L11: 23.05.2022

Ch 7: Convolutional Neural Network (CNN)

Ch 4: dense network

- consisting of dense layers only
- rarely used

Slide 7-1

Solution:

CNN : consisting of mainly convolutional layers, based on convolution

- lower memory complexity
- good for feature learning

7.1 Convolutional layer

Slide 7-2

CNN supports various data types:

data type		$\underline{X}_{l-1}, \underline{A}_l, \underline{X}_l$	$\underline{W}_l$
1D signal	eg Time series, audio	2D matrix	3D Tensor
2D "	eg image	3D Tensor	4D "
3 D Tensor	eg Tensor	4D "	5D "

1 D more : multiple input/output signals/images/features maps/channels

eg RGB image : 3 x 2D image → 3D Tensor

2D convolutional layer :
$\uparrow$
Data type

$$\underline{\underline{X}}_{l-1} \longrightarrow \boxed{\begin{matrix} \underline{\underline{W}}_l \\ \underline{b}_l \\ \phi_l \end{matrix}} \longrightarrow \underline{\underline{X}}_l$$

- input tensor – $\underline{\underline{X}}_{l-1}$ – dim – $H_{l-1} \times W_{l-1} \times C_{l-1}$
  - H_{l-1}: Height, W_{l-1}: Width, C_{l-1}: Channel

  C_{l-1} 2D features maps of size $H_{l-1} \times W_{l-1}$
  - C_{l-1}: # channels

- output tensor – $\underline{\underline{X}}_l$ $\quad H_l \times W_l \times C_l$

  C_l 2D feature maps of size $H_l \times W_l$

- Kernel tensor $\underline{\underline{W}}_l$ $\quad K_l \times K_l \times C_{l-1} \times C_l$
  - K_l: size of Kernel / Kernel width

  C_l 3D impulse response / filter / kernel of size $K_l \times K_l \times C_{l-1}$

- bias vector $\underline{b}_l$ $\quad C_l \times 1$

  One bias vector for each funtion map / output channel

- activation ~~function~~ tensor $\underline{\underline{A}}_l$ $\quad H_l \times W_l \times C_l$

- activation function $\phi_l(\,)$ element wise $\quad \underline{\underline{X}}_l = \phi_l(\underline{\underline{A}}_l)$

$$[\underline{\underline{A}}_l]_{hwo} = \underbrace{\sum_{i=1}^{K_l}\sum_{j=1}^{K_l}}\ \underbrace{\sum_{c=1}^{C_{l-1}}}_{\substack{\uparrow \\ \text{combining all} \\ \text{feature maps}}} [\underline{\underline{W}}_l]_{ijco}\ [\underline{\underline{X}}_{l-1}]_{\underbrace{h+i-1,\, w+j-1,\, c}} + [\underline{b}_l]_o$$

3D spatial - channel processing

$\underline{hw}$ → spatial correlation, called convolution in DL

c → combines all C_{l-1} input channels

o → one of C_l output channels

$$\left.\begin{aligned} 1 \le h \le H_l &= H_{l-1} - K_l + 1 \\ 1 \le w \le W_l &= W_{l-1} - K_l + 1 \end{aligned}\right\} \text{Output size reduced by } K_l - 1 \text{ in each dimension}$$

$$1 \le o \le C_l$$

Short notation : $\underline{\underline{A}}_l = \underline{\underline{W}}_l * \underline{\underline{X}}_{l-1} + \underline{b}_l$

Slide 7-5

Typical kernel size : $K_l \times K_l$: 3×3, 5×5, but also 1×1

" no of channels : C_l ~1, ~10, ~100

Number of parameters :

$$N_{p,l} = \underbrace{K_l^2 C_l C_{l-1}}_{\underline{\underline{W}}_l} + \underbrace{C_l}_{\underline{b}_l} \approx K_l^2 C_{l-1} C_l$$

quite small, independent of image source unlike DNN

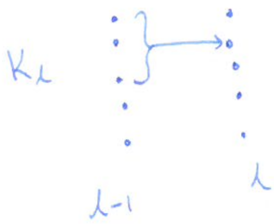
No of multiplications $N_{x,l} = \underbrace{H_l W_l C_l}_{\#\text{ elements in } \underline{\underline{X}}_l} \cdot \underbrace{K_l K_l C_{l-1}}_{\#\text{ multiplications per element of } \underline{\underline{X}}_l} \approx H_l W_l \cdot N_{p,l}$

quite large, depends on o/p size

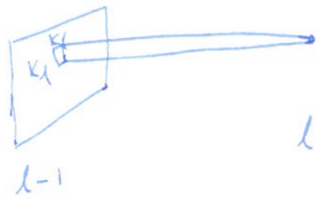
Properties of a convolutional layer:

P1. Sparse connection b/w input and output:

1D



2D

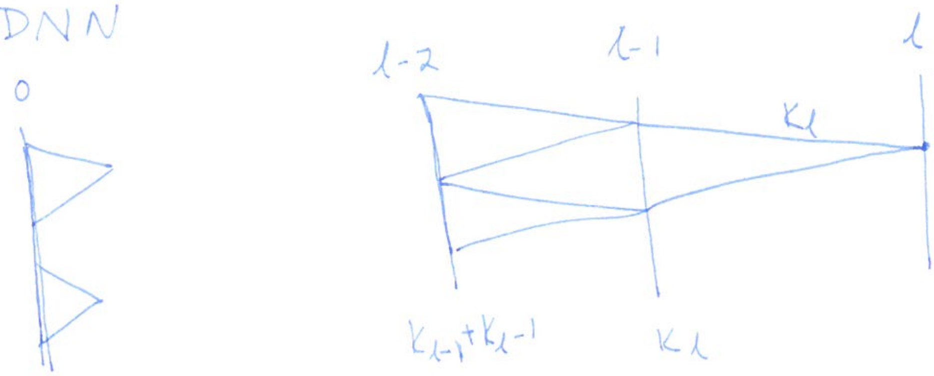


small receptive fields K_ℓ

$K_\ell \times K_\ell$

$\hat{=}$ focus on local patterns, eg edges, contours, features etc.

In DNN



A neuron in layer ℓ has a receptive field width

$$K_1 + \cdots + K_\ell - (\ell+1)$$

P2 Parameter sharing: same kernel tensor is used for all neurons/pixels

Slide 7-8

P1 + P2

- fewer parameters (N_p)
- reduced model capacity and chances/risk of overfitting

P3 Shift invariant $\longleftrightarrow$ translation-equivariant

translation of $\underline{\underline{X}}_{l-1}$ $\rightarrow$ same translation in $\underline{\underline{X}}_l$

P4 one feature map in the o/p contains one feature eg horizontal edges.

$\rightarrow$ need feature maps/kernels for multiple features: C_l

P5
$$H_l = H_{l-1} - K_l + 1$$
$$W_l = W_{l-1} - K_l + 1$$
: image becomes smaller.

7.2 Variants of convolution

Simple variants:

Slide 7-9.

7-10

Output size: standard

$$H_l = H_{l-1} - K_l + 1$$

including all variants

$$\left\lfloor \frac{H_{l-1} + 2P - (K_l - 1) \cdot D - 1}{S} \right\rfloor + 1$$

		Default
$P \geq 0$:	Padding	0
$S \geq 1$:	Stride	1
$D \geq$:	dilation	1

all variants can be combined.

$\lfloor \ \rfloor$: next lower integer

Other modifications :

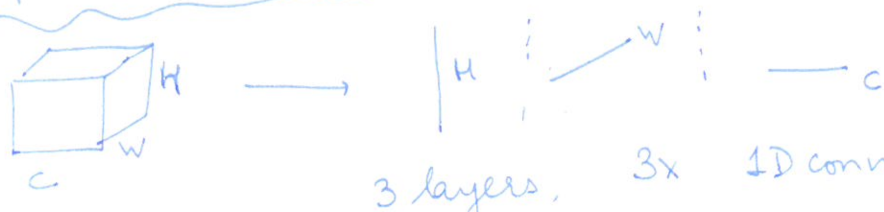
1. Standard convolution :

3D Kernel $K \times K \times C$ $\sum_h \sum_w \sum_c$

joint spatial-channel processing → high complexity

Separable convolutions : Factorization of kernel to reduce the complexity

2. Spatial and depthwise separable convolution



3 layers, 3x 1D convolution

Price to pay : lower accuracy and retention of data

3. Depthwise separable convolution

2 layer 2D

Q Can't dimension reduction techniques like PCA, LDA work in these
Ilias scenarios?