1. 머신러닝 어플리케이션 설정하기

날짜	@2023년 10월 5일

▼ 목차

Train/Dev/Test 세트

1️⃣ Making Good Choices

- 신경망을 훈련시킬 때는 많은 결정을 내려야 함
  - 신경망이 몇 개의 층을 가지는지
  - 각각의 층이 몇 개의 은닉 유닛을 가지는지
  - 학습률은 무엇인지
  - 서로 다른 층에 사용하는 활성화 함수는 무엇인지
- 새로운 애플리케이션을 시작할 때는 이 모든 것에 대한 올바른 값을 추측하는 것이 거의 불가능함

- 딥러닝은 다양한 분야에서 이용되며, 어떤 분야나 애플리케이션의 직관이 다른 애플리케이션 영역에 거의 적용되지 않는 경우가 많음
- 가지고 있는 데이터의 양, 입력 특성의 개수, GPU/CPU 등 훈련을 진행하는 컴퓨터 설정 등 다양한 요인에 의해 최고의 선택이 결정됨

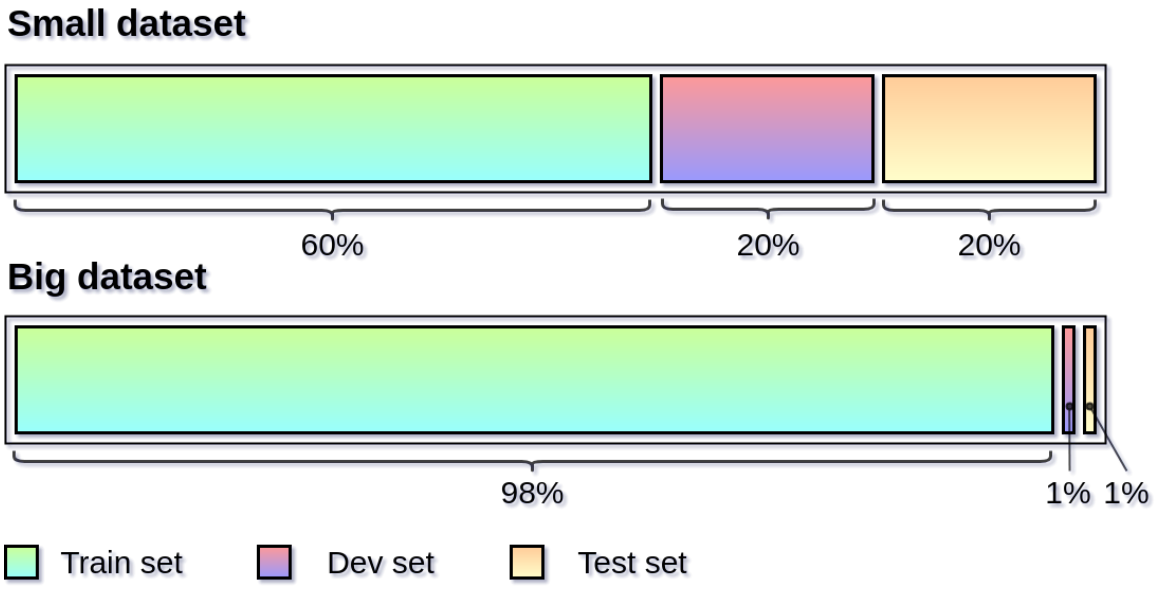
🔹 따라서, **데이터셋을 잘 설정**하고 다음의 **사이클을 효율적으로 반복**함으로써 최선의 신경망을 찾을 수 있음

1. **아이디어**
   - 특정 개수의 층과 유닛을 가지고 특정 데이터 세트에 맞는 신경망을 생성
2. **코드 작성 및 실험 진행**
   - 특정 네트워크 혹은 설정이 얼마나 잘 작동하는지 등의 결과를 얻음
3. **얻은 결과에 기반해 아이디어를 개선**

2️⃣ Train/Dev/Test Sets

💠 train, dev, test set를 설정하여 사이클의 반복을 더욱 빠르게 하고 알고리즘의 편향과 분산을 효율적으로 측정할 수 있음

- train set (훈련 세트)
  - 알고리즘을 훈련
- dev set (개발 세트/교차 검증 세트)
  - 다양한 모델 중 어떤 모델이 가장 좋은 성능을 내는지 확인
- test set (테스트 세트)
  - 훈련과 검증 과정을 충분히 거친 뒤, 최종 모델이 나왔을 때 적용
  - 알고리즘이 얼마나 잘 작동하는지 편향 없이 측정



각 세트의 크기?

- small dataset (100~10,000 samples)
  - train : test = 70 : 30
  - train : dev : test = 60 : 20 : 20
- big dataset (more than 1,000,000 samples)
  - dev와 test 비율을 훨씬 작게 설정해도 됨 (e.g., 98 : 1 : 1)
    - 두 세트 모두 알고리즘이 얼마나 잘 작동하는지 확인하는 것이 목표이므로, 안정적이고 신뢰도 높은 추정치를 제공할 수 있을 정도로 충분히 크기만 하면 되기에 전체 데이터셋에서 차지하는 비율은 중요하지 않음

3️⃣ Mismatched Train/Test Distribution

train/test 분포가 일치하지 않을 경우, 일반적으로 **dev와 test가 같은 분포에서 오도록** 설정하는 것이 좋음

e.g., 사용자의 모든 사진이 업로드된 앱에서, 고양이 사진만을 찾아 보여주고자 할 경우

- train: 인터넷에서 다운받은 고양이 사진
  - 고해상도이며 잘 정돈됨
- dev, test: 사용자에 의해 업로드된 고양이 사진
  - 저해상도이거나 흐릿하게 찍혔을 가능성이 높음

test set은 항상 필요한가?

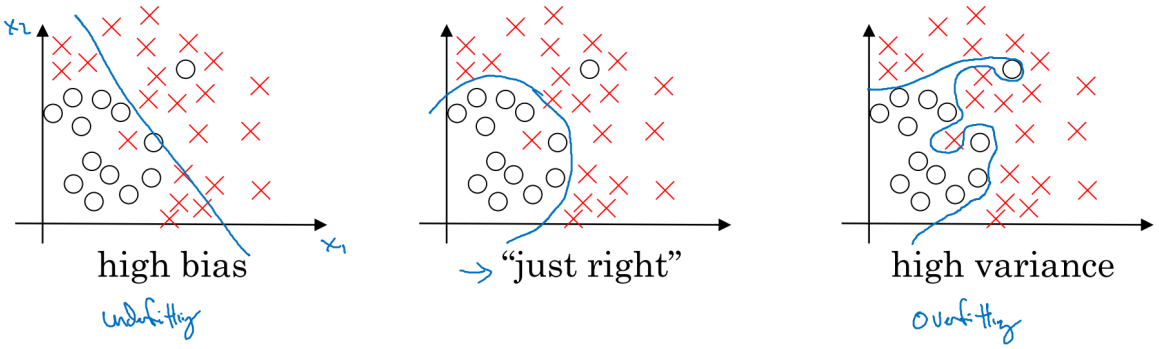
- 🔹 test set의 목표는 최종 네트워크의 성능에 대한 비편향 추정을 제공하는 것이므로, 비편향 추정이 필요 없는 경우에는 test set을 사용하지 않아도 됨

- 이 경우 test set을 train set에 포함시켜 훈련시킨 후 다른 모델 아키텍트를 시도, 이것을 dev set에서 평가하는 과정을 반복하여 좋은 모델을 찾음
  - dev set에 데이터를 맞추기 때문에 성능에 대한 비편향 추정을 주지 않음
  - 과적합이 일어날 수 있음

편향/분산

1️⃣ Bias and Variance

e.g., 회귀 문제 (2차원 예제에서 결정 경계 그리기)



- underfitting(과소적합): high **bias**, low variance
  - 로지스틱 회귀 등 간단한 분류기를 사용하여 분류가 제대로 이루어지지 않은 경우
- overfitting(과대적합): high **variance**, low bias
  - 깊거나 큰 신경망과 같이 복잡한 분류기를 사용하여 데이터에 완벽하게 맞춘 경우

2️⃣ Example: Cat Classification

- 💠 train, dev set error를 확인함으로써 편향과 분산 문제가 있는지 알아볼 수 있음

- train set에 대하여 알고리즘이 얼마나 적합한지 (bias)
- train set에서 dev set로 갈 때 오차가 얼마나 커지는지 (variance)

Assumption!

🔹 아래 분석은 다음 가정에 근거하여 이루어짐
1. train, dev set이 같은 확률 분포에서 옴
2. 사람이 사진을 보고 고양이인지 아닌지를 정확히 판단할 수 있음

- 인간의 판단에 대한 오차가 0에 가까움 = 베이지안 (최적) 오차가 0에 가까움
  - 예를 들어 이미지가 아주 흐릿해서 인간 혹은 그 어떤 시스템도 잘 분류하지 못하는 경우에는 베이지안 오차는 훨씬 커지며, 분류기의 성능 분석에 대한 세부 방식이 달라지게 됨

High Variance (높은 분산)

e.g., train set error 1%, test set error 11%

- train set에서는 매우 잘 분류됐지만 상대적으로 dev set에서는 잘 분류되지 못함
- 즉, train set에 **과대적합**되어 dev set가 있는 교차 검증 세트에서 일반화되지 못함

High Bias (높은 편향)

e.g., train set error 15%, test set error 16%

- train set에 대해 잘 작동되지 않는 것으로 보아 **과소적합됨**
- dev set에 대한 일반화는 합리적으로 이루어졌다고 볼 수 있음
  - dev set에 대한 성능이 train set보다 1%밖에 나쁘지 않기 때문

High Bias, High Variance

e.g., train set error 15%, test set error 30%

- train set에 대해서도 잘 작동되지 않으며, dev set에 대한 일반화도 이루어지지 않음

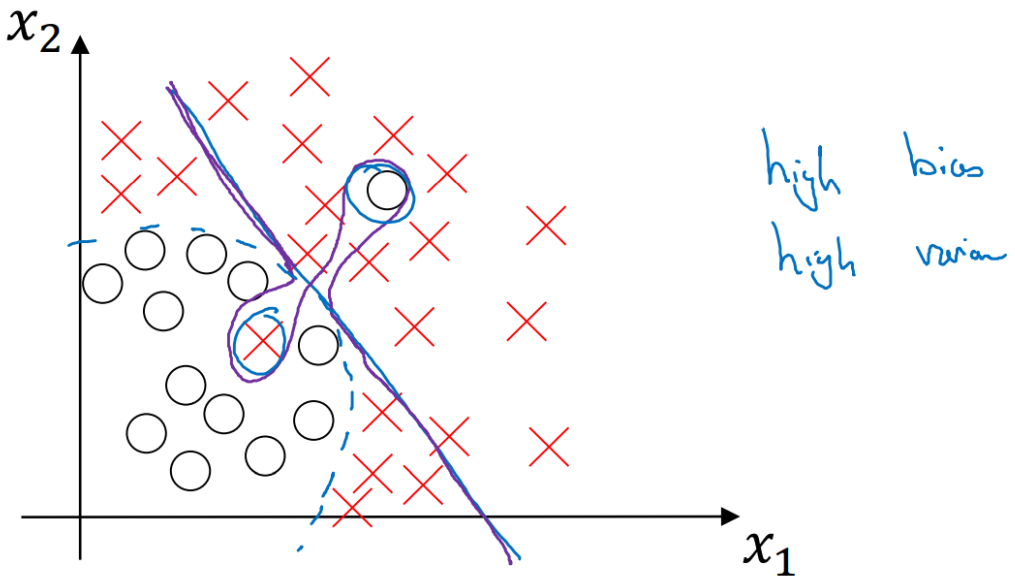
Low Bias, Low Variance

e.g., train set error 0.5%, test set error 1%

- 베이지안 오차(0으로 가정됨)에 근접한 값이므로, 두 성능 모두 합리적임을 알 수 있음

3️⃣ High Bias and High Variance

편향과 분산이 모두 높은 경우 2차원 예제에서의 그래프는 어떻게 그려질까?



- 편향이 높으므로 (과소적합) 기본적으로 거의 선형
- 분산이 높으므로 (과대적합) 중간에 많은 굴곡을 가져 일부 데이터에 과대적합된 형태를 띰

머신러닝을 위한 기본 레시피

1. 최초의 모델 훈련 후 알고리즘의 편향 문제를 판단하기 위해 train set의 성능 확인
    - 편향이 높을 경우 (과소적합)
        - **더 많은 은닉층 혹은 은닉 유닛을 갖는 네트워크를 선택**
            - 과소적합을 해결하고자 할 때 가장 적합한 방법!
        - 더 오랜 시간 훈련시키거나 다른 발전된 최적화 알고리즘을 사용
        - 다양한 신경망 아키텍처 탐색
2. 편향이 낮은 것을 확인했다면, 분산 문제를 판단하기 위해 dev set의 성능 확인
    - 분산이 높을 경우 (과대적합)
        - **더 많은 데이터를 수집**

        - 과대적합을 해결하고자 할 때 가장 적합한 방법이지만, 얻지 못할 수 있음
    - 정규화 시도
    - 다양한 신경망 아키텍처 탐색

Bias-variance Tradeoff?

- ML 시대에는 편향과 분산 중 한쪽을 줄이려 할 때 다른 한쪽이 증가하는 문제가 자주 발생함
- 현대의 DL 빅데이터 시대에는 더 큰 네트워크를 훈련시키거나 더 많은 데이터를 얻는 이 두 단계가 편향만을 감소시키거나 분산만을 감소시키는 좋은 툴이 되어줌
    - 편향과 분산의 균형을 신경쓸 필요가 줄어듦!

출석퀴즈 오답노트

▼ 5. bias-variance trade-off는 무엇을 설명하는 개념인가요?

- 모델이 항상 높은 분산을 가져야 한다는 것을 설명한다. (X)
- 모델의 복잡성과 정확성 간의 관계를 설명한다. (X)
- 훈련 데이터의 크기와 모델 성능 간의 상관관계를 설명한다. (X)