Exploring the Black Box (How Neural Network Learn)

For those mathematically inclined or the curious mind, this section is created specially for you.

Having the basic understanding of neural network from our workshop, this addendum is to build onto what you have learnt previously.

Some lingos you should know/have learnt:

- Cost/Loss **Function** - A technique we use to measure the performance/correctness of our algorithm/machine learning model

- Cost/Loss **Value** - This value signifies **how well or poorly a certain machine learning model behaves after each iteration of training/learning**

- Weights - Control the signal (or the strength of the connection) between two neurons. In other words, a weight decides how much influence the input will have on the output.

- Backpropagation - A way of propagating the total loss back into the neural network to know how much of the loss every node is responsible for.

The Gist:

When training a machine learning model, an **optimization algorithm** is used to achieve the minimum cost/loss value. (Local Minimum - We will explain what it means shortly)

This optimization algorithm make uses of **differentiation (calculus)** to find the **minimum point** in the cost function

During each training iteration, the cost value is lowered over time as the weight (Random initial value) is slowly tuned to w which is the optimal weight where is cost/loss value is the lowest). By using differentiation, at each learning step, the differentiated value along the cost function is measured and compared against the previous iteration. The goal here is to get the differentiated value as close to zero **(Recall: In calculus class we learn that min/max point a.k.a stationary point is at where dy/dx = 0)**
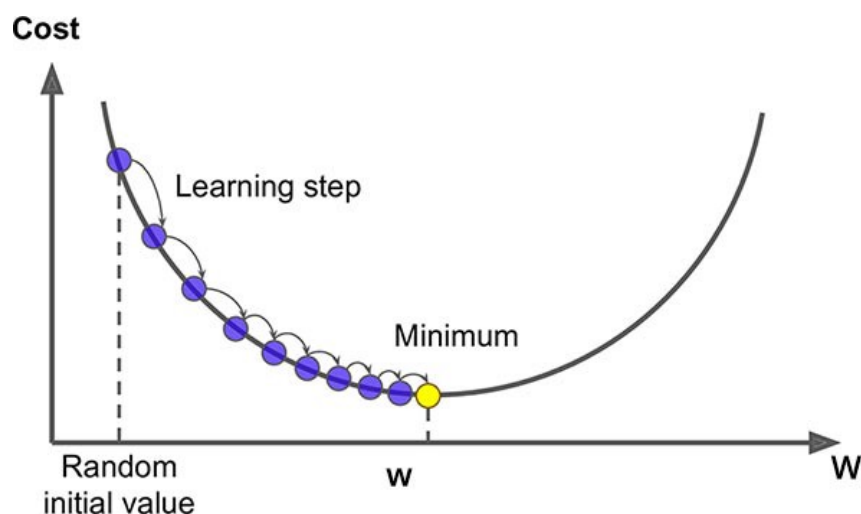
However, life isn't that simple. As the cost function can get pretty complex in machine learning/deep learning. Therefore, when we have cost function C as seen **below**, finding the true/global minimum (See yellow circle) is no longer as simple as we would like it to be.
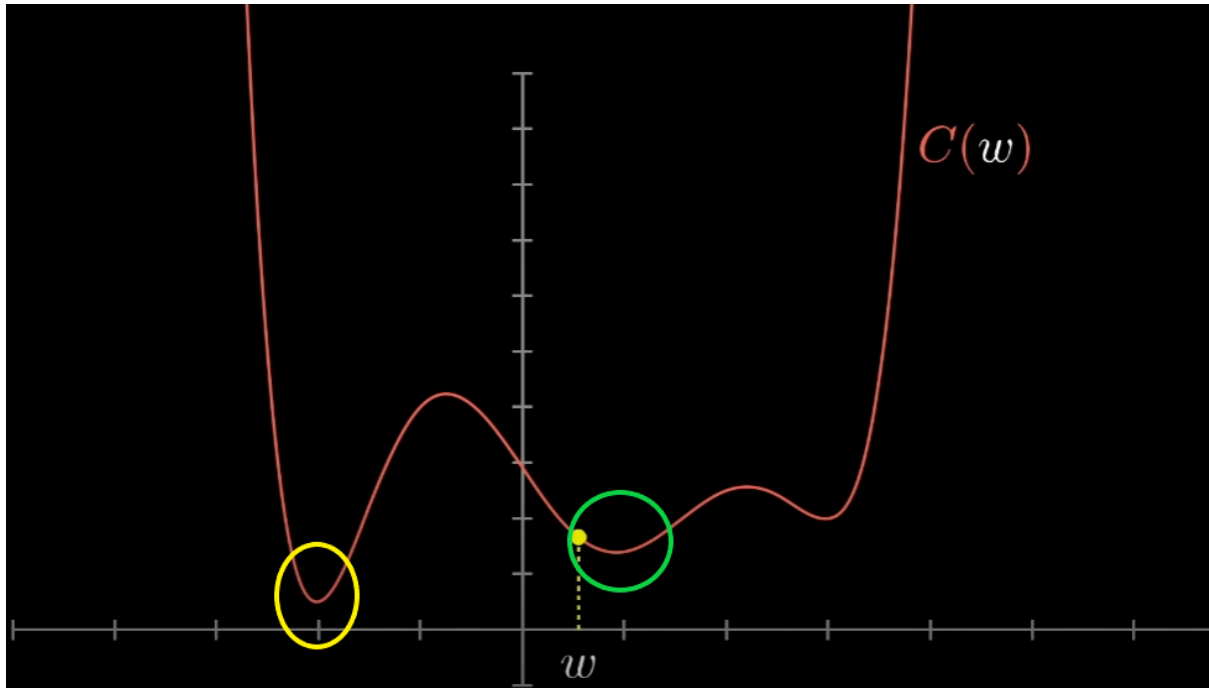
However, with the optimization algorithm, we would still be able to find the local minimum (Green Circle) and still obtain decently accurate results.

This optimization is known as **gradient descent**

> 💡 Gradient descent is **an optimization algorithm used to minimize some function by iteratively moving in** the direction of steepest descent as defined by the negative of the gradient. In machine learning, we use gradient descent to update the parameters of our model.

To have a more comprehensive understanding, you might find this video helpful: (Start: 3:02)

https://youtu.be/IHZwWFHWa-w?t=182

Readings:

https://towardsdatascience.com/how-neural-network-learn-3b56c175b5ca

https://www.kdnuggets.com/2015/12/how-do-neural-networks-learn.html