

인공지능 기초

인공지능_ Day04

김새봄

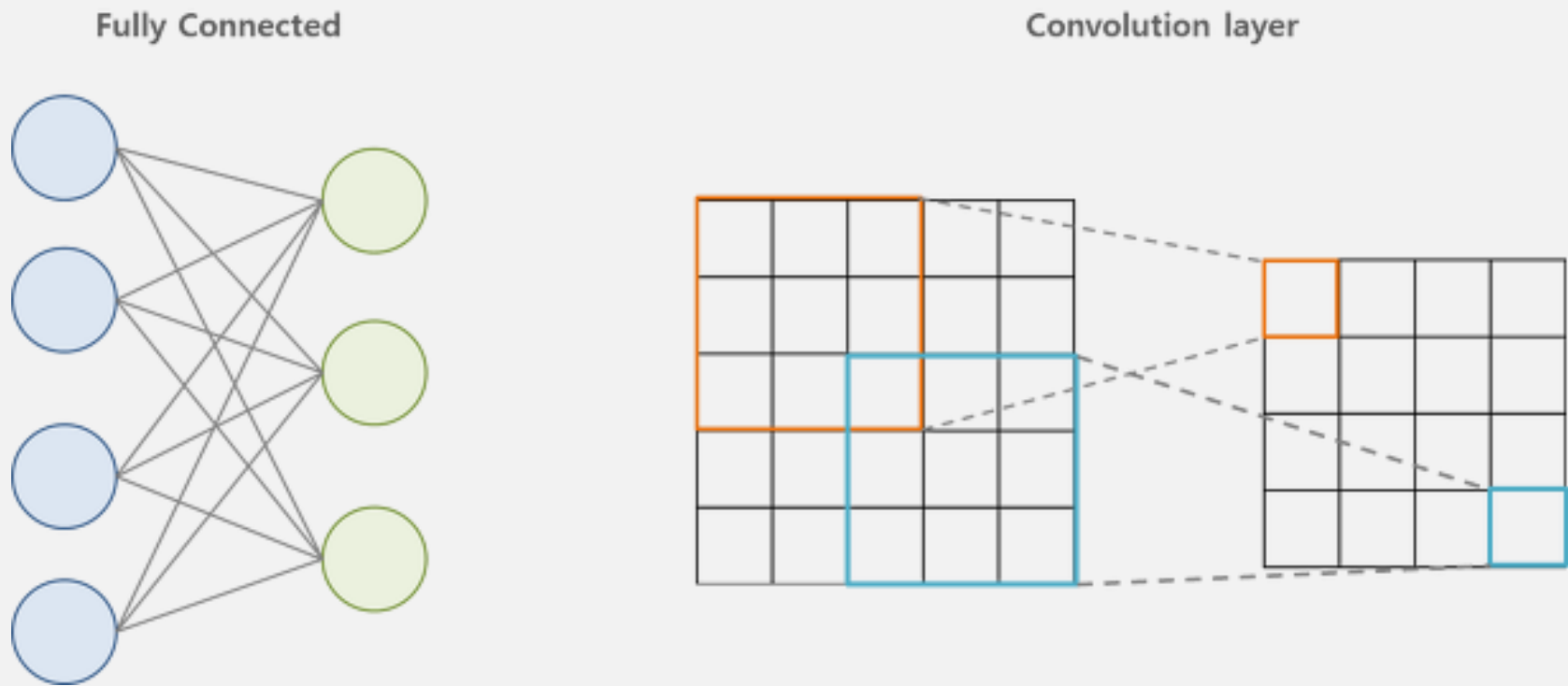
회귀분석과 분류분석	01
이미지 분석 - CNN 모델	02
이미지 분석 - 데이터 셋	03
실습	04

회귀분석과 분류분석

Table 1. 회귀분석과 분류분석 정리

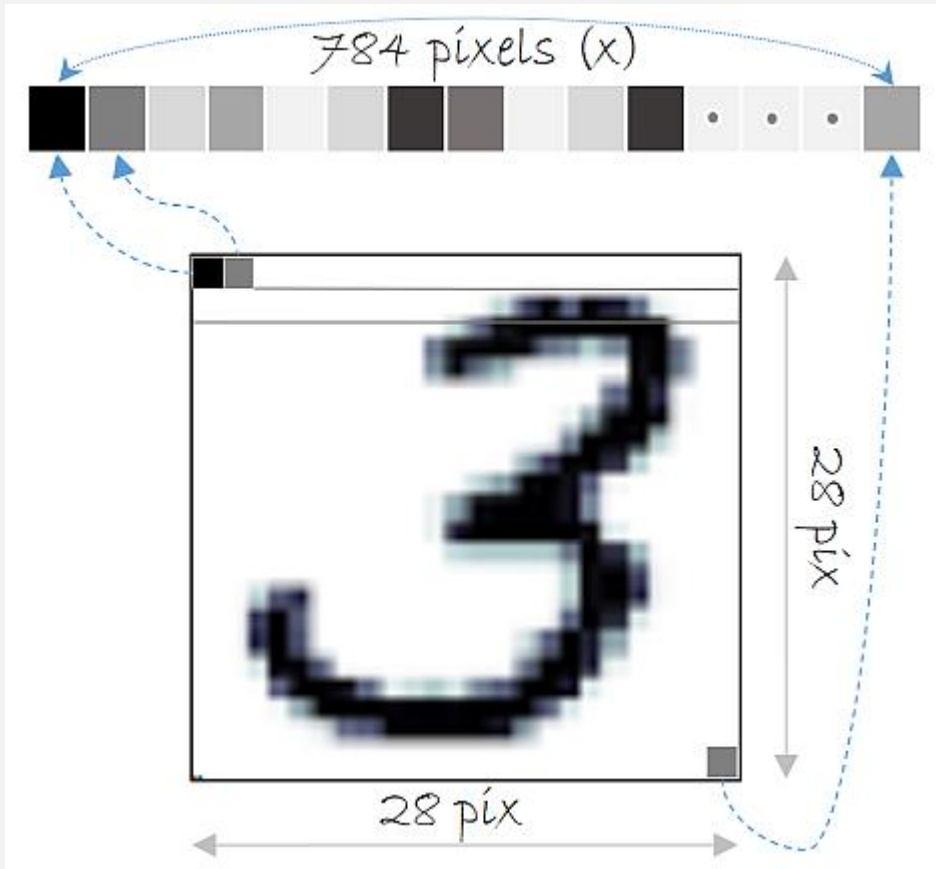
구분	회귀분석	분류	
		이진분류	다중분류
평가지표 Loss	mse, mae R2	Binary_Crossentropy accuracy	Catergorical_Crossentropy accuracy
Last Layer Activation	Default (Linear)	Simoid	Softmax
Last Layer node의 갯수	1이상	1	y레벨의 수만큼
Metrics	mse, mae	mse, mae + acc	mse, mae + acc
One Hot Encoding	X	X	0

이미지 분석 - CNN 모델



완전연결(fully connected)신경망과 합성곱(convolution)신경망

1. 이미지 분석에서 완전연결(fully connected)신경망의 문제점



출처 <https://excelsior-cjh.tistory.com/180>

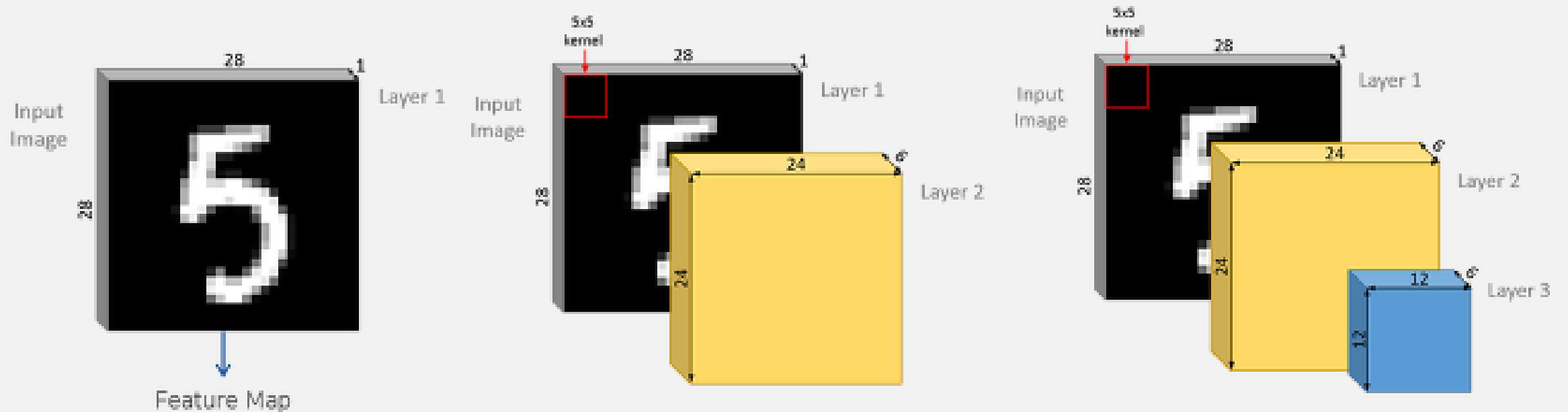
- 데이터 형상의 무시
- 변수의 개수
- 네트워크 크기
- 학습시간의 문제



이미지 데이터의 경우 3차원(세로, 가로, 채널)의 형상을 가지며, 공간적 구조(spatial structure)를 지님.

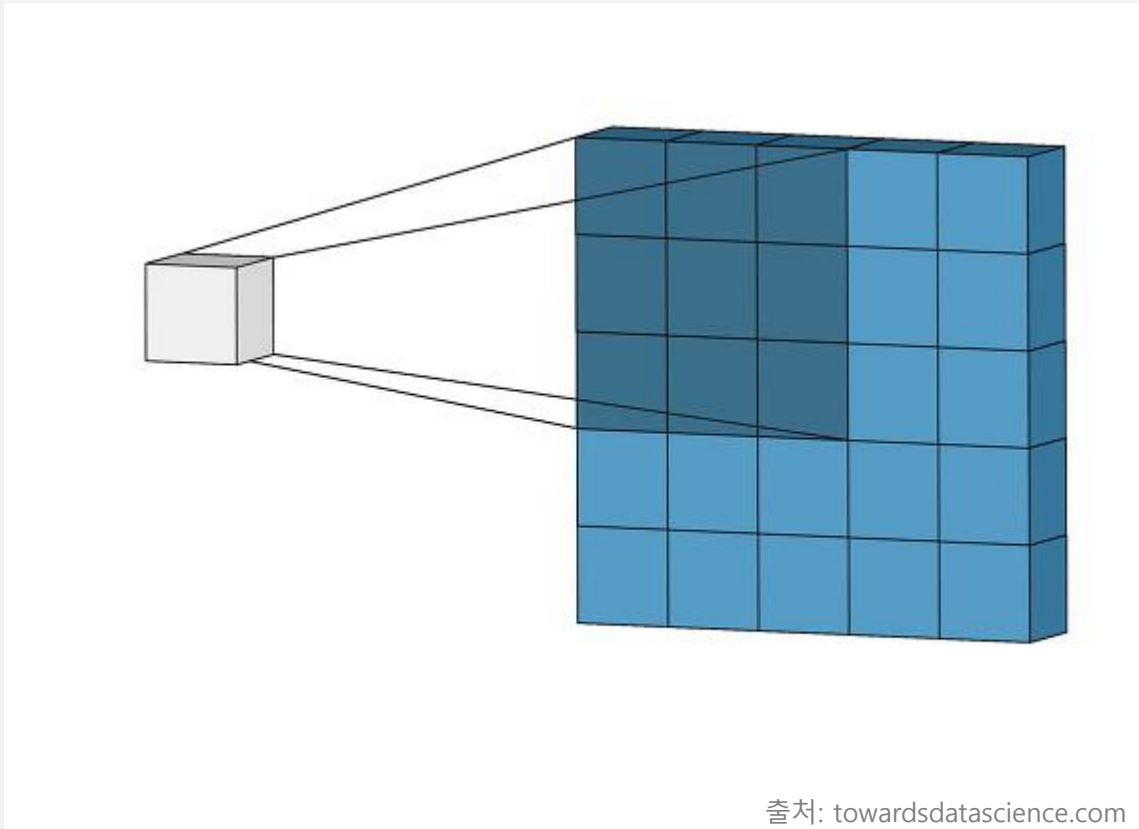
예를 들어 공간적으로 가까운 픽셀은 값이 비슷하거나, RGB의 각 채널은 서로 밀접하게 관련 있음

2. 합성곱층(Convolutional Layer, Conv Layer)



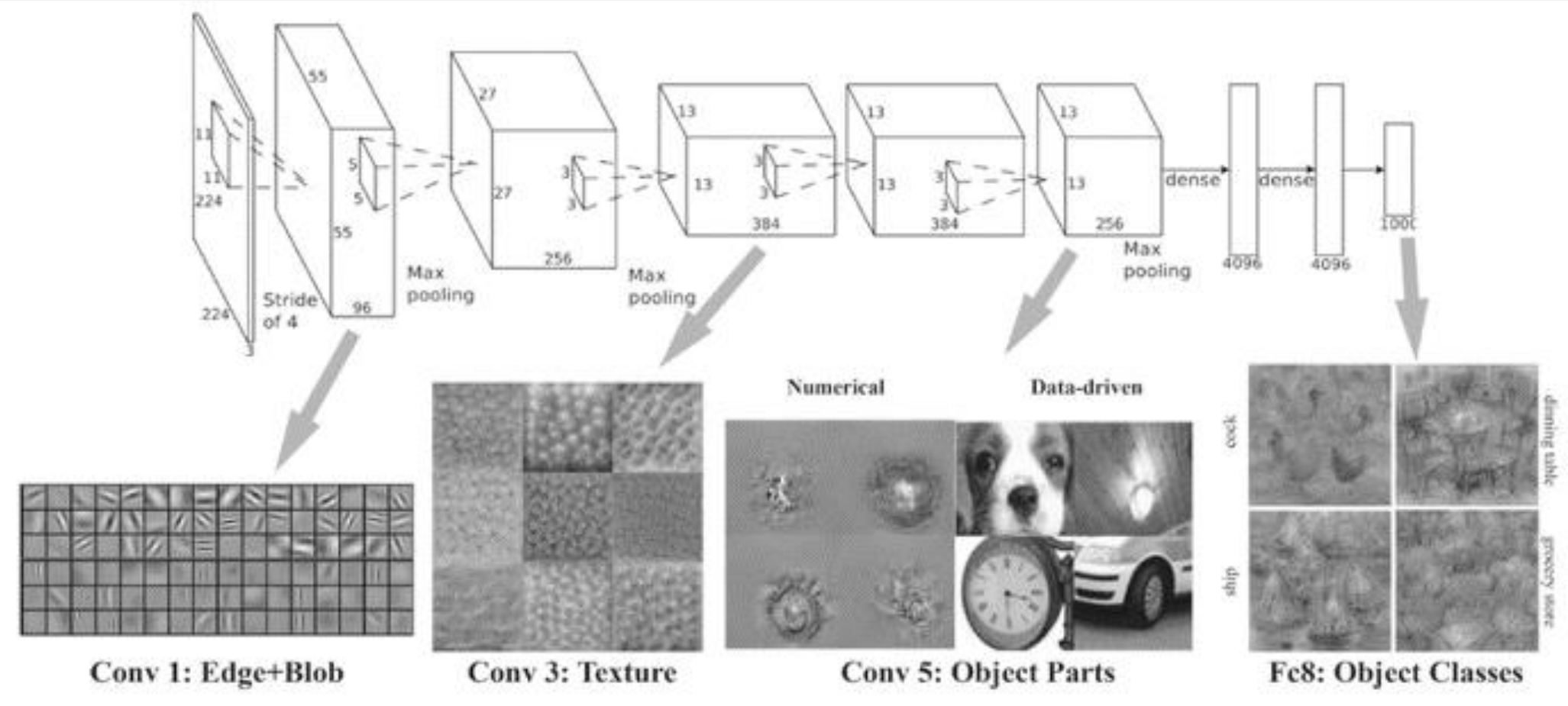
- 이미지 데이터는 일반적으로 채널, 세로, 가로 이렇게 3차원으로 구성된 데이터
- 합성곱에서는 3차원 데이터(1, 28, 28)를 입력하고 3차원의 데이터로 출력하므로 형상을 유지 가능
- CNN에서는 이러한 입출력 데이터를 특징맵(Feature Map)이라고 함

2. 합성곱층(Convolutional Layer, Conv Layer)

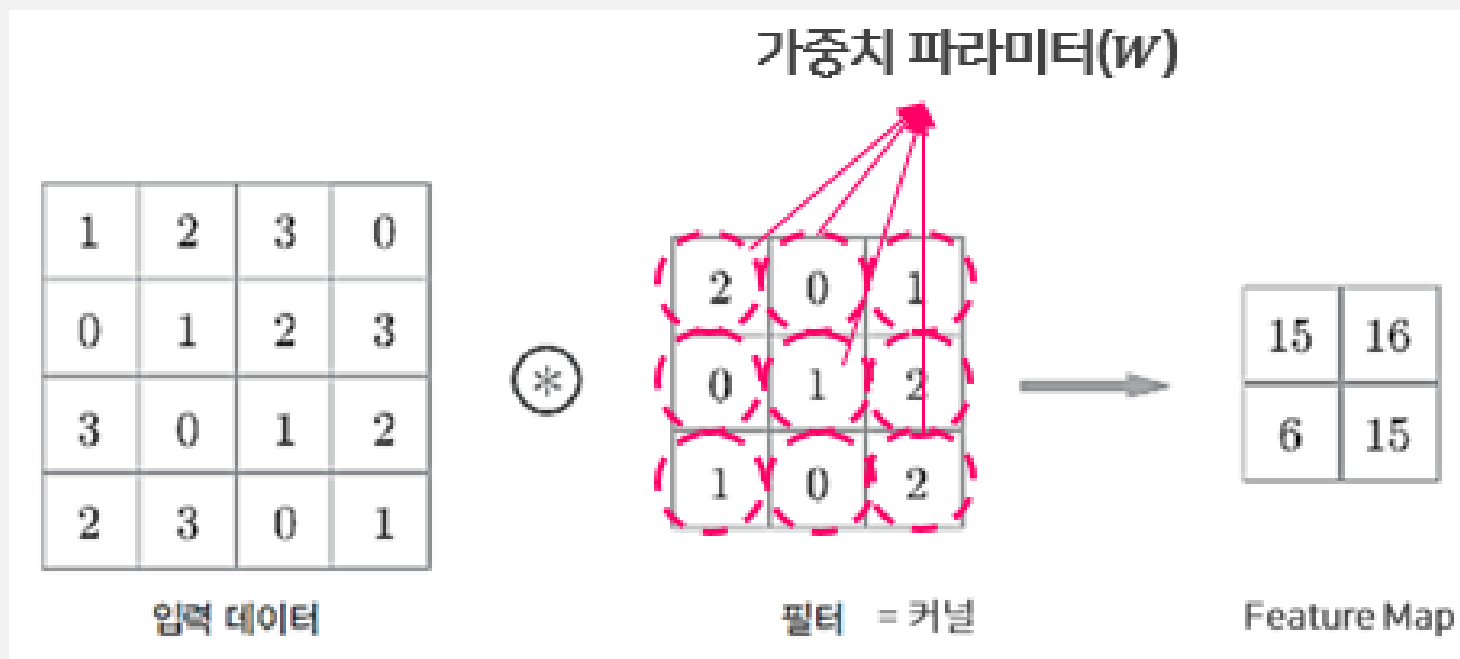


- 합성곱층 뉴런의 수용영역(receptive field)안에 있는 픽셀에만 연결
- 앞의 합성곱층에서는 저수준 특성에 집중하고, 그 다음 합성곱층에서는 고수준 특성으로 조합

2. 합성곱층(Convolutional Layer, Conv Layer)

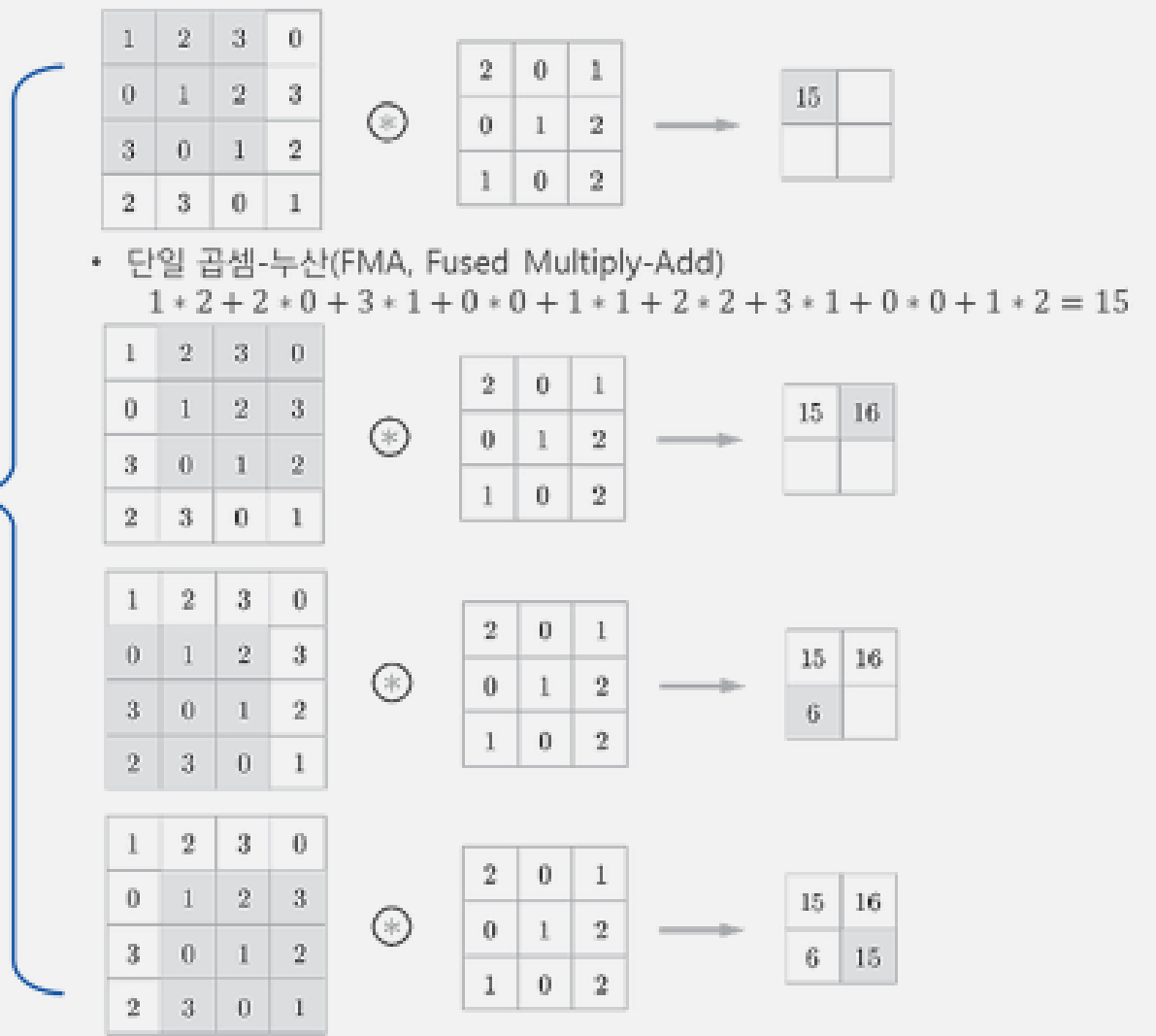
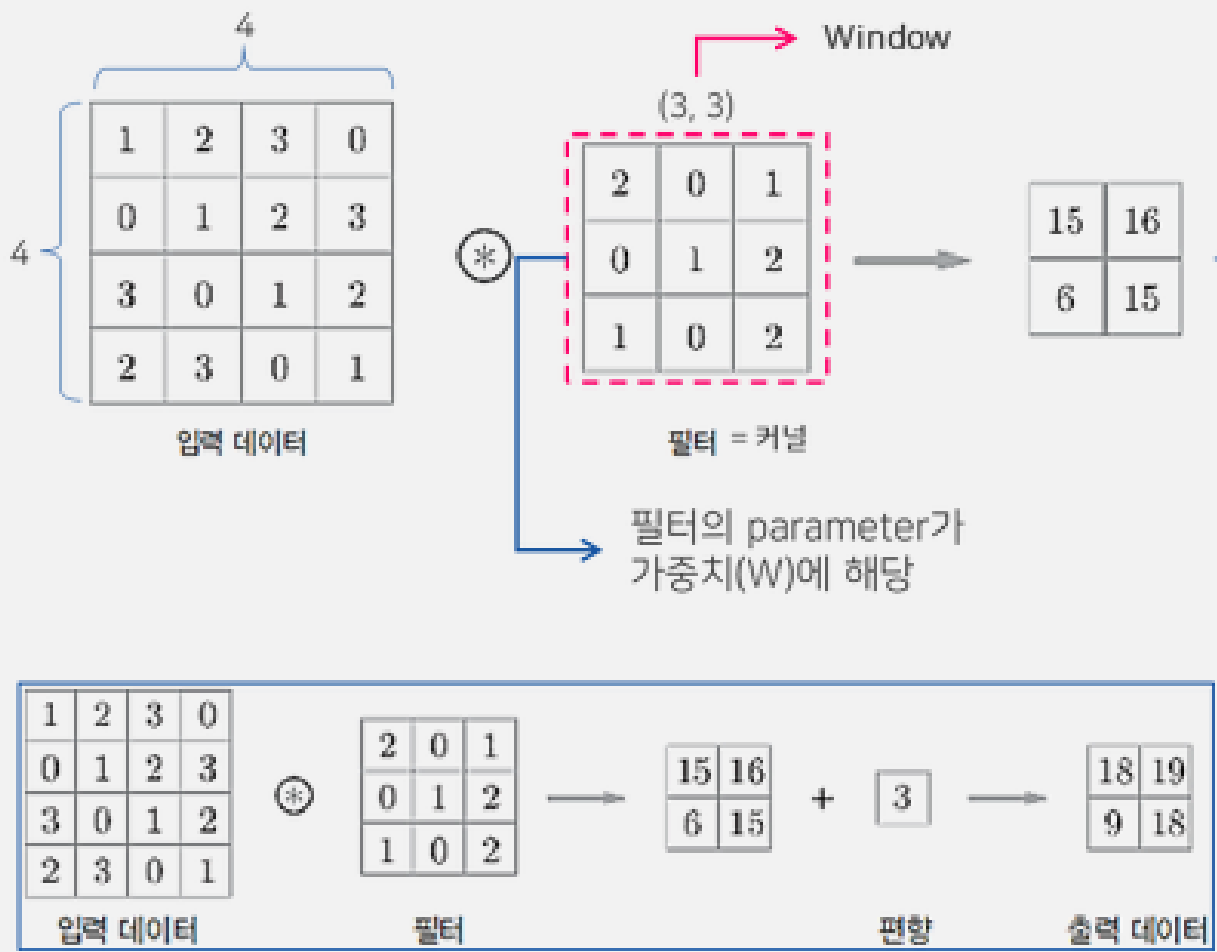


3. 필터(Filter)

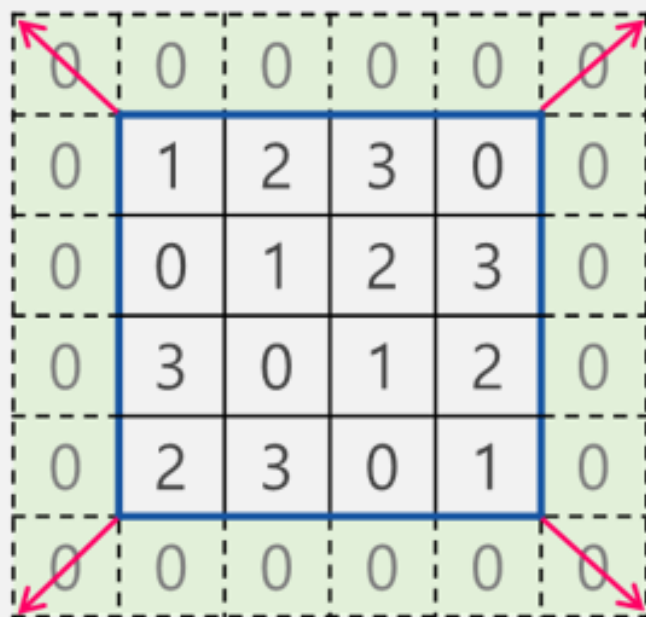


- 필터가 합성곱층에서의 가중치 파라미터(W)에 해당
- 학습단계에서 적절한 필터를 찾도록 학습
- 입력데이터에 필터를 적용하여 필터와 유사한 이미지의 영역을 강조하는 특성맵(feature map)을 출력하여 다음 층(layer)으로 전달

4. 합성곱의 연산



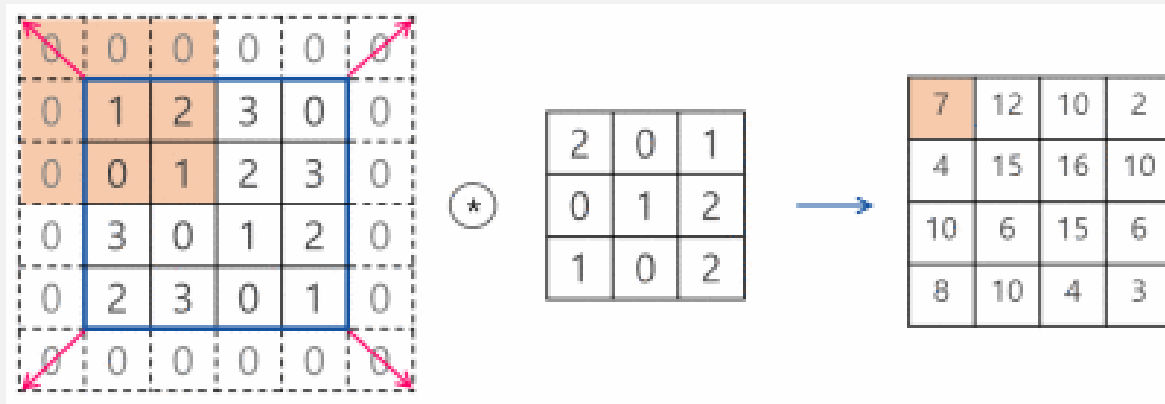
5. 패딩(Padding)



1쪽짜리 zero- padding

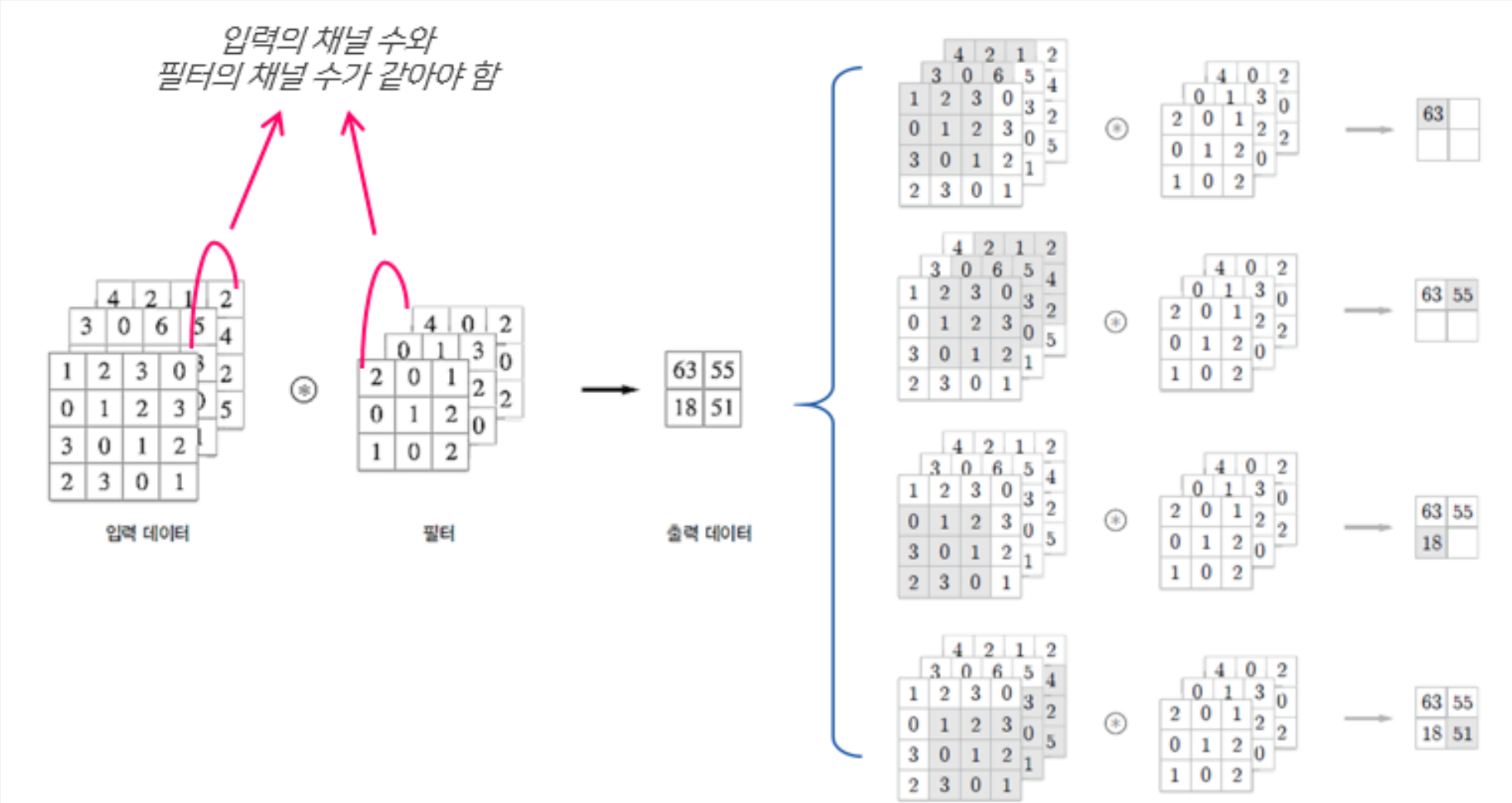
- 데이터의 크기는 Conv Layer를 지날 때 마다 작아짐
- 가장자리 정보가 사라지는 문제 발생
- 합성곱 연산을 수행하기 전, 입력데이터 주변을 특정값으로 채워 늘리는 것
- 주로 zero-padding을 사용함

6. 스트라이드(Stride)

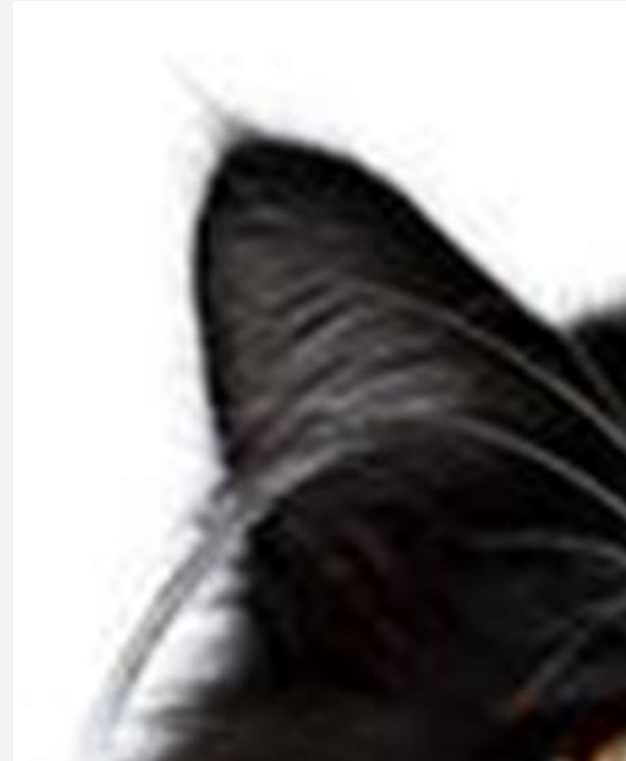
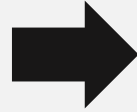


- 필터가 이동할 간격
- 출력 데이터의 크기를 조절하기 위해 사용
- 보통 1과 같이 작은 값이 더 잘 작동됨

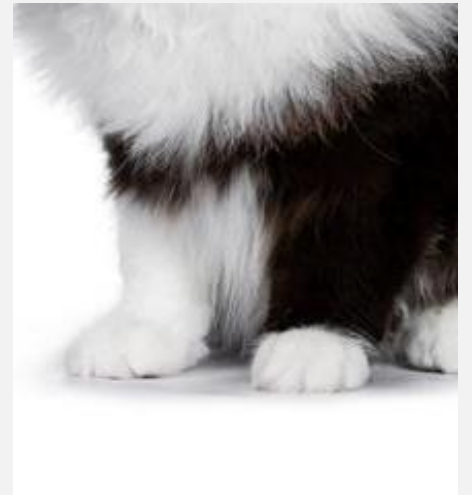
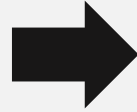
7. 3차원 데이터의 합성곱



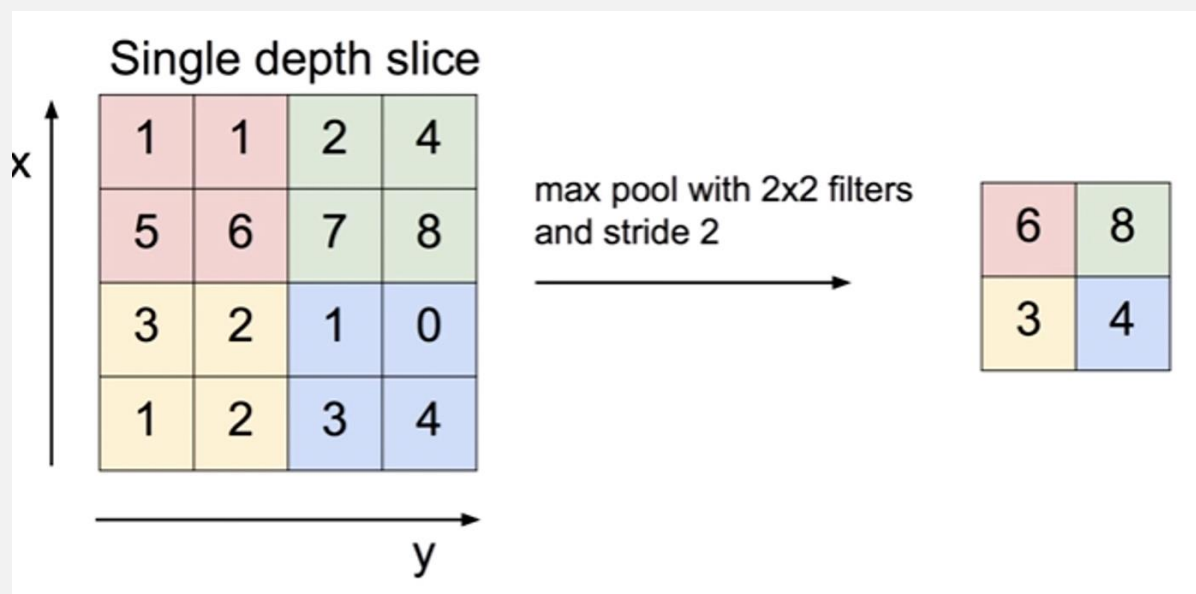
8. 풀링층(Pooling layer)



8. 풀링층(Pooling layer)



8. 풀링층(Pooling layer)



- 그림의 사이즈를 점진적으로 줄이는 법
Max-Pooling, Average Pooling
- Max-Pooling : 해당영역에서 최대값을 찾는 방법
- Average Pooling : 해당영역에서 평균값을 계산하는 방법

왼쪽 이미지는
2 x 2 filter로 2 stride를 적용하여
4x4 이미지를 2x2이미지로 변환한 예제

```
from keras.models import Sequential
from keras.layers import Dense, Conv2D, Flatten
```

```
model = Sequential()
# model.add(Dense(units=10, input_shape=(3,)))
model.add(Conv2D(filters=10, kernel_size=(3, 3), # kernal_size는 이미지를 자르는 규격을 의미
                 input_shape=(8, 8, 1))          # (rows, rows, channels)의 형태 => channels의 1은 흑백, 3은 칼라)
model.add(Conv2D(7, (2, 2), activation='relu'))
model.add(Flatten())
model.add(Dense(32, activation='relu'))
model.add(Dense(10, activation='softmax'))
```

```
from keras.models import Sequential
from keras.layers import Dense, Conv2D, Flatten, MaxPooling2D

model = Sequential()
model.add(Conv2D(filters=64, kernel_size=(3, 3),
padding='same',
input_shape=(28, 28, 1)))
model.add(MaxPooling2D())
model.add(Conv2D(32, (2, 2),
padding='valid',
activation='relu'))
# padding = 'valid' 가 default
model.add(Flatten())
model.add(Dense(32, activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(10, activation='softmax'))
```

이미지 분석 - 데이터셋

MNIST 손글씨 이미지 분류하기

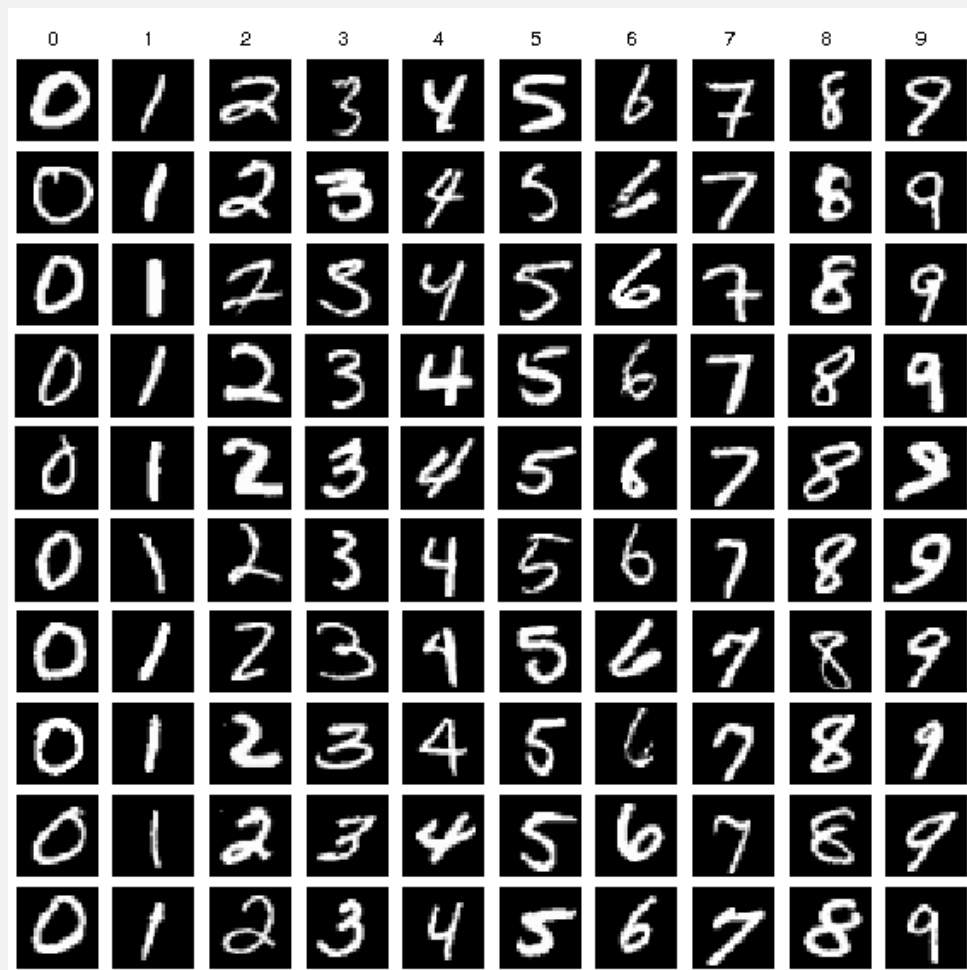


Figure 1. MNIST 손글씨 이미지 데이터셋

- 0 ~ 9까지 10가지로 분류될 수 있는 손글씨 숫자 이미지 70,000개
- train- set 60,000개, test-set 10,000개로 구성
- 28x28 픽셀로 구성되어 있음
- 0 ~ 255 사이의 숫자 행렬로 표현됨

NAVER Cloud x **BIT** BITCamp

- 이미지는 0 ~ 255 사이의 값을 갖는 28x28 크기의 NumPy array
- 이미지는 디지털화된 형태의 픽셀(pixel)들로 구성됨.
- 각 픽셀은 0에서 255 사이의 값을 가질 수 있는데, 이는 색상의 강도(intensity)를 나타냄
- 0 이 가장 어두운 색이며, 숫자가 높을 수록 밝은 색

사례2. MNIST 손글씨 이미지 데이터셋

FASHOIN MNIST

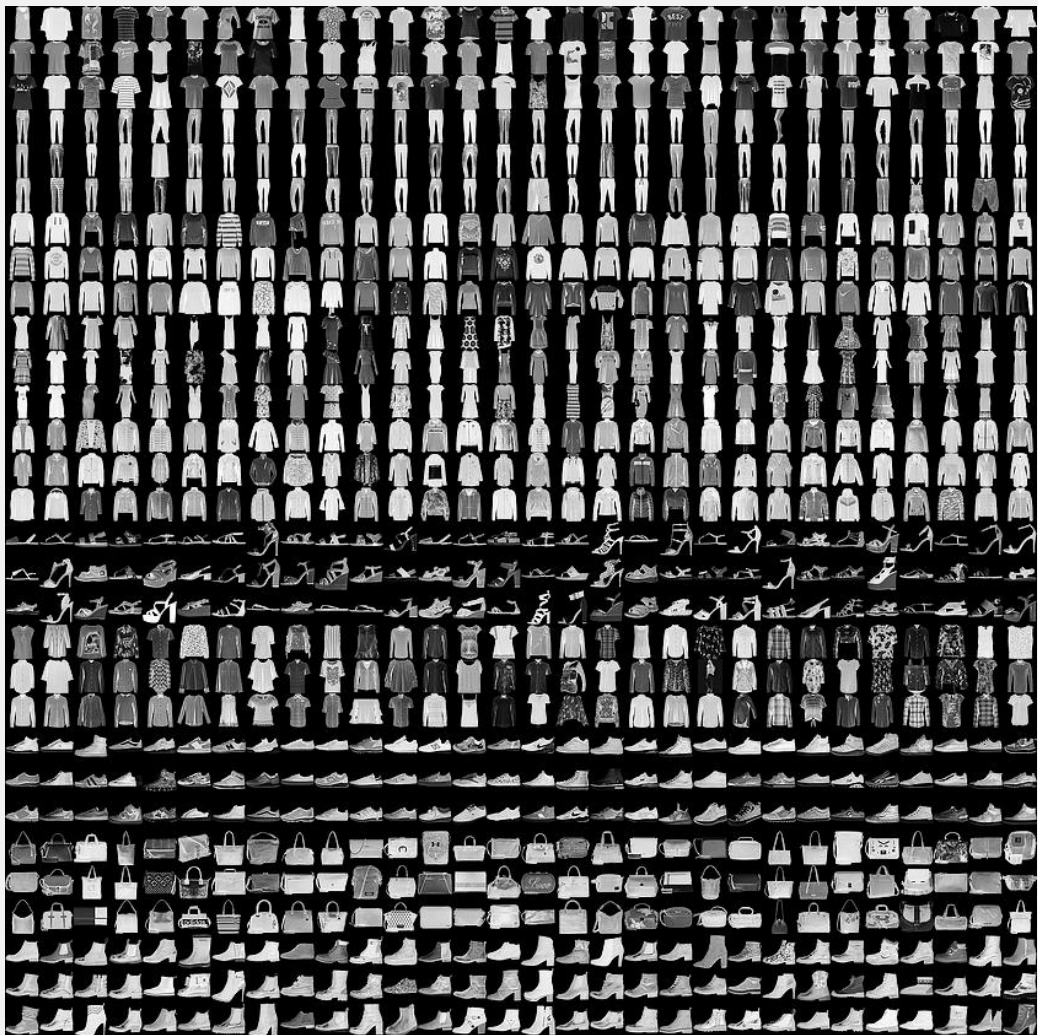


Figure 3 패션-MNIST 샘플 (Zalando, MIT License)

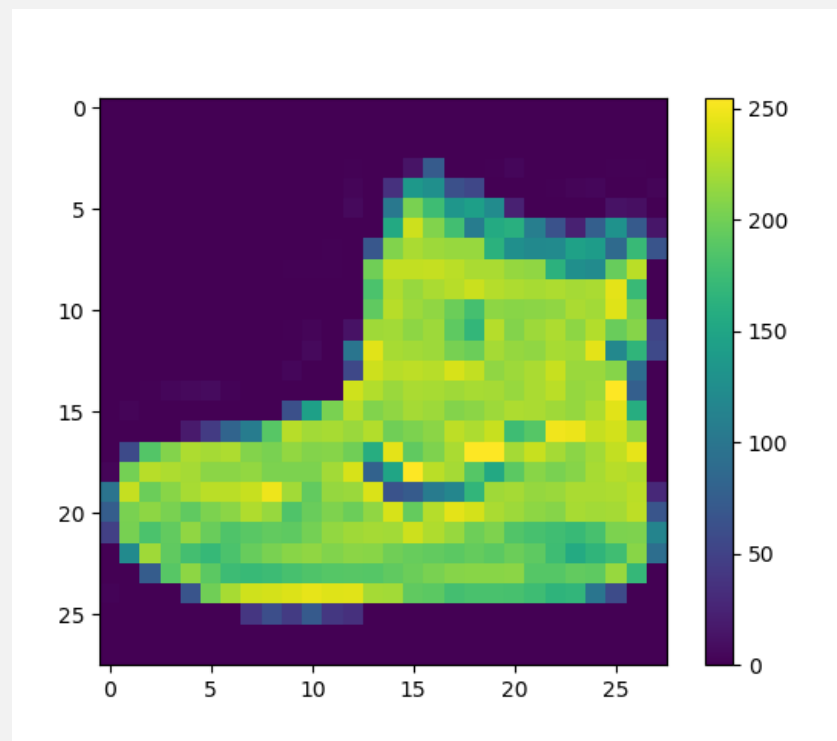
- train- set 60,000개, test-set 10,000개로 구성된 이미지 데이터 세트
- 10개 클래스의 레이블과 28x28 회색조 이미지

Label	Description
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

FASHOIN MNIST

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 13 73 0 0 1 4 0 0 0 1 1 0]
0 0 0 0 0 0 0 0 0 0 0 0 3 0 36 136 127 62 54 0 0 0 1 3 4 0 0 3]
0 0 0 0 0 0 0 0 0 0 0 0 6 0 102 204 176 134 144 123 23 0 0 0 12 10 0]
0 0 0 0 0 0 0 0 0 0 0 0 0 0 155 236 207 178 107 156 161 109 64 23 77 130 72 15]
0 0 0 0 0 0 0 0 0 0 0 1 0 69 207 223 218 216 216 163 127 121 122 146 141 88 172 66]
0 0 0 0 0 0 0 0 0 1 1 1 0 200 232 232 233 229 223 223 215 213 164 127 123 196 229 0]
0 0 0 0 0 0 0 0 0 0 0 0 0 183 225 216 223 228 235 227 224 222 224 221 223 245 173 0]
0 0 0 0 0 0 0 0 0 0 0 0 0 193 228 218 213 198 180 212 210 211 213 223 220 243 202 0]
0 0 0 0 0 0 0 0 0 1 3 0 12 219 220 212 218 192 169 227 208 218 224 212 226 197 209 52]
0 0 0 0 0 0 0 0 0 6 0 99 244 222 220 218 203 198 221 215 213 222 220 245 119 167 56]
0 0 0 0 0 0 0 0 0 4 0 55 236 228 230 228 240 232 213 218 223 234 217 217 209 92 0]
0 1 4 6 7 2 0 0 0 0 0 237 226 217 223 222 219 222 221 216 223 229 215 218 255 77 0]
3 0 0 0 0 0 0 0 62 145 204 228 207 213 221 218 208 211 218 224 223 219 215 224 244 159 0]
0 0 0 18 44 82 107 189 228 220 222 217 226 200 205 211 230 224 234 176 188 250 248 233 238 215 0]
57 187 208 224 221 224 208 204 214 208 209 200 159 245 193 206 223 255 255 221 234 221 211 220 232 246 0]
92 228 224 221 211 211 214 205 205 205 220 240 80 150 255 229 221 188 154 191 210 204 209 222 228 225 0]
33 198 210 222 229 229 234 249 220 194 215 217 241 65 73 106 117 168 219 221 215 217 223 223 224 229 29]
34 212 204 193 205 211 225 216 185 197 206 198 213 240 195 227 245 239 223 218 212 209 222 220 221 230 67]
33 183 194 213 197 185 190 194 192 202 214 219 221 220 236 225 216 199 206 186 181 177 172 181 205 206 115]
22 219 193 179 171 183 196 204 210 213 207 211 210 200 196 194 191 195 191 198 192 176 156 167 177 210 92]
0 74 189 212 191 175 172 175 181 185 188 189 188 193 198 204 209 210 210 211 188 188 194 192 216 170 0]
0 0 0 66 200 222 237 239 242 246 243 244 221 220 193 191 179 182 182 181 176 166 168 99 58 0 0]
0 0 0 0 0 0 40 61 44 72 41 35 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
```

사례4. 패션-MNIST 샘플 중 신발



CIFAR-10 이미지 분류

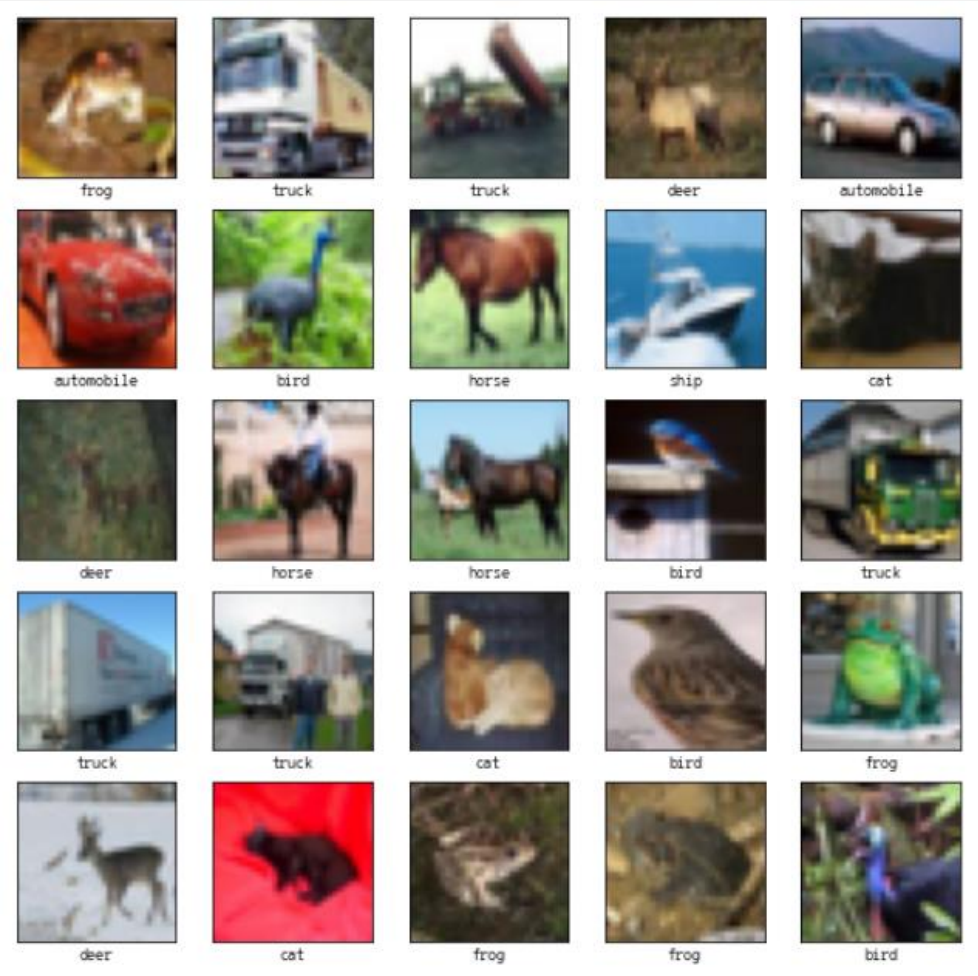


Figure 5. CIFAR-10 소형 이미지 분류 데이터

- 10개 클래스의 60000개 32x32 컬러 이미지로 구성되며 클래스당 6000개의 이미지
- 50000개의 훈련 이미지와 10000개의 테스트 이미지

Label	Description
0	airplane
1	automobile
2	bird
3	cat
4	deer
5	dog
6	frog
7	horse
8	ship
9	truck

CIFAR-100 이미지 분류

- 이 데이터 세트는 각각 600개의 이미지를 포함하는 100개의 클래스가 있다는 점을 제외하면 CIFAR-10과 동일함

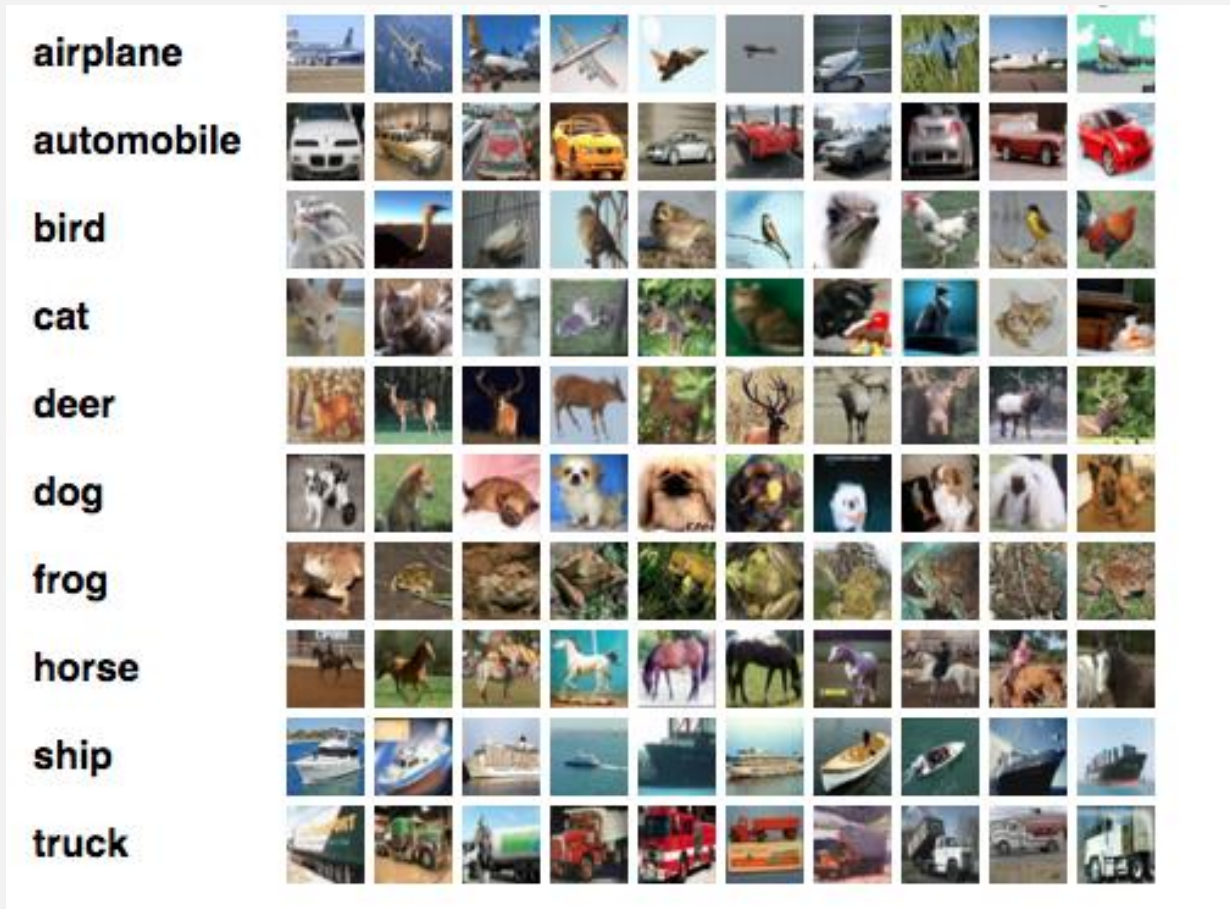


Figure 6. CIFAR-100 이미지 분류 데이터

```
label= ['beaver', 'dolphin', 'otter', 'seal', 'whale', # aquatic mammals
        'aquarium', 'fish', 'flatfish', 'ray', 'shark', 'trout', # fish
        'orchids', 'poppies', 'roses', 'sunflowers', 'tulips', # flowers
        'bottles', 'bowls', 'cans', 'cups', 'plates', # food containers
        'apples', 'mushrooms', 'oranges', 'pears', 'sweet peppers', # fruit and vegetables
        'clock', 'computer', 'keyboard', 'lamp', 'telephone', 'television', # household electrical devices
        'bed', 'chair', 'couch', 'table', 'wardrobe', # household furniture
        'bee', 'beetle', 'butterfly', 'caterpillar', 'cockroach', # insects
        'bear', 'leopard', 'lion', 'tiger', 'wolf', # large carnivores
        'bridge', 'castle', 'house', 'road', 'skyscraper', # large man-made outdoor things
        'cloud', 'forest', 'mountain', 'plain', 'sea', # large natural outdoor scenes
        'camel', 'cattle', 'chimpanzee', 'elephant', 'kangaroo', # large omnivores and herbivores
        'fox', 'porcupine', 'possum', 'raccoon', 'skunk', # medium-sized mammals
        'crab', 'lobster', 'snail', 'spider', 'worm', # non-insect invertebrates
        'baby', 'boy', 'girl', 'man', 'woman', # people
        'crocodile', 'dinosaur', 'lizard', 'snake', 'turtle', # reptiles
        'hamster', 'mouse', 'rabbit', 'shrew', 'squirrel', # small mammals
        'maple', 'oak', 'palm', 'pine', 'willow', # trees
        'bicycle', 'bus', 'motorcycle', 'pickup truck', 'train', # vehicles 1
        'lawn-mower', 'rocket', 'streetcar', 'tank', 'tractor' # vehicles 2]
```

실습

1. CNN 시각화
2. CNN - MNIST
3. CNN - Fashion MNIST
4. CNN - cifar10
5. CNN - cifar100
6. Pandas
7. Heatmap (matplotlib + seaborn)

Day04. 인공지능 Study

1. 인공지능 개념 정리 - 머신러닝, 딥러닝
2. 퍼셉트론 (Perceptron)
3. 다층 퍼셉트론 (Multi-Layer Perceptron: MLP)
4. 옵티마이저 (Optimizer)
5. 학습률 (learning rate)
6. 경사하강법 (Gradient Descent)
7. 손실함수 (Loss Function)
8. 활성화 함수 (Activation Function) - Sigmoid, ReLU, Softmax

9. 회귀분석

10. 결정계수 R2 score

11. 분류분석

12. 원 핫 인코딩 (One Hot Encoding)

13. 난수값 (random_state)

14. 정확도 accuracy score

15. 과적합 (overfitting)

16. 합성곱신경망(CNN)

수고하셨습니다.