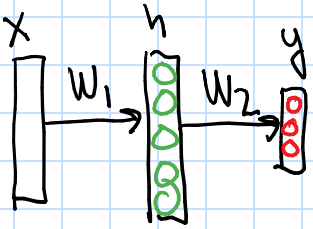
Question from Piazza

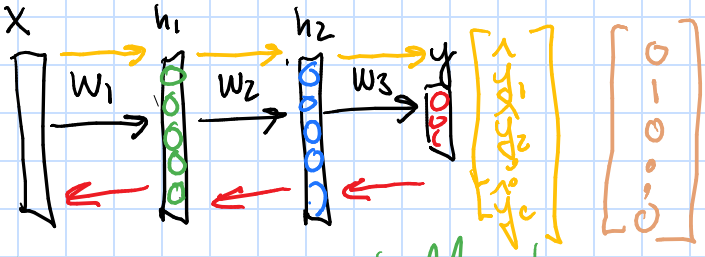
Exam #2 : 04/09 - 04/12



MLP w/ one hidden layer

$$h = f(W_1 x + b_1)$$
$$\hat{y} = g(W_2 h + b_2)$$

where $x \in \mathbb{R}^D$, $W_1 \in \mathbb{R}^{K \times D}$, $b_1 \in \mathbb{R}^{K \times 1}$, $h \in \mathbb{R}^{K \times 1}$, $W_2 \in \mathbb{R}^{C \times K}$, $y \in \{0,1\}^{C \times 1}$



MLP w/ two hidden layers

$$h_1 = f_1(W_1 x + b_1)$$
$$h_2 = f_2(W_2 h_1 + b_2)$$
$$y = g(W_3 h_2 + b_3)$$

$$\delta_k(n) = e_k(n)\, \varphi_k'(v_k(n))$$

$$w_{ji} = N(0, \sigma)$$

Local Gradients

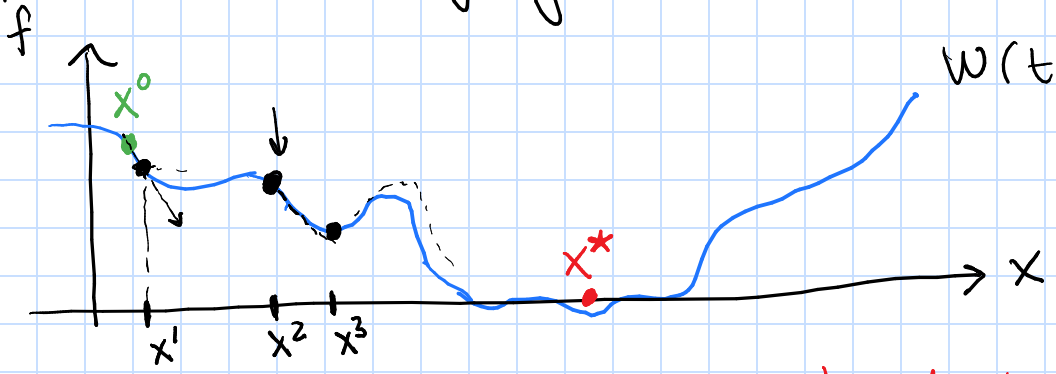
$$\delta_j(n) = e_j(n)\, \varphi_j'(v_j(n))$$

- or -

$$\delta_j(n) = \varphi_j'(v_j(n)) \sum_k \delta_k(n) w_{kj}(n)$$

Momentum

- Find the weights of a neural net is a non-convex optmization problem. We are going to find a local minimum.



$$w(t+1) = w(t) + \Delta w(t)$$

Classical Momentum

$$v_{t+1} = \mu v_t - \varepsilon \nabla f(x_t)$$

$$x_{t+1} = x_t + v_{t+1}$$

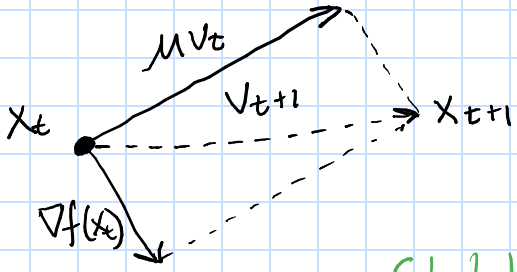
convergence $\rightarrow \mathcal{O}(1/T)$

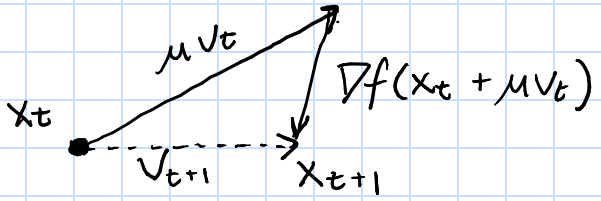
Nesterov's Accelerated Gradient

$$v_{t+1} = \mu v_t - \varepsilon \nabla f(x_t + \mu v_t)$$

$$x_{t+1} = x_t + v_{t+1}$$

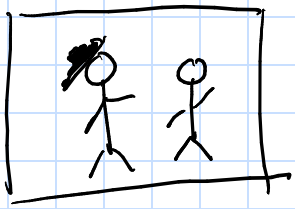
convergence $\rightarrow \mathcal{O}(1/T^2)$



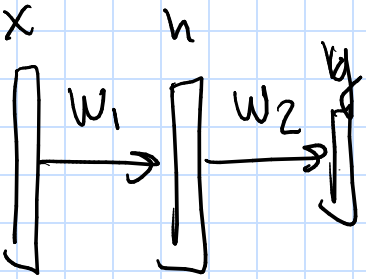


Stutskever et al. "On the importance of initialization and momentum", ICML, 2013.

Deep Learning



$x =$ raw input vector $\rightarrow$ slightly higher level representation $\rightarrow \cdots \rightarrow$ higher level representations



people $\rightarrow$ (even higher level of representations) $\rightarrow$ waiting

$$\delta_j(n) = \varphi_j'(v_j(n)) \sum_k \delta_k(n) w_{kj}(n)$$

previous layers

Traditional ML: $f: x \rightarrow y$

New Idea: $x \mapsto h \rightarrow y$

Model the hidden/latent layers.