**Section: Optimization in Machine Learning**

**Definition:**  
Optimization refers to the process of adjusting a model's parameters (weights and biases) to minimize the loss function, improving prediction accuracy.

**1. Gradient Descent (GD) – The “Clumsiest Optimizer”**

* **Mechanism:** Iterates over the **entire training set** before updating weights.
* **Learning Rate:** Must be small to avoid algorithm failure.
* **Problem:** Very slow because:
  + Each update is tiny.
  + Many epochs are needed.
  + Descent is gradual, like “snailing” down the gradient.

**Key Takeaway:** GD is theoretically sound but inefficient for practical use.

**2. Stochastic Gradient Descent (SGD)**

* **Improvement:** Updates weights **more frequently**, during a single epoch.
* **How it works:** Uses **batches** (mini batches) instead of the whole dataset.
  + Example:
    - Dataset: 10,000 points
    - Batch size: 1,000
    - Epoch updates: 10 instead of 1
* **Trade-off:** Slightly less precise than batch GD, but much faster.

**Hardware Advantage:**

* Batching allows **parallel processing** on CPU/GPU cores, giving a huge speed boost.

**Terminology Notes:**

* **Mini-batch Gradient Descent:** Updates after each batch (common in practice).
* **Stochastic Gradient Descent:** Technically updates after each input (batch size = 1).
* **Batch Gradient Descent:** Updates once per full epoch (batch = full dataset).

**3. Why Mini-batch / SGD is Preferred**

* Faster convergence.
* Compatible with modern hardware (parallelization).
* Practically, it’s the standard method in the industry.

✅ **Summary:**

* **GD (Batch GD):** Slow, updates once per epoch.
* **SGD (Mini-batch):** Faster, updates more frequently, widely used in practice.