**Name: Jannatul Ferdous Umama**

**Id:20-42616-1**

**Report on Activation Function**

To introduce non-linearities into the decision border of a neural network, activation functions are used. The purpose of adding nonlinearities to data is to mimic real-world circumstances. The data we deal with in real life is almost always nonlinear. The effectiveness of neural networks is due to this.

We'll now examine the most prevalent categories of activation functions utilized in deep learning. Python will be used to teach us how to implement.

**1. Step Activation Function:**

Step activation is employed in the perceptron network. This is frequently used in single-layer networks for converting to a binary (0 or 1) or bipolar output (-1 or 1). These go by the names Bipolar step function and Binary step function, respectively. If the input value is greater than a threshold value, the output in this scenario is 1 (a neuron will fire), otherwise it is 0 or -1 in the case of a bipolar step function (neuron will not fire).

**Advantages:**

* The step function is quick to use and effective in terms of computation.
* When we wish to output a clear decision border for binary classification issues, it is helpful.

**Disadvantages:**

* It can be challenging to apply the step function in some optimization techniques that rely on gradient information since it is not differentiable at zero.
* Only the good or negative outcome is disclosed; no information regarding the size of the input is given.

**2. Sigmoid Activation Function:**

One of the most utilized activation functions in deep learning is the sigmoid activation function. The sigmoid function's curve is S-shaped, as implied by its name. The sigmoid function changes values between 0 and 1.The sigmoid function has the following mathematical function:



function of sigmoid activation

The sigmoid derivative is:



**Advantages:**

* It is a smooth function that is simple to distinguish, the sigmoid activation function is helpful for optimization techniques that rely on gradient data.
* It is helpful for binary classification issues since it produces values between (0, 1) that can be read as probabilities.
* It is frequently utilized in conventional neural networks and, in some situations, can yield successful outcomes.

**Disadvantages:**

* The vanishing gradient problem, where the gradient gets very small as the input gets closer to the tails of the function, affects the sigmoid function and makes it challenging to train deep neural networks.
* When the input is extremes of either size, it tends to "saturate," which can cause the "gradient disappearing" problem and sluggish learning.

**3. Tanh Activation Function:**

There are two types of activation functions: linear and non-linear. Tangent hyperbolic is referred to as tanh. A non-linear activation function is tanh. It is an exponential function that is primarily utilized in hidden layers of multilayer neural networks.

**Advantages:**

* It is a smooth function that is simple to distinguish, the tanh activation function is helpful for optimization techniques that rely on gradient data.
* It produces values in the (-1, 1) range, which can be helpful for particular kinds of issues.
* It is frequently applied in classic neural networks' hidden layers, when the objective is to learn a nonlinear representation of the input data.

**Disadvantages:**

* The vanishing gradient problem, which also affects the sigmoid function, might affect the tanh function, making it challenging to train deep neural networks.
* In contemporary neural networks, it is not as frequently utilized as the sigmoid or ReLU activation functions.

**4. ReLU Activation Function:**

If the input, let's say x, is positive, the output of the rectified linear activation function (RELU), which is a piecewise linear function, will be x. Otherwise, it produces nothing.

ReLU is widely used today, but it has some issues. Consider the scenario where the neural network's backpropagation algorithm fails if the input is less than zero and the output is zero. Dying ReLU is the term used to describe this issue. Leaky ReLU, an improvised variation of ReLU, is what we use to solve this issue.

The mathematical representation of ReLU function is,



The derivative of ReLU is,



**Advantages:**

* Modern neural networks frequently use the ReLU activation function since it is simple to use and computationally effective.
* Compared to the sigmoid and tanh functions, it is more resistant to the vanishing gradient problem, making it a better choice for deep neural networks.
* It has been demonstrated to work well in a variety of applications, including natural language processing, image and audio recognition, and others.

**Disadvantages:**

* The "dying ReLU" issue can cause a significant number of the network's neurons to become dormant and produce zero values for all inputs in the ReLU function. This may occur if the learning rate is set too high or the weights are initialized with very low values.
* The ReLU function produces values in the [0, infinite] region, which can cause numerical instability and make comparing the outputs of several neurons challenging.

**5. ELU Activation Function:**

Exponential Linear Unit, sometimes known as ELU, is a function that tends to converge costs to zero more quickly and yield more precise results. ELU has a second alpha constant, which is positive, unlike other activation functions.

ELU and RELU are quite similar, however ELU uses negative inputs. For non-negative inputs, they are both in identity function form.

**Advantages:**

* Similar to the ReLU function but with a smooth curve that permits for negative values, the Exponential Linear Unit (ELU) activation function can be helpful for some types of situations.
* It can lessen the possibility of the "dying ReLU" problem and has been demonstrated to be beneficial in deep neural networks.
* In some cases, like speech recognition and picture classification, it has been demonstrated to enhance the performance of neural networks.

**Disadvantages:**

* The exponential factor, the ELU function requires more computation time than the ReLU function.
* It may not be required for simpler models because it is less frequently utilized than the ReLU function.

**6. SELU Activation Function:**

The activation functions known as SELUs, or Scaled Exponential Linear Units, cause self-normalization. Neuronal activations in the SELU network automatically converge to zero mean and unit variance.

Mathematically, it is expressed as:





**Advantages:**

* As a self-normalizing function, the Scaled Exponential Linear Unit (SELU) activation function may automatically normalize the activations of the preceding layer, enhancing the stability and convergence of the neural network.
* For a variety of deep learning tasks, it has been demonstrated to deliver state-of-the-art performance with less hyperparameter tuning than other activation functions.
* It can lessen the possibility of the "dying ReLU" issue and is especially beneficial for deep neural networks with numerous layers.

**Disadvantages:**

* Due to the usage of the exponential term, the SELU function is computationally more expensive than the ReLU and ELU functions.
* For simpler models, it might not be necessary, and if applied incorrectly, it could lead to worse performance.