"""

<신경망 종류>

- 인공신경망, 심층신경망

<인공신경망(Artificial Neural Network, ANN)>

- 은닉계층(Hidden Layer)이 없는 경우

- 입력계층 + 출력계층 또는 입/출력 계층만 사용하는 경우

- 1개인 계층 : 입력 - 계층을 1개 또는 2개를 사용함

(Input)과 출력(Output)을 동시에 담당하는 계층

- 2개인 계층 : 입력계층(Input Layer)과, 출력계층(Output Layer) 2개 사용됨

<심층신경망(Deep Neural Network, DNN)>

- 은닉계층(Hidden Layer)이 있는 경우

- 입력계층 + 은닉계층...n개 + 출력계층

- 계층이 3개 이상 사용함

- 첫번째 계층 : 입력계층(Input Layer)

- 마지막 계층 : 출력계층(Output Layer)

- 중간계층 : 은닉계층(Hidden Layer)

(히든 계층이라고도 칭함, 여러개 계층 존재 가능)

<계층>

- 계층은 여러 계층이 존재함

- 훈련에 참여하는 계층과 훈련에 참여하지 않는 계층으로 구분합니다.

- 훈련에 참여하는 계층은 주로 Dense()로 생성이 되며,

- 훈

련에 참여하지 않는 계층을 "전처리 계층" 이라고 합니다.

"""

"""

- compile() : 손실을 줄이기 위한 방법을 설정하는 함수

- loss : 손실을 줄이기 위해 사용되는 함수 설정(손실함수라고 칭합니다.)

: 손실함수에는 categorical\_crossentropy, sparse\_categorical\_crossentropy,

binary\_crossentropy가 있습니다.

(손실함수)

\* categorical\_crossentropy

- 다중분류에 사용됨(특별한 경우에만 사용됨)

- 단, 종속변수 값의 형태가 원-핫인코딩된 데이터인 경우에 사용됩니다.

- 원-핫인코딩 종속변수 형태 : [[0, 0, 1], [0, 1, 0], [1, 0, 0] ...]

\* sparse\_categorical\_crossentropy

- 다중분류에 사용됨(분류에서 주로 사용됨)

- 종속변수 형태 : [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ...]

\* binary\_crossentropy

- 이진분류에 사용됨

- 종속변수 형태 : [0, 1, 1, 0, 0, 1...]

- metrics : 훈련시 출력할 값 -> 정확도 값 출력(손실률은 기본적으로 출력됨)

"""

### 4. 모델 훈련시키기

# - fit()만 재실행하면, 기존의 훈련에서 계속 이어서 수행하게 됩니다.

# - 깨끗한 상태로 재실행(초기화)하려면, 모델 생성부터 다시 실행해야 합니다.

# - epochs : 훈련 반복 횟수

# : 훈련을 반복하면서 스스로 가중치를 부여하면서 성능을 높여 갑니다.

# : 반복 횟수가 많다고 해서 성능일 좋아진다고 볼 수는 없습니다.

# (좋아질수도, 나빠질수도 있습니다.)

# : 성능에 영향을 미치는 값이기에 -> 하이퍼파라메터 튜닝 대상 입니다.

# - 최적의 epoch 확인 방법

# : loss(손실율)은 낮고, accuracy(정확도)는 높은 시점의 epoch를 선정합니다.

# - 훈련에 사용된 데이터 갯수 : 1500

# 1500/1500 = 총 데이터 갯수 / 배치사이즈(기본값 32)

# = 훈련데이터 48,000건 / 기본값 32 = 1,500

with tf.device("/CPU:0") :

model.fit(train\_scaled, train\_target, epochs=5)

### 모델에서 사용한 계층 확인하기

# - Model: "sequential" : 모델 이름

# - Layer (type) : 모델 이름

# - Output Shape : 출력 갯수(크기)

# - Param : 훈련 중에 사용된 특성의 총 갯수

# (Param 계산방법) : 모델이 알아서 하기에 우리는 몰라도 됩니다.

# Param = (입력의 갯수 + 1) \* 출력 갯수

# = (784 + 1) \* 10

# = 7850