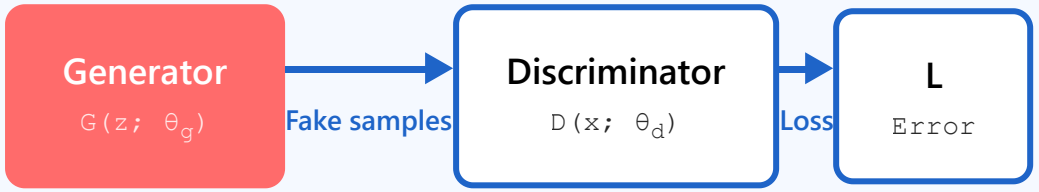
Gradient Flow

Backpropagation Through Networks



Gradient Backpropagation

Chain Rule

$$\partial L/\partial G = \partial L/\partial D \times \partial D/\partial G$$

Gradient Signals

- Real vs fake classification error
- Chain rule application

Key Challenge

- Gradient magnitude balance
- Stable signal propagation

⚠ Vanishing Gradients

When D is too strong, gradients become too small for G to learn

⚡ Exploding Gradients

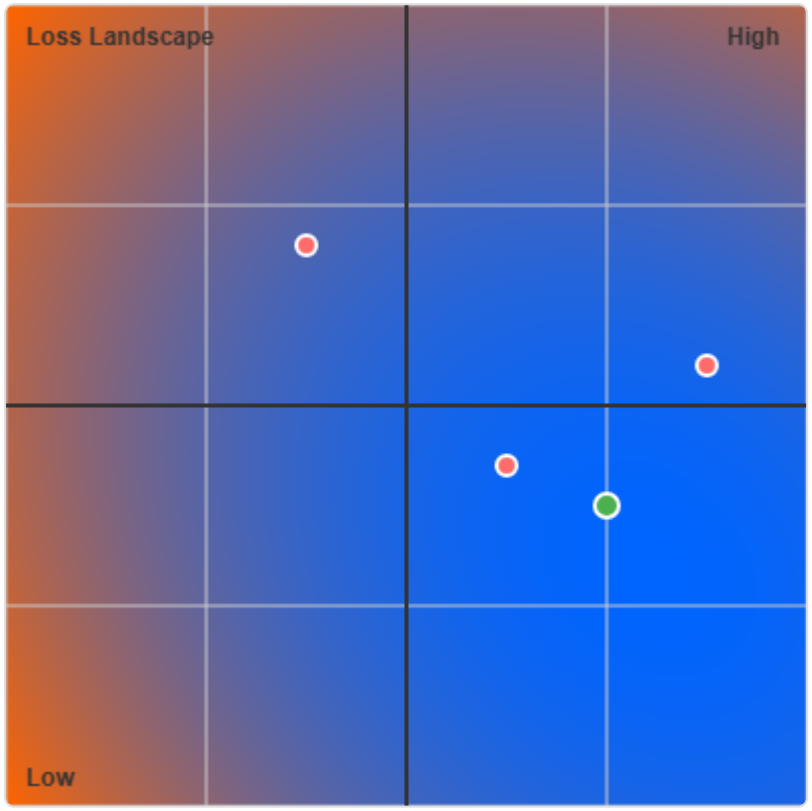
In early training, gradients can become unstable and too large

✓ Solutions

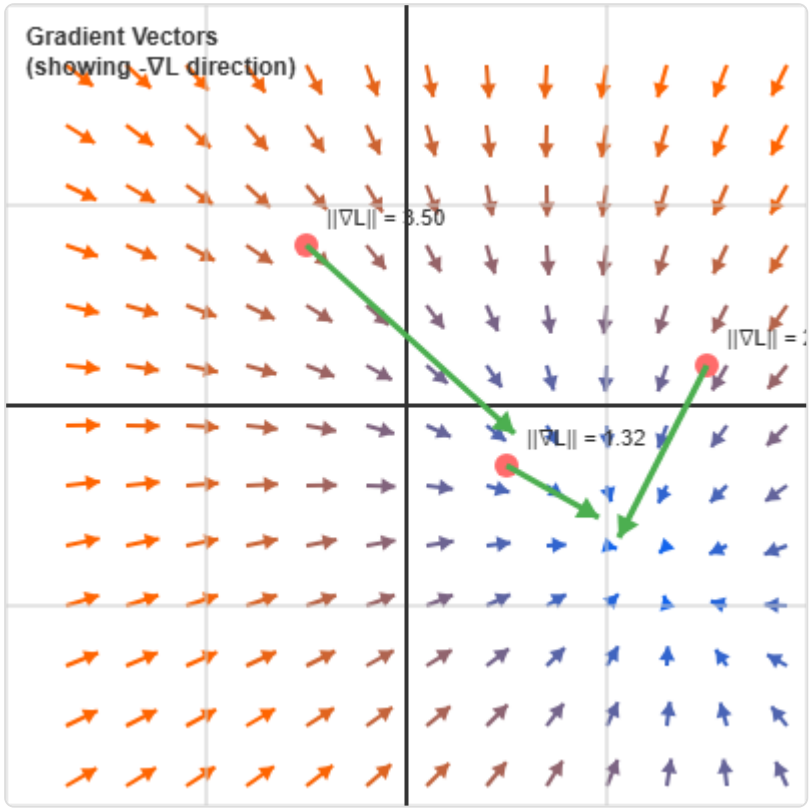
Gradient penalty techniques (e.g., WGAN-GP) help stabilize training

Interactive Gradient Flow Visualization

Loss Landscape



Gradient Field



Reset | Add Sample Point | Clear Points | Gradient Scale: Normal

Numerical Example: 2D Gradient Computation

Step 1: Define Loss Function

```
L(x, y) = (x - 1)² + (y + 0.5)² + 0.5·x·y
```

This represents a simplified discriminator loss in 2D parameter space

Step 2: Compute Partial Derivatives

```
∂L/∂x = 2(x - 1) + 0.5·y
∂L/∂y = 2(y + 0.5) + 0.5·x
```

Step 3: Example at Point (0.5, -0.3)

```
x = 0.5, y = -0.3

∂L/∂x = 2(0.5 - 1) + 0.5·(-0.3) = -1.0 - 0.15 = -1.15
∂L/∂y = 2(-0.3 + 0.5) + 0.5·(0.5) = 0.4 + 0.25 = 0.65
```

Gradient Vector: $\nabla L = (-1.15, 0.65)$
Magnitude: $||\nabla L|| = \sqrt{(1.15^2 + 0.65^2)} \approx 1.32$

Step 4: Parameter Update (Gradient Descent)

```
Learning rate: α = 0.1

x_new = x - α·(∂L/∂x) = 0.5 - 0.1·(-1.15) = 0.615
y_new = y - α·(∂L/∂y) = -0.3 - 0.1·(0.65) = -0.365
```

Updated Position: (0.615, -0.365)
Movement towards minimum: Point moves closer to (1, -0.5)

Gradient Magnitude Impact

- $||\nabla L|| < 0.01$: Vanishing
- $0.01 \leq ||\nabla L|| \leq 10$: Normal

Backprop Through Generator

$\partial L/\partial \theta_g = \partial L/\partial x \cdot \partial x/\partial \theta_g$

Chain rule multiplies gradients

- $||\nabla L|| > 10$: Exploding
- Affects learning speed

Deep networks: many multiplications

Can cause vanishing/exploding

Critical Insights

- Gradient direction: steepest ascent
- Negative gradient: descent
- Zero gradient: local min/max
- Balance is crucial for GANs

Real GAN Scenario

If D too good: $\partial L/\partial D \to 0$

Then: $\partial L/\partial G \to 0$ (vanishing)

G cannot learn effectively

Needs balanced training