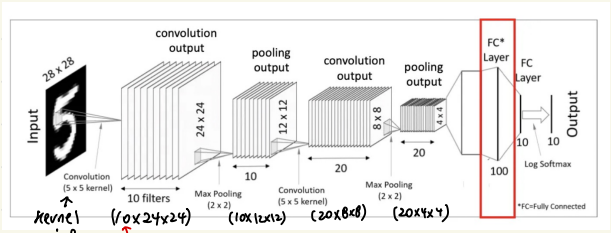
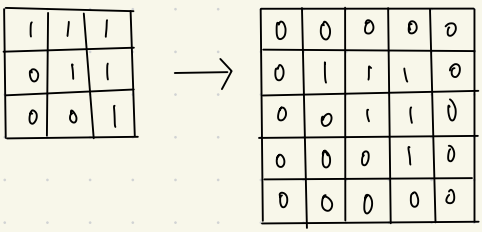
CNN Structure



kernel size. (10x24x24) "channels" (10x12x12) (20x8x8) (20x4x4)

* RGB has 3 channels ex) 3x64x64
* Greyscale has 1 channel. ex) 1x28x28

* Padding:



Why?

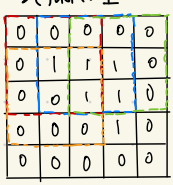
if input data is $\mathbb{R}^{5\times5}$, and apply kernel 3x3, $\mathbb{R}^{5\times5} \to \mathbb{R}^{3\times3}$

After padding to data, $\mathbb{R}^{5\times5} \to \mathbb{R}^{n\times n}$, and when we apply 3x3 kernel, $\mathbb{R}^{n\times n} \to \mathbb{R}^{5\times5}$.

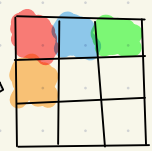
It prevents loosing information.

* Stride: How far the filter moves in every step along

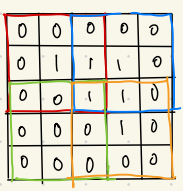
Default: 1



$\left(\frac{5-3}{1}+1\right) \times \left(\frac{5-3}{1}+1\right) = 3\times3$



Stride = 2



$\left(\frac{5-3}{2}+1\right) \times \left(\frac{5-3}{2}+1\right) = 2\times2$

* Dimension: $\left(\frac{d_1-k_1}{s}\right)+1 \times \left(\frac{d_2-k_2}{s}\right)+1$

* Convolutional layer: Convolution + (Normalize) + Activation

Optional

Code example w/ PyTorch:

```
nn.Sequential(
    nn.Conv2d(input_channel, output_channel, kernel_size, stride, padding, bias=False),),
    nn.BatchNorm2d(output_size),
    nn.LeakyReLU(0.2),
)
```