SGD

Stochastic Gradient Descent (SGD) is a popular optimization algorithm widely used in machine learning for training models. It addresses the computational limitations of Batch Gradient Descent (BGD), especially when dealing with large datasets.

Core Idea:

- BGD computes the gradient of the error function using the entire training dataset in each iteration. This can be computationally expensive for large datasets.
- SGD tackles this challenge by utilizing a smaller subset of data (often a single data point or a mini-batch of data points) in each iteration to calculate the gradient. This significantly reduces the computational cost per iteration.

Steps in SGD:

1. **Initialization:** Similar to BGD, SGD starts with randomly initialized model parameters (weights and biases).
2. **Data Selection:**
    - In each iteration, SGD randomly selects a single data point (**single-step SGD**) or a small mini-batch of data points (**mini-batch SGD**) from the training dataset.
3. **Calculate Error:**
    - The model's prediction is made on the selected data point(s).
    - The error (difference between prediction and actual target value) is calculated.
4. **Calculate Gradient:**
    - The gradient of the error function with respect to the model parameters is calculated using only the information from the selected data point(s).
5. **Update Parameters:**

- The model parameters are updated in the direction opposite to the gradient, scaled by the learning rate. This step adjusts the parameters based on the error observed in the selected data point(s).

6. **Iteration:**
    - Steps 2-5 are repeated for a specified number of epochs (complete passes through the entire dataset, even though individual data points might be selected multiple times due to randomness).

Benefits of SGD:

- **Faster Training:** By using a smaller subset of data in each iteration, SGD significantly reduces the computational cost compared to BGD. This makes it more suitable for training models with large datasets.
- Avoidance of Local Minima: Due to the noisy updates in SGD, it has the ability to escape from local minima and converges to a global minimum.

Drawbacks of SGD:

- **Noisy updates:** Since SGD uses a smaller subset of data to calculate the gradient, the updates can be noisier compared to BGD, which uses the entire dataset. This can lead to fluctuations in the learning process.
- Slow Convergence: SGD may require more iterations to converge to the minimum since it updates the parameters for each training example one at a time.
- Sensitivity to Learning Rate: The choice of learning rate can be critical in SGD since using a high learning rate can cause the algorithm to overshoot the minimum, while a low learning rate can make the algorithm converge slowly.
- Less Accurate: Due to the noisy updates, SGD may not converge to the exact global minimum and can result in a suboptimal solution. This can be mitigated by using techniques such as learning rate scheduling and momentum-based updates.