Part – A

1. Explain Activation functions

- Define & compare the following activation functions: Sigmoid, ReLU, Tanh and Leaky ReLU.

Ans:

Sigmoid function:

A sigmoid function is a mathematical function with a characteristic "S" shaped curve or sigmoid curve. It transforms any value in the domain $(-\infty, \infty)$ to a number between 0 and 1. The sigmoid function formula is:

$$sigma(x) = \frac{1}{1+e^{(-x)}}$$

ReLU function:

The rectified linear activation function or ReLU for short is a piecewise linear linear function that will output the input indirectly if it is positive, otherwise it will output zero. It has become the default activation function for many types of neural networks because a model that uses it is easier to train and often achieve better performance. The ReLU function formula is:

$$f(x) = max(0, x)$$

Tanh function:

The tanh function, short for hyperbolic tangent function, is another commonly used activation function in neural networks. It maps any real-valued number into a value between -1 and 1. This function is similar to sigmoid function but offers some advantages that make it more suitable for certain applications. Tanh function formula is:

$$\tanh(x) = \frac{e^{x} - e^{(-x)}}{e^{x} + e^{(-x)}}$$

Leaky ReLU function:

The leaky ReLU function formula is:

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ a \cdot x & \text{if } x \leq 0 \end{cases}$$

Leaky ReLU introduces a small slope (a) for negative inputs, preventing neurons from dying completely. It addresses the issue of dying ReLU by allowing a small, non-zero gradient for negative inputs. This ensures that neurons with negative inputs still receive some signal during backpropagation.

Comparision table:

The comparision among the activation functions are given below:

Sigmoid	ReLU	Tanh	Leaky ReLU
This function output range in (0,1).	This function output range is (-1,1).	This function output range is $[0,\infty)$.	This function output range is $(-\infty,\infty)$.
It is not zero-centered function.	It is zero-centered function.	It is not zero-centered function.	It is not zero-centered function.
It is used in output layer.	It is used in hidden layer.	It is used in hidden layer.	It is used in hidden layer.
It's computation cost is high.	It's computation cost is high.	It's computation cost is low.	It's computation cost is low.

- Discuss their usecases & limitations.

Ans:

Sigmoid function:

Usecase:

- It is used as an activation function in for neurons in artificial neural network.
- For modeling the probability of a binary outcome, it is used in logistic regression.
- To adjust the intensity values for enhancing the contrast between dark & light regions.

Limitation:

- The sigmoid function can cause gradients in a neural network. This can make the model less accurate.
- It is not zero-centered function.
- The function can sometimes return small values as outputs, which can cause the vanishing gradient problem.

ReLU function:

Usecase:

- It is simple and computationally efficient.
- It helps to mitigate vanishing gradient problem.
- It encourages sparse representation.

Limitation:

- It suffers from 'dying ReLU' problem.
- It decreases the ability of the model to fit or train from the data properly.

Tanh function:

Usecase:

- It increases the flexibility.
- It increases power of neural networks to model complex and nuanced data.

Limitation:

- It causes vanishing gradient problem.

- It leads to slow learning or convergence issues.

Leaky ReLU function:

Usecase:

- It can speed up the learning process of deep learning neural networks.
- It is less sensitive than the traditional ReLU.
- It solves the 'dying ReLU' problem.

Limitation

- It may introduce noise in certain applications.
- It allows small slope for negative inputs.
- It can be problematic sometimes for the model.

2. Discuss the optimization Algorithms.

- Compare SGD, Adam, RMSprop.

Ans:

SGD:

SGD stands for stochastic gradient descent. It is like a smart shortcut for machine learning algorithms to find the best settings quickly. It is an iterative optimization algorithm widely used in machine learning and deep learning applications to find the model parameters that correspond to the best fit between predicted and actual outputs.
The formula of SGD is:

$$\theta = \theta - \eta \cdot \nabla L(\theta)$$

where,

η = learning rate

$\nabla L(\theta)$ = Gradient of the loss function with respect to the parameters.

Key features:

i)Simple and commonly applied in optimization task.
ii)Needs careful adjustment of the learning rate to achieve optimal performance.
iii)Prone to getting stuck in local minima or saddle points.

Adams:

Adam stands for Adaptive Moment Estimation. It is an optimization algorithm used in machine learning & deep learning. It combines the main ideas from both optimization techniques: momentum and RMSprop.

Key features:

i)It combines momentum & adaptive learning rate.
ii)It is well suited for sparse gradient.

RMSprop:

RMSprop stands for Root Mean Square Propagation. It modifies the learning rate for each parameter according to the size of its gradient, utilizing a moving average of squared gradients with exponential decay. It ensures that the learning rate is adapted for each weight in the model, allowing for more nuanced updates.

Key features:

i)It helps to stabilize training by normalizing gradients.

ii)It adapts the learning rate for each parameter.

iii)It is effective for non-stationary problems.

- Explain how learning rate impacts model training and how it is addressed in modern optimizers.

Ans:

Impact of learning rate in model training:

i) In case of too high learning rate, the model might take too large steps during optimization, it causes overshooting. It can become unstable, with the loss function.

ii)In case of too low learning rate, the model takes very small steps which results slow progress. It has risk of getting stuck.

Addressing learning rate issues in modern optimizers:

The techniques that are used by the modern optimizers are given below:

a)Learning rate schedules:

Many optimizers use learning rate schedules to adjust the learning rate during training. They are:

i)Step decay

ii)Exponential decay

iii)Cosine annealing

b)Adaptive learning rates:

It helps to avoid the need for manual tuning of a global learning rate. Some popular adaptive optimizers include:

i)Adagrad

ii)RMSprop

iii)Adam

c)Learning rate finder:

A technique known as the learning rate finder is employed. This involves performing a brief training session with various learning rates to identify the optimal one, where loss function shows the most significant decrease.