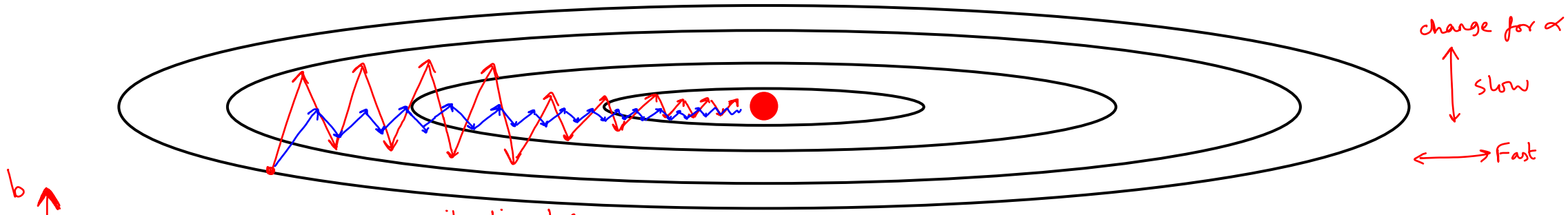
deeplearning.ai

Optimization Algorithms

RMSprop

RMSprop (Speeds up Gradient descent) – RMS = Root Mean Sq



on iteration t:
compute dw, db on current minibatch

$S_{dw} = \beta S_{dw} + (1-\beta)\, dw^2$ → 2 → element wise squared.

$S_{db} = \beta S_{db} + (1-\beta)\, db^2$

$w = w - \alpha \cdot \frac{dw}{\sqrt{S_{dw}}}$, $b = b - \alpha \cdot \frac{db}{\sqrt{S_{db}}}$

→ So for gradient to be small in Vertical direction (b) we expect S_{db} to be large

→ For gradient to be large in Horizontal direction (w) we expect S_{dw} to be small

If S_{dw} is small ⇒ update to w is large

If S_{db} is large ⇒ update to b is small

⇒ RMS prop graph becomes as shown in blue above

Damps out the vertical spikes

Speeds up Gradient descent as horizontal move is high

can ↑ α for faster learning

Just to make sure $\sqrt{S_{dw}}$ & $\sqrt{S_{db}}$ dont ever become 0 (w or b = ∞)

we can do $\sqrt{S_{dw} + \varepsilon}$ & $\sqrt{S_{db} + \varepsilon}$

$\varepsilon = 10^{-8}$

Adam optimization combines Momentum + RMS prop

Just to clarify, lets call β of Momentum = β_1

β of RMS prop = β_2

Andrew Ng