Q1 An Activation Function decides whether a neuron should be activated or not. This means that it will decide whether the neuron's input to the network is important or not in the process of prediction using simpler mathematical operations.

Q2 These functions are mathematical operations in neural networks that introduce non-linearity. Common examples include Sigmoid, Tanh, ReLU, Leaky ReLU, and ELU.

Q3 Activation function helps in better training, learning process and, better generalizing capability. It controls the activation of every unit in the layer by working on weight and bias sum. This process incorporates the non-linearity in output of any neuron. The weights are updated based on the error in the output.

The activation function also aids in the normalization of any input's output in the range between 1 to -1. Because the neural network is occasionally trained on millions of data points, the activation function must be efficient and should reduce the computation time.

Q4 the advantages of sigmoid activation function:- The main reason why we use sigmoid function is because it exists between (0 to 1). Therefore, it is especially used for models where we have to predict the probability as an output.Since probability of anything exists only between the range of 0 and 1, sigmoid is the right choice.

Disadvantages of sgmoid function:-It can produce output values close to 0 or 1, which can cause problems with the optimization algorithm.The gradient of the sigmoid function becomes very small near the output values of 0 or 1, which makes it difficult for the optimization algorithm to adjust the weights and biases of the neurons.

Q5 The Rectified Linear Unit is the most commonly used activation function in deep learning models. The function returns 0 if it receives any negative input, but for any positive value x it returns that value back. So it can be written as $f(x)=max(0,x)$. The model trained with ReLU converged quickly and thus takes much less time when compared to models trained on the Sigmoid function. We can clearly see overfitting in the model trained with ReLU. This is due to the quick convergence. The model performance is significantly better when trained with ReLU.

Q6 One major benefit is the reduced likelihood of the gradient to vanish. This arises when $a>0$ .In this regime the gradient has a constant value. In contrast, the gradient of sigmoids becomes increasingly small as the absolute value of x increases. The constant gradient of ReLUs results in faster learning. The other benefit of ReLUs is sparsity. Sparsity arises when $a\leq0$. The more such units that exist in a layer the more sparse the resulting representation. Sigmoids on the other hand are always likely to generate some non-zero value resulting in dense representations. Sparse representations seem to be more beneficial than dense representations.

Q7 Leaky Rectified Linear Unit, or Leaky ReLU, is a type of activation function based on a ReLU, but it has a small slope for negative values instead of a flat slope. The slope coefficient is determined before training, i.e. it is not learnt during training. Because ReLU is known for vanishing gradients, since any values less than zero are mapped to zero. This is true regardless of the number of layers. LeakyReLU on the other hand, maps the values less than zeros to a very small positive number. This prevents vanishing gradient from occurring.

Q8 The softmax activation function simplifies this for you by making the neural network's outputs easier to interpret! The softmax activation function transforms the raw outputs of the neural network into a vector of probabilities, essentially a probability distribution over the input classes. The softmax function is a function that turns a vector of K real values into a vector of K real values that sum to 1.The output of the function is always between 0 and 1, which can be used as a probability score. The input can be positive or negative but the output is always a positive value bounded by 0 and 1.

Q9 We observe that the gradient of tanh is four times greater than the gradient of the sigmoid function. This means that using the tanh activation function results in higher values of gradient during training and higher updates in the weights of the network.tanh is also like logistic sigmoid but better. The range of the tanh function is from (-1 to 1). tanh is also sigmoidal (s - shaped). The advantage is that the negative inputs will be mapped strongly negative and the zero inputs will be mapped near zero in the tanh graph.

In [ ]: