Deep Learning - Optimizers

Optimizers

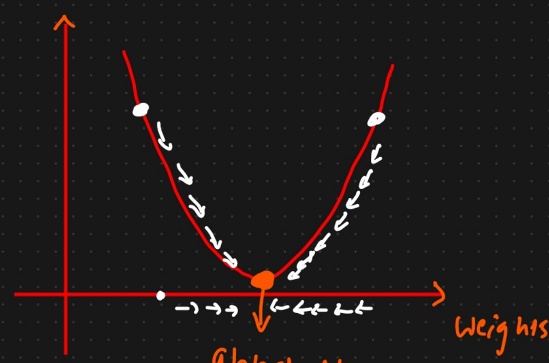
1. Gradient Descent ✓
2. SGD ✓
3. Mini batch SGD ✓
4. SGD With Momentum
5. Adagrad And RMS PROP
6. Adam Optimizers

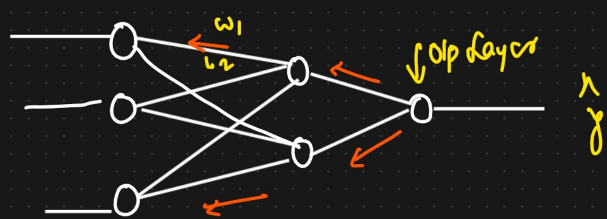
(1) GRADIENT DESCENT

Weight updation Formula

$$\boxed{W_{new} = W_{old} - \eta \frac{\partial L}{\partial W_{old}}}$$

Learning Rate





Optimizers

Loss fn ↓ or cost fn ↓ ⇒ Backpropogation

$\boxed{n = 10000}$

MSE

$\boxed{\text{Loss fn} = (y - \hat{y})^2}$ → 1 Records

$\boxed{\text{Cost fn} = \frac{1}{n} \sum_{i=1}^{n} (y - \hat{y})^2}$ → n Records

Epochs, Iteration

$\boxed{1 \text{ Epoch}}$

10000 → $\hat{y}$ ⇒ Cost function ↓

weight will get update ↓

Disadvantage

n = 1000000 → 64, 8gb

→
←

① Resource Intensive {Huge Ram, High class GPU}

10K * 100

② Stochastic GRADIENT DESCENT

Advantage

① {Solve Resource Issue}

Disadvantage

① Time complexity

② Convergence will be very slow.

n = 1000000

Epoch 1

1 record → ← } → Iteration 1

2 record → ← }

3rd record → ← }

n = 10000

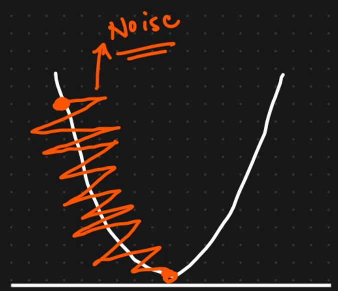
1 Epoch = 10K Iteration

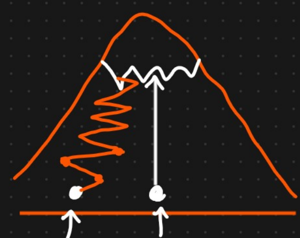
2 Epoch = 10K

$loss = (y - \hat{y})^2$

100 Epochs

10K * 100





SGD

n = 10000

100 Epochs

Epoch 1

1 record → ← loss } Iteration 1

2nd record → ← loss } Iteration 2

⋮

1 Epoch = 10K Iteration

100 Epochs = ?

100 * 10K ← Algebra

Huge Resources

③ Mini batch SGD

batch size

Cost function

⇓

n = 100000

batch-size = 1000

$\frac{1000\cancel{00}}{1\cancel{00}} = 100$ Iteration

Epoch 1

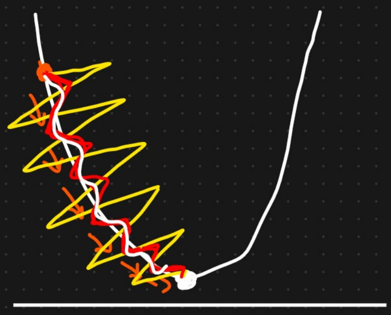
100 records → ←

SGD ⇄ Slow Convergence

Advantage

① Convergence speed will increase

* Noise will be less



(*) Efficient Resource Usage.

⇓

Mini batch SGD → Noise is there

Convergence become faster

(*) SGD with Momentum

$$W_{new} = W_{old} - \eta \frac{\partial L}{\partial W_{old}}$$

$$b_{new} = b_{old} - \eta \frac{\partial L}{\partial b_{old}}$$

$$W_t = W_{t-1} - \eta \frac{\partial L}{\partial W_{t-1}}$$

{Exponential Weighted Average}

⇓

ARIMA, ARMA

⇓

Time Series

Exponential Weighted Average {Smoothening}

t_1 t_2 t_3 t_4 - - - t_n

a_1 a_2 a_3 a_4 - - - a_n

β: Hyperparameter

β: 0 to 1

$\boxed{\beta : 0.95}$ ←

$$V_{t_1} = a_1$$

$$V_{t_2} = \beta * V_{t_1} + (1-\beta) * a_2$$

$$= \beta * a_1 + (1-\beta) * a_2$$

$$= 0.95 * a_1 + 0.05 * a_2$$

$$V_{t_3} = \beta * V_{t_2} + (1-\beta) * a_3$$

$$= \beta \left[0.95\, a_1 + 0.05 * a_2\right] + (1-\beta) * a_3$$

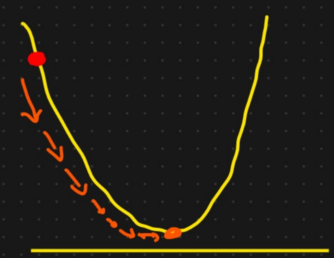
$$= 0.95\,(0.95\, a_1 + 0.05 * a_2) + (0.05) * a_3.$$

Advantage

1. Reduces the noise
2. Quick Convergence

Recap

1. Gradient Descent [Rich] → 1 Epoch = 1 Iteration
2. ↑ SGD { Noise } → 1 Epoch = n Iteration
3. ↑ Mini batch SGD → 1 Epoch = datasize/batch-size
4. SGD with Momentum
5. Adagrad : Adaptive GRADIENT DESCENT



$$\boxed{W_t = W_{t-1} - \eta \frac{\partial L}{\partial W_{t-1}}}$$

$$\boxed{\eta = \text{fixed}}$$

$$\boxed{W_t = W_{t-1} - \eta' \frac{\partial L}{\partial W_{t-1}}}$$

η' → dynamic value

$$\downarrow\downarrow\downarrow \; \eta' = \frac{\eta}{\sqrt{\alpha_t + \epsilon}} \;\uparrow\uparrow\uparrow$$

η → Initial Learning Rate

ϵ → small value

$$\boxed{W_t \approx W_{t-1}} \Leftarrow$$

$$\alpha_t = \sum_{i=1}^{t} \left(\frac{\partial L}{\partial W_t}\right)^2$$

$t=1$ $\quad$ $t=2$ $\quad$ $t=3$ $\quad$ .... $\quad$ Convergence

$\eta = 0.01$ $\quad$ $\eta = 0.005$ $\quad$ $\eta = 0.003$

Exponential Weight Average

⑥ Adadelta And RMSPROP

$t=1$ $\quad$ $S_{dw_t} = 0$ $\quad$ {Dynamic LR + Smoothening}

$$\eta' = \frac{\eta}{\sqrt{S_{dw} + \epsilon}}$$

$t=2$

$$\boxed{S_{dw_t} = \beta * S_{d_{w_{t-1}}} + (1-\beta)\left(\frac{\partial L}{\partial w_{t-1}}\right)^2}$$

$t=3$

$\beta = 0.95$

$$\boxed{w_t = w_{t-1} + \eta' \frac{\partial L}{\partial w_{t-1}}}$$

⑦ Adam optimizer [You should Akam]

SGD with Momentum + RMSPROP [Dynamic LR + Smoothening]

$$\boxed{\begin{aligned} w_t &= w_{t-1} - \eta' V_{dw} \\ b_t &= b_{t-1} - \eta' V_{db} \end{aligned}}$$

$$\boxed{\eta' = \frac{\eta}{\sqrt{S_{dw} + \epsilon}}}$$

$V_{dw} = 0$

Smoothening

$$\boxed{V_{dw_t} = \beta * V_{dw_{t-1}} + (1-\beta)\frac{\partial L}{\partial w_{t-1}}}$$

$$\boxed{V_{db_t} = \beta * V_{db_{t-1}} + (1-\beta)\frac{\partial L}{\partial b_{t-1}}}$$