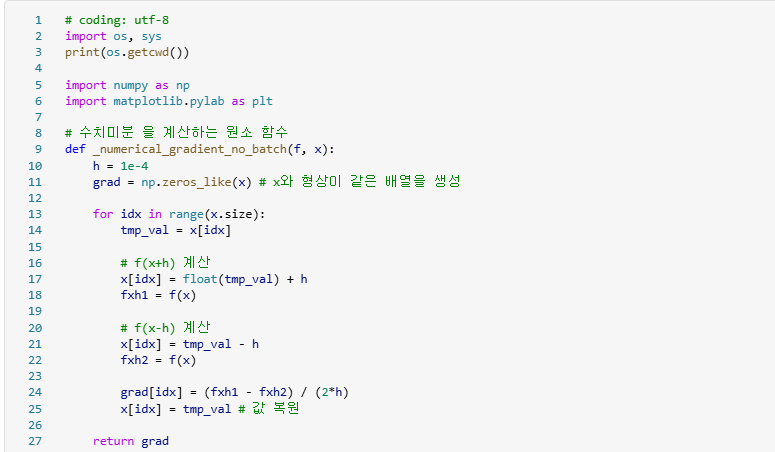
# <오전>

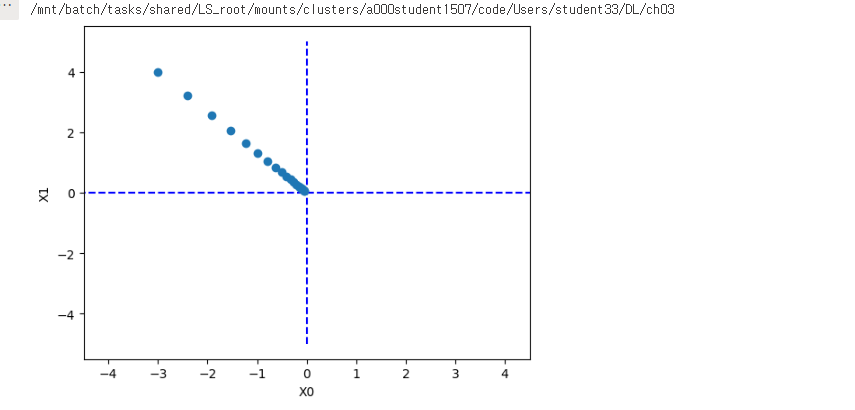
## 신경망 학습

### f(x0, x1) = x02 + x12의 Gradient descent 과정을 2차원 평면에 표현하기









### 정규분포로 초기화된 가중치(2x3)에

### Cross entropy error 손실함수를 써서 새로운 가중치 산출

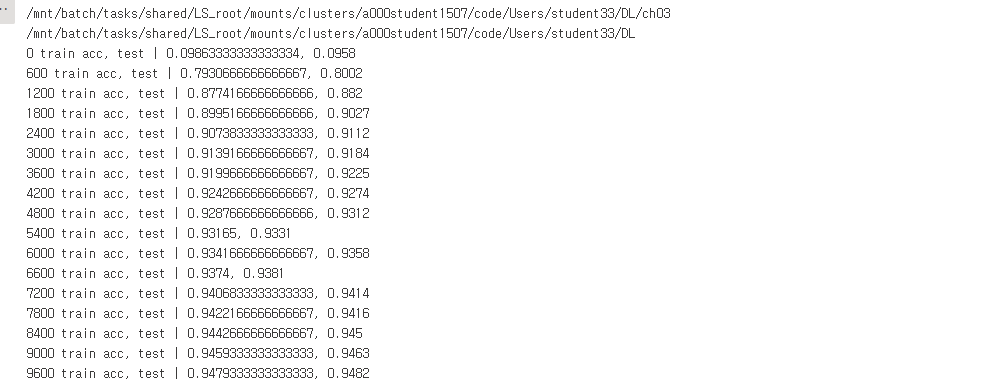


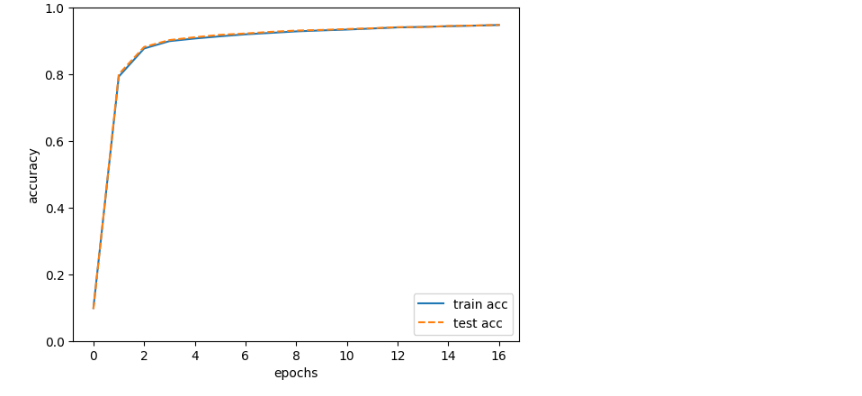
### MNIST 숫자 데이터를 인식하고 검증하는 코드를 구현





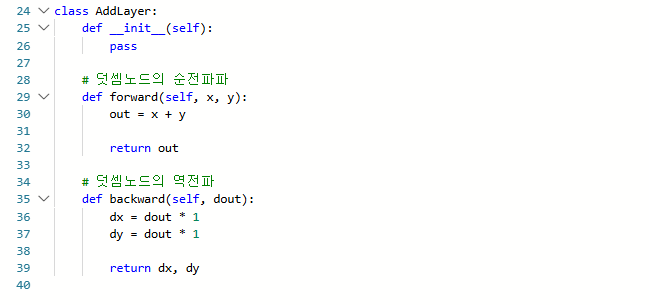
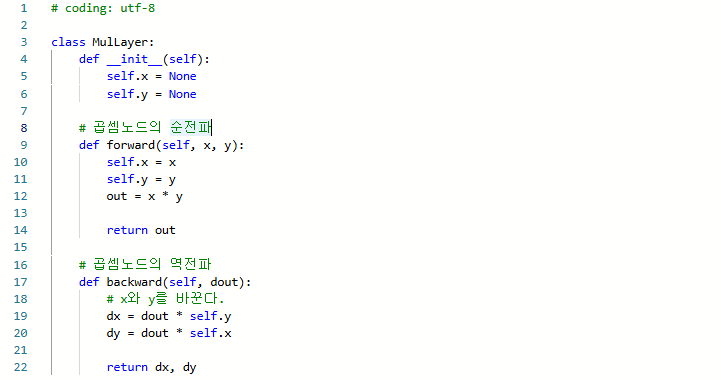






## 오차역전파법

### MulLayer(곱셈노드)와 AddLayer(덧셈노드) 구현



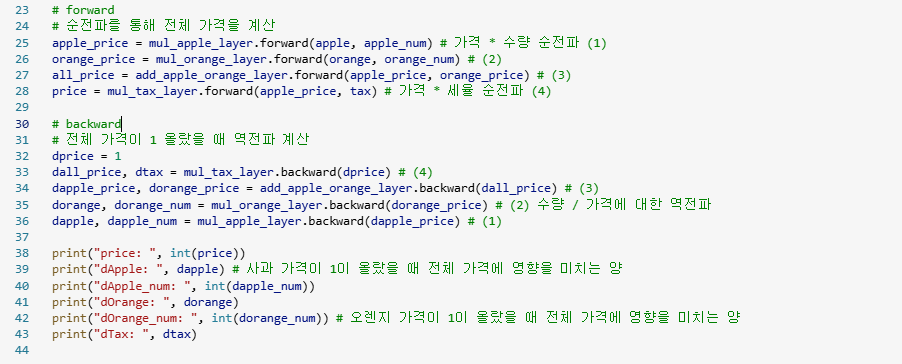
### 사과 구입의 순전파와 역전파 구현

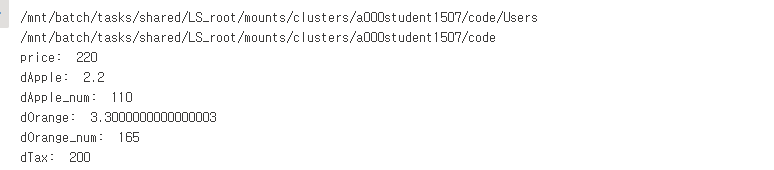


# <오후>

### 사과와 귤 구입의 순전파, 역전파 구현

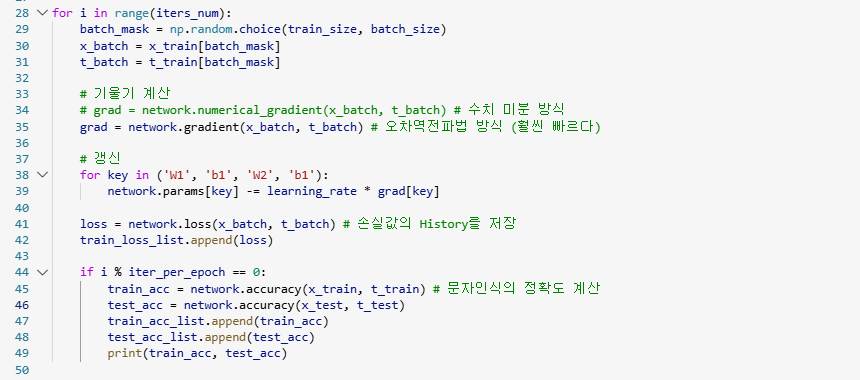






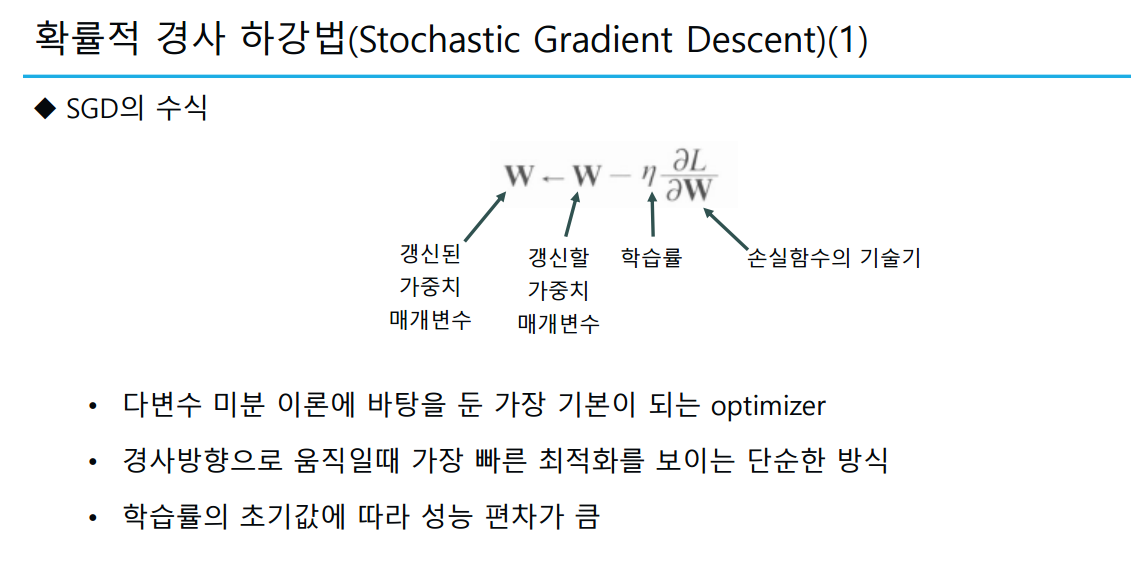
### MNIST 숫자 데이터를 인식하고 검증하는 코드 구현

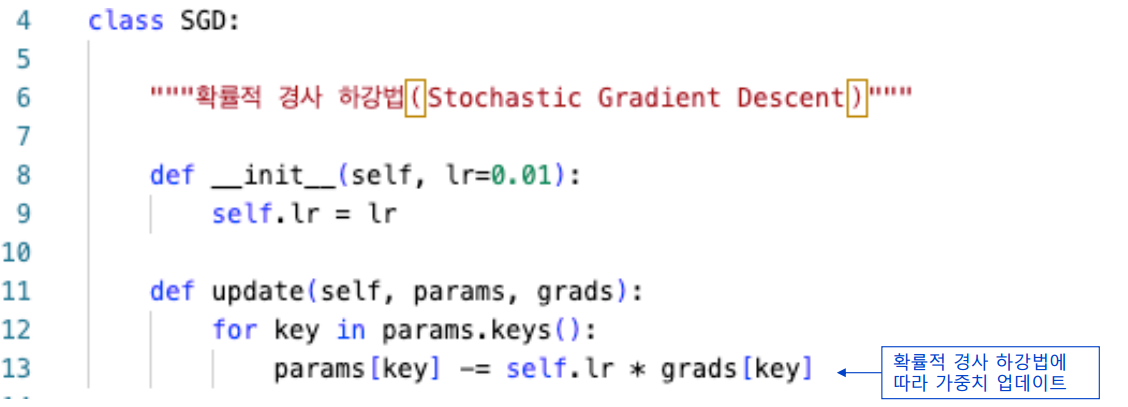




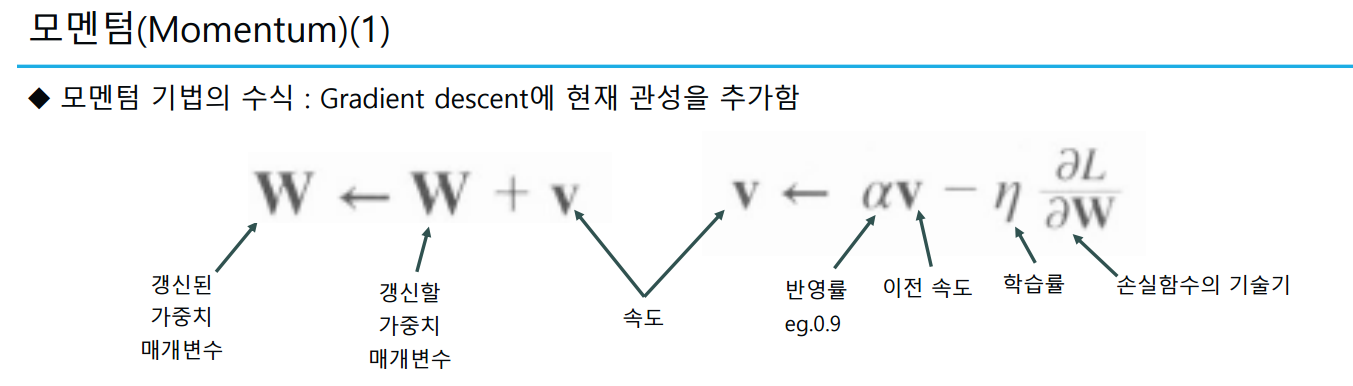
## 학습관련기술들

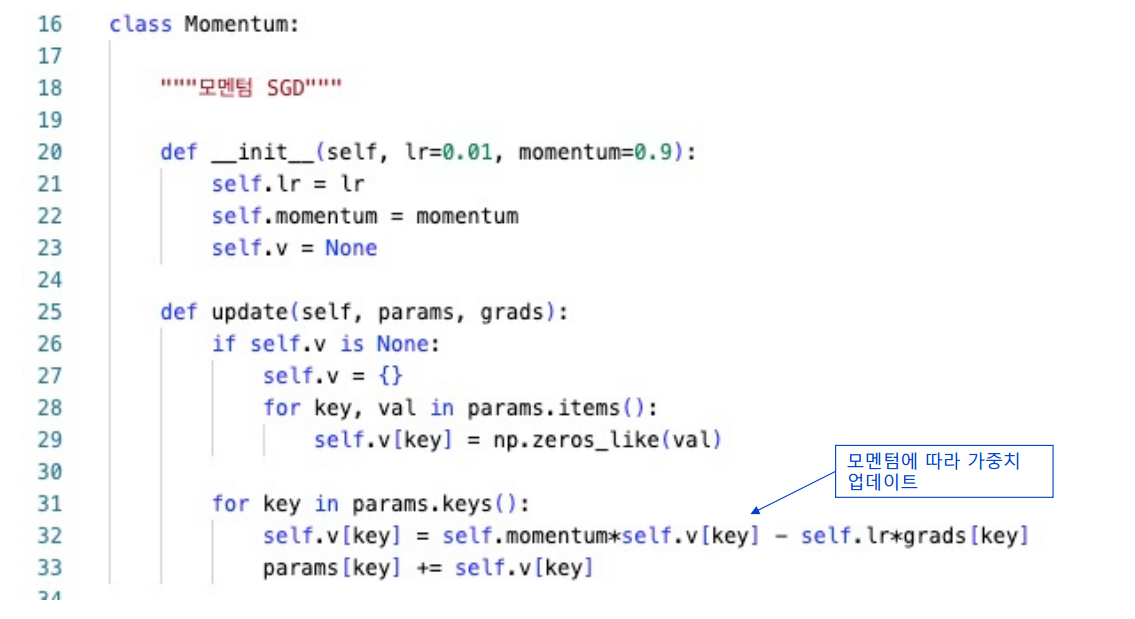
### 확률적 경사 하강법



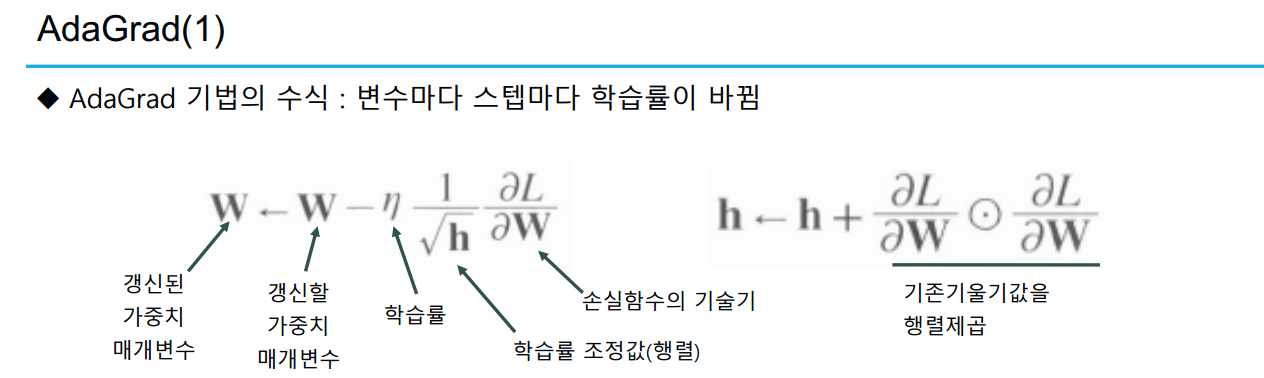


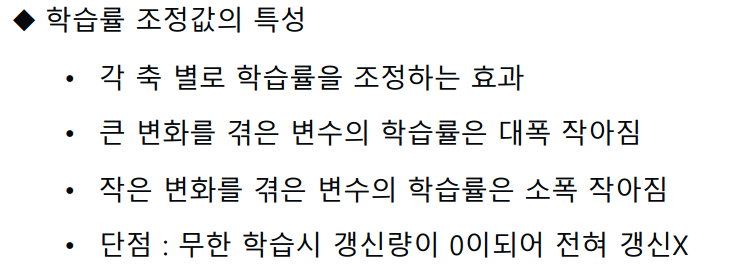
### 모멘텀

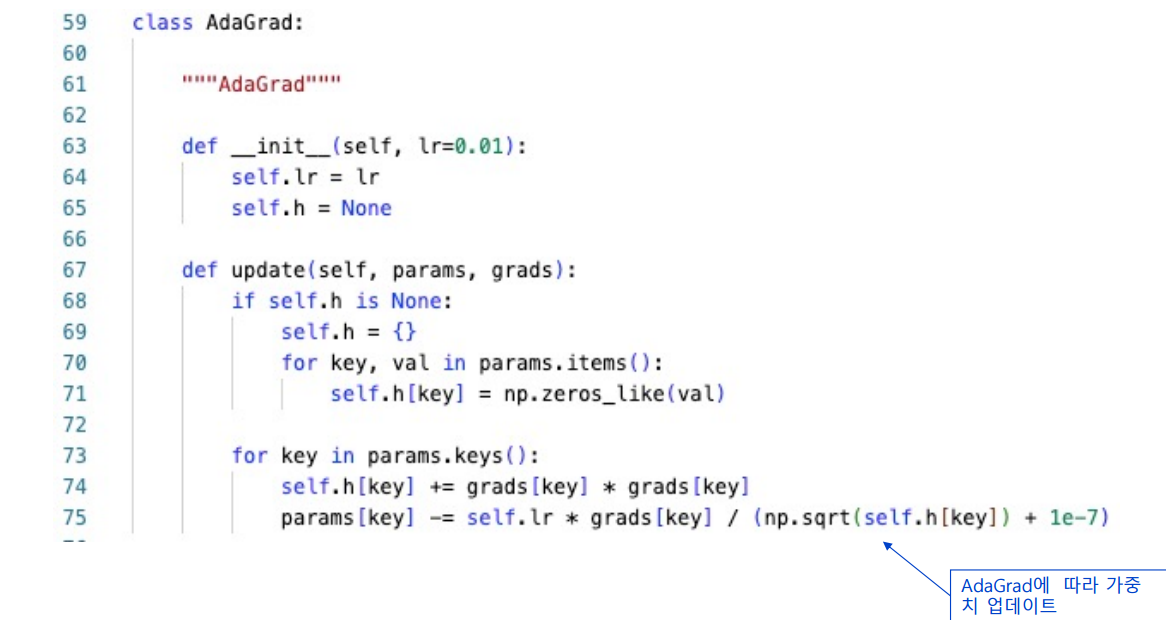




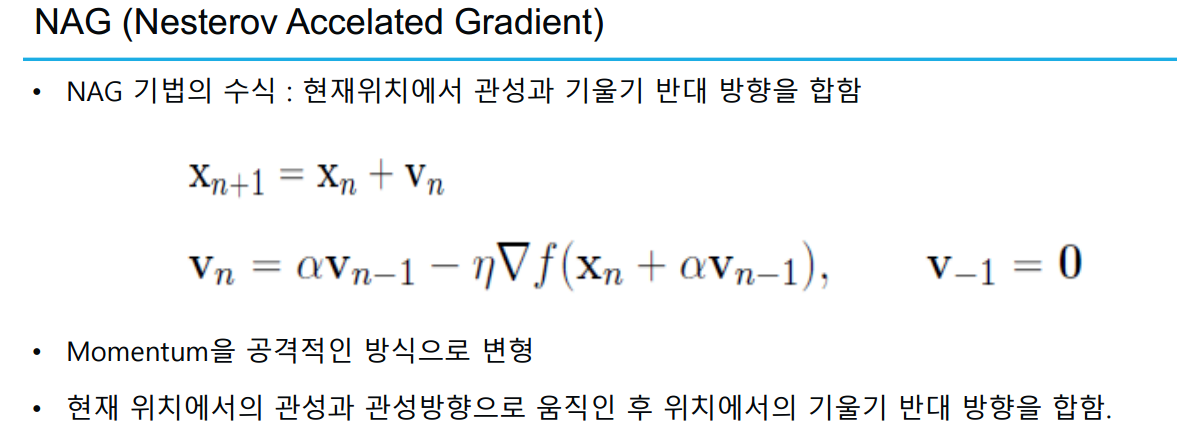
### AdaGrad



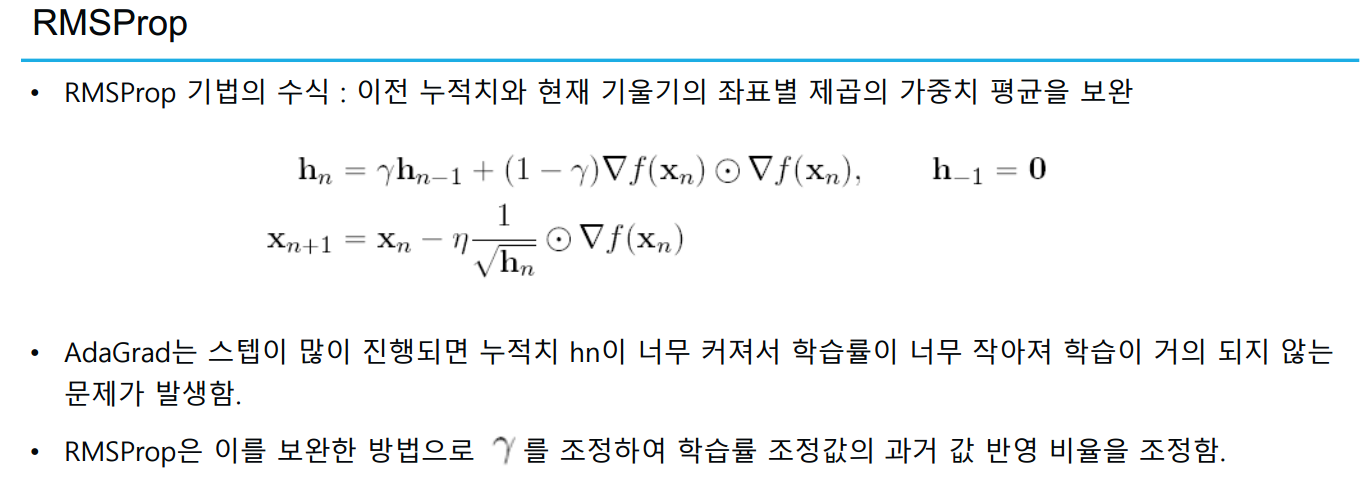




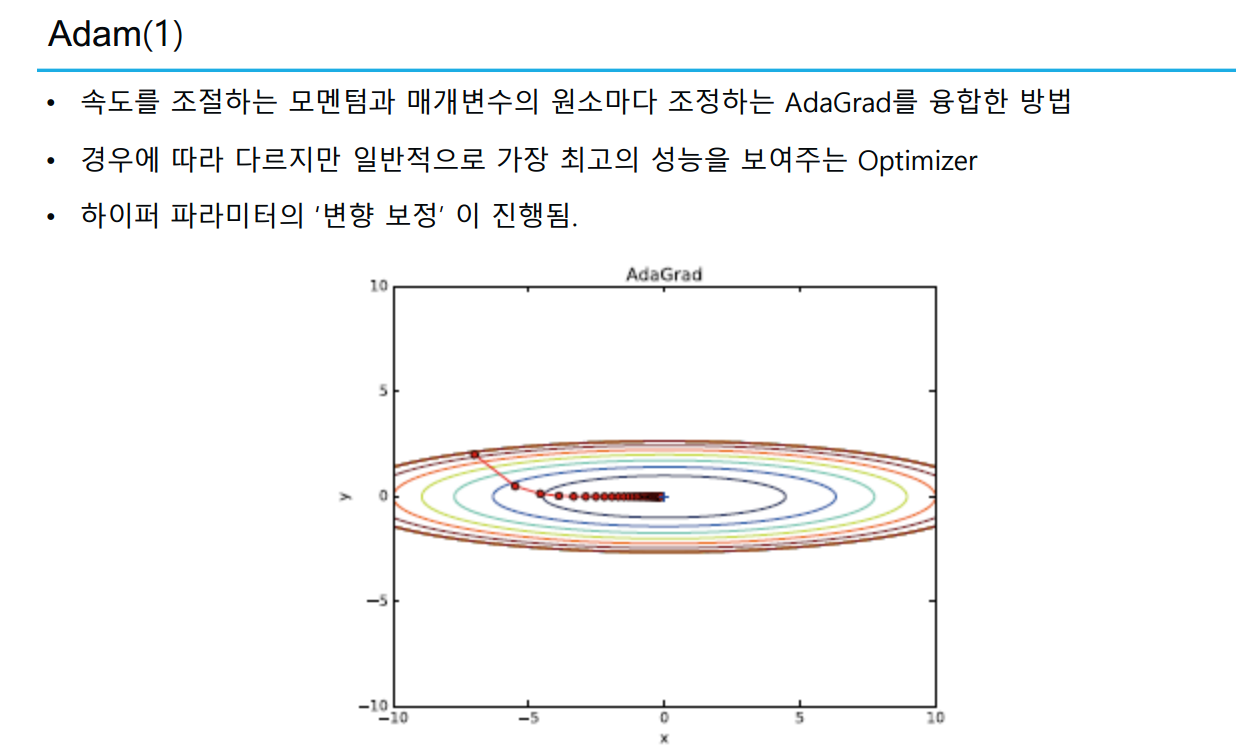
### NGA



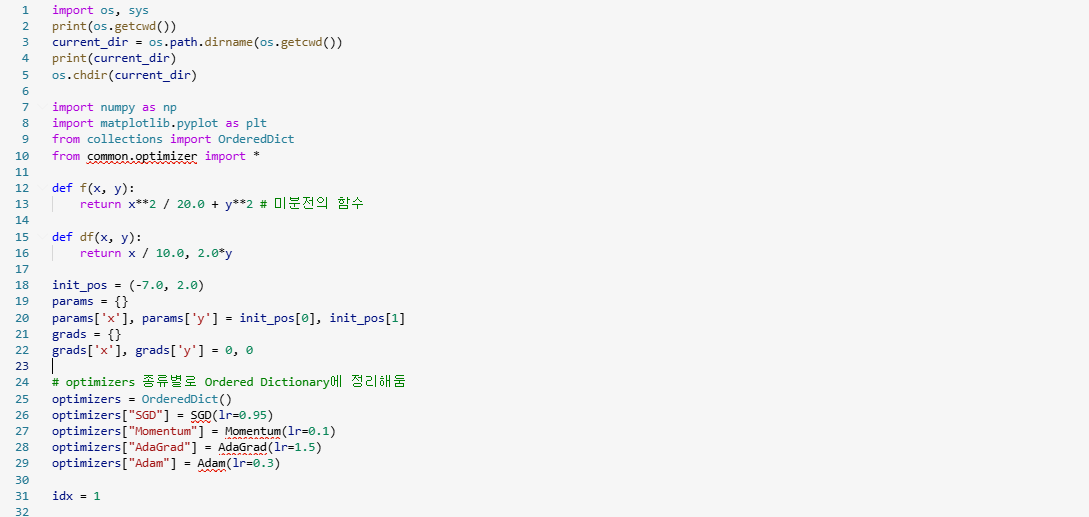
### RMSProp

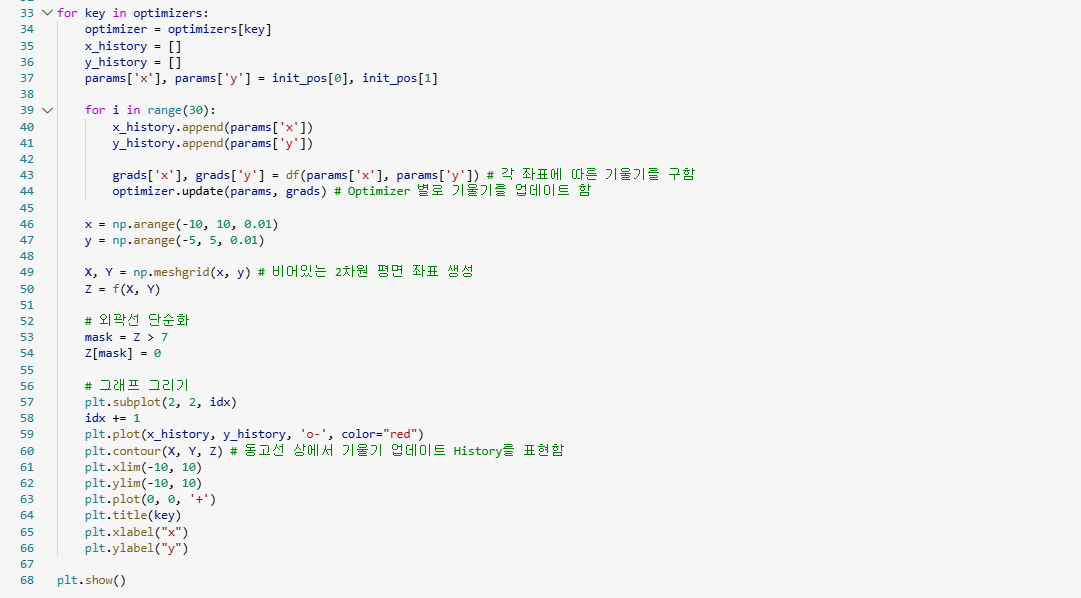


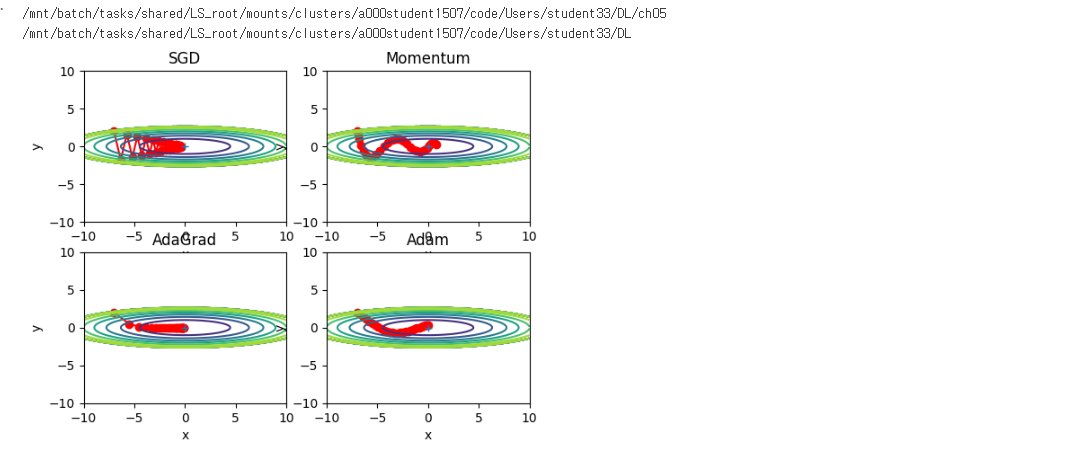
### Adam



### 각 Optimizer를 활용하여 최적화를 수행하는 프로그램 구현







SGD = 시간이 많이 걸림. 진동 횟수가 많음

Momentum = SGD 보다는 낫지만 그래도 조금은 시원찮다.

Adam = 곡선이 그려짐

Adagrad = 가장 이상적임

상황에 따라 달라 최적의 결과를 얻을 수 있도록 이것저것 시도하는 게 좋다.