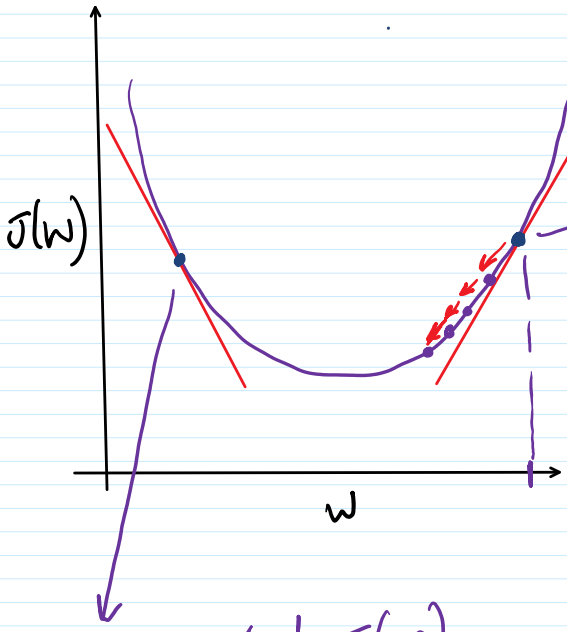
for training faster and getting max/min faster.

Gradient Descent :—



$$W = W - \alpha \frac{d}{dW} J(W)$$

$\frac{d}{dW} J(W) > 0$

$W = W - \alpha \cdot (+ve)$

$W = W - (+ve)$

W ↓ decreasing

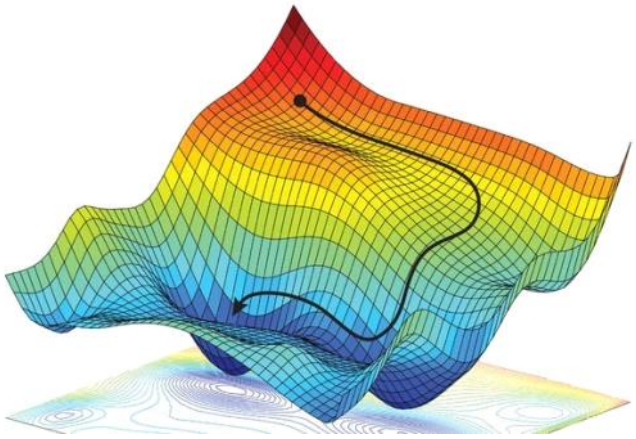
$$W = W - \alpha \frac{d}{dW} J(W)$$

$\frac{d}{dW} J(W) < 0$

$W = W - \alpha \cdot (-ve)$

$= W - (-ve)$

W ↑ increasing



Illy for "b".

Choosing "α".

Too small

Too large.

In general,

$$\theta_{t+1} = \theta_t - \alpha \cdot \left.\frac{\partial L(\theta)}{\partial \theta}\right|_{\theta=\theta_t}$$

$$\theta_{t+1} = \theta_t - \alpha \cdot \nabla_t L(\theta)$$

Let MSE loss for images.

$$L(\theta) = \sum_{j=1}^{L} \sum_{i=1}^{M} (\hat{y}_{ij} - y_{ij})^2$$

① Batch Gradient Descent :-

The entire training set is used in each iteration.

(slow, too heavy)

② Stochastic Gradient Descent :-

The weights are updated after computing gradients for every training sample.

(random directions)

③ Mini-batch Gradient Descent :-

Update weights using batches from the whole dataset.

$$\left(\frac{e}{m}\right)$$

e ← # of samples

m ← # of batches.

① Momentum-based Gradient Descent :-

Nestron Momentum / (1983)

[Nestron Accelerated Gradient]

$$V_t = \gamma V_{t-1} + \alpha \nabla_t L(\theta)$$

$$\theta_{t+1} = \theta_t - V_t \quad \rightarrow ①$$

$$\Rightarrow \theta_{t+1} = \theta_t - [\gamma V_{t-1} + \alpha \nabla_t L(\theta)]$$

$$\theta_{t+1} = \theta_t - \gamma V_{t-1} - \alpha \nabla_t L(\theta)$$

γV_{t-1} → Momentum factor (to get moment to move faster)

⇒ Exponentially decaying average

⇒ Exhibits oscillations near minimum/maximum.

Modify learning rate (α) instead?

→ reduce 'α' after every 'n' iterations

→ Exponential decay

$$\alpha = \alpha_0^{-kt}$$

$$\alpha = \alpha_0 \, e^{-kt}$$

② Ada Grad Optimization

(Adaptive Gradient)

$$r_t = r_{t-1} + (\nabla \theta_t)^2 \quad , \; r_0 \geq 0$$

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\delta + \sqrt{r_t}} \nabla \theta_t$$

δ: tiny number to avoid divide by 0.

⇒ As time progresses, r_t will get to a large value, hence, the moment does not happen.

③ Ada M Optimization

(Adaptive Moments)

$$s_t = \ell_1 s_{t-1} + (1-\ell_1) \nabla \theta_t \qquad 0 \leq \ell_1 \leq 1$$

$$r_t = \ell_2 r_{t-1} + (1-\ell_2)(\nabla \theta_t)^2 \qquad 0 \leq \ell_2 \leq 1$$

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\delta + \sqrt{\tilde{r}_t}} \cdot \tilde{s}_t$$

$$\tilde{r}_t = \frac{r_t}{1+\ell_2^t}, \quad \tilde{s}_t = \frac{s_t}{1-\ell_1^t}$$