Optimation in Deep Learning

- In Deep Learning, with the help of loss function, the performance of the model evaluated.
- This loss is used to train the network so that it performs Better.
- Essentially, we try to minimize the loss function.
- Lower loss means the model performs better
- The process of minimizing any mathematical function is called optimization.
- optimers are algorithms used to change the feature of neural network such as weight and learning rate so that the loss is reduced
- The Goal of an optimizer is to minimize the objective function.

Need for optimization

- ~~Presence~~ Presence of Local minima reduces the model performance
- To minimize the loss value (Training error)
- To select appropriate weight values and other associated model parameters.

Types of optimization

1. Gradient descent

- it starts with some coefficients seen
- it moves towards lower weight and updates the values of coefficient and repeat until the local min is reach

Disadvantages

- Expensive to calculate a gradient if the size of the data is huge
- Not suitable for Non-convex function

2. Stochastic Gradient Descent :

- Instead of taking the whole data set for each iteration randomly select batches of the data
- Select the initial parameter w and learning data
- Randomly shuffle the data and each iteration to reach the approximate minimum.

---

fast is than GD but the computation

- sinceonly few batches are

Disadvantage ↑

3. Stochastic Gradient descent with Momentum :

- Since SGD is a noisy path we are going for SGD with momentum
- momentum helps in fast convergance of the loss function
- SGD oscillates b/w either direction of the gradient and update the weight.
- By adding the fraction of the previous update to the current update will make the process a bit faster.

4. Mini batch gradient descent :

- only a subset of the dataset is used for calculating the loss function
- It takes only fewer iterations so faster than SGD
- it is smoother than SGD
- it has good balance b/w speed and accuracy

5. Adagrad Adaptive gradient descent

- it uses different learning rates for each iteration
- The change in the learning rate depends upon the difference in the parameters unit training.

and