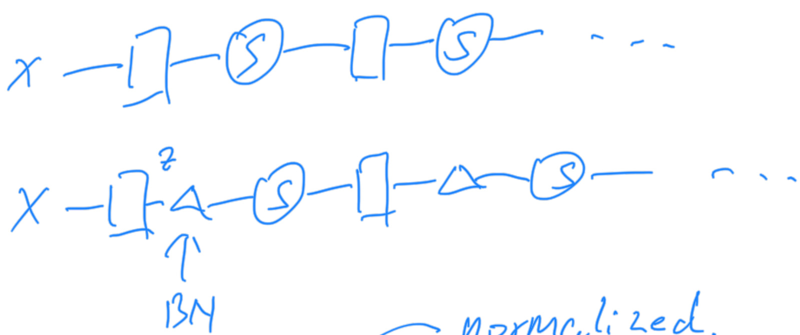
Lecture 4

1. Batch normalization



$z = WX$ normalized.

$$z = ax + b = \underbrace{\begin{bmatrix} a & b \end{bmatrix}}_{W} \underbrace{\begin{bmatrix} x \\ 1 \end{bmatrix}}_{X}$$

Batch size m

$$z_1 = WX_1, \quad z_2 = WX_2, \quad \cdots, \quad z_m = WX_m$$

$$\mu = \frac{1}{m}\sum_{1}^{m} z_i, \quad \tilde{z}_j = z_j - \mu$$

$$\sigma = \frac{1}{m}\sum_{1}^{m} \tilde{z}_j^2$$

$$\hat{z}_j = \frac{\tilde{z}_j}{\sqrt{\sigma + \epsilon}}$$

$$y = \gamma \hat{z} + \beta$$

$H = \alpha z_j + \beta$

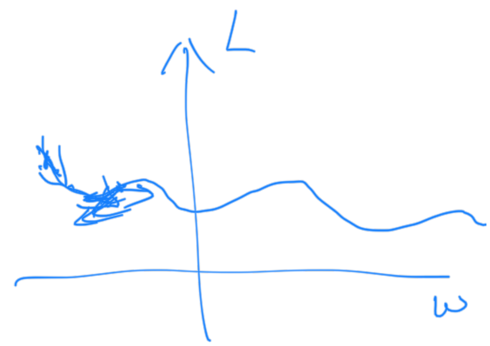
learned

2. Optimization

Gradient descent

$\rightarrow w_{t+1} = w_t - \lambda \frac{\partial L}{\partial w_t}$

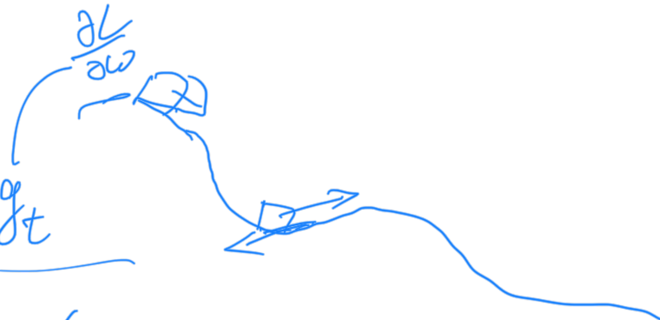
$L = \sum_{i=1}^{m} L_i$



m — full dataset

SGD m — batch size

Momentum



GD : $w_{t+1} = w_t - \lambda g_t$

momentum : $u_{t+1} = \gamma u_t - \lambda g_t$

$w_{t+1} = w_t + u_{t+1}$

momentum example : $\gamma = 0.9$, $u_0 = 0$

$u_1 = 0.9 \cdot 0 - \lambda g_0 = -\lambda g_0$

$w_1 = w_0 - \lambda g_0$

$t = 2$ $\quad u_2 = 0.9 \cdot u_1 - \lambda g_1$

$$= -0.9\lambda g_0 - \lambda g_1 = -\lambda(0.9 g_0 + g_1)$$

$$W_2 = W_1 - \lambda(\underline{0.9 g_0} + \underline{g_1})$$

γ change from 0.5 to 0.9 over iterations.

Adam

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$

$$V_t = \beta_2 V_{t-1} + (1-\beta_2) g_t^2$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \quad \hat{V}_t = \frac{V_t}{1-\beta_2^t}$$

$$U_t = -\lambda \frac{\hat{m}_t}{\sqrt{\hat{V}_t} + \varepsilon}$$

$$W_{t+1} = W_t + U_t$$

$\beta_1 = 0.9$
$\beta_2 = 0.999$
$\varepsilon = 10^{-8}$

input → $\boxed{NN}$ → output   Loss

3. convolutional NN

a. convolution

One-D convolution

$$(X * w)(t) = \sum_{a=-\infty}^{\infty} x[a]\,w[t-a]$$

Two-D conv

$$(I * K)(i,j) = \sum_m \sum_n I[m,n]\,K[i-m, j-n]$$

$$= \sum_m \sum_n I[i-m, j-n]\,K[m,n]$$

cross-correlation

$$(I * K)(i,j) = \sum_m \sum_n I[i+m, j+n]\,K[m,n]$$

$$I = [1\ 2\ 3\ 2\ 1] \qquad K = [1\ 0\ -1]$$

$$I * K = [-2\ 0\ 2]$$

$$I = \begin{bmatrix} 1 & 2 & 1 & 3 & 0 \\ 2 & 0 & 1 & 1 & 2 \\ 3 & 2 & 1 & 0 & 1 \\ 1 & 1 & 2 & 1 & 1 \end{bmatrix} \qquad K = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

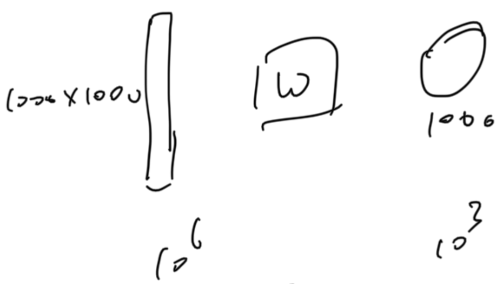
$$\sum \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 1 \\ 3 & 2 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} = 0$$

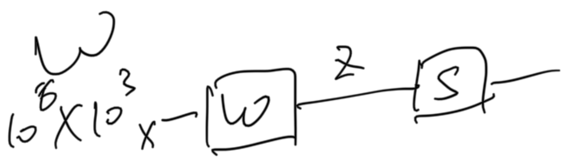
$$\sum \begin{bmatrix} 2 & 1 & 3 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} = 4$$

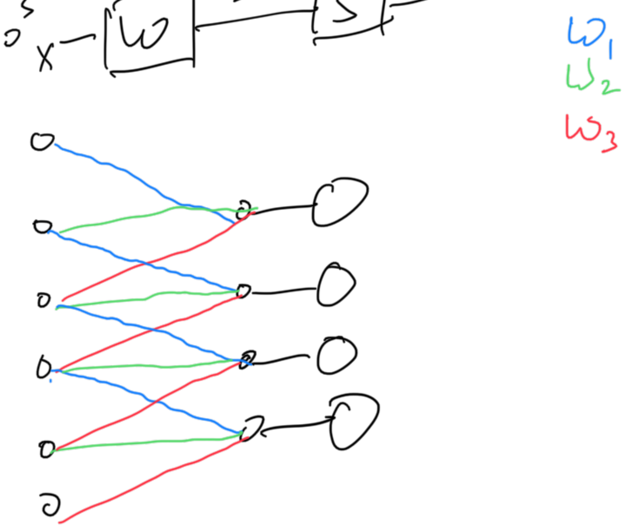
$$Z_3 \begin{bmatrix} 0 & 1 & 1 \\ 2 & 1 & 0 \end{bmatrix} \odot \begin{bmatrix} \cdots \\ -1 & 0 & 1 \end{bmatrix} = 1$$

$$Z_2 \begin{bmatrix} 1 & 3 & 0 \\ 1 & 1 & 2 \\ 1 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} = 2$$

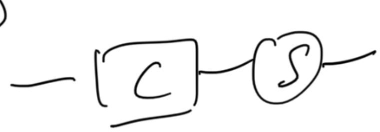
$$= \begin{bmatrix} 0 & 4 & 2 \\ 6 & & \\ & & \end{bmatrix}$$







$\Rightarrow$



$$(x_{i+1} \cdots x_{i+m}) \begin{bmatrix} w_1 \\ \vdots \\ w_m \end{bmatrix} = z_i$$

$$\frac{\partial z_i}{\partial w_j} = x_{i+j}$$

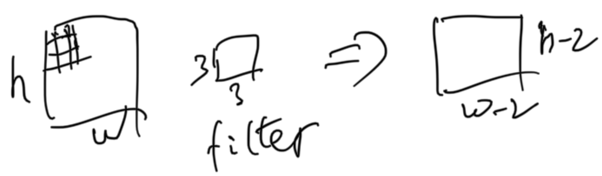
Term

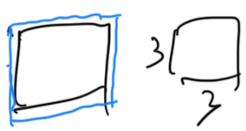
receptive field aka.

filter size

3x3, 5x5, 7x7,

padding size





padding # = 1 for each side.

stride — step size, S

filter size — F

padding size — P

input image $w \times h$

size of output after conv

output

output size $m \times n$

$$m = \frac{(W - F + 2P)}{S} + 1$$

$$n = \frac{(h - F + 2P)}{S} + 1$$