유런 10주차 정리

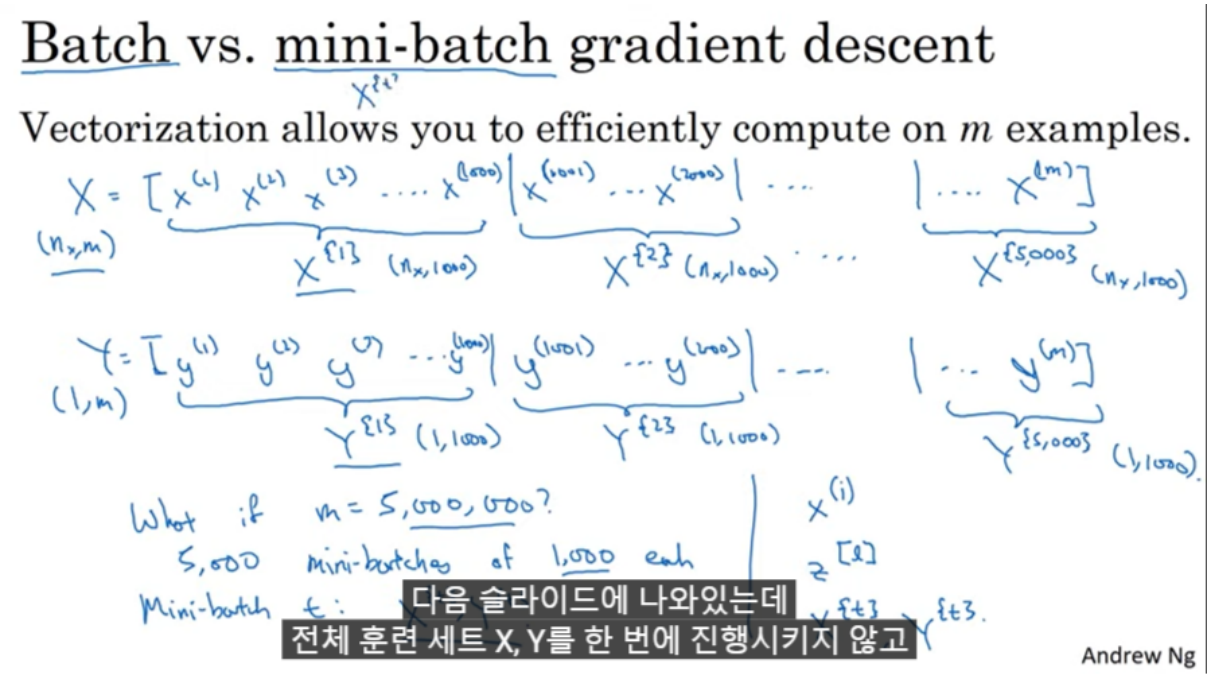
미니 배치 경사 하강법

모델을 빠르게 학습시키는 방법

큰 데이터 세트에서 학습시키는 것은 오래 걸리는 것이기에 좋은 최적화 알고리즘 방법을 찾아야한다.

벡터화 하면 명시적인 for 문이 없어도 훈련이 가능하다.

m 이 너무 크다면 똑같이 느릴 수 있다. 경사 하강법의 다음 단계를 밟기 전에 모든 training set 를 밟아야한다. 거대한 훈련 샘플을 다 돌기 전에 경사 하강법이 진행되도록 하면 더 빠른 최적화를 이룰 수 있다.



baby training set(5000000개) = mini batch(1000개) $x^{\{1000\}}$

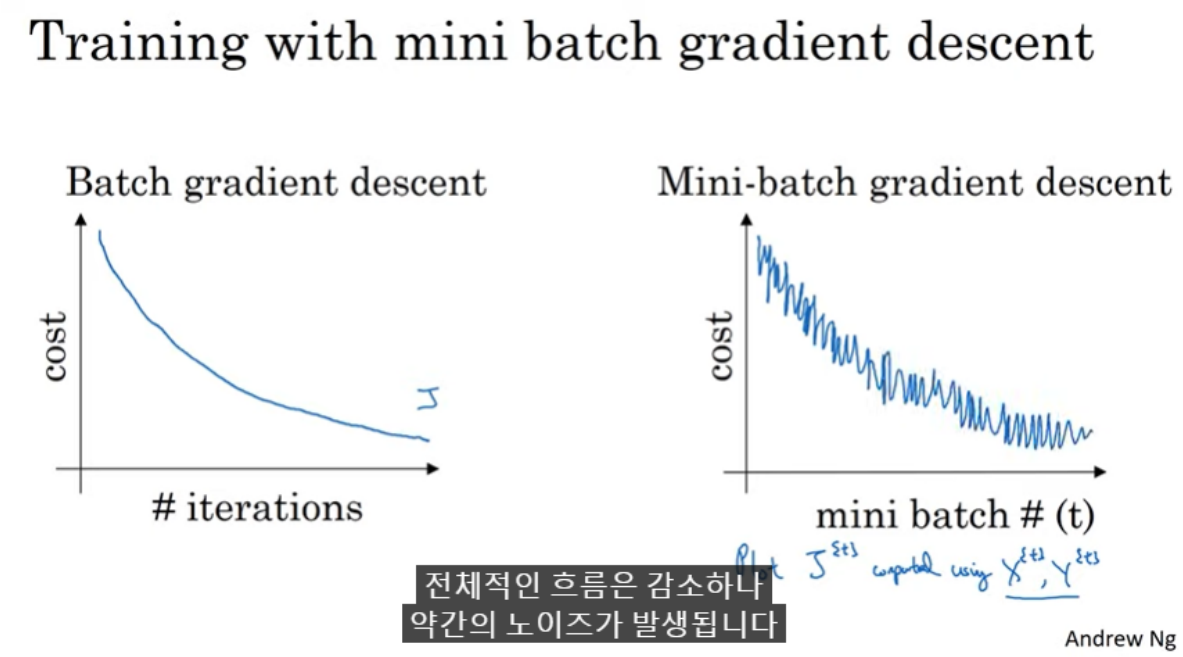
x^{t}, y^{t} 를 가지고 경사하강법을 진행한다.

따라서 배치 경사 하강법에서 훈련 세트를 거치는 한 반복은 오직 하나의 경사 하강법을 거치고,

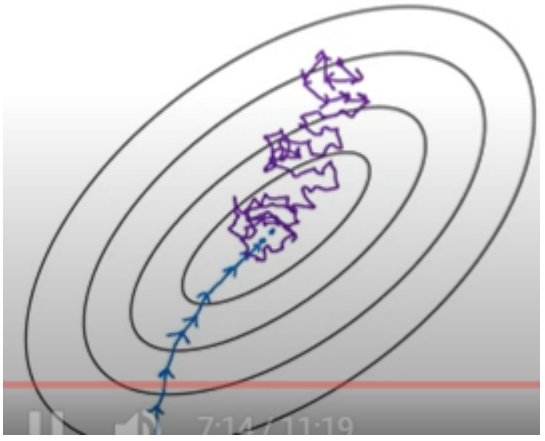
미니 배치 경사 하강법의 경우, 훈련 세트를 거치는 한 반복은 5000개의 경사 하강 단계를 거친다.

훈련 세트가 많다면 미니 배치 경사 하강법이 빠르게 작용한다.

미니 배치 경사하강법 이해하기



모든 반복에서 아래로 내려가진 않지만 큰 흐름은 내려간다.



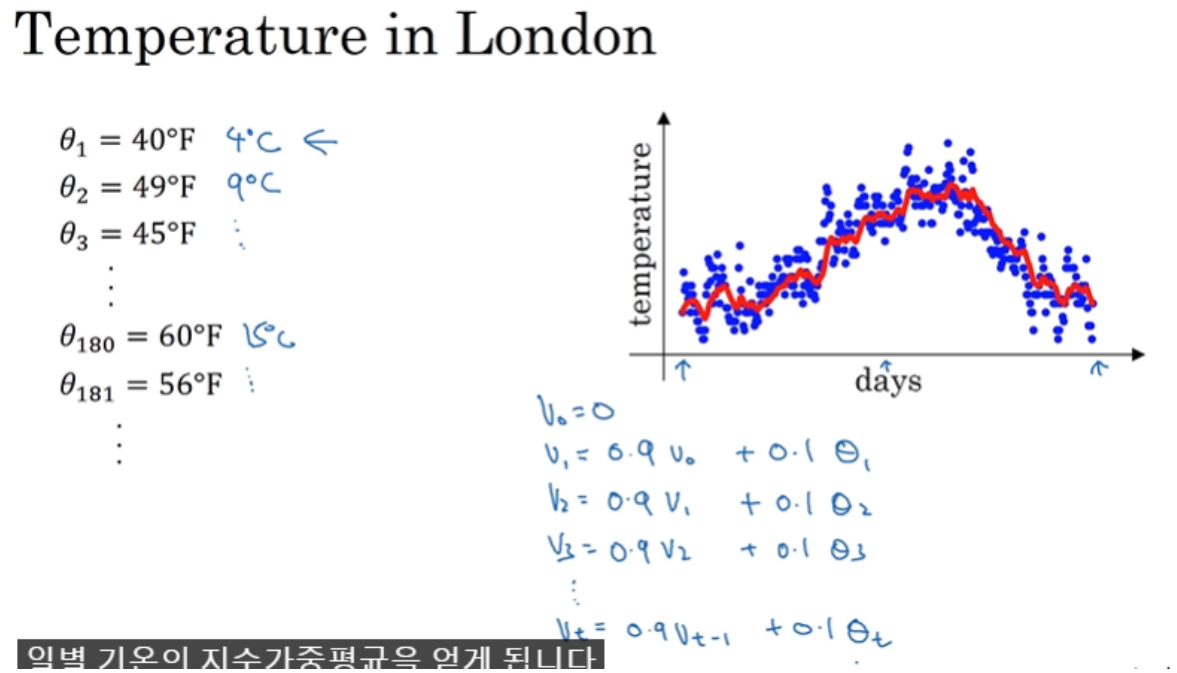
파란색이 batch, 보라색이 mini-batch

보통 미니 배치 크기는 2의 제곱수로 선택한다.

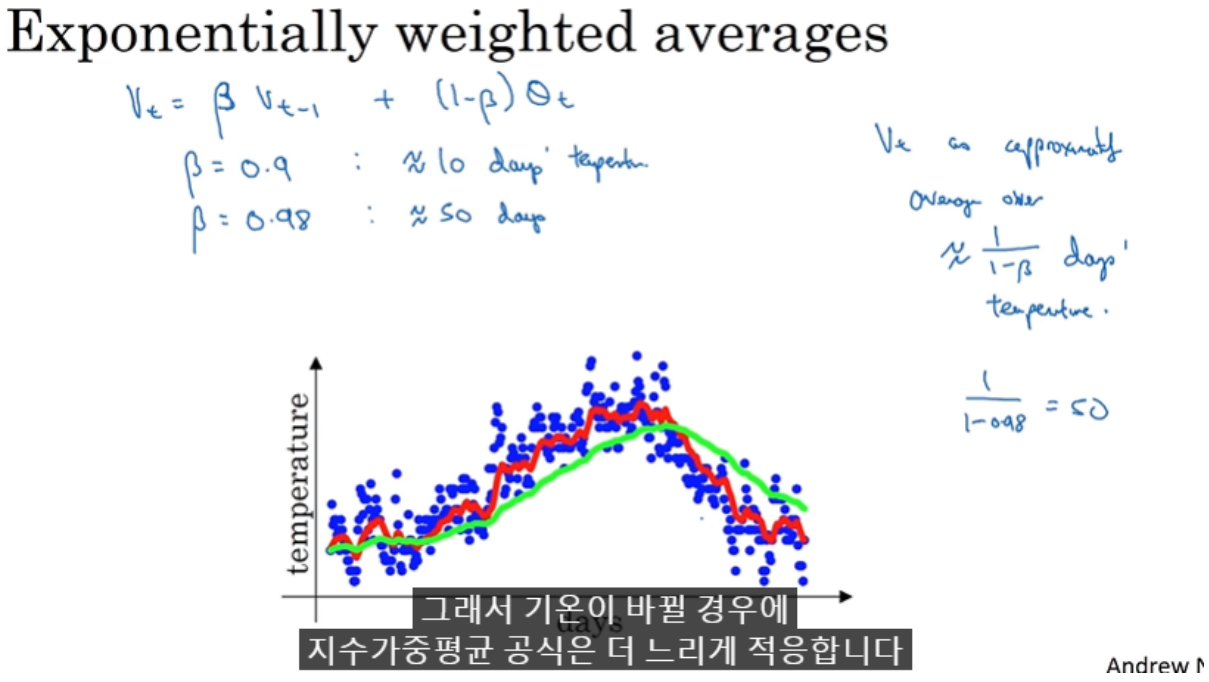
모든 배치에 대해 cpu, gpu 메모리에 맞는지 확인해본다. 메모리에 맞지 않으면 성능이 갑자기 나빠지고 훨씬 나빠지게 된다.

따라서 미니 배치 사이즈를 어떻게 선택하는지에 따라 학습 속도의 차이가 나기에 최적의 값을 찾아내는 것이 중요하다.

지수 가중 이동 평균



이전 값에 0.9를 곱하고 해당 날의 기온에 0.1 를 곱한다. 과거의 경향성과 새로운 경향성을 더한다.

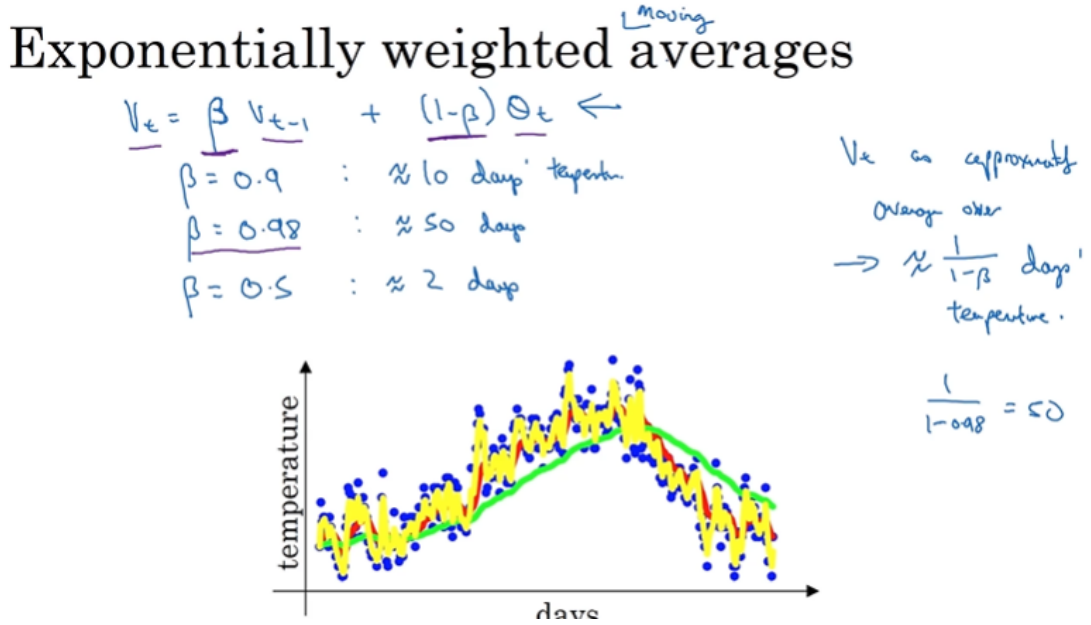


v_t 는 1/1-β 기간 동안 기온의 평균을 의미

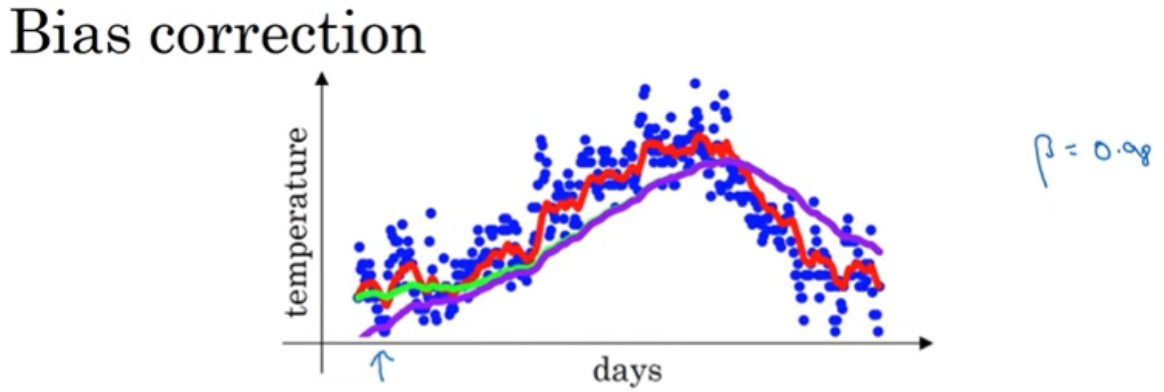
β 가 0.9 와 같은 경우 이는 10일 동안 기온의 평균과 같다. (붉은색)

β 가 1 과 가까우면 50일의 평균과 같다. 값이 커질수록 곡선이 부드러워진다(연두). 더 많은 날짜의 평균을 사용하기 때문이다.

β 가 0.5 이면 2일의 평균을 쓰기 때문에 더 크게 진동한다.



- 얼마의 기간이 이동하면서 평균이 구해졌는가? 는 아래의 식으로 대략적으로 구할 수 있다.
  - β=(1−ε) 라고 정의 하면
  - $(1-\varepsilon)^n = 1/e$ 를 만족하는 n 이 그 기간이 되는데, 보통 ε1 으로 구할 수 있다.
- 지수 가중 이동 평균의 장점은 구현시 아주 적은 메모리를 사용한다는 것이다.



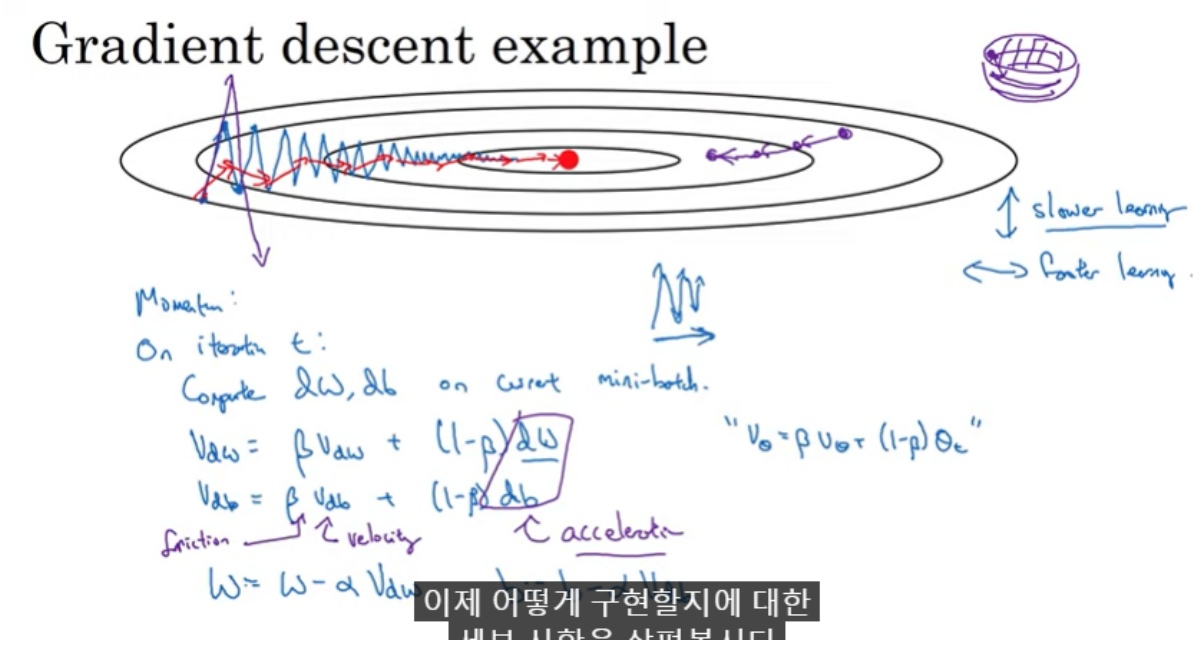
보라색 bias 를 바로잡을 것이다.

보라색 선의 초기 단계에서 0에 가까운 것을 알 수 있다. 한 해의 첫 두 날짜를 추정한 값의 좋지 못한 결과이다.

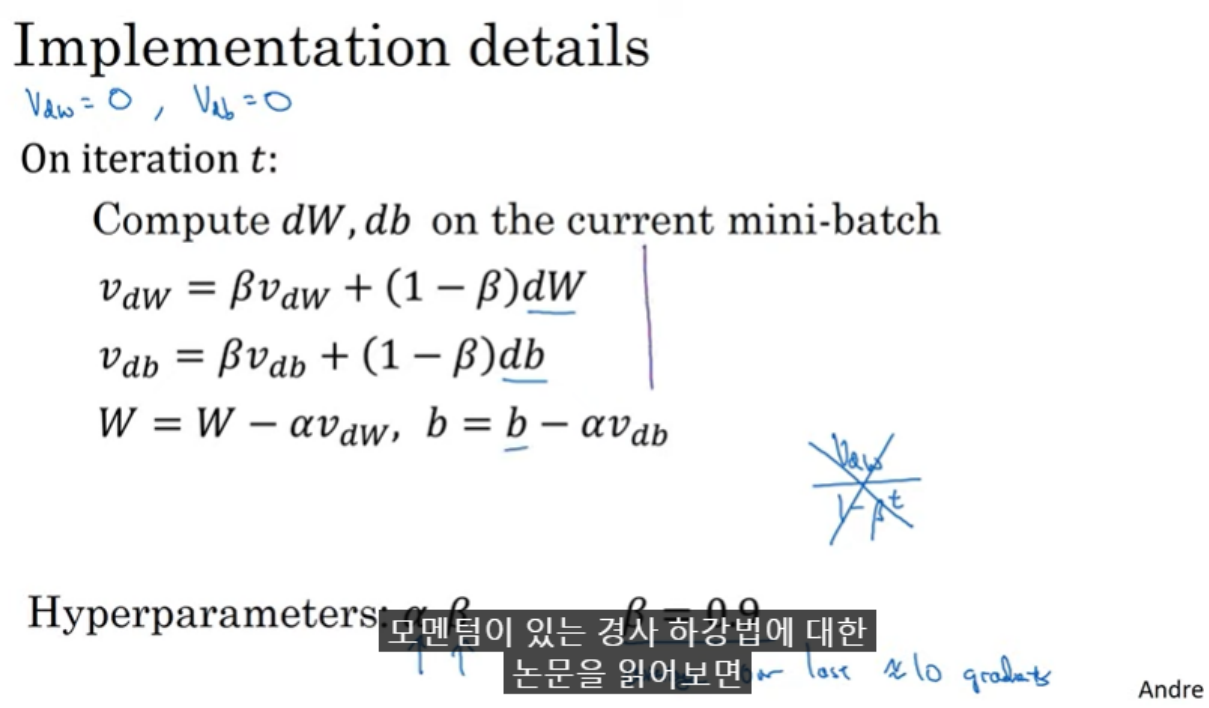
- 따라서 $v_t/(1-\beta_t)$ 를 취해서 초기 값에서 실제값과 비슷해지게 한다.
- 보통 머신러닝에서 구현하지는 않는다. 시간이 지남에 $(1-\beta_t)$ 는 1에 가까워 져서 원하는 값과 일치하게 되기 때문이다.

Momentum 최적화 알고리즘

이전 값을 보고 결정한다. 수직 방향은 양수 음수가 번갈아 가면서 나오므로 잘 안 움직이고 수평 방향은 오른쪽을 향하기 때문에 오른쪽으로 가속화된다.



Algorithm



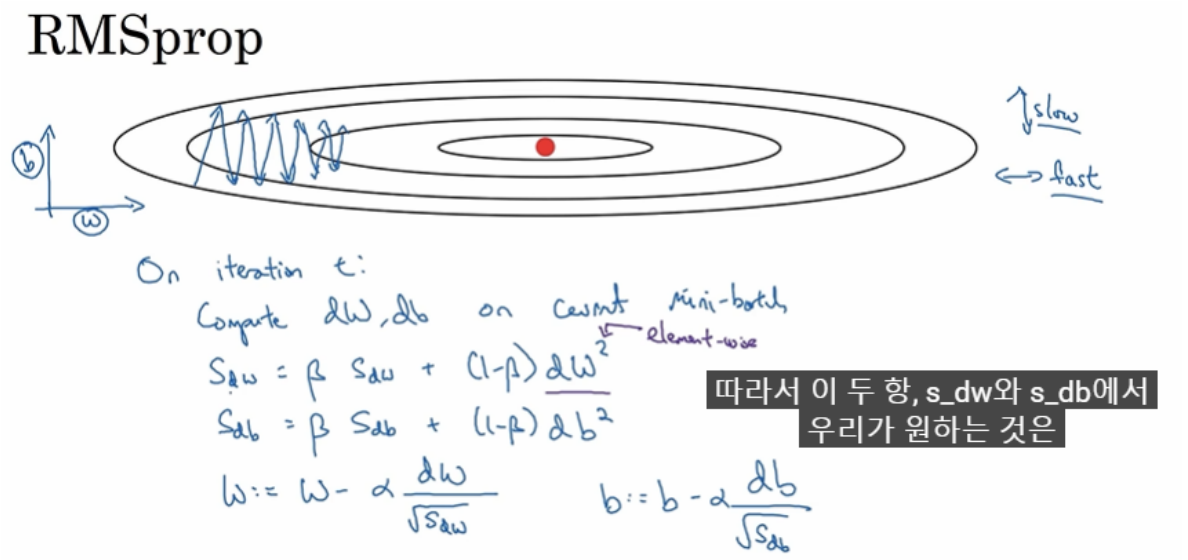
RMSProp 최적화 알고리즘(Root Mean Square)

반복 t 에서 현재의 미니배치에 대한 보통의 도함수 dw db 를 계산할 것이다.

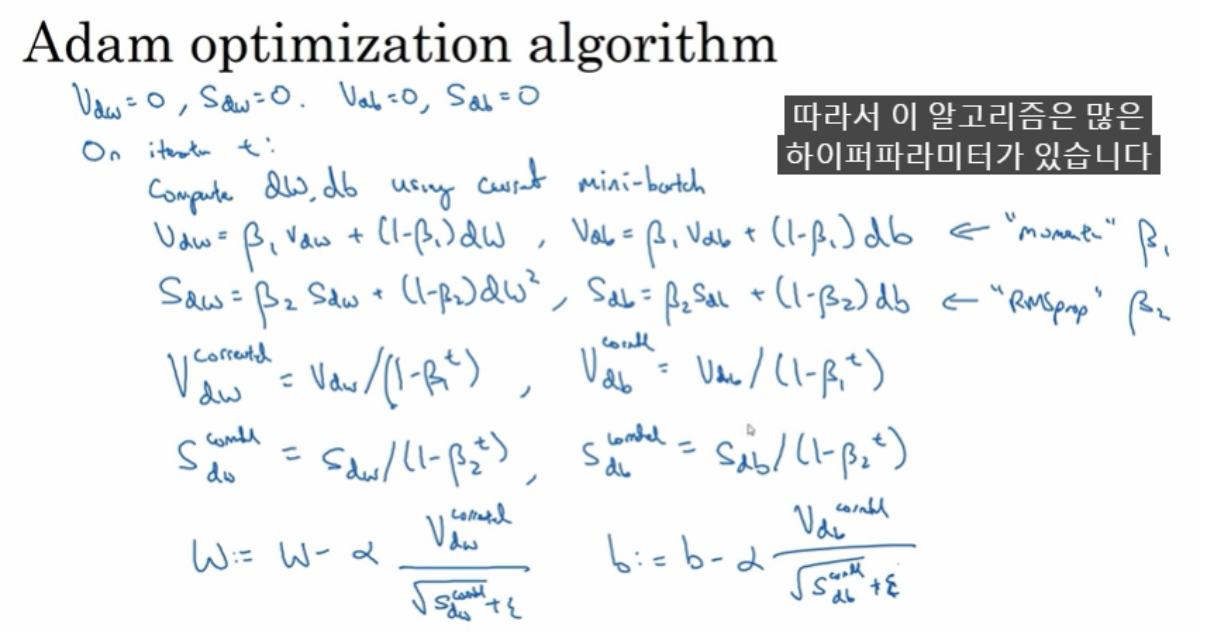
지수가중 평균을 유지하기 위해 새로운 표기법 s_dw을 사용할 것이다

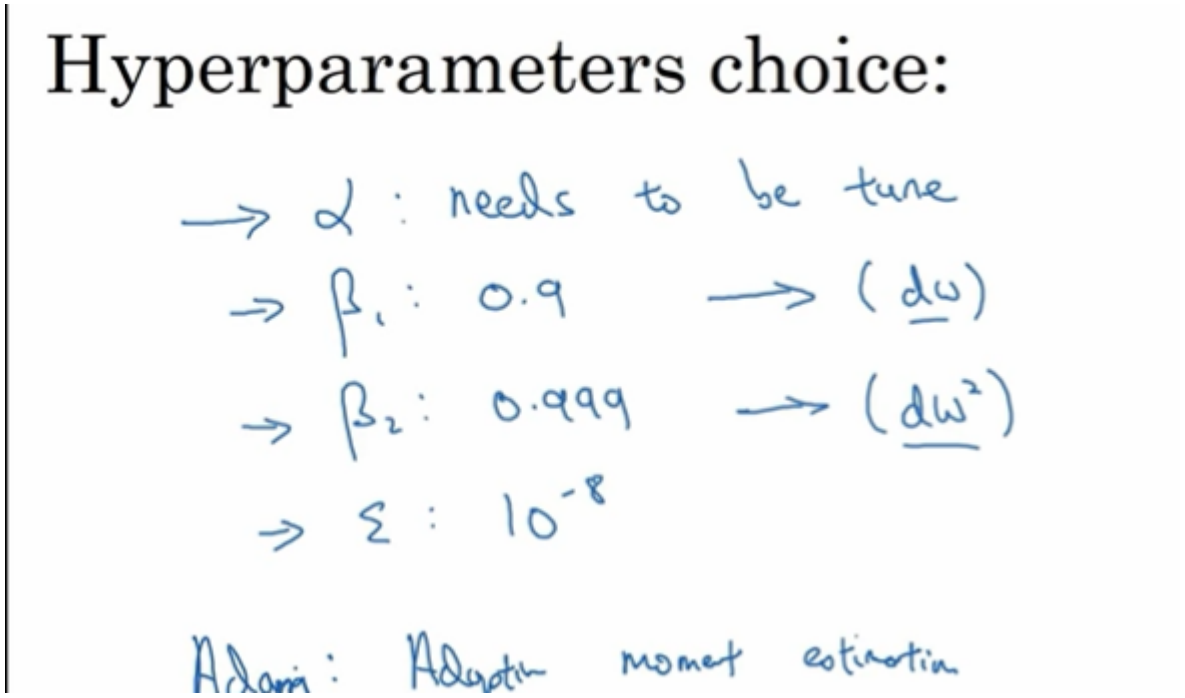
⇒ 도함수의 제곱을 지수가중평균하는 것이다. 비슷하게 s_db 도 사용할 것이다.

매개변수 업데이트 식은 아래와 같다.



Adam 최적화 알고리즘

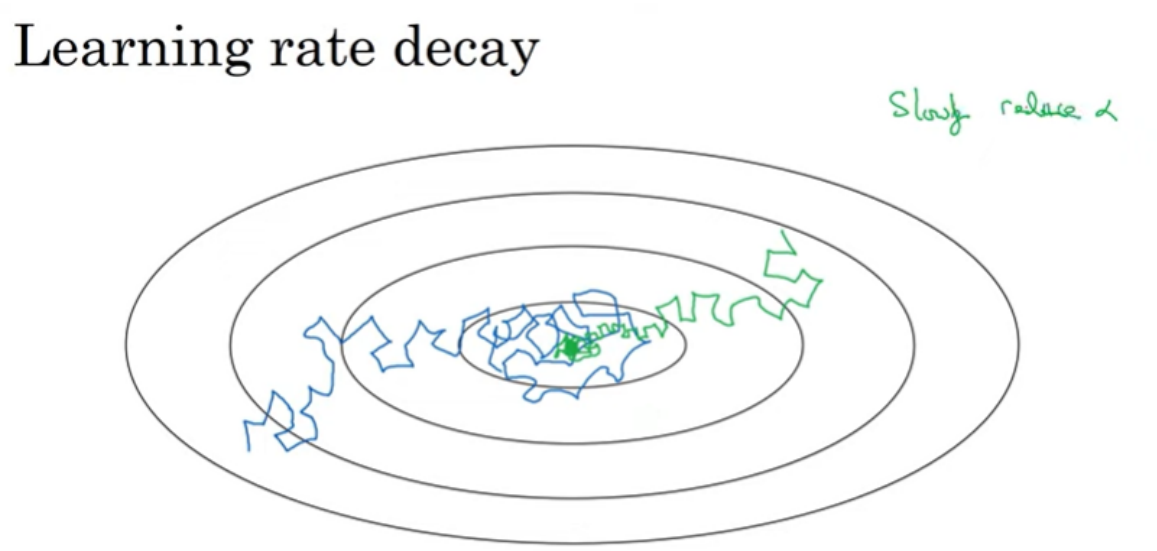




학습률 감쇠

왜 학습률 감쇠가 필요한가

- 작은 미니 배치 일수록 minimum 에서 배회할 가능성이 높기 때문이다.



학습률 감쇠 방법

- α=1+(*decay rate* ×*epoch num*1)α0
- $\alpha = 0.95^{epoch\ num}\alpha_0$ (exponential decay 라고 부른다.)

- $k/\sqrt{epochnum})\alpha_0$
- $k/\sqrt{batchnum})\alpha_0$
- step 별로 α 다르게 설정