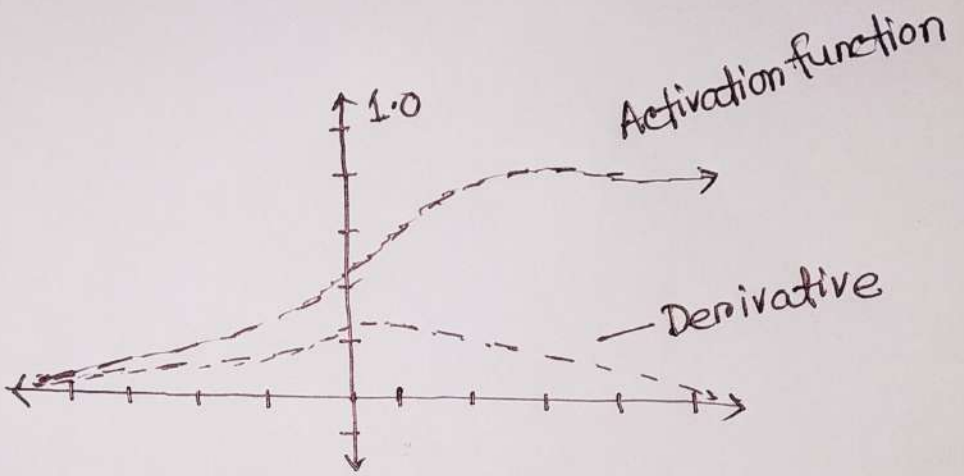
# Topic: Vanishing gradient problem.

The Vanishing gradient problem is an affair that occurs when training deep neural networks, where the gradients used to update the network become very disappear as they are transferred from the output layer to previous layer.




Training Neural Network-

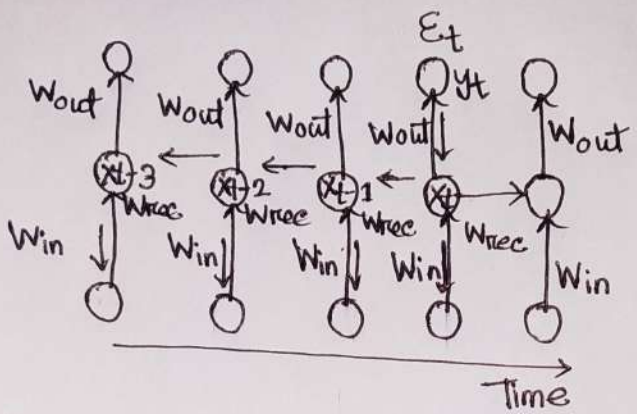
1. Weight optimization
2. Backpropagation
3. optimization landscape exploration
4. Efficiency and Scalability
5. Generalization and learning

Impact of Vanishing gradient problem on Deep neural network-

1. Limited learning capacity
2. Difficulty in capturing the long-term dependencies
3. Slow convergence and training instability
4. Preferential learning in shallow layers
5. Architectural design consideration

Why it is significant –

The Vanishing gradient problem causes the gradient to shrink. But if a gradient is small, it won't be possible to effectively update the weights and biases of the initial layers with each training session.




$$\frac{\delta \varepsilon}{\delta \theta} = \sum_{1 \le t \le T} \frac{\delta \varepsilon_t}{\delta \theta}$$

$$\Rightarrow \frac{\delta \varepsilon_t}{\delta \theta} = \sum_{1 \le k \le t} \left( \frac{\delta \varepsilon_t}{\delta x_t} \frac{\delta x_t}{\delta x_k} \frac{\delta^+ x_k}{\delta \theta} \right)$$

So, finally

$$\frac{\delta x_t}{\delta x_k} = \prod_{t \ge i \ge k} \frac{\delta x_i}{\delta x_{i-1}} = \prod_{t \ge i \ge k} W_{rec}^T \, diag\left(\sigma'(x_{i-1})\right)$$

$W_{rec} \sim$ Small → Vanishing

$W_{rec} \sim$ Large → Exploding

In this case we can – (Vanishing)

I. Initilize Weights
II. Echo State Networks designed
III. Long-Short Term Memory Networks (LSTMs)

In this case we can – (Exploding)

I. Truncated Backpropagation
II. Penalties
III. Gradient clipping.

Source –

→ Superdatascience.com/blog
→ engati.com/glossary
→ kdnuggets.com/2022