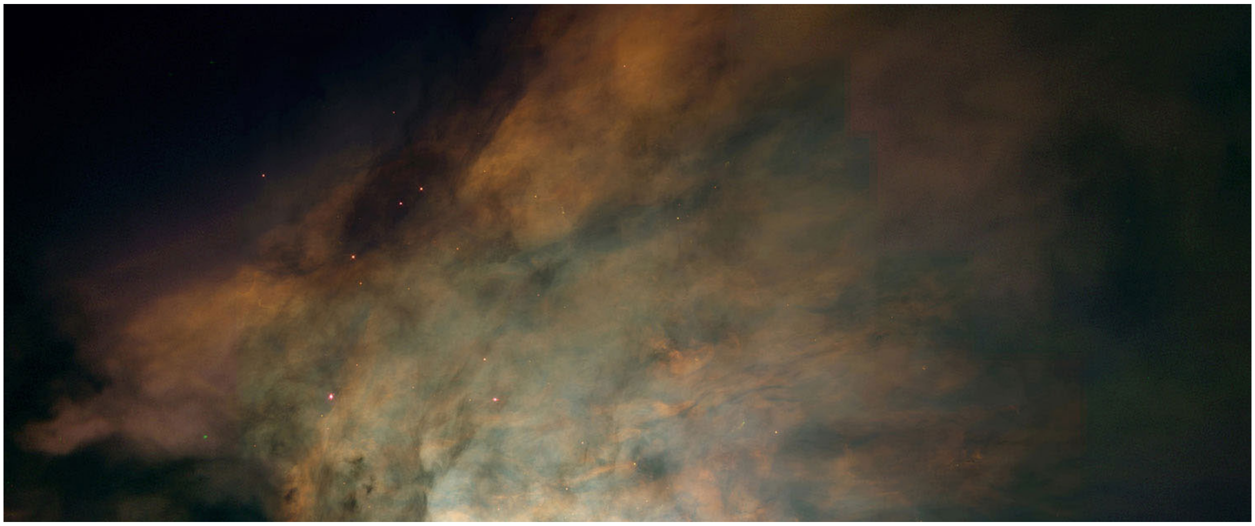
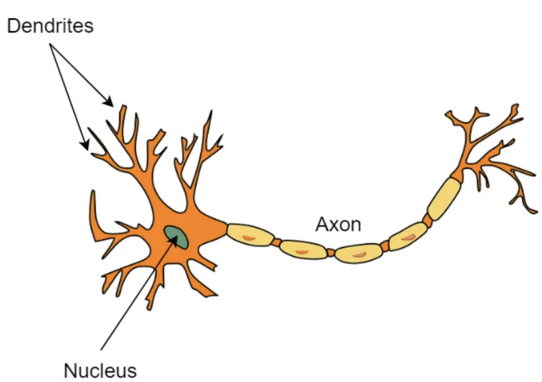
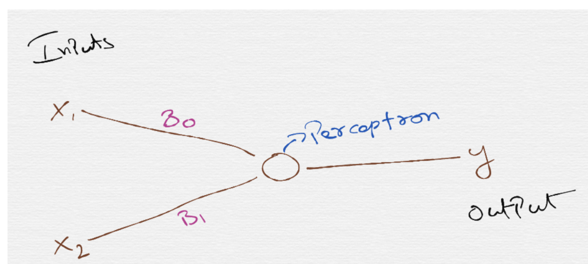


Introduction to Neural Networks

These following notes were written by Vasanth based on the lecture given by Mr. Sudarsun Santhiappan, PhD, held on 20th May 2024

Perceptron



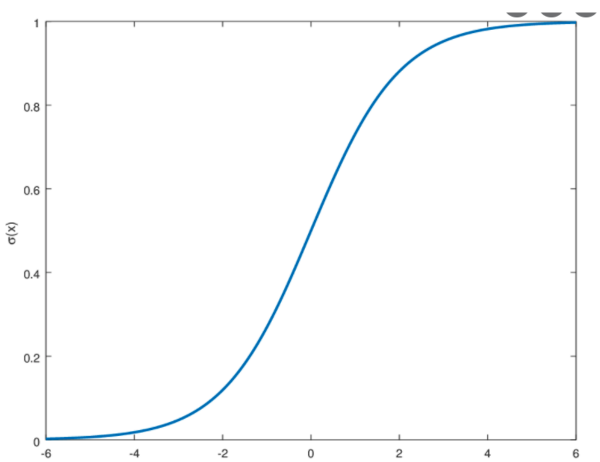


The concept of perceptrons is indeed inspired by the biological neurons in our brains. Perceptrons are the building blocks of artificial neural networks, and they work in a similar way to how a single neuron processes information. Biological neurons receive signals from other neurons through junctions called synapses. These signals have varying strengths, influencing how much they affect the receiving neuron. Perceptrons mimic this by having multiple inputs, each associated with a weight. The weight determines the importance of that particular input to the perceptron's output.

A perceptron takes various inputs, multiplies each by its corresponding weight, and sums them up. This weighted sum is then passed through an activation function, which determines the output of the perceptron. The activation function introduces non-linearity, allowing the perceptron to learn more complex patterns compared to a simple linear model. Perceptrons use a learning algorithm to adjust the weights automatically. By comparing the desired output (from labeled training data) with the actual output, the perceptron can fine-tune its weights to improve the accuracy of its predictions.

Sigmoid Activation function

Sigmoid is a common activation function used in perceptrons. It takes a continuous input value and transforms it into an output value between 0 and 1.



$$Sigmoid(x) = \frac{1}{1+e^{-x}}$$

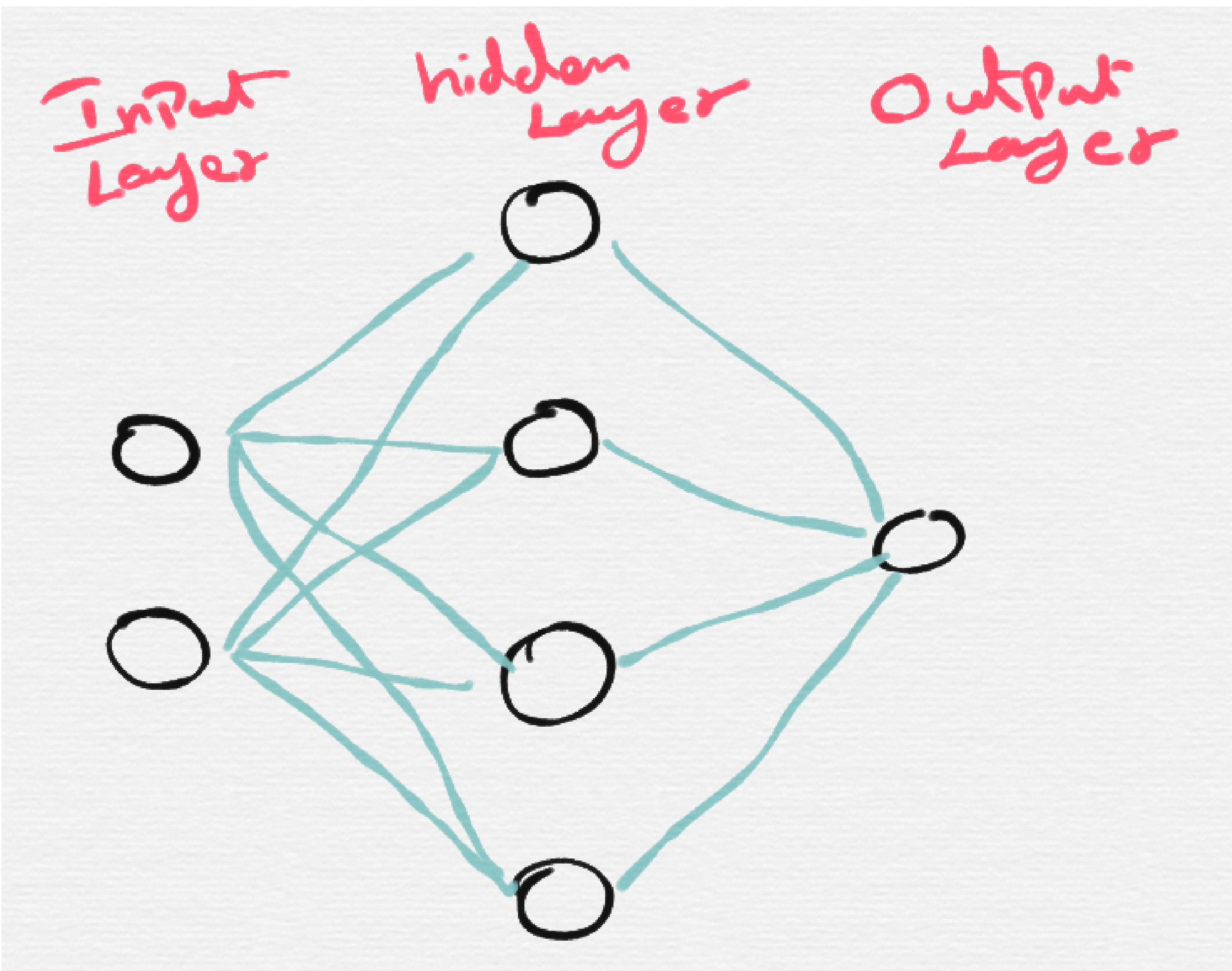
Multi Layer Perceptron (MLP)

While a single perceptron has limited capabilities, connecting multiple perceptrons in layers forms a neural network. This allows the network to learn intricate relationships between inputs and outputs, making them powerful tools for various machine learning tasks.

A multilayer perceptron (MLP) is a type of artificial neural network architecture that utilizes multiple hidden layers. Each hidden layer consists of several nodes, and these nodes are **fully connected** to the nodes in the preceding layer. In other words, every node in a hidden layer receives input from **all** the nodes in the previous layer.

For example, consider an MLP with two input nodes and a hidden layer containing four nodes. Each of the two input nodes will be connected to all four nodes in the hidden layer. This results in a total of 2 (input nodes) x 4 (hidden layer nodes) = 8 connections.

This type of connection scheme, where every node in a layer connects to every node in the subsequent layer, is referred to as a **fully connected layer**. It's a fundamental building block of MLPs and enables them to extract intricate features from the data they process.



Softmax

It is only used in the output layer in the network, Softmax function converts real values into probabilities, The higher probability will be the end result.

$$f(y_i) = \frac{e^{y_i}}{\sum_{k=0}^{n} e^{y_k}}$$

https://www.youtube.com/watch?v=8ah-qhvaQqU