Name: MD NAFIS ULFAT

ID:20-42248-1

Section: B

Activation Function:

Activation function checkpoint is a technique in neural networks that involves using different activation functions in each layer to prevent the vanishing gradient problem during training, leading to faster convergence and better performance.

There are many activation functions available, but some of the commonly used activation functions in deep learning include sigmoid, tanh, Relu, Step, Elu, Selu, .

Sigmoid:

Sigmoid is an activation function that maps any input to a value between 0 and 1.

Sigmoid function:

f(x) = 1 / (1 + e^-x)

Here are three advantages and three disadvantages of using sigmoid as an activation function:

Advantages:

- Sigmoid is a smooth function that is differentiable, making it easy to use in optimization algorithms.
- Sigmoid is useful in binary classification problems where the output should be a probability value between 0 and 1.
- Sigmoid can be used to normalize inputs to a neural network, ensuring that all inputs are within a certain range.

Disadvantages:

- Sigmoid has a vanishing gradient problem, which means that the derivative of the function approaches zero as the input moves away from zero, leading to slow learning.
- Sigmoid is not zero-cantered, which can make it difficult to optimize in deep neural networks.
- Sigmoid can suffer from saturation problems when the input is too large or too small, which can lead to slow learning or incorrect predictions.

Tanh:

The Tanh (hyperbolic tangent) function is an activation function that maps any input to a value between -1 and 1.

Hyperbolic tangent (tanh) function:

f(x) = (e^x - e^-x) / (e^x + e^-x)

Here are three advantages and three disadvantages of using Tanh as an activation function:

Advantages:

- Tanh is a zero-cantered function, which can make it easier to optimize in deep neural networks compared to the sigmoid function.
- Tanh is a smooth function that is differentiable, making it easy to use in optimization algorithms.
- Tanh can be used to normalize inputs to a neural network, ensuring that all inputs are within a certain range.

Disadvantages:

- Tanh has a vanishing gradient problem, which means that the derivative of the function approaches zero as the input moves away from zero, leading to slow learning.
- Tanh suffers from saturation problems when the input is too large or too small, which can lead to slow learning or incorrect predictions.
- Tanh is symmetric around the origin, which can lead to the vanishing gradient problem becoming more severe as the network depth increases.

ReLU:

The Rectified Linear Unit (ReLU) function is an activation function that returns the input if it is positive and returns zero if it is negative.

Rectified Linear Unit (ReLU) function:

f(x) = max(0, x)

Here are three advantages and three disadvantages of using ReLU as an activation function:

Advantages:

- ReLU is a simple function that is easy to implement and efficient to compute.
- ReLU does not suffer from saturation problems like the sigmoid and TanH functions, which can lead to faster learning and better convergence.
- ReLU helps to solve the vanishing gradient problem by allowing gradients to flow more easily through the network.

Disadvantages:

- ReLU can suffer from the "dying ReLU" problem, where neurons can become permanently inactive if their input falls below zero, leading to a dead network.
- ReLU is not a smooth function and is not differentiable at the origin, which can lead to optimization difficulties.
- ReLU can lead to the issue of "gradient explosion" if the learning rate is too high, where gradients become too large and cause the model to diverge.

Step:

The Step function is an activation function that returns 1 if the input is greater than zero, and returns 0 otherwise.

Step function:

f(x) = 1, if x >= 0

f(x) = 0, if x < 0

Here are three advantages and three disadvantages of using Step as an activation function:

Advantages:

- Step is a simple function that is easy to implement and efficient to compute.
- Step can be used to model binary classification problems where the output must be either 0 or 1.
- Step can be used as a threshold function to determine whether an input is above or below a certain value.

Disadvantages:

- Step is not a smooth function and is not differentiable at the origin, which can lead to optimization difficulties.
- Step does not allow gradients to flow through the network, which can make learning difficult.
- Step can lead to issues with "gradient explosion" or "gradient vanishing" if the learning rate is too high or too low, respectively.

ELU:

The Exponential Linear Unit (ELU) is an activation function that has a smooth curve and is similar to the Rectified Linear Unit (ReLU) function.

Exponential Linear Unit (ELU) function:

f(x) = x, if x > 0

f(x) = alpha * (e^x - 1), if x <= 0

where alpha is a hyperparameter typically set to 1.0.

Here are three advantages and three disadvantages of using ELU as an activation function:

Advantages:

- ELU is a smooth function that is differentiable at all points, which makes it easier to optimize than functions like Step or ReLU.
- ELU has a range of negative values, which allows it to capture negative input values more accurately than ReLU.
- ELU has been shown to improve the performance of deep neural networks, especially in tasks where there is a high level of noise or uncertainty in the data.

Disadvantages:

- ELU can be computationally expensive to compute, especially when compared to simpler activation functions like Step or ReLU.
- ELU may not always lead to better performance than ReLU, and may not be necessary for all tasks.
- The exponential function used in ELU can result in a vanishing gradient problem, similar to the sigmoid function, which can make optimization difficult.

SELU:

The Scaled Exponential Linear Unit (SELU) is an activation function that is similar to ELU but has additional scaling properties.

Scaled Exponential Linear Unit (SELU) function:

f(x) = scale * (x, if x > 0

f(x) = scale * alpha * (e^x - 1), if x <= 0

where alpha and scale are hyperparameters set to 1.67326 and 1.0507, respectively.

Here are three advantages and three disadvantages of using SELU as an activation function:

Advantages:

- SELU has been shown to improve the performance of deep neural networks, especially when compared to other activation functions like ReLU or ELU.
- SELU can automatically adjust the scale of the output values of each neuron, which helps to reduce the impact of initialization and normalization on model performance.
- SELU can help to mitigate the vanishing gradient problem that can occur in deep neural networks, which can help to improve training speed and accuracy.

Disadvantages:

- SELU can be computationally expensive to compute, especially when compared to simpler activation functions like Step or ReLU.
- SELU requires careful initialization of network weights in order to function properly, which can make model training more difficult.
- SELU may not always lead to better performance than other activation functions, especially in shallow neural networks or when the input data has a limited range of values.