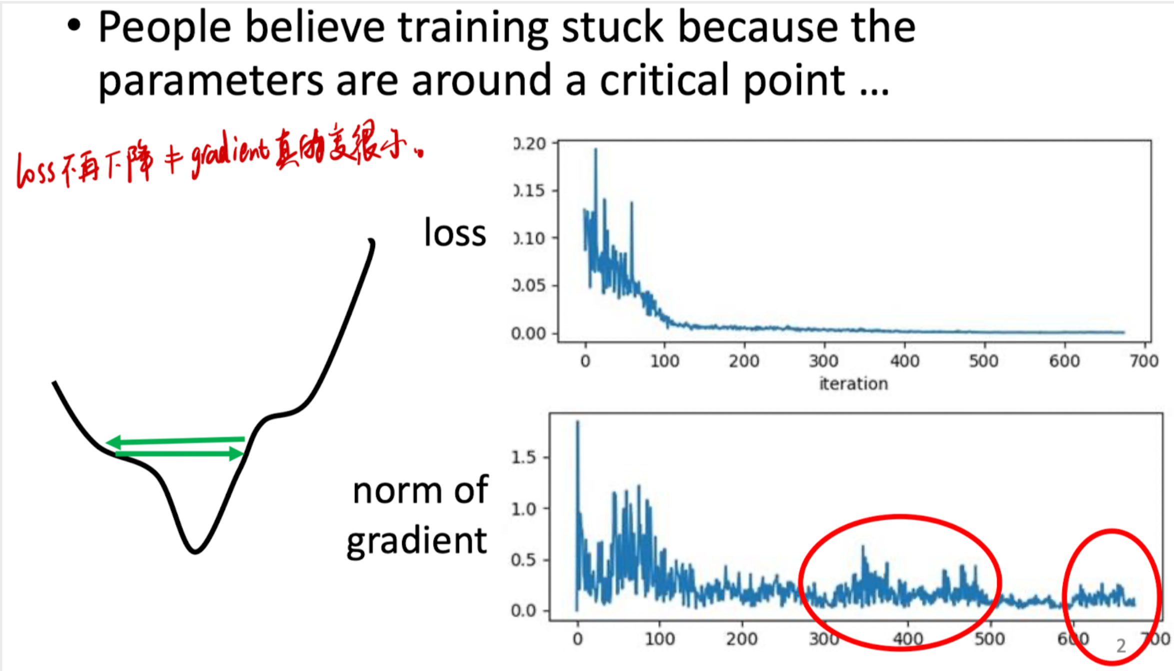
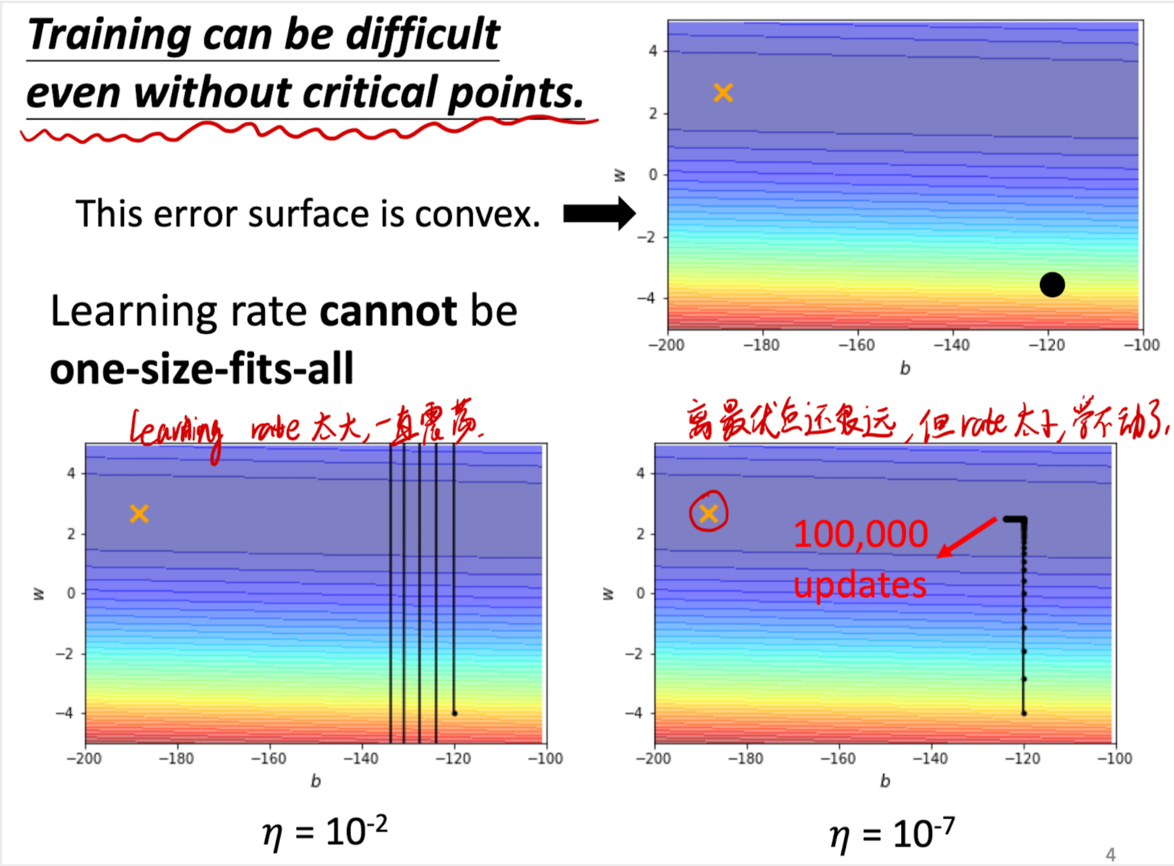
2.3 Adaptive Learning Rate

- Root Mean Square
- RMSProp
- Adam: RMSProp + Momentum

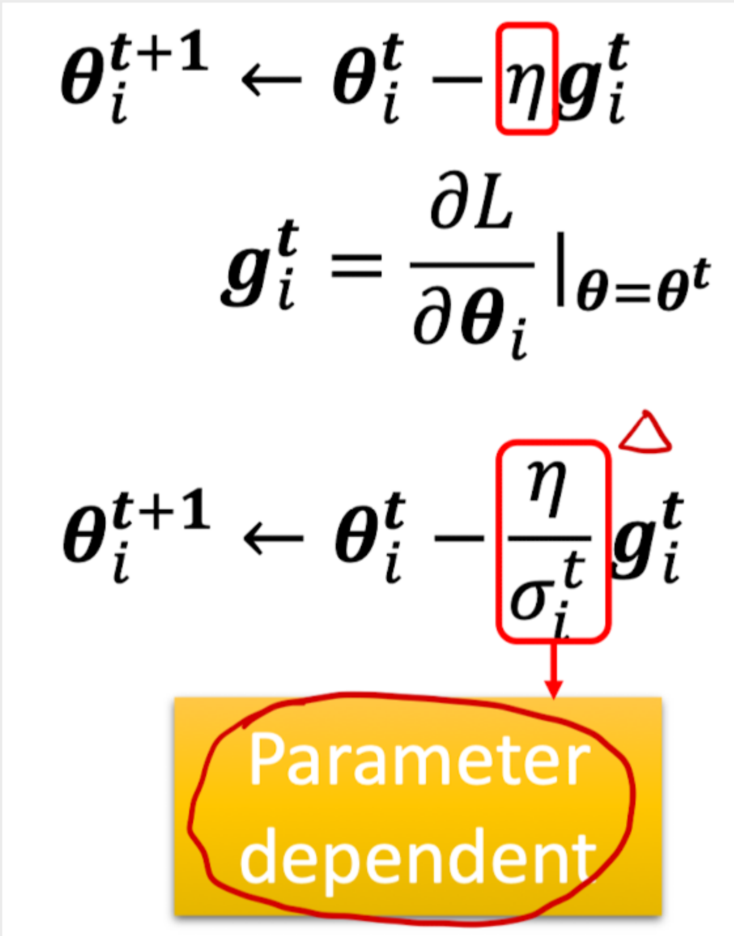
loss不再下降不意味着gradient真的变得很小，他可能在震荡，卡住的时候可以算gradient的norm。



没有critical point时候训练也可能是困难。下图左上角X是error surface的最低点，这个error surface是convex的。



learning rate的特制化：learning rate需要自动根据gradient的大小做调整
如何做：加上一个dependant on 参数i并且dependant on于iteration t。
parameter dependant & iteration dependant。



parameter dependant的常见计算方式

1.Root Mean Square

1

Root Mean Square $\theta_i^{t+1} \leftarrow \theta_i^t - \boxed{\frac{\eta}{\sigma_i^t}} g_i^t$

$$\theta_i^1 \leftarrow \theta_i^0 - \frac{\eta}{\sigma_i^0} g_i^0 \qquad \sigma_i^0 = \sqrt{\left(g_i^0\right)^2} = \left|g_i^0\right|$$

$$\theta_i^2 \leftarrow \theta_i^1 - \frac{\eta}{\sigma_i^1} g_i^1 \qquad \sigma_i^1 = \sqrt{\frac{1}{2}\left[\left(g_i^0\right)^2 + \left(g_i^1\right)^2\right]}$$

$$\theta_i^3 \leftarrow \theta_i^2 - \frac{\eta}{\sigma_i^2} g_i^2 \qquad \sigma_i^2 = \sqrt{\frac{1}{3}\left[\left(g_i^0\right)^2 + \left(g_i^1\right)^2 + \left(g_i^2\right)^2\right]}$$

$$\vdots$$

$$\theta_i^{t+1} \leftarrow \theta_i^t - \frac{\eta}{\sigma_i^t} g_i^t \qquad \sigma_i^t = \sqrt{\frac{1}{t+1}\sum_{i=0}^{t}\left(g_i^t\right)^2}$$

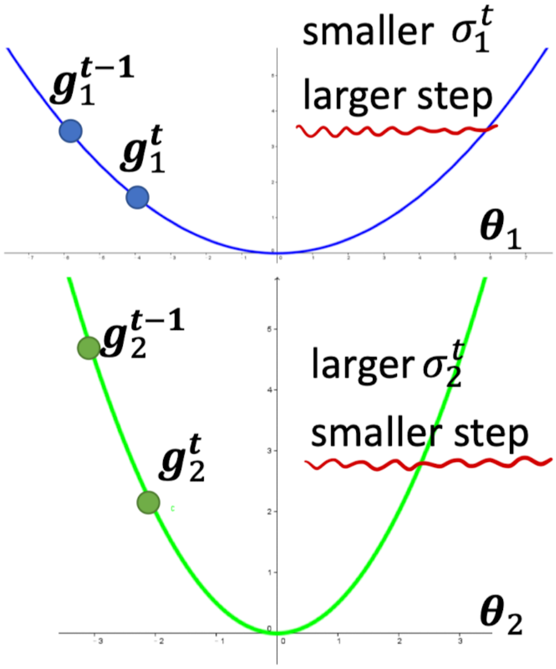
6

why?
参数1的坡度平坦，平均gradient小，step更大；参数2坡度陡峭，平均gradient大，step小

$$\theta_i^{t+1} \leftarrow \theta_i^t - \boxed{\frac{\eta}{\sigma_i^t}} g_i^t$$

$$\sigma_i^t = \sqrt{\frac{1}{t+1}\sum_{i=0}^{t}\left(g_i^t\right)^2}$$

Used in $\boxed{\textbf{Adagrad}}$



2.RMSProp

第一个方法的问题/；就算是一个参数，它需要的learning rate也会随着时间而改变

② RMSProp

$$\boldsymbol{\theta}_i^{t+1} \leftarrow \boldsymbol{\theta}_i^t - \boxed{\frac{\eta}{\sigma_i^t}} \boldsymbol{g}_i^t$$

$$\boldsymbol{\theta}_i^1 \leftarrow \boldsymbol{\theta}_i^0 - \frac{\eta}{\sigma_i^0} \boldsymbol{g}_i^0 \qquad \sigma_i^0 = \sqrt{\left(\boldsymbol{g}_i^0\right)^2}$$

α要能调

$$0 < \alpha < 1$$

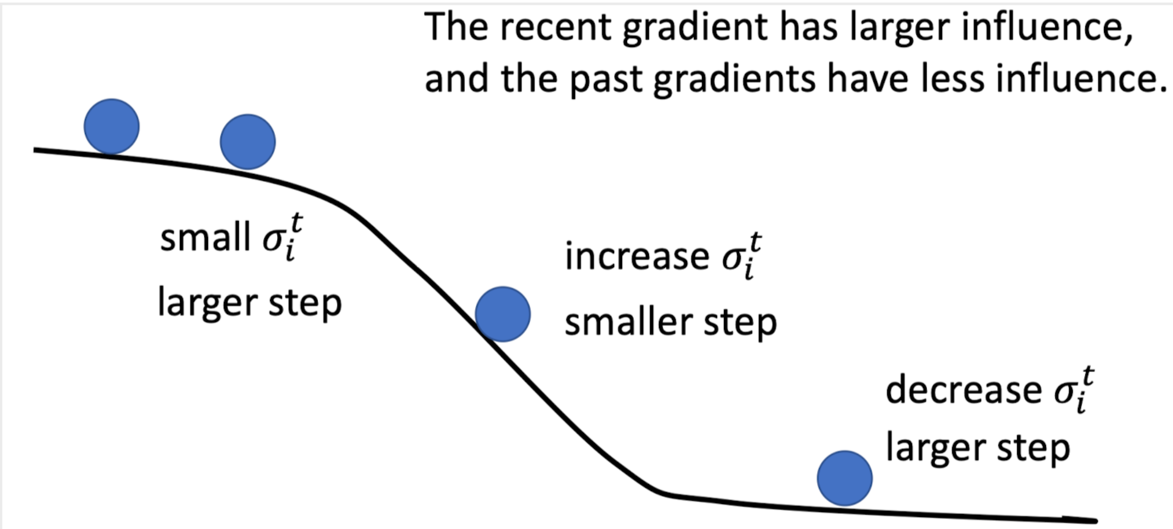
$$\boldsymbol{\theta}_i^2 \leftarrow \boldsymbol{\theta}_i^1 - \frac{\eta}{\sigma_i^1} \boldsymbol{g}_i^1 \qquad \sigma_i^1 = \sqrt{\alpha\left(\sigma_i^0\right)^2 + (1-\alpha)\left(\boldsymbol{g}_i^1\right)^2}$$

$$\boldsymbol{\theta}_i^3 \leftarrow \boldsymbol{\theta}_i^2 - \frac{\eta}{\sigma_i^2} \boldsymbol{g}_i^2 \qquad \sigma_i^2 = \sqrt{\alpha\left(\sigma_i^1\right)^2 + (1-\alpha)\left(\boldsymbol{g}_i^2\right)^2}$$

$$\vdots$$

$$\boldsymbol{\theta}_i^{t+1} \leftarrow \boldsymbol{\theta}_i^t - \frac{\eta}{\sigma_i^t} \boldsymbol{g}_i^t \qquad \sigma_i^t = \sqrt{\alpha\left(\sigma_i^{t-1}\right)^2 + (1-\alpha)\left(\boldsymbol{g}_i^t\right)^2}$$

可以自己调整现在的gradient的权重，从而动态调整learning rate。
如下图，当走到下坡中间的时候，如果是adagrad，它反应会比较慢，step会很大；但如果用RMSProp，把a的值设小一点，让新的刚看到的gradient影响比较大的话，就可以很快让分母变大，就可以很快的让步伐变小。



3.Adam: RMSProp + Momentum

Algorithm 1: *Adam*, our proposed algorithm for stochastic optimization. See section 2 for details, and for a slightly more efficient (but less clear) order of computation. g_t^2 indicates the elementwise square $g_t \odot g_t$. Good default settings for the tested machine learning problems are $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. All operations on vectors are element-wise. With β_1^t and β_2^t we denote β_1 and β_2 to the power t.

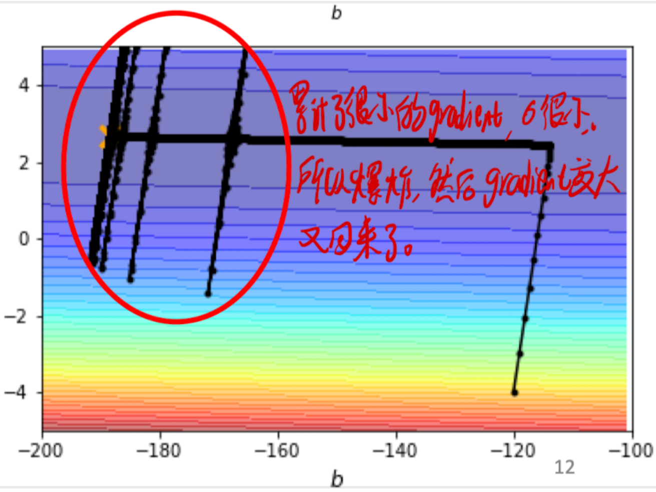
Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector
  $m_0 \leftarrow 0$ (Initialize 1st moment vector) → for momentum
  $v_0 \leftarrow 0$ (Initialize 2nd moment vector) → for RMSprop
  $t \leftarrow 0$ (Initialize timestep)
  while θ_t not converged **do**
    $t \leftarrow t + 1$
    $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep t)
    $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)
    $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)
    $\widehat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)
    $\widehat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
    $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \widehat{m}_t / (\sqrt{\widehat{v}_t} + \epsilon)$ (Update parameters)
  end while
  return θ_t (Resulting parameters)

11

adagrad存在的的问题：会突然爆炸

$$\theta_i^{t+1} \leftarrow \theta_i^t - \boxed{\frac{\eta}{\sigma_i^t}} g_i^t$$

$$\sigma_i^t = \sqrt{\frac{1}{t+1} \sum_{i=0}^{t} (g_i^t)^2}$$



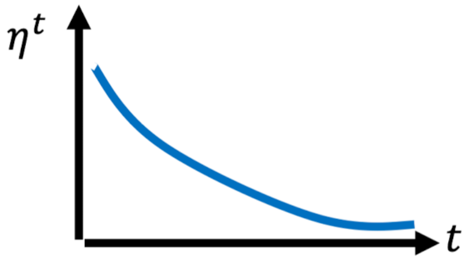
12

Learning rate scheduling
η不是固定的值，而是和时间有关的，不是常数。
- Learning Rate Decay
- Warm up

Learning Rate Scheduling

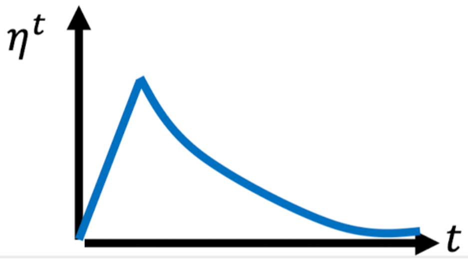
$$\boldsymbol{\theta}_i^{t+1} \leftarrow \boldsymbol{\theta}_i^t - \frac{\boxed{\eta^t}}{\sigma_i^t}\boldsymbol{g}_i^t$$

η与时间有关，越后面η越小。



① ***Learning Rate Decay***

As the training goes, we are closer to the destination, so we reduce the learning rate.



② ***Warm Up***

Increase and then decrease?

learning rate 先变大后变小。

Summary

(Vanilla) Gradient Descent

$$\boldsymbol{\theta}_i^{t+1} \leftarrow \boldsymbol{\theta}_i^t - \eta\boldsymbol{g}_i^t$$

Various Improvements

$$\boldsymbol{\theta}_i^{t+1} \leftarrow \boldsymbol{\theta}_i^t - \frac{\eta^t}{\sigma_i^t}\boldsymbol{m}_i^t$$

③Learning rate scheduling

①Momentum: weighted sum of the previous gradients

考虑大小与方向

Consider direction

②root mean square of the gradients

只考虑大小

only magnitude