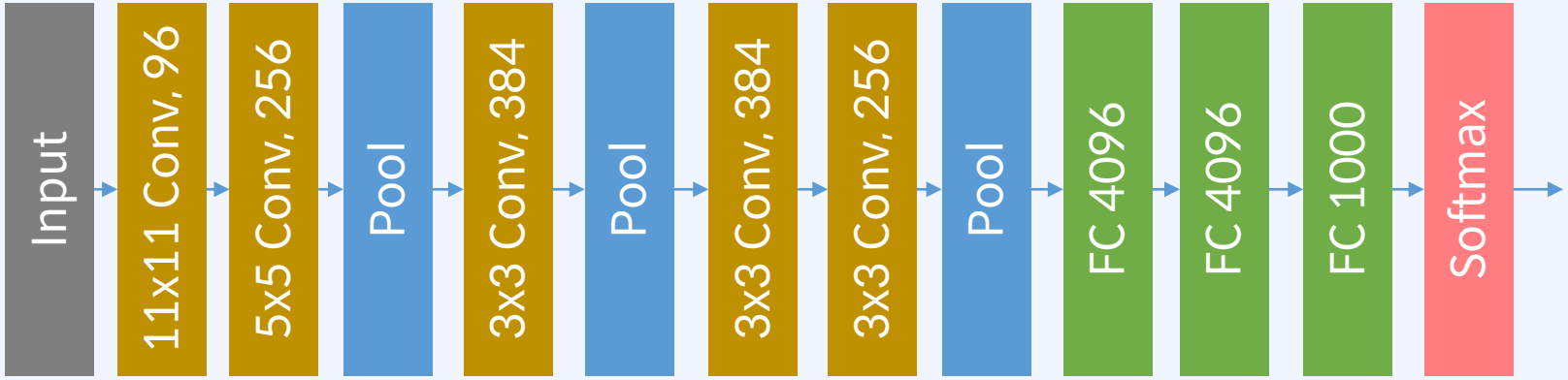
Ch1. CNN Architecture

ImageNet Classification with Deep Convolutional Neural Networks

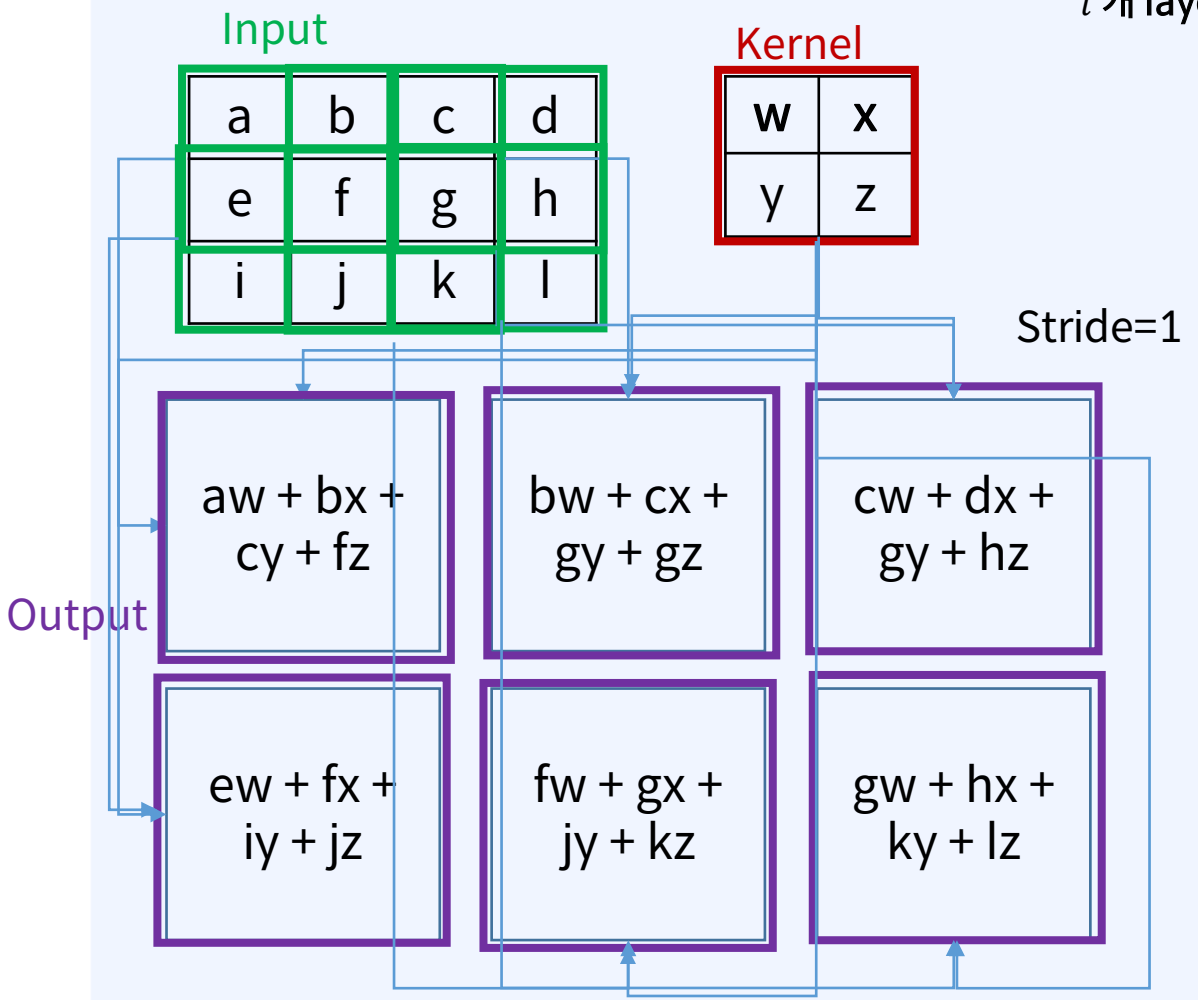
AlexNet Architecture

- Convolution layer 5개 + fully connected layer 3개



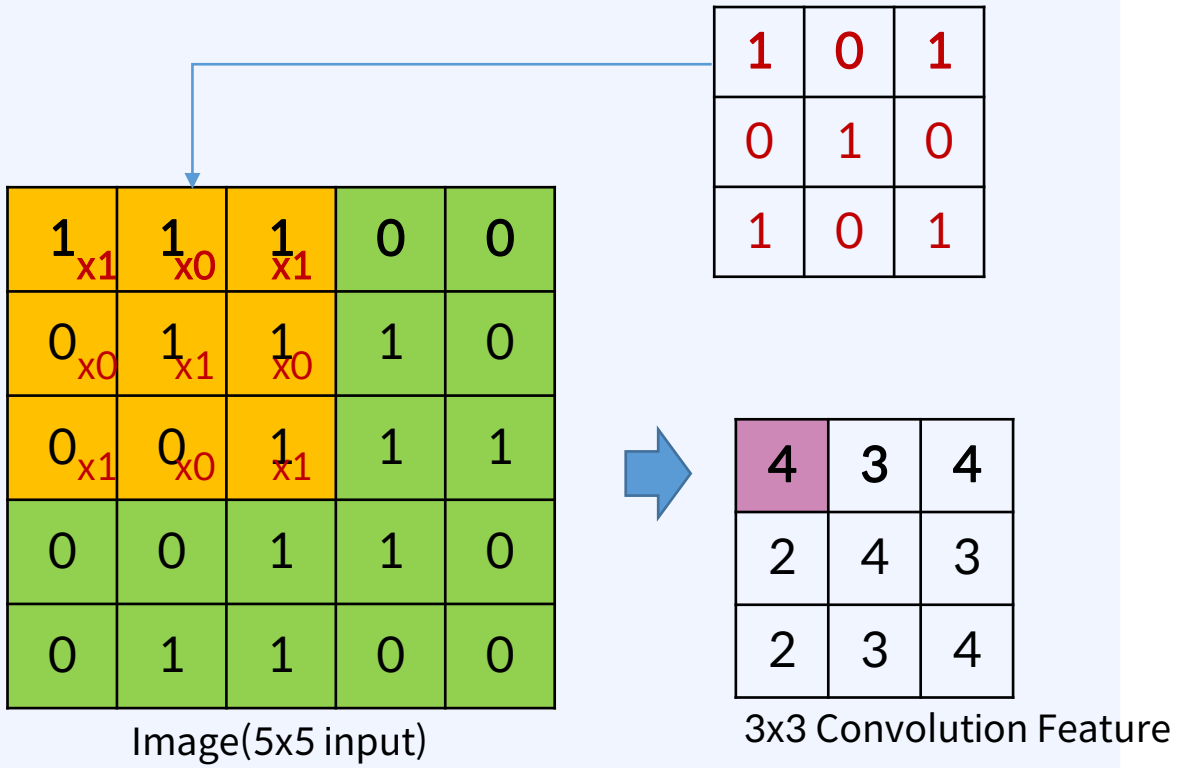
Convolution

Convolution operation

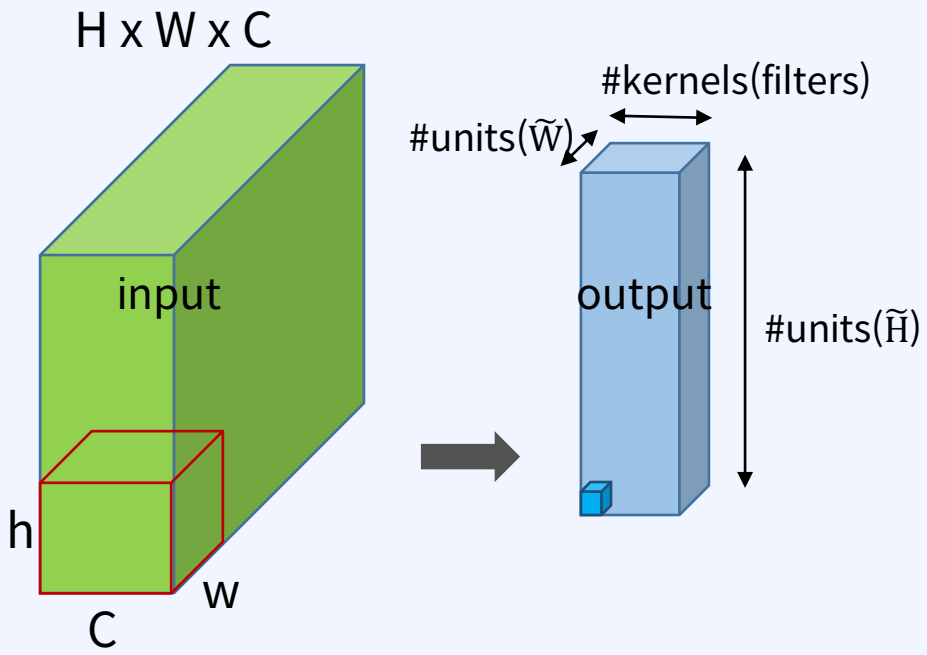


l 개 layer의 m x m kernel을 사용한다고 했을 때 수식 :

$$f(x) = \sum_{a=1}^{m-1} \sum_{b=1}^{m-1} w_{ab} x_{(i+a)(j+b)}^{l-1}$$

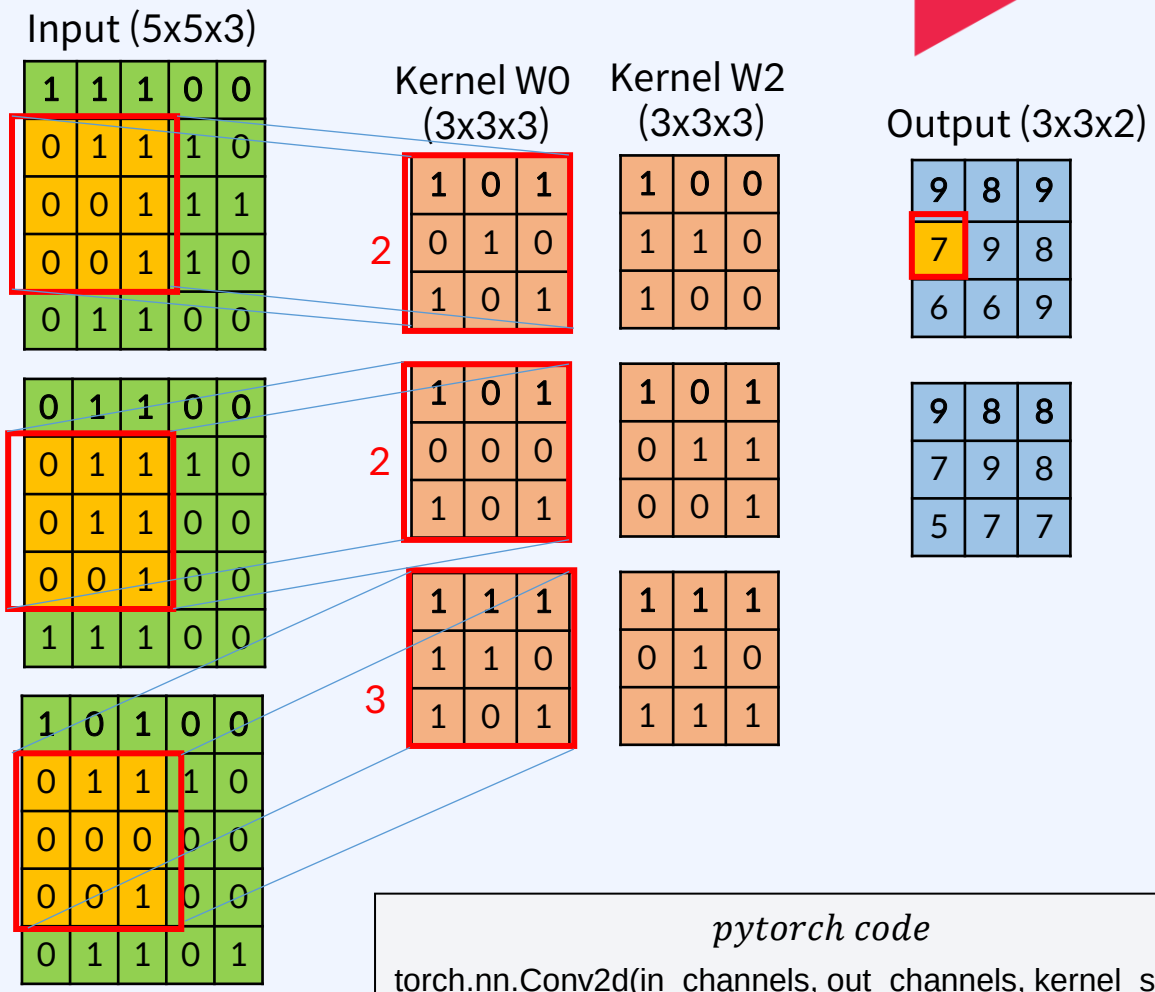


2-D Convolution



$Kernel_1 = (w, h, C)$

** h: height, w: width, C: channel*



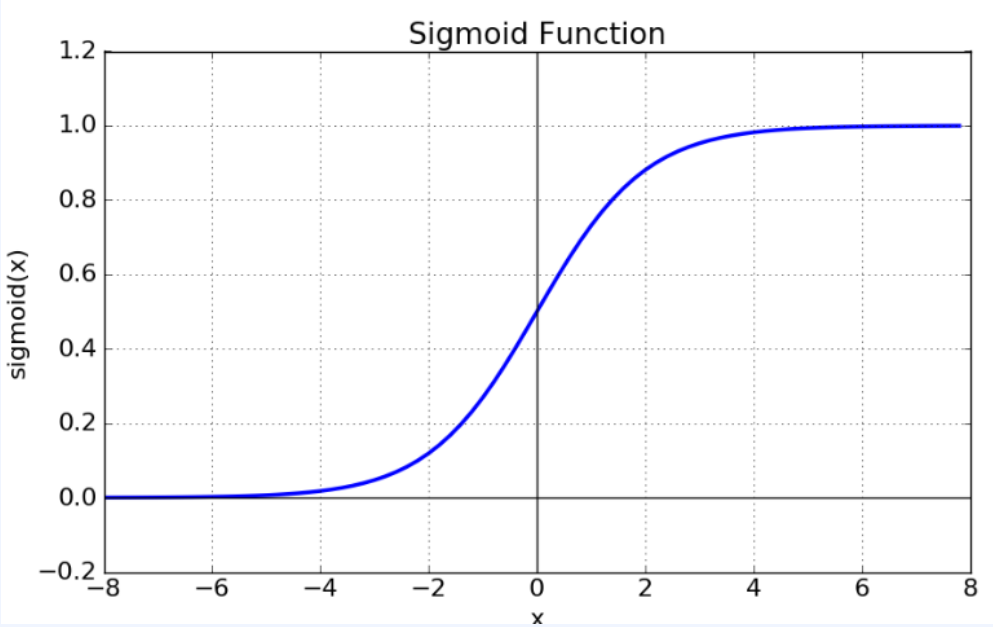
pytorch code

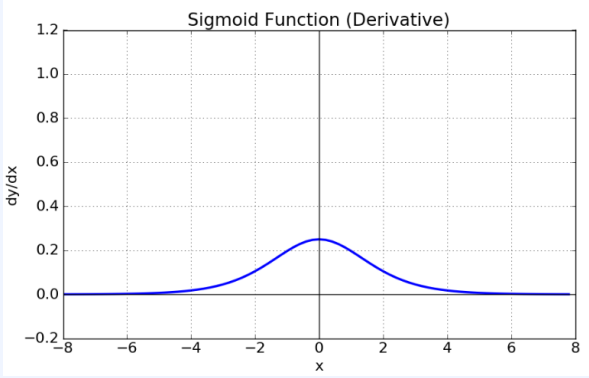
torch.nn.Conv2d(in_channels, out_channels, kernel_size,..)

Activation

Activation

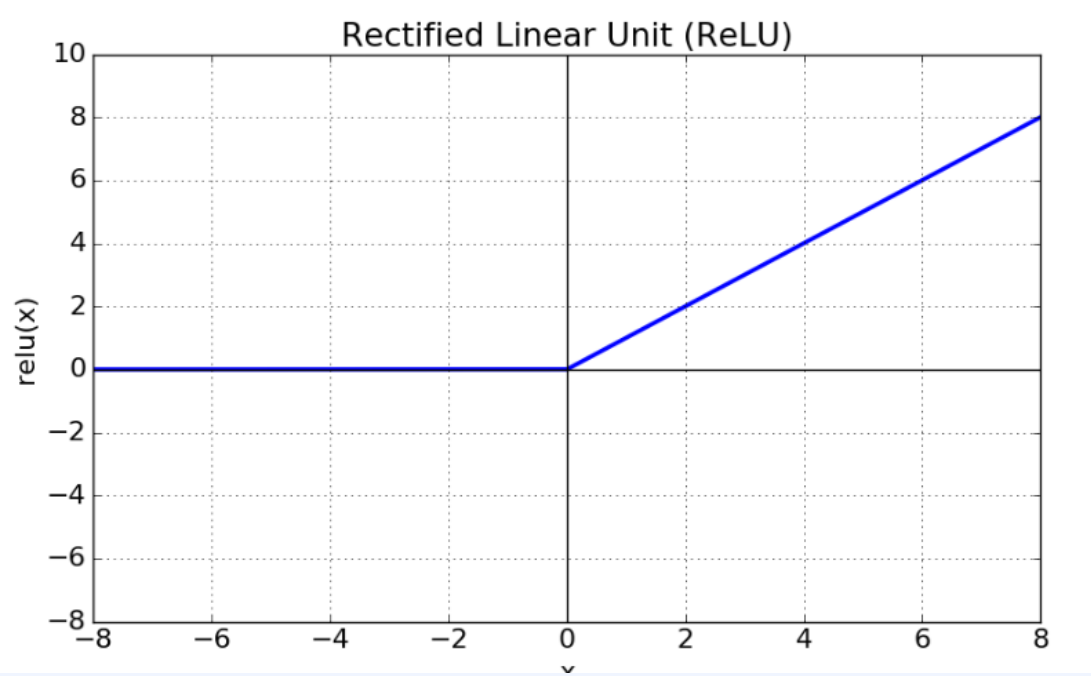
- Sigmoid function
  - $sigmoid(x) = \frac{1}{1+e^{-x}}$





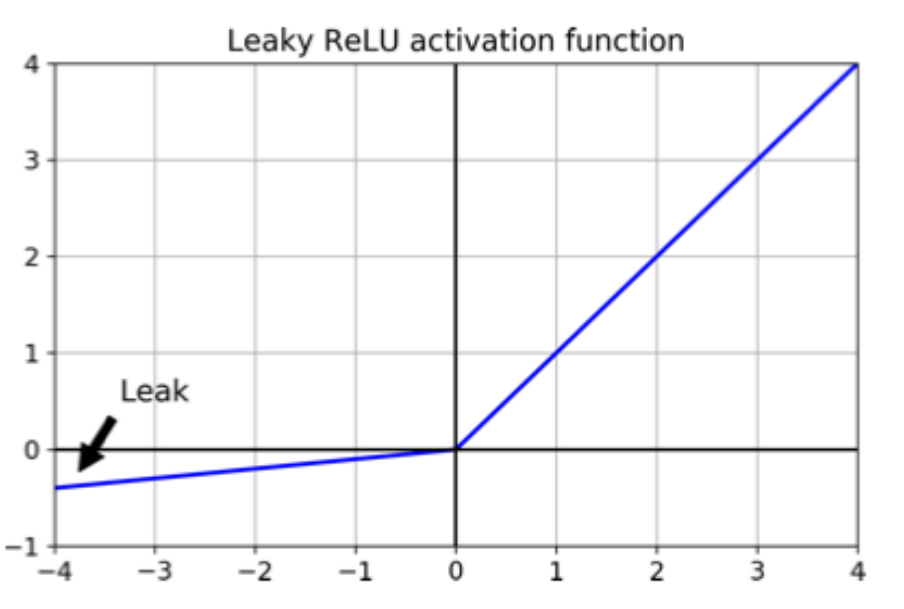
Activation

- ReLU(Rectified Linear Unit)
  - $f(x) = \max(0, x)$



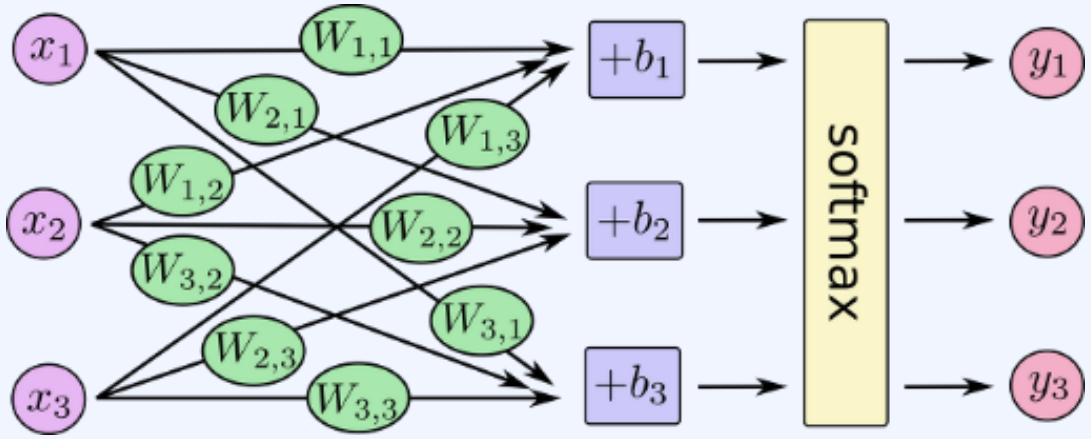
Activation

- Leakly ReLU
  - $f_a(x) = \max(ax, x)$



Activation

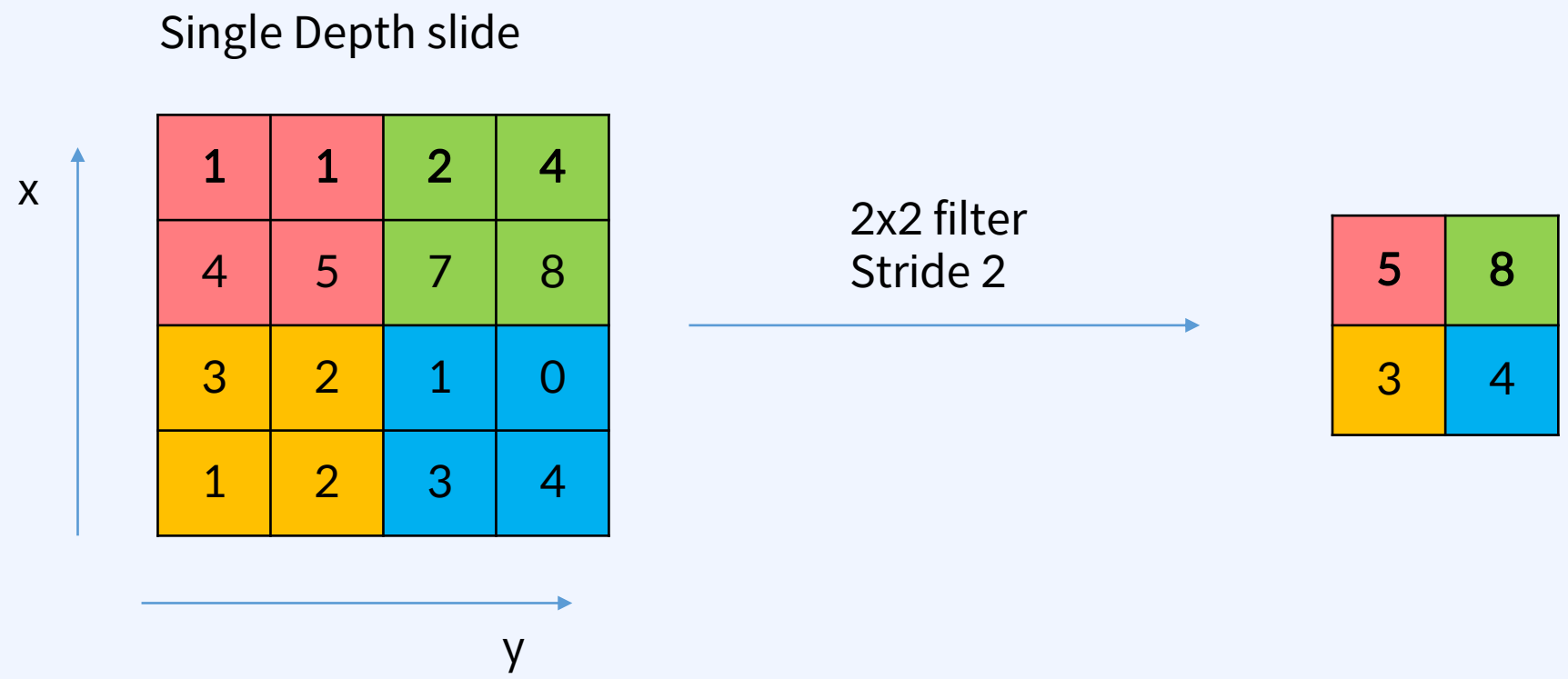
- Softmax
  - $p_j = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$, for j = 1, … , K



Pooling

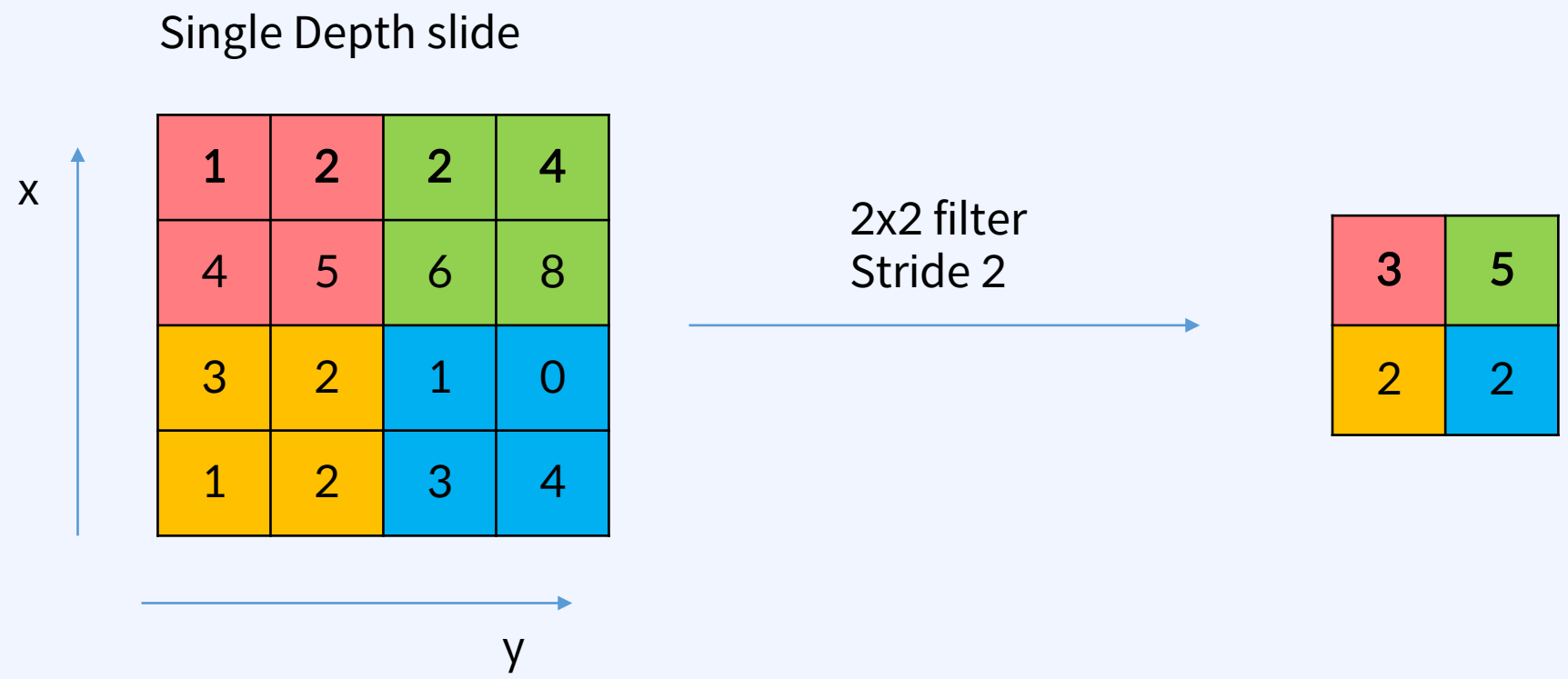
Pooling

- MaxPooling



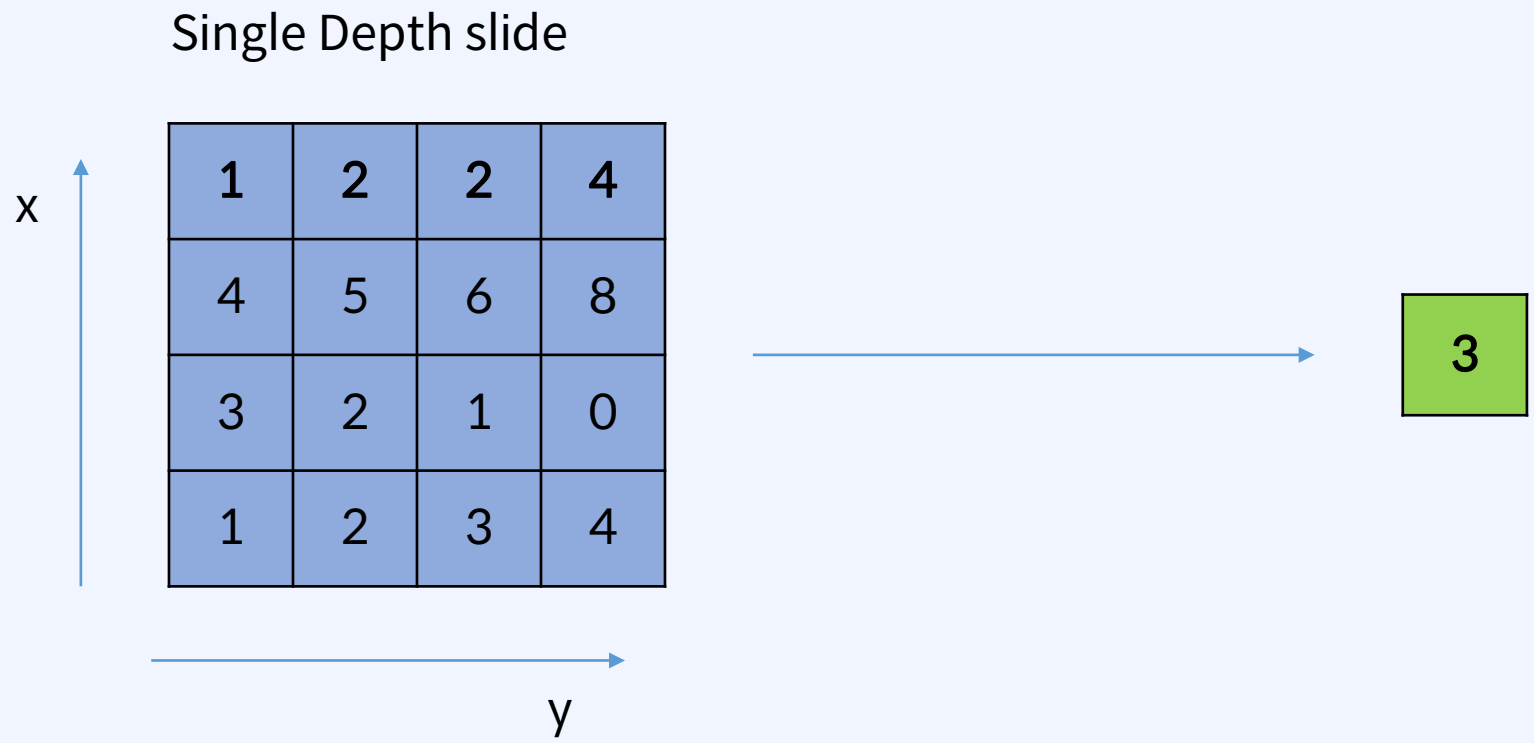
Pooling

- AvgPooling



Pooling

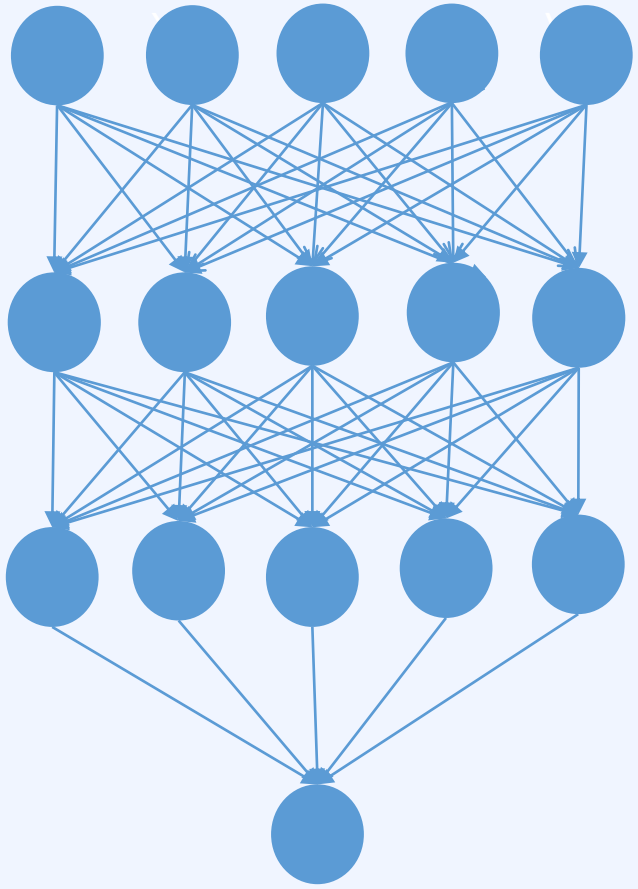
- GlovalAvgPooling(GAP)



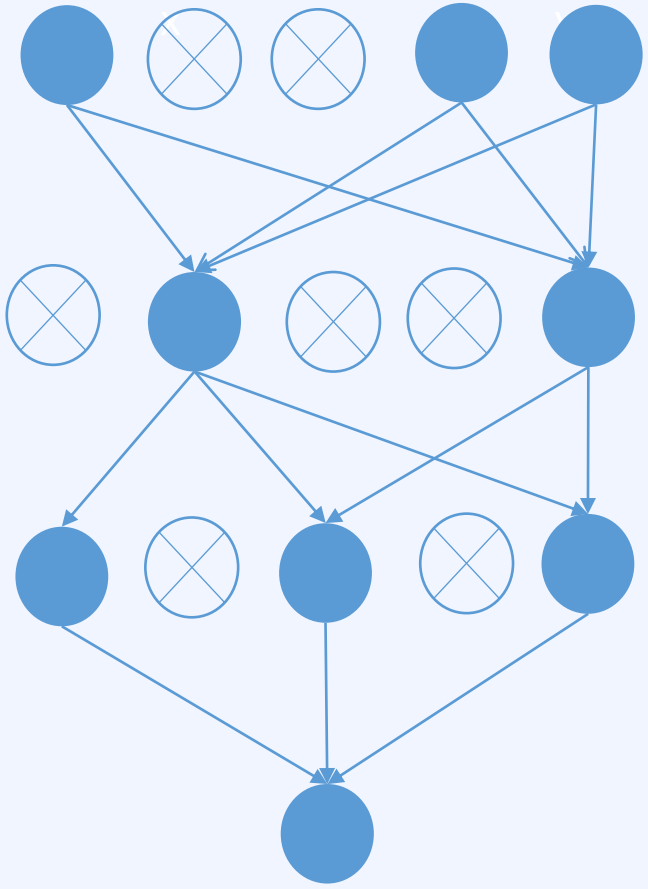
Dropout, Fully Connected layer

Dropout

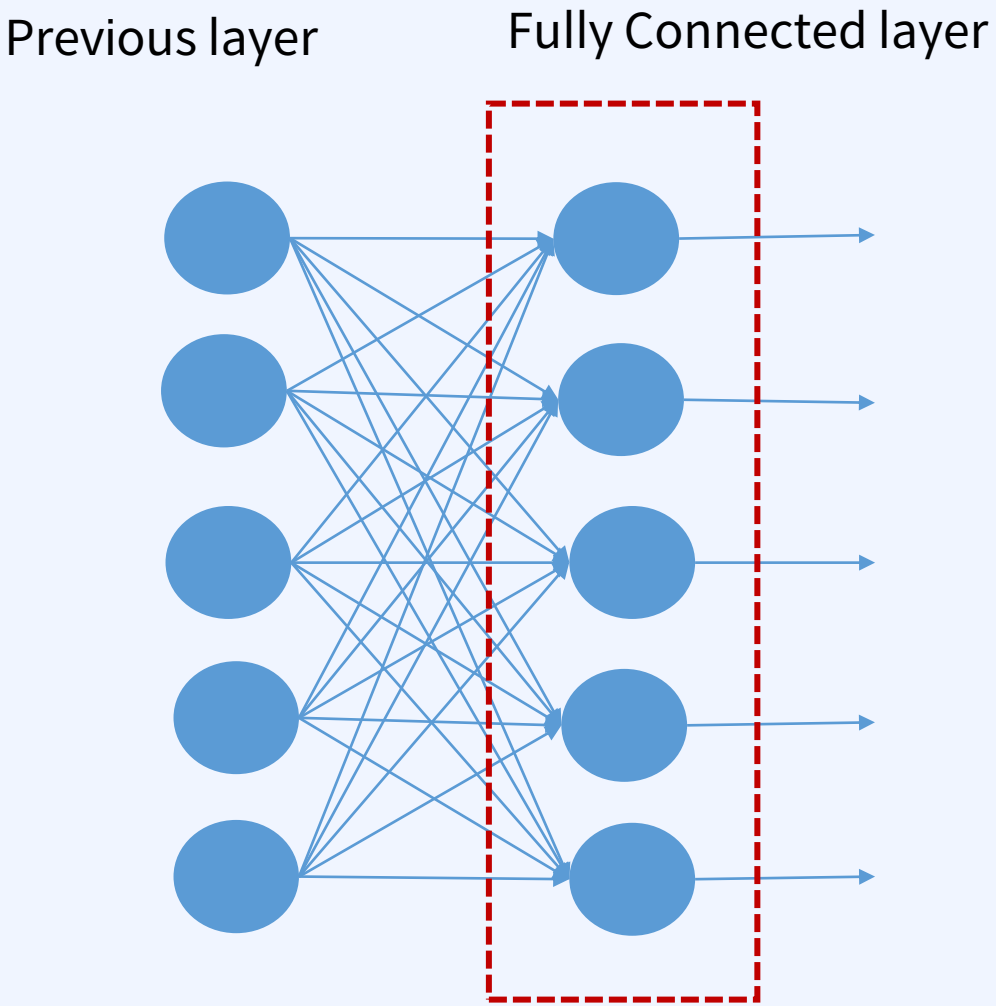
Standard Neural Net



Applying dropout



Fully Connected Layer



Summary