DEEP LEARNING

Deep Learning ones to mimic the human brain.

First, we have to understand how neurons work.

In a neuron we have different parts:

* Cell Body
* Dendrites: are actually receivers, receivve message and signals from other sources
* Axon: are actually transformers. Are the responsible ones of transforming the captured signal to be able to resend it.
* Terminals: are responsible for passing the signal to other neurons in the last stage.

Our mail is receiving signals by seeing, touching… this signals are transfered through the dendrites to the neuron and then, through axons and terminals are transferred to other neurons, finally giving the output to our brain.

For computer format, we have the same concepts.

We use our variables as input layer (our data). Then our data goes through several number of neurons (hidden layers) giving us at the end the output layer (y).

We have to add some weights to the axons, to give the ability to the neurons to decide which signal is important and which is not. Then, weights go through the neurons and, in the neurons, we simply add the inputs, multiplied by weights and adding the bias. Then we multiply the result by the activation function to decide if the input will go to next step or not.

**Activation Function**

The activation function is a function which decides which signals will pass and which not. There are several types of activation functions

* Treshold function: type of boolean function. We define a treshold and if the result is greater than this treshold, y will be activated and, if not, result will be zero (won’t be activated). It is a linear function.
* Sigmoid function: adds non-linearity. Is good for classification. The advantage upon linear functions is that limits output between 0 and 1. The formula is (1/(1+e^(-x))
* Hyperbolic Tangent Function: really similar to Sigmoid function but the range of the output is between -1 and 1 instead than 0 and 1. The formula is also quite similar to the sigmoid ((2/(1+e^(-2x))-1). Reformulating it to the sigmoid: 2sigmoid(2x)-1. Another difference than the one in the range, is the existing one in the slope of change, which is steeper in Hyperbolic tangent.
* ReLu Function: One of the most famous functions and specially used in time series analysis. It gives output x if x positive and if it’s negative, it is 0. The advantage is that it prevents to activate a portion of the neurons and with this, it prevents to have a dense network, making the model computationally less expensive.

Which function should we use? It depends. The type of the model is really important, as well as the type and characteristics of the data.

**How a neural network actually works?**

In practice, we will have several nodes in the hidden layer. Each node in the input layer would be connected with all the other nodes in the hidden layer.

We have to remember that weights will be sometimes zero, so the connection with weight 0 will be removed.

Let’s make an example. Imagine input 4 (x4), which has 3 connections with its 3 weights, connected to the nodes in the hidden layer. Then, we will get a y, predicted value. We have also ý (actual value).

The next step is to check the cost function (1/2(Y-Ý)^2). **Our objective is to minimize the cost function.**

Then, based on the cost function result, the model gives feedback and the weights are updated.

The final aim is to minimize the overall cost function. Our example for x4 does not happen as a standalone activity, but at the same time than the update and modelling with all the nodes at the same time.

This process of feedback and update is what is called backpropagation.