◾ Vanishing/Exploding gradient descent problem:-

➢ The vanishing gradient problem is an issue that can occur when training deep neural networks. It occurs when the gradients of the loss function with respect to the weights of the network become extremely small, making it difficult for the network to learn.

◾ Karan ?

- The use of certain activation functions, such as the sigmoid function.
- The depth of the network. The more layers a network has, the more difficult it can be for gradients to flow through the network.
- The scale of the network weights. If the weights are initialized to be too large, the gradients can quickly become too small.

◾ Here is a simple example of the vanishing gradient problem:

➢ Imagine that we have a two-layer neural network with the following weights:

w1 = 10

w2 = 10

➢ We also have a loss function that we want to minimize. The gradient of the loss function with respect to the weights of the network is given by the following equations:

dw1 = w2 * dl/dw2

dw2 = dl/dw2

,where `dl/dw2` is the gradient of the loss function with respect to the weight `w2`.

➢ Now, suppose that the loss function is large and the gradient `dl/dw2` is equal to 10. This means that the gradient `dw1` will be equal to 100. However, if the weight `w2` is small, such as 0.1, then the gradient `dw1` will be much smaller, such as 10.

➢ This process continues as we backpropagate the gradients through the network. The gradients will become progressively smaller as we move closer to the input layer. This is because the gradients are multiplied by the weights of the previous layers.

➢ If the gradients become too small, then the weights of the network will not be updated effectively. This can make it difficult for the network to learn.

➢ **There are a number of techniques that can be used to mitigate the vanishing gradient problem, including:**

- Using activation functions that have a non-zero gradient at zero, such as the rectified linear unit (ReLU) function.
- Using weight initialization techniques that initialize the weights to be small.
- Using techniques such as batch normalization and dropout to stabilize the training process.

Conclusion:

➢ The vanishing gradient problem is a challenge that can occur when training deep neural networks. However, there are a number of techniques that can be used to mitigate the problem and train successful deep learning models.

(Hami sanga several layer ko neural network xa , ani training ko bela hamle tesla (neural network kaa) parameters haru(i.e weights,bias…) adjust garna parni hunxa , jo chahi gradients ko help bata hunxa .

Ani vanishing gradient problem occurs when these gradients becomes too small while training ,jasle garda ti parameters haru bistarai bistarai badhxan, jasle garda optimal solution maa pugna dherai nai steps lagxa.
As a result model le teti easily / effectively learn garna sakdaina.

This problem typically happens when you have many layers and certain activation functions (like sigmoid or tanh) that squash values into a limited range. The gradients can get smaller and smaller as they move backward through these functions, making it hard for the network to train properly.)

➢ In contrast, the exploding gradient problem occurs when the gradients become excessively large during backpropagation. This can lead to numerical instability during training, causing weight updates to be so large that the network's parameters diverge, and the training process becomes unstable.

◾ How to solve this problem:-

- Changing how we initialize weights
- Using non saturating activation functions
- Batch normalization
- Gradient clipping

For better understanding this topic go through the links below:-

- https://www.youtube.com/watch?v=2f_45VzKEfE&list=PLM8lYG2MzHmQn55ii0duXdO9QSoDF5myF&index=19

- https://www.youtube.com/watch?v=uCrevbBh0zM&list=PLKnIA16_RmvYuZauWaPlRTC54KxSNLtNn&index=20