



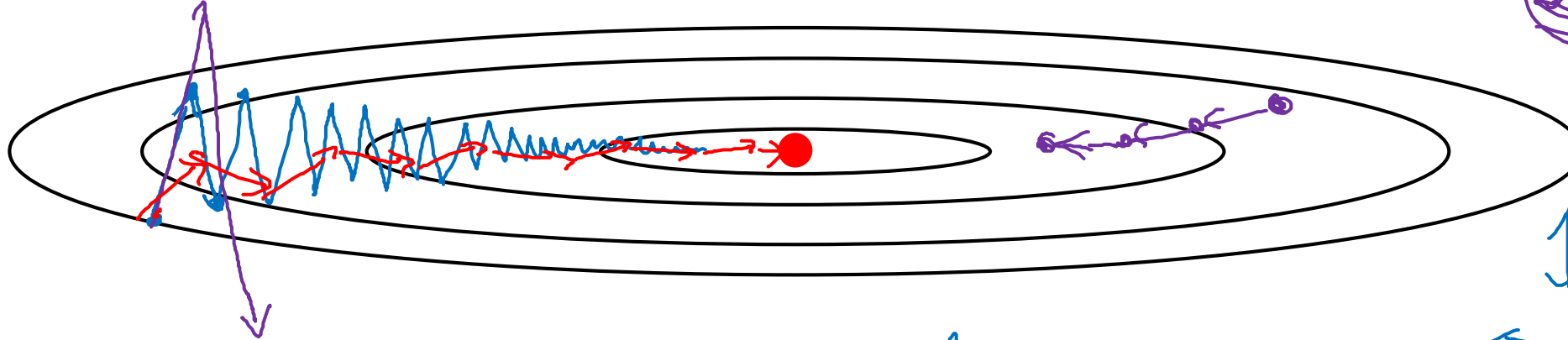
deeplearning.ai

# Optimization Algorithms

---

## Gradient descent with momentum

# Gradient descent example



↑ slower learning  
↔ faster learning

Momentum:

On iteration  $t$ :

Compute  $\Delta W, \Delta b$  on current mini-batch.

$$V_{\Delta W} = \beta V_{\Delta W} + (1-\beta) \Delta W$$

$$V_{\Delta b} = \beta V_{\Delta b} + (1-\beta) \Delta b$$

Friction — ↑ velocity

$$W := W - \alpha V_{\Delta W}$$

$$b := b - \alpha V_{\Delta b}$$

$$V_{\Theta} = \beta V_{\Theta} + (1-\beta) \Theta_t$$

deep learning ai

# Implementation details

$$v_{dW} = 0, \quad v_{db} = 0$$

On iteration  $t$ :

Compute  $dW, db$  on the current mini-batch

$$\left. \begin{aligned} \rightarrow v_{dW} &= \beta v_{dW} + (1 - \beta) dW \\ \rightarrow v_{db} &= \beta v_{db} + (1 - \beta) db \end{aligned} \right\} \quad \left| \quad \underbrace{v_{dW} = \beta v_{dW} + dW}_{\leftarrow}$$

$$W = W - \underbrace{\alpha}_{\text{learning rate}} v_{dW}, \quad b = \underline{b} - \underbrace{\alpha}_{\text{learning rate}} v_{db}$$

$$\frac{v_{dW}}{1 - \beta^t}$$

Hyperparameters:  $\alpha, \beta$

$$\underline{\beta = 0.9}$$

average over last  $\approx 10$  gradients