[ ] Start coding or generate with AI.

Difference Between Gradient Descent, Stochastic Gradient Descent, and Vanishing Gradient

These three terms are related to optimization and training of neural networks, but they have different roles.

Gradient Descent (GD)

Definition:

Gradient Descent is an optimization algorithm used to minimize the loss function by updating the model's parameters (weights) in the direction of the steepest descent.

It computes the gradient (derivative) of the loss function and updates the weights iteratively.

Types of Gradient Descent:

Batch Gradient Descent (BGD)

Uses the entire dataset to compute the gradient before updating weights.

Pros: More stable convergence.

Cons: Slow for large datasets, high memory usage.

Stochastic Gradient Descent (SGD)

Uses a single randomly chosen sample from the dataset to update weights.

Pros: Faster, updates happen frequently.

Cons: High variance in updates (noisy learning).

Mini-Batch Gradient Descent (MBGD)

Uses a small batch of samples (instead of the entire dataset or one sample).

Pros: Balances stability and speed.

Cons: Requires choosing the right batch size.

Stochastic Gradient Descent (SGD)

Definition:

A variant of gradient descent that updates weights after each individual training example, rather than after processing the full dataset.

Instead of computing gradients on the entire dataset, it computes on one random sample at a

time.

Advantages of SGD: ✓ Faster than batch gradient descent.

Useful for large datasets.

Helps escape local minima due to randomness.

Disadvantages of SGD: ✕ Noisy updates may cause fluctuations.

Requires a good learning rate to converge well.

May not converge to the optimal solution but instead oscillate around it.

Key Difference from Batch GD:

Batch GD computes gradients for the whole dataset → More accurate but slow.

SGD updates the model after each sample → Noisy but much faster.

Vanishing Gradient Problem

Definition:

Occurs when the gradients become extremely small (close to zero) during backpropagation, causing the weight updates to slow down significantly.

This happens mostly in deep networks with many layers, especially with sigmoid or tanh activation functions.

Feature	Gradient Descent (GD)	Stochastic Gradient Descent (SGD)	Vanishing Gradient
Definition	Optimization algorithm for updating weights	Variant of GD that updates per sample	Issue where gradients shrink to near-zero
How It Works	Computes gradient over full dataset	Computes gradient on one random sample	Happens in deep networks due to backpropagation
Pros	Stable convergence	Fast and memory efficient	-
Cons	Slow for large datasets	Noisy updates	Early layers stop learning
Solution	Use Mini-Batch GD	Use momentum/Adam optimizer	Use ReLU, BatchNorm, ResNets

[ ] Start coding or generate with AI.