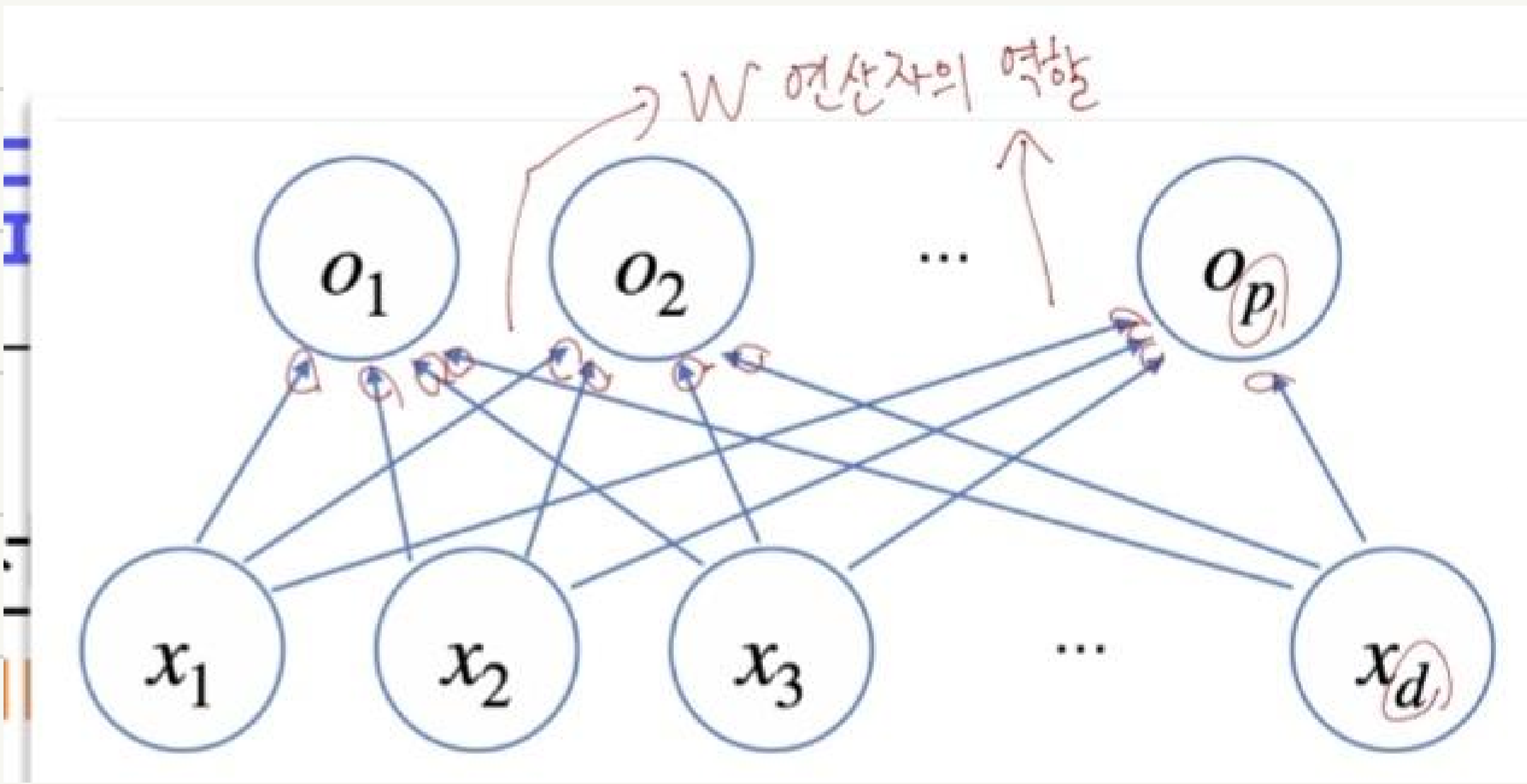


# # 딥러닝 학습 방법 이해하기

선형모델

$$\begin{matrix} \text{출력} & = & \text{입력} & \times & \text{가중치행렬(연산자)} & + & \text{편편} \\ (n \times p) & & (n \times d) & & (d \times p) & & (n \times p) \end{matrix}$$

(모든 행에 대해서  
복제한 행렬  
(행의 값이 다 같음))



$\text{softmax}(o)$ : 출력 벡터  $o$ 에  $\text{softmax}$  함수를 합성하면  
출력 벡터가 확률 벡터가 됨 (분류 문제)  
(확률을 할 때는 원-핫 벡터로 최대값을 가진 주소만  
1로 출력하는 연산을 사용해서  $\text{softmax}$ 를 사용하지 않음)

$$\text{softmax}(o) = \left( \frac{\exp(o_1)}{\sum_{k=1}^p \exp(o_k)}, \dots, \frac{\exp(o_p)}{\sum_{k=1}^p \exp(o_k)} \right)$$

(전행의 결과물)      (지수함수)

신경망: 선형모델 + 합성 함수

↳ 비선형 근사를 하기 위해 사용되는 함수  
(합성 함수가 없으면 딥러닝은 선형모델과 차이가 없음)  
→ Sigmoid, tanh, ReLU...

다층 퍼셉트론 (MLP): 신경망이 여러층 합성된 함수

⇒ 층이 깊을수록 목적 함수를 근사하는 데 필요한 뉴런(노드)이 빨리 줄어들어 효율적 학습 가능

역전파 알고리즘

↳  $L = 1, \dots, L$  까지 순차적인 신경망 계산: 순전파  
↳  $L = L, \dots, 1$  순서로 연쇄법칙을 통해 그라디언트 벡터 계산  
↳ 연쇄법칙 기반 자동 미분

(역전파는 각 노드의 텐서값 (각 뉴런의 값)을 메모리에 다 기억해야 함  
∴ 순전파보다 메모리가 많이 필요.)

딥러닝 학습 방법 = 역전파 + 확률적 경사하강법