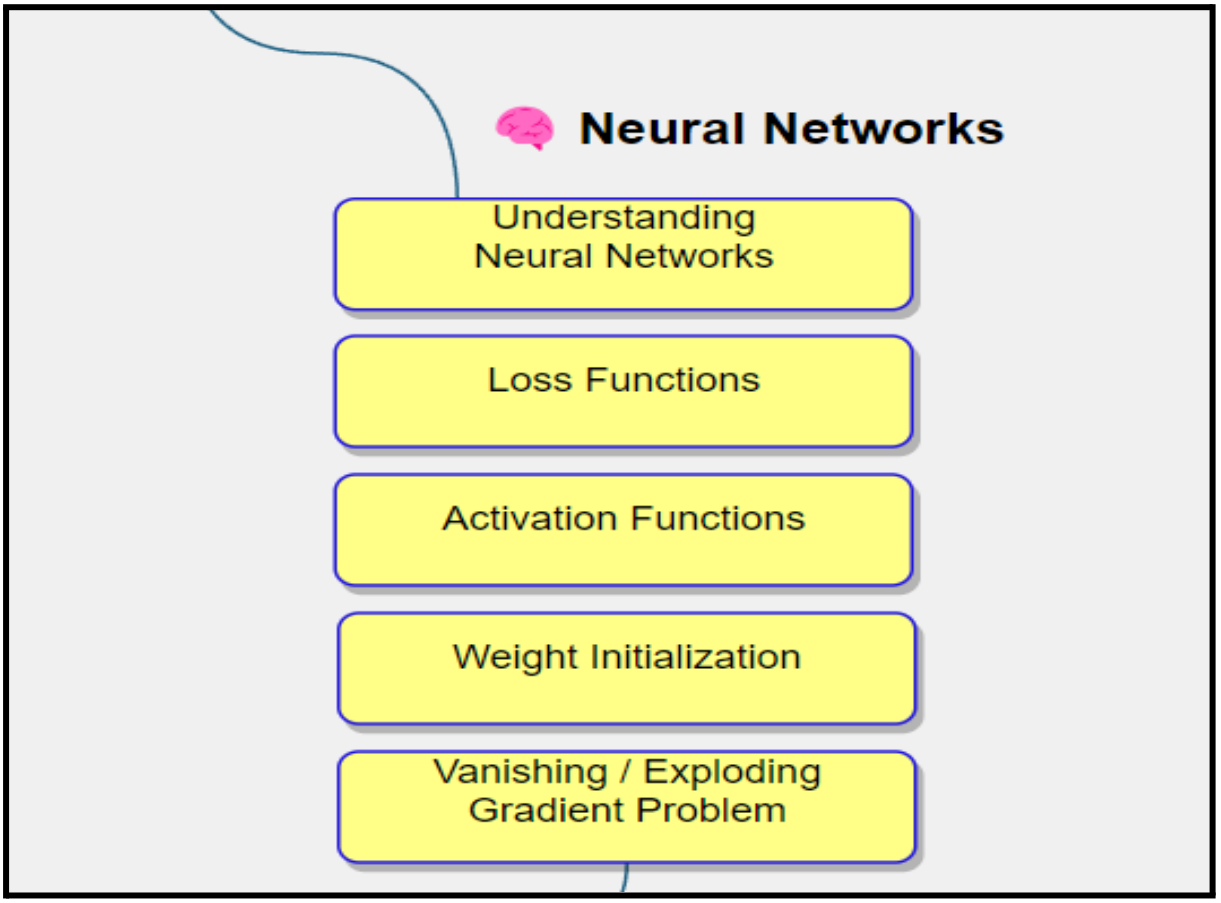
.

TECHNEEDS
WEEK-2
NOTES



INTRODUCTION TO DEEP LEARNING

Today, deep learning is a method which is used as an alternative to machine learning and has many benefits. It is a technique which mimics the human brain and has **NEURAL NETWORKS**

- Neural networks are one of the most beautiful programming paradigms ever invented.

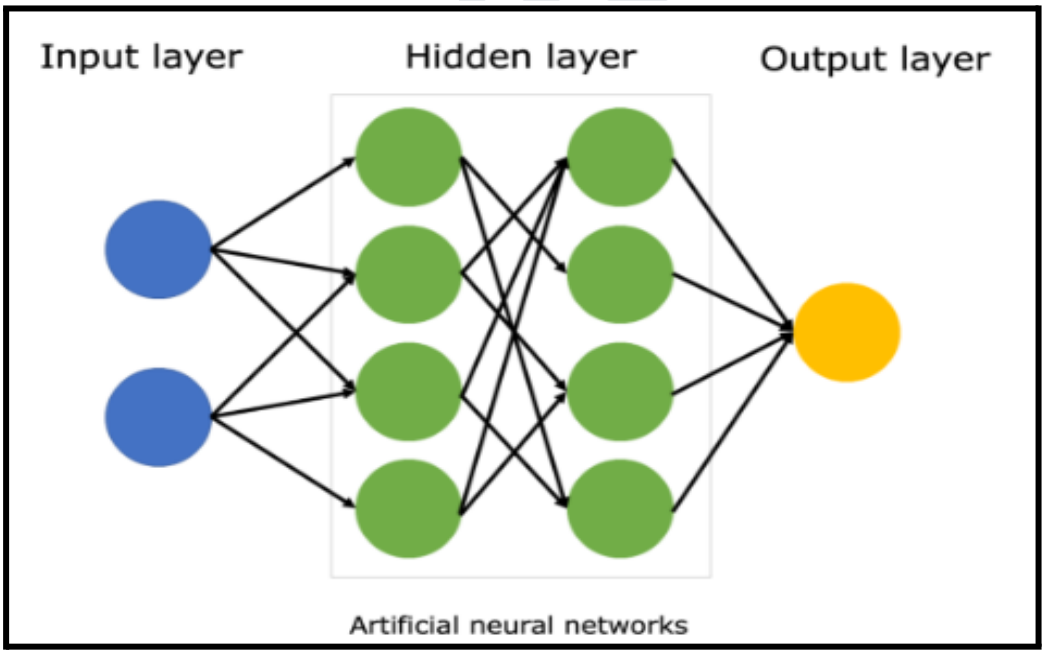
.

- In a neural network we don't tell the computer how to solve our problem. Instead, it learns from observational data, figuring out its own solution to the problem at hand.

LAYERS IN THE NEURAL NETWORK:

A neural network consists of three layers:

- **Input Layer:** The input layer is the first layer of the neural network where the features are added to the computer system and the weights initialise those features.
- **Hidden Layer:** In the hidden layer, the activation functions are used which helps generate the output.
- **Output Layer:** Produce the result for given inputs.



EXAMPLE USE CASE:

The Human Analogy

.

When you were a child, encountering new creatures for the first time, such as dogs and cats, you initially couldn't distinguish between them. Over time, through repeated exposure and guidance from others, you began to notice distinct features and patterns that differentiated these animals. For example, you learned that dogs often have specific traits like barking, a certain body shape, and a particular way of moving, while cats have different characteristics, such as meowing, a more slender body, and different movements. Learning from experience and identifying patterns is analogous to how neural networks operate.

Neural Networks and Learning

Exposure and Data Input:

- **Human Learning**: As a child, you repeatedly see various animals and are exposed to different features through your senses.
- **Neural Network**: The network is provided with a large dataset of images or data points, each labelled correctly (e.g…images labelled as 'dog' or 'cat').

Identifying Features:

.

- **Human Learning**: Over time, your brain starts to recognize patterns and features that differentiate dogs from cats.
- **Neural Network**: The network uses layers of interconnected nodes (neurons) to process the input data. Early layers might focus on simple features like edges and colours, while deeper layers identify more complex patterns and characteristics.

Learning and Adjustment:

- **Human Learning**: Through trial and error, and with feedback (e.g., someone correcting you), you adjust your understanding and improve your ability to distinguish between different animals.
- **Neural Network**: The network undergoes a training process using algorithms like backpropagation. It adjusts the weights of the connections between neurons based on the errors in its predictions. This is akin to the feedback you receive as a child, allowing the network to improve its accuracy over time.

Making Predictions:

- **Human Learning**: Eventually, you become proficient at identifying dogs and cats, even if they are in different poses or environments.
- **Neural Network**: After sufficient training, the network can accurately classify new, unseen images of dogs and cats, based on the patterns it has learned.

WORKING OF NEURAL NETWORK(MATHEMATICALLY)

So as discussed earlier, there are different layers in a neural network. Now the neural network's mathematical computation occurs in the neural network's hidden layer.

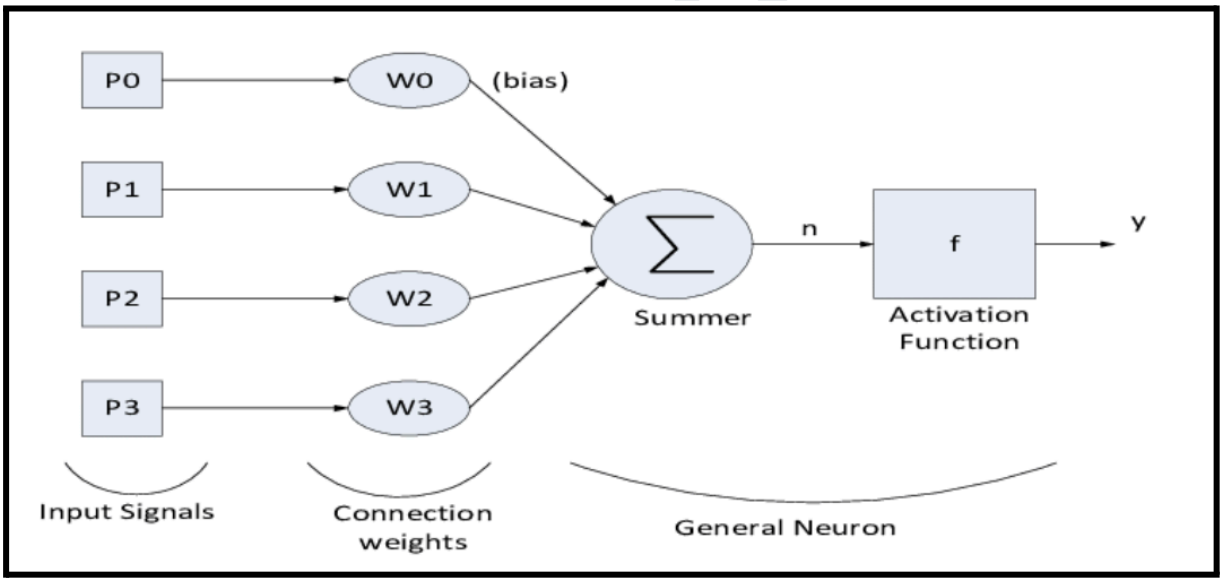
In the hidden layer, this process takes place in two steps:

1. Step1: Summing up the product of the weights and the features.

.

$$y_j = \sum_{\substack{i=1,\\ j=1}}^{n} w_{ij}\, x_j$$

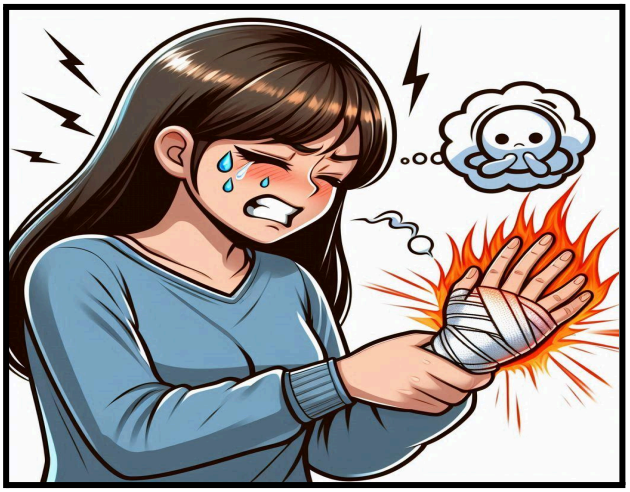
2. Step 2: Applying the activation function to the output of the summer i.e y for the above case.



UseCase Example:

Consider the following scenario: if you hold a hot object against your hand, a neuron in that specific area of your hand will fire. Your brain will immediately remove the hot object from your hands once they send information to it based on the circumstances.
The same thing occurs here as well. Every time we apply the activation function to the output obtained from the summation, in this case, "Y," the output is produced by the activation function. And the result determines what action is required.

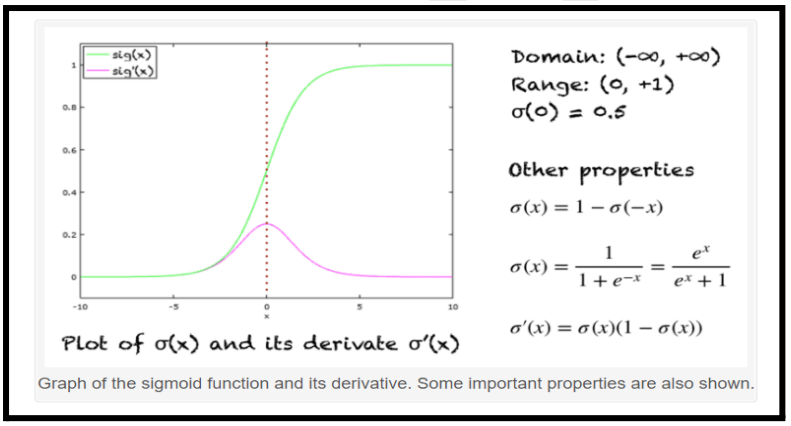
.



ACTIVATION FUNCTIONS:

1. SIGMOID FUNCTION:

A sigmoid function is any mathematical function whose graph has a characteristic S-shaped or sigmoid curve.

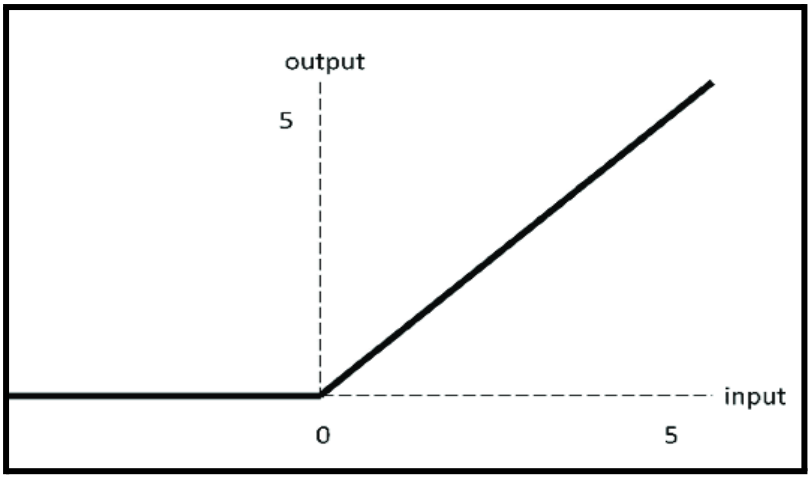


Graph of the sigmoid function and its derivative. Some important properties are also shown.

2.ReLU FUNCTION:

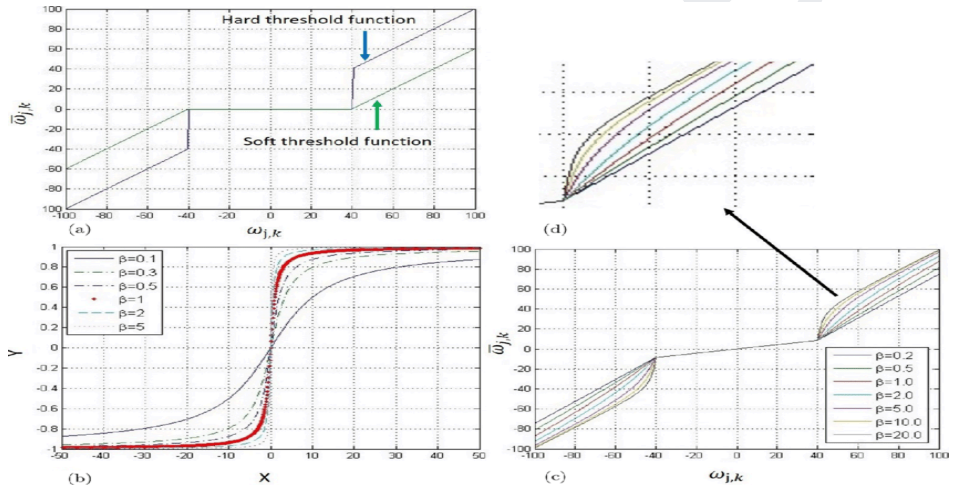
The rectified linear unit (ReLU) or rectifier activation function introduces the property of nonlinearity to a deep learning model and solves the vanishing gradients issue. It interprets the positive part of its argument. It is one of the most popular activation functions in deep learning.

.



3.THRESHOLD FUNCTION:

A threshold function is a function that takes the value 1 if a specified function of the arguments exceeds a given threshold and 0 otherwise.

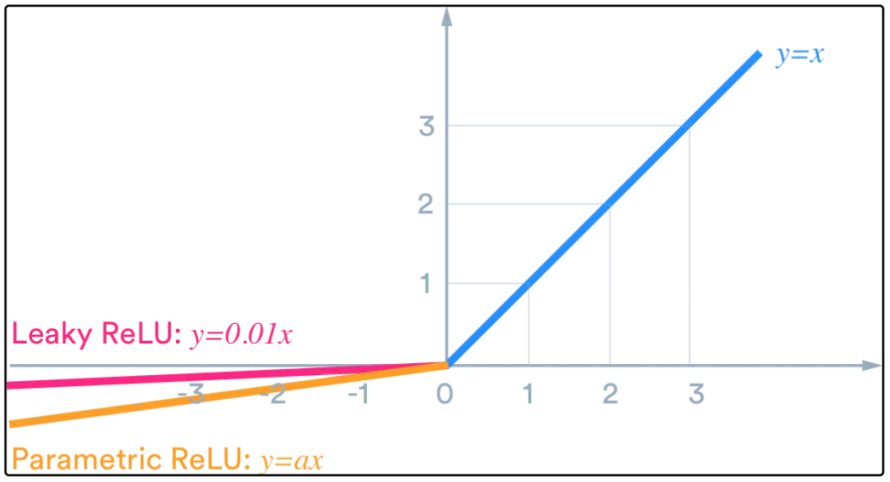


4.LEAKY RELU:

f(x)=max(0.01*x , x).

This function returns x if it receives any positive input, but for any negative value of x, it returns a really small value which is 0.01 times x. Thus it gives an output for negative values as well.

.



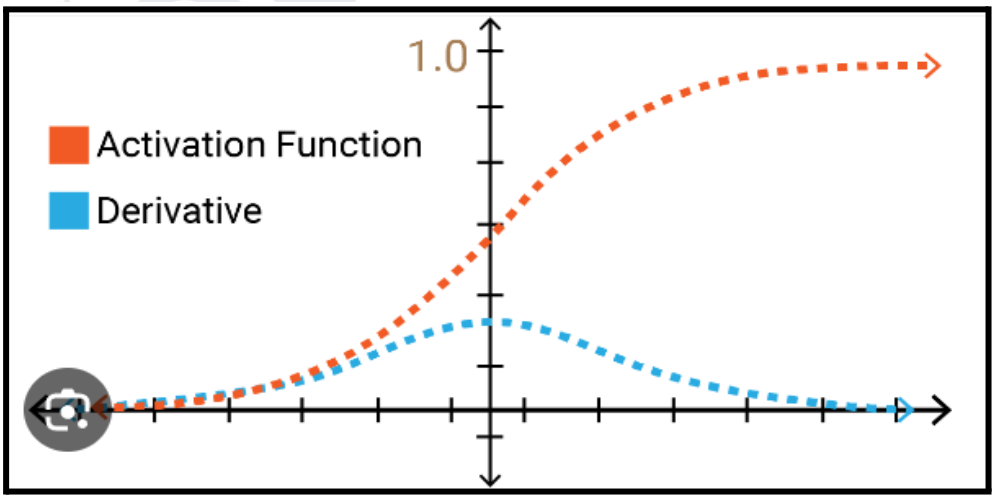
Leaky ReLU and Parametric ReLU

VANISHING GRADIENT PROBLEM:

Vanishing gradient problem is a phenomenon that occurs during the training of deep neural networks, where the gradients that are used to update the network become extremely small or "vanish" as they are back propagated from the output layers to the earlier layers.

Reason of this problem:

The sigmoid function is a function that varies from 0 to 1 but derivative of sigmoid function vary from 0 to 0.25.This causes the gradient to become smaller if there are many layers in the neural network. And if we want to update the value of weights by backpropagation, then we will not be able to do that.



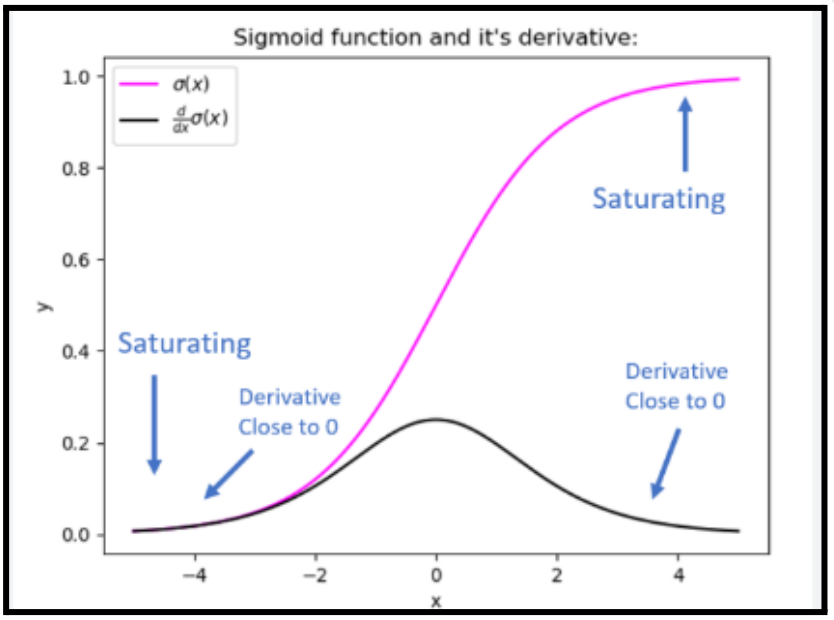
.

EXPLODING GRADIENT PROBLEM:

The exploding gradient problem is a challenge encountered during the training of deep neural networks, particularly in the context of gradient-based optimization methods such as backpropagation. This issue occurs when the gradients of the network's loss with respect to the parameters (weights) become excessively large.

Reason for this problem:

Due to the very high value of weights, the value of the derivation of the loss function becomes very high and if there are many layers then the derivative of the loss function becomes higher. Now if we update the value weight, we get the value in negative, due to which old and new values of weights vary a lot.Due to this reason the gradient descent will never converge and will explode after every backpropagation.



.

Refer to this for reading material:

INTRODUCTION TO DEEP LEARNING

BLOG 1: Deep Learning 1
BLOG 2: Deep Learning 2

NEURAL NETWORK

BLOG 1: Neural Network 1
BLOG 2: Neural Network 2

ACTIVATION FUNCTIONS

BLOG 1: Activation Functions

VANISH/EXPLODE GRADIENT PROBLEMS

BLOG1: Vanish/Explode Gradient problems
BLOG2: Vanish/Explode problems