UNIT – 1

NEURAL NETWORKS AS UNIVERSAL APPROXIMATES

The concept of Neural Networks has been around us for a few decades. Why did it take so much time to pace up? What is this sudden boom that Neural Networks and Deep Learning created? What makes Neural Networks this hype-worthy? Let's explore.
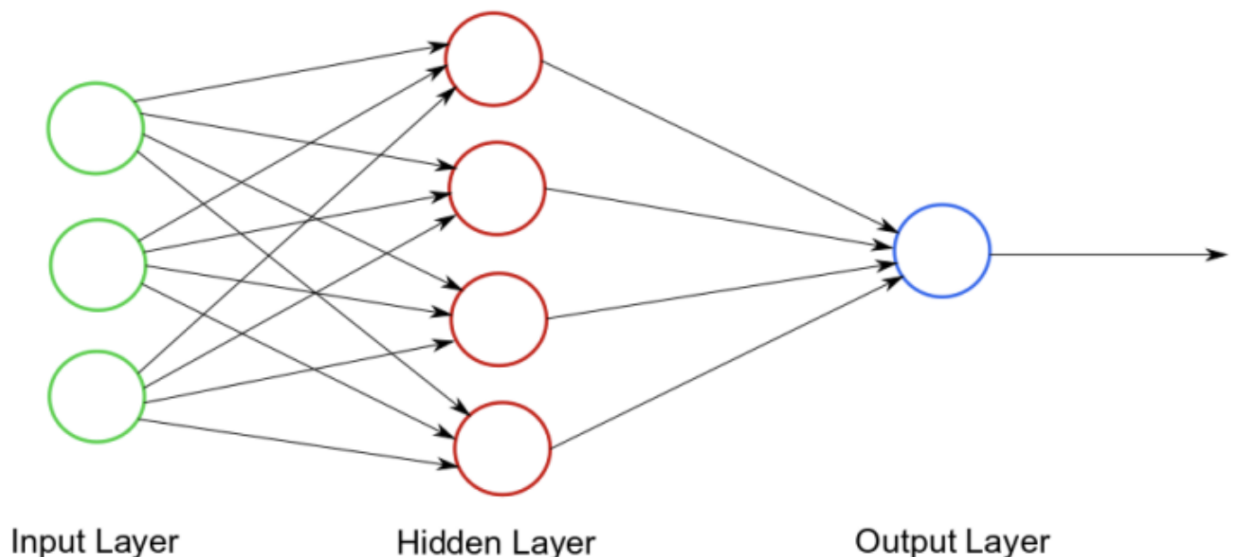

The architecture of a Neural Network

To get a brief overview of what Neural Networks are, a neural network is simply a collection of Neurons(also known as activations), that are connected through various layers. It attempts to learn the mapping of input data to output data, on being provided a training set.

The *training* of the neural network later facilitates the predictions made by it on a testing data of the same distribution. This mapping is attained by a set of trainable parameters called *weights*, distributed over different layers. The weights are *learned* by

the backpropagation algorithm whose aim is to minimize a loss function. A loss function measures how ***distant*** the predictions made by the network are from the actual values. Every layer in a neural network is followed by an activation layer that performs some additional operations on the neurons.

The Universal Approximation Theorem

Mathematically speaking, any neural network architecture aims at finding any mathematical function $y= f(x)$ that can map attributes(x) to output(y). The accuracy of this function i.e. mapping differs depending on the distribution of the dataset and the architecture of the network employed. The function $f(x)$ can be arbitrarily complex. The Universal Approximation Theorem tells us that Neural Networks has a kind of ***universality*** i.e. no matter what $f(x)$ is, there is a network that can approximately approach the result and do the job! This result holds for any number of inputs and outputs.
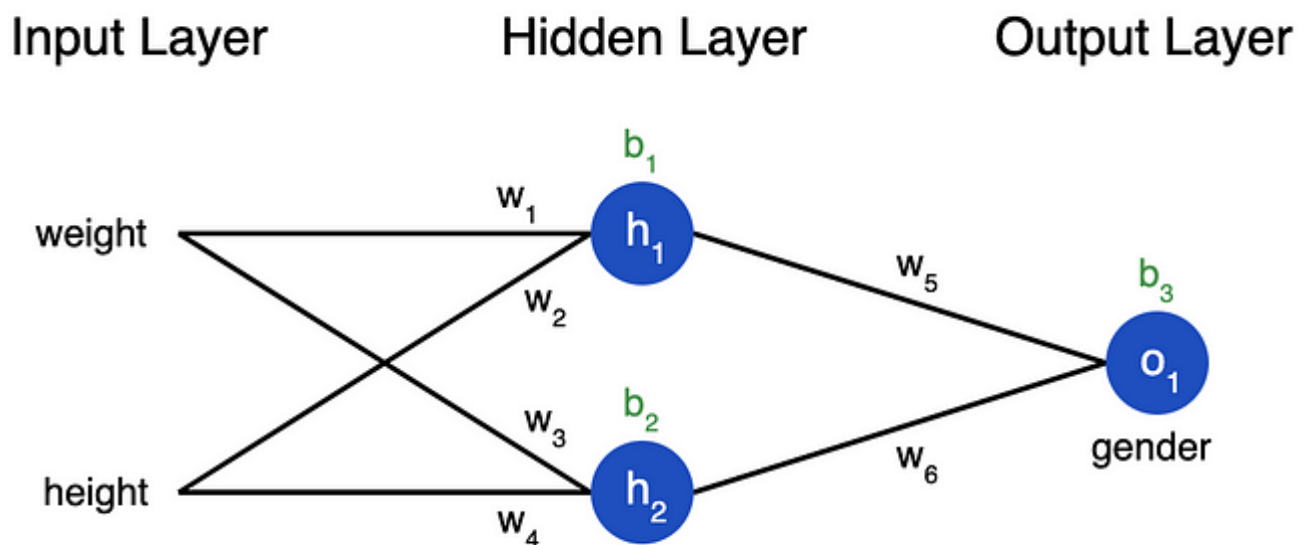
image source

If we observe the neural network above, considering the input attributes provided as height and width, our job is to predict the gender of the person. If we exclude all the activation layers from the above network, we realize that h_1 is a linear function of both weight and height with parameters w_1, w_2, and the bias term b_1. Therefore mathematically,

$$h_1 = w_1*\text{weight} + w_2*\text{height} + b_1$$

Similarily,

$$h2 = w_3*\text{weight} + w_4*\text{height} + b_2$$

Going along these lines we realize that o1 is also a linear function of h_1 and h2, and therefore depends linearly on input attributes weight and height as well. This essentially boils down to a [linear regression](#) model. Does a linear function suffice at approaching the Universal Approximation Theorem? The answer is NO. This is where activation layers come into play.

An activation layer is applied right after a linear layer in the Neural Network to provide non-linearities. Non-linearities help Neural Networks perform more complex tasks. An activation layer operates on activations (h_1, h2 in this case) and modifies them according to the [activation function](#) provided for that particular activation layer. Activation functions are generally non-linear except for the [identity function](#). Some commonly used activation functions are [ReLu](#), [sigmoid](#), [softmax](#), etc. With the introduction of non-linearities along with linear terms, it becomes possible for a neural network to model any given function approximately on having appropriate parameters(w_1, w_2, b_1, etc in this case). The parameters converge to *appropriateness* on training suitably. You can get better acquainted mathematically with the Universal Approximation theorem from [here](#).

Neural networks have the capability to map complex functions and have been a theory on paper since forever. What made it a prodigy in Machine learning suddenly?

The boom

A recent explosion for interest in deep learning models is credited to the high computational resources and the enriching data that the world has to offer nowadays. Deep neural networks are data-hungry models. This boom is also majorly credited to the inexpensive high-speed computing that has arrived in the hands of the common folks. This unprecedented increase in data along with computational power has created wonders in almost all domains of life.

Deep learning models are firmly believed to extract features from raw data automatically, a concept also known as feature learning. No matter what you feed into a sufficiently large and deep Neural Network, it can learn hidden features and relations between attributes and later leverages those same relations to predict results. This comes real handy and requires minimal preprocessing of data. Along with this, the tools and frameworks(PyTorch, Tensorflow, Theano) used to design and build these data-driven models are increasing by the day, are pretty high level, and are easily available. They require an inconsiderable low-level understanding of programming languages. On top of it, research by top companies has proved that this domain is indeed worth spending valuable time and money on.
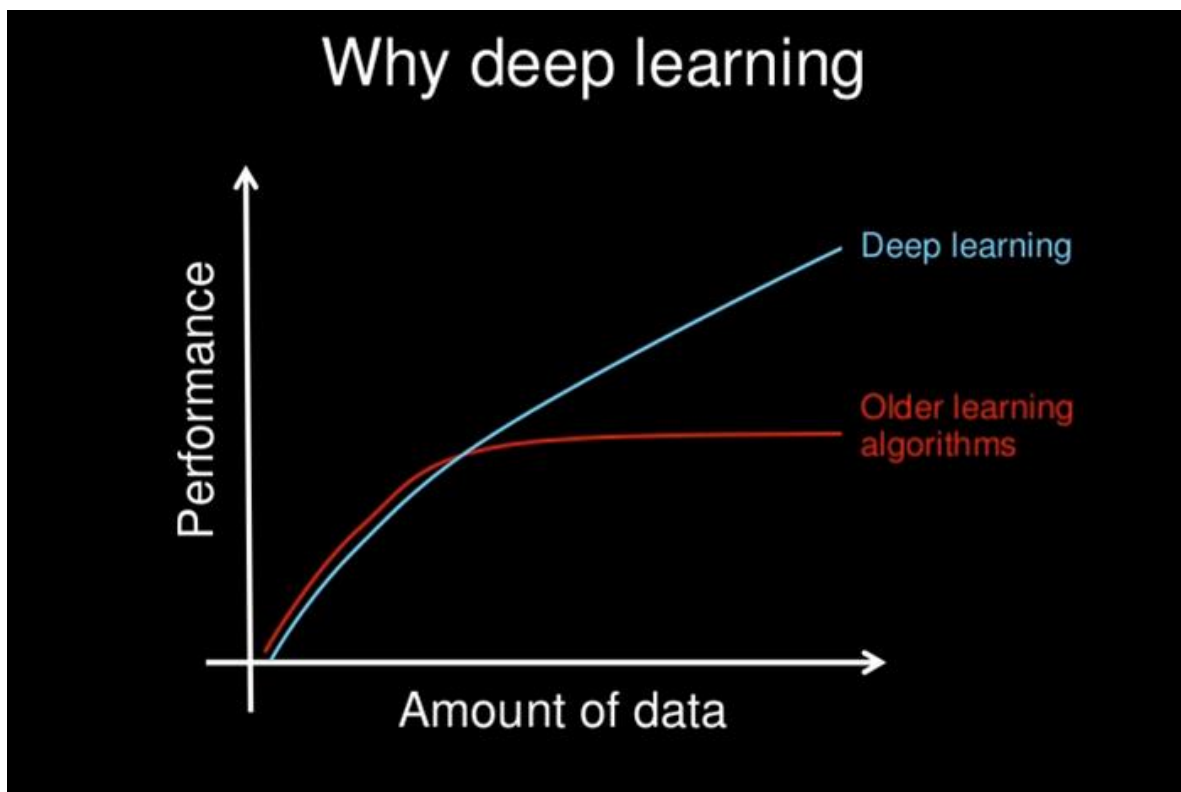
models scaled with data

It is highly acclaimed that deep learning models are indeed scalable with data. This indicates that the results almost always get better with the increase in data and by employing larger models. These larger models require more computation to train. With the advent of conducive computational environments that are easily accessible nowadays, it has, therefore, become easier to experiment and improve the algorithms and architectures in real-time, thus giving rise to better and better practices in short spans. That being said, Deep Neural Networks have wide found applications in many domains like Computer Vision, Natural Language Processing,

Recommender Systems, and much more. Various cross-domain applications have also picked up pace recently.

In October 2019, Google published the results of its Quantum Supremacy experiment on "Sycamore," its 54-qubit processor. The quantum computer conducted the target calculation in 200 seconds, which would take about 10,000 years for the world's fastest supercomputers. With this increase in computational power by each passing day, there's no knowing when Machine Learning will transcend superhuman boundaries. One can only speculate.

Reference:

https://towardsdatascience.com/neural-networks-and-the-universal-approximation-theorem-8a389a33d30a