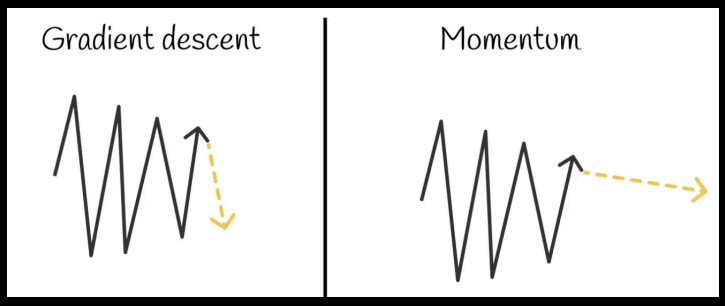
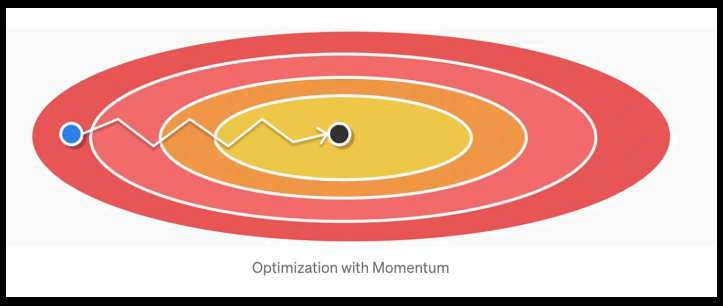
SGD with Momentum





$$w_{new} = w_{old} - \eta \frac{\partial L}{\partial w_{old}} \rightarrow w_t = w_{t-1} - \eta \frac{\partial L}{\partial w_{t-1}}$$

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

$t \rightarrow (t-1)$

$t_{-2}, t_{-1}, (t), t_{+1}, t_{+2}$

Past — old / happend instance

$w_{t-1} \approx w_t$

impact of w_{t-1} in the output of w_t

Time Series

Removal → Noise / Jitter / Oscillations

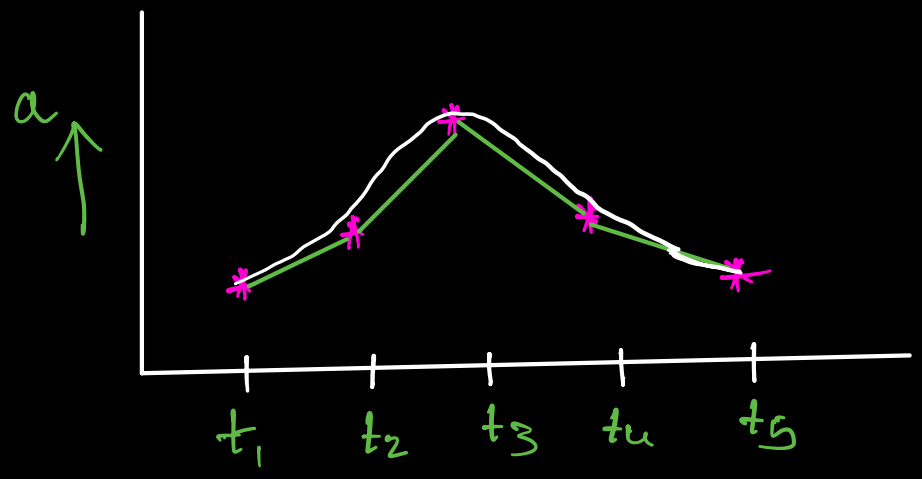
1) Exponential Weighted Average ✓
2) Rolling Mean ✗

Exponential Weighted Average (EWA) Smoothing

Time $t_1\ t_2\ t_3\ t_4 \ldots\ t_n$

Value $a_1\ a_2\ a_3\ a_4 \ldots\ a_n$

t_1 should be controlling t_2



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

$= (0.95 * a_1) + ((1-0.95) * a_2)$

Recommended β value should be close to 1

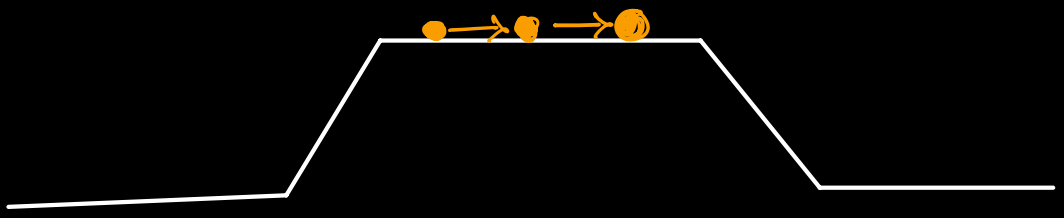
$\beta = 0.95$

$= \{ \underbrace{0.95 * a_1}_{\text{High Impact}} + \underbrace{0.05 * a_2}_{\text{Low impact}} \}$

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

$= 0.95 \{\ \} + (1-\beta) a_3$

Momentum is the push towards Global Minima



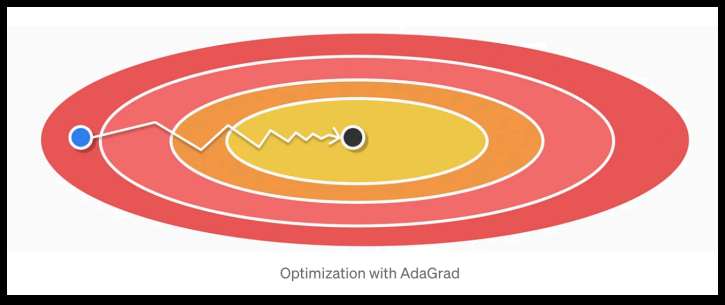
Plateace Region Problems

Advantages of SGD + Momentum

1) Noise Reduction
2) Smoothing of the noise
3) Smooth Convergence

Adagrad (Adaptive Gradient Descent)

learning Rate η = fixed



Optimization with AdaGrad

$$w_n = w_0 - \eta \frac{\partial L}{\partial w_0} \Rightarrow w_n = w_0 - \eta' \frac{\partial L}{\partial w_0}$$

$$\eta' = \frac{\eta}{\sqrt{\alpha^t + \epsilon}}$$

η' = $\frac{\eta}{\text{large value}}$ → very small value

η' = small value

$$\alpha^t = \sum_{i=1}^{t} \left(\frac{\partial L}{\partial w_t}\right)^2$$ → Large value

Slope Squared

Adaptive learning → Dynamic learning Rate

When we start we need high L·R

When we converge we need lower L·R

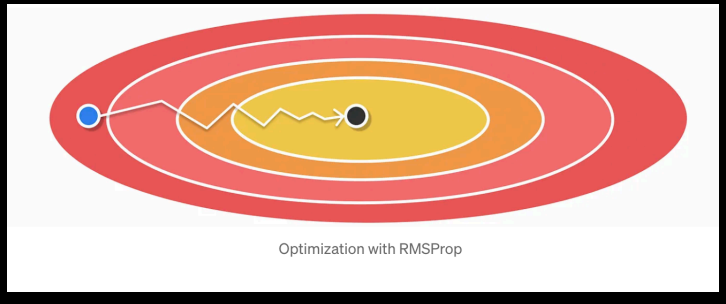
Disadvantage

1) Slow Convergence only in the end.
   ↳ very small update
2) η' = Possibility to come α will be very small
   $\alpha^t \approx 0$

Note :—

- ✓ SGD with Momentum (Academically)
- ✓ Adam (Industry Applications)

RMSProp (Root Mean Squared Propagation)



(EWA) → Exponential Weighted Average

$$\eta' = \frac{\eta}{\sqrt{Sdw_t + \epsilon}}$$

$$Sd_{wt} = 0$$

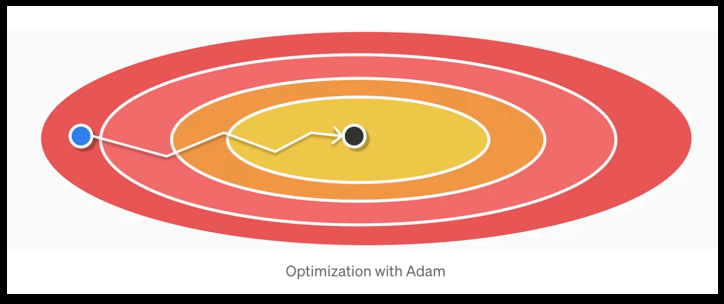
$$= \beta * Sd_{wt-1} + (1-\beta)\left(\frac{\partial L}{\partial w_t}\right)^2 \quad *$$

① SGD with Momentum → No DLR

② Adagrad → DLR

③ RMSProp → SGD with Momentum + Exponential Weighted Avg LR

Adam Optimizer (Adaptive Momentum Estimation)



Optimization with Adam

⇒ RMSProp + SGD with M

D·L·R + SGD with M

$$w_t = w_{t-1} - \eta' V_{dw_t}$$

where, $V_{dw_t} = \beta * V_{dw_{t-1}} + (1-\beta)\left(\frac{\partial L}{\partial w_t}\right)$

$$\eta' = \frac{\eta}{\sqrt{Sd_{wt} + \epsilon}}$$

Bias Calculation

$$b_t = b_{t-1} - \eta \, Vd_{b_t}$$

where $Vd_{b_t} = \beta * Vd_{b_{t-1}} + (1-\beta)\left(\frac{\partial L}{\partial b_t}\right)$

$$\eta' = \frac{\eta}{\sqrt{Sd_t + \epsilon}}$$

Learning Rate Scheduler

Dynamic learning Rate

CLR

Epochs = 100

if epochs $\leq$ 25;
    lr = 0.01
elif epochs > 25 and $\leq$ 50
    lr = 0.001
elif epochs > 50 and $\leq$ 75
    lr = 0.0001
else epochs > 75
    lr = 0.00001

Remaining Topics

1) Back propagation
2) Regularization
3) Callbacks

Task Val Accuracy = 98%
Loss = .02

① MNIST
↳ Different Activation Functions

mish
↳ Optimizers

Final Report

8 Activation, 8 Optimizers
= 64 combinations