

## Agenda

1. Learning Rate: 경사하강알고리즘  $W_{t+1} := W_t - \alpha \frac{\partial}{\partial W_t} cost(W_t)$ 에서  $\alpha$

lr이 너무 크면 발산한다, 너무 작으면 너무 오래 걸린다.

실험을 통해 적절한 lr을 찾아야 한다.

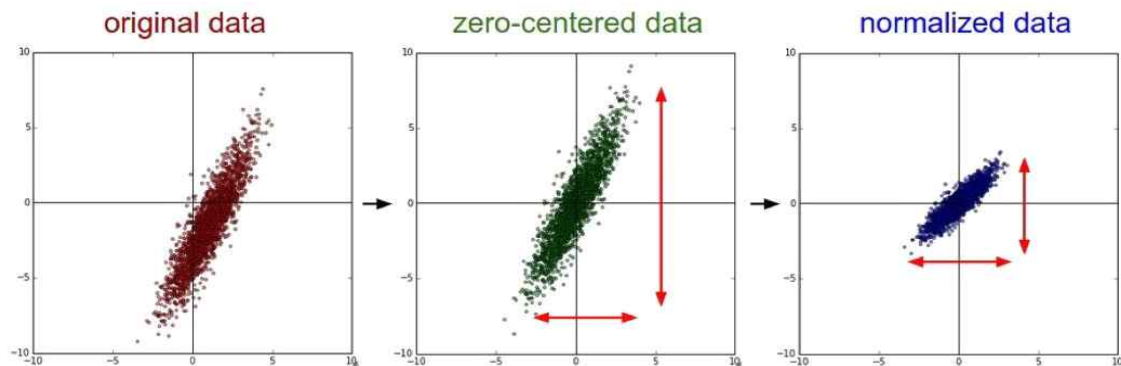
핑퐁되어 갇힐 수 있기 때문에 iteration에 따라 lr을 줄여준다.

2. Data Preprocessing

lr은 모든 feature의 가중치에 동일하게 적용되므로

각 feature 스케일의 차이가 크면 동일한 영향을 주지 못한다.

이러한 이유로 data preprocessing을 통해 스케일을 맞춰준다.



3. Avoid Overfitting

- More training data

- Reduce the number of features

- Regularization

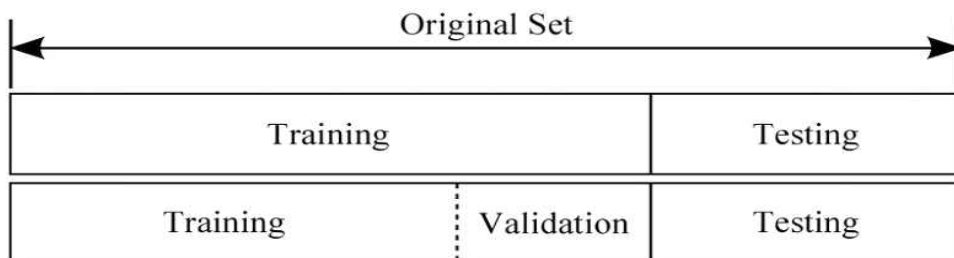
: let's not have too big numbers in the weight  
cost function + regularization

$$\frac{1}{N} \sum_i^N D(S(WX_i + b), L_i) + \lambda \sum W^2, \lambda \text{는 정규화의 강도}$$

4. Performance evaluation

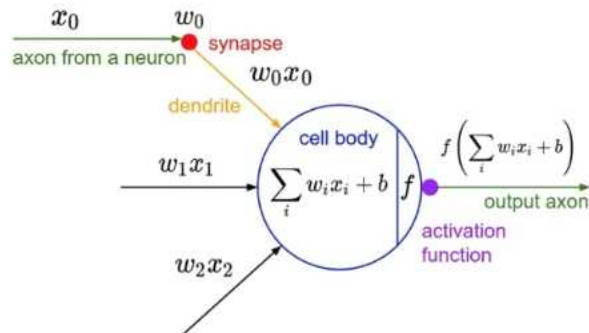
- use the learned parameter(W, b)

- use the unseen data = do not use training data

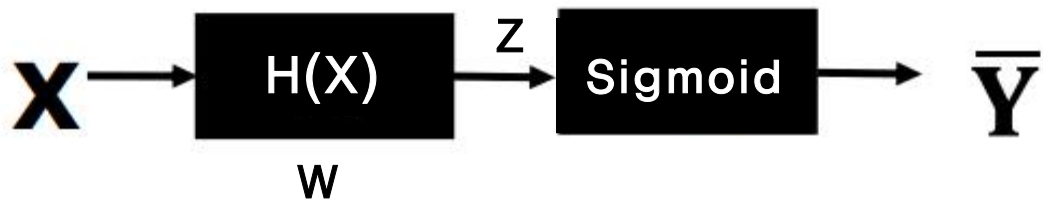


## 뉴럴네트워크(Neural Network)

activation function: 일정 기준 이상이면 1, 아니면 0



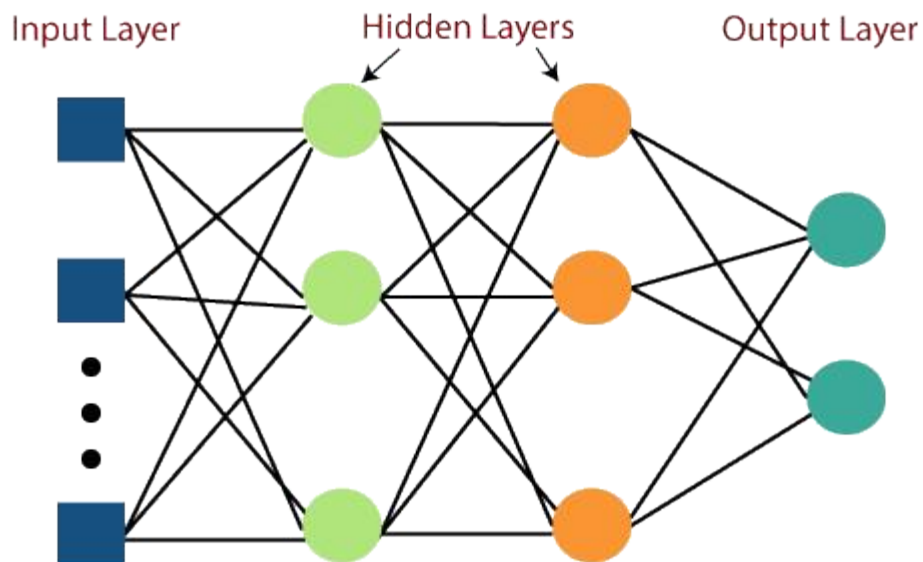
Perceptron(neuron) = 선형분류기 = Feed-forward network



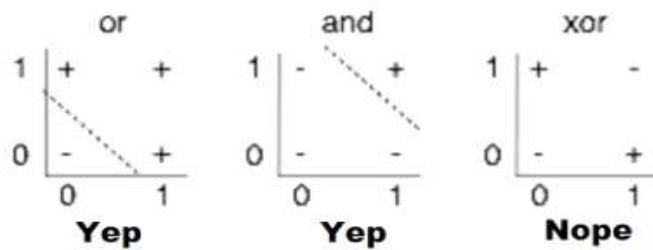
## MLP(Multi Layered Perceptron)

: 퍼셉트론(뉴런) 하나가 노드 하나가 되어 구성한다.

1.입력층 2. 은닉층 3. 출력층



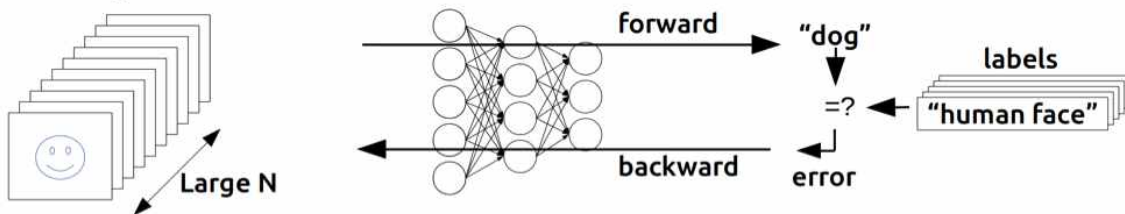
※ AND/OR 문제와 XOR 문제를 선형으로 풀 수 있을까?



## ★ 역전파(Backpropagation)

: 역방향으로 오차를 전파하면서 각 층의 가중치를 업데이트하는 방법

### Training



## ★ CNN(Convolutional Neural Network)

영상 전체로 feature가 기술되는 것이 아니라

영상의 부분과 부분의 결합을 통해서 최종적인 feature가 만들어진다는

영상의 부분 = Receptive field

※ 역전파가 레이어를 많이 쌓은 신경망에서는 정상 동작하지 않았다

Gradient vanishing 문제: 너무 깊다 보니 역전파 작업을 하면서 gradient가 소실하여 어느 순간 0이 된다.

Other rising ML algorithms SVM, Random Forest

## ★ Weigh Initialization : w, b의 초기화

## ★ Dropout

Geoffrey Hinton's summary

1. Labeled datasets were too small. → 빅데이터
2. Computers were too slow. → GPU
3. We initialed the weights in a stupid way. → w, b의 초기화
4. We used the wrong type of non-linearity.