Part. 03

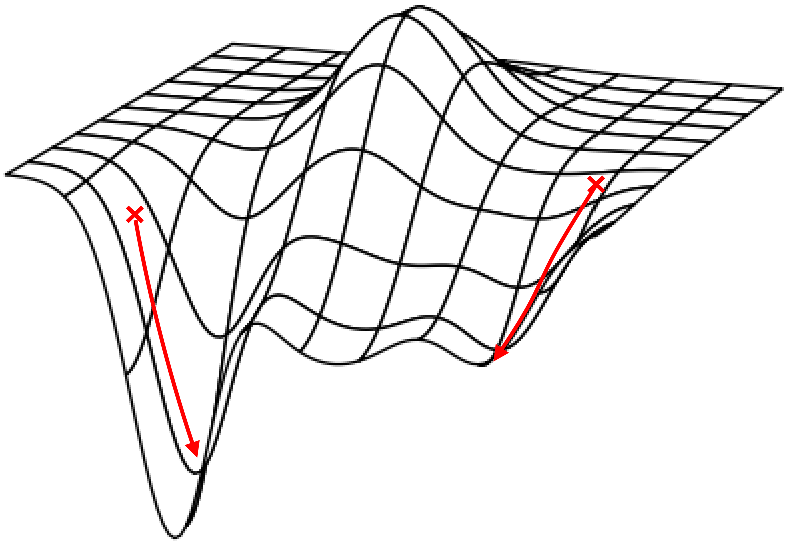
Optimization Algorithms

심화 최적화 알고리즘

FASTCAMPUS
ONLINE

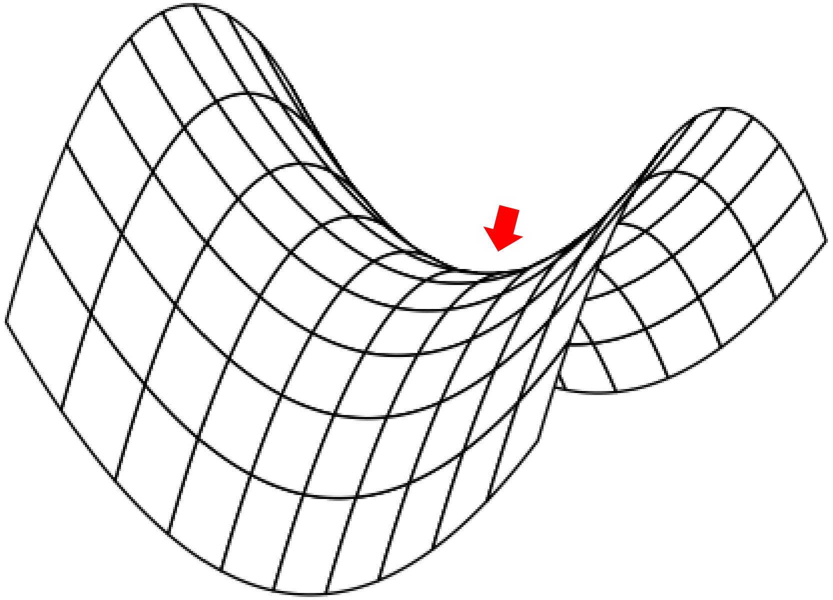
강사. 신제용

I Local Minimum



경사 하강법을 사용할 경우, 초기값에 따라 Local minimum에 빠질 위험이 있다.

Saddle Point

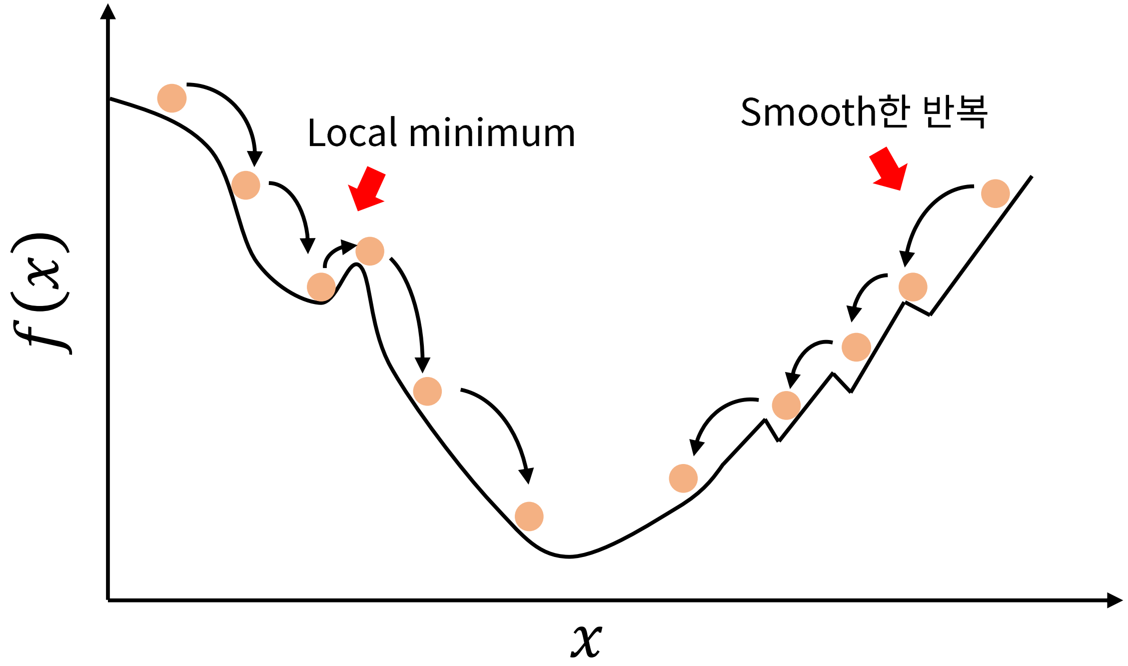


안장점(Saddle point)은 기울기가 0이 되지만 극값이 아닌 지점을 말한다.

경사 하강법은 안장점에서 벗어나지 못한다.

Momentum

관성 (Momentum) : 돌이 굴러 떨어지듯, 이동 벡터를 이용해 이전 기울기에 영향을 받는 알고리즘



$$\boldsymbol{v}_t = \gamma \boldsymbol{v}_{t-1} + \eta \nabla f(\boldsymbol{x}_{t-1})$$

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} - \boldsymbol{v}_t$$

γ : 관성 계수 (momentum term) ≈ 0.9

η : 학습율 (Learning rate)

$\boldsymbol{v}_t$: t번째 step에서의 $\boldsymbol{x}$의 이동 벡터

관성(Momentum)을 이용하면 Local minimum과 잡음에 대처할 수 있다.

이동 벡터($\boldsymbol{v}_t$)를 추가로 사용하므로, 경사 하강법 대비 2배의 메모리를 사용한다.

I AdaGrad

적응적 기울기 (Adaptive gradient; AdaGrad) : 변수별로 학습율이 달라지게 조절하는 알고리즘

Element-wise 제곱

$$\boldsymbol{g}_t = \boldsymbol{g}_{t-1} + (\nabla f(\boldsymbol{x}_{t-1}))^2$$

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} - \frac{\eta}{\sqrt{\boldsymbol{g}_t + \epsilon}} \cdot \nabla f(\boldsymbol{x}_{t-1})$$

Element-wise 곱

$\boldsymbol{g}_t$: t번째 step까지의 기울기의 누적 크기

ϵ : 0으로 나누는 것을 방지하는 작은 값 $\approx 10^{-6}$

기울기가 커서 학습이 많이 된 변수는 학습율을 감소시켜, 다른 변수들이 잘 학습되도록 한다.

$\boldsymbol{g}_t$가 계속해서 커져서 학습이 오래 진행되면 더이상 학습이 이루어지지 않는 단점이 있다.

RMSProp

RMSProp : AdaGrad의 문제점을 개선한 방법으로, 합 대신 지수평균을 사용

$$\boldsymbol{g}_t = \gamma \boldsymbol{g}_{t-1} + (1-\gamma)(\nabla f(\boldsymbol{x}_{t-1}))^2$$

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} - \frac{\eta}{\sqrt{\boldsymbol{g}_t + \epsilon}} \cdot \nabla f(\boldsymbol{x}_{t-1})$$

γ : 지수 평균의 업데이트 계수

변수 간의 상대적인 학습율 차이는 유지하면서 $\boldsymbol{g}_t$가 무한정 커지지 않아 학습을 오래 할 수 있다.

Adam

Adaptive moment estimation (Adam) : RMSProp과 Momentum의 장점을 결합한 알고리즘

$$\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\nabla f(\boldsymbol{x}_{t-1})$$

$$\boldsymbol{g}_t = \beta_2 \boldsymbol{g}_{t-1} + (1-\beta_2)\left(\nabla f(\boldsymbol{x}_{t-1})\right)^2$$

$$\widehat{\boldsymbol{m}}_t = \frac{\boldsymbol{m}_t}{1-\beta_1^t}, \widehat{\boldsymbol{g}}_t = \frac{\boldsymbol{g}_t}{1-\beta_2^t}$$

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} - \frac{\eta}{\sqrt{\widehat{\boldsymbol{g}}_t + \epsilon}} \cdot \widehat{\boldsymbol{m}}_t$$

$\beta_1 \approx 0.9$

$\beta_2 \approx 0.999$

$\epsilon \approx 10^{-8}$

$\widehat{\boldsymbol{m}}, \widehat{\boldsymbol{v}}$: 초기값이 0인 것을 고려하여 보정한 값

Adam 최적화 방법은 가장 최신의 기술(state-of-the-art)이며, 딥러닝에서 가장 많이 사용된다.