Machine learning handbook

Optimizer

Taeyang Yang

Update: July 17, 2018

BCILAB, UNIST

1

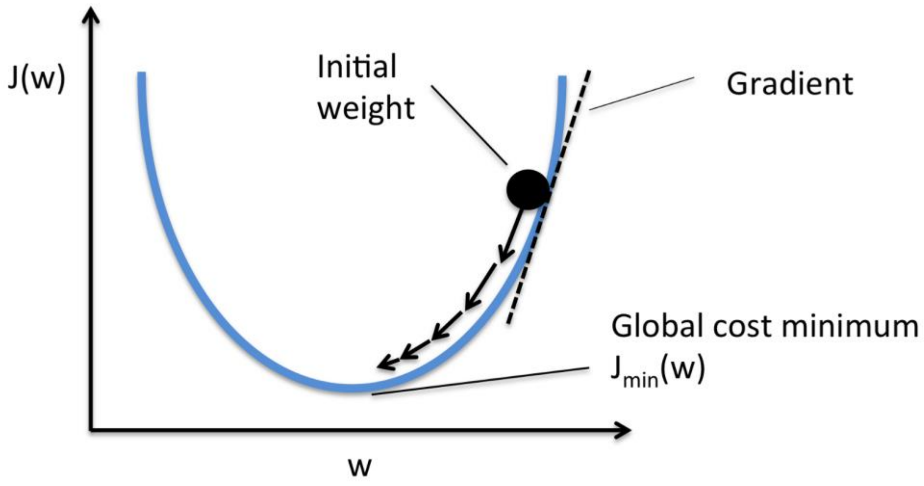
Gradient descent algorithm

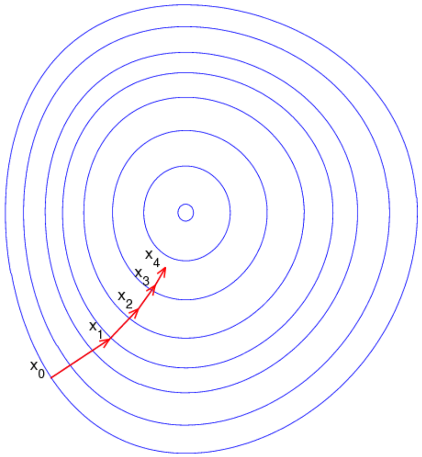
- Gradient Descent (GD)
- Full-batch Gradient Descent
- Stochastic Gradient Descent (SGD)
- Mini-batch (Stochastic) Gradient Descent

Gradient Descent (GD): algorithm

- Update rule

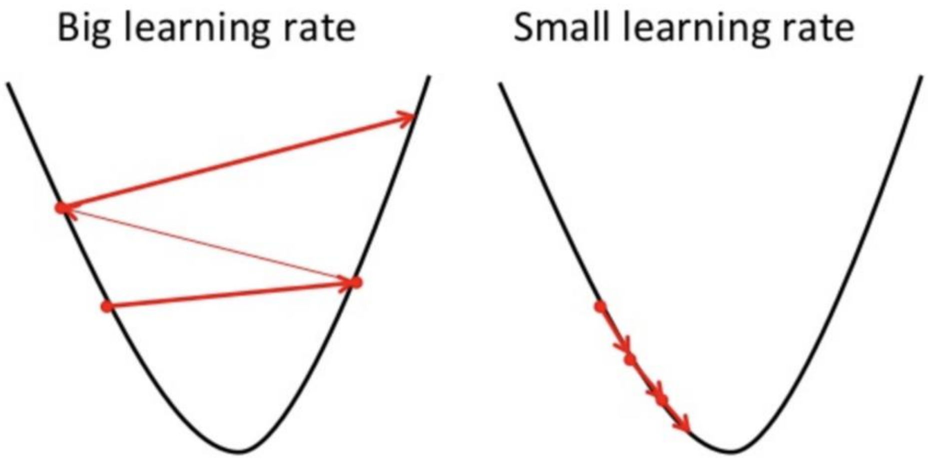
$$\theta = \theta - \eta * \frac{\partial}{\partial \theta} J(\theta)$$



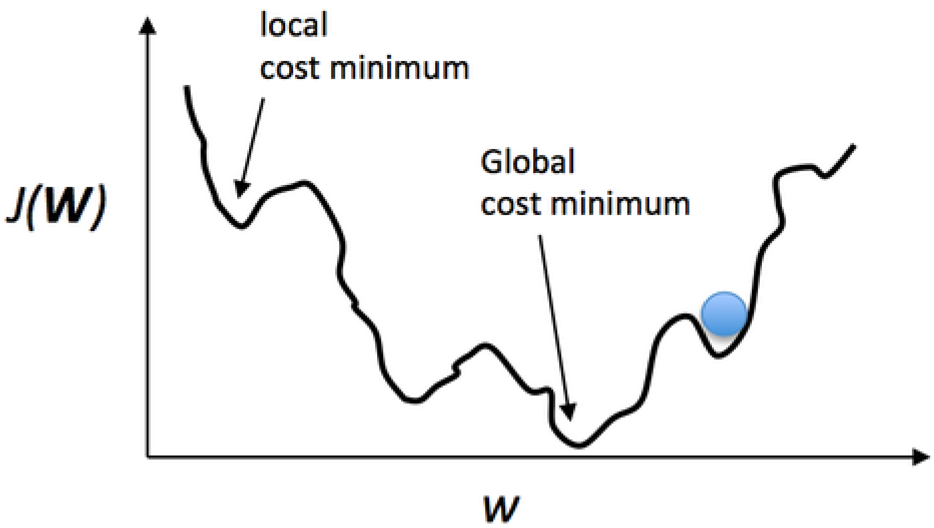


Gradient Descent (GD): Learning rate

$$\theta = \theta - \eta * \frac{\partial}{\partial \theta} J(\theta)$$

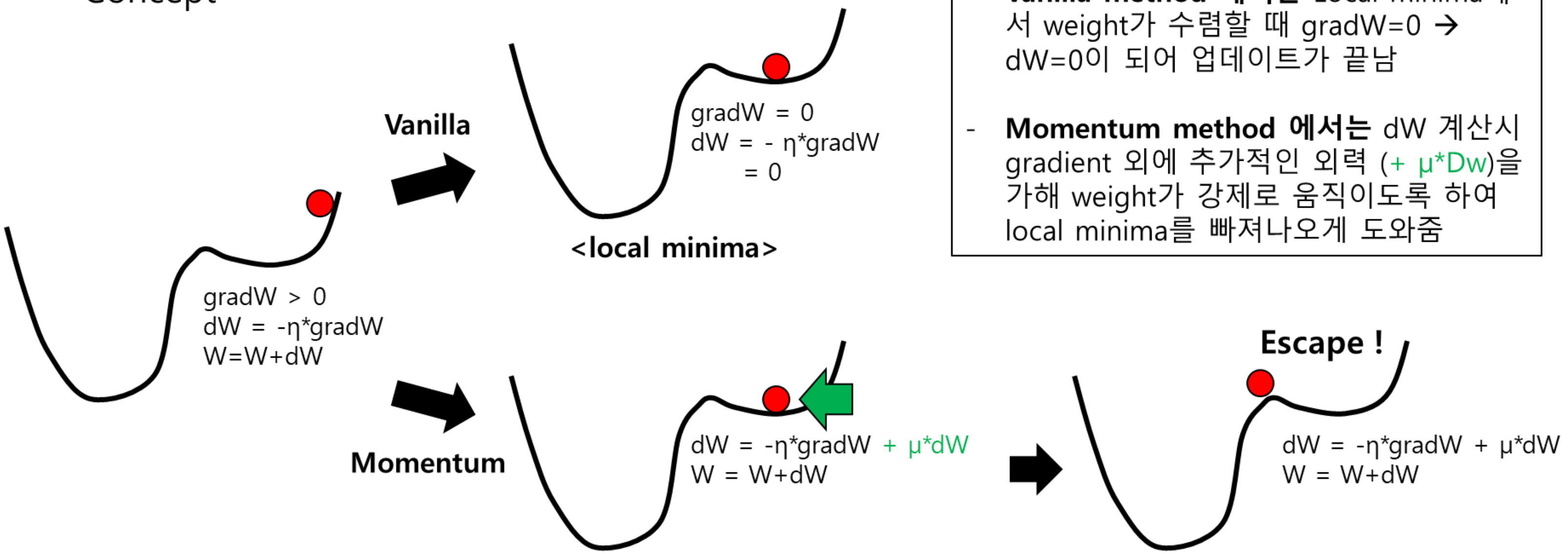


Gradient Descent (GD): Local minimum problem



Gradient Descent (GD): Momentum

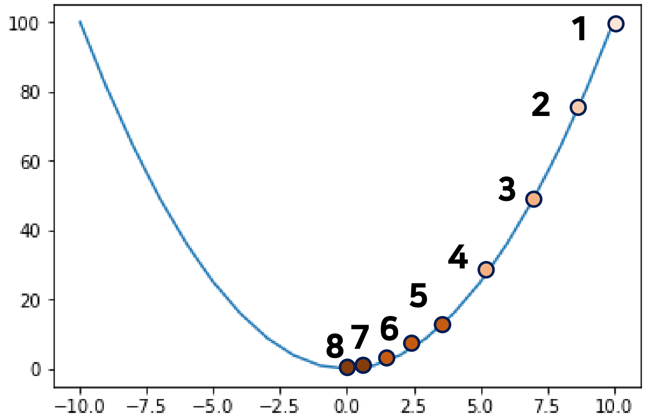
- Concept



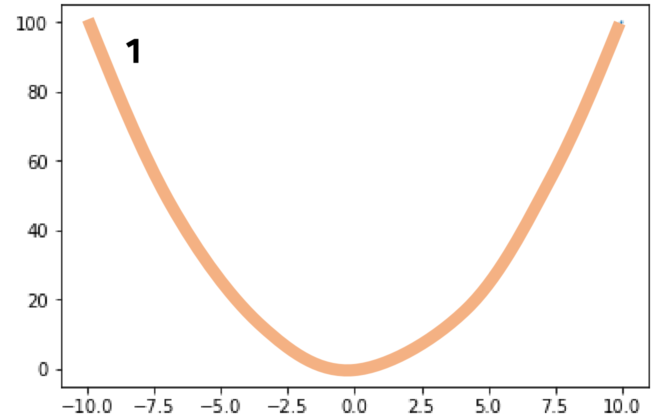
- **Vanilla method 에서는** Local minima에서 weight가 수렴할 때 gradW=0 → dW=0이 되어 업데이트가 끝남
- **Momentum method 에서는** dW 계산시 gradient 외에 추가적인 외력 (+ μ*Dw)을 가해 weight가 강제로 움직이도록 하여 local minima를 빠져나오게 도와줌

Comparison

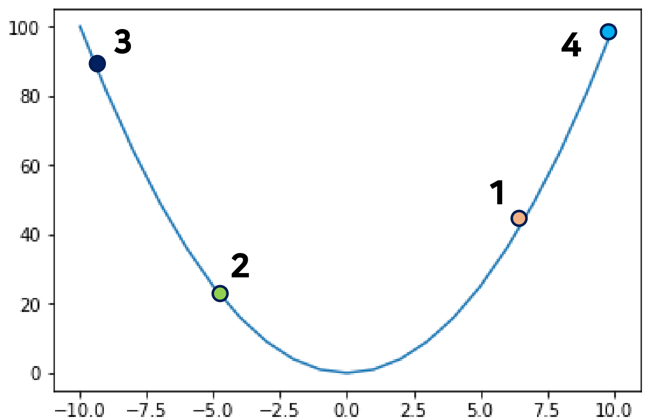
Gradient descent



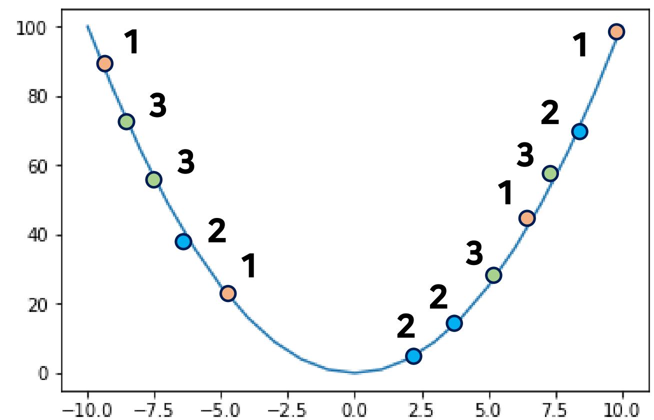
Full-batch gradient descent



Stochastic gradient descent



Mini-batch gradient descent



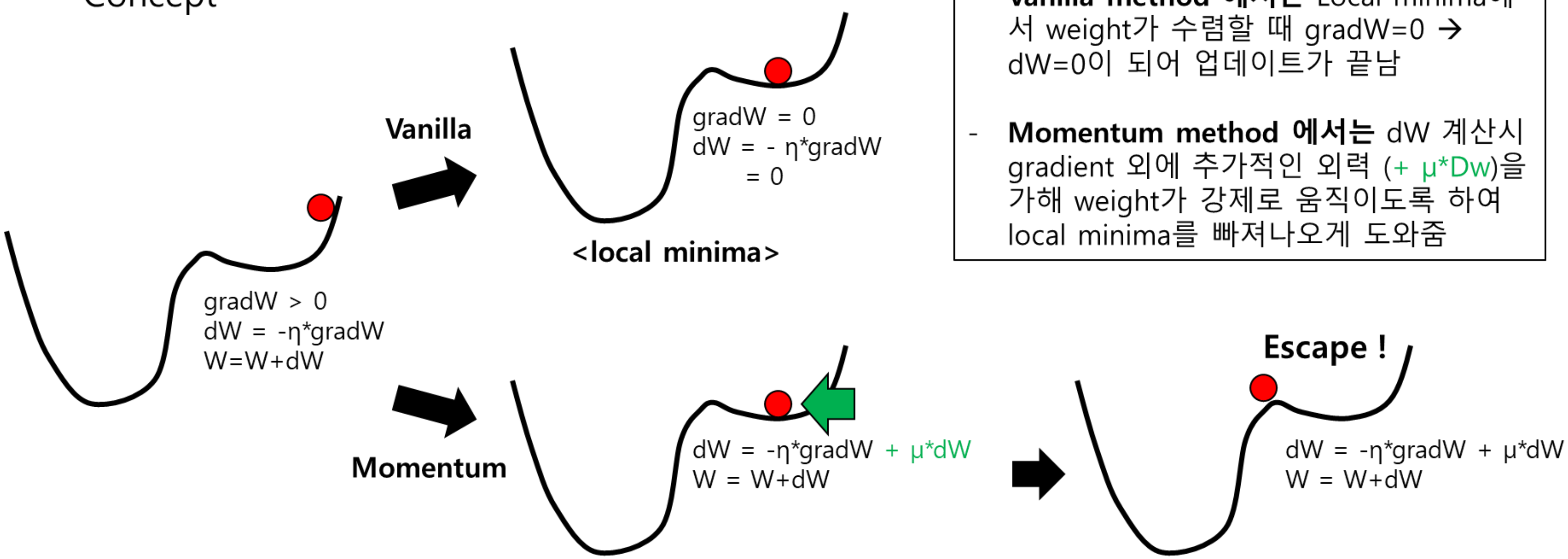
2

Further materials

- Momentum
- Regularization (Weight decay)

Momentum

- Concept



- **Vanilla method 에서는** Local minima에서 weight가 수렴할 때 gradW=0 → dW=0이 되어 업데이트가 끝남
- **Momentum method 에서는** dW 계산시 gradient 외에 추가적인 외력 (+ μ*Dw)을 가해 weight가 강제로 움직이도록 하여 local minima를 빠져나오게 도와줌

Momentum

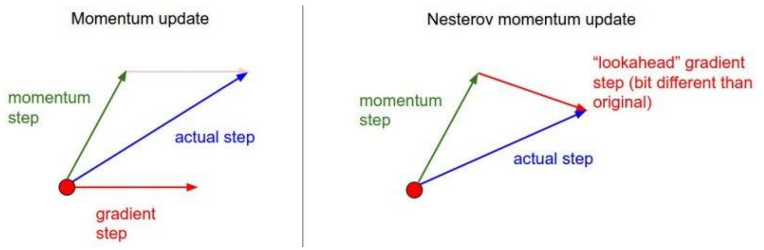
- Parameters update methods
  - **Vanilla update**

$$\Delta Weight_t = -\eta * Gradient_t$$
$$Weight_t = Weight_{t-1} + \Delta Weight_t$$

  - **Momentum update**
    - 직전 w 업데이트 ($\Delta Weight_{t-1}$) * momentum(μ)를 현재 w 업데이트에 추가

$$\Delta Weight_t = -\eta * Gradient_t + \mu * \Delta Weight_{t-1}$$
$$Weight_t = Weight_{t-1} + \Delta Weight_t$$

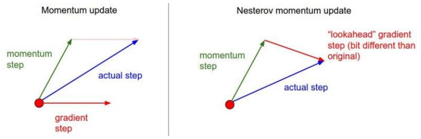
  - **Nesterov momentum update**
    - Momentum update와 비슷하지만, momentum를 더한 위치에서 Gradient 계산



*Bengio, Y., Boulanger-Lewandowski, N., & Pascanu, R. (2013, May). **Advances in optimizing recurrent networks.** In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on (pp. 8624-8628). IEEE.*

***Ilya Sutskever's thesis**, http://www.cs.utoronto.ca/~ilya/pubs/ilya_sutskever_phd_thesis.pdf*

Momentum: Nesterov momentum update



- Parameters update methods
  - **Nesterov momentum update**
    - Momentum update와 비슷하지만, momentum를 더한 위치에서 Gradient 계산

$$Weight\_ahead_t = Weight_t + \mu * \Delta Weight_{t-1}$$

- $Weight\_ahead_t$ 이용해서 $Gradient\_ahead_t$ 계산

$$\Delta Weight_t = -\eta * Gradient_{ahead_t} + \mu * \Delta Weight_{t-1}$$
$$Weight_t = Weight_{t-1} + \Delta Weight_t$$

- 정리된 표현

$$\Delta Weight_t = -\eta * Gradient_t + \mu * \Delta Weight_{t-1}$$
$$Weight_t = Weight_{t-1} + \Delta Weight_t + \mu * \Delta Weight_t - \mu * \Delta Weight_{t-1}$$
$$= Weight_{t-1} + (1 + \mu) * \Delta Weight_t - \mu * \Delta Weight_{t-1}$$

Regularization (Weight decay)

- Weight 들의 absolute 값 총합을 줄이기 위한 방법 (극단적인 weight 값 방지)

- Weight 각각의 일정 비율 ($\lambda, lambda$) 을 곱해준 값을 gradient 값에서 빼준다.
- 이 때, learning rate 곱하기 전에 빼주기 !
- 매우 작은 값을 사용한다 (RBM에서는 0.0002)

- 예시 1: Vanilla update에서,

$$\Delta\theta = \eta(\frac{\partial E}{\partial\theta} - \lambda\theta)$$

- 예시 2: Momentum update에서,

$$\Delta\theta = \eta\left(\frac{\partial E}{\partial\theta} - \lambda\theta\right) + \mu\Delta\theta$$