SMAI-S25-L21: More on BP

C. V. Jawahar

IIIT Hyderabad

April 8, 2025

Recap of Back Propagation

1. Initialize the network with random weights.
2. For all the samples, compute the output of the neural network $\mathbf{x}_{p+1}$
3. Compute the loss for the full batch of N samples as

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} L(\mathbf{x}_{p+1}^{i}, \mathbf{y}_i)$$

4. Adjust all the parameters (such as weight matrices) as

$$\theta^{k+1} \leftarrow \theta^{k} - \Delta\theta$$

or

$$\theta^{k+1} \leftarrow \theta^{k} - \eta \frac{\partial L}{\partial \theta}$$

5. Repeat steps 2-4 until convergence.

Challenges in Training NNs

- Non-Convex Optimization
- Many parameters to Control

Why simple GD works?

- Second order methods (that needs Hessian matrix) is compute intensive
- Nature of Loss Function (simultaneously all gradients vanish?)
- Multiple effective solutions

Refinements over BP

Over years, backpropagation algorithm has been refined significantly with many minor but critical innovations. What all can change in the simple version we saw early?

1. **step 1:** Initialization can be smarter.
2. **step 2:** Computing loss over a full batch and updating it once is not the best.
3. **step 3:** Loss function can be different. There re many other loss functions available beyond MSE.
4. **step 4:** Update rule can be different. What we saw here is too simple.

Refinements over BP

- Better Initialization
- Effective Gradient Estimates
- Better Update Rule: Momentum

$$\theta^{k+1} \leftarrow \theta^k - \Delta\theta$$

$$\Delta\theta = \eta \frac{\partial L}{\partial \theta}$$

$$\Delta\theta = \eta \frac{\partial L}{\partial \theta} + \gamma \Delta\theta^{t-1}$$

- Choice of Optimizer (such as Adam (adaptive momentum estimation))
- Loss Function
- Termination

Regularization

- L1/L2 regularization
- Data Augmentation
- Dropout
- Normalization (including in the middle of the network)
- etc.

Questions?