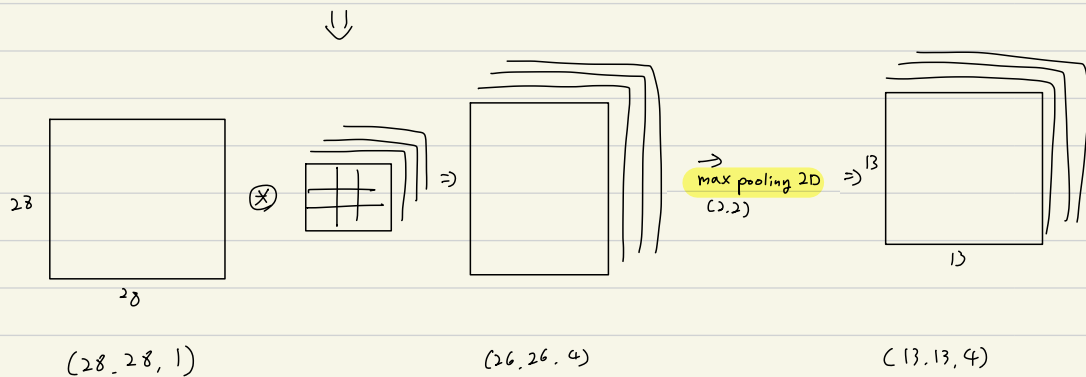
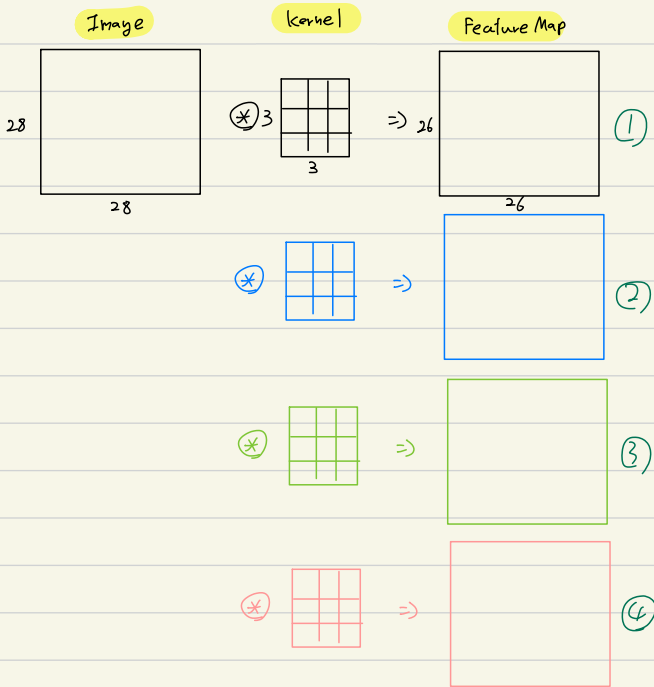
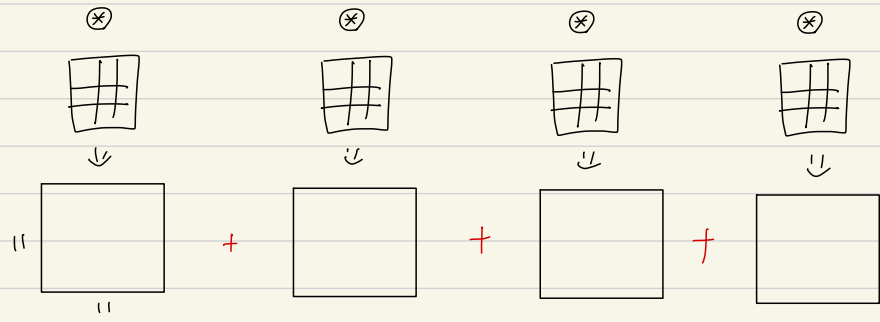
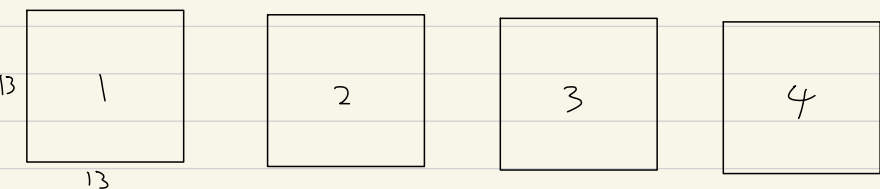
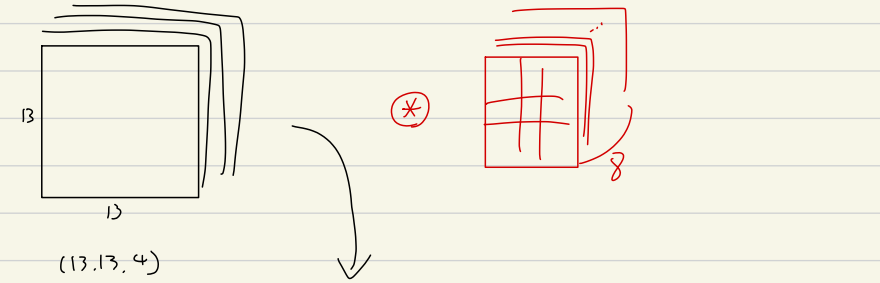


9월 2주

- `model.add(layers.Conv2D(4, (3,3), activation='relu', input_shape=(28,28,1)))`
 Output Feature Map이 4개 (kernel이 3) (gray & blue)
(높이, 폭, 채널)

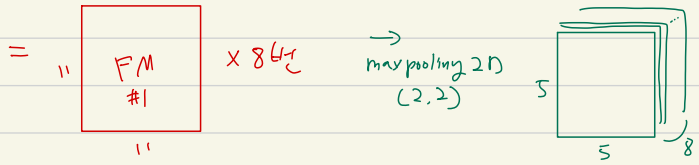


• Conv2D (8, (3, 3), activation = 'relu')



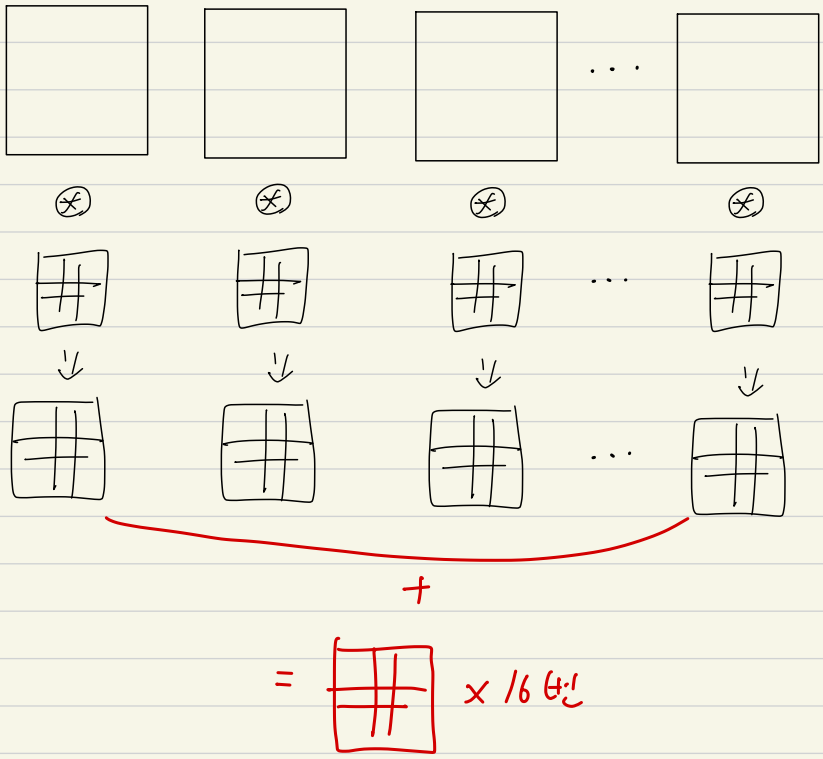
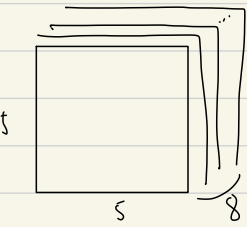
컨볼루션

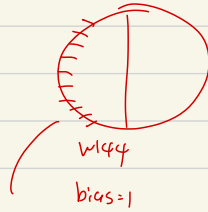
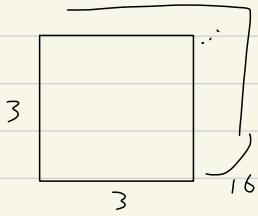
Feature Map
+ bias



* 23 4가지 컬러를 사용하지 않고서 8을 색 다르게 한다
↳ 동시에 padding은 크게해 더더없이 학습된다.

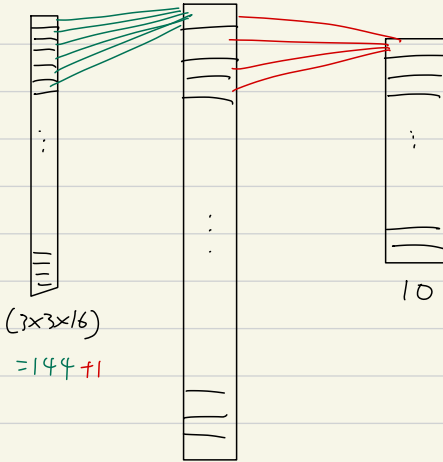
• conv 2D(16, (3, 3), activation = 'relu')





⇒ Flatten

Feature
vector



$$\begin{aligned} (32+1) \times 10 \\ &= 320 + 10 \\ &= 330 \end{aligned}$$

$$\begin{aligned} &\text{weight} \quad \text{bias} \quad 32768 \\ &(144+1) \times 32 \quad \text{노란색} \\ &= 4608 + 32 = 4640 \end{aligned}$$

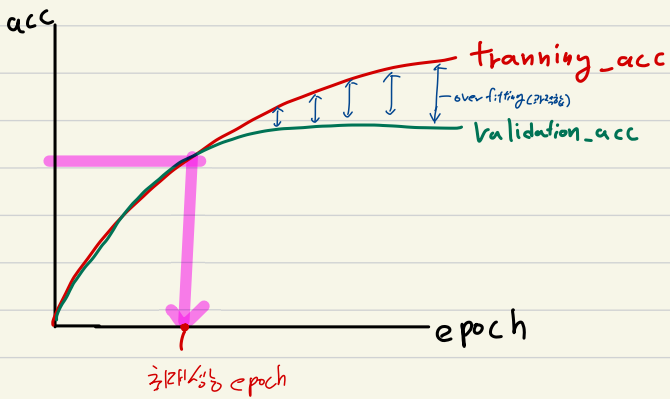
마지막에 max pooling 해 1x1x16을 만들면

$$(16+1) \times 32 = 544$$

$\frac{1}{8}$ (연산량)

if, 성능이 좋으면 max pooling 하는게 이득

* 2층은 one hot encoding 한 weight 330



Fit(, Validation_data(test))

최적화되는 과정에서 test-dataset을 이용해