

5. 하이퍼파라미터 튜닝

Tuning Process

Hyperparameters

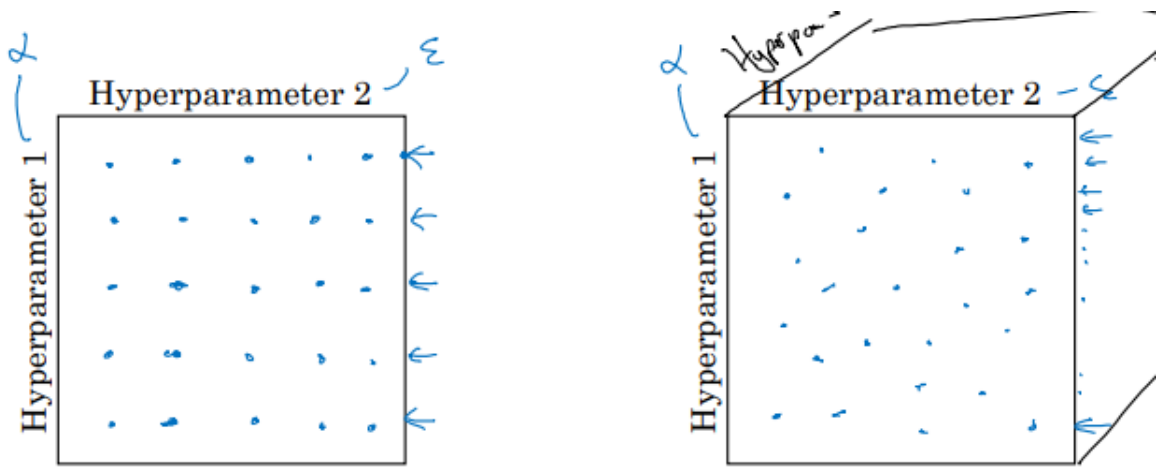
- α (learning rate)
- β (momentum) - 주로 0.9로 설정
- $\beta_1, \beta_2, \epsilon$
- number of layers
- number of hidden units
- learning rate decay
- mini-batch size

하이퍼파라미터 중요도

- 1st: α (learning rate)
- 2nd: momentum, number of layers, mini-batch size
- 3rd: number of layers, learning rate decay
- others

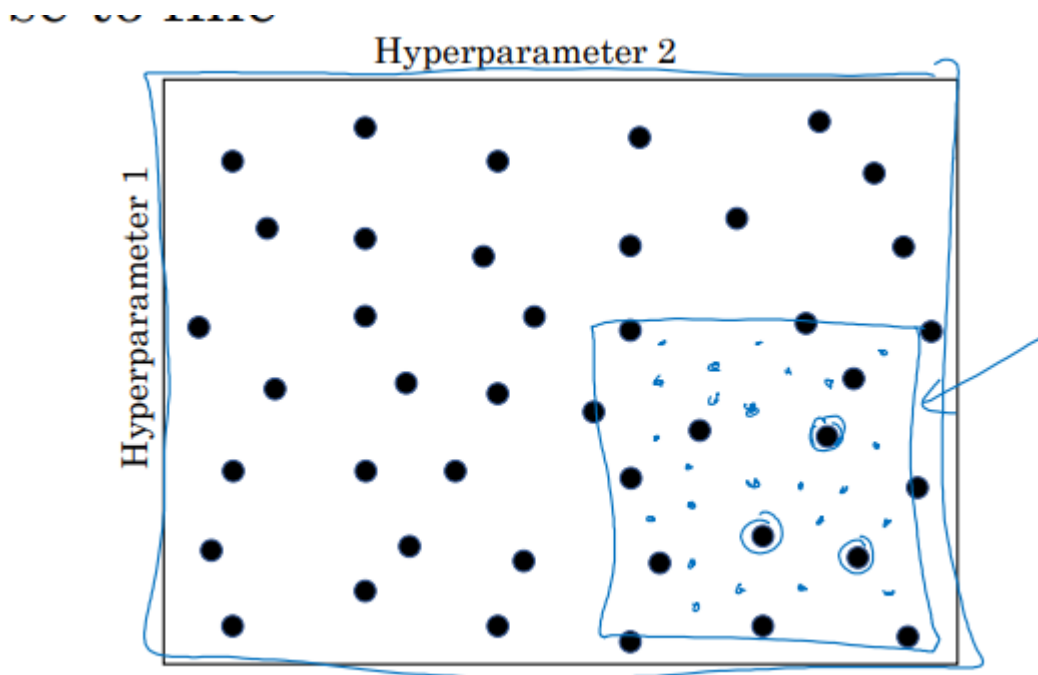
Try random values (Don't use a grid)

- 격자점 탐색
 - 하이퍼파라미터 수가 적을 때는 괜찮지만 딥러닝에서는 사용하지 않는 것이 좋음
- 랜덤 탐색
 - 무작위로 값을 선택하는 것이 더 효율적임
 - 어떤 하이퍼파라미터가 문제 해결에 더 중요한지 알 수 없기 때문에 더 좋은 하이퍼파라미터를 찾을 수 있음



Coarse to fine

- 하이퍼파라미터의 범위를 좁혀 더 좁은 범위에서 랜덤으로 탐색함



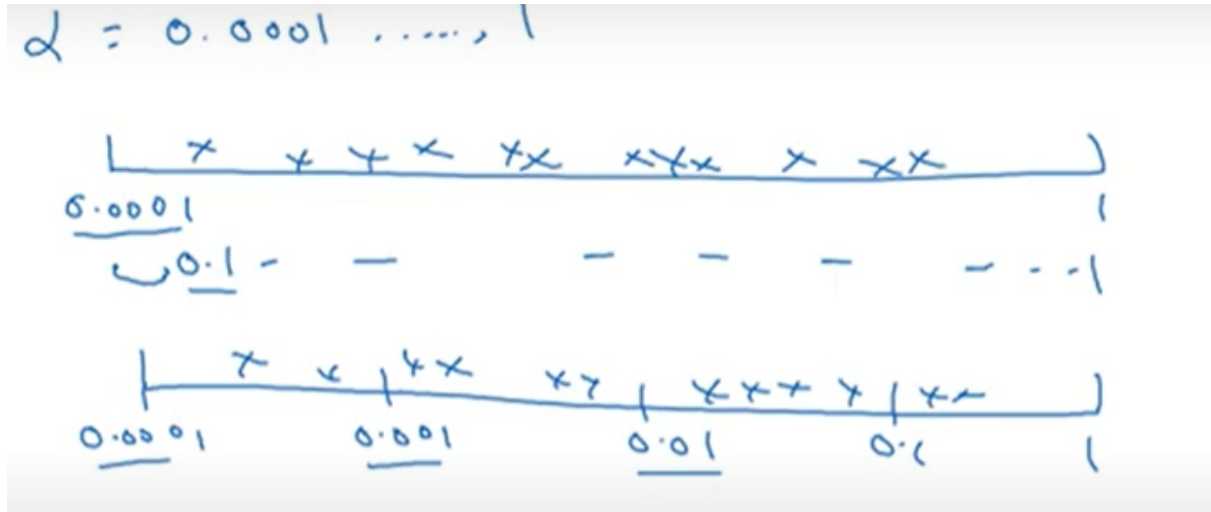
Using an appropriate scale to pick hyperparameters



uniformly random한 방식보다는 적절한 척도를 정해 하이퍼파라미터를 랜덤으로 탐색하는 것이 좋다

Appropriate scale for hyperparameters

- learning rate을 튜닝하는 경우 linear scale 대신 log scale을 적용하는 것이 더 합리적이다.

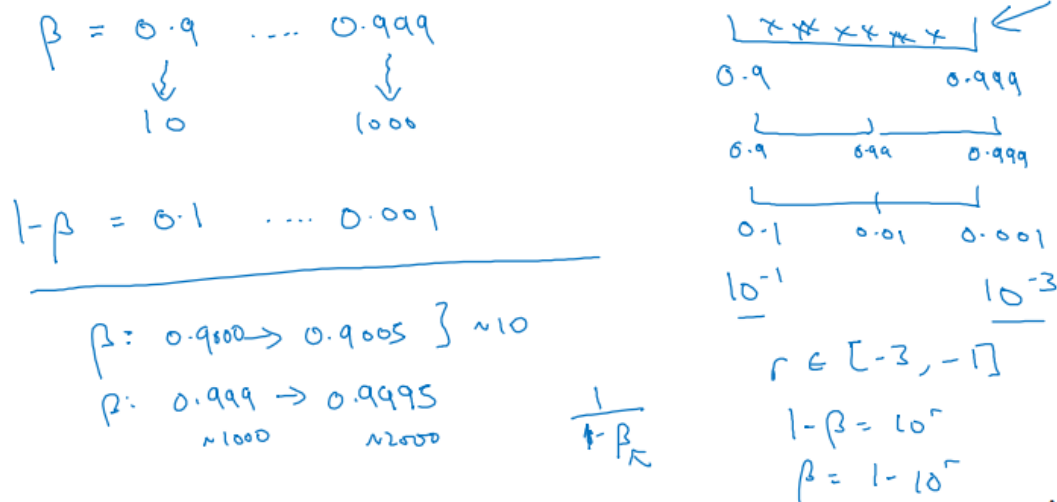


- implementation

```
r = -4 * np.random.rand()
alpha = 10 ^ r
```

Hyperparameters for exponentially weighted averages

- $0.9 \leq \beta \leq 0.99$ 인 경우 0.9 와 0.99 사이의 값을 uniformly random하게 탐색하는 것보다 $(1 - \beta)$ 에 대해 값을 탐색하는 것이 더 합리적임
- beta가 1에 가까울수록 결과에 크게 영향을 미치기 때문에 샘플을 더 세밀하게 뽑아야 하기 때문임



Andrew Ng

Hyperparameter Tuning in Practice

Re-test hyperparameters occasionally

- 이전에 찾은 하이퍼파라미터가 다음 번에는 최적값이 아닐 수 있기 때문에 항상 re-test 해야함

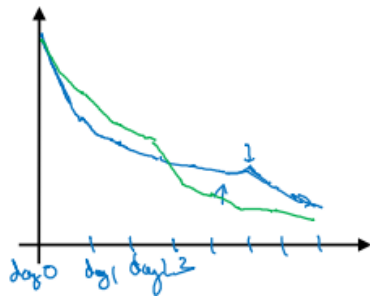
튜닝 방법 1 - Babysitting one model

- 데이터가 방대하지만 컴퓨팅 자원(CPU, GPU)이 적을 때
- 한번에 하나의 모델을 학습
- 하나의 모델로 매일 성능을 지켜보면서 학습 속도를 바꾸는 방식

튜닝 방법 2 - Training many models in parallel

- 컴퓨팅 자원이 충분해 여러 모델을 한번에 학습 시킬 수 있을 때 사용
- 동시에 여러 모델을 학습 시킴
- 여러 하이퍼파라미터 설정을 시험해보고 마지막에 가장 좋은 성능을 내는 것을 선택

Babysitting one model



Training many models in parallel

