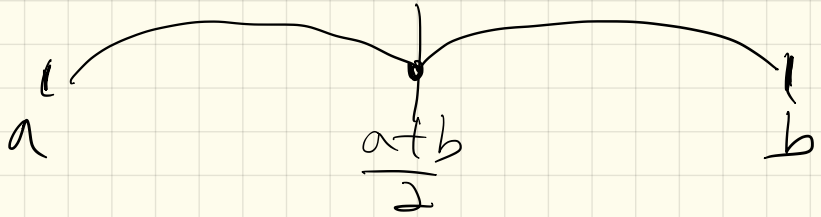
$$\Delta\beta(t+1) = (\eta+\gamma)\left(\delta \Delta\beta(t) + (1-\delta)\nabla f(\theta) \right)$$



$\hat{M}_t$ vs M_t

$M_0 = 0$

$M_1 = (1-\eta_1)\nabla L(\beta(1))$

$M_2 = \eta_1(1-\eta_1)\nabla L(\beta(1)) + (1-\eta_1)\nabla L(\beta(2))$

$M_3 = \eta_1^2(1-\eta_1)\nabla L(\beta(1)) + \eta_1(1-\eta_1)\nabla L(\beta(2)) + (1-\eta_1)\nabla L(\beta(3))$

---

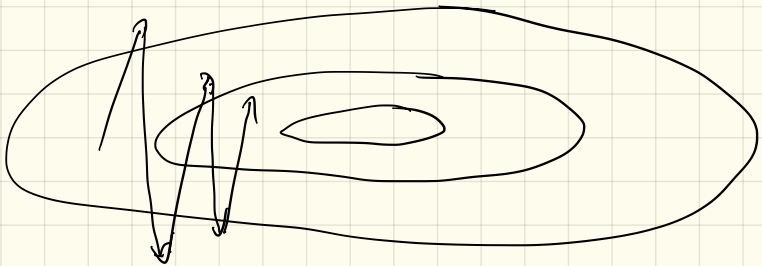
a, b, c

$\frac{1}{3}a + \frac{1}{3}b + \frac{1}{3}c$

$\Rightarrow \frac{(1-\eta_1)(1-\eta_1^3)}{1-\eta_1} = 1-\eta_1^3$

Optimization

⇒ GD ⇐ 1st order optimization



zig-zag

γ ↑ step size

learning rate.

⇒ 2nd order Newton's method.

$$M_t = \frac{1}{t}\sum_{i=1}^{t} \nabla L(\beta(i))$$

$$G_t = \frac{1}{t}\sum_{i=1}^{t} \left(\nabla L(\beta(i))\right)^2 \leftarrow \text{element wise}$$

Momentum : M_t

AdaGrad : G_t

RMSProp : G_t

Adam : M_t, G_t

$$\beta(t+1) = \beta(t) - \frac{\gamma M(t)}{\sqrt{G_t}} \leftarrow \text{element wise.}$$

$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad X^2 = \begin{bmatrix} x_1^2 \\ \vdots \\ x_n^2 \end{bmatrix}, \quad \sqrt{X} = \begin{bmatrix} \sqrt{x_1} \\ \vdots \\ \sqrt{x_n} \end{bmatrix}$$

① random shuffle.