Optimizers - SGD, Momentum, Adam

Optimizers are algorithms that update model parameters (weights and bias) in order to minimize the loss function. they control how and how fast a neural network learns from data.

1. Stochastic Gradient Descent (SGD)

Training a neural network means minimizing a loss function $L(\theta)$ by updating parameters θ using gradients.

Update Rule:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta L(\theta_t)$$

- η : learning rate
- $\nabla_\theta L$: gradient of Loss

Idea:

More parameters in the direction of steepest descent.

Limitation:

- Slow convergence
- Oscillates in narrow valleys

2. Momentum

Momentum adds velocity to smooth updates.

- Velocity Update

$$v_t = \beta v_{t-1} + \nabla_\theta L(\theta_t)$$

- Parameter Update

$$\theta_{t+1} = \theta_t - \eta v_t$$

- $\beta \in [0, 1)$: momentum coefficient

Idea

- Accumulates past gradients
- Dampens oscillations
- Accelerate learning on consistent directions

3. Adam (Adaptive Moment Estimation)

Adam combines Momentum + Adaptive learning rates.

- First Moment (Mean)

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\nabla L$$

- Second Moment (Variance)

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)(\nabla L)^2$$

- ~~Bais~~ Bias Correction

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t}$$

- Update Rule

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

- Goal:

Efficiently minimize the loss while maintaining stable and fast convergence.

- Conclusion

Adam is robust for most problems, but SGD + Momentum often generalizes better.