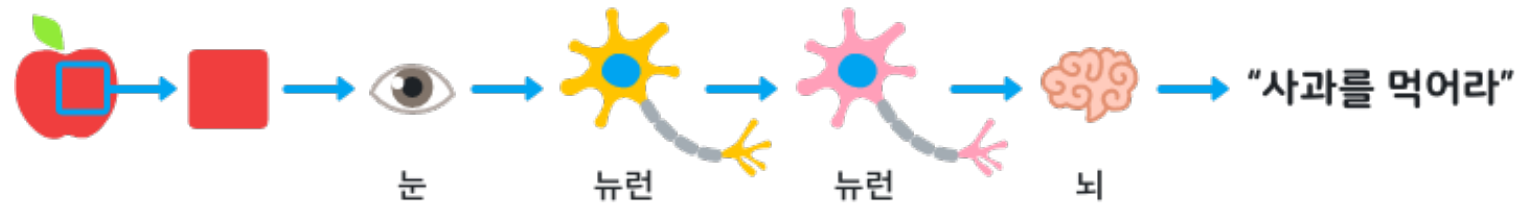


Course 10

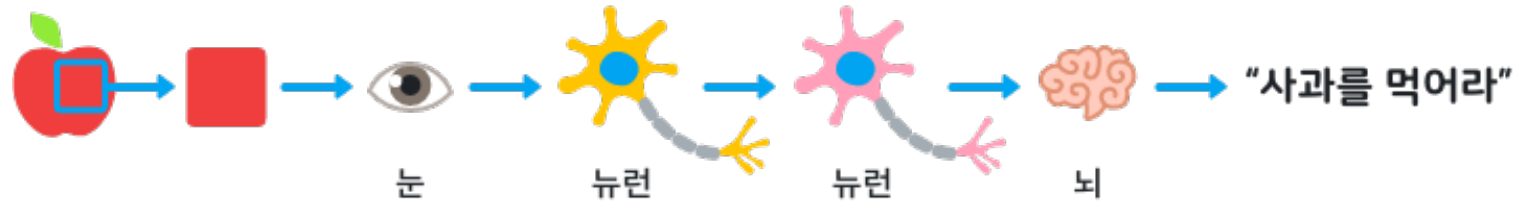
인공신경망

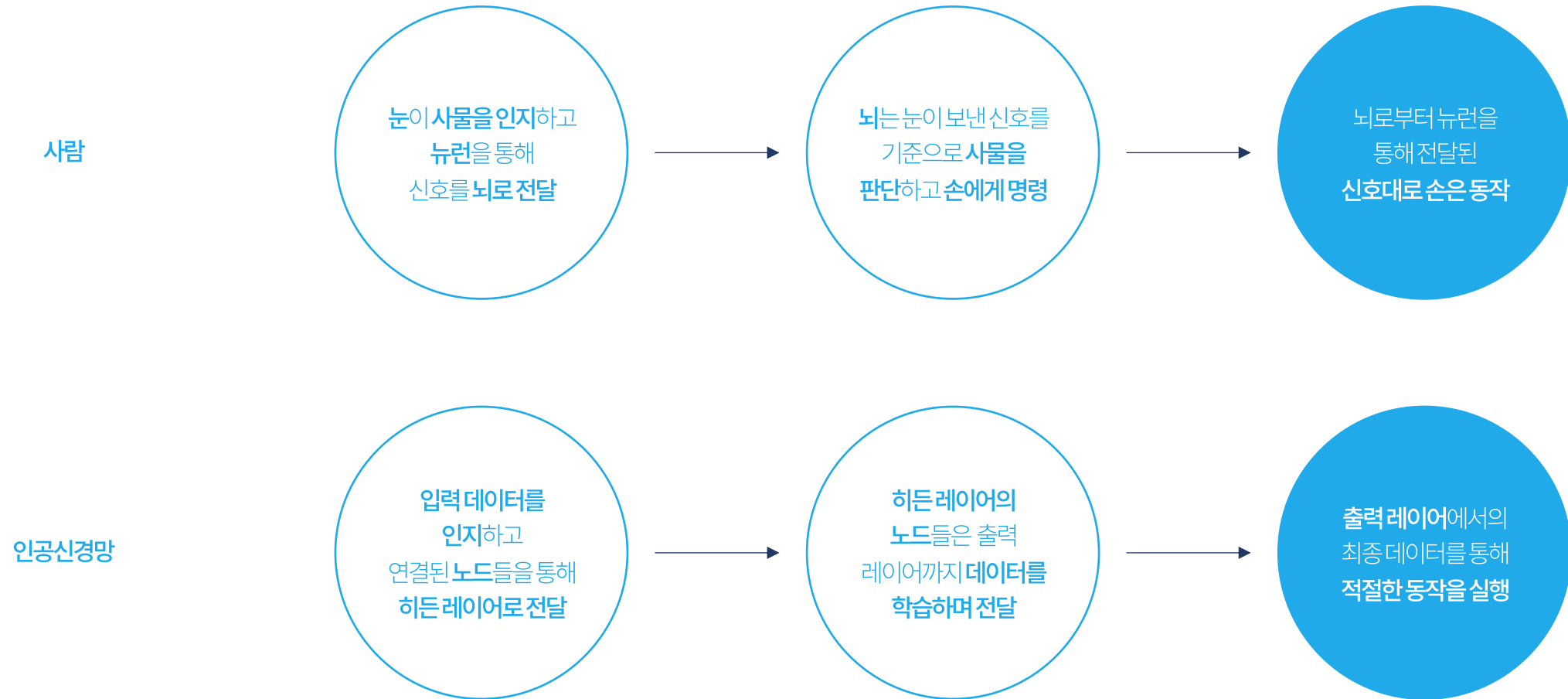
사람을 본따 만든 컴퓨터만의 신경망

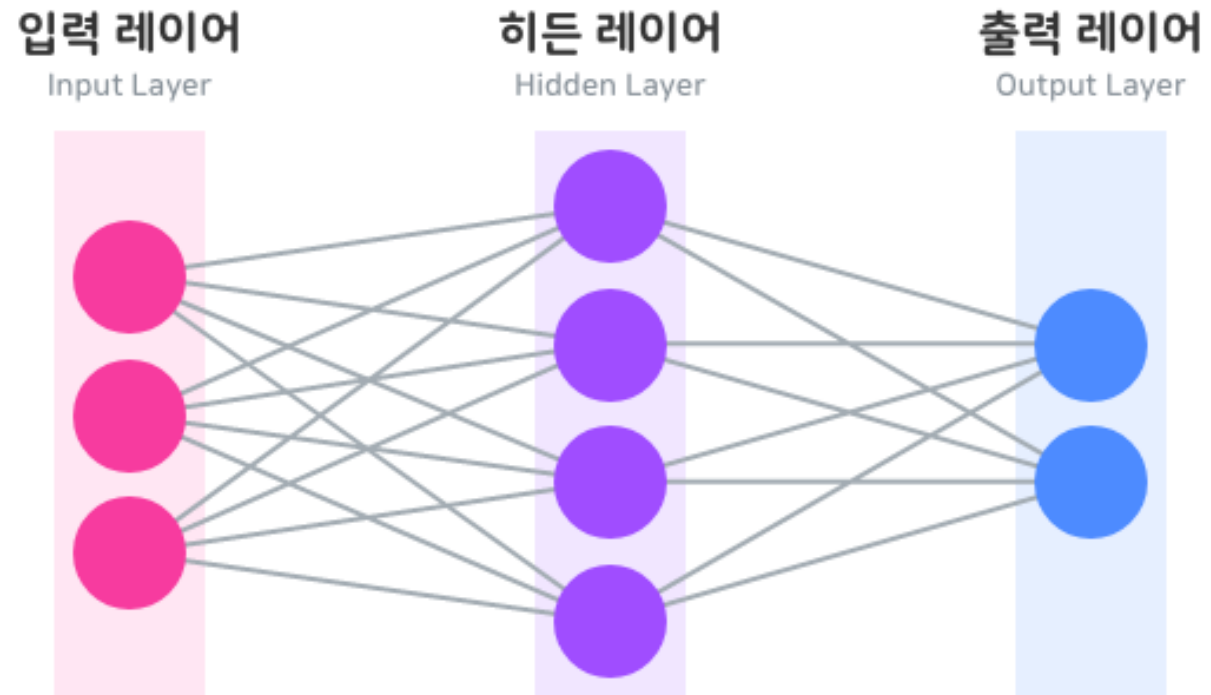
[인간의 뇌는 어떻게 동작할까요?]



인공신경망은 인간의 뇌가 동작하는 방식에
영감을 얻어 만들어진 모델입니다.

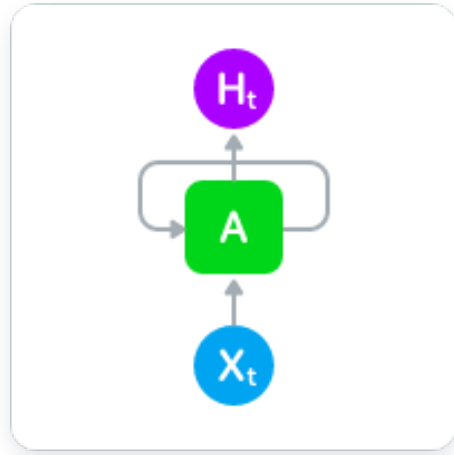




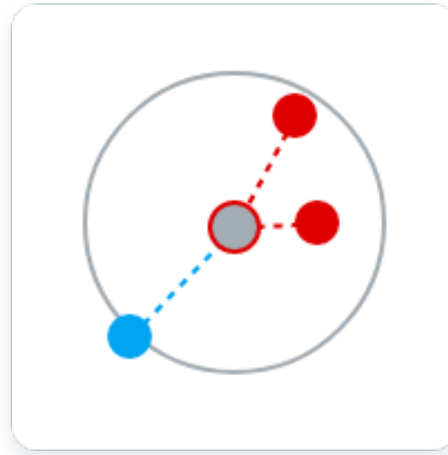


다음 중 인공신경망을 응용한
딥러닝 모델이 아닌 것은?

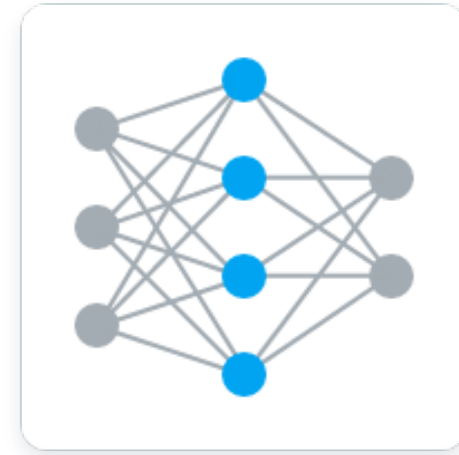
순환 신경망



kNN

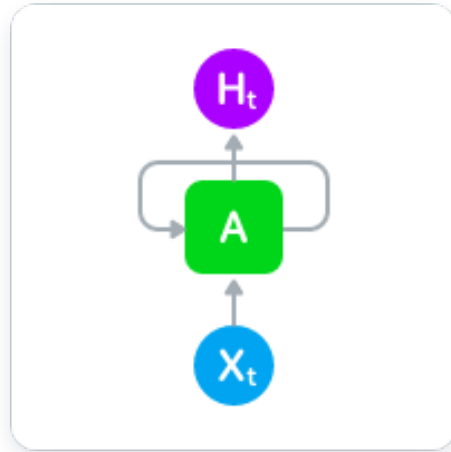


완전 연결 신경망

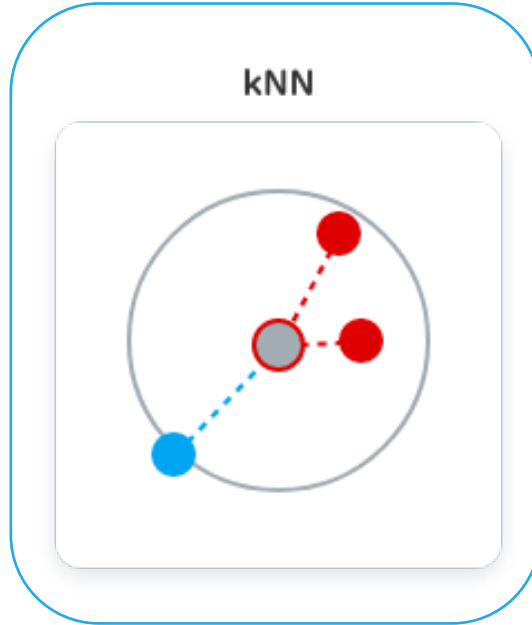


다음 중 인공신경망을 응용한
딥러닝 모델이 아닌 것은?

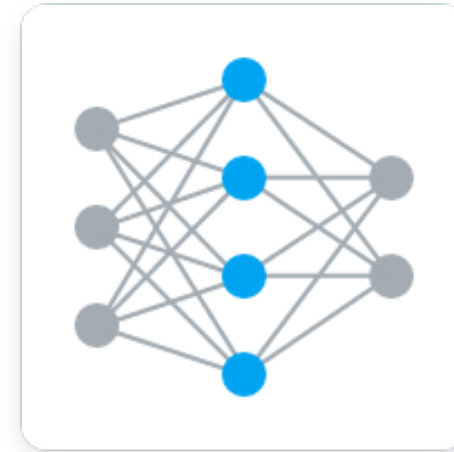
순환 신경망



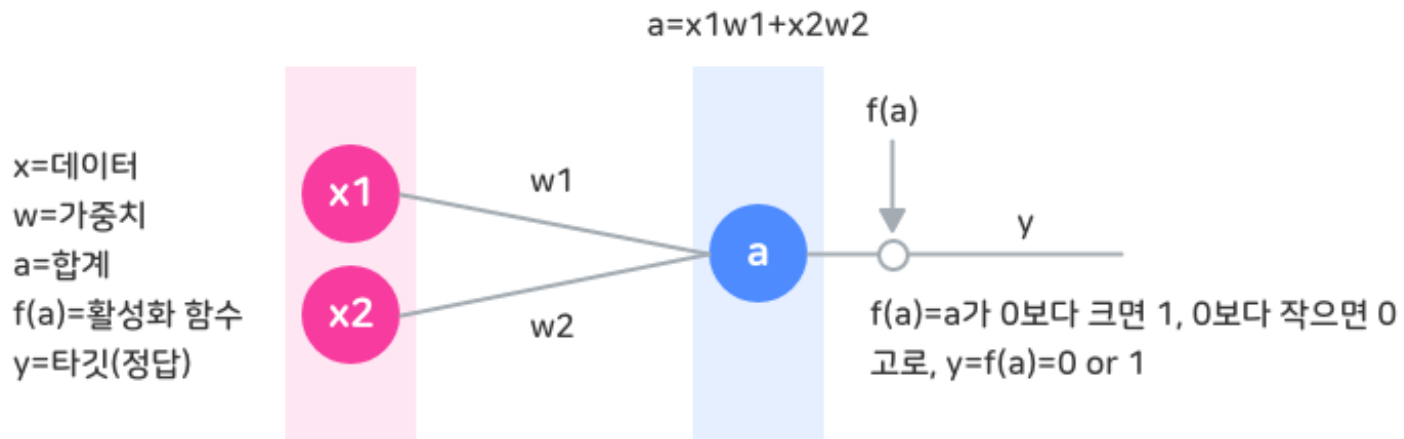
kNN



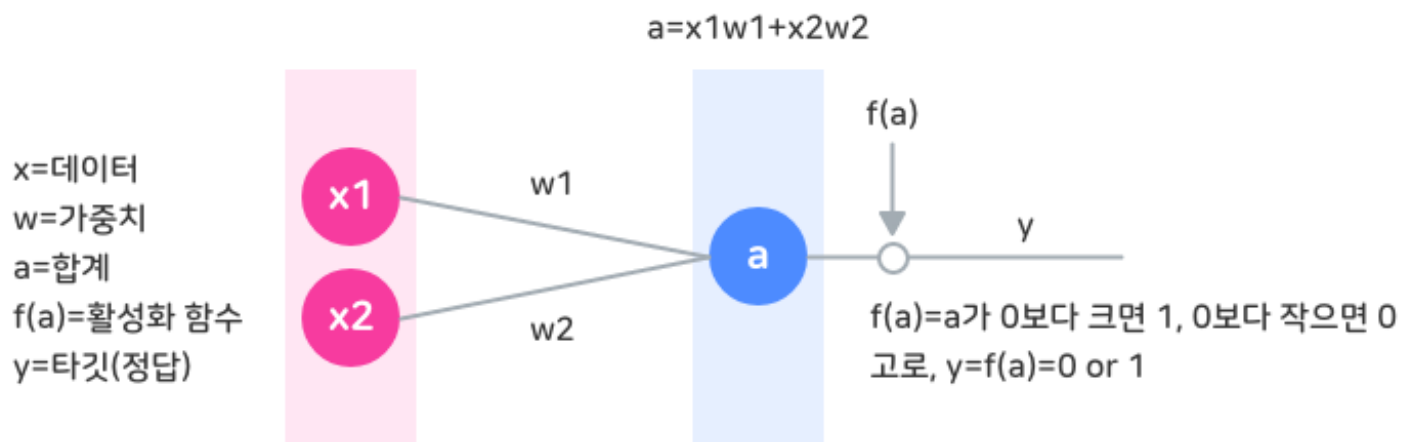
완전 연결 신경망

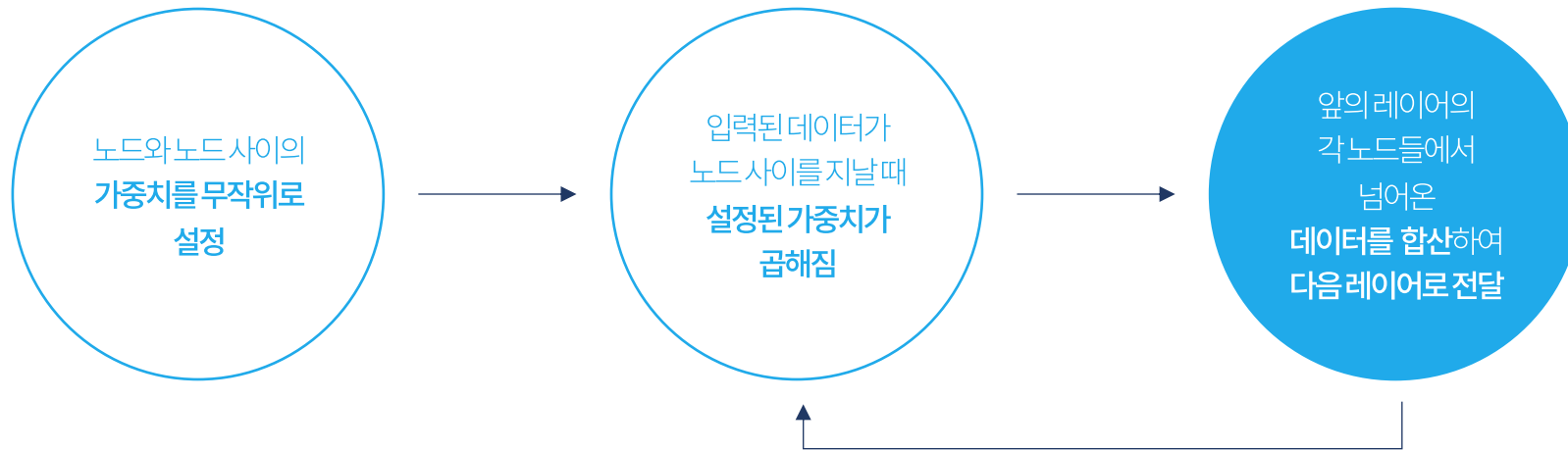


컴퓨터는 이 데이터가 올바른 데이터인지 판단하기 위해서
데이터에 여러가지 값을 더하고 곱해보며 정답을 찾아 학습해 나갑니다.



컴퓨터는 이 데이터가 올바른 데이터인지 판단하기 위해서
데이터에 여러가지 값을 더하고 곱해보며 정답을 찾아 학습해 나갑니다.
가중치(Weight)



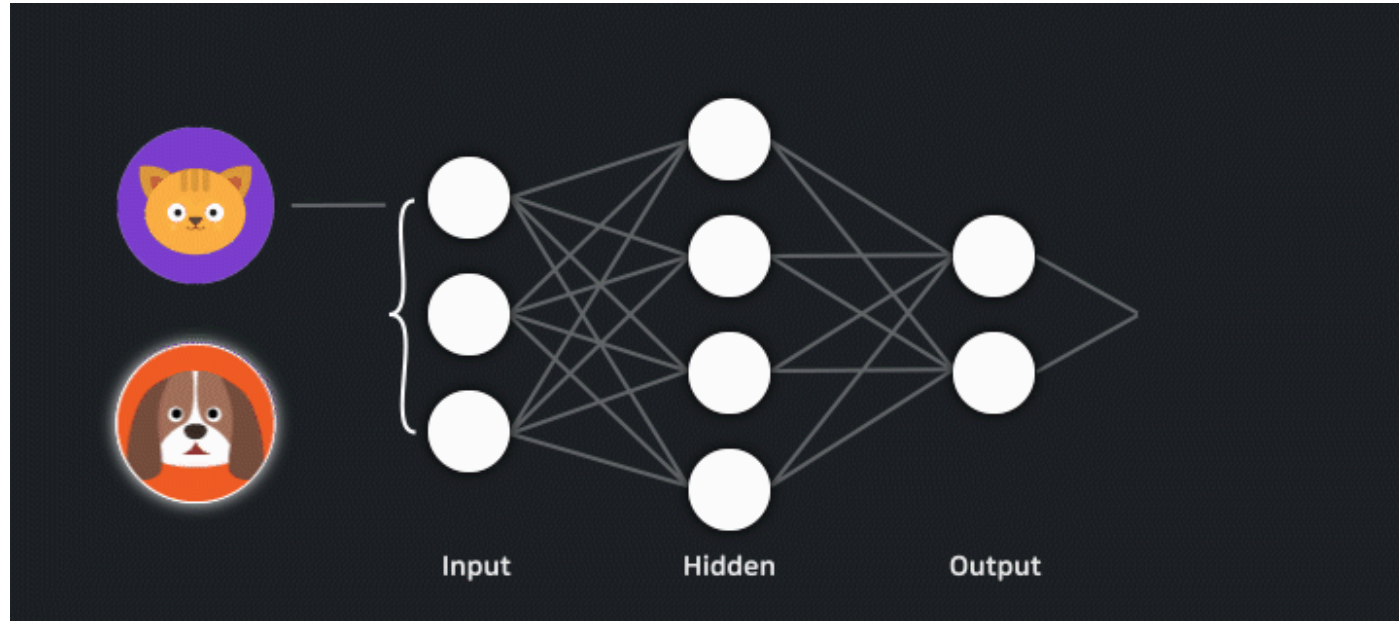


활성화 함수(Activation Function)

컴퓨터가 결과를 쉽게 판단할 수 있도록 데이터를 변환시켜주는 함수

(예) 계단 함수(Step Function): 'a가 0보다 크면 1, 0보다 작으면 0'

컴퓨터는 데이터를 학습하는 과정에서 **스스로 가중치를 수정**하고,
중요성이 떨어진다고 판단되는 노드들과의 연결은 해제하며
최적의 값을 찾아나갑니다.



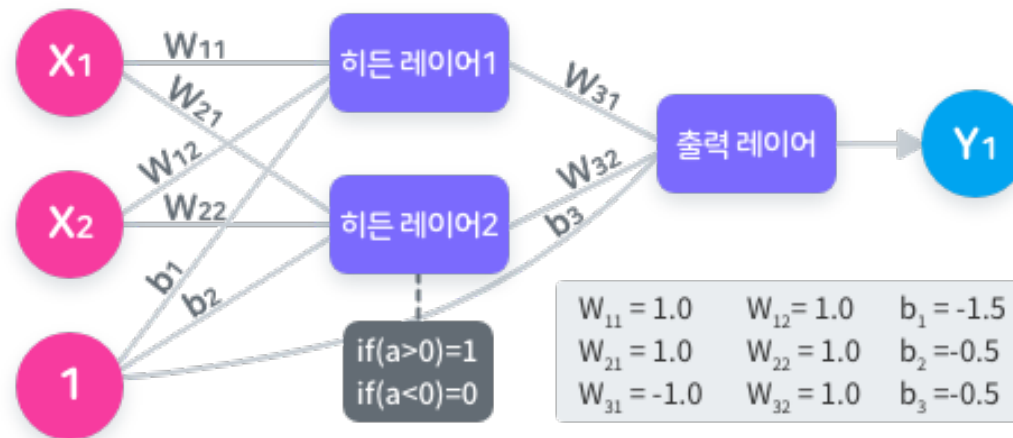
편향(Bias)

출력 데이터가 자꾸만 **0**을 출력하는 것을 막기 위해 입력 데이터를 임의로 조정하는 역할
입력 데이터와 가중치를 곱한 다른 데이터의 **결맞음**들과 편향을 합산하여 진행

인공신경망을 계산해보자!

아래 주어진 입력 데이터 X_1 과 X_2 와 가중치(W)를 계산해 출력 데이터 Y 를 알아내 봅시다!

- W_{1x}, W_{2x} 의 값은 0과 가까운 임의의 숫자가 주어진다.
- 히든 레이어로 향하면서 입력값(x)과 가중치(W)의 연산이 이루어진다. (히든 레이어1= $X_1W_{11}+X_2W_{12}+b_1$)
- 히든 레이어에서 활성화 함수를 통해 a 가 0보다 작으면 0, 0보다 크면 1을 반환한다

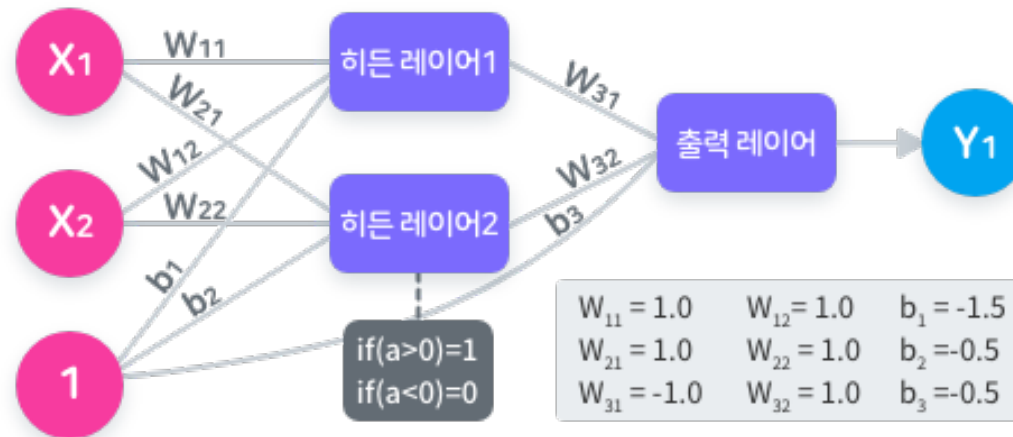


X1	X2	Y1
0	0	
0	1	
1	0	
1	1	

인공신경망을 계산해보자!

아래 주어진 입력 데이터 X_1 과 X_2 와 가중치(W)를 계산해 출력 데이터 Y 를 알아내 봅시다!

- W_{1x}, W_{2x} 의 값은 0과 가까운 임의의 숫자가 주어진다.
- 히든 레이어로 향하면서 입력값(x)과 가중치(W)의 연산이 이루어진다. (히든 레이어1= $X_1W_{11}+X_2W_{12}+b_1$)
- 히든 레이어에서 활성화 함수를 통해 a 가 0보다 작으면 0, 0보다 크면 1을 반환한다



X_1	X_2	Y_1
0	0	0
0	1	1
1	0	1
1	1	0



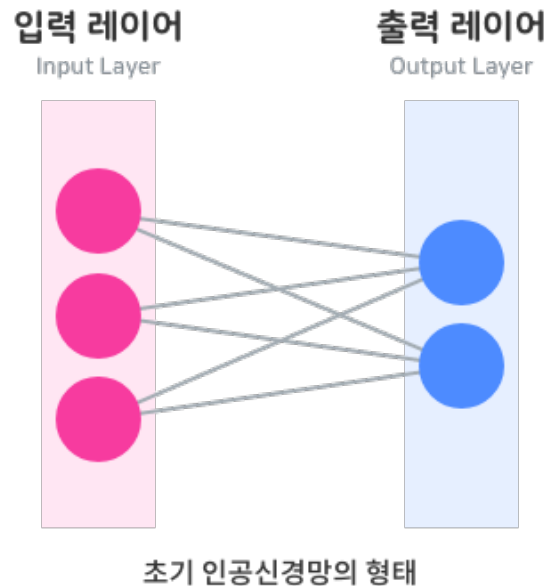
손실 함수(Loss Function)

출력 데이터와 정답과의 차이를 계산하는 함수

손실 함수가 계산된 결괏값을 손실 점수라고 하며, 손실 점수를 최소화하는 것이 인공신경망의 목표

초기 신경망에는 히든 레이어가 없이

각각의 입력 데이터에 가중치를 곱해 합산한 데이터만으로 결과를 판단하는 형태였습니다.



AND 문제			OR 문제			NAND 문제			XOR 문제		
에드	리마	출력	에드	리마	출력	에드	리마	출력	에드	리마	출력
0	0	0	0	0	0	0	0	1	0	0	0
0	1	0	0	1	1	0	1	1	0	1	1
1	0	0	1	0	1	1	0	1	1	0	1
1	1	1	1	1	1	1	1	0	1	1	0

에드					에드				에드					에드			
○		○	●		○		●	●	○		●	○		○		●	○
×		○	○		×		○	●	×		○	○		×		○	●
	×					×				×					×		
							×									×	




두 사람 모두 사탕을 받은 경우 ●				둘 중 한 사람이라도 사탕을 받은 경우 ●				두 사람 모두 사탕을 받은 경우만을 제외하고 ●				둘 중 한 사람만 사탕을 받은 경우 ●			
---------------------	--	--	--	-------------------------	--	--	--	----------------------------	--	--	--	-----------------------	--	--	--

초기 신경망의 형태로는 **XOR 문제**와 같은
 생각보다 단순한 문제를 해결하지 못했지만,
 히든 레이어의 등장으로 레이어에서 **데이터의 수정이 가능**해지면서
 이는 인공신경망의 또 다른 발전으로 이어졌습니다.

회로문제를 풀어봅시다!

회로문제를 풀어봅시다. 입력 데이터인 A와 B는 각각 값이 존재한다면 1을, 존재하지 않으면 0을 표시할 것입니다.

입력되는 두 데이터가 회로의 조건을 충족한다면 1을, 충족하지 않으면 0을 우측 표에 기입해 봅시다.




게이트	AND	OR	NAND																																													
논리기호																																																
진리표	<table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	F	0	0	0	0	1	0	1	0	0	1	1	1	<table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	F	0	0	0	0	1	1	1	0	1	1	1	1	<table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	F	0	0	1	1	0	1	0	1	1	1	1	0
A	B	F																																														
0	0	0																																														
0	1	0																																														
1	0	0																																														
1	1	1																																														
A	B	F																																														
0	0	0																																														
0	1	1																																														
1	0	1																																														
1	1	1																																														
A	B	F																																														
0	0	1																																														
1	0	1																																														
0	1	1																																														
1	1	0																																														

입력		출력		
A	B	F		
		AND	OR	NAND
0	0			
1	0			
0	1			
1	1			

회로 문제를 풀어봅시다!

회로 문제를 풀어봅시다. 입력 데이터인 A와 B는 각각 값이 존재한다면 1을, 존재하지 않으면 0을 표시할 것입니다.

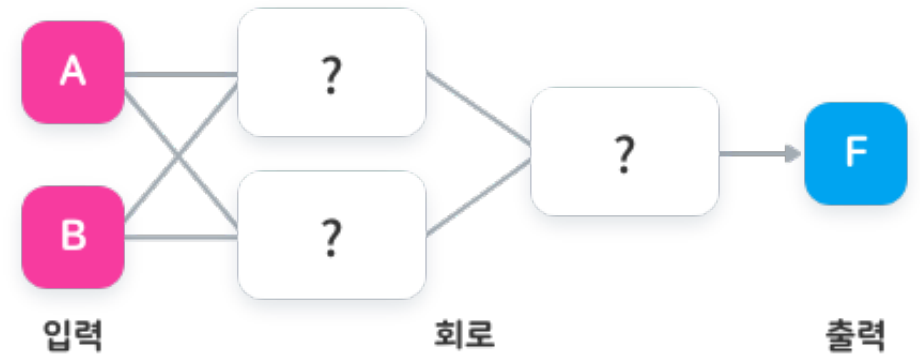
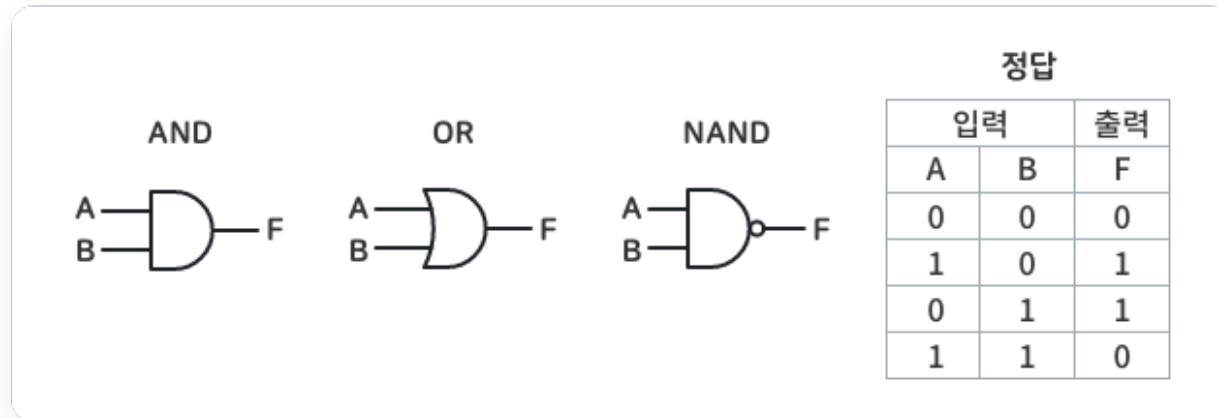
입력되는 두 데이터가 회로의 조건을 충족한다면 1을, 충족하지 않으면 0을 우측 표에 기입해 봅시다.

게이트	AND	OR	NAND																																													
논리기호																																																
진리표	<table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	F	0	0	0	0	1	0	1	0	0	1	1	1	<table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	F	0	0	0	0	1	1	1	0	1	1	1	1	<table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	F	0	0	1	0	1	1	1	0	1	1	1	0
A	B	F																																														
0	0	0																																														
0	1	0																																														
1	0	0																																														
1	1	1																																														
A	B	F																																														
0	0	0																																														
0	1	1																																														
1	0	1																																														
1	1	1																																														
A	B	F																																														
0	0	1																																														
0	1	1																																														
1	0	1																																														
1	1	0																																														

입력		출력		
A	B	F		
		AND	OR	NAND
0	0	0	0	1
1	0	0	1	1
0	1	0	1	1
1	1	1	1	0

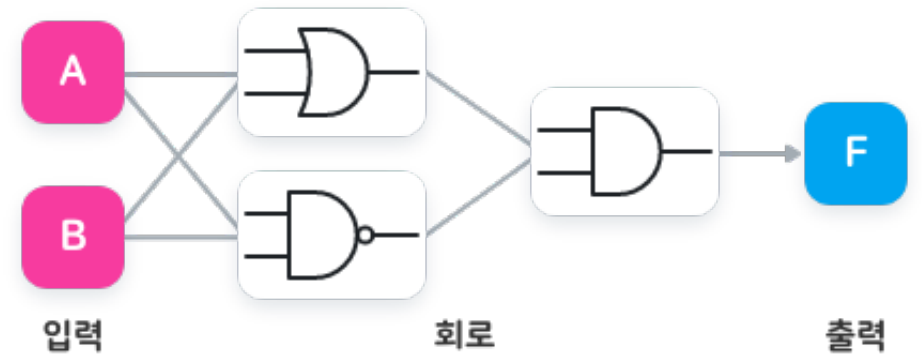
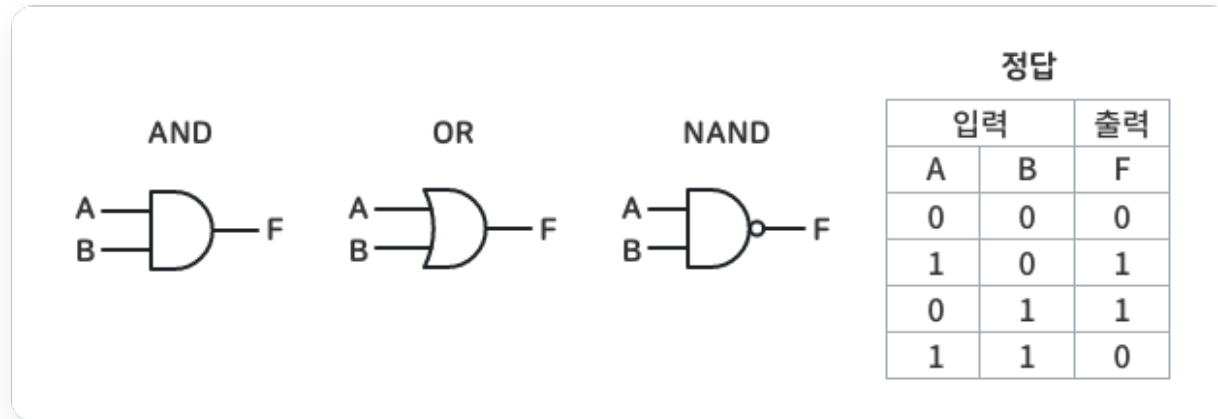
회로를 좀 더 복잡하게 만들어봅시다!

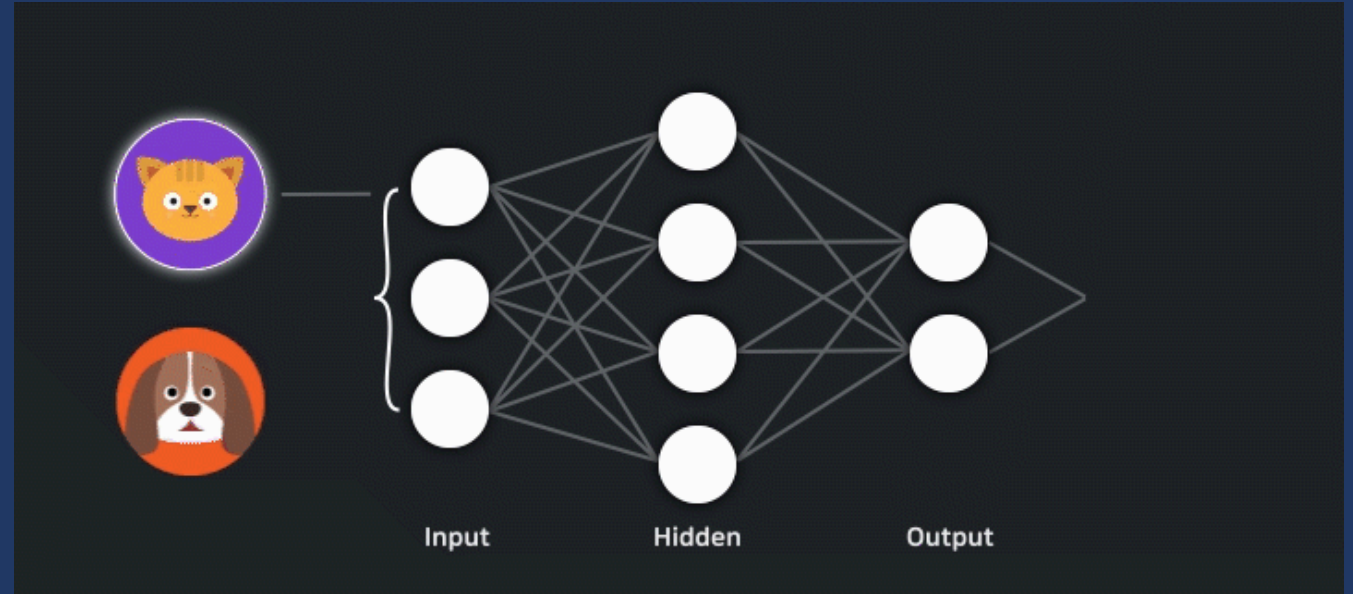
회로를 조합해서 정답과 같은 결과를 출력하는 회로를 만들어봅시다.



회로를 좀 더 복잡하게 만들어봅시다!

회로를 조합해서 정답과 같은 결과를 출력하는 회로를 만들어봅시다.

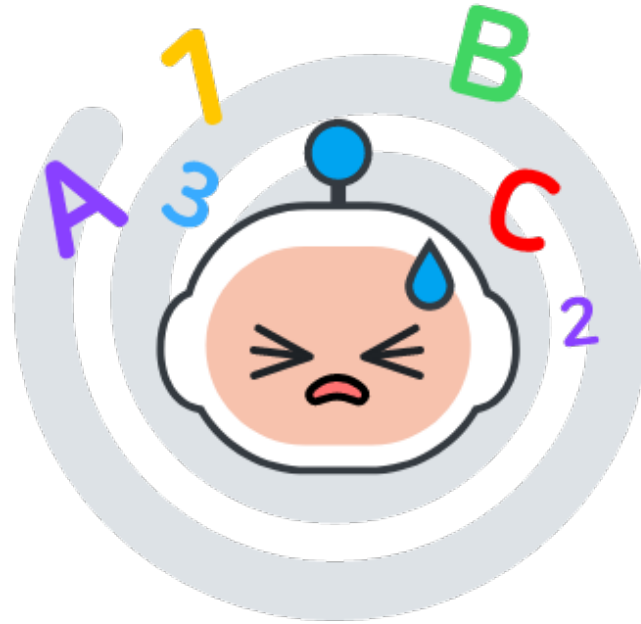




역전파(Backpropagation)

레이어의 수가 많을수록 어떤 가중치가 부적절한 결과를 초래했는지 찾기 어렵기 때문에
입력 레이어에서 출력 레이어의 방향으로 거처온 경로를 **다시 역으로 거슬러 올라가면서**
최적의 값이 나올 수 있도록 **가중치를 수정**해 나가는 방법

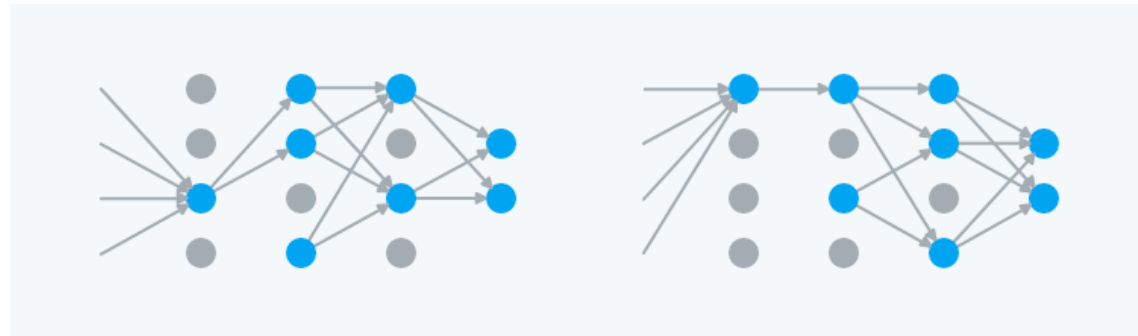
학습 데이터를 무리하여 **일정 수준 이상으로 학습**시킬 경우에는 새로운 데이터에 대해 오차가 증가하는 **과적합이 발생**할 수 있습니다.



Dropout

전체 신경망에서 일정 비율의 노드를 임의로 잘라내 학습하는 방법

신경망을 분할하여 각 부분마다 먼저 학습을 마친 후, 이 결과 데이터로 다시 전체 신경망을 학습



초매개변수 조정

에폭(Epoch, 학습 횟수), 레이어의 수와 같은 초매개변수를 조정하는 방법

수고하셨습니다.