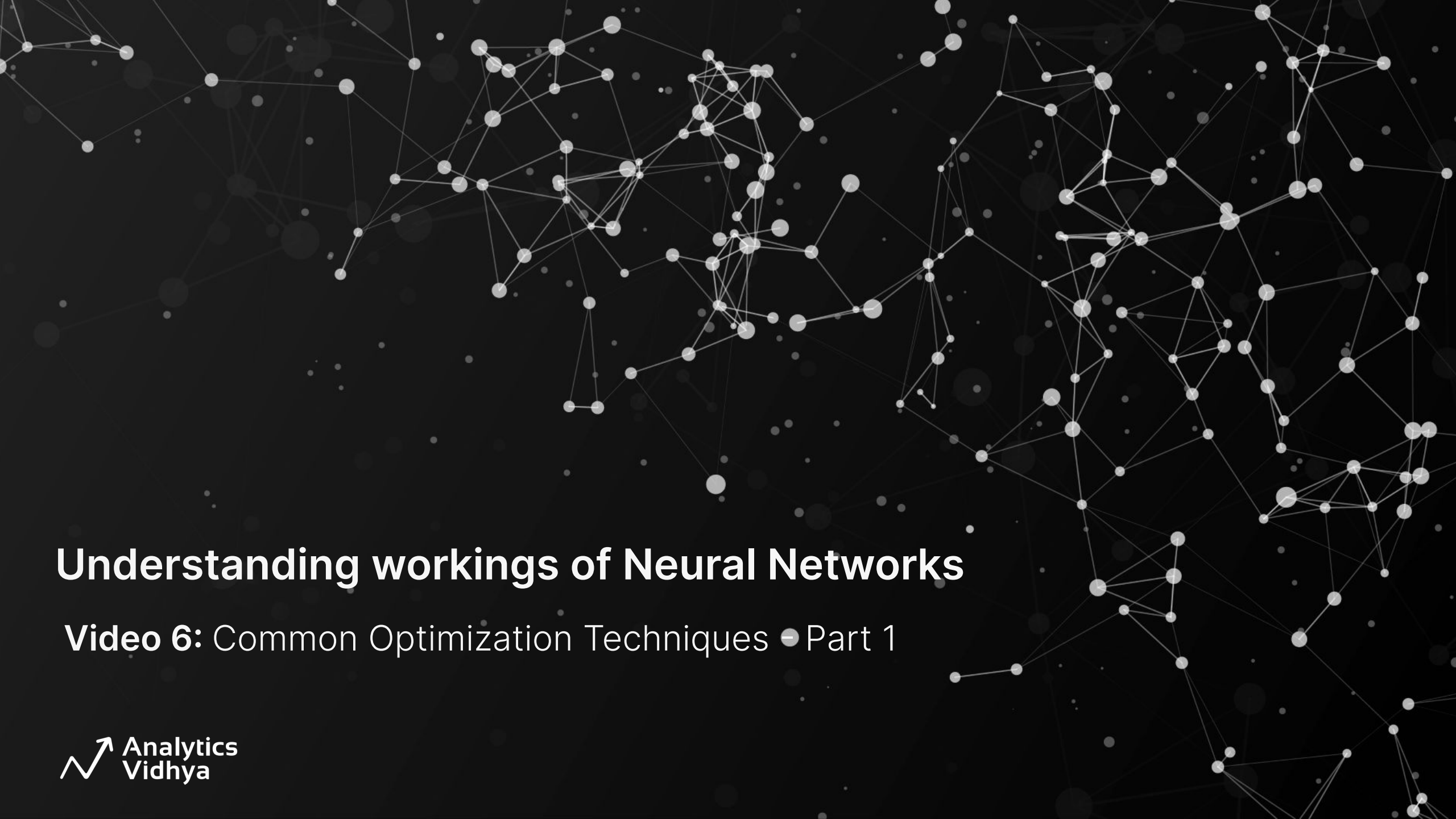


# Understanding workings of Neural Networks

**Video 6:** Common Optimization Techniques - Part 1

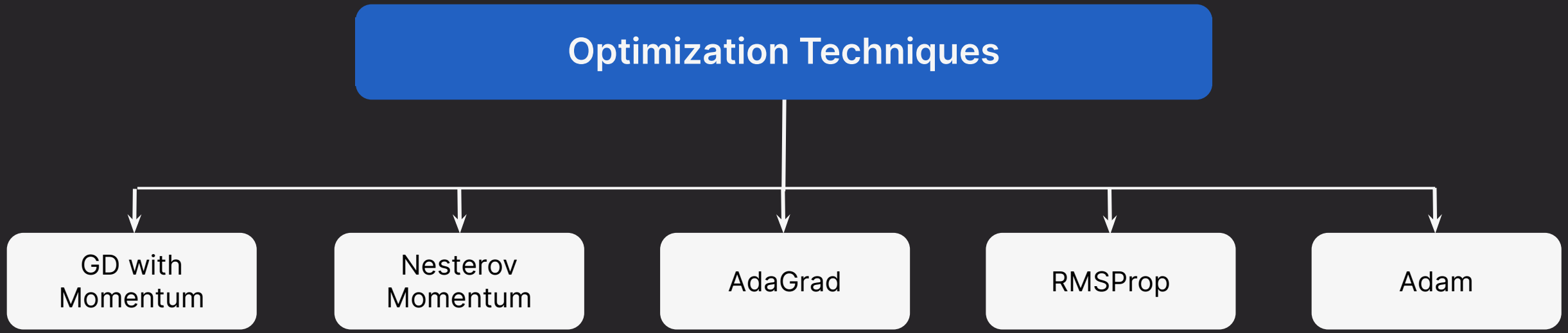
Analytics Vidhya

Analytics Vidhya

# In Air

Analytics Vidhya

# Common Optimization Techniques

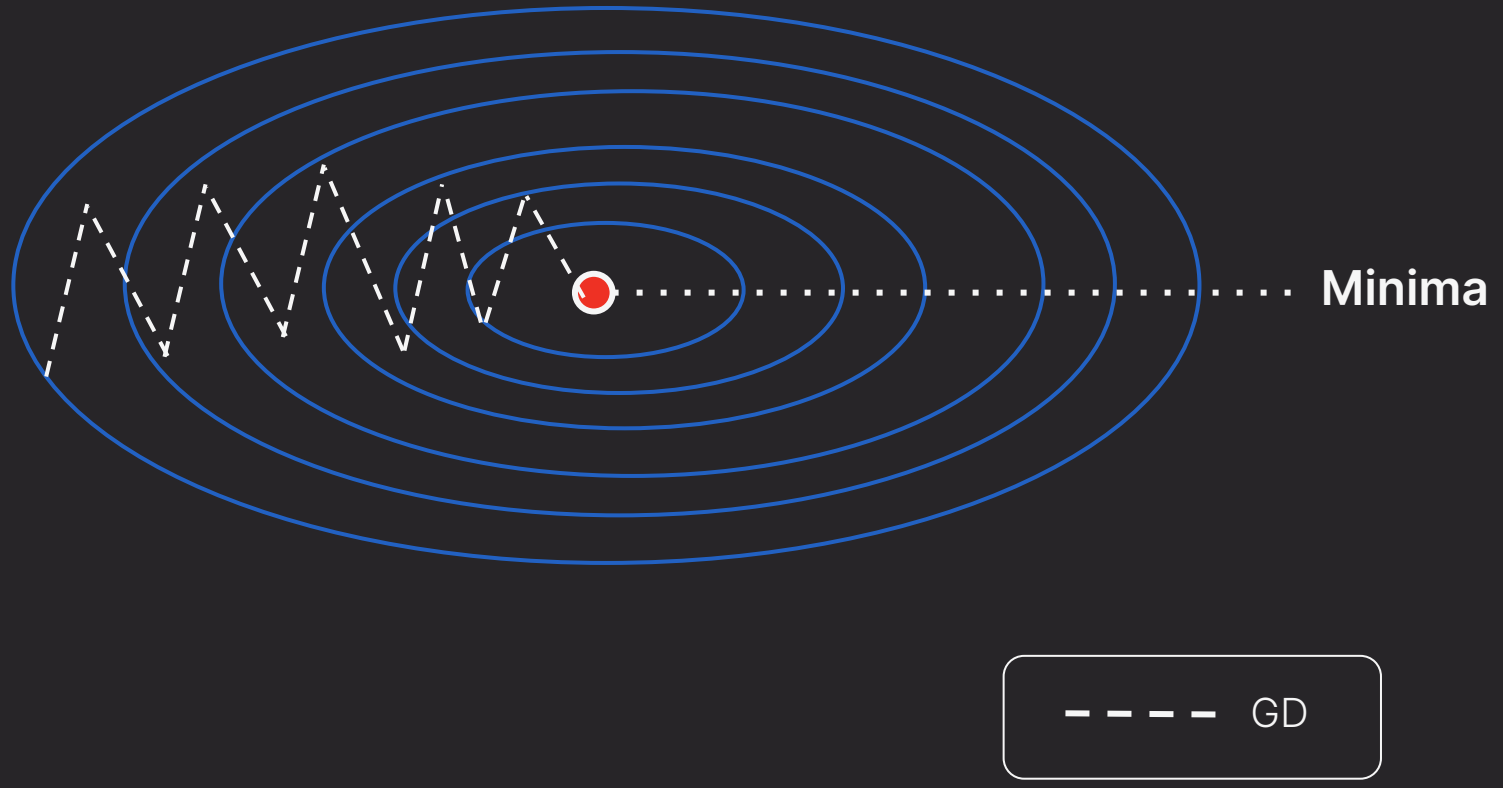


Analytics Vidhya

# Gradient Descent with Momentum

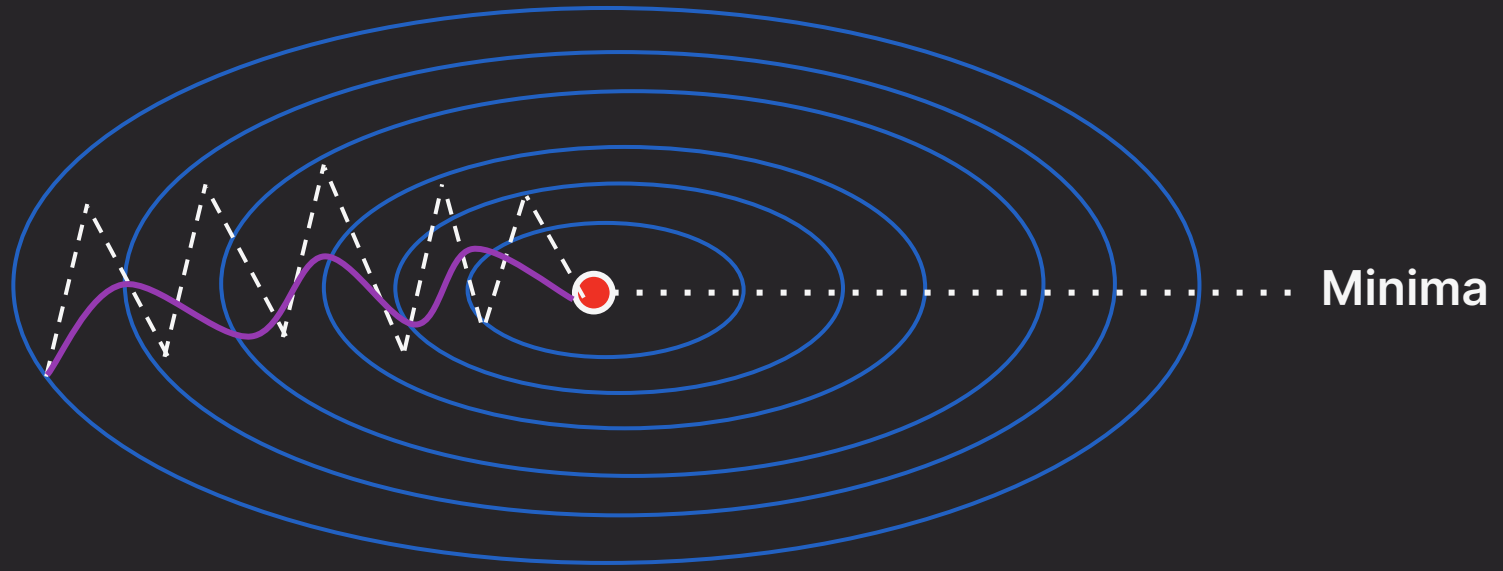
Analytics Vidhya

# Stochastic Gradient Descent with Momentum



Analytics Vidhya

# Stochastic Gradient Descent with Momentum




**Approach:** Finding the moving average of gradients

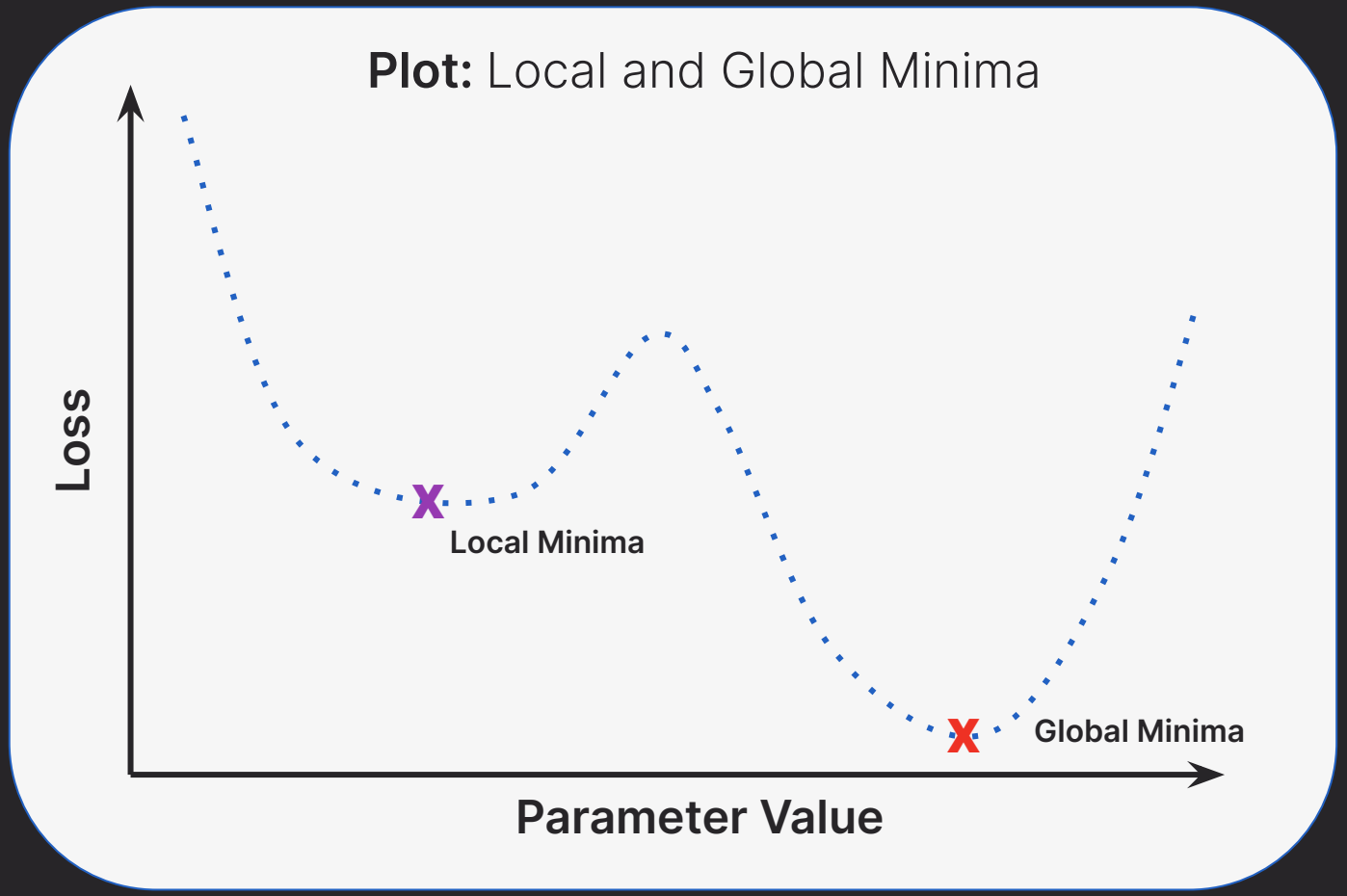
- Accelerate SGD
- Dampen the turbulence

GD

GD with Momentum

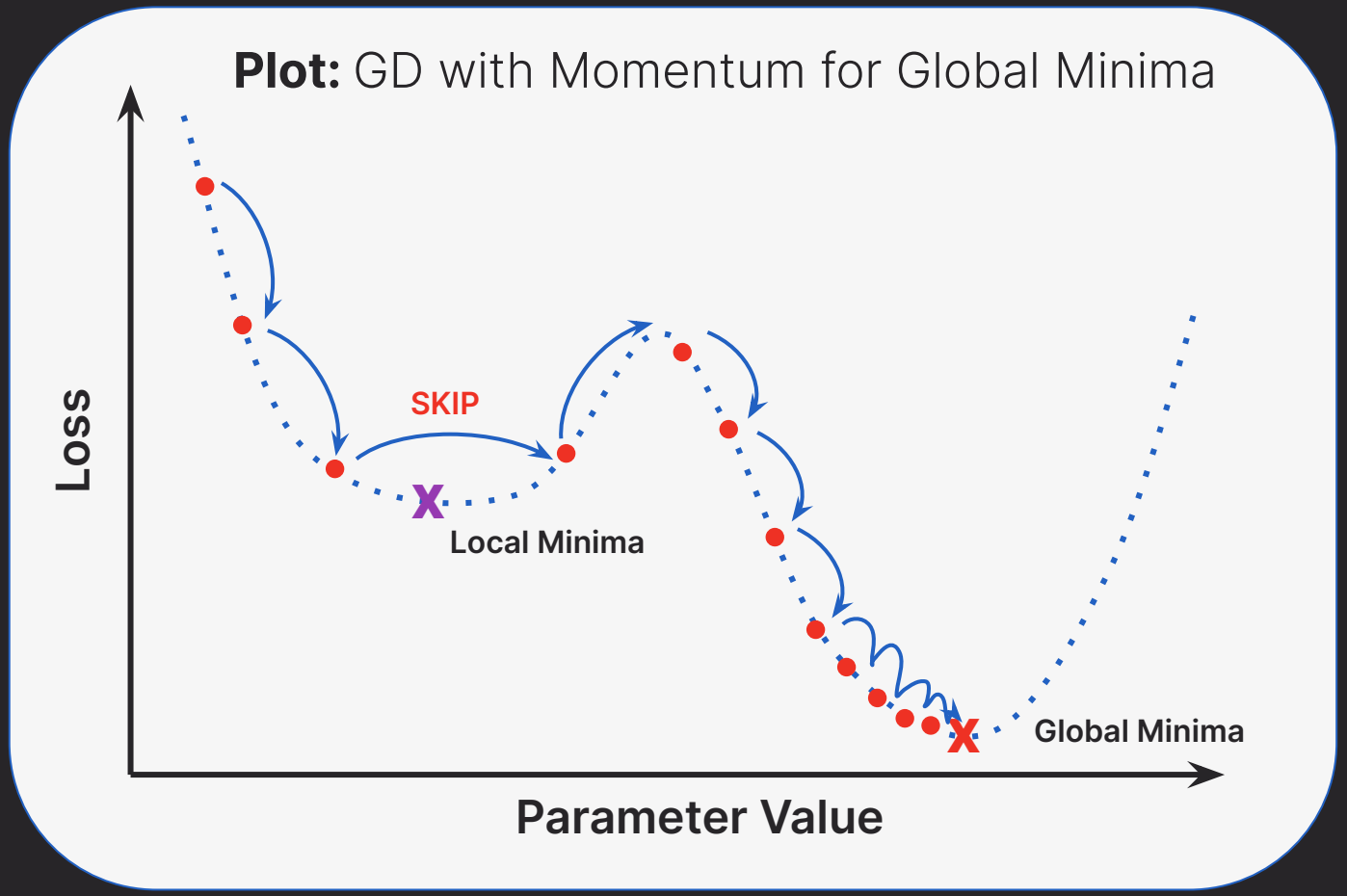
Analytics Vidhya

# Stochastic Gradient Descent with Momentum



Analytics Vidhya

# Stochastic Gradient Descent with Momentum



# Stochastic Gradient Descent with Momentum

**Moving average of gradients** → Simple Moving Average

Computationally expensive procedure

**Exponential Average**

Analytics Vidhya

# The Math Behind Exponential Average

1. Compute the gradient $gt$ at time step $t$.

$$g_t = \nabla_w \mathrm{L}(w)$$

Analytics Vidhya

# The Math Behind Exponential Average

1. Compute the gradient $gt$ at time step $t$.

$$g_t = \nabla_w L(w)$$

2. Calculate the moving average based on the previous timestamp value and the current gradient $gt$.

$$v_t = \beta v_{t-1} + (1 - \beta) g_t$$

$v_t$ = Moving Average

$\beta$ = Momentum Term

$\beta = 0.9 \sim$ Avg of 10 time stamps $\sim \frac{1}{1 - \beta}$

90% - weightage to the previous gradients

10% - weightage to current gradient

Analytics Vidhya

# The Math Behind Exponential Average

1. Compute the gradient $gt$ at time step $t$.

$$g_t = \nabla_w L(w)$$

2. Calculate the moving average based on the previous timestamp value and the current gradient $gt$.

$$v_t = \beta v_{t-1} + (1 - \beta) g_t$$

3. Update the weights $w$ using the moving average and learning rate.

$$w = w - \eta v_t$$

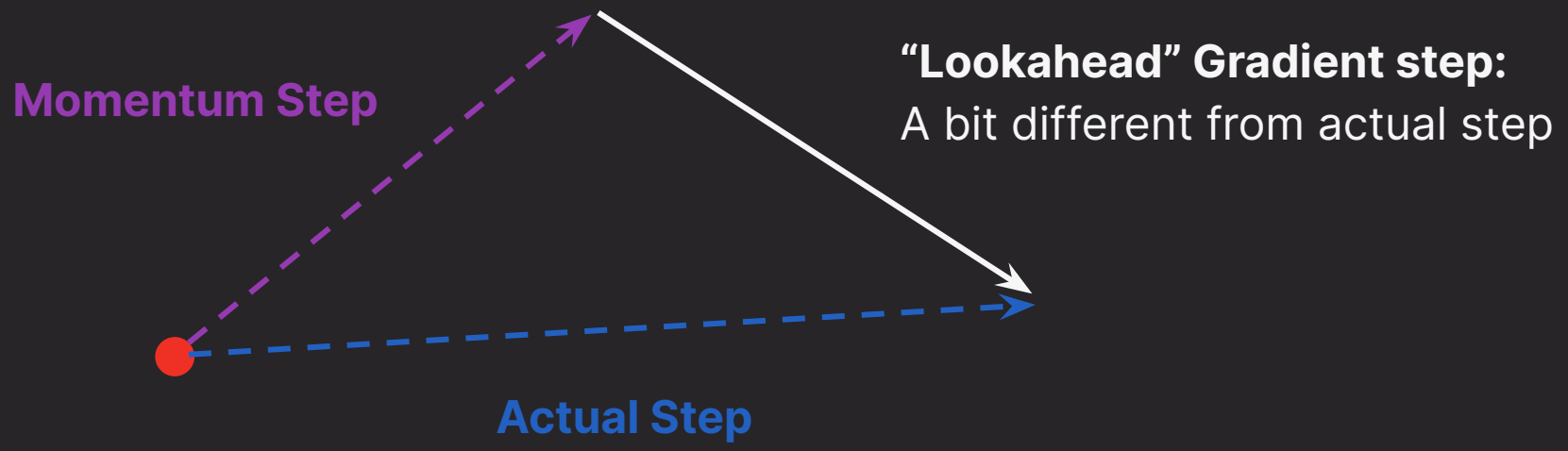
$\eta$ = Learning Rate

# Nesterov Momentum

Analytics Vidhya

# Nesterov Momentum

**Nesterov** refines momentum by looking at gradient or future steps.




**“Lookahead” Gradient step:**
A bit different from actual step

Analytics Vidhya

# The Math Behind Nesterov Momentum

1. Compute the gradient $gt$ at time step $t$.

$$w_{\text{lookahead}} = w - \beta v_{t-1}$$

Analytics Vidhya

# The Math Behind Nesterov Momentum

1. Compute the gradient $gt$ at time step $t$.

$$w_{\text{lookahead}} = w - \beta v_{t-1}$$

2. Compute the gradient $gt$ at the lookahead weight $w_{\text{lookahead}}$

$$g_t = \nabla_w L(w_{\text{lookahead}})$$

Analytics Vidhya

# The Math Behind Nesterov Momentum

1. Compute the gradient $gt$ at time step $t$.

$$w_{\text{lookahead}} = w - \beta v_{t-1}$$

2. Compute the gradient $gt$ at the lookahead weight $w_{\text{lookahead}}$

$$g_t = \nabla_w L(w_{\text{lookahead}})$$

3. Repeat steps 3 and 4 (same as SGD with Momentum).

$$v_t = \beta v_{t-1} + (1 - \beta) g_t$$

$$w = w - \eta v_t$$

Analytics Vidhya

# **Hands-on:** GD with Momentum & Nesterov Momentum