Vanishing Gradient Problem

Backprop

Output	Hidden 3	Hidden 2	Hidden 1	Input
1.0 100%	0.25 25%	0.06 6%	0.02 2%	~0 ≈0%

MATHEMATICAL CAUSE

$$\text{gradient} \propto \prod(\partial a_l/\partial z_l) \rightarrow \text{Product of many terms} < 1$$

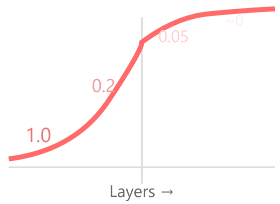
⚠ Consequences

- Early layers learn extremely slowly
- Weights barely change
- Training loss plateaus
- Network behaves like shallow

ReLU Activation	Skip Connections	Careful Init	Batch Norm

Activation Function Comparison

Sigmoid (σ) ❌ Gradient Vanishing



$\sigma'(x) = \sigma(x)(1-\sigma(x)) \leq 0.25$

ReLU ✅ Gradient Preserved

$\text{ReLU}'(x) = 1 \ (\text{if } x > 0)$

📊 Activation Functions & Gradient Vanishing Risk

⚠ High Risk

Sigmoid (σ)

$\sigma'(z) = \sigma(z)(1-\sigma(z))$

Max gradient: 0.25

$0.25^5 = 0.00098$ ❌

⚠ Medium Risk

Tanh

$\tanh'(z) = 1 - \tanh^2(z)$

Max gradient: 1.0

Saturates at extremes → vanish

✅ Lower Risk

ReLU

ReLU'(z) = 1 if z>0 else 0

Gradient: 0 or 1

No saturation for z>0 ✓

🔗 Chain Rule Multiplication: How Gradients Vanish

Backpropagation through layers:

$\partial L/\partial W_1 = \partial L/\partial a_4 \times \boldsymbol{\partial a_4 / \partial z_4} \times \partial z_4 / \partial a_3 \times \boldsymbol{\partial a_3 / \partial z_3} \times \partial z_3 / \partial a_2 \times \boldsymbol{\partial a_2 / \partial z_2} \times \partial z_2 / \partial a_1 \times \boldsymbol{\partial a_1 / \partial z_1} \times \partial z_1 / \partial W_1$

Blue terms: activation derivatives (σ', tanh', ReLU')

Example with Sigmoid activation:

❌ Bad Case (Sigmoid)

```
Layer 1: grad × W₁  × σ'
= 1.0 × 1.0 × 0.25 = 0.25
Layer 2: 0.25 × 1.0 × 0.25 = 0.0625
Layer 3: 0.0625 × 1.0 × 0.25 = 0.0156
Layer 4: 0.0156 × 1.0 × 0.25 ≈ 0.004
```

✅ Good Case (ReLU)

```
Layer 1: 1.0 × 1.0 × 1.0 = 1.0
Layer 2: 1.0 × 1.0 × 1.0 = 1.0
Layer 3: 1.0 × 1.0 × 1.0 = 1.0
Layer 4: 1.0 × 1.0 × 1.0 = 1.0
```

10 layers: $0.25^{10} \approx 0.00000095$
Practically zero! 📉

Gradient preserved!
Can train very deep networks ✓

📉 Real Vanishing Scenario

Sigmoid Max Gradient		10 Layers		Gradient Scale
$\sigma' = 0.25$	×	$n = 10$	→	$0.25^{10} \approx 0.000001$

With 20 layers: $0.25^{20} \approx 9.09 \times 10^{-13}$ → Effectively zero! ❄️