

Week 2-2 Adam Algo, GC with Momentum (Exponentially Weighted MA) + RMS prop

笔记本: DL 2 - Deep NN Hyperparameter Tunning, Regularization & Optimization

创建时间: 2021/1/9 11:15

更新时间: 2021/1/9 12:33

adam paper:

<https://arxiv.org/pdf/1412.6980.pdf>

Exponentially weighted moving averages

Temperature in London

$$\theta_1 = 40^\circ\text{F} \quad 4^\circ\text{C} \leftarrow$$

$$\theta_2 = 49^\circ\text{F} \quad 9^\circ\text{C}$$

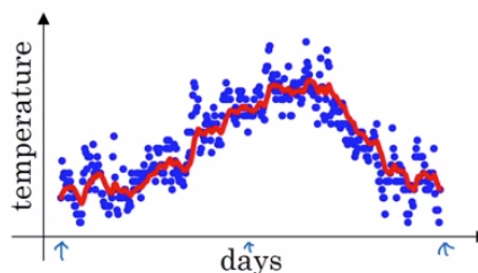
$$\theta_3 = 45^\circ\text{F} \quad \vdots$$

\vdots

$$\theta_{180} = 60^\circ\text{F} \quad 15^\circ\text{C}$$

$$\theta_{181} = 56^\circ\text{F} \quad \vdots$$

\vdots



$$V_0 = 0$$

$$V_1 = 0.9 V_0 + 0.1 \theta_1$$

$$V_2 = 0.9 V_1 + 0.1 \theta_2$$

$$V_3 = 0.9 V_2 + 0.1 \theta_3$$

\vdots

$$V_t = 0.9 V_{t-1} + 0.1 \theta_t$$

Exponentially weighted averages

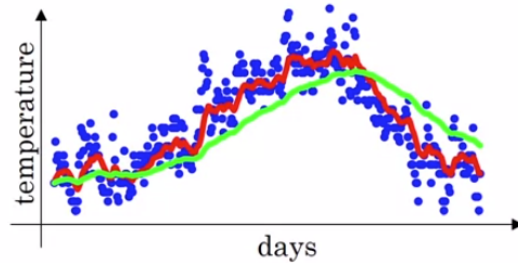
$$v_t = \beta v_{t-1} + (1-\beta) \theta_t$$

$$\beta = 0.9 : \approx 10 \text{ days' temper.}$$

$$\beta = 0.98 : \approx 50 \text{ days}$$

v_t is approximately
average over
 $\approx \frac{1}{1-\beta}$ days'
temperature.

$$\frac{1}{1-0.98} = 50$$



Andrew Ng

Exponentially weighted averages

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

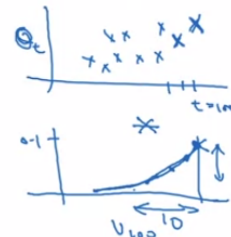
$$v_{100} = 0.9v_{99} + 0.1\theta_{100}$$

$$v_{99} = 0.9v_{98} + 0.1\theta_{99}$$

$$v_{98} = 0.9v_{97} + 0.1\theta_{98}$$

...

$$\begin{aligned} \rightarrow v_{100} &= 0.1 \theta_{100} + 0.9 (0.1 \theta_{99} + 0.9 v_{98}) \\ &= 0.1 \theta_{100} + 0.1 \times 0.9 \theta_{99} + 0.1 (0.9)^2 \theta_{98} + 0.1 (0.9)^2 \theta_{97} + 0.1 (0.9)^4 \theta_{96} + \dots \\ &\quad \underbrace{0.9^{10}}_{\approx 0.35 \approx \frac{1}{e}} \quad \underbrace{(1-\epsilon)^{\frac{1}{\epsilon}}}_{\approx \frac{1}{e}} \end{aligned}$$



$$0.1 \theta_{98} + 0.9 v_{97}$$

$$\begin{aligned} &0.98^50 \approx \frac{1}{e} \\ &\rightarrow 0.98^50 \approx \frac{1}{e} \end{aligned}$$

Andrew Ng

(all these weights approximately
add up to 1)
after $1/\epsilon$ -> $1/e$ weight, rather
small

Implementing exponentially weighted averages

$$v_0 = 0$$

$$v_1 = \beta v_0 + (1 - \beta) \theta_1$$

$$v_2 = \beta v_1 + (1 - \beta) \theta_2$$

$$v_3 = \beta v_2 + (1 - \beta) \theta_3$$

...

$$V_\theta := 0$$

$$V_\theta := \beta v + (1 - \beta) \theta_1$$

$$V_\theta := \beta v + (1 - \beta) \theta_2$$

:

$$\rightarrow V_\theta = 0$$

Repeat {

Get next θ_t

$$V_\theta := \beta V_\theta + (1 - \beta) \theta_t \leftarrow$$

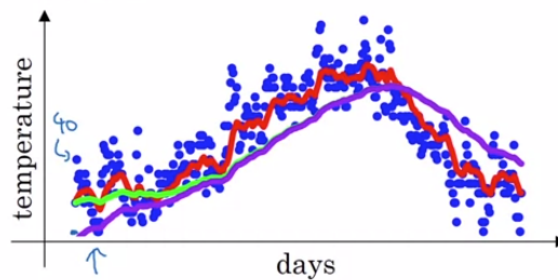
}

Andrew Ng

(memory efficient)

Bias correction

Bias correction



$$\beta = 0.98$$

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

$$V_0 = 0$$

$$V_1 = \cancel{0.98 V_0} + 0.02 \theta_1$$

$$V_2 = 0.98 V_1 + 0.02 \theta_2$$

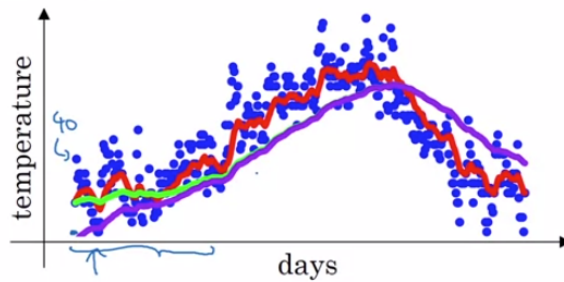
$$= 0.98 \times 0.02 \times \theta_1 + 0.02 \theta_2$$

$$= 0.0196 \theta_1 + 0.02 \theta_2$$

Andrew Ng

if we initialize with 0, it gives the purple curve

Bias correction



$$\beta = 0.98$$

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

$$v_0 = 0$$

$$v_1 = \cancel{0.98 v_0} + 0.02 \theta_1$$

$$\begin{aligned} v_2 &= 0.98 v_1 + 0.02 \theta_2 \\ &= 0.98 \times 0.02 \times \theta_1 + 0.02 \theta_2 \\ &= 0.0196 \theta_1 + 0.02 \theta_2 \end{aligned}$$

$$\frac{v_t}{1 - \beta^t}$$

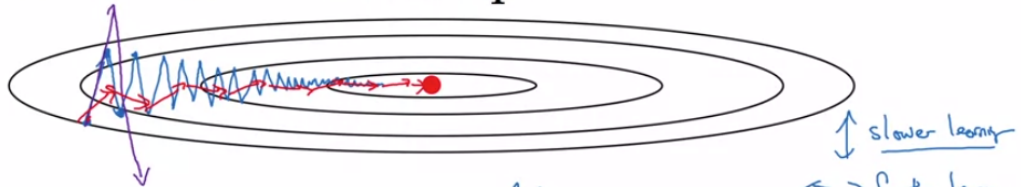
$$t=2: 1 - \beta^t = 1 - (0.98)^2 = 0.0396$$

$$\frac{v_2}{0.0396} = \frac{0.0196 \theta_1 + 0.02 \theta_2}{0.0396}$$

Andrew Ng

average of these derivatives

Gradient descent example



Momentum:

On iteration t :

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

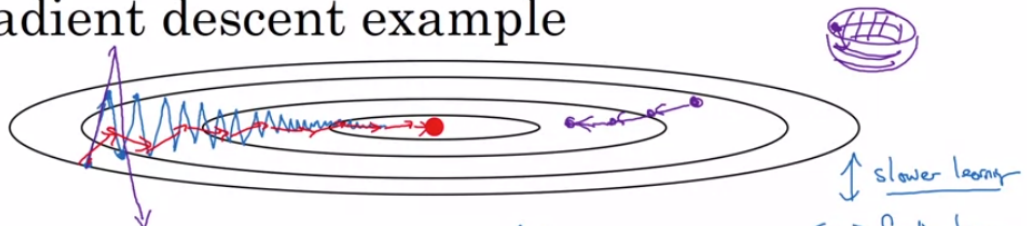
$$v_{\Delta W} = \beta v_{\Delta W} + (1 - \beta) \Delta W$$

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

$$v_0 = \beta v_0 + (1 - \beta) \theta_t$$

$$W := W - \alpha v_{\Delta W}, \quad b := b - \alpha v_{\Delta b}$$

Gradient descent example



Momentum:

On iteration t :

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

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

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

$$v_0 = \beta v_0 + (1 - \beta) \theta_t$$

friction \rightarrow velocity \rightarrow acceleration

$$W := W - \alpha v_{\Delta W}, \quad b := b - \alpha v_{\Delta b}$$

(add friction so that its horizontal speed along the bowl gets smaller)

Implementation details

$$v_{dW} = 0$$

On iteration t :

Compute dW, db on the current mini-batch

$$v_{dW} = \beta v_{dW} + (1 - \beta) dW$$

$$v_{db} = \beta v_{db} + (1 - \beta) db$$

$$W = W - \alpha v_{dW}, \quad b = b - \alpha v_{db}$$



A handwritten diagram showing the update rule for v_{dW} . It consists of a horizontal line with v_{dW} written above it and $1 - \beta$ written below it. A diagonal line is drawn across the entire expression, indicating that the $(1 - \beta) dW$ term is crossed out or omitted.

Hyperparameters: α, β

$\uparrow \uparrow$

$$\beta = 0.9$$

average over last ≈ 10 gradients

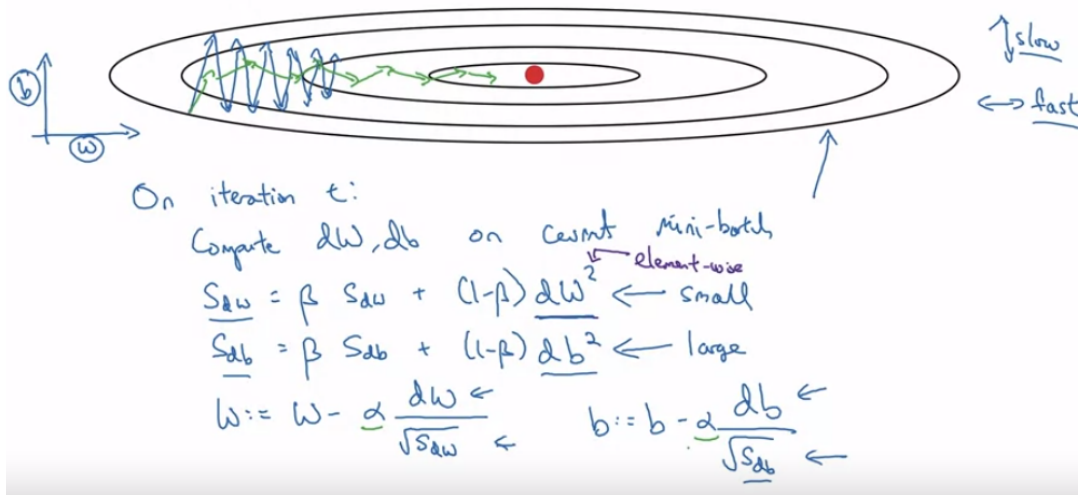
Andrew Ng

In practice, people don't usually do this (bias correction) because after just ten iterations, your moving average will have warmed up and is no longer a bias estimate.

We sometimes omit the $(1 - \beta)$ term, but it is not so intuitive and we may need to adjust the learning rate correspondingly

RMSprop

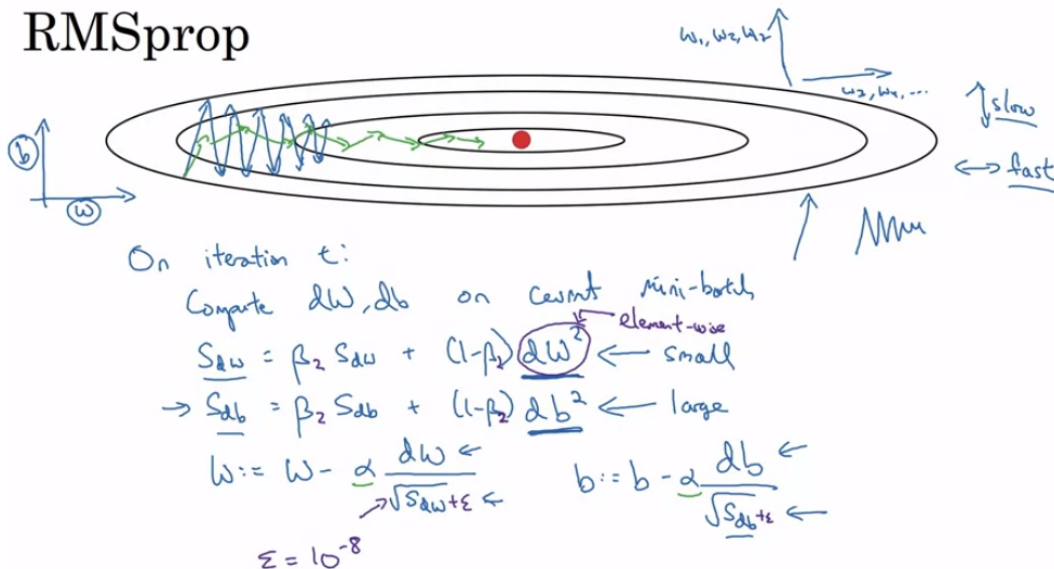
RMSprop



larger db gets more scale division,
also enables larger learning rate

We also add a epsilon for numerical stability (division by zero)

RMSprop



Adam

Adam optimization algorithm

$V_{dw}=0, S_{dw}=0, V_{db}=0, S_{db}=0$

On iteration t :

Compute dw, db using current mini-batch

$$V_{dw} = \beta_1 V_{dw} + (1-\beta_1) dw, \quad V_{db} = \beta_1 V_{db} + (1-\beta_1) db \quad \leftarrow \text{"momentum"} \beta_1$$

$$S_{dw} = \beta_2 S_{dw} + (1-\beta_2) dw^2, \quad S_{db} = \beta_2 S_{db} + (1-\beta_2) db^2 \quad \leftarrow \text{"RMSprop"} \beta_2$$

$$V_{dw}^{corrected} = V_{dw} / (1-\beta_1^t), \quad V_{db}^{corrected} = V_{db} / (1-\beta_1^t)$$

$$S_{dw}^{corrected} = S_{dw} / (1-\beta_2^t), \quad S_{db}^{corrected} = S_{db} / (1-\beta_2^t)$$

$$W := W - \alpha \frac{V_{dw}^{corrected}}{\sqrt{S_{dw}^{corrected} + \epsilon}}, \quad b := b - \alpha \frac{V_{db}^{corrected}}{\sqrt{S_{db}^{corrected} + \epsilon}}$$

Andrew Ng

(there is db^2)

Hyperparameters choice:

$\rightarrow \alpha$: needs to be tune

$\rightarrow \beta_1$: 0.9 $\rightarrow (dw)$

$\rightarrow \beta_2$: 0.999 $\rightarrow (dw^2)$

$\rightarrow \epsilon$: 10^{-8}

Adam: Adaptive moment estimation

Adaptive moment estimation (first order momentum + second order momentum)

