

1) Explain the architecture of Spark.

Components of Spark Architecture:

Driver Program

Cluster Manager

Worker Node

When a Driver Program in the Apache Spark architecture executes, it calls the real program of an application and creates a Spark Context. The Driver includes several other components, including DAG Scheduler, Task Scheduler, Backend Scheduler, and Block Manager, all of which are responsible for translating user-written code into jobs that are actually executed on the cluster.

The cluster Manager manages the execution of various jobs in the cluster. Spark Driver works in conjunction with the Cluster Manager to control the execution of various other jobs. The Cluster Manager does the task of allocating resources for the job. Many worker nodes can be used to process an RDD created in Spark Context and the results can also be cached.

The executor is in charge of carrying out these duties. The life span of executors is the same as that of Spark Application.

2) Explain activation function.

The activation function decides whether a neuron should be activated or not by calculating the weighted sum and further adding bias to it. The purpose of the activation function is to introduce non-linearity into the output of a neuron.

A neural network without an activation function is essentially just a linear regression model. The activation function does the non-linear transformation to the input making it capable of learning and performing more complex tasks.

3) List different types of activation function with their formula.

Step Function:

Simplest kind of activation function.

$$f(x) = 1; \text{ if } x \geq 0$$

$$f(x) = 0; \text{ if } x < 0$$

Sigmoid Function:

S shaped Function

$$f(x) = 1/(1 + e^{-x})$$

ReLU:

Rectified Linear Unit

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

Leaky ReLU:

Improved version of ReLu

$f(x) = ax; x < 0$

$f(x) = x; \text{ otherwise}$

4) Explain Hybrid Inheritance with Code.

Inheritance consisting of multiple types of inheritance is called Hybrid Inheritance.

#Hybrid Inheritance:

```
class Animal:
    def eat(self):
        print("Eating")

class Mammal(Animal):
    def sleep(self):
        print("Sleeping")

class Dog(Mammal, Animal):
    def bark(self):
        print("Barking")

obj = Dog()
obj.sleep()
obj.eat()
obj.bark()
```

5) Explain Neural Networks.

Neural networks extract identifying features from data, lacking pre-programmed understanding. Network components include neurons, connections, weights, biases, propagation functions and a learning rule. Neurons receive inputs, governed by threshold and activation functions. Connections involve weights and biases regulating information transfer. Learning, adjusting weights and biases, occurs in 3 stages: input computations, output generation, and iterative refinement enhancing the network's proficiency in diverse tasks.