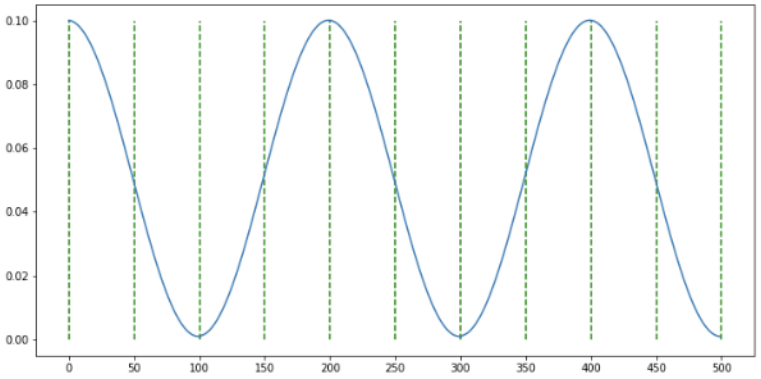
[6주차] NN Advanced

Learning Rate Scheduler

CosineAnnealing LR 과 CyclicLR은 모두 특정한 주기를 따르면서 큰 학습률에 도달했을 때 local minimum에서 탈출하기 쉽고, 다양한 크기의 학습률로 학습함으로써 더 나은 솔루션을 찾을 수 있다.

1. CosineAnnealing LR

learning rate가 cosine 함수의 주기를 따르면서 증감하는 과정을 반복하는 방식이다.




```
scheduler = CosineAnnealingLR(optimizer, T_max=100, eta_min=0
```

`T_max` : 학습률이 감소할 주기

`eta_min` : learning rate의 하한선

2. CyclicLR

CosineAnnealingLR은 단순한 cosine 곡선인 반면에 CyclicLR은 3가지 형태로 주기적인 learning rate 증감을 반복하는 방식을 지원한다. step size는 최저점(최고점)에서 최고점(최저점)으로 도달하는 주기를 말하며, 연구에서는 일반적으로 미니 배치의 2-10배의 크기가 좋다고 주장하고 있다. 또한, max_lr 의 1/3 또는 1/4을 base_lr로 정하는 것을 제안하였다.

`base_lr` : learning rate의 하한선

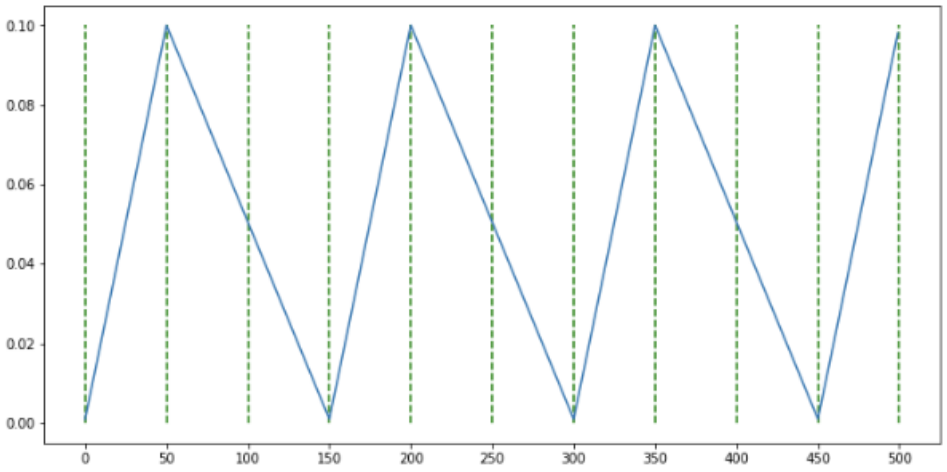
`max_lr` : learning rate의 상한선

`step_size_up` : base_lr → max_lr로 증가하는 에포크 수

`step_size_down` : max_lr → base_lr로 감소하는 에포크수

`gamma` : mode가 exp_range인 경우 사용하는 파라미터, 지수식 밑에 해당

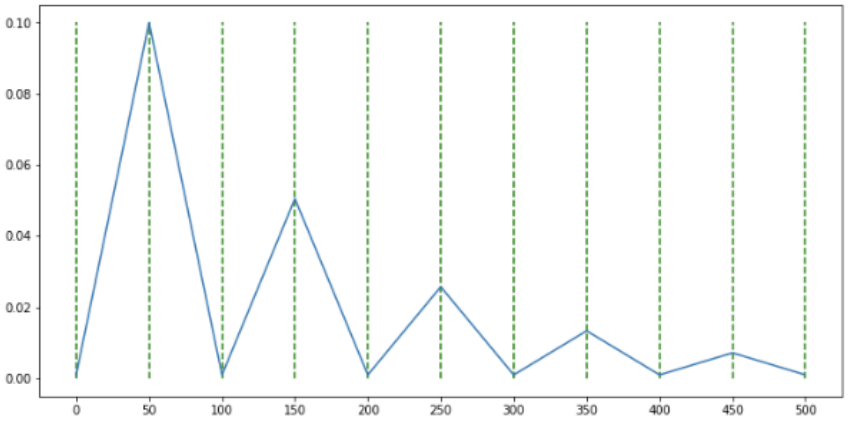
1) `mode` = triangular




```
scheduler = CyclicLR(optimizer, base_lr=0.001, max_lr=0.1,
                                                step_size_up=50, ste
                                                mode='triangular')
```

2) `mode` = triangular2

주기가 반복되면서 learning rate의 상한선이 점점 1/2 씩 줄어들다가 마지막에 수렴하게 된다.

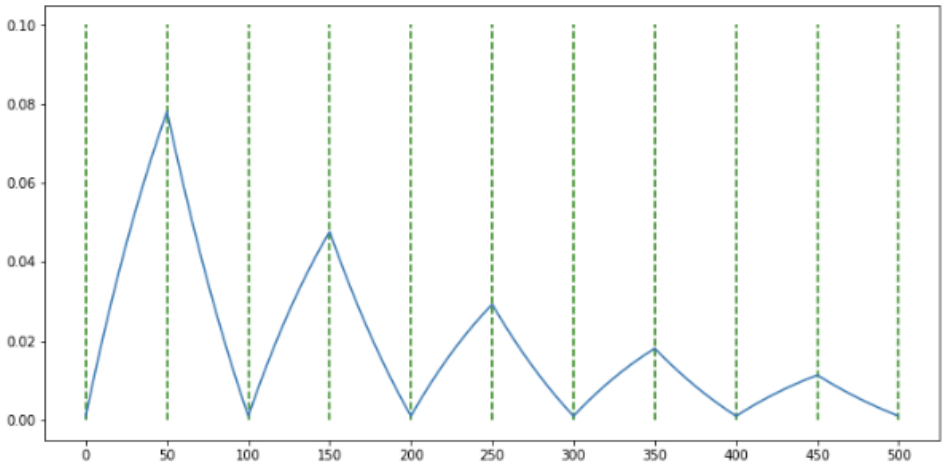



```
scheduler = CyclicLR(optimizer, base_lr=0.001, max_lr=0.1,
                                                step_size_up=50, ste
                                                mode='triangular')
```

- step_size_down=None 으로 설정할 경우, 증감하는 주기가 모두 step_size_up=50 가 된다

3) `mode` = exp_range

선형적으로 증감하는 이전 모드들과 달리, 지수적으로 증감한다.




```
scheduler = CyclicLR(optimizer, base_lr=0.001, max_lr=0.1,
                                                step_size_up=50, ste
                                                 mode='exp_range', g
```

출처

https://gaussian37.github.io/dl-pytorch-lr_scheduler/

https://velog.io/@iissaacc/Cyclical-Learning-Rate

Training Error와 Generalization Error사이 간극을 줄이는 방안

세션에서 소개한 방법 : Penalty Term, Dropout, Data Augmentation

Regularization (1) - Early Stopping

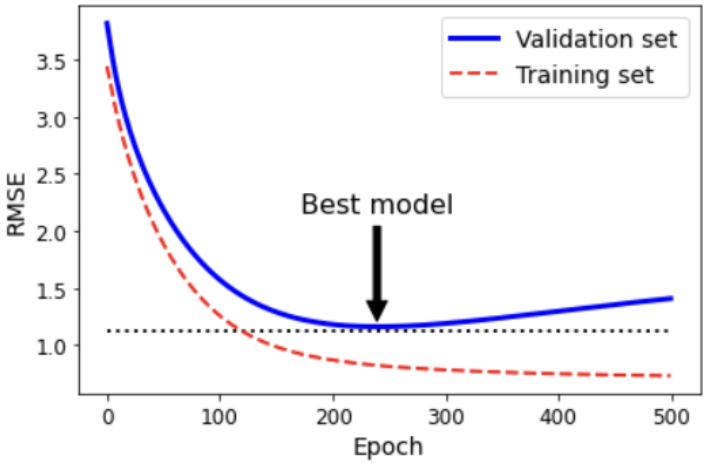
모델 훈련하는 epoch 수에 따라, 과적합이 발생할 수 있다. 너무 많은 epoch를 설정한 경우, 과대적합이 나타나는 한편, epoch 수가 너무 적으면 과소적합이 나타난다.

이와 같은 문제 상황에서 반복 학습 알고리즘을 규제하는 방법으로 **Early Stopping**이 있다. 이는 epoch 수를 높게 설정하여 모델을 돌렸을 때, **validation error가 최솟값에 도달**

하면 **훈련을 중지**시키는 것이다. 이로써 불필요한 학습을 줄이고 **시간과 리소스를 절약**할 수 있다.

일반화 성능이 최대가 되는 지점을 **sweet spot**이라 하며, 이 이점에서 최적의 모델이 결정된다.

그러나, validation dataset에 의존하기 때문에 train dataset과 분포가 다른 경우 잘못된 판단을 내릴 위험이 있다.



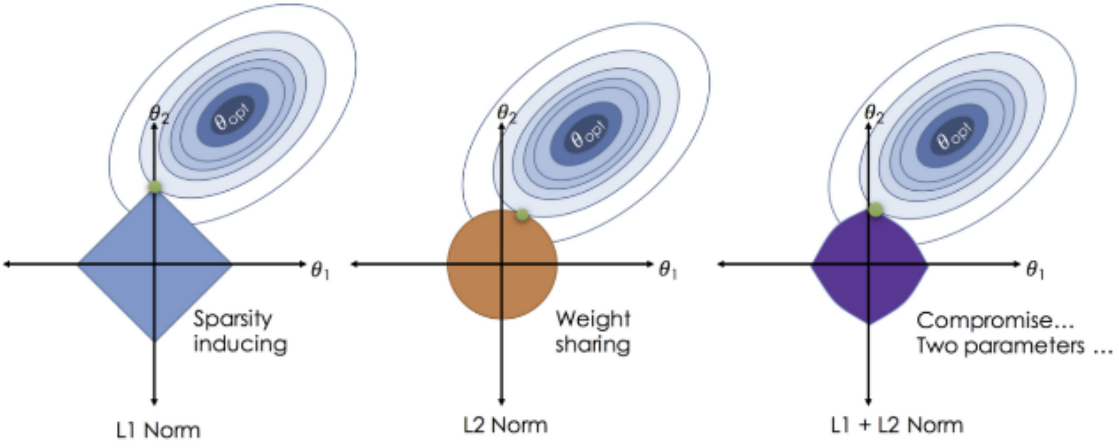
Regularization (2) - Elastic Net (L1 + L2 Norm)

엘라스틱 넷 (Elastic Net) 은 릿지 회귀와 라쏘 회귀를 절충한 모델이다. 주로 **특성의 개수가 훈련 데이터 개수보다 많은 경우** 또는 **일부 특성이 강하게 연관**된 경우 Elastic Net이 선호된다. 이때, 상관성이 높은 다수의 변수들을 **모두 선택하거나 제거**한다. (group effect) 단, L1과 L2 규제가 결합된 규제로 인해 수행 시간이 상대적으로 오래 걸린다는 단점이 있다.

Elsatic Net은 아래와 같이 표현되며, Lasso와 Ridge의 비중을 **혼합 비율 r**로 조절한다.

$$argmin_{\beta} MSE(\theta) + r\alpha \times \Sigma_n|\theta_1| + \frac{1-r}{2}\alpha \times \Sigma_n{\theta_1}^2$$

- r = 0 : 릿지 회귀
- r = 1 : 라쏘 회귀



출처 : https://soobarkbar.tistory.com/30