[Week11]_문가을

딥러닝 2단계 : 심층 신경망 성능 향상 시키기

5. 하이퍼 파라미터 튜닝

튜닝 프로세스

<hyperparameters> - 중요도 순으로.

- 학습률(α)
- 모멘텀(Momentum) 알고리즘의 β : *보통 0.9*
- 은닉 유닛의 수
- 미니배치 크기
- 은닉층의 개수
- 학습률 감쇠(learning rate decay) 정도
- 아담(Adam) 알고리즘의 $\beta 1$, $\beta 2$, ϵ : *보통 0.9, 0.999, 10^(-8)*

hyperparameter 조합을 찾을 때

- grid → hyperparameter 수가 적을 때 사용가능.
- 딥러닝에서는 무작위로 점을 설정하는 것이 좋음.

→ 각 하이퍼 파라미터 마다 다양한 값을 시험해볼 수 있음.

- 정밀화 접근

좋은 성능을 보이는 점들 주변으로 작은 영역으로 확대하여, 조밀하게 점을 설정할 수 있도록 함.

적절한 척도(scale) 선택하기

hyperparameter 선택에 적절한 척도를 정하기

- 무작위로 뽑는 것이 합리적인 하이퍼파라미터들 : 은닉 유닛의 수, 은닉층의 수
- 하지만, 학습률의 경우,
  - 1 과 0.0001 사이의 값 중에 균일하게 무작위 값을 고르게 되면, 90%의 값이 1 과 0.1 사이에 존재하기 때문에, 공평하다고 할 수 없음.
  - 따라서 선형 척도 대신 로그 척도에서 하이퍼파라미터를 찾는 것이 합리적.
- 파이썬 구현:

r = -4*np.random.rand() → r = [-4,0] 사이의 무작위 값.

α = 10**r → α = [10^-4, 1] 사이의 무작위 값.

- 지수 가중 이동 평균에서 사용되는 *β :*

마찬가지로 0.9 와 0.999 사이의 값을 탐색하는 것은 비합리적임

→ 1 - *β = 0.1~0.001 로그 척도에서 하이퍼 파라미터 찾기.*

<왜 선형척도에서 샘플을 뽑는 것은 안 좋은가?>

β가 1에 가깝다면, 아주 조금만 바뀌어도 결과가 아주 많이 바뀌게 되기 때문임.

하이퍼 파라미터 튜닝 실전

1. baby sitting one model (모델 돌보기 = 판다 접근)

데이터는 방대하지만 컴퓨터 자원이 많지 않아서, 적은 숫자의 모델을 한 번에 학습 시킬 수 있을 때 사용함.

→ 며칠 동안 매일 모델을 돌보며 학습 속도를 조금씩 바꿔가며 학습시키는 것임.

2. training many models in parallel (동시에 여러 모델 훈련 = 캐비어 접근)

충분한 컴퓨터 자원이 있다면 사용 가능.

→ 하이퍼 파라미터 설정이 다른 여러 모델을 동시에 다룸. → 마지막에는 최고 성능을 보이는 것을 고르면 됨.

6. 배치 정규화

배치 정규화

하이퍼 파라미터 탐색을 쉽게 만들어주고, 신경망과 하이퍼 파라미터의 상관관계를 줄여줌.

심층 신경망에서 각 층의 활성화 값(z^[l])을 정규화.

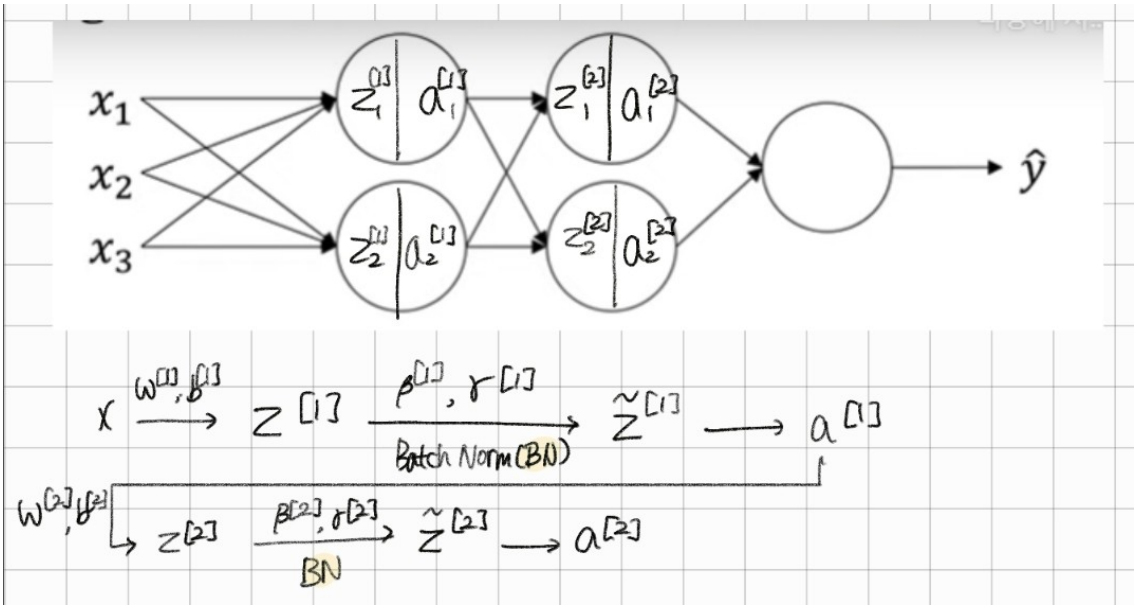
<배치 정규화 구현>

평균이 0이고 표준편차가 1이 되도록 만듦. → 이후 선형 변환을 통해 평균과 분을 다르게 함.

- $\mu = \frac{1}{m}\sum_i z^{(i)}$
- $\sigma^2 = \frac{1}{m}\sum_i (z^{(i)} - \mu)^2$
- $z^{(i)}_{norm} = \frac{z^{(i)} - \mu}{\sqrt{\sigma^2 + \epsilon}}$
- $\tilde{z}^{(i)} = \gamma z^{(i)}_{norm} + \beta$

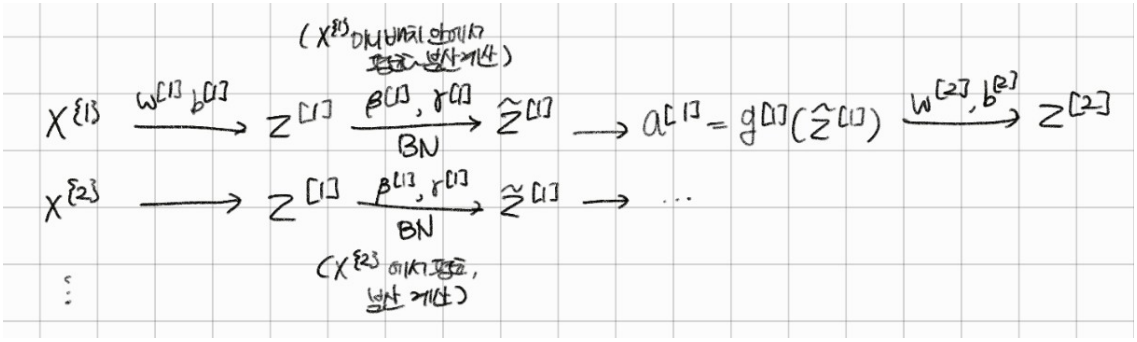
- γ 와 β 는 모델에서 학습 시킬 수 있는 변수임.
- 정규화 이후 다시 선형변환하는 이유는 항상 같은 분포 값을 갖지 않게 하기 위함임.

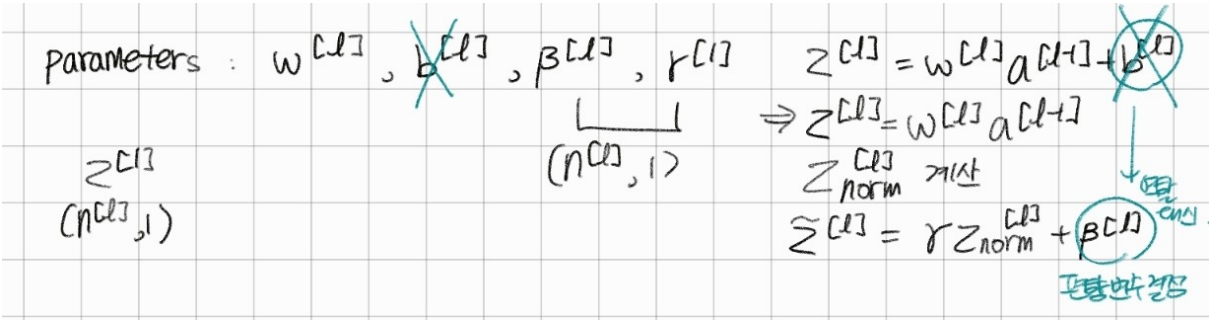
배치 정규화 적용시키기



텐서플로에서 배치 정규화를 구현하는 코드 : tf.batch_normalization

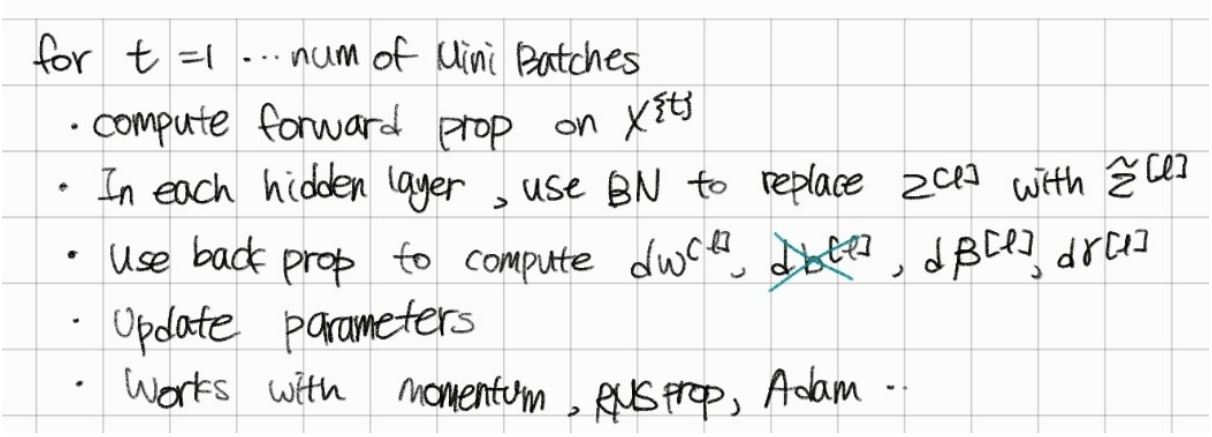
< mini-batch에서 정규화 적용>





배치 정규화를 쓴다면, 상수항 b^[l]은 없앨 수 있음. ← 배치 정규화 과정에서 z의 평균을 빼주면 사라지기 때문임.

<경사 하강법 적용>



배치 정규화가 잘 작동하는 이유는 무엇인가요?

1. 입력 특성 X를 평균 0, 분산 1로 정규화하는 것이 학습속도를 올리는 것처럼, 배치 정규화도 비슷한 일을 함.
2. 앞선 층에서 매개변수가 바뀌어 은닉층의 값이 계속 바뀌더라도 평균과 분산이 동일하게 하여, 입력값이 바뀌어서 발생하는 문제인 공변량 변화를 안정화 시킴. 뒤쪽에 있는 층들이 학습하기에 용이하게 함.
3. 파라미터의 정규화

- 배치 정규화의 또 다른 효과는 파라미터의 정규화(regularization)임. 미니배치로 계산한 평균과 분산은 전체 데이터의 일부로 추정한 것이기 때문에 잡음이 끼어있음.
- 드롭아웃의 경우 은닉유닛에 확률에 따라 0 혹은 1을 곱하기 때문에 곱셈 잡음이 있음. 배치 정규화의 경우 곱셈잡음($\times 1/\sigma 1$)과 덧셈 잡음($+(-\mu)$)이 동시에 있음. 따라서 약간의 정규화 효과가 있다.
- 은닉층에 잡음을 추가한다는 것은 이후 은닉층이 하나의 은닉 유닛에 너무 의존하지 않도록 만듦.
- 큰 미니배치를 사용시 이 정규화 효과는 상대적으로 약해짐.

테스트 시의 배치 정규화

독립된 μ와 σ^2 의 추정치 사용

→ 미니 배치들의 지수가중평균을 추정치로 사용함.