

딥러닝을 이용한 효과적인 학습



학습데이터와 경사하강법

- 영상처럼 학습 데이터 차원이 크고, 은닉층 계층 구조가 많은 경우 효율적 학습 방법 필요
 - 전체 데이터가 아닌 부분 데이터를 사용 (가중치 업데이트 과정에서 전체 데이터를 사용할 필요는 없음)

$$w = w - \alpha \frac{\partial}{\partial w} cost(w)$$

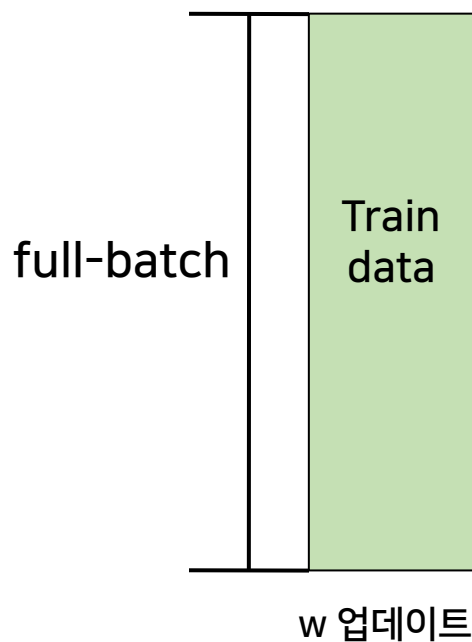
어느 정도의 학습 데이터를 사용할까?

$$W = w - \alpha \frac{1}{m} \sum_{k=1}^m (d_k - y_k) x_k$$

미니배치

- 풀 배치(Full-batch) : 전체 학습 데이터를 사용해 가중치 업데이트
- 확률적 경사 하강법 : 학습 데이터를 분할해서 가중치 업데이트

경사하강법
(Gradient Descent)



확률적 경사 하강법
(Stochastic Gradient Descent)



미니배치

● 경사하강법을 이용할 기울기 계산

7개 데이터에 대한 기울기 : 7 5 -4 -3 1 5 -8

전체에 대해 계산할 경우 :

7	5	-4	-3	1	5	-8
---	---	----	----	---	---	----

3/7만큼 이동

두 부분으로 나누어 계산할 경우 :

7	5	-4	-3	1	5	-8
---	---	----	----	---	---	----

8/3만큼 이동 -5/4만큼 이동

하나씩 계산할 경우 :

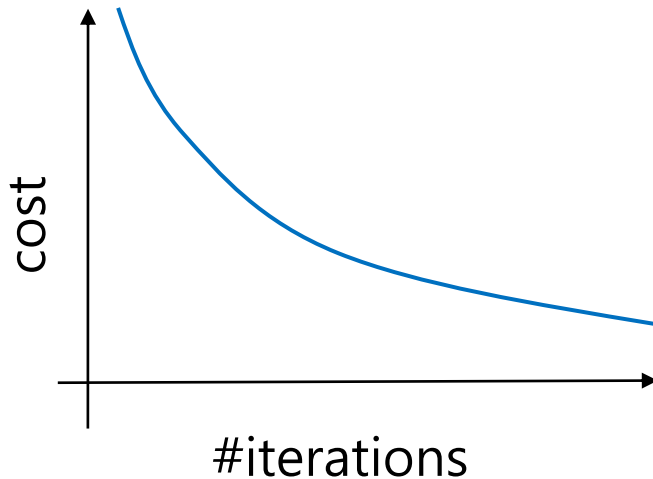
7	5	-4	-3	1	5	-8
---	---	----	----	---	---	----

7만큼 이동

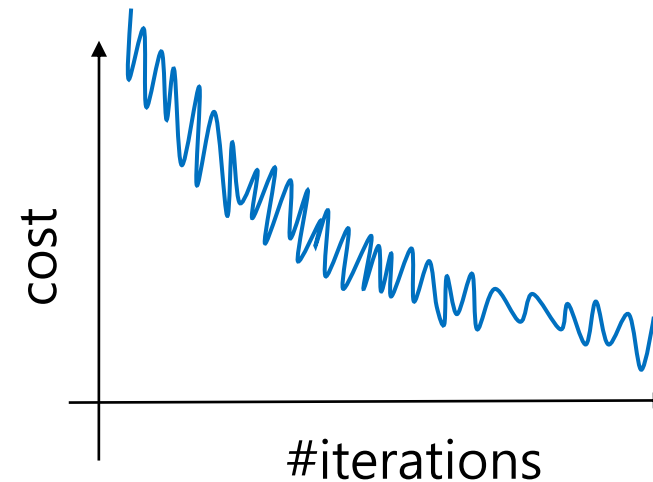
미니배치를 이용한 경사하강법

- 많은 데이터를 가진 신경망을 학습할 때 확률적 경사하강법 사용
 - 한 번 학습에 수행되는 시간을 줄이고, 전체 데이터에 대해서 반복적으로 학습 수행
 - 전체 데이터를 사용하는 경사하강법보다 수렴 속도가 빠름

풀 배치 경사하강법

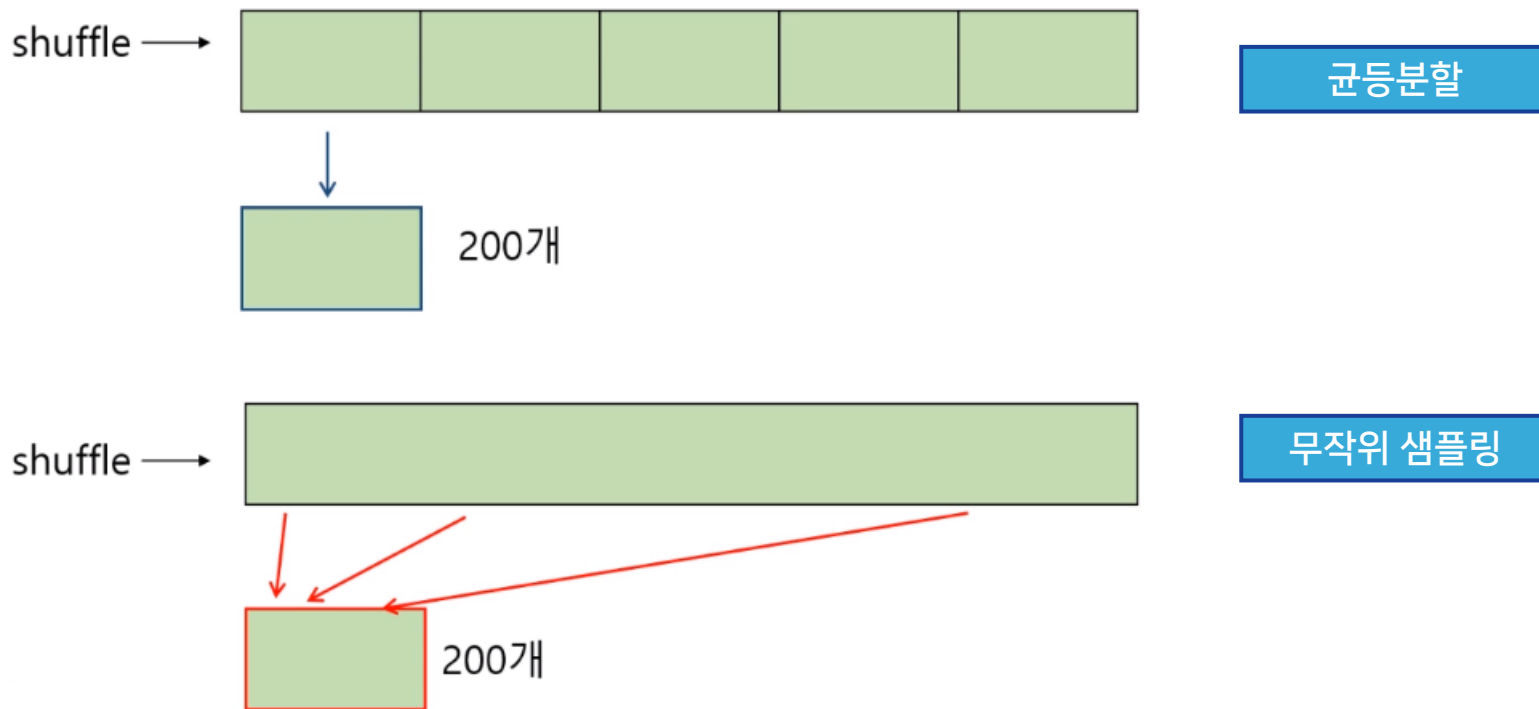


미니배치 경사 하강법



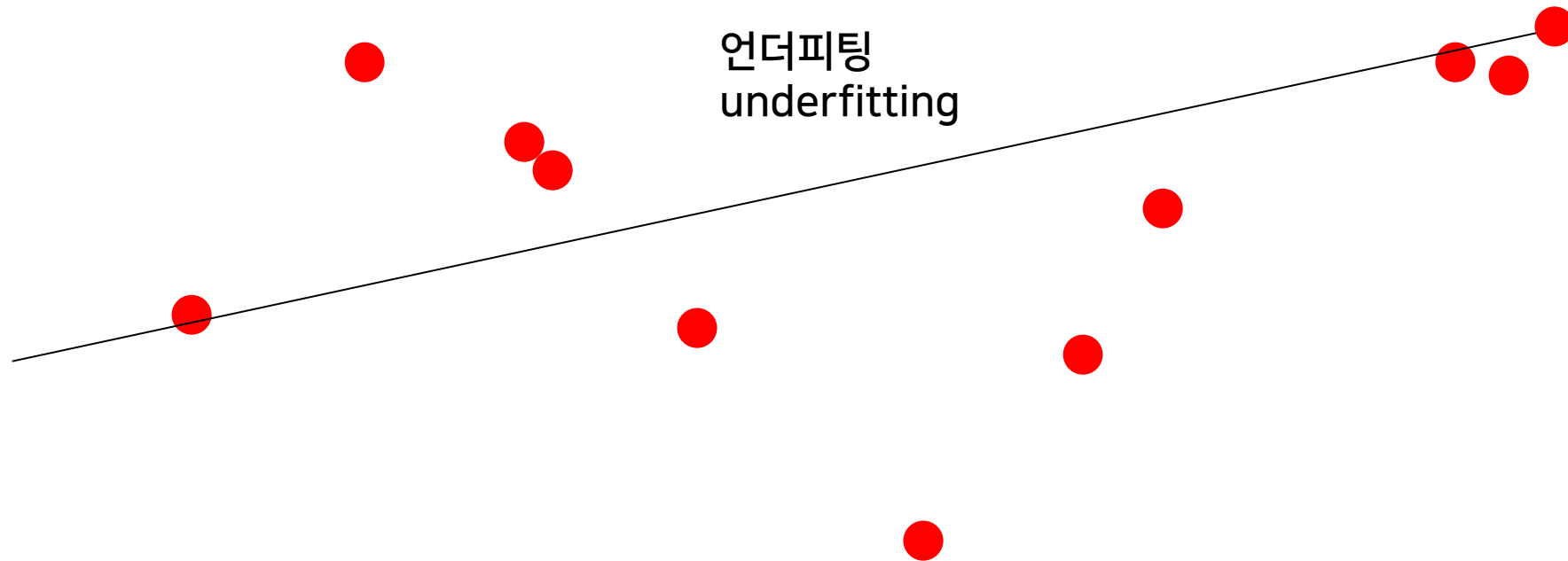
미니배치

- 전체 데이터를 n 개의 데이터로 나누어 중복되지 않게 구성하거나 무작위로 샘플링할 수 있음



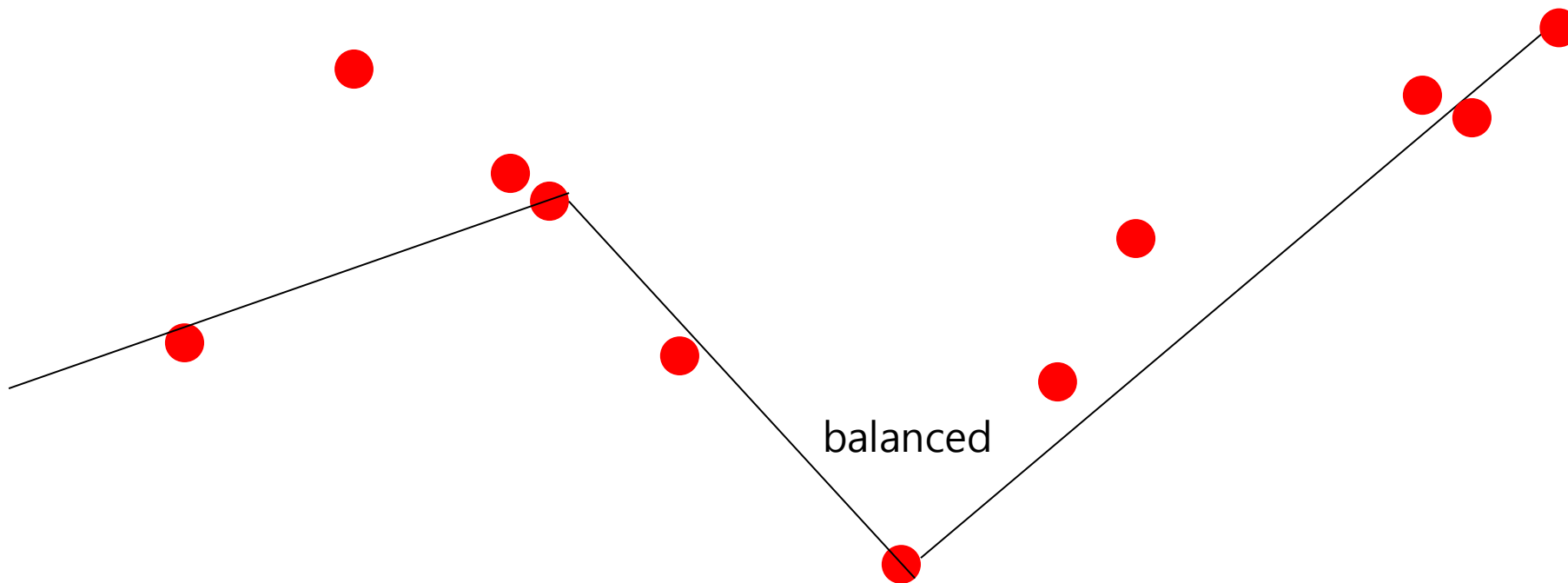
과적합(Overfitting)

- 언더피팅(underfitting) : 전체 데이터를 잘 설명하지 못하게 되어 학습이 덜 이루어짐



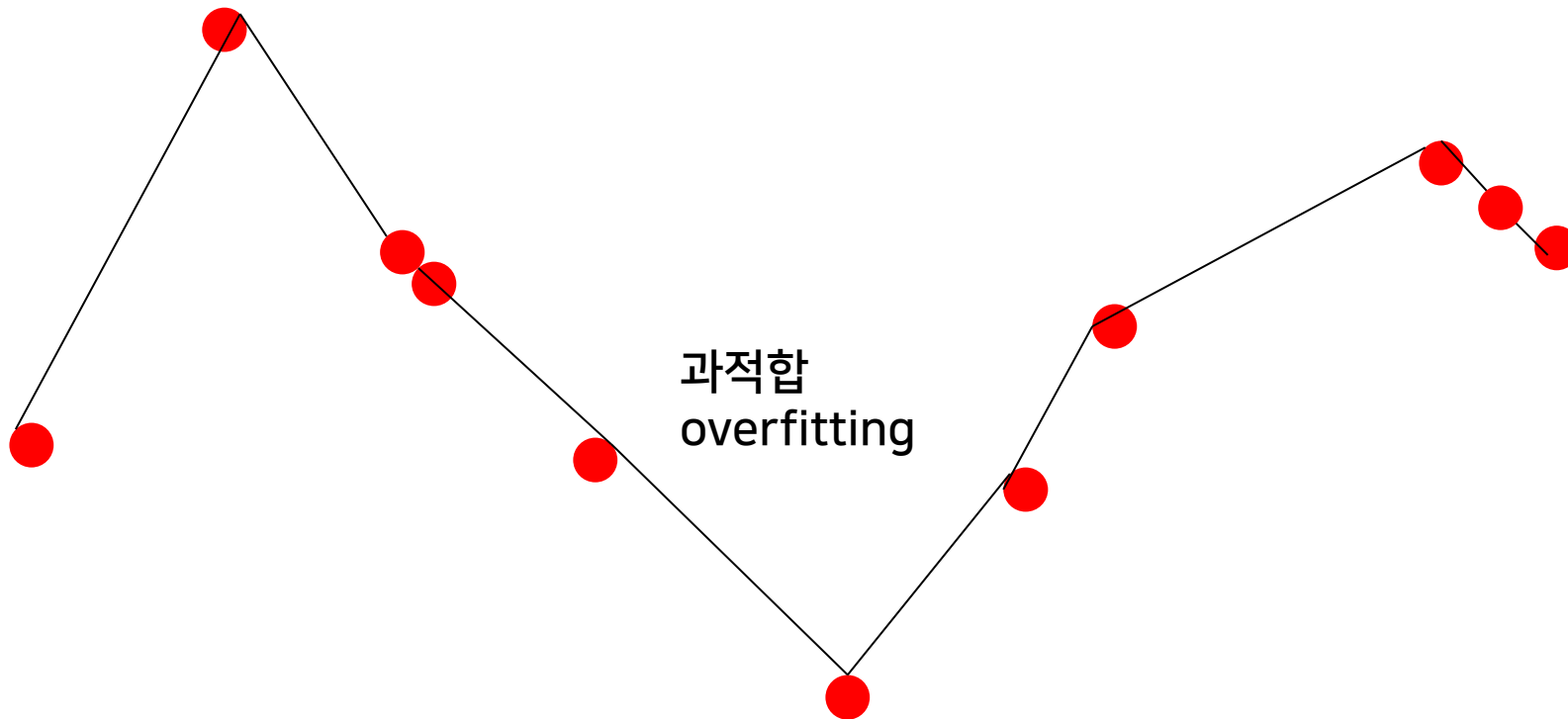
과적합(Overfitting)

- 전체적으로 데이터를 잘 표현, 적절하게 학습된 상황



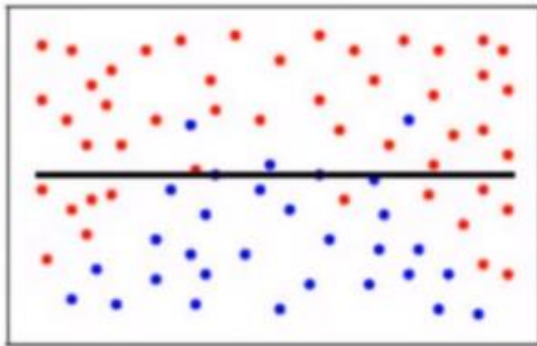
과적합(Overfitting)

- 과적합(Overfitting) : 학습 데이터에 대한 인식률은 높는데, 테스트 데이터에 대한 인식률이 낮을 경우
 - 학습 데이터와 다르거나 멀리 떨어진 데이터를 사용할 경우 성능이 급격히 떨어짐(일반화 성능이 좋지 않음)

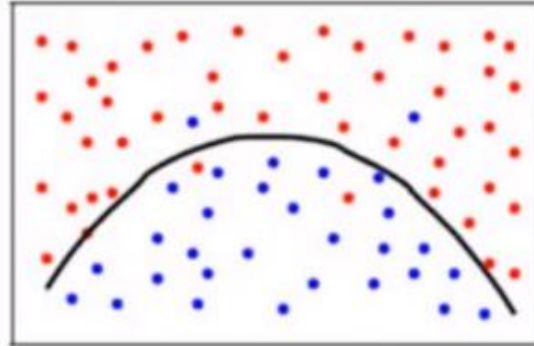


분류 문제에서 과적합(Overfitting)

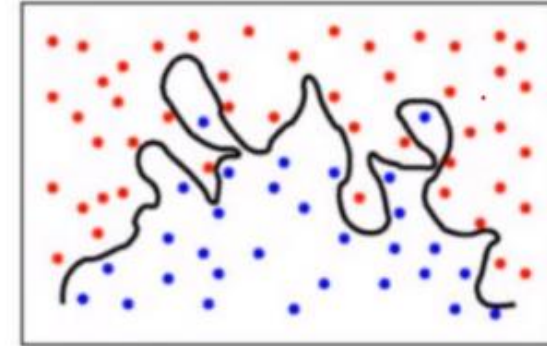
- 학습 후 성능의 문제가 과적합으로부터 발생한 것인지 항상 검증해야 함



언더피팅
(지나치게 단순한 상태)



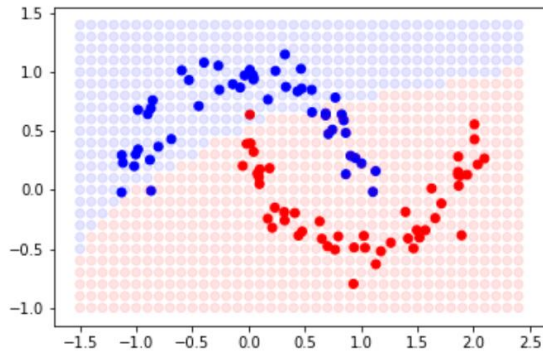
tomrobertshaw.net



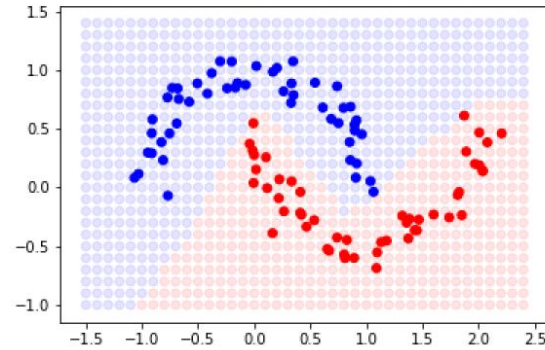
오버피팅
(일반화 성능이 떨어짐)

다층신경망에서 과밀학습

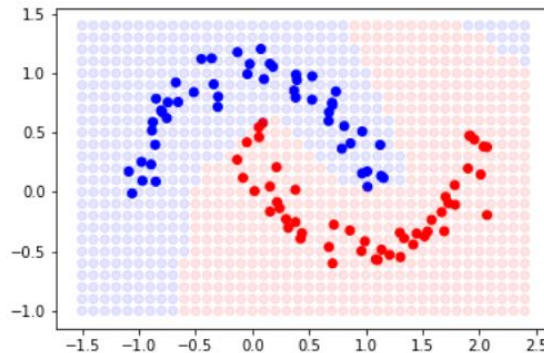
- 은닉층이 깊어지면 조금 더 복잡한 형태의 결정경계면 가짐
 - 많은 수의 은닉층을 사용할 경우 과적합 발생 가능성에 대해 대응



은닉층 개수 : 1개



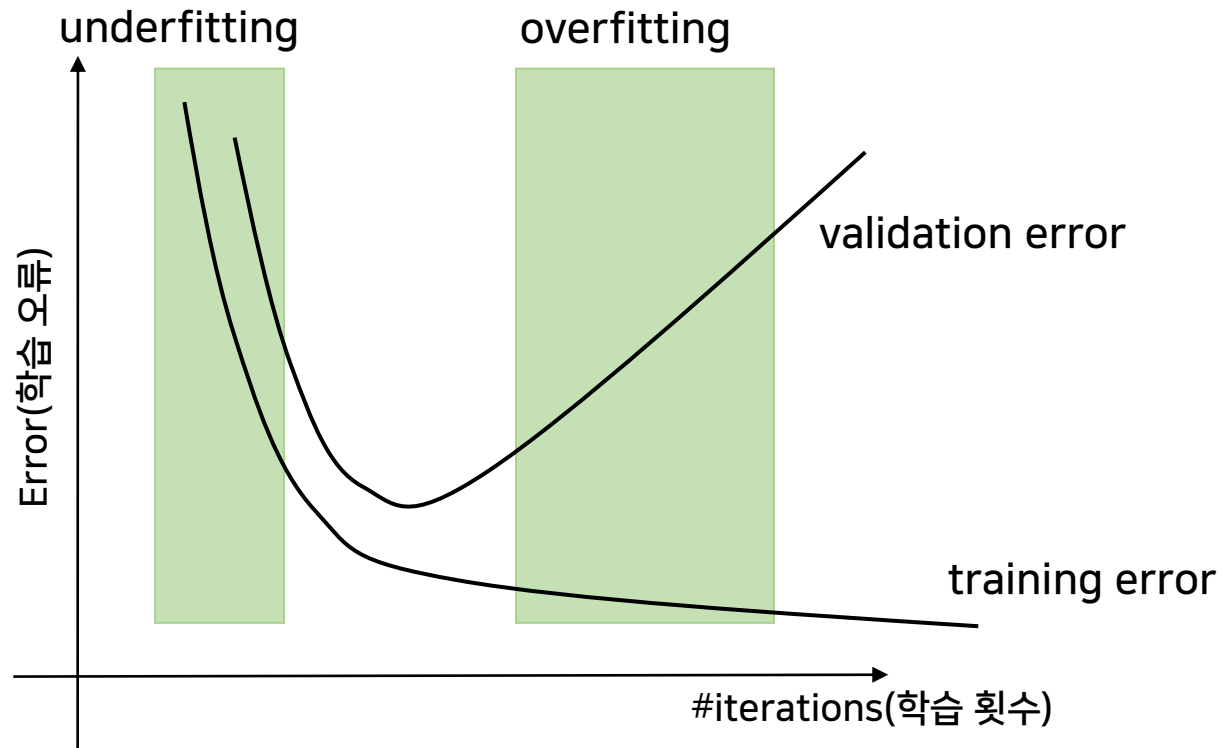
은닉층 개수 : 3개



은닉층 개수 : 5개

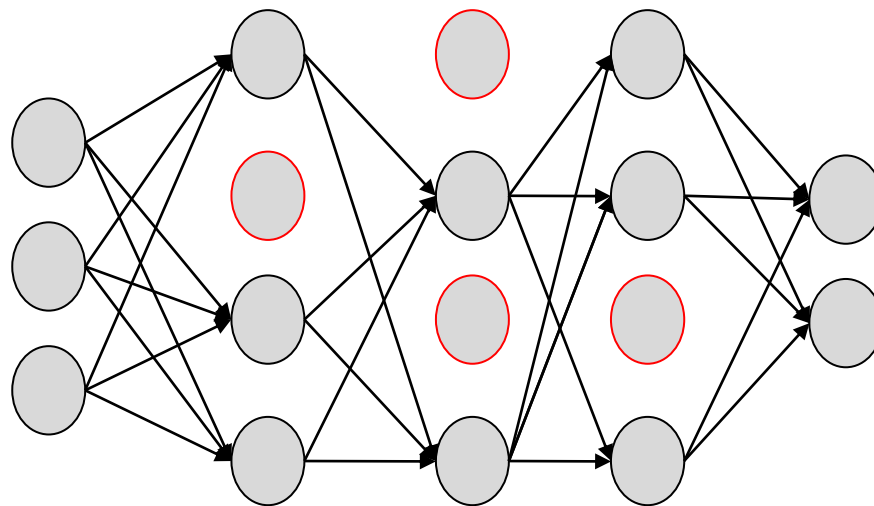
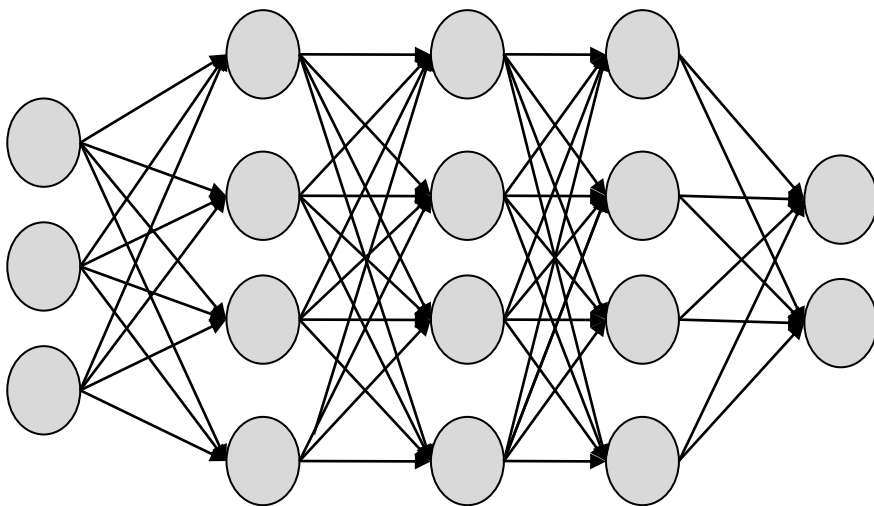
과적합 검출

- 데이터 셋을 학습 셋, 검증 셋, 테스트 셋으로 구성
- 학습 오류는 감소하나 검증 셋 오류가 증가하는 경우 과적합 상태이며, 학습 오류와 검증 오류가 모두 감소하는 경우 학습이 잘 수행되는 것으로 볼 수 있음



신경망 드롭아웃(Dropout)

- 드롭아웃(Dropout) : 구조 변경없이 가중치를 0으로 만들어서 과적합이 발생하지 않도록 하는 전
- 학습단계에서 랜덤으로 임의의 뉴런을 선택, 삭제해 신호를 더 이상 다음 계층으로 전달하지 않음



드롭아웃 효과

- 다소 부정확한 다수의 약한 분류기들을 Voting 방식으로 결합함으로써 과적합 회피

약한 분류기(weak classifier)

약한 분류기(weak classifier) → 강한 분류기(strong classifier)

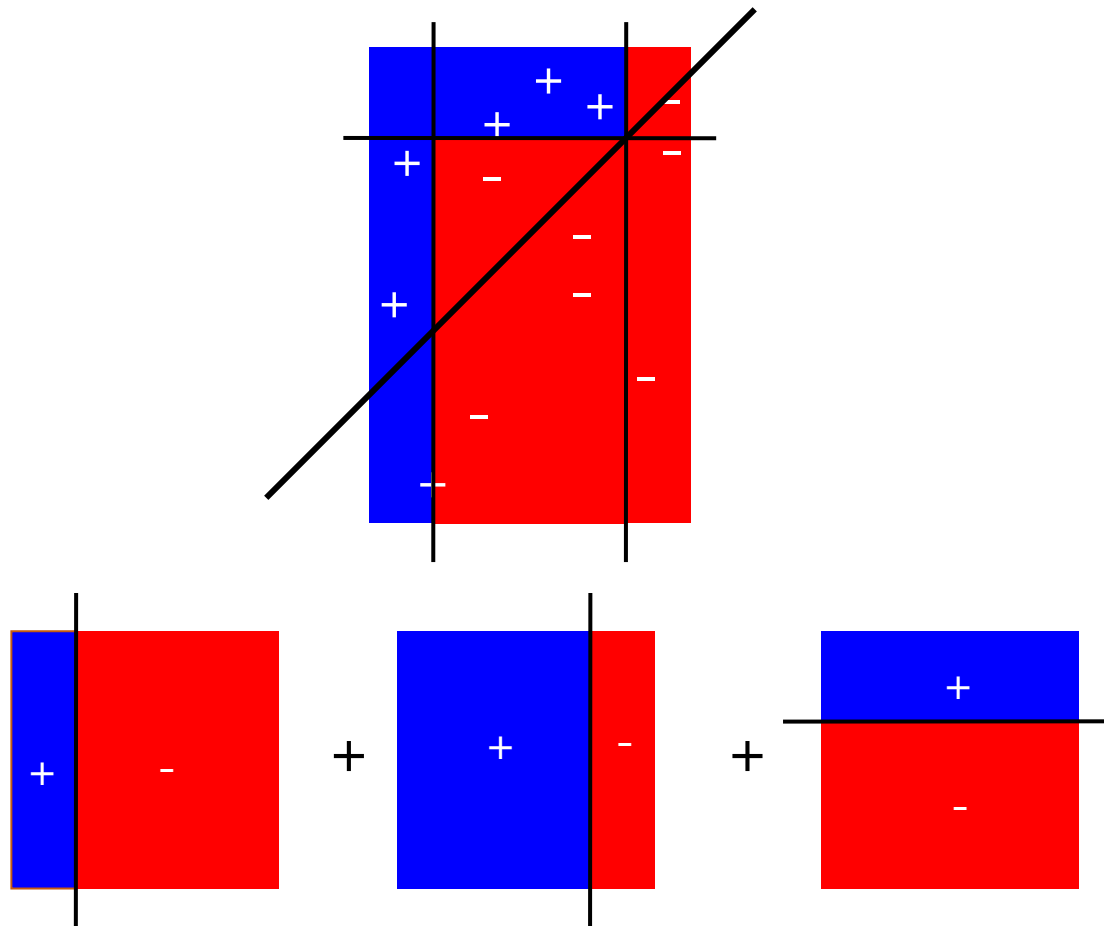
약한 분류기(weak classifier)

- 가중치 값이 현저하게 큰 특정 가중치 값이 드롭아웃될 확률이 높아져 전체적으로 서로 적응해 학습에 참여(co-adaptation)

[0.72 0.01 -0.02] → [0 0.01 -0.02]

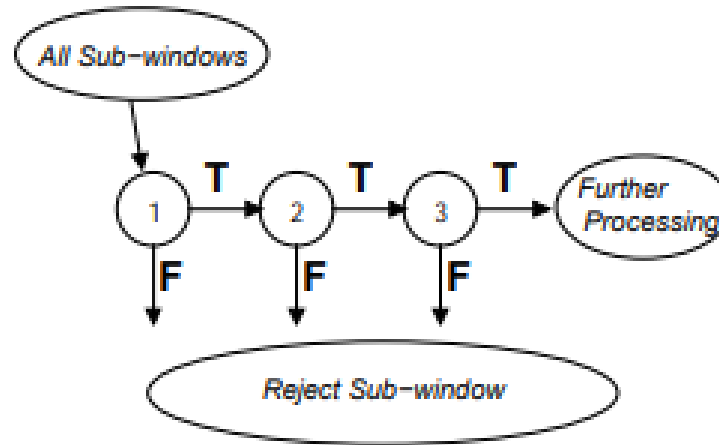
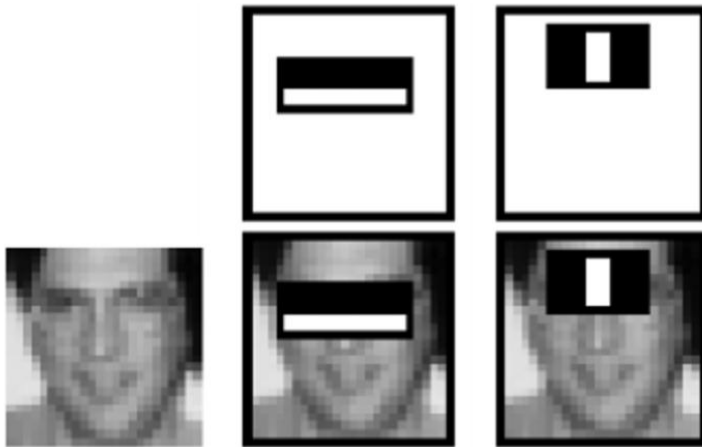
앙상블 방법을 사용한 분류기 성능 개선 사례 #1

- 약한 분류기들의 조합으로 강한 분류기를 만들어 성능 개선



앙상블 방법을 사용한 분류기 성능 개선 사례 #2

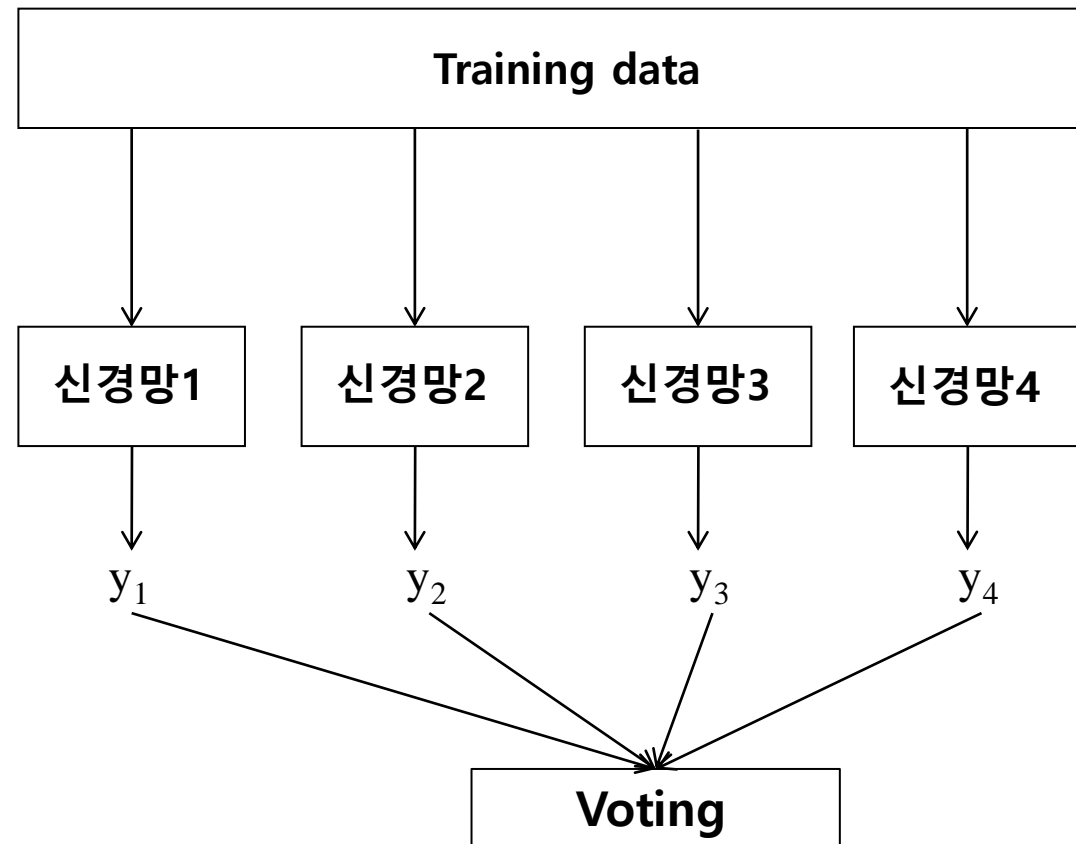
- 명암이라는 간단한 특징으로 얼굴인지 아닌지를 검출
 - 간단한 분류기를 연속적으로 수천 개 조합해 얼굴 검출 알고리즘으로 사용



Rapid Object Detection Using a Boosted Cascade of Simple Features

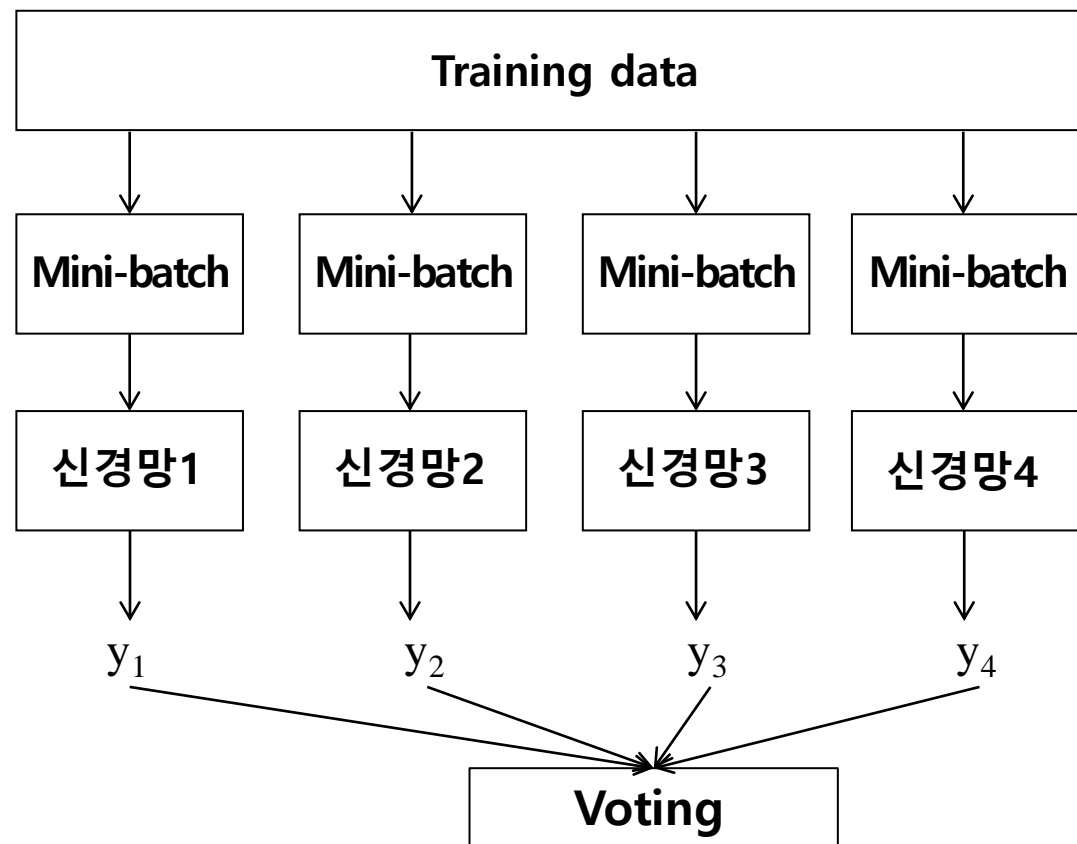
신경망 앙상블 학습

● 앙상블 알고리즘 구조



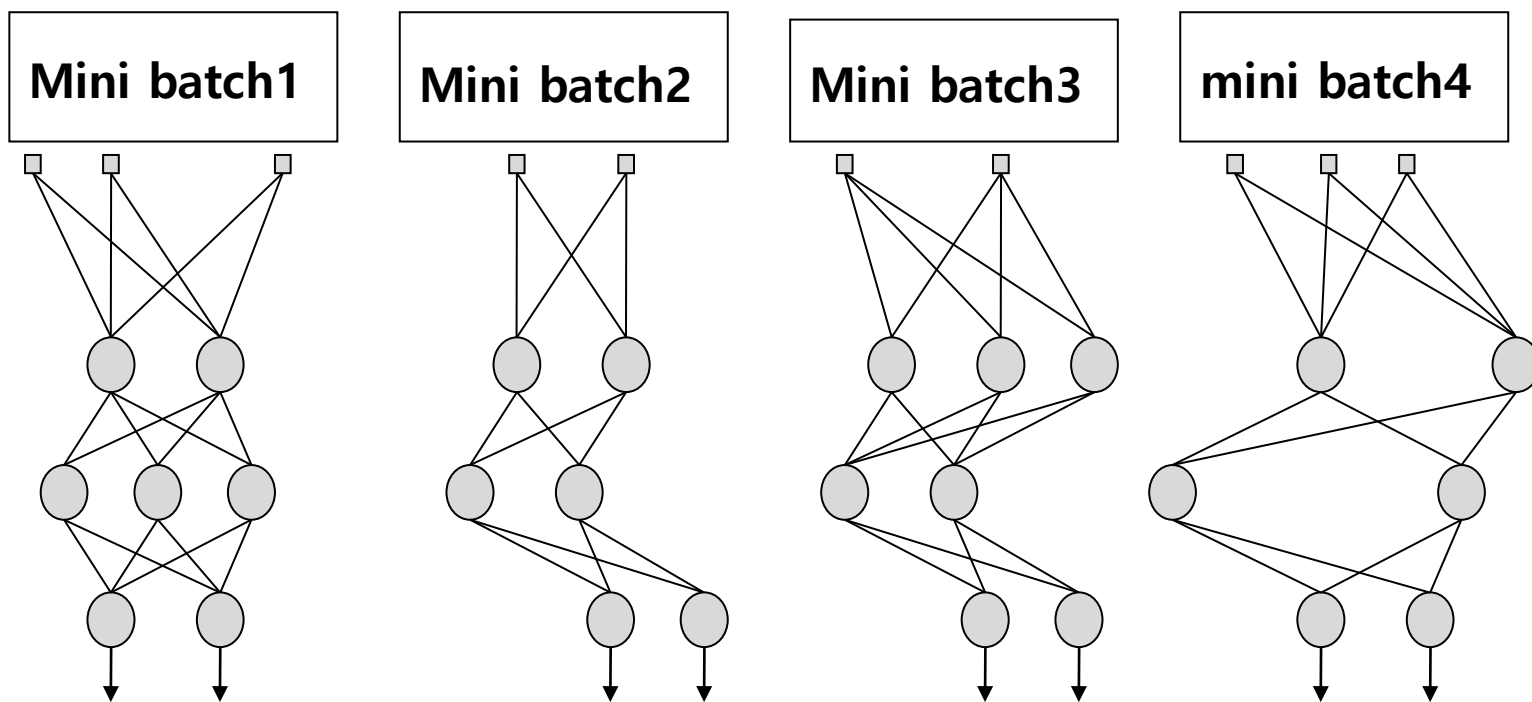
신경망 미니 배치 학습

- 학습 데이터를 다수의 데이터 셋으로 분할해서 각각 학습
 - 각각의 분류기들을 학습시키고 Voting을 통해 최종 분류하는 형태가 드롭아웃 구조와 유사



신경망 미니 배치 dropout

- 드롭아웃 과정을 통해 일부 가중치들이 제거되어 연결관계가 끊어진 것처럼 됨
- 각각의 개별적인 분류기들이 학습되는 구조



다양성 역설 (Paradox of Diversity)

- 다양성 높음 : 같은 데이터에 대해 분류기들이 서로 다른 값 출력
 - 다양성이 낮으면 Voting 효과가 거의 발생하지 않음
- 다양성의 역설 : 정확도와 다양성이 상호 대립되는 특성을 보이는 것

