Summary :
Intro to Deep Learning

by : Dhea Fajriati Anas

TABLE OF CONTENTS

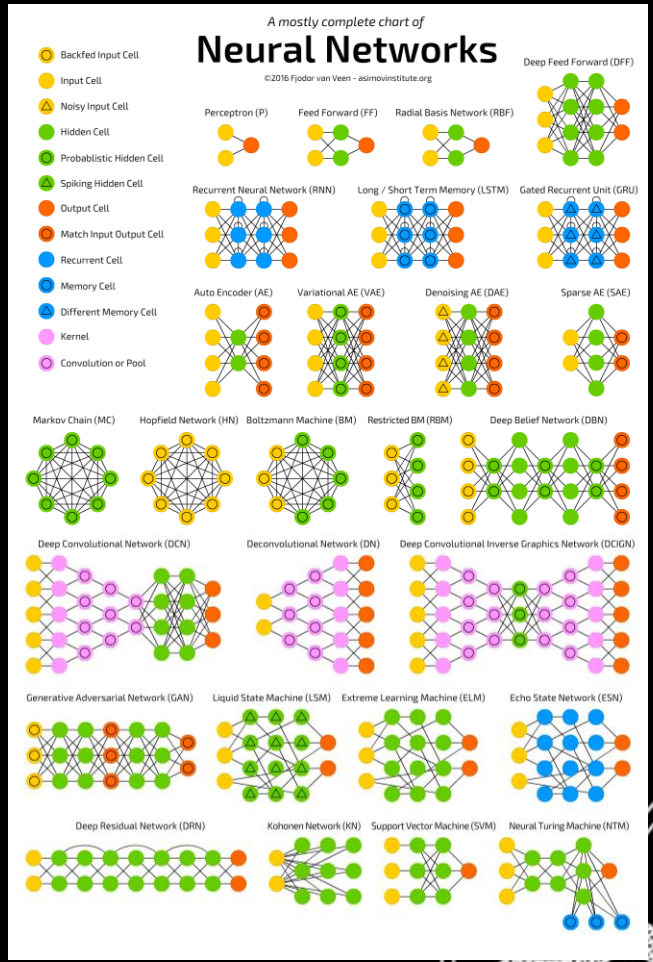
01

INTRODUCTION

NEURAL NETWORKS

Artificial neural networks (ANNs) are comprised of a node layers, containing an input layer, one or more hidden layers, and an output layer.

Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.



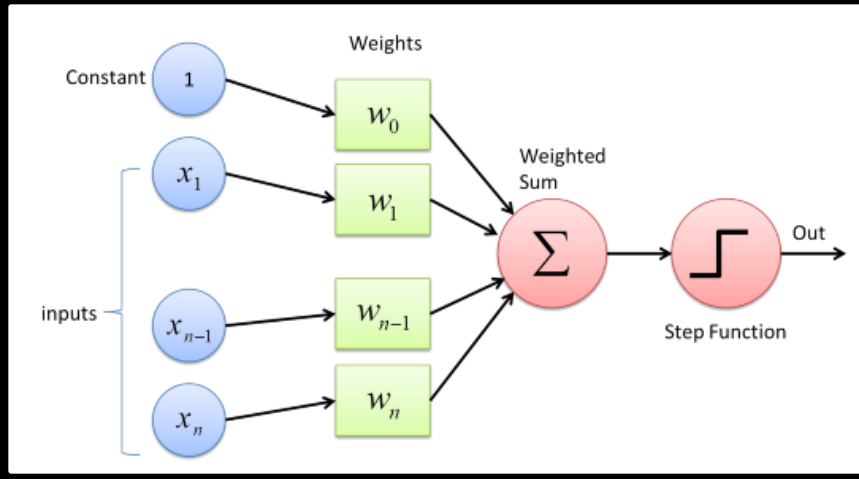
02

MULTI LAYER PERCEPTRON (MLP)

PERCEPTRON

Perceptron is a single layer neural network and a **multi-layer perceptron** is called Neural Networks. A neural network is an interconnected system of perceptrons, so it is safe to say perceptrons are the foundation of any neural network.

The perceptron is supervised learning that used for binary classification, and using the Heaviside step function for the activation function.



The perceptron consists of 4 parts.

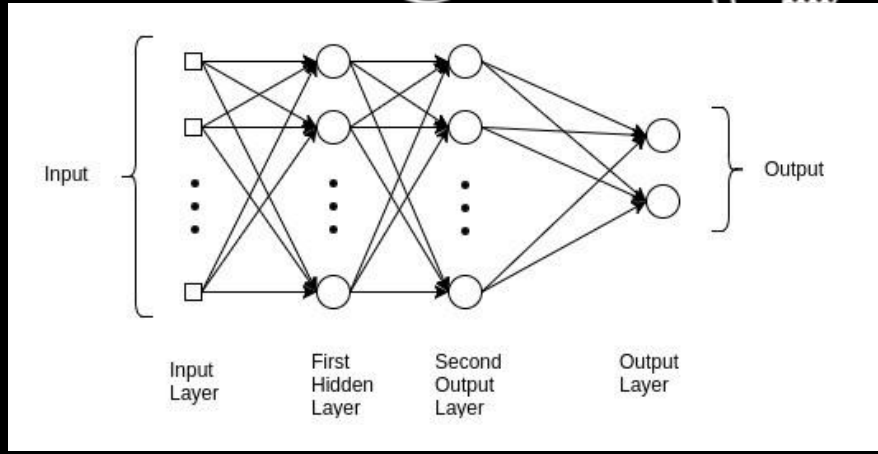
1. Input values or One input layer
2. Weights and Bias
3. Net sum
4. Activation Function

MLP

In the Multilayer perceptron, there can more than one linear layer (combinations of neurons).

If we take the simple example the three-layer network, first **layer** will be the *input layer* and last will be *output layer* and middle layer will be called *hidden layer*.

We feed our input data into the input layer and take the output from the output layer. We can increase the number of the hidden layer as much as we want, to make the model more complex according to our task.



03

BACKPROPAGATION AND ACTIVATION FUNCTION

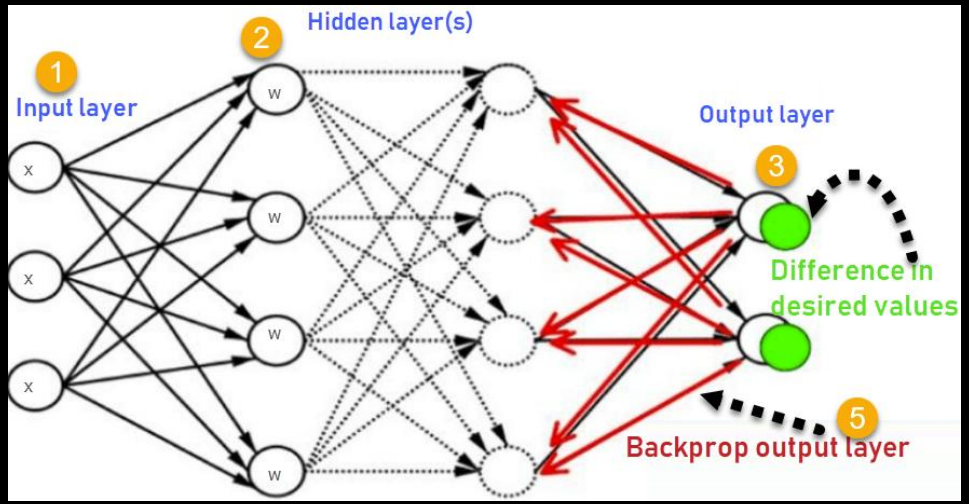
BACKPROPAGATION

The backpropagation algorithm in neural network computes the gradient of the loss function for a single weight by the chain rule.

How Backpropagation Algorithm Works ?

1. Inputs X, arrive through the preconnected path
2. Input is modeled using real weights W. The weights are usually randomly selected.
3. Calculate the output for every neuron from the input layer, to the hidden layers, to the output layer.
4. Calculate the error in the outputs
5. Travel back from the output layer to the hidden layer to adjust the weights such that the error is decreased.

Keep repeating the process until the desired output is achieved

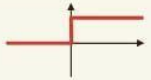
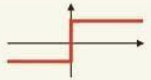
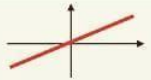
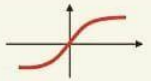


ACTIVATION FUNCTION

Activation function use to get the output of node, also known as **Transfer Function**. It is used to determine the output of neural network like yes or no. It maps the resulting values in between 0 to 1 or -1 to 1 etc. (depending upon the function).

The activation functions can be basically divided into 2 types :

1. Linear Activation Function
2. Non-linear Activation Functions

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$	Multi-layer Neural Networks	
Rectifier, ReLU (Rectified Linear Unit)	$\phi(z) = max(0, z)$	Multi-layer Neural Networks	
Rectifier, softplus	$\phi(z) = \ln(1 + e^{z})$	Multi-layer Neural Networks	

04

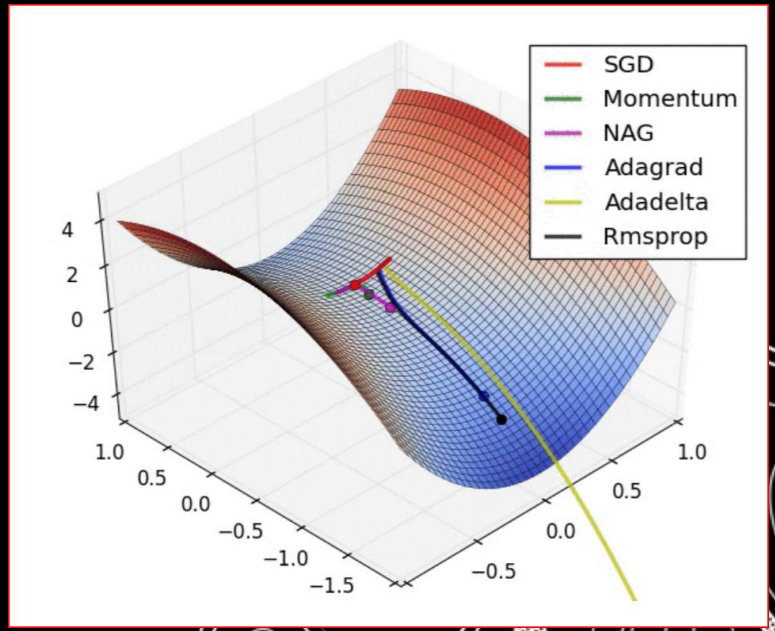
OPTIMIZER

DEFINITION

Optimizers are algorithms or methods used to change the attributes of the neural network such as weights and learning rate to reduce the losses. Optimizers are used to solve optimization problems by minimizing the function.

Types of optimizers :

- Gradient Descent
- Stochastic Gradient Descent (SGD)
- Mini Batch Stochastic Gradient Descent (MB-SGD)
- SGD with momentum
- Nesterov Accelerated Gradient (NAG)
- Adaptive Gradient (AdaGrad)
- AdaDelta
- RMSprop
- Adam



05

LOSS FUNCTION

DEFINITION

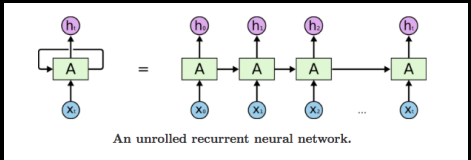
Loss is nothing but a prediction error of Neural Network, and the method to calculate the loss is called **Loss Function.** In simple words, the Loss is used to calculate the gradients, and gradients are used to update the weights of the Neural Network.

1	Loss functions in Regression based problem	a. Mean Square Error Loss
		b. Mean Absolute Error Loss
		c. Huber Loss
2	Loss functions in Binary classification-based problem	a. Binary Cross Entropy Loss
		b. Hinge Loss
3	Loss functions in Multiclass classification-based problem	a. Multiclass Cross Entropy Loss
		b. Spare Multiclass Cross Entropy Loss
		c. Kullback Leibler Divergence Loss

06

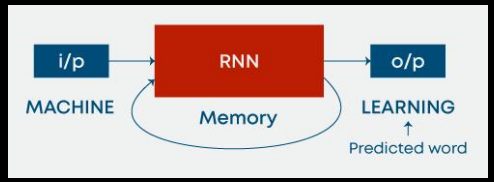
OTHER TYPES OF NEURAL NETWORKS

RNN



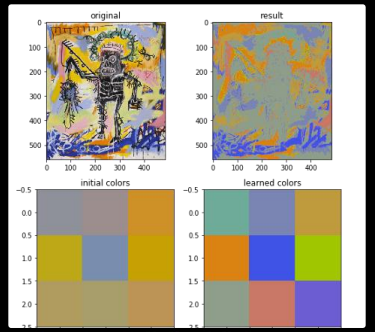
Recurrent Neural Network is a generalization of feedforward neural network that has an internal memory. RNN is recurrent in nature as it performs the same function for every input of data while the output of the current input depends on the past one computation.

LSTM



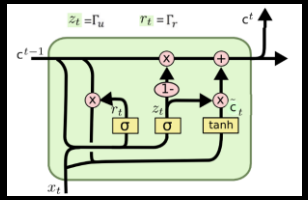
Long Short-Term Memory (LSTM) networks are a modified version of recurrent neural networks, which makes it easier to remember past data in memory. The vanishing gradient problem of RNN is resolved here.

SOM



Self Organizing Map (SOM) is an unsupervised neural network machine learning technique. SOM is used when the dataset has a lot of attributes because it produces a low-dimensional, most of times two-dimensional, output. The output is a discretised representation of the input space called map.

GRU



The Gated Recurrent Unit (GRU) is a type of Recurrent Neural Network (RNN) that, in certain cases, has advantages over long short term memory (LSTM). GRU uses less memory and is faster than LSTM, however, LSTM is more accurate when using datasets with longer sequences.

]


THANKS !