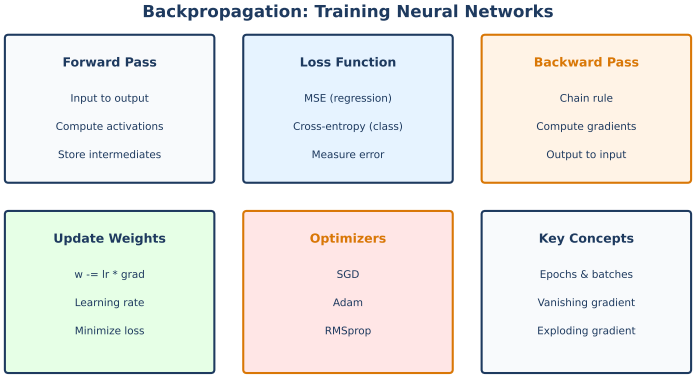
Lesson 35 Summary: Backpropagation

Data Science with Python – Key Concepts

Data Science Program

Backpropagation Overview



Backprop computes gradients to update neural network weights

The Training Loop

How neural networks learn:

- **Forward pass:** Compute predictions
- **Loss:** Measure error
- **Backward pass:** Compute gradients
- **Update:** Adjust weights

Repeat for many epochs until convergence

The Chain Rule

Core mathematical principle:

- **Composition:** $\frac{dL}{dw} = \frac{dL}{da} \cdot \frac{da}{dz} \cdot \frac{dz}{dw}$
- **Layer by layer:** Gradients flow backward
- **Efficient:** Reuse intermediate computations

Chain rule enables efficient gradient computation

Loss Functions

Measure prediction error:

- **MSE:** $(y - \hat{y})^2$ for regression
- **Cross-entropy:** $-y \log(\hat{y})$ for classification
- **Gradient:** Points toward loss reduction

Loss function defines what the network optimizes

Gradient Descent

Optimization algorithm:

- **Update rule:** $w = w - \eta \nabla L$
- **Learning rate:** Step size η
- **Direction:** Negative gradient reduces loss

Small learning rate = slow; large = unstable

SGD and Mini-Batches

Practical training:

- **Batch GD:** All data per update (slow)
- **SGD:** One sample per update (noisy)
- **Mini-batch:** Best of both (32-256 samples)

Mini-batch SGD is standard practice

Advanced Optimizers

Improvements over SGD:

- **Momentum:** Accumulates velocity
- **RMSprop:** Adapts learning rate per parameter
- **Adam:** Combines momentum + RMSprop

Adam is the default choice for most tasks

Gradient Problems

Training challenges:

- **Vanishing:** Gradients shrink to zero (deep nets)
- **Exploding:** Gradients grow unbounded
- **Solutions:** ReLU, batch norm, gradient clipping

These problems limited early deep learning

Quick Reference

Key Concepts:

Term	Description
Forward pass	Input to output computation
Backward pass	Gradient computation via chain rule
Learning rate	Controls update step size
Epoch	One pass through entire dataset
Batch size	Samples per gradient update

Backpropagation enables efficient neural network training