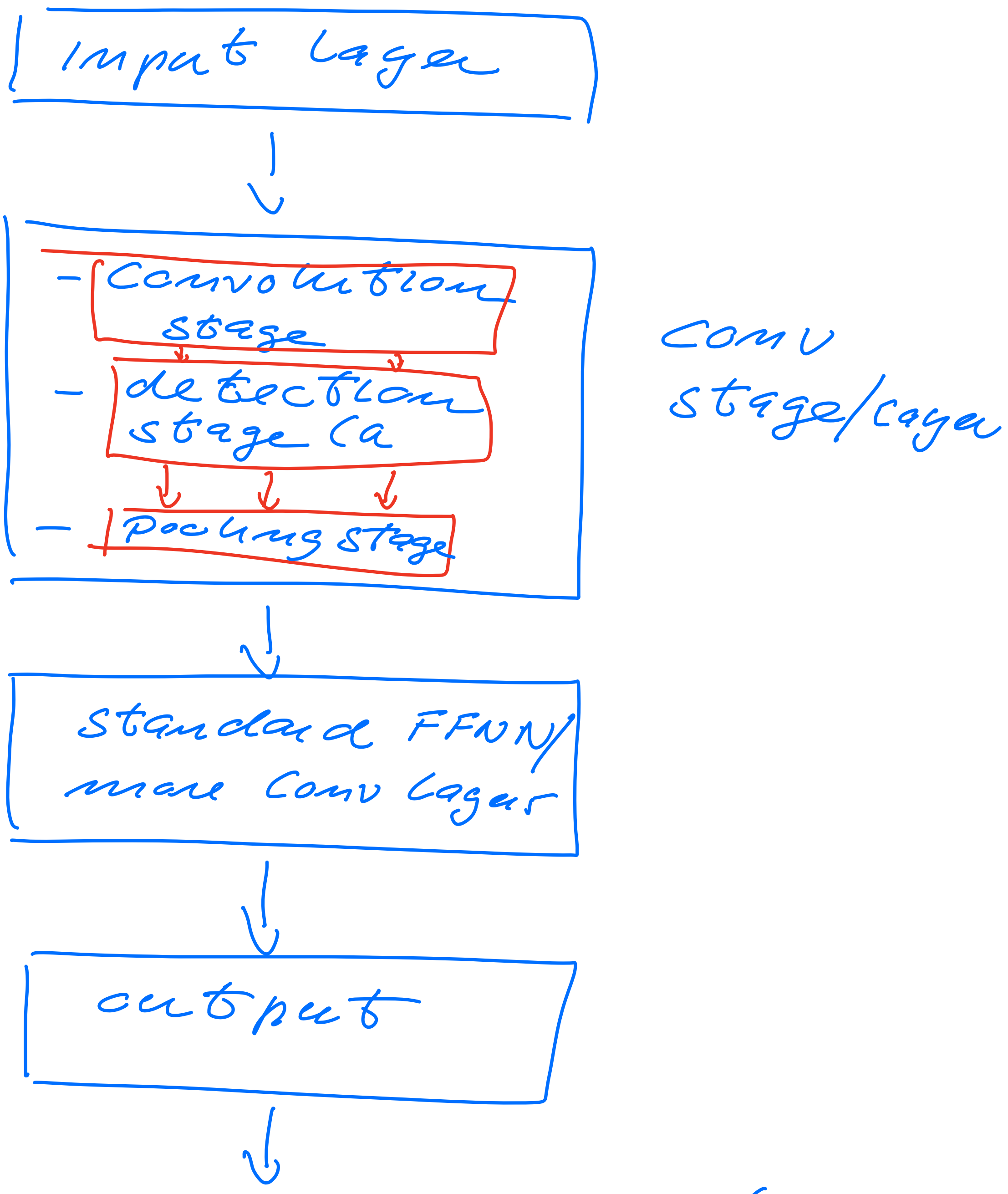
Comp Sci Program, 2/28, 2022

CNN



compare with data and assess quality of model via cost/loss/risk..

function,

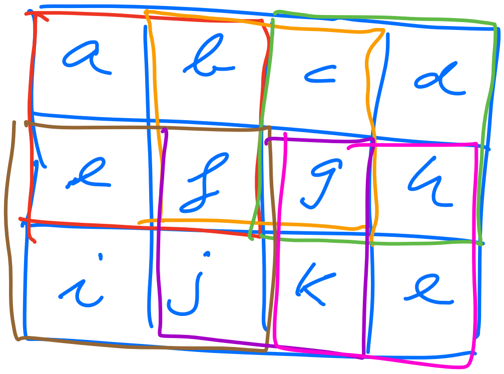
conv. stage

I

kernel/filter

K

3x4



$$\begin{matrix} a & b & c & d \\ e & f & g & h \\ i & j & k & e \end{matrix}$$

$$\begin{matrix} w & x \\ y & z \end{matrix}$$

2x2

$\boxed{\text{Dim } K \leq I}$

Stride = 1

$\boxed{aw + bx + ey + fz}$

$\boxed{bw + cx + fy \quad g \cdot z}$

$\boxed{cw + dx + sy + h \cdot z}$

$\boxed{ew + fx + iy + j \cdot z}$

$\boxed{fw + gx + jy + k \cdot z}$

$\boxed{gw + hx + ky + ez}$

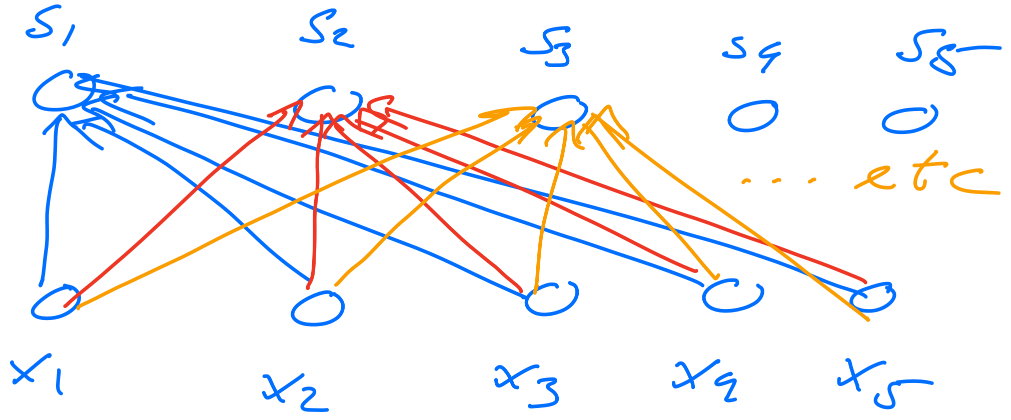
Filter-parameters are tuned by the CNN

$(I * K)$ produces an output which goes into the detection which goes into a detector stage = application of a user chosen activation function. Typical activation is a ReLU or ELU or similar functions

→ fed into a pooling stage.

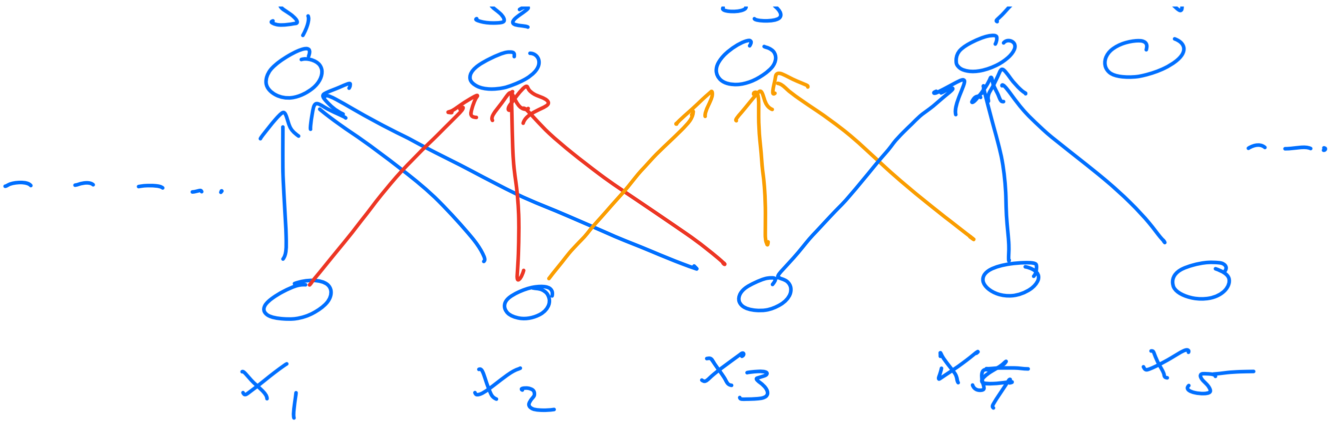
- Sparse connectivity

FFNN = feed forward NN

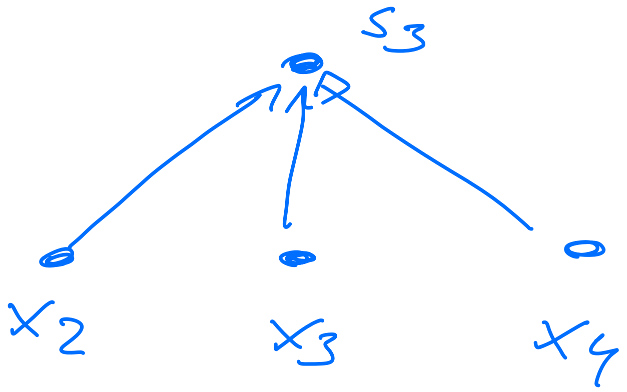


Need 5 × 5 weights to train.

- CNN



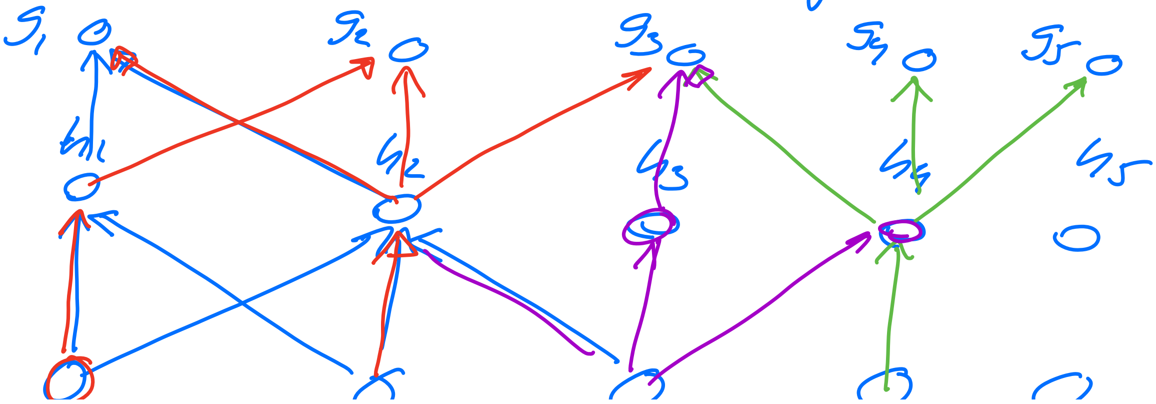
viewed from above

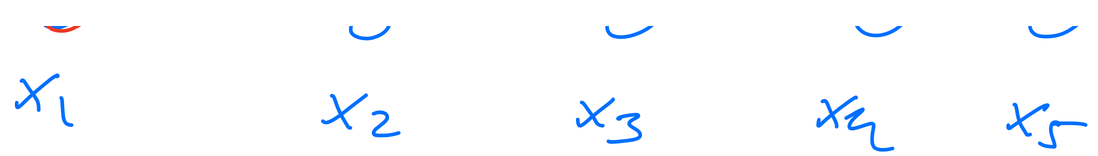


Number of weights: 3×5

parameter sharing

share the same parameters for more than one function





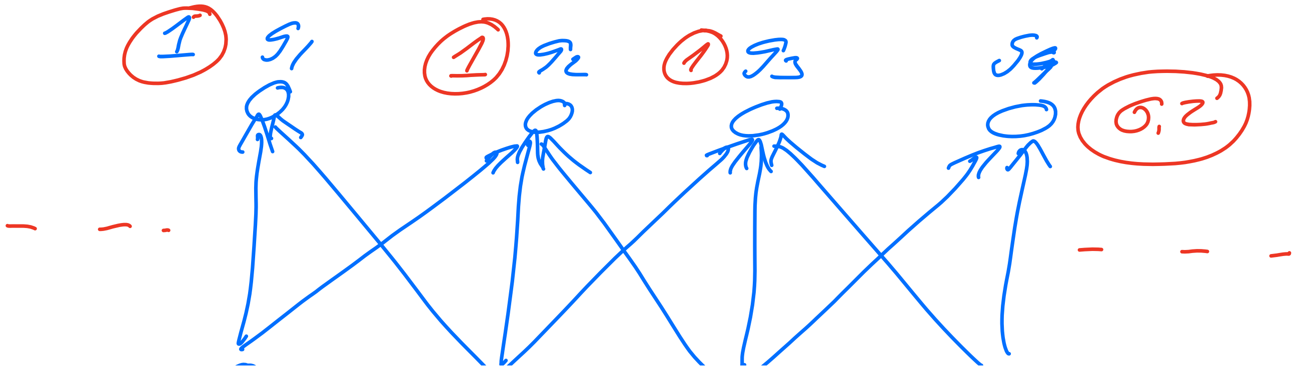
Direct connections are sparse, however, units can be in deep layers, connected to many or all of the units in the input image.

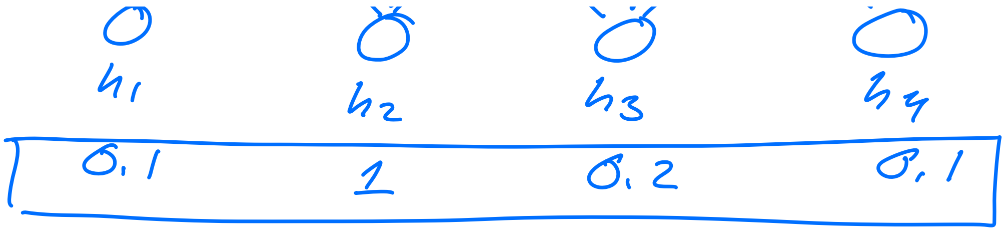
The output goes into an activation function.

⇒ pooling stage.

Pooling replaces the output at certain location (= node) with a summary statistics of the nearby points!

Max pooling:





output of activation function