Activation Functions

The activation function defines the output of a neuron or node given an input or set of input.It decides whether a neuron should be activated or not. This means that it will decide whether the neuron's input to the network is important or not in the process of prediction using simpler mathematical operations. Activation functions are also responsible for introducing non-linearity into the neural network, allowing for more complex decision boundaries and more accurate predictions. There are several types of activation functions, including Step, Sigmoid, ReLU, and Tanh.

Here is the characteristics of six of the most commonly used activation functions shown in this report; "Step, Sigmoid, Tanh, ReLU, ELU, and SELU. Each type has its own strengths and weaknesses, and the choice of activation function can significantly impact the performance of a neural network".

The binary activation function is the easiest. It's based on binary classifier the output is 0 if values are negatives else 1. See this activation function as a threshold in binary classification .The main advantage of the step activation function is that it is computationally efficient and produces clear results .The main problem is that the gradient of the step function is zero. This makes the step function not so useful since, during back-propagation. It loses any knowledge of gradients in input values. A value that is 10000000 times the threshold has the same influence as the threshold value. Also, ignores any input value just below threshold. Additionally, it cannot be used for multi-class classification.

The linear activation function, also known as "no activation," or "identity function" (multiplied x1.0), is where the activation is proportional to the input.The function doesn't do anything to the weighted sum of the input, it simply spits out the value it was given. However, a linear activation function has two major problems :It's not possible to use backpropagation as the derivative of the function is a constant and has no relation to the input x. All layers of the neural network will collapse into one if a linear activation function is used. No matter the number of layers in the neural network, the last layer will still be a linear function of the first layer. So, essentially, a linear activation function turns the neural network into just one layer. The linear activation function shown above is simply a linear regression model. Because of its limited power, this does not allow the model to create complex mappings between the network's inputs and outputs.

Non-linear activation functions solve the following limitations of linear activation functions:They allow backpropagation because now the derivative function would be related to the input, and it's possible to go back and understand which weights in the input neurons can provide a better prediction.They allow the stacking of multiple layers of neurons as the output would now be a non-linear combination of input passed through multiple layers. Any output can be represented as a functional computation in a neural network.

The Sigmoid function is a non-linear function that outputs values between 0 and 1. It is commonly used for models where we have to predict the probability as an output. Since probability of anything exists only between the range of 0 and 1, sigmoid is the right choice because of its range. It is suitable for both classification and regression tasks, as it provides a smooth transition from 0 to 1. In multiclass classification, it returns the probability output for each class. Another advantage of this function is that, when used with (-infinity, +infinity) as in the linear function, it returns a value in the range of (0, 1). This function has a few drawbacks, including the vanishing gradient problem. Incase This is not a zero-centered function; a zero-centered function is one where the function range has 0 in the middle.

The output of the tanh activation function is Zero centered; hence we can easily map the output values as strongly negative, neutral, or strongly positive.This type of activation function is most commonly used in classification problemsThis function is non-linear in nature, so it can easily backpropagate the errors. There is a similar drawback to the sigmoid function; it still has the vanishing gradient problem. It is computationally expensive (exponential in nature).

Since only a certain number of neurons are activated, the ReLU function is far more computationally efficient when compared to the sigmoid and tanh functions. ReLU accelerates the convergence of gradient descent towards the global minimum of the lose function due to its linear, non-saturating property. ReLU (rectified linear unit) activation functions are a type of continuous activation function that returns either 0 or the input value depending on whether the input is greater than or less than 0. This type of activation function is most commonly used in deep learning networks. ReLU output is not zerocentered; it decreases the efficiency of the neural network. During backpropagation, the weights' gradients will either be uniformly positive or uniformly negative.

- ELU is a strong alternative for f ReLU because of the following advantages:ELU becomes smooth slowly until its output equal to -α whereas RELU sharply smoothes.Avoids dead ReLU problem by introducing log curve for negative values of input. It helps the network nudge weights and biases in the right direction.ELU does not suffer from the problem of vanishing gradients and exploding gradients. Unlike ReLU, ELU does not suffer from the problems of dying neurons. Using ELU leads to a lower training time and higher accuracy in neural networks as compared to ReLU and its variants. It increases the computational time because of the exponential operation included no learning of the ‘a’ value takes place.

SELUs, or Scaled Exponential Linear Units, are activation functions that induce selfnormalization. SELU network neuronal activations automatically converge to a zero mean and unit variance. This function is similar to the ELU function, but it is scaled to ensure that the output of the entire layer remains constant. This helps to reduce the problem of vanishing gradients, as it prevents the output

from becoming too small. Compared to ReLUs, SELUs cannot die. SELUs learn faster and better than other activation functions without needing further processing. This is a relatively new activation function, so it is not yet used widely in practice.

Without an activation function is just a linear regression model as these functions actually do the non-linear computations to the input of a neural network making it capable to learn and perform more complex tasks. Thus, it is quite essential to study the derivatives and implementation of activation functions, also analyze the benefits and downsides for each activation function, for choosing the right type of activation function that could provide non-linearity and accuracy in a specific neural network model.

Necessity to use an appropriate activation function based on the specific task and dataset. Activation functions are useful because they add non-linearities into neural networks, allowing the neural networks to learn powerful operations. If the activation functions were to be removed from a feedforward neural network, the entire network could be re-factored to a simple linear operation or matrix transformation on its input, and it would no longer be capable of performing complex tasks such as image recognition.