Lecture FYS5429, February 13, 2024

Basics of NNs

- Model/architecture
  - input layer
  - hidden layer(s) with hidden nodes

$$\Theta = \{ W^{(1)}, b^{(1)}, W^{(2)}, b^{(2)}, \ldots W^{(L)}, b^{(L)} \}$$

  - output layer with output $\hat{y}$
  - compare with target value y through a cost function

$$C(\Theta, x, \hat{y}, y)$$

- Training of θ through Feed Forward stage + Back propagation (gradients)

$$\frac{\partial C}{\partial \theta}$$

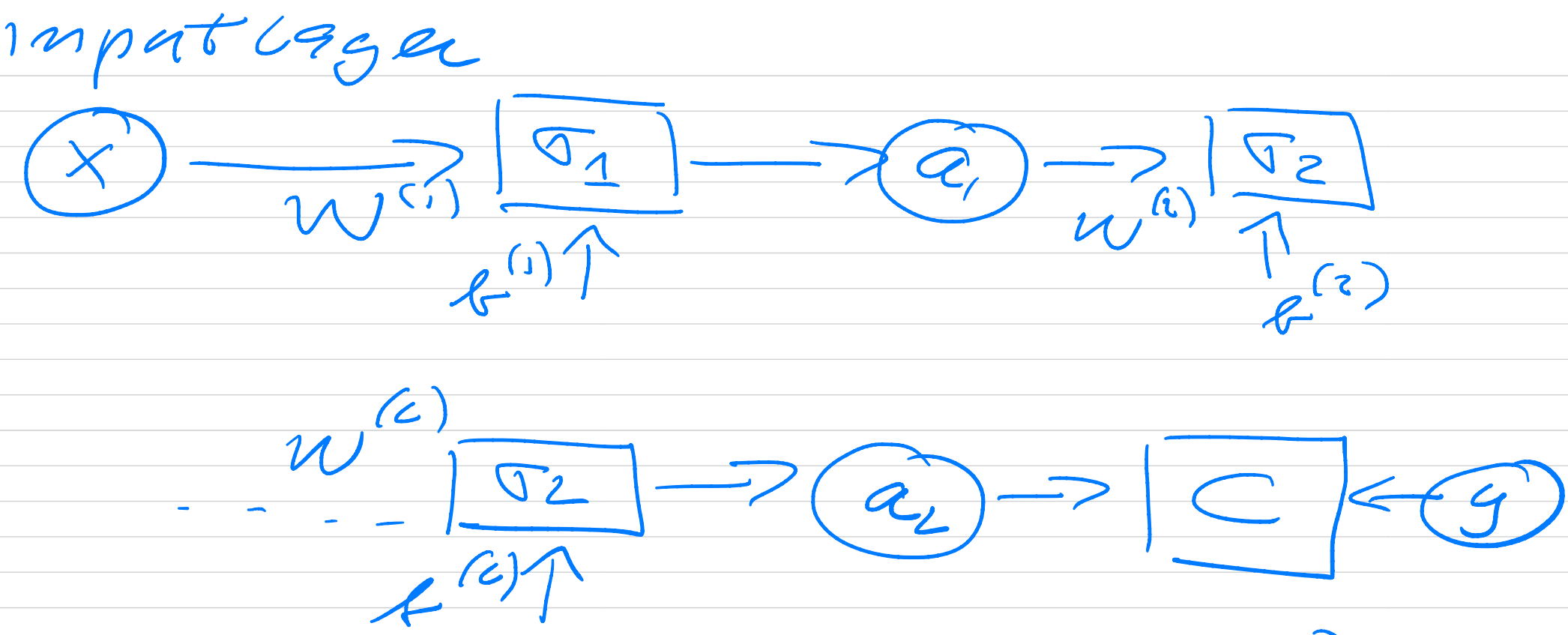
$$\theta^{(e)} \leftarrow \theta^{(e)} - \eta \frac{\partial C}{\partial \theta^{(e)}}$$

η ↑ learning rate

ADAM

RMSprop

ADAgrad



$\sigma_1(z_1)$ $\quad z_1 = W^{(1)}x + b^{(1)}$

activation function

cost function

Hyperparameters

$$-C(\epsilon; x) + \lambda \|w\|_2^2$$

$$-\iota \frac{\quad}{\quad} + \lambda \|w\|_1$$

- the activation function
- epoch + batch

Basics of a CNN

A typical CNN has 4 stages

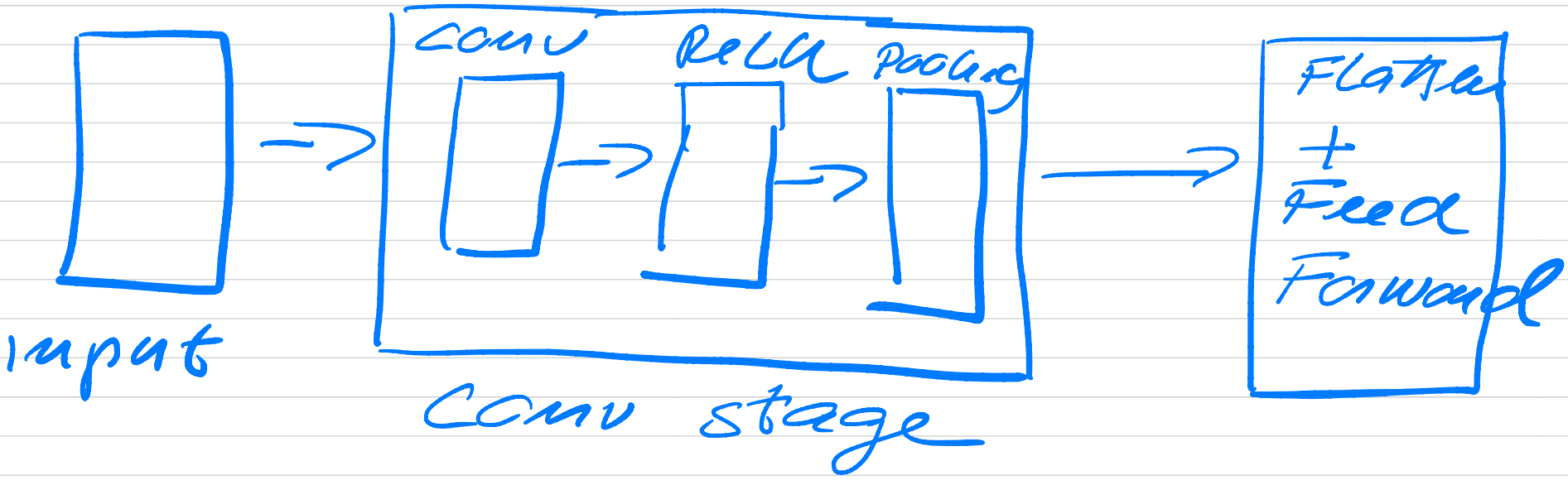
(i) several convolutions in parallel to produce a set of lower dimensionality images (parameters trained by back propagation)

(ii) Run through a non-linear activation function (ReLU) only hyperparameters

(iii) Pooling stage to further reduce the dimensionality

Only hyperparameter

(iv) Flattening + Feed Forward NN. Training with Back propagation



Convolution introduces a set of hyperparameters

- K number of filters
- F the spatial extension
- S Stride parameter
- P padding parameter

common settings

F = 3	S = 1	P = 1
F = 5	S = 1	P = 1 or 2
F = 5	S = 2	P optional