Universal Approximation Theorem

Any continuous function f: $[0, 1]^n \to [0, 1]$ can be approximated arbitrarily well by a neural network with at least 1 hidden layer with a finite number of weights.

History

- Proposed in the 1980’s notably by George Cybenko, Kurt Hornik and Stuart E.Dreyfus

- Formalised as the Universal Approximation Theorem(UAT) in Cybenko’s paper “Approximation by superposition of a sigmoidal function”

- The theorem claimed that feedforward neural networks with an activation function can be used to approximate any continuous function

Significance

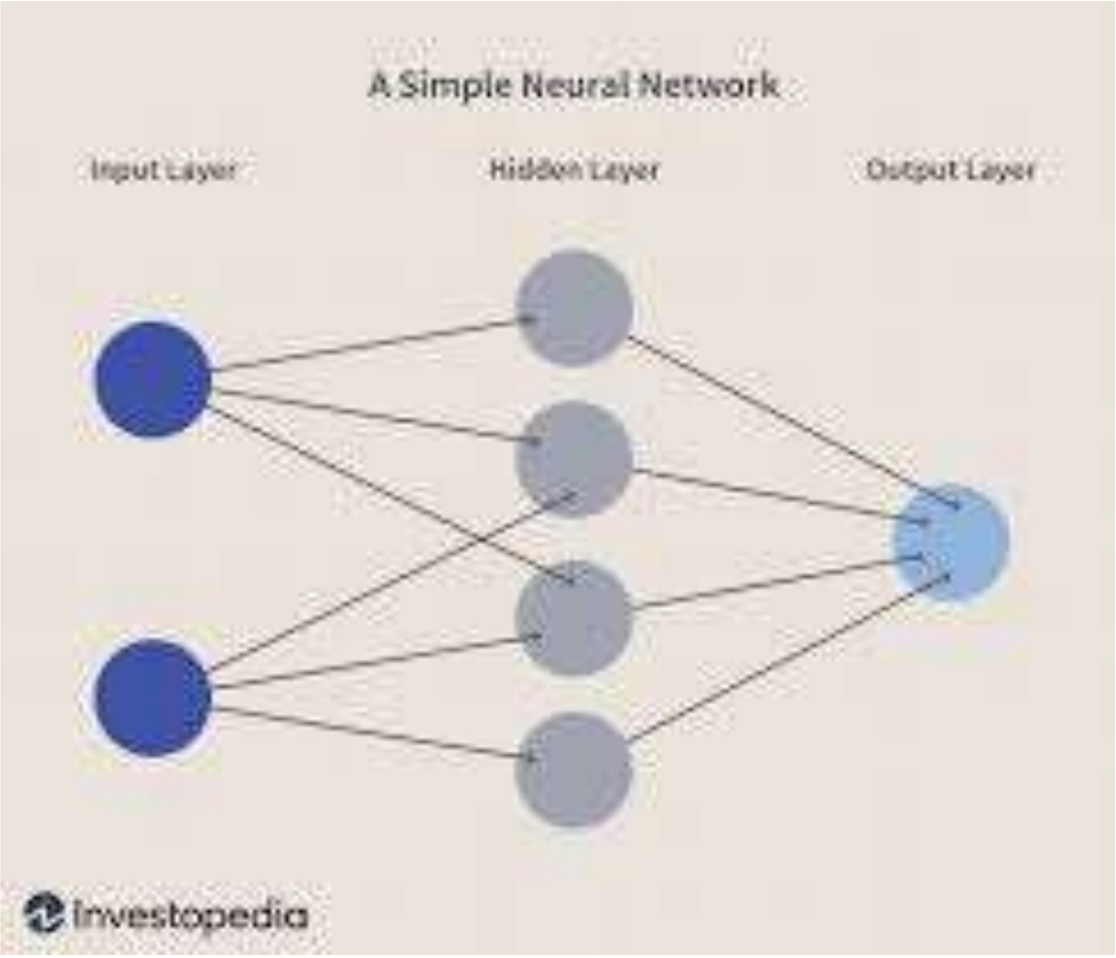
- UAT popularised the use of neural networks to approximate even very complex functions

- A key reason for the versatility and effectiveness of neural networks in various applications.

- Explains how neural networks can find relationships between inputs outputs even when the actual function in unknown.

Neural Networks

- A neural network is made of interconnected neurons.
- Each neuron is a mathematical function applied on the input
- The function applied varies based on parameters of each neuron.
- Parameters are altered during training so that the result resembles the training data



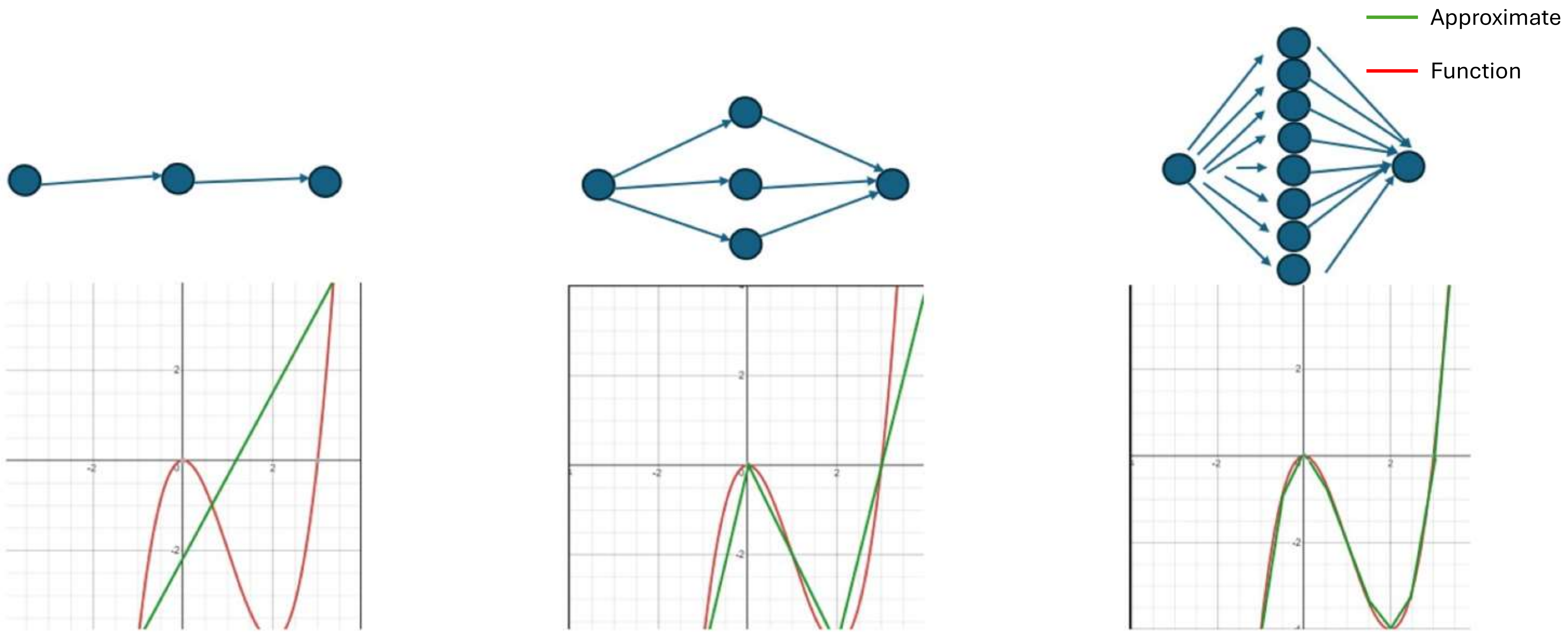
Activation Functions

- Activation functions are the mathematical functions applied by neurons.
- They introduce non-linearity to the outputs of the neural network.
- Enables neural networks to approximate non-linear functions.
- By making the output of each neuron independent and compounds the results for the output.

Activation Functions

- Sigmoid Function - $\sigma(x) = \frac{1}{1+e^{-x}}$
- Hyperbolic Tangent - $\tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$
- Rectified Linear Unit - ReLU(x) = max(0,x)

Visual Representation



1 neuron

3 neurons

8 neurons

Visual Representation

- The approximate function is a series of lines that follows the shape of function

- As we increase the number of lines/neurons we get a more accurate approximation.

- The parameters of a neuron is like the length and direction of each line

Training

- The parameters in neural networks are called weights and biases.

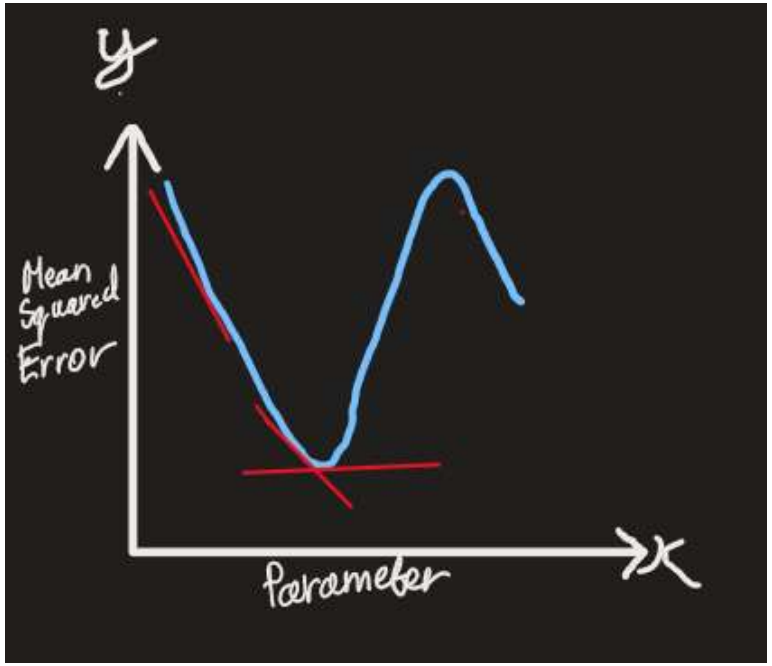
- A weight is a multiplier applied onto the input

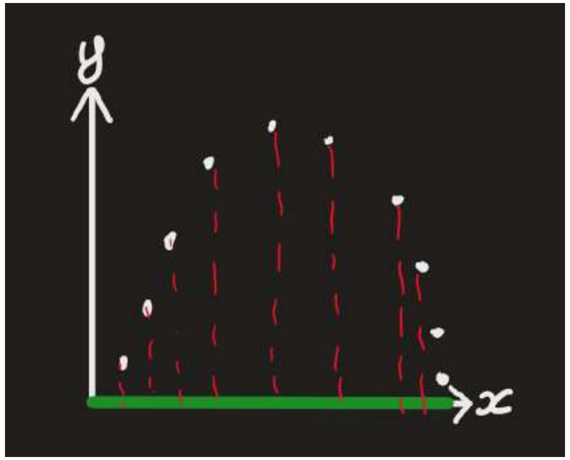
- A bias is an addition applied onto the input

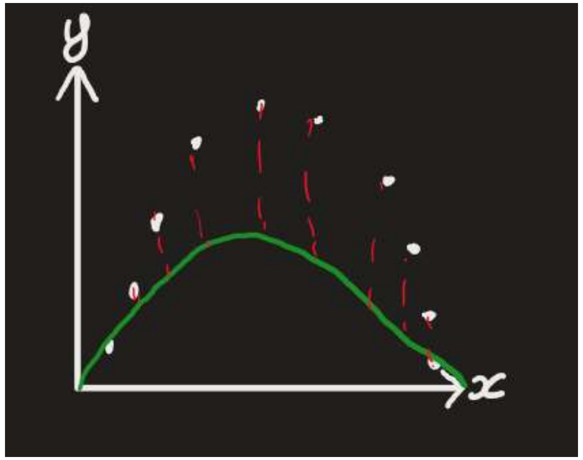
- Weights and Biases are optimised during training through back propagation.

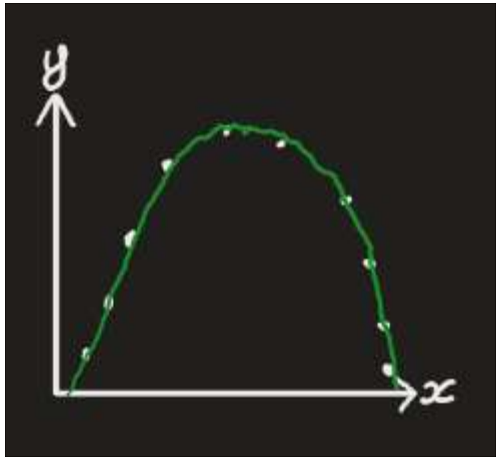
Backpropagation

Using the data set, we work backwords assigning values to parameters and compare output from the network, to the dataset. We change the parameter values till we minimise the error function







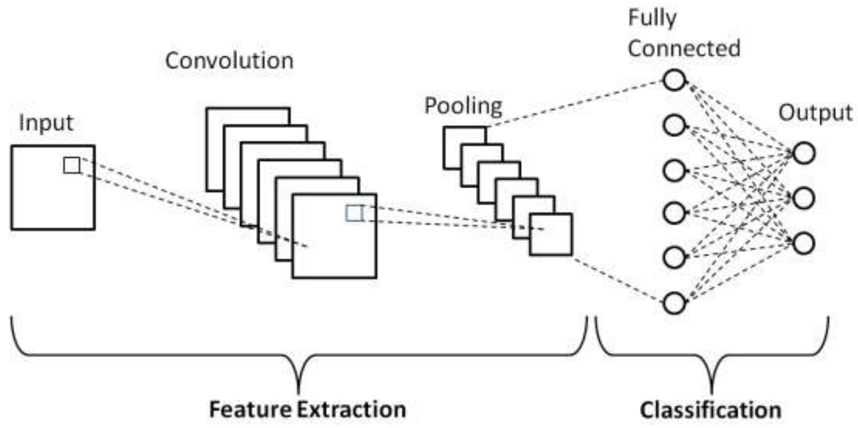


Applications of Neural Networks

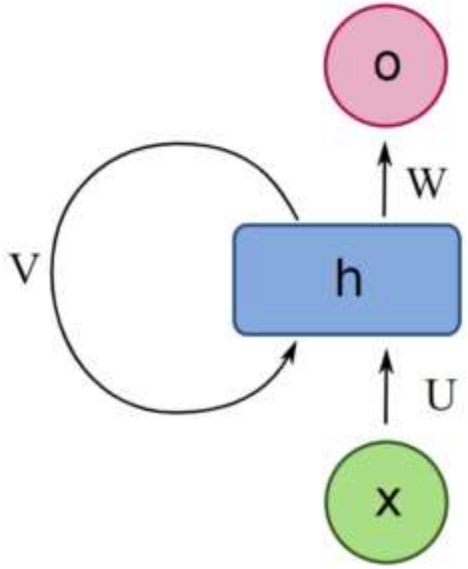
- Machine Learning & Artificial Intelligence
- Image and Video Processing
- Natural Language Processing
- Control Systems

Types of Neural Networks

- Convolutional neural networks(CNN), optimised for image processing & identification, by looking for patterns in images that resemble objects using layers of filters. Instead of analysing individual pixel data.



- Recurrent neural networks(RNN), optimised for sequential data by having feedback loops in its layers, allowing it to remember previous inputs.



Long Short-Term Memory