Step function:

In machine learning, the step function is a simple activation function that returns 1 if the input is greater than or equal to 0 and 0 otherwise. It is an easily implementable threshold function with both advantages and disadvantages.

Advantages-Step function is a basic function that is straightforward to comprehend and apply in a neural network.

The step function generates a binary output, which may be beneficial in some situations, such as binary classification tasks.

The step function takes just a little amount of computing, making it computationally efficient.

Disadvantages-Non-differentiability: As the step function is not differentiable, it cannot be employed in backpropagation methods requiring the computation of derivatives.

Vanishing Gradient: The step function has a gradient of zero everywhere except at the discontinuity point, which may pose issues when training deep neural networks. The vanishing gradient issue may hinder or perhaps prohibit neural network convergence.

The output of the step function may be unstable if the input to the function is noisy or subject to tiny oscillations. This might cause the network output to oscillate between 0 and 1, making it difficult to utilise for certain applications.

Overall, the step function is a basic activation function that may be beneficial for certain applications, but it is inappropriate for many contemporary deep learning systems due to its limitations. ReLU, ELU, and SELU are the recommended activation functions because to their differentiability and ability to avoid vanishing gradients.

Sigmoid function:

In machine learning, the sigmoid function is a prominent activation function that translates every input value to a value between 0 and 1. It has both benefits and drawbacks.

Advantages:

Nonlinearity: Being a nonlinear function, the sigmoid function is helpful for representing nonlinear data interactions.

Smoothness: The sigmoid function is a smooth, differentiable function at every point. This makes it handy for optimization procedures, such as gradient descent, that require the computation of derivatives.

The sigmoid function may be interpreted probabilistically as a probability distribution, making it helpful for probabilistic models and binary classification issues.

Disadvantage-

Vanishing Gradient: The sigmoid function has a tiny gradient in the region when its output approaches 0 or 1. This might result in the vanishing gradient issue during backpropagation, in which the gradient decreases as it propagates through the layers of a neural network, making it difficult to train deep neural networks.

The sigmoid function may saturate when the input is either very big or extremely little, resulting in gradients that are near to zero. This might result in the network ceasing to learn since the weights are no longer being updated.

Not zero-centered: As the sigmoid function is not zero-centered, it might be challenging to train a neural network since the mean of the activations in a layer can move away from zero.

In general, the sigmoid function is a helpful activation function in some circumstances, but its drawbacks might restrict its use in deep learning networks. Several activation functions, including ReLU, ELU, and SELU, have been created to solve some of the sigmoid function's drawbacks.

Tanh function-

The tanh function is a widely used activation function in machine learning that converts any input value to a value between -1 and 1. It has both benefits and drawbacks.

Advantage-

Nonlinearity: Since the tanh function is nonlinear, it is excellent for representing nonlinear data relationships.

Smoothness: The tanh function is a smooth, differentiable function at every point. This makes it handy for optimization procedures, such as gradient descent, that require the computation of derivatives.

The tanh function is zero-centered, which may aid in training a neural network by preventing the mean of the activations in a layer from deviating from zero.

Disadvantage-

The tanh function has a negligible gradient in the region when its output is close to -1 or 1. This might result in the vanishing gradient issue during backpropagation, when the gradient decreases as it propagates through the layers of a neural network, making it challenging to train deep neural networks.

When the input is extremely big or very little, the tanh function might saturate, resulting in gradients that are near to zero. This may prevent the network from learning, since the weights will no longer be updated.

The tanh function takes much more computation than other activation functions, such as the ReLU function.

In general, the tanh function is a helpful activation function in some circumstances, but its drawbacks might restrict its use in deep learning frameworks. To overcome some of the shortcomings of the tanh function, other activation functions, such as ReLU, ELU, and SELU, have been created.

Relu-

In machine learning, the Rectified Linear Unit (ReLU) function is a prominent activation function that transfers any input value to a value between 0 and infinity. It has both benefits and drawbacks.

Advantages:

Non-linearity: Being a nonlinear function, the ReLU function is helpful for representing nonlinear data relationships.

Sparsity: The ReLU function may provide sparse representations in which the majority of activations in a given layer are zero. This may assist in reducing overfitting and enhancing generalisation.

The ReLU function is computationally efficient due to the fact that it needs just basic thresholding operations.

The ReLU function is immune to the vanishing gradient issue since the gradient is always 1 for inputs higher than 0. This facilitates the training of deep neural networks.

Disadvantage-

The ReLU function may create neurons whose output is always 0 (dead neurons). This may occur when the input to the ReLU function is negative, causing a significant chunk of the network to be idle and reducing its ability to represent complicated data connections.

Non-zero mean: Since the ReLU function is not zero-centered, it might be challenging to train a neural network because the mean of the activations in a layer can move away from zero.

Overall, the ReLU function is a helpful activation function in many scenarios; nevertheless, its drawbacks may restrict its use in some deep learning systems. Alternative activation functions, such Leaky ReLU, ELU, and SELU, have been created to solve some of the ReLU function's drawbacks.

Selu-

Scaled Exponential Linear Unit (SELU) is a relatively recent activation function used in machine learning that is intended to solve some of the limitations of previous activation functions. It has both benefits and drawbacks.

Advantages:

Self-normalization: The SELU function is intended to be self-normalizing, which means that the mean and variance of the output of each layer of a neural network will remain constant regardless of the network's depth. This may help limit the chance of gradients disappearing or exploding, which can make training deep neural networks challenging.

Nonlinearity: Since the SELU function is nonlinear, it is excellent for representing nonlinear data relationships.

Continuous differentiation: The SELU function is a continuously differentiable function at all locations. This makes it handy for optimization procedures, such as gradient descent, that require the computation of derivatives.

Zero-centered: The SELU function is zero-centered, which may aid in the training of neural networks by preventing the mean activations in a layer from deviating from zero.

Disadvantage:

Sensitivity to initialization: The SELU function is sensitive to the initialization of a neural network's weights and biases. This implies that proper setup is essential to maximise the SELU function's advantages. The SELU function is meant to perform most effectively with feedforward neural networks that use fully-connected layers with equal input and output dimensions. It may not perform as well with other neural network types or layer combinations.

Overall, the SELU function is a promising activation function that has shown excellent performance in a variety of deep learning applications. Yet, its sensitivity to initialization and restricted application may limit its use in some circumstances.

Elu-

Similar to the ReLU function, the Exponential Linear Unit (ELU) function is an activation function used in machine learning. It has both benefits and drawbacks.

Advantages:

Nonlinearity: Since the ELU function is nonlinear, it is excellent for representing nonlinear data relationships.

Smoothness: The ELU function is a smooth, differentiable function at every point. This makes it handy for optimization procedures, such as gradient descent, that require the computation of derivatives.

Negative values: The ELU function permits negative values, which may avoid the death of neurons and enhance the network's ability to simulate complicated data connections.

Zero-centered: The ELU function is nearly zero-centered, which may aid in the training of neural networks by keeping the mean activations in a layer from deviating from zero.

Disadvantages:

Complexity of computation: The ELU function is more computationally intensive than the ReLU function because exponential functions must be calculated.

The ELU function is sensitive to the selection of hyperparameters such as the alpha parameter, which defines the function's negative saturation value. Improper selection of hyperparameters may result in subpar neural network performance.

In general, the ELU function is a useful activation function that has several benefits over the ReLU function; nevertheless, its computational complexity and hyperparameter sensitivity may restrict its use in some circumstances.