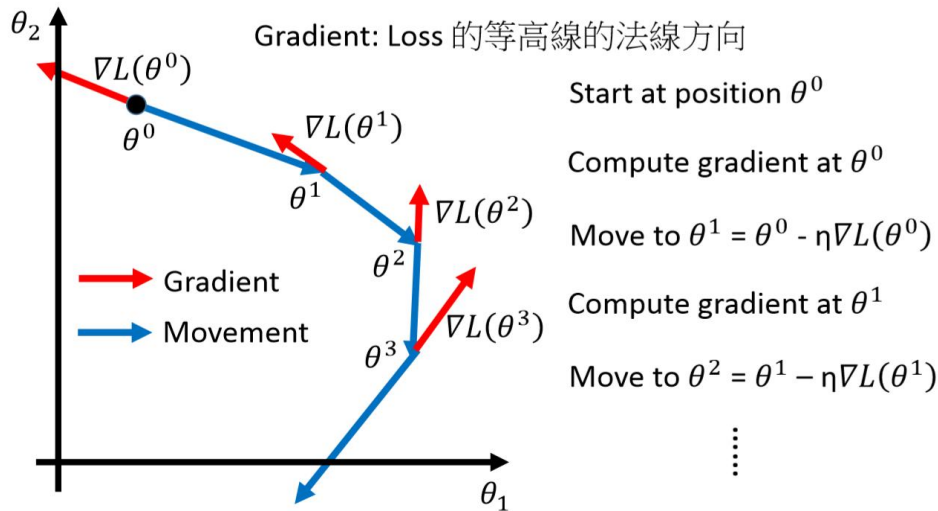


New Optimizers for Deep Learning

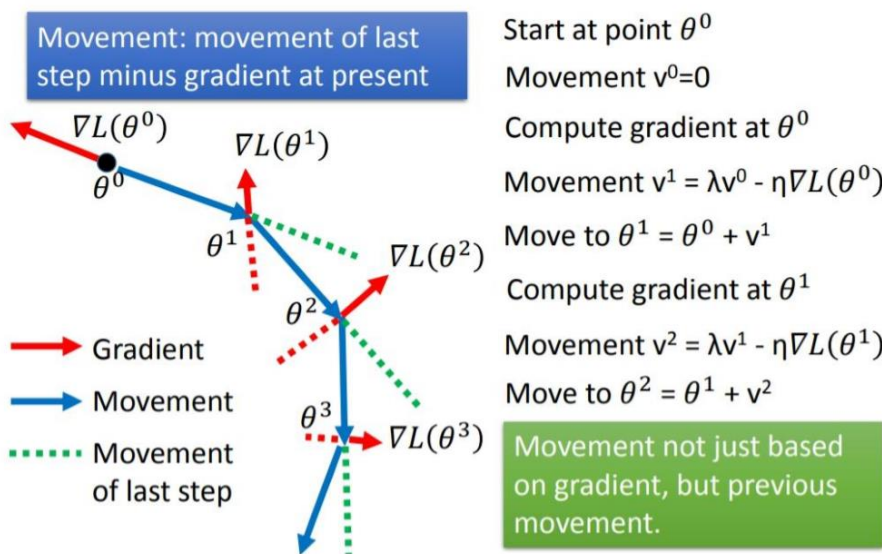
Some Notations

- θ_t : 第 t 次迭代的模型参数
- $\nabla L(\theta_t)$ or g_t : 用于计算 θ_{t+1} 的梯度 θ_t
- m_{t+1} : 从第 0 到 t 次迭代积累的动量, 计算梯度 θ_t

SGD



SGDM



v^i is actually the weighted sum of all the previous gradient: $\nabla L(\theta^0), \nabla L(\theta^1), \dots, \nabla L(\theta^{i-1})$

$$v^0 = 0$$

$$v^1 = -\eta \nabla L(\theta^0)$$

$$v^2 = -\lambda \eta \nabla L(\theta^0) - \eta \nabla L(\theta^1)$$

Start at point θ^0

Movement $v^0=0$

Compute gradient at θ^0

Movement $v^1 = \lambda v^0 - \eta \nabla L(\theta^0)$

Move to $\theta^1 = \theta^0 + v^1$

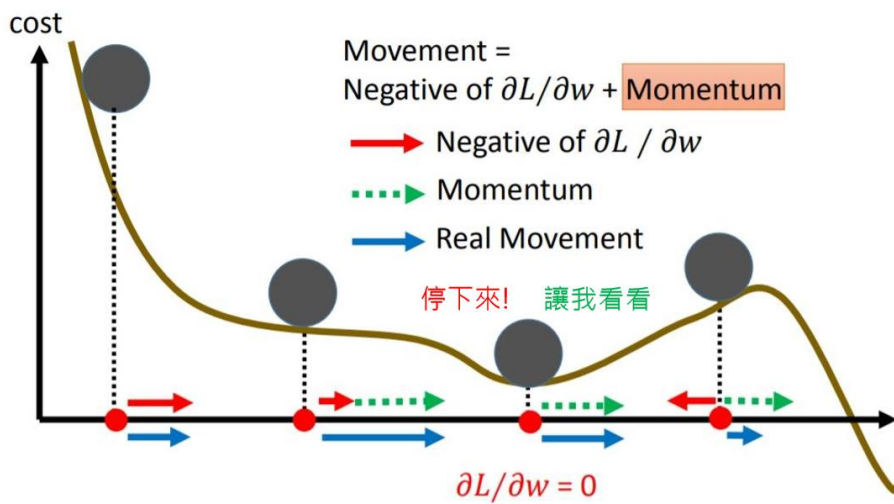
Compute gradient at θ^1

Movement $v^2 = \lambda v^1 - \eta \nabla L(\theta^1)$

Move to $\theta^2 = \theta^1 + v^2$

Movement not just based on gradient, but previous movement.

Why momentum?



Adagrad

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{\sum_{i=0}^{t-1} (g_i)^2}} g_{t-1}$$

RMSProp

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{v_t}} g_{t-1}$$

$$v_1 = g_0^2, v_t = \alpha v_{t-1} + (1 - \alpha)(g_{t-1})^2$$

Adam

● SGDM

$$\theta_t = \theta_{t-1} - \eta m_t$$

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_{t-1}$$

● RMSProp

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{v_t}} g_{t-1}$$

$$v_1 = g_0^2, v_t = \beta_2 v_{t-1} + (1 - \beta_2)(g_{t-1})^2$$

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} \hat{m}_t$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

$$\beta_1 = 0.9$$

$$\beta_2 = 0.999$$

$$\epsilon = 10^{-8}$$

AMSGrad

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} m_t$$

$$\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$$

RAdam

$$p_t = p_\infty - \frac{2t\beta_2^t}{1 - \beta_2^t}$$

$$p_\infty = \frac{2}{1 - \beta_2} - 1$$

$$r_t = \sqrt{\frac{(p_t - 4)(p_t - 2)p_\infty}{(p_\infty - 4)(p_\infty - 2)p_t}}$$

When $p_t \leq 4$ (first few steps of training)

$$\theta_t = \theta_{t-1} - \eta \hat{m}_t$$

When $p_t > 4$

$$\theta_t = \theta_{t-1} - \frac{\eta r_t}{\sqrt{\hat{v}_t + \epsilon}} \hat{m}_t$$

L2 regularization

$$L_{l_2}(\theta) = L(\theta) + \frac{1}{2}\gamma||\theta||^2$$

$$\begin{aligned}\text{SGD: } \theta_t &= \theta_{t-1} - \nabla L_{l_2}(\theta_{t-1}) \\ &= \theta_{t-1} - \nabla L(\theta_{t-1}) - \gamma|\theta_{t-1}|\end{aligned}$$

$$\begin{aligned}\text{SGDM: } \theta_t &= \theta_{t-1} - \lambda m_{t-1} - \eta(\nabla L(\theta_{t-1}) + \gamma|\theta_{t-1}|) \\ m_t &= \lambda m_{t-1} + \eta(\nabla L(\theta_{t-1}) + \gamma|\theta_{t-1}|)\end{aligned}$$

$$\begin{aligned}\text{Adam: } m_t &= \lambda m_{t-1} + \eta(\nabla L(\theta_{t-1}) + \gamma|\theta_{t-1}|) \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2)(\nabla L(\theta_{t-1}) + \gamma|\theta_{t-1}|)^2\end{aligned}$$

$$\begin{aligned}\text{SGDWM: } \theta_t &= \theta_{t-1} - m_t - \gamma\theta_{t-1} \\ m_t &= \lambda m_{t-1} + \eta(\nabla L(\theta_{t-1}))\end{aligned}$$

$$\text{AdamW: } m_t = \beta_1 m_{t-1} + (1 - \beta_1)(\nabla L(\theta_{t-1}))$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\nabla L(\theta_{t-1}))^2$$

$$\theta_t = \theta_{t-1} - \eta\left(\frac{1}{\sqrt{\hat{v}_t + \varepsilon}} \hat{m}_t - \gamma\theta_{t-1}\right)$$

SGDM vs Adam

SGDM	Adam
<ul style="list-style-type: none">● Slow● Better convergence● Stable● Smaller generalization gap	<ul style="list-style-type: none">● Fast● Possibly non-convergence● Unstable● Larger generalization gap

Advices

SGDM	Adam
<ul style="list-style-type: none">● Computer vision Image classification Segmentation Object detection	<ul style="list-style-type: none">● NLP QA Machine translation Summary● Speech synthesis● GAN● Reinforcement learning