신경망 최적화 방법

기본 Gradient 방법

$$w_{k+1} = w_k - \mu_k g(w_k) = w_k - v_k$$

Decay

- 스텝 사이즈를 감소

$$\mu_{k+1} = \mu_k \frac{1}{1 + \text{decay}}$$

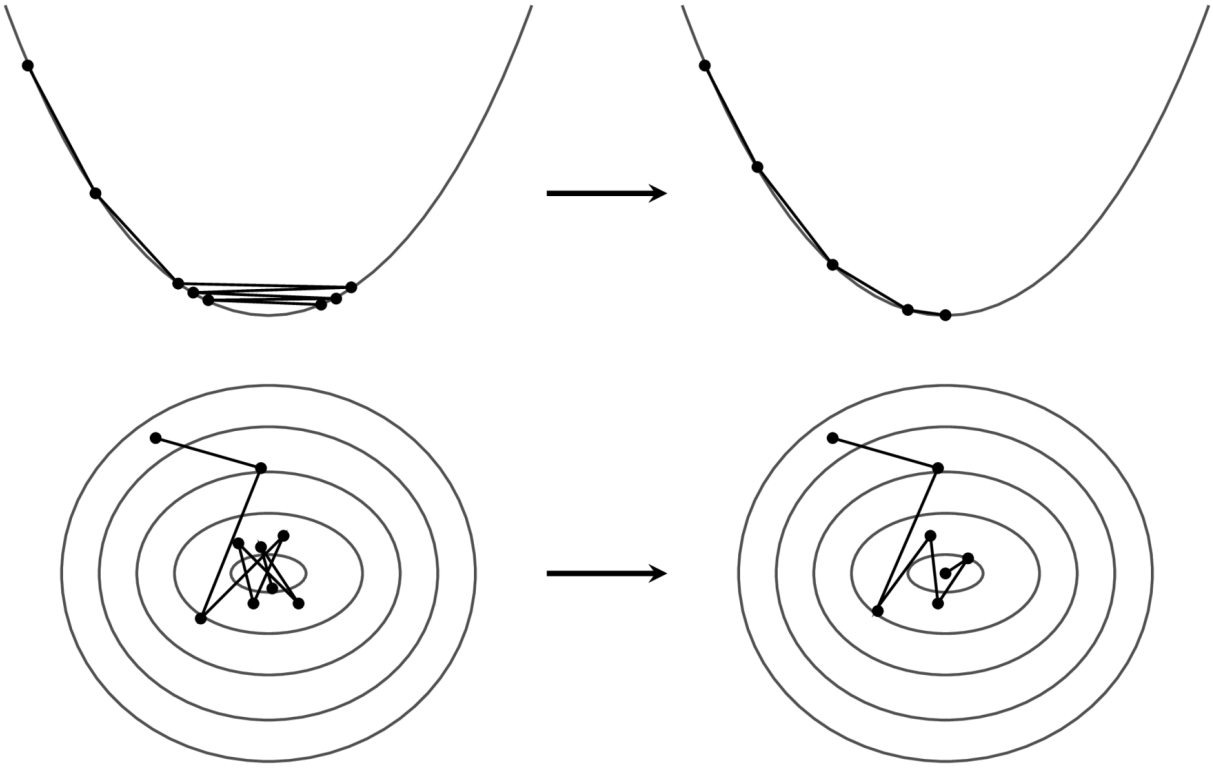

그림 55.4: Decay

Momentum

- 진행하던 방향으로 계속 진행

$$v_{k+1} = \text{momentum} \cdot v_k - \mu_k g(w_k)$$

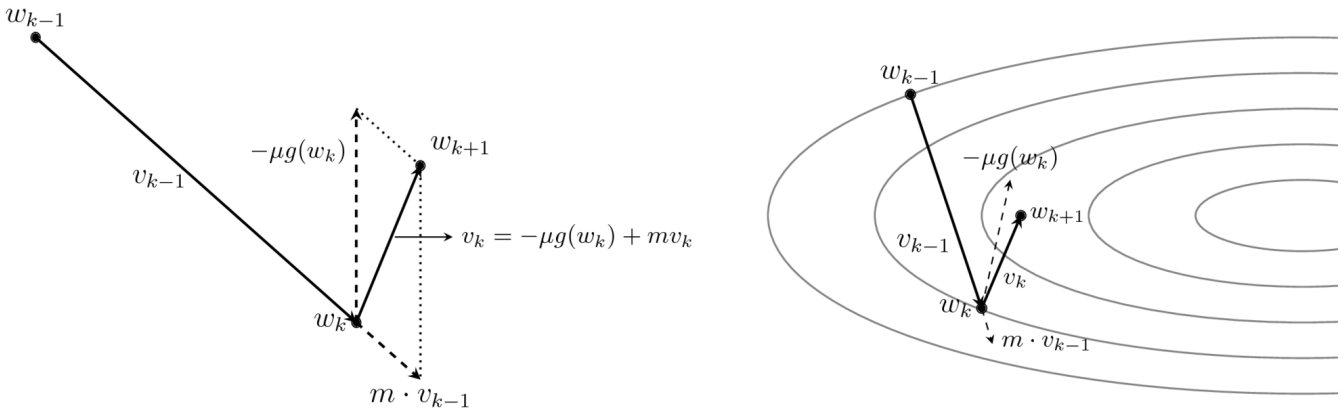

그림 55.5: Momentum

Nesterov momentum

- Momentum 방식으로 이동한 후의 그레디언트를 이용

$$v_{k+1} = \text{momentum} \cdot v_k - \mu_k g(w_k + \text{momentum} \cdot v_k)$$

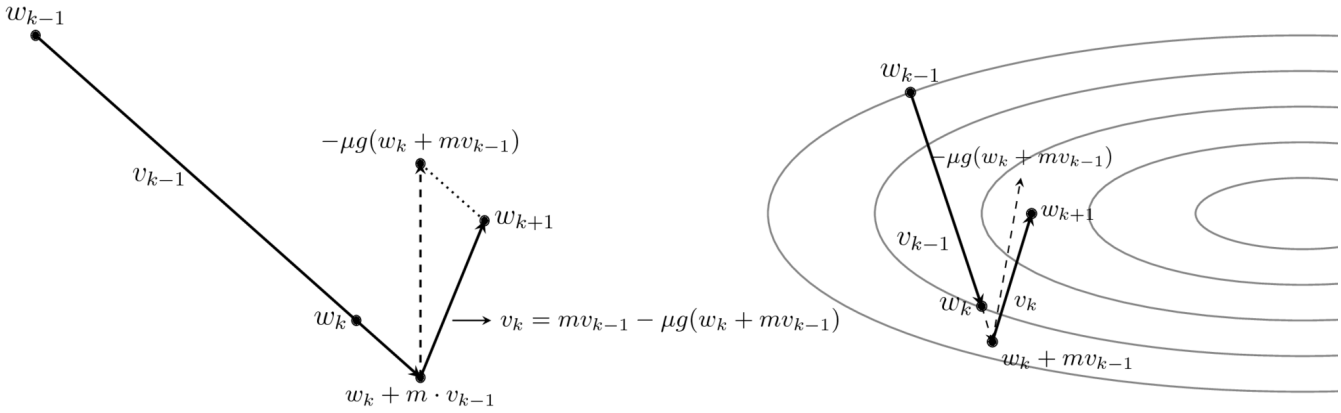

그림 55.6: Nesterov momentum

Adagrad

- **Ada**ptive **grad**ient 방법
- 많이 이동한 가중치는 적게 변화

$$G_{k+1} = G_k + g^2$$

$$w_{k+1} = w_k - \frac{\mu_k}{\sqrt{G_k + \epsilon}} g(w_k)$$

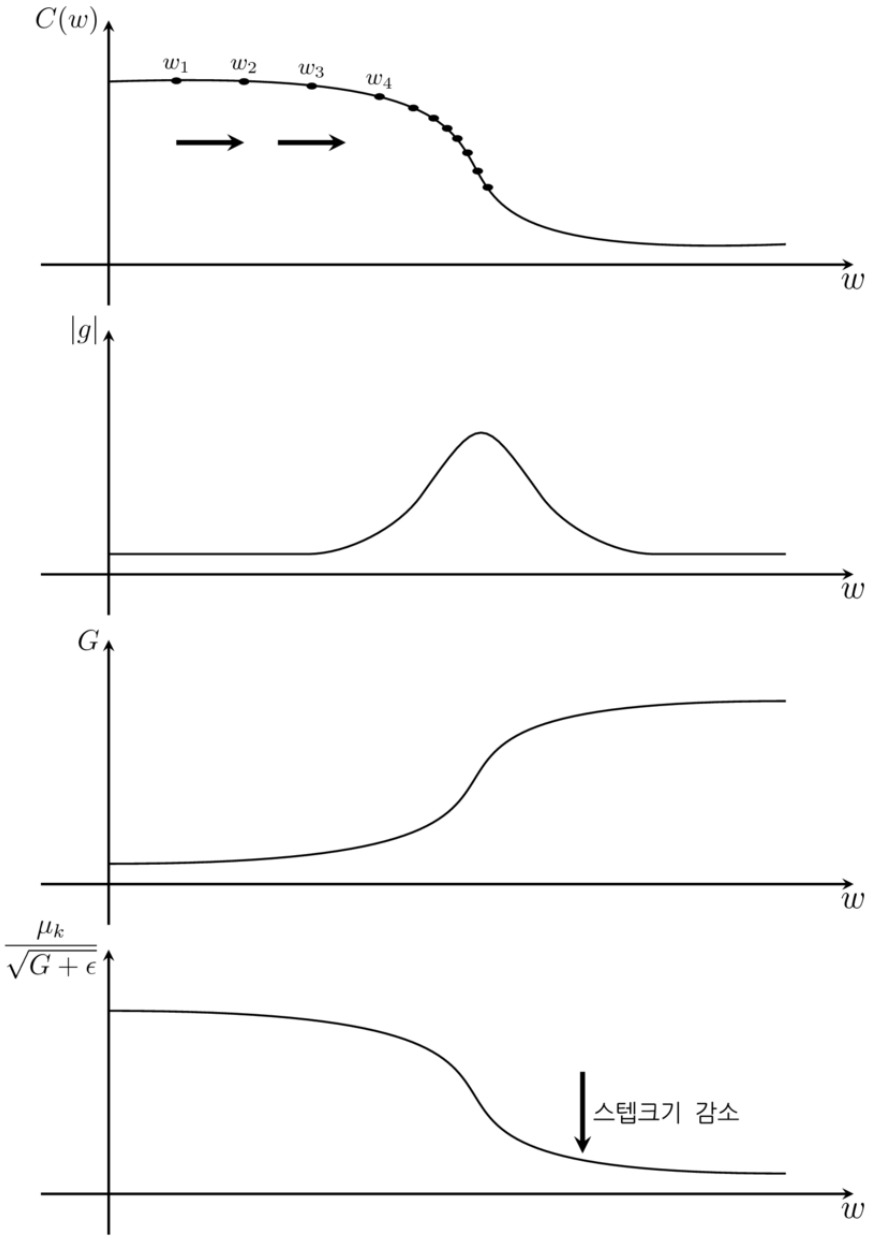

그림 55.7: Adagrad

RMSProp

- 누적 변화를 지수 평균으로 계산

$$G_{k+1} = \gamma G_k + (1-\gamma) g^2$$

$$w_{k+1} = w_k - \frac{\mu_k}{\sqrt{G_k + \epsilon}} g(w_k)$$

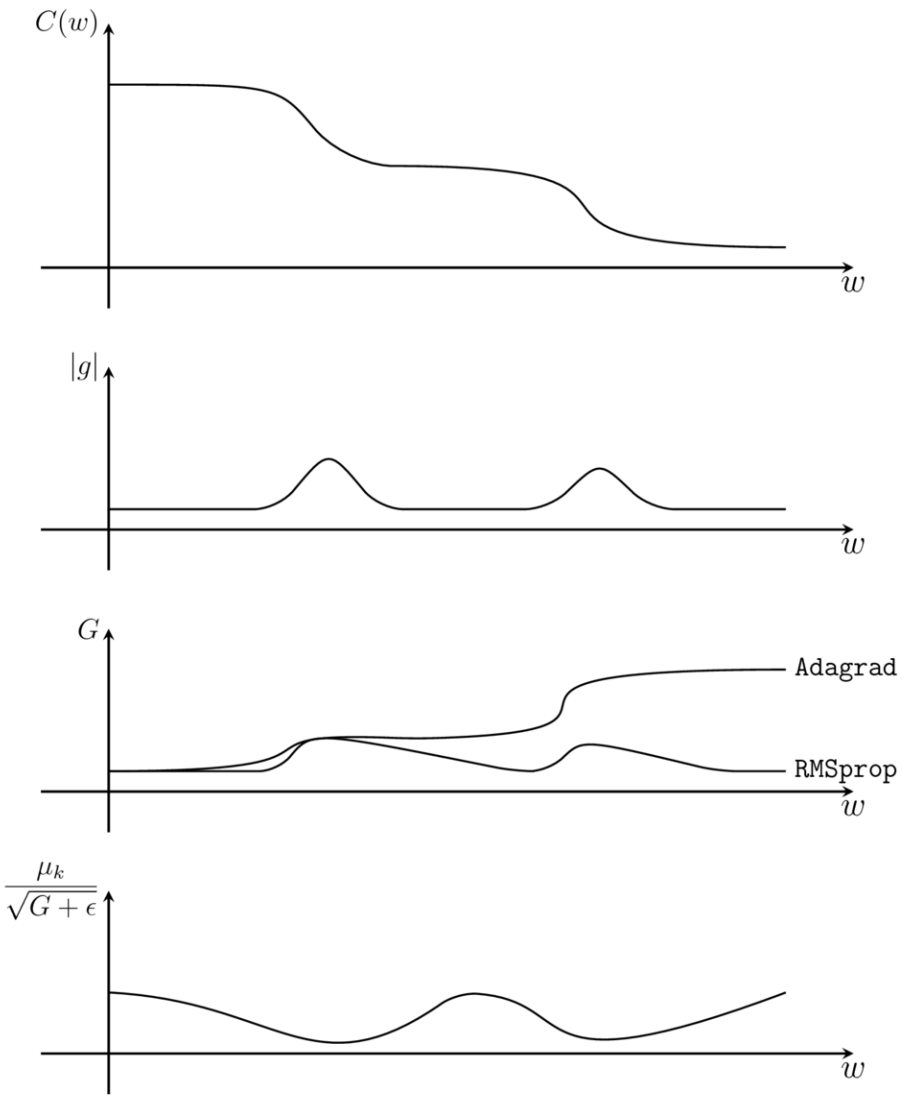

그림 55.8: RMSProp

AdaDelta

- 스텝 사이즈도 가중치의 누적 변화에 따라 감소

$$G_{k+1} = \gamma G_k + (1-\gamma) g^2$$

$$\mu_{k+1} = \gamma \mu_k + (1-\gamma) \Delta_k^2$$

$$\Delta_k = \frac{\sqrt{\mu_k + \epsilon}}{\sqrt{G_k + \epsilon}} g(w_k)$$

$$w_{k+1} = w_k - \Delta_k$$

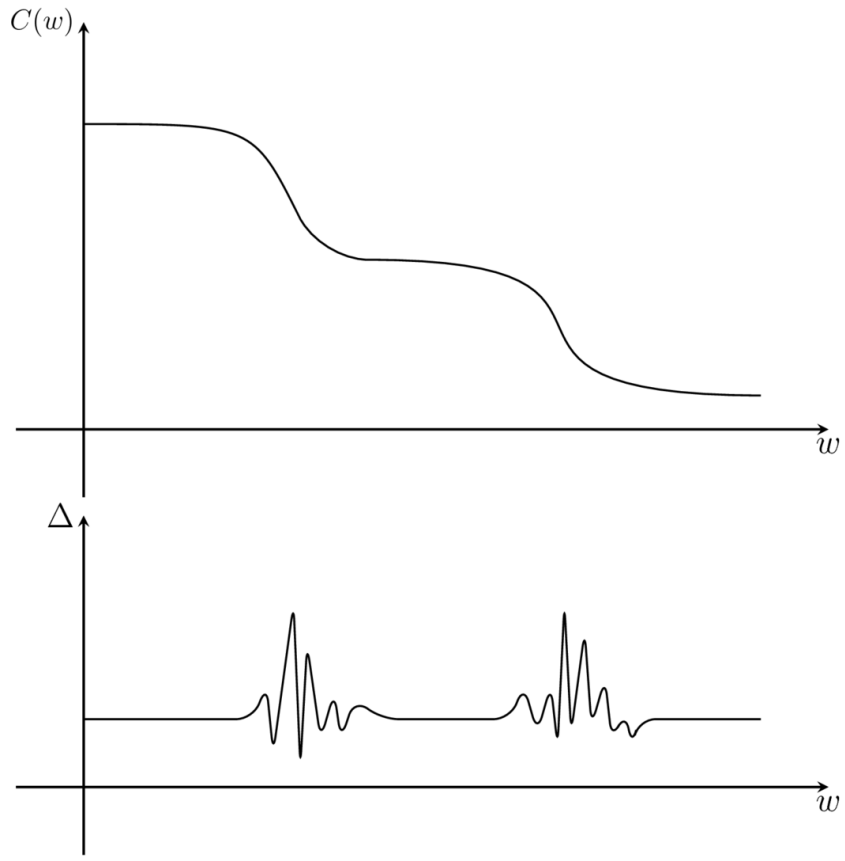

그림 55.9: AdaDelta

Adam

- **Ada**ptive **m**omentum 방법

$$G_{k+1} = \gamma G_k + (1-\gamma) g^2$$

$$v_{k+1} = \gamma_v v_k + (1-\gamma_v) g_k^2$$

$$\hat{G}_k = \frac{G_k}{1-\beta_1}$$

$$\hat{v}_k = \frac{v_k}{1-\beta_2}$$

$$w_{k+1} = w_k - \frac{\mu_k}{\sqrt{\hat{G}_k + \epsilon}} \hat{v}_k$$

참고 자료

- http://www.denizyuret.com/2015/03/alec-radfords-animations-for.html (http://www.denizyuret.com/2015/03/alec-radfords-animations-for.html)