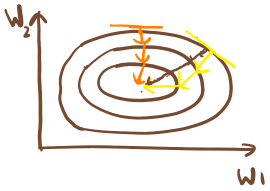
Adaptive Gradient (Adagrad)

Saturday, October 9, 2021 2:18 PM



momentum

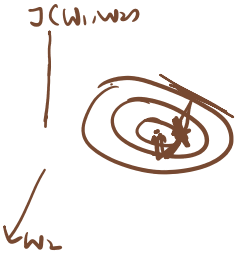
$$m = \beta m + \eta \left.\frac{dL}{dW}\right|_{W=W_0} \rightarrow (m)$$

NAG

$$m = \beta m + \eta \left.\frac{dL}{dW}\right|_{W=(W_0-\beta m)} \rightarrow (NAG)$$

∵ m in both cases is proportional to gradient

here direction is not much of a concern for m



Adagrad ⇒ correct the direction at initial steps

Algorithm ⇒ β_t – scaling factor ⇒ $\beta_t = \beta_{t-1} + \left(\frac{dL}{dW}\right)^2$ where initialize $\beta = 0 \rightarrow (1)$

$$\eta' = \frac{\eta}{\sqrt{\beta_t + \varepsilon}}$$

β = scaling factor

ε – Smoothening factor (avoid division by 0)

$$W_n = W_0 - \eta \frac{\left.\frac{dL}{dW}\right|_W}{\sqrt{\beta_t + \varepsilon}} \rightarrow (11)$$

$\frac{\eta}{\sqrt{\beta_t+\varepsilon}} \cdot \frac{dL}{dW}$ ⇒ Learning rate

Q – Why we need the scaling factor??

$$\beta_t = \beta_{t-1} + \nabla_\theta J(\theta) * \nabla_\theta J(\theta) \rightarrow (1)$$

$\frac{dL}{dW} = \nabla_\theta J(\theta)$

square term – Magnitude of vectors

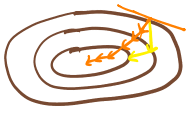
$$W = W - \eta \frac{\nabla_\theta J(\theta)}{\sqrt{\beta_t + \varepsilon}} \rightarrow \text{vector} \rightarrow (11)$$

($\sqrt{\beta_t+\varepsilon} \sim 0$)

$$\text{Unit vector} = \frac{\text{Vector}}{|\text{vector}|}$$

$$W \Rightarrow W - \eta \frac{\nabla_\theta J(\theta)}{\sqrt{\beta_t}} \Rightarrow W - \eta \frac{\nabla_\theta J(\theta)}{\sqrt{|\nabla_\theta J(\theta)|^2}} = W - \eta \frac{\nabla_\theta J(\theta)}{|\nabla_\theta J(\theta)|}$$

Unit vector (scaling to unit)



step 1 ⇒ $W - \eta$ (unit vector)

small value (smaller than $\frac{dL}{dW}$)

$\frac{dL/dW}{(dL/dW)^2}$

Step 1 ⇒ $\beta_0 = 0, W = W_0$

$$\beta_1 = \beta_0 + \left(\frac{dL}{dW}\right)^2 \Rightarrow \beta_1 = \left(\left.\frac{dL}{dW}\right|_{W=W_0}\right)^2$$

$$W_1 = W_0 - \eta \frac{\frac{dL}{dW}}{\sqrt{\beta_1 + \varepsilon}} = W_0 - \eta \frac{\left.\frac{dL}{dW}\right|_{W=W_0}}{\sqrt{\left(\left.\frac{dL}{dW}\right|_{W=W_0}\right)^2 + \varepsilon}} \rightarrow (1)$$

Step 2 ⇒ $\beta_1 = \left(\left.\frac{dL}{dW}\right|_{W=W_0}\right)^2$, $W = W_1$

$$\beta_2 = \beta_1 + \left(\left.\frac{dL}{dW}\right|_{W=W_1}\right)^2 \Rightarrow \left(\left.\frac{dL}{dW}\right|_{W=W_0}\right)^2 + \left(\left.\frac{dL}{dW}\right|_{W=W_1}\right)^2$$

(Previous) (Current)

$$W_2 = W_1 - \eta \frac{\left.\frac{dL}{dW}\right|_{W=W_1}}{\sqrt{\beta_2 + \varepsilon}} \rightarrow (11)$$

$$\sqrt{\beta_2 + \varepsilon}$$

$$\Rightarrow \sqrt{\beta_2 + \varepsilon} \geq \sqrt{\beta_1 + \varepsilon}$$

gradient is adapting based on loss function

Similarly we can generalize (11)

Advantage of Adagrad

1) initially adjust the direction
2) less tuning of learning rate

Disadvantages

1) Stops early before reaching global minimum due to decaying in gradient
2) Takes longer time to converge

* Not recommended *