

# 딥러닝 입문

기본 용어 이해하기

# 목 차

01 Epoch과 Batch Size

02 Gradient Descent

03 Stochastic Gradient Descent

04 Batch Size

05 과적합(Overfitting)

06 과적합 해결하기

# 01 Epoch과 Batch size

- ▶ 둘 다 정수의 값을 갖는다.
- ▶ epoch의 수는 학습 알고리즘이 전체 학습 데이터 세트에 대해 동작하는 횟수를 정의하는 파라미터
- ▶ 하나의 에포크(epoch)는 하나 이상의 배치로 구성.
- ▶ 배치 크기(Batch size)는 모델의 처리할 샘플 수를 정의하는 변수(파라미터)이다.

## 02 Gradient Descent

- ▶ 경사 하강법(Gradient Descent)은 가장 낮은 곳을 찾아가는 과정.
- ▶ 경사 하강법(Gradient Descent)은 1차 근사값 발견을 위한 최적화 알고리즘이다.
- ▶ 기본 개념은 함수의 기울기(경사)를 구하고 경사의 절대값이 낮은 쪽으로 계속 이동시켜 최소값에 이를때까지 반복시키는 것.

## 02 Gradient Descent

- ▶ 업데이트는 모델에 따라 다르지만 인공 신경망의 경우,  
역전파 업데이트 알고리즘이 사용된다..
- ▶ 배치 사이즈에 따라 다음과 같이 분류된다.

Batch Gradient Descent      Batch Size = Total training set

Stochastic Gradient Descent      mini Batch

# 02 Gradient Descent – Batch Gradient Descent

▶ Batch는 Total training set 을 의미

▶ 장점

- 전체 업데이트가 한번에 이뤄지므로 계산의 횟수가 적다.
- 전체 데이터에 대해 미분값을 계산하여 갱신하므로 안정적으로 수렴.

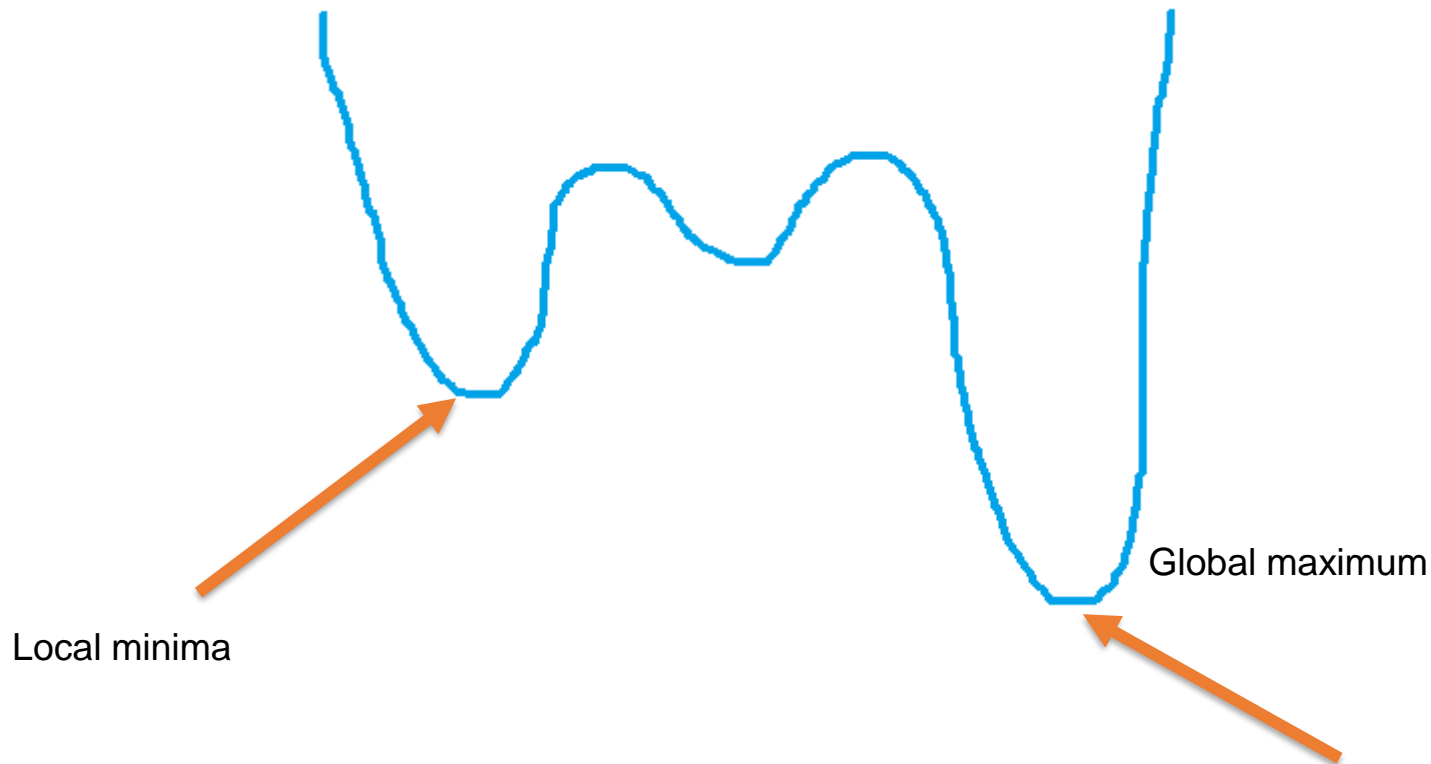
▶ 단점

- 한번의 갱신에 전체 Training set이 사용되므로 학습이 오래 걸림.
- 전체 데이터 셋이 사용되므로 상대적으로 많은 메모리가 요구.

## 02 Gradient Descent

### ▶ 경사 하강법의 문제 - local minima

현재 찾고 싶은 값은 global minima(함수 전체에서의 최소값)이지만, 어떤 경우에는 local minima에 빠져 벗어나지 못하는 현상이 발생

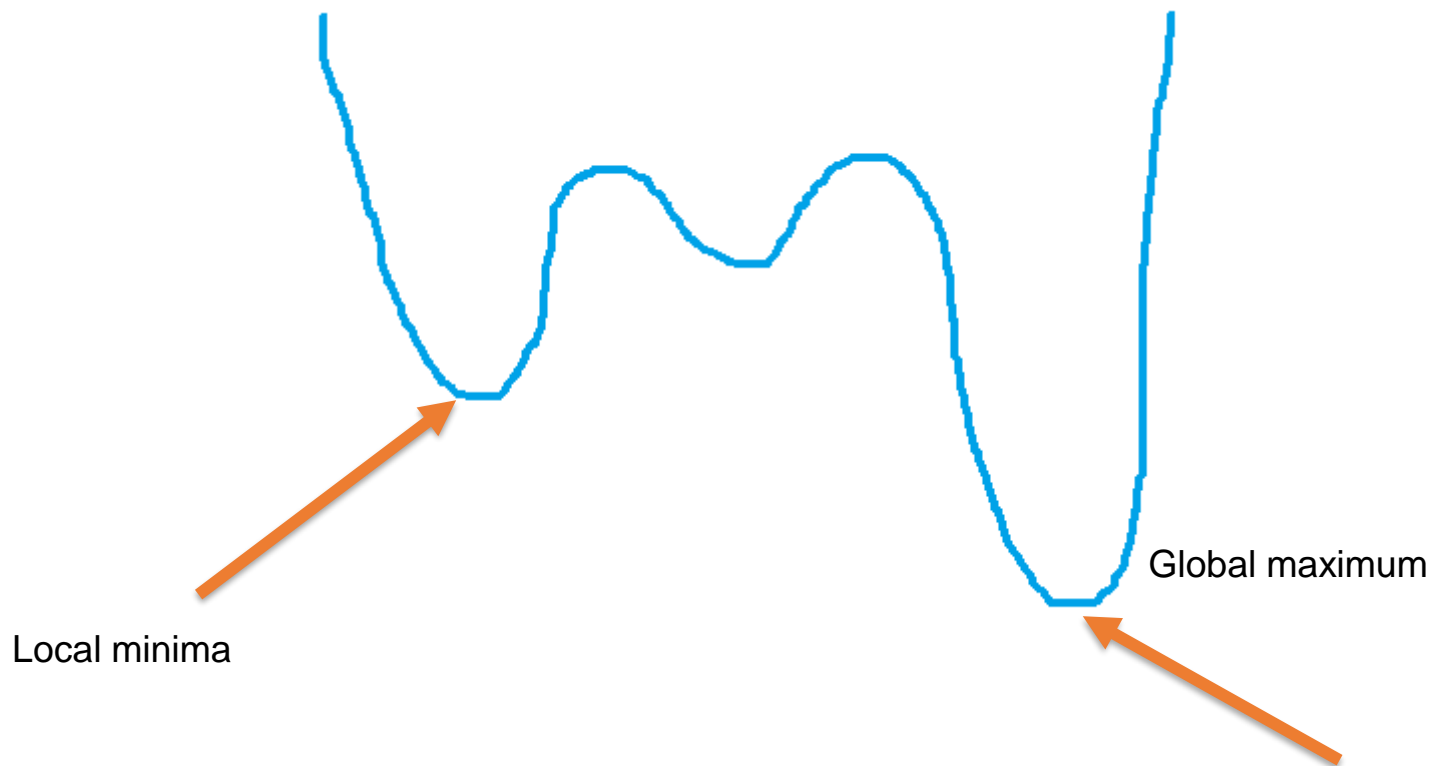


## 02 Gradient Descent

### ▶ 경사 하강법의 문제 - local minima

최근에는 실제로 딥러닝 수행시, local minima에 빠질 확률이 거의 없다고 보여진다.

실제 딥러닝 모델에서는  $w$ 가 수도 없이 많아, 수많은  $w$ 가 모두 local minima에 빠져야  $w$ 업데이트가 정지되므로 사실상 이에 대한 고려가 필요가 없다는 의견이 다수.





## 02 Gradient Descent – Learning Rate

### ▶ Learning Rate

Learning Rate는 학습 속도를 의미합니다.

A. Learning Rate가 지나치게 작다면 최솟값에 도달하기 위해 굉장히 많은 연산이 요구.

B. Learning Rate가 지나치게 크다면 반대쪽을 오가며 매우 큰 거리를 이동하게 되어 최솟값에서 점점 멀어지게 된다. 따라서 적절한 Learning Rate를 설정하는 것이 매우 중요.

## 03 Stochastic Gradient Descent

- ▶ SGD(Stochastic Gradient Descent)는 최적화 알고리즘이다.
- ▶ 알고리즘의 역할은 로그 손실 또는 평균 제곱 오차와 같은 성능 측정에 대해 잘 수행되는 내부 모델 매개 변수를 찾는 것.
- ▶ 전체 데이터 (Batch)가 아닌 일부 데이터의 모음(Mini-Batch)를 사용하는 방법  
A. Mini-Batch를 사용하여 다수 부정확할 수 있지만, 계산 속도가 빠르다.
- ▶ 매단계에서는 모델을 이용하여 예측하고 실제 결과와 비교하고, 오차를 계산하고 오차를 이용하여 내부 모델의 매개 변수를 업데이트 한다.

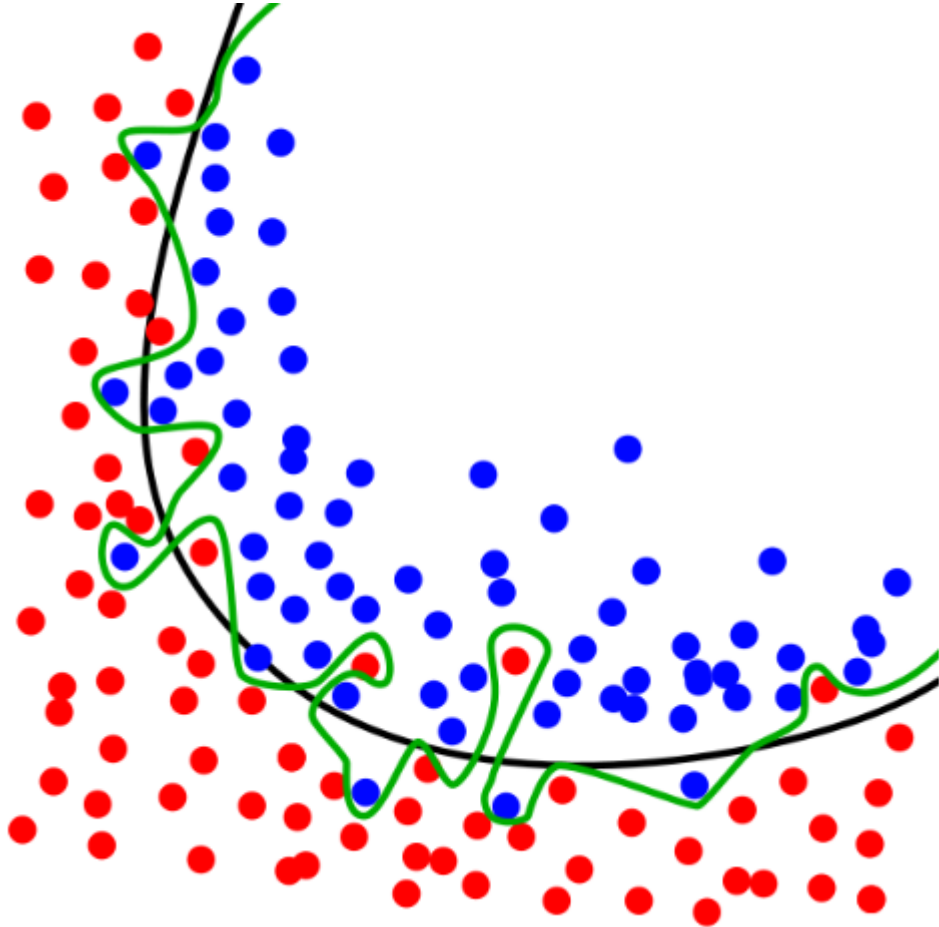
## 04 Batch Size

▶ mini-batch gradient descent의 경우, 보통 많이 쓰이는 배치는 32, 64, 128이 사용.

▶ 배치 사이즈에 학습 프로세스에 미치는 영향

<https://machinelearningmastery.com/how-to-control-the-speed-and-stability-of-training-neural-networks-with-gradient-descent-batch-size/>

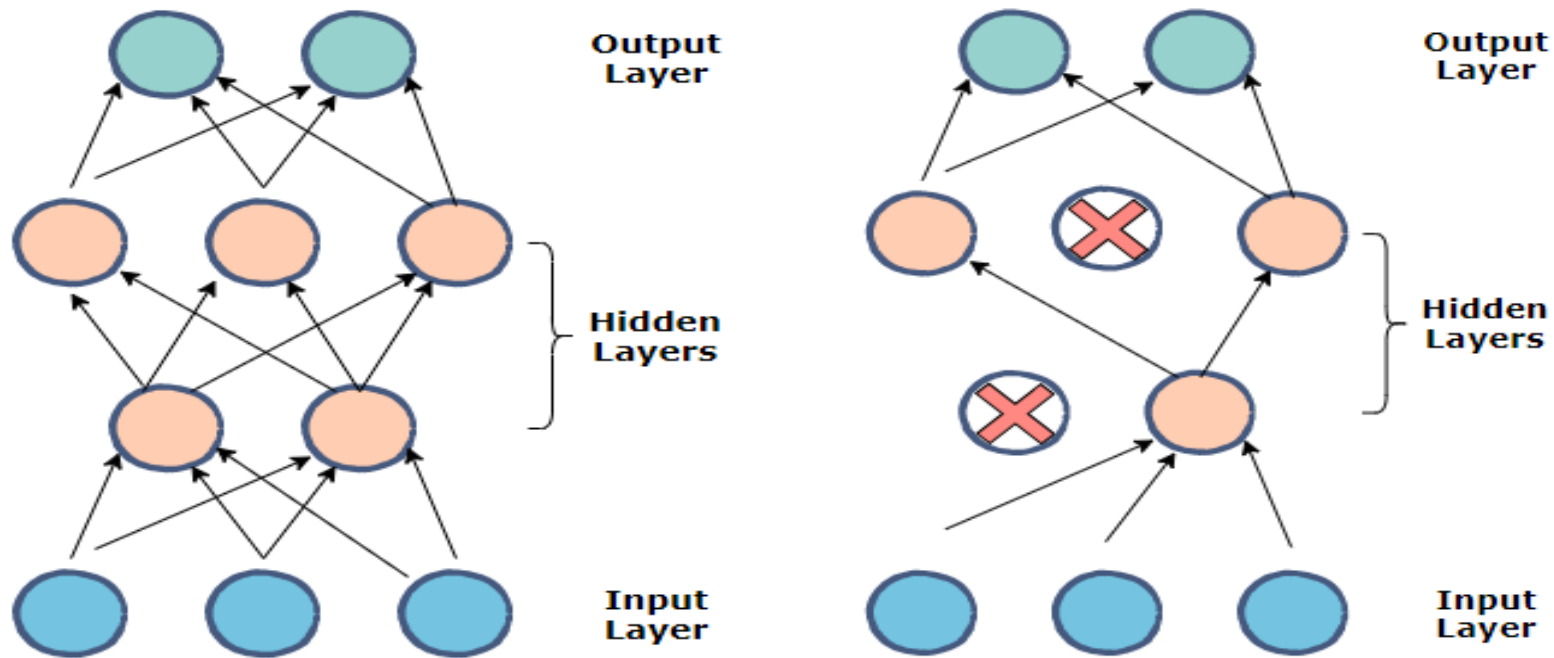
## 05 과적합(Overfitting)



학습용 데이터에만 넘 잘 맞는 것.

## 06 과적합 해결하기

### ▶ Dropout – 과적합 해결



왼쪽이 원래 신경망, 오른쪽에 Dropout된 신경망

## 06 과적합 해결하기

- ▶ Batch Normalization(배치 정규화)
- ▶ Data Augmentation

## 06 REFERENCE

- ▶ 실제 시뮬레이션으로 gradient descent에 대한 이해가 좋음.

[https://angeloyeo.github.io/2020/08/16/gradient\\_descent.html](https://angeloyeo.github.io/2020/08/16/gradient_descent.html)