BLG561E FALL 2021

Deep Learning

09.11.2021

Gözde ÜNAL

3 components of an general ML algorithm : DL is a sub-class

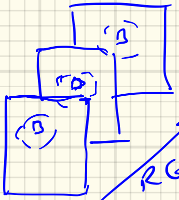
1) Hypothesis function :
2) Loss function :
3) Optimization : optime loss fn.

Today: CONVOLUTIONAL NEURAL NETWORKS (CNNs)

★ Only the hypothesis functions change, i.e. the 1 st component ONLY

Then you can pick 2 & 3 freely according to your task.

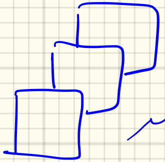
Now → Say, you have large (spatial) data, let's use a FCN / Fully Connected hypothesis



RGB channels → 3

FCN

$\underline{W}^1$



6 channels

$\underline{\underline{X}}$: $\underbrace{32 \times 32 \times 3}_{\text{very small image}} \Rightarrow \underbrace{3072 \times 1}_{X}$

a^1 = activation 1st layer : $28 \times 28 \times 6 = 4074 \times 1$

$\underline{a}^1 = \underline{\underline{W}}^1 \underline{X}$

$\underline{\underline{W}}^1 : \underbrace{3072 \times 4704}$ → $\approx$ 14 Million ! parameters.

e.g. if we had $256 \times 256 \times 3$ $\approx$ 600K dimension

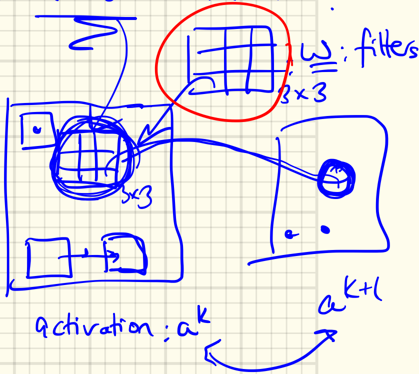
→ FCNs would need a huge # parameters !!

⇒ CNNs can handle large spatial data: Q: How?

They restrict the weights in 2 ways: (in contrast to FCNs)

1) Activations between layers occur in a "local" manner. (sparsity):

2) Parameter sharing: All activations share the same weight.



1st component:

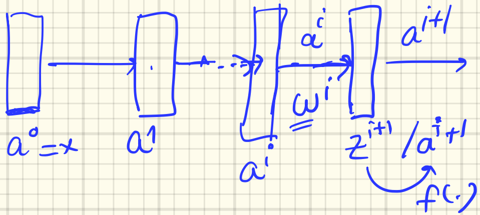
New hypothesis function for CNNs:

(1) $z^{i+1} = \underline{w}^i * a^i + b^i$

$a^{i+1} = f(z^{i+1}) = f(w^i * a^i + b^i)$

$*$: convolution operation

f: ReLU or other nonlinearity



Compare

CNN

convolution

$$z^{i+1} = \underline{\underline{w}}^{i} * \underline{a}^{i} + \underline{b}^{i}$$

$$a^{i+1} = f(\underline{\underline{w}}^{i} * \underline{a}^{i} + \underline{b}^{i}) = f(z^{i+1})$$

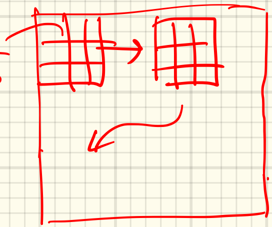
FCN

matrix. vector multiplication

$$z^{i+1} = \underline{\underline{w}}^{i} \cdot \underline{z}^{i} + \underline{b}^{i}$$

$$a^{i+1} = f(z^{i+1})$$

eg. w^i { 3x3

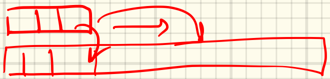


(1) locality (sparsity)

(2) shered connections

Note: other components can be added $\left\{ \begin{array}{l} \text{maxpool} \\ \text{BN} \quad \text{etc.} \end{array} \right.$

1D conv



For illustration:

→ 1-D CNN. $\boxed{a^{i+1} = f(w^i * a^i + b^i)}$

Recall: FC layer

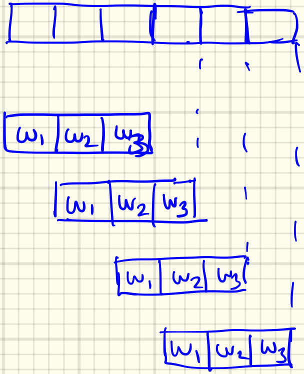
$$a^{i+1} = f(w^i \cdot a^i + b^i)$$

Say $w^i = (w_1, w_2, w_3)$: 1D filter

Input: $\underset{6\times1}{a^i}$ ⇒

$\underset{3\times1}{w^i}$ filter:

4 possible "valid" shifts



Write convolution to a matrix operator

$$\underline{\underline{w^i}} = \begin{bmatrix} w_1 & w_2 & w_3 & 0 & 0 & 0 \\ 0 & w_1 & w_2 & w_3 & 0 & 0 \\ 0 & 0 & w_1 & w_2 & w_3 & 0 \\ 0 & 0 & 0 & w_1 & w_2 & w_3 \end{bmatrix} \quad 4\times6$$

$$\text{CNN: } \underbrace{a^{i+1} = f(w^i \cdot a^i + b^i)}_{\text{FCN}}$$

What happens at backprop?

Recall for FC: backprop:

upstream gradient: $$\frac{\partial l}{\partial a^i} = \underline{g^i} = \underbrace{(w^{i+1})^T}_{6\times4} \cdot \underbrace{\underline{g^{i+1}}}_{4\times1} \cdot \underline{f'(z^{i+1})}$$

→ let's transpose in CNN backprop :

$$\underline{\underline{W}}_i^T{}_{6\times 4} = \begin{bmatrix} w_1 & 0 & 0 & 0 \\ w_2 & w_1 & 0 & 0 \\ w_3 & w_2 & w_1 & 0 \\ 0 & w_3 & w_2 & w_1 \\ 0 & 0 & w_3 & w_2 \\ 0 & 0 & 0 & w_3 \end{bmatrix}_{6\times 4}$$

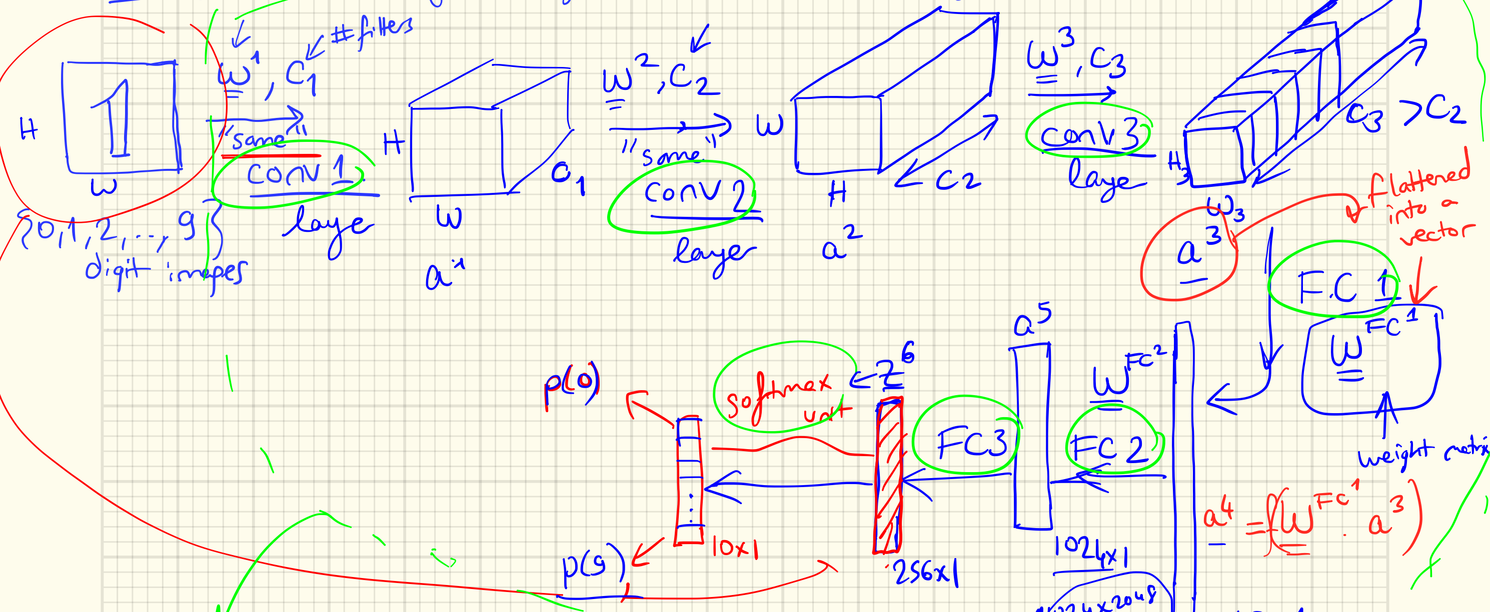
⟶ filter is <u>flipped</u>

∴ In CNNs, backprop just <u>flips</u> convolutions.

2D Conv: flip vertically & horizontally.

Encoder Module

Ex: CNN (mainly) for images. Build a CNN for Digit Classification (10 class)

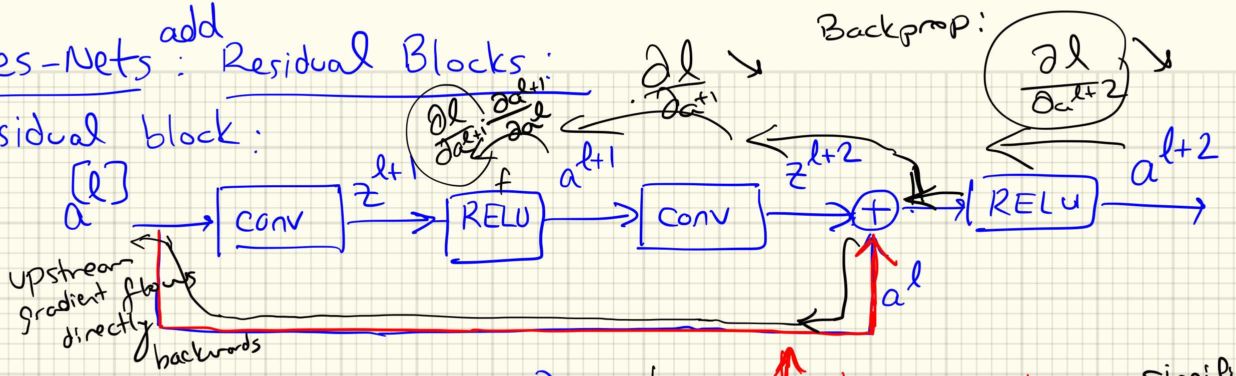


1. this whole model defines our hypothesis function →
2. loss fn: cross-entropy, multi-class hinge loss
3. pick an optimizer, eg. Adam.

You have a deep learning CNN classifier.

Ex: Res-Nets add Residual Blocks:

1 residual block:



$$a^{l+1} = f(W^{l} * a^{l} + b^{l})$$

$$z^{l+2} = W^{l+1} * a^{l+1} + b^{l+1}$$

$$a^{l+2} = f(z^{l+2} + a^{l})$$

$\uparrow$ Skip connection: significance.

- think of vanishing gradient problem

∴ ResNet: its skip connections helped reduce vanishing grad. problem, particularly in deep networks

$+a^{l}$ newly added in a ResNet block

$$a^{l+2} = f\left(W^{l+1} * f(W^{l} * a^{l} + b^{l}) + b^{l+1} + a^{l}\right)$$

(ResNet 2015): $\approx$ 150 layers!! very deep network