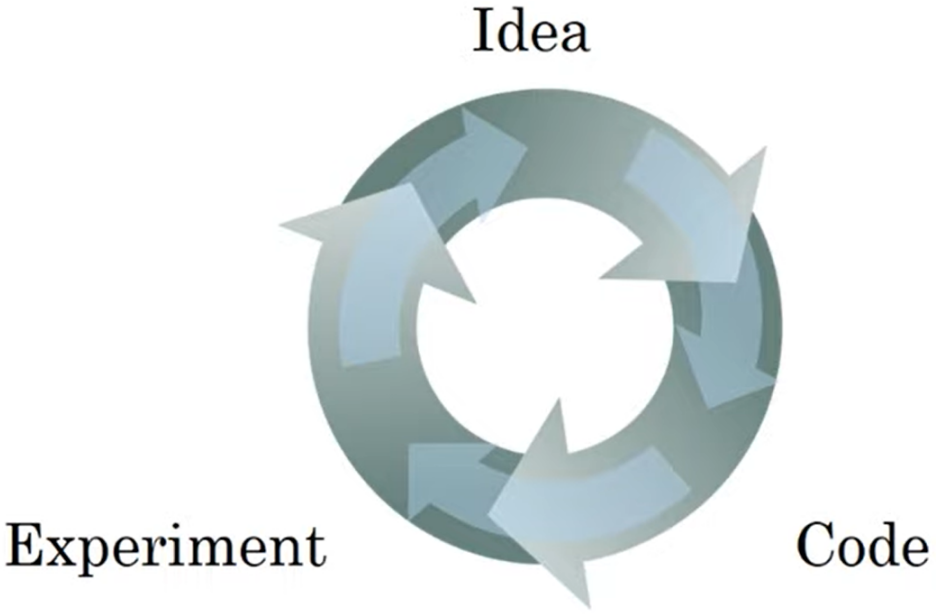
[딥러닝 2단계] 1. 머신러닝 어플리케이션 설정하기

훈련, 개발, 테스트 세트를 적절하게 설정하는 것은 좋은 성능을 내는 네트워크를 빠르게 찾는데 큰 영향을 준다.

하이퍼파라미터

- 신경망의 층 수
- 각각의 층의 은닉 유닛 수
- 학습률
- 활성화 함수



딥러닝을 적용하는 것은 아이디어 → 구현 → 테스트의 반복적인 과정

- 사이클을 효율적으로 도는 것과 데이터 세트를 잘 설정하는 것이 중요

Train/Dev/Test Sets

- Training Set
- Dev Set
  - Hold-out, cross validation(교차검증) 세트
  - 서로 다른 알고리즘을 시험하고 어떤 알고리즘이 더 잘 작동하는지 확인
- Test Set
  - 단일 분류기의 성능 평가
  - 최종 분류기가 어느 정도 성능인지 신뢰있는 추정치를 제공
  - 최종 네트워크의 성능에 대한 비편향 추정을 제공

전통적인 방법

- 과정: 훈련 세트에 대해 계속 훈련 알고리즘을 적용시키면서 학습, 개발 세트에 대해 다양한 모델 중 어떤 모델이 가장 좋은 성능을 내는지 확인, 더 발전시키고 싶은 최종 모델이 나오면 테스트 세트에 그 모델을 적용시켜 성능 측정
- 모든 데이터를 가져와서 위 세 개 세트로 나눔.

머신러능에서는 60/20/20 비율. 10,000개 이하의 샘플에서 최적의 관행이었음.

빅데이터 시대에 100만 개 이상의 샘플 데이터가 생기며 개발, 테스트 세트의 비율이 작아짐. (100만 개 - 98/1/1, 그 이상 - 99.5/0.25/0.25)

→ 데이터 세트가 적다면 전통적인 머신러닝 비율을, 훨씬 크다면 전체 데이터의 20% 혹은 10% 보다 더 작게 설정

Mismatched train/test distribution

딥러닝의 또 다른 트렌드는 더 많은 사람들이 일치하지 않는 훈련/테스트 분포에서 훈련시킨다는 것

예) 사용자가 사진을 업로드하면 그 중 고양이 사진을 찾아 보여주는 앱

- 훈련 세트: 웹페이지의 고양이 사진
  - 전문가스럽고 잘 정돈된 고양이 사진
- 개발/테스트 세트: 앱을 사용하는 사용자들로부터 업로드된 사진
  - 흐릿한 저해상도의 사진

→ 따라서 두 가지 데이터의 분포가 달라질 수 있음.

→ 경험적인 규칙: **dev/test 세트는 같은 분포에서 와야 함!**

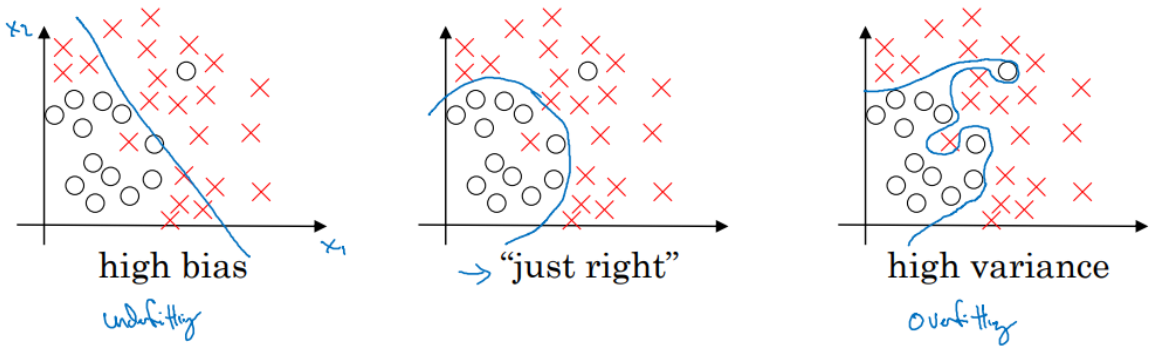
- 개발 세트를 사용해 다양한 모델을 평가하고 성능을 개선할 것이기 때문

비편향 추정이 필요없는 경우, 테스트 세트는 없어도 괜찮음.

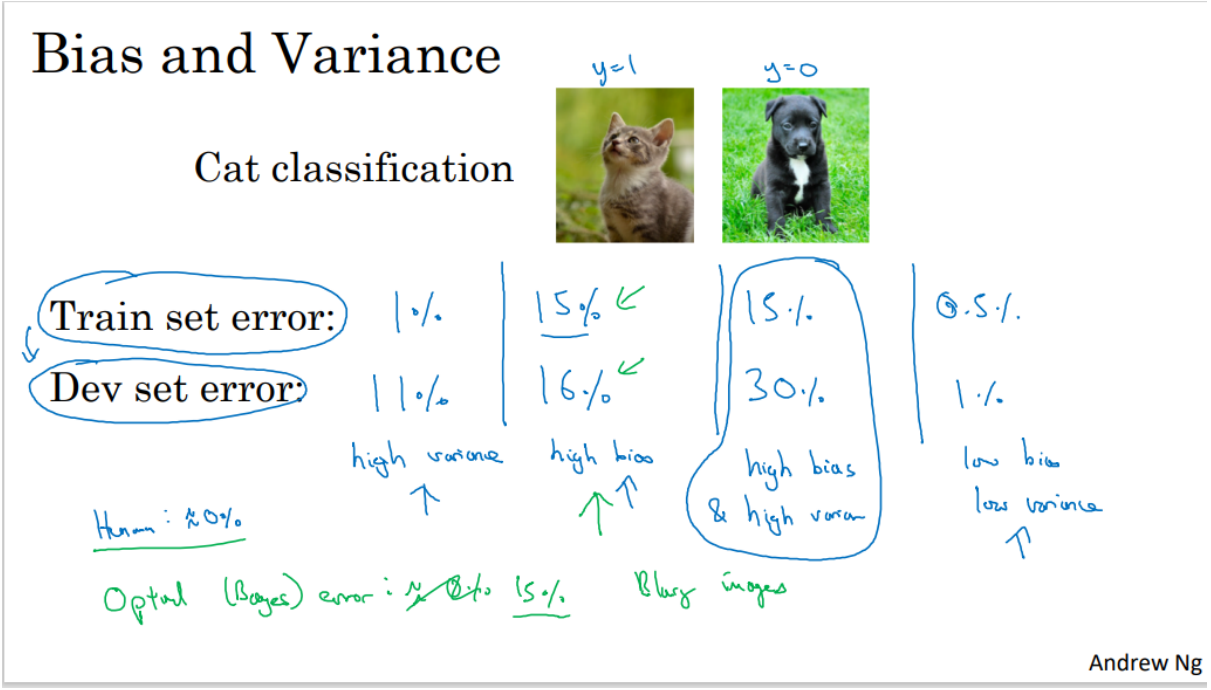
- 개발 세트만 있는 경우에 모든 테스트 세트를 훈련 세트에서 훈련시키고, 이것을 개발 세트에서 평가함.

→ 훈련, 개발, 테스트 세트를 설정하는 것은 **효율적으로 반복**할 수 있도록 하며, **알고리즘의 편향과 분산을 더 효율적으로 측정**할 수 있도록 함.

편향/분산



- **high bias(편향)**
  - 데이터에 맞지 않음
  - 데이터의 과소적합(underfitting)
- **high variance(분산)**
  - 데이터에 완벽하게 맞음
  - 데이터 과대적합(overfitting)
  - 복잡함, 깊은 신경망, 많은 은닉 유닛



최적의 오차(베이지안 Optimal error)가 0인 경우,

(사람이 분류했을 때의 오차율이 0%인 경우)

또한 훈련 세트와 개발 세트가 같은 확률 분포에서 왔을 경우에

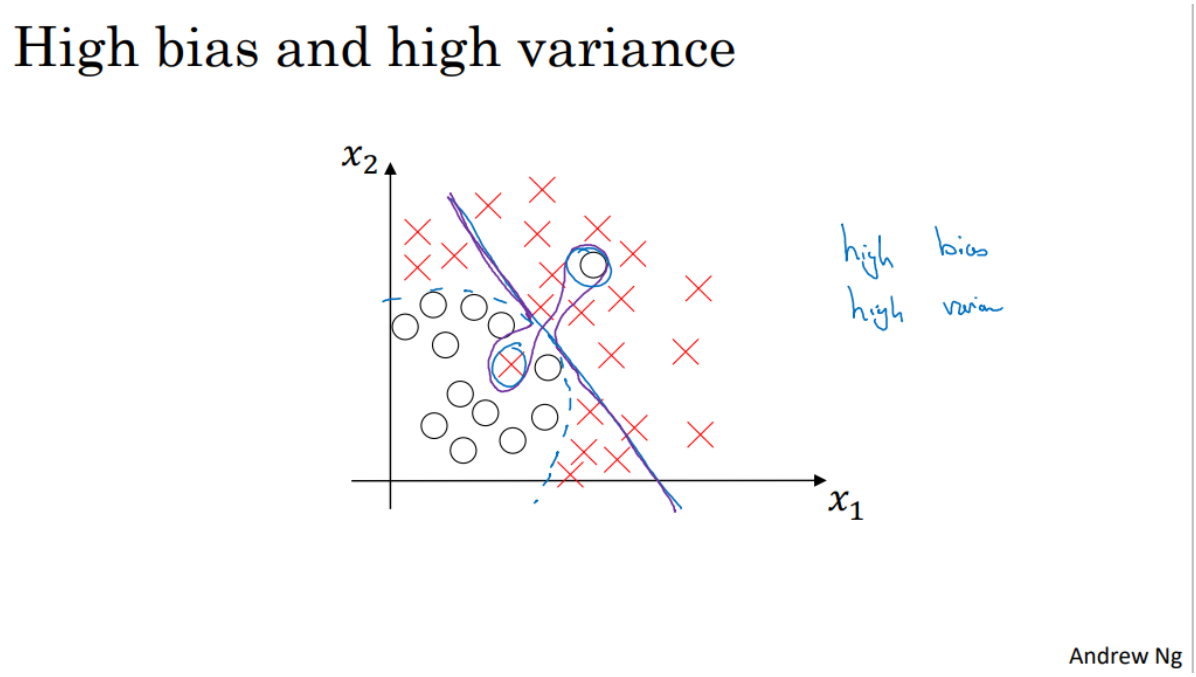
1. 훈련 세트 오차 1 & 개발 세트 오차 11
    a. 훈련세트에 overfitting
    b. high variance
2. 훈련 세트 오차 15 & 개발 세트 오차 16
    a. 훈련세트에 underfitting
    b. high bias
    c. 합리적
3. 훈련 세트 오차 15 & 개발 세트 오차 30
    a. high bias
    b. high variance
4. 둘 다 낮음
    a. low bias
    b. low variance

즉, 훈련 세트의 오차가 크면 high bias, 훈련 세트와 개발 세트 오차의 차이가 크면 high variance임.

최적 오차가 15%인 경우에,

훈련 세트 오차가 15%인 것은 매우 합리적이므로 high bias가 아닌 low variance라고 부름.

High bias & High variance 그래프



- 선형 함수 → underfitting → high bias
- 일부의 데이터에 overfitting → 훈련 세트 오차 낮아짐 → high variance

머신러닝을 위한 기본 레시피

1. High bias (훈련 세트에 맞지 않음) 해결책
    a. 더 많은 은닉층 수/은닉 유닛 개수
    b. 오랜 시간 훈련
    c. 신경망 아키텍처 교체
2. High variance (개발 세트와 훈련 세트의 차이) 해결책
    a. 더 많은 데이터
    b. 정규화

c. 신경망 아키텍처 교체

편향 - 분산 tradeoff

딥러닝 이전에는 해결책들이 편향을 증가시키면 분산을 감소시키거나 편향을 감소시키면 분산을 증가시켰음.

그러나 현대의 딥러닝 빅데이터 시대에는 서로 영향을 미치지 않음.

정규화를 사용하여 분산을 감소시키면 편향이 조금 증가할 수는 있음.