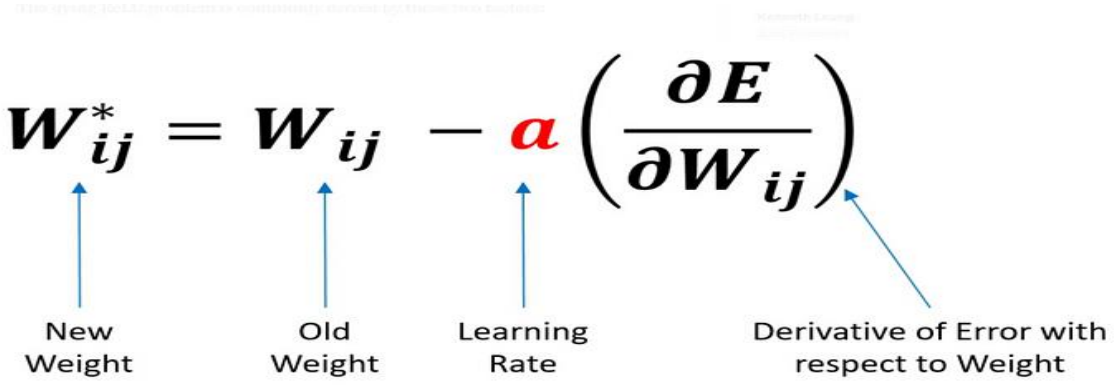
Avoiding the Vanishing Gradient Problem: ReLU can help to avoid the vanishing gradient problem, which is a common issue in deep neural networks. The vanishing gradient problem occurs when the gradients of the weights become too small during the backward propagation step, making it difficult for the optimizer to update the weights.

The constant gradient of ReLUs (when the input is positive) allows for faster learning compared to sigmoid activation functions where the gradient becomes increasingly small as the absolute value of *x* increases. This results in more efficient training of deep neural networks and has been one of the key factors in the recent advances in deep learning.

Non-linearity: Non-linear activation functions are necessary to learn complex, non-linear relationships between inputs and outputs.

Computational Efficiency: ReLU is computationally efficient and requires only a threshold operation, which is much faster than other activation functions like sigmoid or hyperbolic tangent.

Speed of Convergence: ReLU speeds up the convergence of neural networks, compared to other activation functions. This is because the gradient of ReLU is either 0 or 1, which makes it easier for the optimizer to make rapid updates.



$$f(x) = \begin{cases} x, & x \geq 0 \\ scale * x, & x < 0 \end{cases}.$$

Using an Exponential Linear Unit (ELU): The ELU function has a negative slope when the input is negative, which helps to avoid dead ReLU by allowing for some gradient flow even when the inputs are negative.

$$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha(\exp(x) - 1), & x < 0 \end{cases}$$