CODE BASICS

DEEP LEARNING TAKEAWAYS

codebasics.io

Chapter:

Model Optimization: Training Algorithms

Model Optimization Overview

1. Model Optimization is a process of finding the best way to train a model such that we can train it faster by using less compute resources and the model performs well during prediction phase
2. Model optimization can be done using various ways such as by using different optimizers (GD, Momentum, RMSProp etc), through regularization techniques (L1,L2, Dropout), hyperparameter tunning and so on

Exponentially Weighted Moving Average

1. Exponentially Weighted Moving Average (EWMA) gives more weight to recent data, smoothing out fluctuations over time.
2. The decay factor (β) controls how much past data influences the current average, with higher values giving longer memory. People tend to use 0.9 value for this factor a lot

Optimizer: Gradient Descent with Momentum

1. GD with momentum accelerates convergence by building on past gradients, reducing the time to reach the minimum.
2. The momentum term smooths out oscillations, especially in regions with steep, narrow valleys, leading to a more stable optimization path.
3. Momentum helps GD escape small local minima and flat regions, making it more effective in complex loss landscapes.
4. The momentum coefficient (β) controls how much influence previous gradients have on the current update.
5. GD with momentum is widely used in deep learning to achieve faster and more reliable training.

Optimizer: RMSProp

1. RMSProp uses a exponentially weighted moving average of squared gradients to reduce oscillations.
2. Helps models converge faster and works well with noisy gradients.

Optimizer: Adam

1. Adam combines momentum and RMSProp for efficient updates.
2. Tracks both mean and squared gradients to stabilize weight updates.