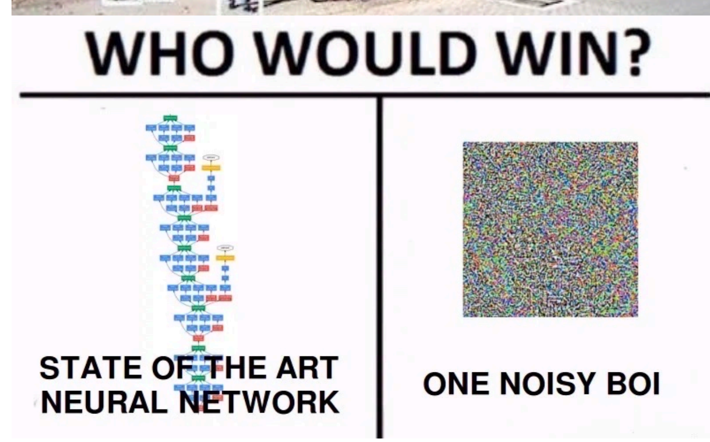
March 24, 2023

DSML: Computer Vision.

CNNs: Operations under the hood (continued)

Class starts @ 9:05 pm.



Recap:

* We trained a CNN, and observed overfitting.

* Steps taken to deal with the problem:

  (a) Architecture change: Reduce size of input to the Dense layers; use more blocks of conv 2D + maxpooling.

  (b) Train longer: 10 epochs were insufficient; train until validation accuracy plateaus.

  (c) Regularization: Add batch normalization to make the model converge faster, drop-out to make it more robust.

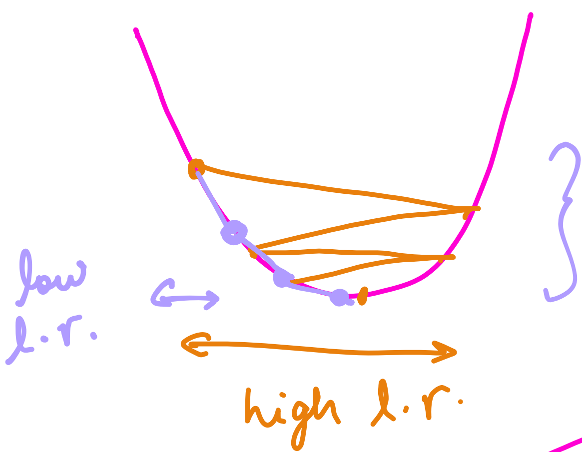
Agenda for today:

* Continue exploring solutions to improve accuracy:
  (a) Learning Rate Schedule: Reduce learning rate on plateau, perform early stopping.

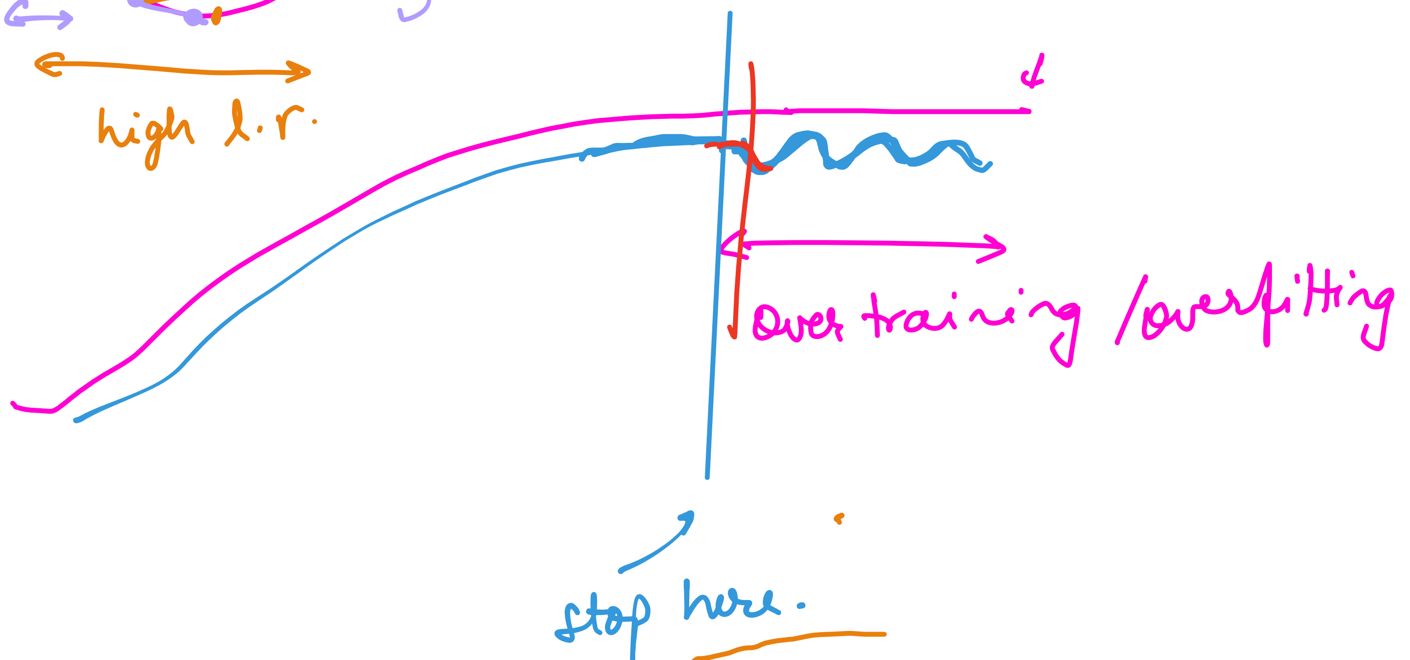
  (b) L1 - L2 regularization: Prevent weights from fluctuating.

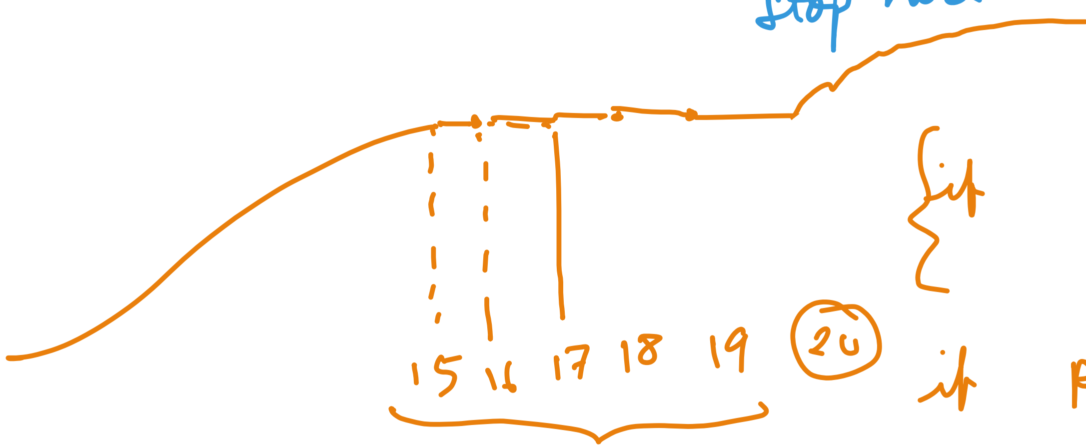
  (c) Increase training data size through augmentation: Use crops, shifts, flips, rotate, etc.

* Implement backpropagation for a CNN from scratch. Math + Code!!

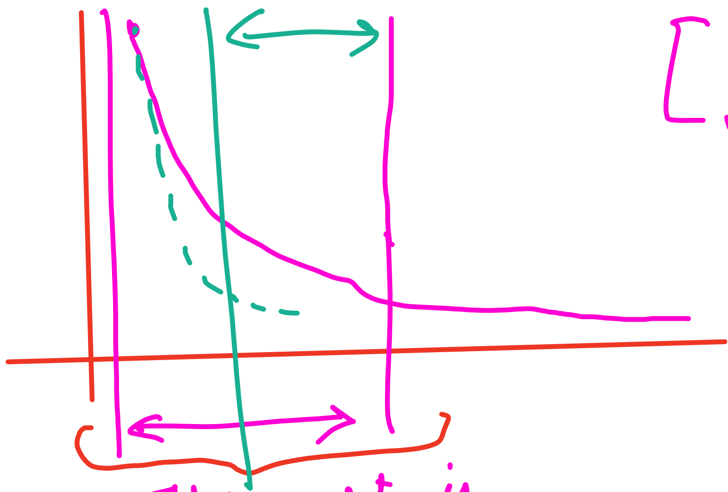


$$\frac{loss^{(t+1)}}{loss^{(t)}} \leq \delta$$





if $|loss^{(t+1)} - loss^{(t)}| \leq \delta$:
    plateau += 1
if plateau == patience:
    → reduce l.r.



This part is just to bring the loss down.

[3.2   8.0   12.1]

↓

[2.33e-3   1.67e-4   12.1e-8]

90% of the cases, we see that the magnitude of our is changing more than the direction.

→ $\boxed{\text{L1 \& L2 regularization}}$

$$\overset{d}{\begin{bmatrix} k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 \\ k_7 & k_8 & k_9 \end{bmatrix}} \quad \approx 100$$

$$\underline{L1} : \quad L' = \underbrace{L}_{\approx 98} + \lambda \underbrace{\sum_{i=1}^{9} |k_i|}_{900.}$$

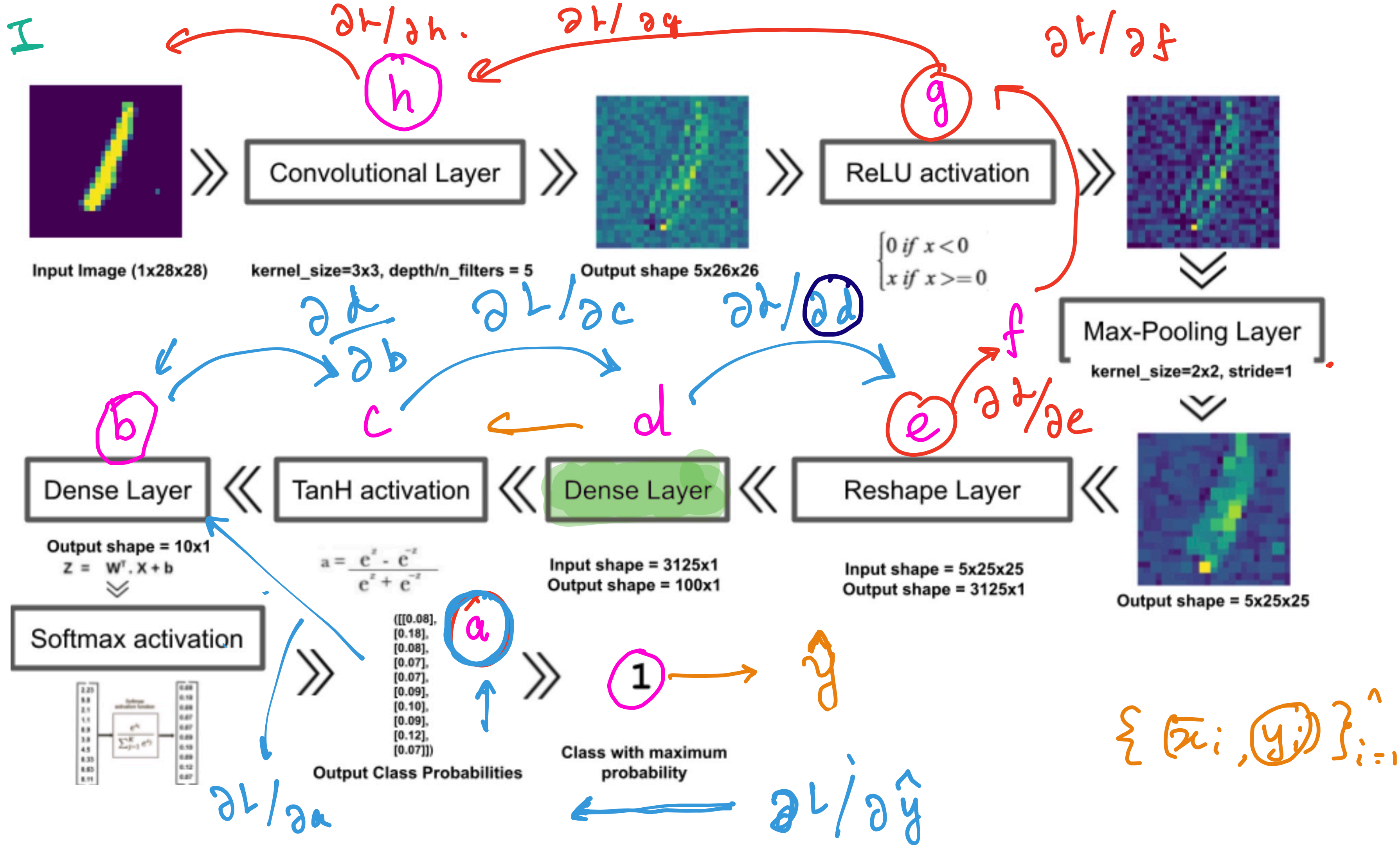
1

L : original loss.

$$\underline{\underline{L_2}} : \quad L' = \underbrace{L}_{\approx 98} + \lambda \underbrace{\sum_{i=1}^{9} k_i^2}_{90000}$$

$$1.1 \xrightarrow{w} (1.1)^2 \xrightarrow{w} (1.1)^3 \rightarrow (1.1)^{20}$$

$$0.9 \xrightarrow{w} (0.9)^2 \xrightarrow{w} (0.9)^3 \rightarrow \dots (0.9)^{20}$$



<u>Loss</u> : Categorical Cross Entropy / Log Loss.

$$\mathcal{L}(\theta) = -\sum_{i=1}^{n} y_i \log(\hat{y}_i)$$

$$d = f(\hat{y})$$

$$\hat{y} = \text{argmax}(a)$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \leftarrow \begin{bmatrix} 0.5 \\ 0.33 \\ 1 - 0.83 \end{bmatrix} \rightleftarrows 0.5$$

$$a = \text{softmax}(b)$$

$$b = \tanh(c)$$

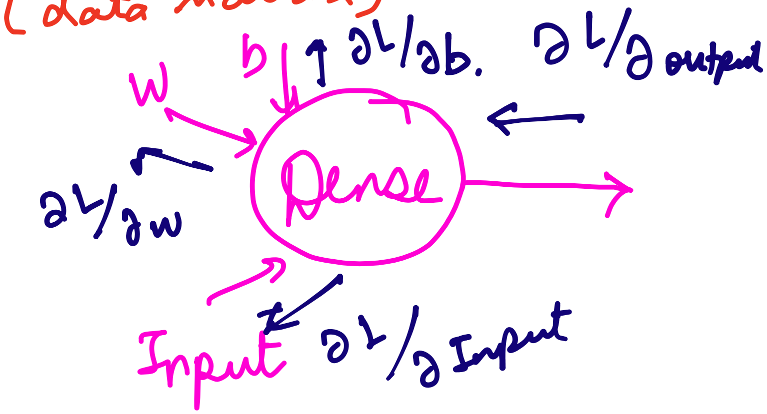
$$c = \text{Dense}(d).$$

Dense :

Input $\rightarrow$ X (data matrix)

Weights $\rightarrow$ W

bias $\rightarrow$ $\bar{b}$



broadcast addition.

$$\underline{W.X + \overbrace{\bar{b}}}$$

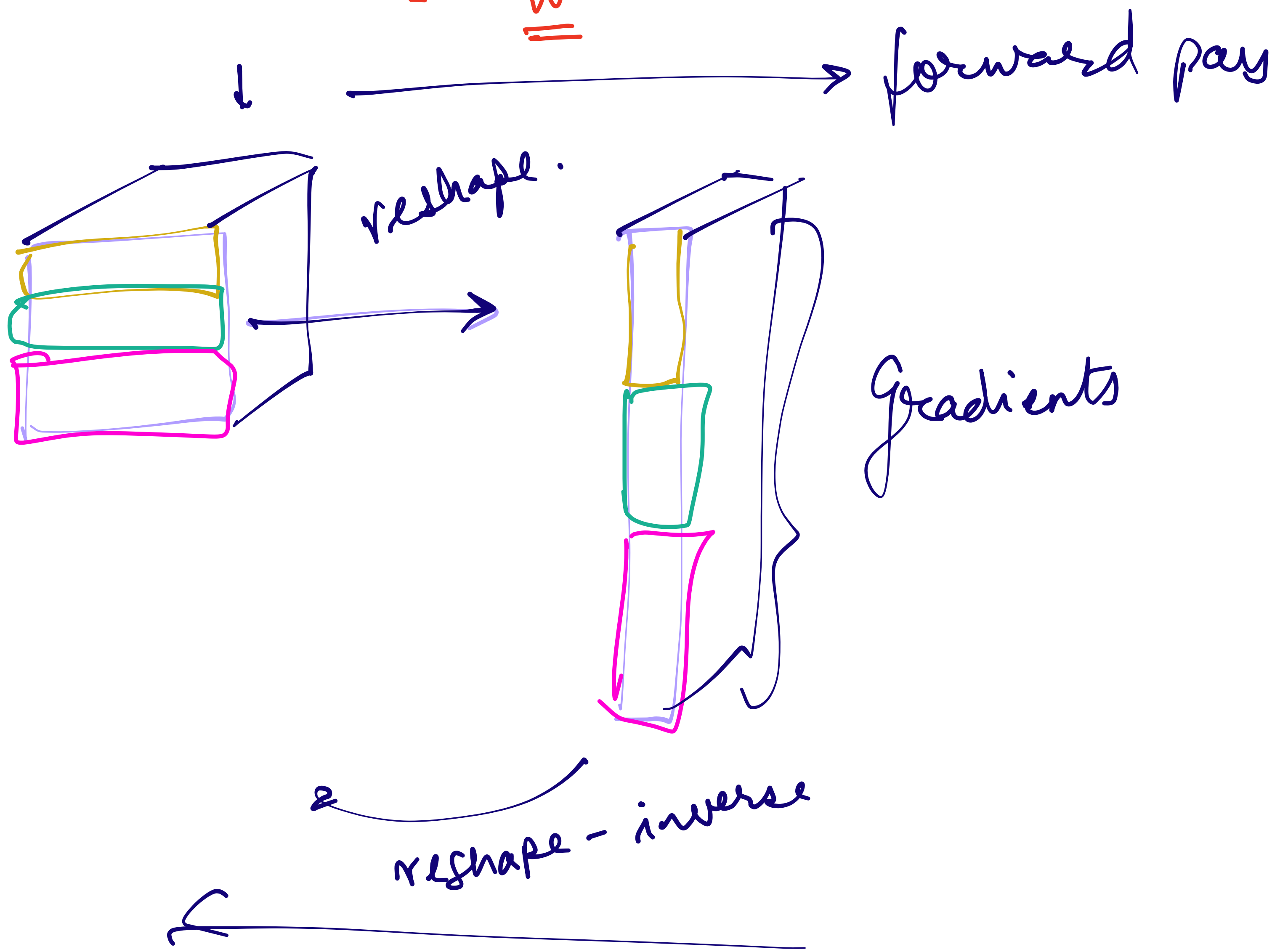
$$\nabla_{X}(\cdot) = X^{T} \qquad \nabla_{\bar{b}}(\cdot) = 1$$

$$\nabla_{\bar{x}}(Ax) = A^{T}$$

$$\frac{\partial L}{\partial e} = \underbrace{\frac{\partial L}{\partial d}} \cdot \frac{\partial d}{\partial e}$$

Input gradient $= \nabla_x (Wx + b)$

$= \underline{\underline{W^T}}$



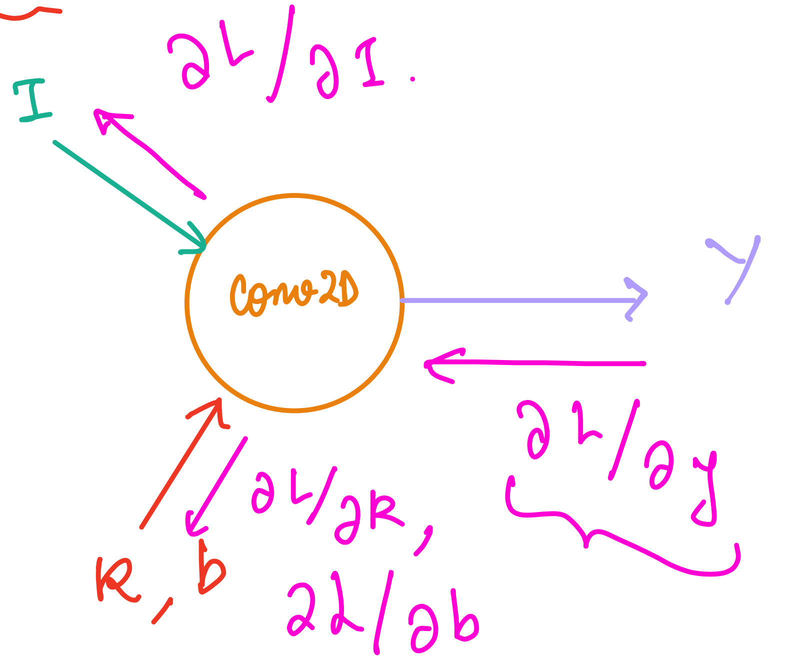
$$\begin{bmatrix} 0 & 9 \\ 0 & 0 \end{bmatrix}$$

$$\max\left(\begin{array}{|c|c|}\hline 0 & 0.08 \\ \hline 0 & 0 \\ \hline\end{array}\right) = \boxed{0.08}$$

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{o/w.} \end{cases}$$

$$f'(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{o/w.} \end{cases}$$

Convolution Layer:



$$\frac{\partial L}{\partial y} \times \frac{\partial y}{\partial k}, \rightarrow x.$$

$$\frac{\partial L}{\partial K_{11}} = \frac{\partial L}{\partial Y_{11}} \cdot X_{11} + \frac{\partial L}{\partial Y_{12}} \cdot X_{12} + \frac{\partial L}{\partial Y_{21}} \cdot X_{21} + \frac{\partial L}{\partial Y_{22}} \cdot X_{22}$$

$$\frac{\partial L}{\partial K_{12}} = \frac{\partial L}{\partial Y_{11}} \cdot X_{12} + \frac{\partial L}{\partial Y_{12}} \cdot X_{13} + \frac{\partial L}{\partial Y_{21}} \cdot X_{22} + \frac{\partial L}{\partial Y_{22}} \cdot X_{23}$$

$$\frac{\partial L}{\partial K_{21}} = \frac{\partial L}{\partial Y_{11}} \cdot X_{21} + \frac{\partial L}{\partial Y_{12}} \cdot X_{22} + \frac{\partial L}{\partial Y_{21}} \cdot X_{31} + \frac{\partial L}{\partial Y_{22}} \cdot X_{32}$$

$$\frac{\partial L}{\partial K_{22}} = \frac{\partial L}{\partial Y_{11}} \cdot X_{22} + \frac{\partial L}{\partial Y_{12}} \cdot X_{23} + \frac{\partial L}{\partial Y_{21}} \cdot X_{32} + \frac{\partial L}{\partial Y_{22}} \cdot X_{33}$$

$$\boxed{\frac{\partial L}{\partial K_i'} = \mathrm{Conv}\left(X, \frac{\partial L}{\partial Y_i}\right)}$$

$$\Uparrow$$

$$\Rightarrow \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{21} & X_{22} & X_{23} \\ X_{31} & X_{32} & X_{33} \end{bmatrix} \underset{\text{conv}}{\circledast} \begin{bmatrix} \frac{\partial L}{\partial Y_{11}} & \frac{\partial L}{\partial Y_{12}} \\ \frac{\partial L}{\partial Y_{21}} & \frac{\partial L}{\partial Y_{22}} \end{bmatrix} = \begin{bmatrix} \frac{\partial L}{\partial K_{11}} & \frac{\partial L}{\partial K_{12}} \\ \frac{\partial L}{\partial K_{21}} & \frac{\partial L}{\partial K_{22}} \end{bmatrix} = \frac{\partial L}{\partial K}$$

$$\Rightarrow \boxed{\frac{\partial L}{\partial K} = \mathrm{Conv}\left(X, \frac{\partial L}{\partial Y}\right)}$$

$$\begin{bmatrix} \frac{\partial L}{\partial y_{11}} \; X_{11} & \partial L/\partial y_{12} \; X_{12} & X_{13} \\ \partial L/\partial y_{21} \; X_{21} & \partial L/\partial y_{22} \; X_{22} & X_{23} \end{bmatrix}$$

$$X_{31} \qquad X_{32} \qquad X_{33}$$

$$\frac{\partial L}{\partial K} = \underbrace{\mathrm{conv}\left(X, \frac{\partial L}{\partial Y}\right)}$$

In signal processing:

Convolution means $\underbrace{\text{flip.lr}}_{\text{Flip left-right.}}(\text{flip.ud}(K))$

K And then apply correlation.

What we use in deep learning is actually called correlation in signal processing!!

$$K = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \xrightarrow{\text{flip lr}} \begin{bmatrix} 2 & 1 \\ 4 & 3 \end{bmatrix} \xrightarrow{\text{flipud}} \begin{bmatrix} 4 & 3 \\ 2 & 1 \end{bmatrix}$$

(signal processing)

$\boxed{\text{Convolution}}$

(Deep learning)

$\boxed{\text{Convolution}}$

Final test: Are we learning anything?

```
Accuracy of the Network on Test data is 98.67612293144208 %
```

Epoch 0 → `0.5383138776615902`

1 → `0.16221502628227547`

2 → `0.11184481654332604`

3 → `0.08838307171782168`

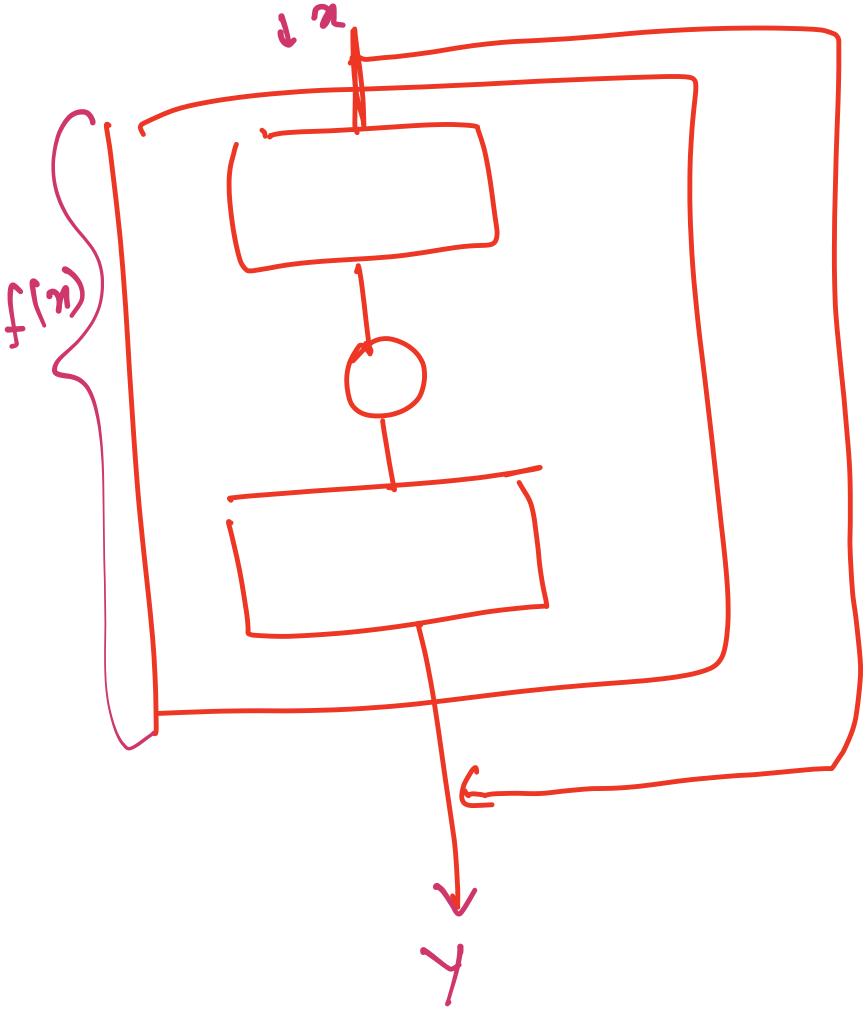
4 → `0.07595734223282834`

5 → `0.06734348866838971`

6 → `0.061430346133807`

7 → `0.05690578701515457`

YES!!



$$y = f(x) + x$$

$$\frac{\partial y}{\partial x} = \underline{\underline{f'(x) + 1}}$$

Andrew Ng → Great as a first course in CNN
(if you're from CS background)

→ Research papers → Not the best option.

Go through Stanford CNN course
by Fei-Fei Li (CS 231n)