

04. 딥러닝(DNN, CNN) #src: 9_ML_DL/07_딥러닝기초~12_CNN_mnist 참고

1) DNN(Deep Neural Network); 입력층과 출력층 사이에 여러 개의 은닉층이 있는 인공 신경망

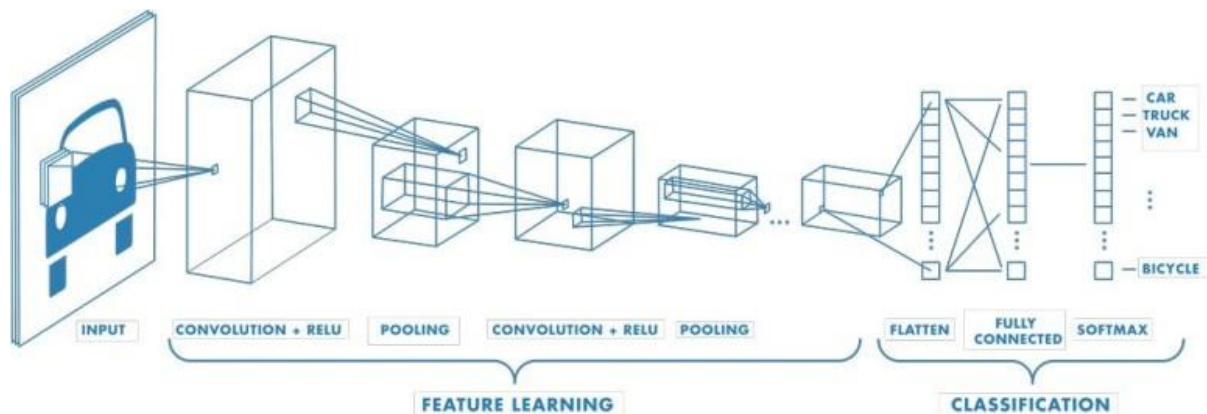
- (1) 단층 퍼셉트론; 중간층이 하나의 노드(or뉴런)으로 구성 ⇒ XOR학습 불가 (∴딥러닝으로 학습)
- (2) 다층 퍼셉트론(MLP); 중간층을 구성하는 노드 여러개, 중간층이 다수로 구성되어 있는 구조
⇒ 입력층과 출력층 사이에 존재하는 중간층을 은닉층(Hidden Layer)라고 부름(딥러닝)
- (3) 입출력 모두 연결해주는 Dense 레이어; 가중치 ↑ ⇒ 입력 뉴런이 출력 뉴런에 미치는 영향 ↑

```
model = Sequential()
model.add(Dense(10, input_dim=2, activation='relu', init='uniform'))
model.add(Dense(20, activation="relu"))
model.add(Dense(10, activation="relu"))
model.add(Dense(1, activation="sigmoid"))
```

#첫번째 인자(10, 20, 10, 1): 출력 뉴런의 수, input_dim: 입력 뉴런의 수
#activation(활성화 함수): linear, relu(주로 은닉층), sigmoid(이진 분류), softmax(다중 분류)
#init(가중치 초기화 방법): uniform(균일 분포), normal(가우시안 분포) – 생략 가능

2) CNN(Convolutional Neural Network); 주로 이미지나 영상 데이터를 처리할 때 사용

- CNN은 이미지의 공간적/지역적 정보를 유지한 채 특징들을 추출, 이미지의 픽셀과 주변 픽셀들의 연관성을 학습 (※DNN은 기본적으로 1차원 형태의 데이터를 사용하므로 2, 3차원 이미지를 1차원으로 바꾸는 과정에서 공간적/지역적 정보가 손실됨)



- (1) Conv2D; 필터링을 통해 이미지의 특징(패턴)을 추출
- (2) MaxPool2D(최대 풀링); 데이터 크기(파라미터 개수) 줄여주고 사소한 변화 무시해줌
- (3) Flatten; 2차원 영상 데이터를 1차원으로 바꿔줌

```
Conv2D(32, (5,5), input_shape=(28, 28, 1), activation='relu')
```

#필터 수, (행, 열), (행, 열, 채널 수), 활성화 함수 순 #padding='valid'; 유효한 영역만 출력(생략)