

인공신경망

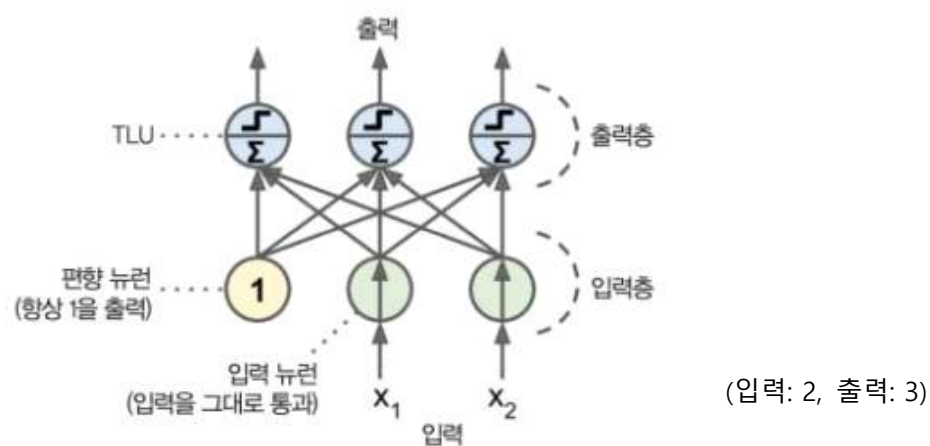
- 뇌에 있는 생물학적 뉴런의 네트워크에서 영감을 받은 머신러닝 모델
- 딥러닝의 핵심, 강력하며 확장성이 좋음

인공 뉴런

- 생물학적 뉴런에 착안한 매우 단순한 신경망 모델
- 하나 이상의 이진(ON/OFF) 입력과 이진 출력 하나를 가짐
- 입력이 일정 개수만큼 활성화되었을 때 출력을 내보냄

퍼셉트론

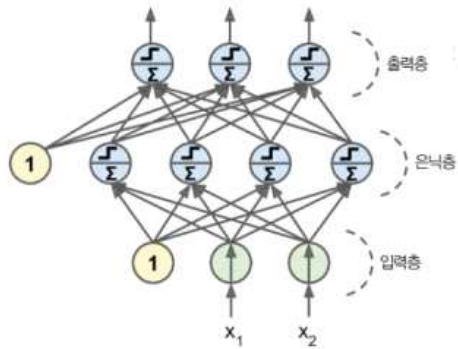
- 가장 간단한 인공 신경망 구조
- 모든 입력은 가중치와 연결
- TLU 또는 LTU라 불리는 인공 뉴런 활용



- 알고리즘
 - 오차가 감소되도록 가중치를 조절하며 뉴런 사이의 관계를 강화
 - 하나의 샘플이 입력될 때 마다 예측한 후에 오차를 계산하여 오차가 줄어드는 방향으로 가중치 조절

다층 퍼셉트론

- 퍼셉트론을 여러 개 쌓아 올린 인공신경망
- 입력층 하나와 은닉층이라 불리는 하나 이상의 TLU층과 출력층으로 구성
- 모든 층은 편향을 포함하며, 다음 층과 완전히 연결되어 있음



- 완전연결 층
 - 층에 속한 각각의 뉴런이 이전 층의 모든 뉴런과 연결되어 있을 때 가리킴
- 심층신경망
 - 여러 개의 은닉층을 쌓아 올린 인공 신경망
- 역전파 훈련 알고리즘
 - 1단계(정방향): 각 훈련 샘플에 대해 먼저 예측을 만든 후 오차 측정
 - 2단계(역방향): 역방향으로 각 층을 거치면서 각 출력연결이 오차에 기여한 정도 측정 (미적분의 연쇄법칙(Chain rule) 적용)
 - 3단계: 오차가 감소하도록 모든 가중치 조정

케라스로 다층 퍼셉트론 구현하기

- 모든 종류의 신경망을 손쉽게 만들어 주는 최상위 딥러닝 API 제공

케라스 시퀀셜 API 활용한 이미지 분류

- 모델 만들기
 - Sequential 클래스 내에 층을 쌓아 순차적 학습 지원

```
model = keras.models.Sequential([
    keras.layers.Flatten(input_shape=[28, 28]),
    keras.layers.Dense(300, activation="relu"),
    keras.layers.Dense(100, activation="relu"),
    keras.layers.Dense(10, activation="softmax")
])
```

- 모델 컴파일

- 손실함수, 옵티마이저, 평가기준 등을 지정함.

```
model.compile(loss="sparse_categorical_crossentropy",
              optimizer="sgd",
              metrics=["accuracy"])
```

- 모델 훈련과 평가

- 모델 훈련: fit() 메서드 호출

```
history = model.fit(X_train, y_train, epochs=30,
                    validation_data=(X_valid, y_valid))
```

- 모델 평가

- 모델 평가: evaluate() 메서드 호출

```
model.evaluate(X_test, y_test)
```

- 모델을 사용한 예측

- 예측 : predict() 메서드 호출
- predict 메서드: 각 클래스에 속할 확률 계산
- predict_class: 가장 높은 확률의 클래스 지정.

```
X_new = X_test[:3]
y_proba = model.predict(X_new)
y_proba.round(2)
```

콜백 사용하기

- 체크포인트(checkpoint) 저장 시 사용
 - 대규모 데이터셋에서 훈련 시에 훈련 도중 일정 간격으로 체크포인트를 저장해야 함
- 조기 종료
 - EarlyStopping 콜백 사용

```
model.compile(loss="mse", optimizer=keras.optimizers.SGD(learning_rate=1e-3))
early_stopping_cb = keras.callbacks.EarlyStopping(patience=10,
                                                  restore_best_weights=True)
history = model.fit(X_train, y_train, epochs=100,
                    validation_data=(X_valid, y_valid),
                    callbacks=[checkpoint_cb, early_stopping_cb])
mse_test = model.evaluate(X_test, y_test)
```

인공 뉴런과 퍼셉트론의 차이점은 무엇인가요?

인공 뉴런: 하나 이상의 이진(ON/OFF) 입력과 이진 출력을 갖습니다.

퍼셉트론: 인공 신경망 구조

딥러닝에서의 역전파(Backpropagation)가 신경망의 학습에 어떻게 기여하는지?

↳ 신경망이 예측한 결과와 실제 정답 간의 차이(오차)를 계산하고 차이를 줄인다.