AI Project

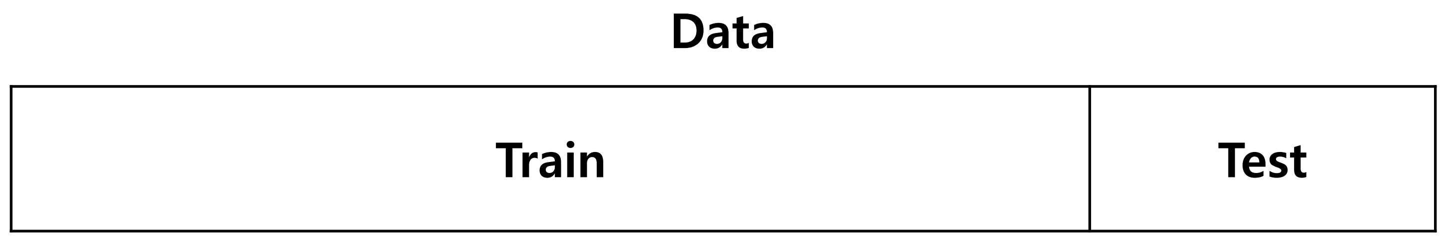
3. 다층 퍼셉트론 모델

GNB 김도훈

Index

1. 2주차 교육 내용 복습
2. 퍼셉트론과 활성화 함수
3. 다층 퍼셉트론 모델
4. 순전파와 역전파
5. Lab 3 : Multilayer Perceptron

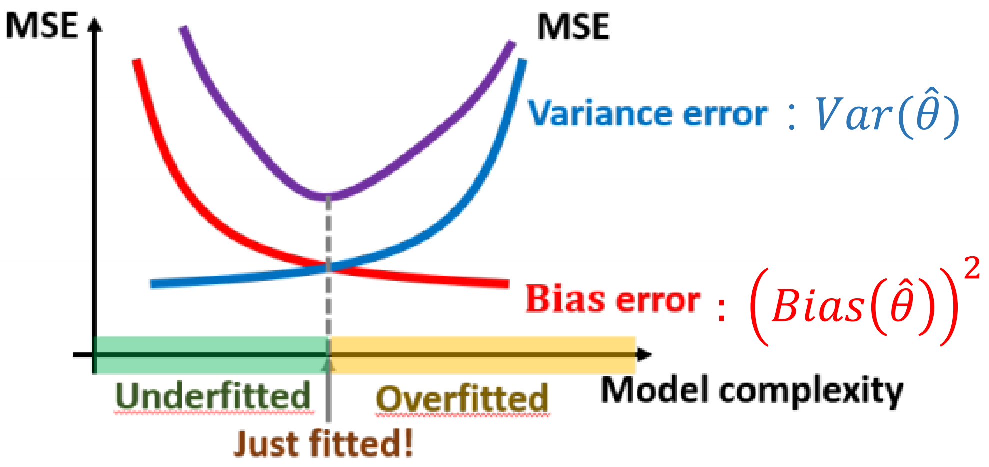
복습: Train / Test split

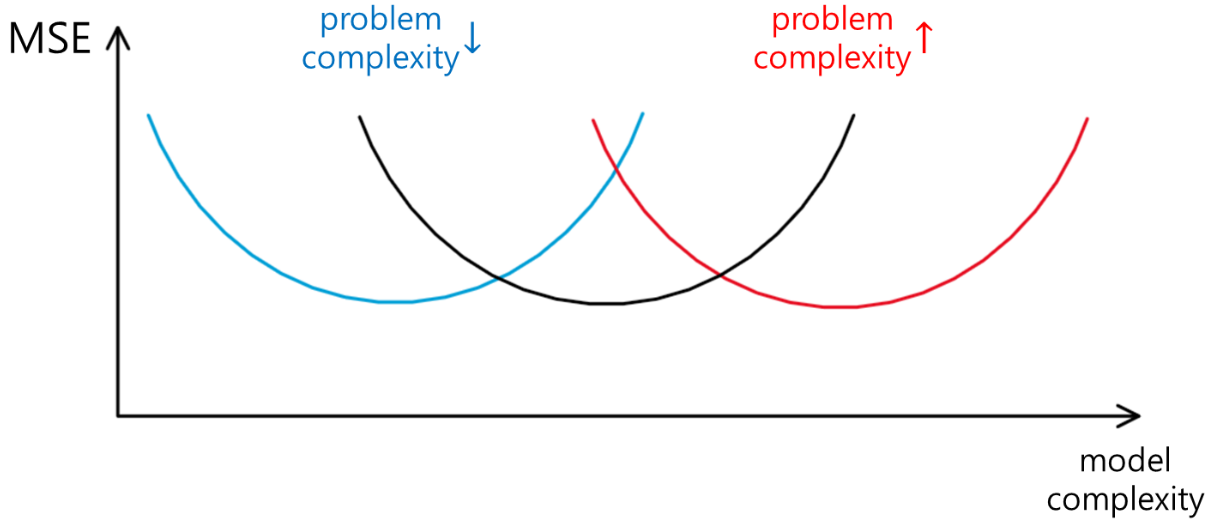


- Train set : 최적해 θ^*를 구하기 위한 데이터
- Test set : 새로운 데이터에 대한 θ^*의 적합도를 평가하기 위한 데이터

- Train error와 Test error를 비교하여 학습이 잘 되었는지 평가하기 위함
- 학습에 사용하지 않은 데이터로 모델의 성능을 평가

복습: 모델 복잡도와 문제 복잡도



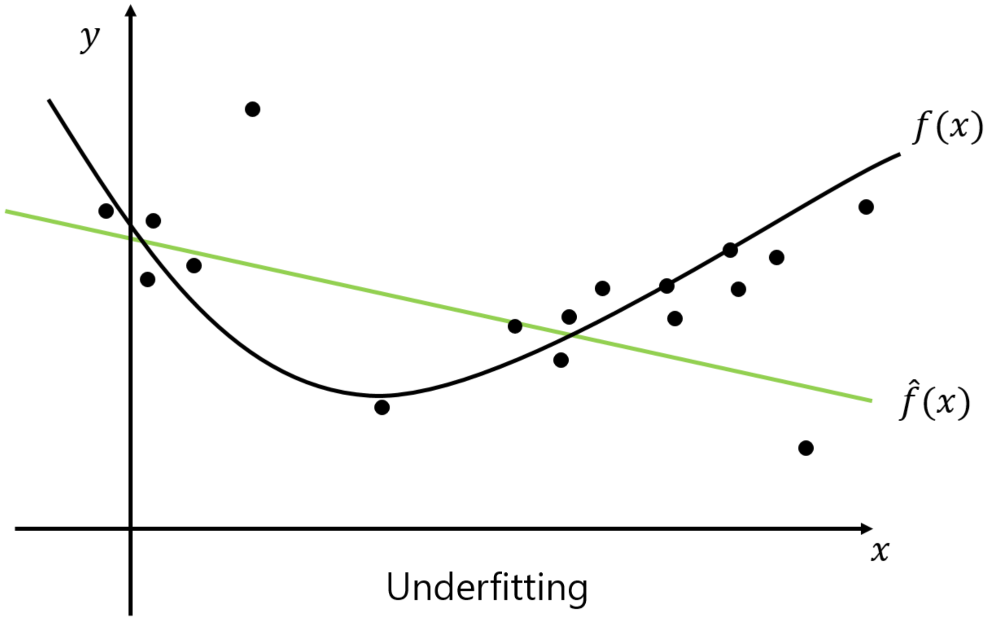


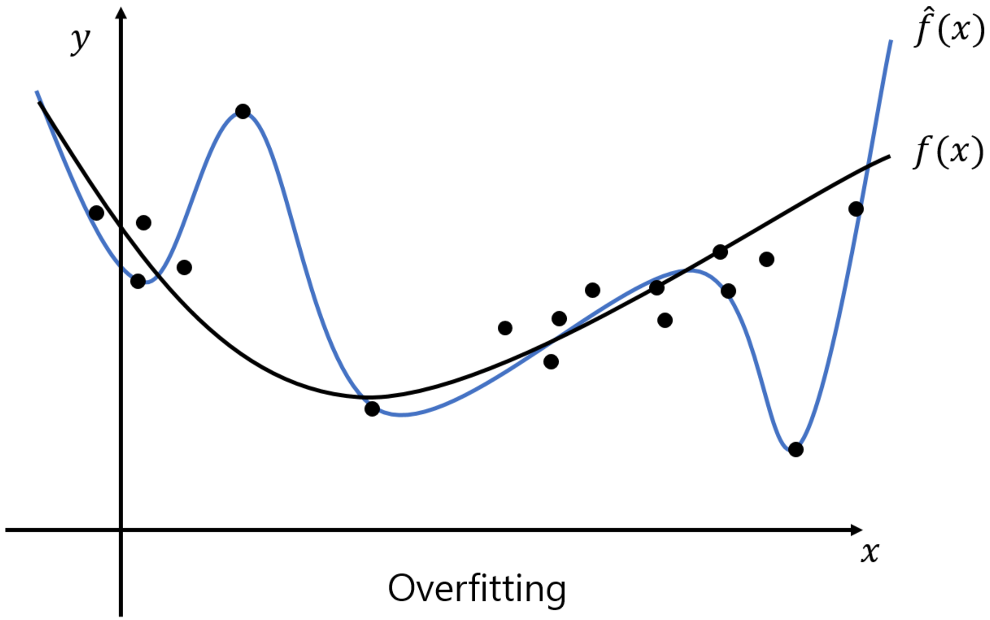
$Var(\hat{\theta})$: test error

$\left(Bias(\hat{\theta})\right)^2$: train error

문제의 복잡도에 맞는 복잡도를 가진 모델을 선택

복습: 과소적합, 과적합 문제





model complexity ↓ $Var(\hat{\theta}) \downarrow$ $\left(Bias(\hat{\theta})\right)^2 \uparrow$

model complexity ↑ $Var(\hat{\theta}) \uparrow$ $\left(Bias(\hat{\theta})\right)^2 \downarrow$

복습: 정규화

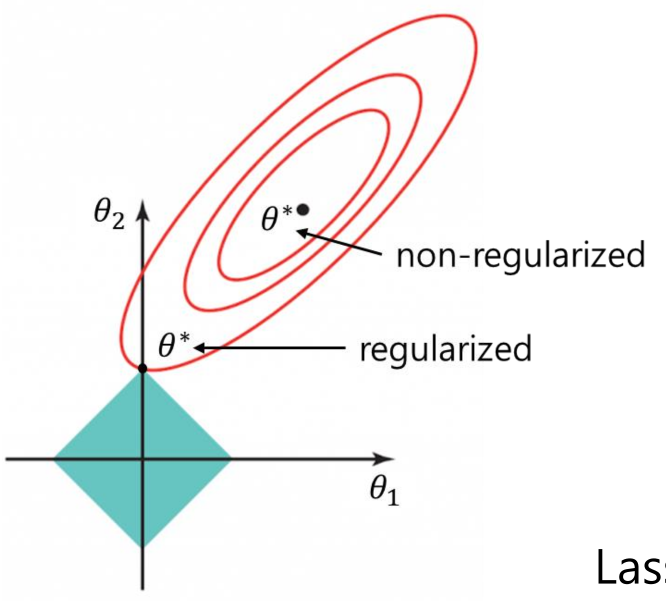
- θ의 크기를 제한 ⇒ 과적합 완화

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, E(\theta)$$
$$s.t.\ \ \|\bar{\theta}\|_p^p \le c^p$$

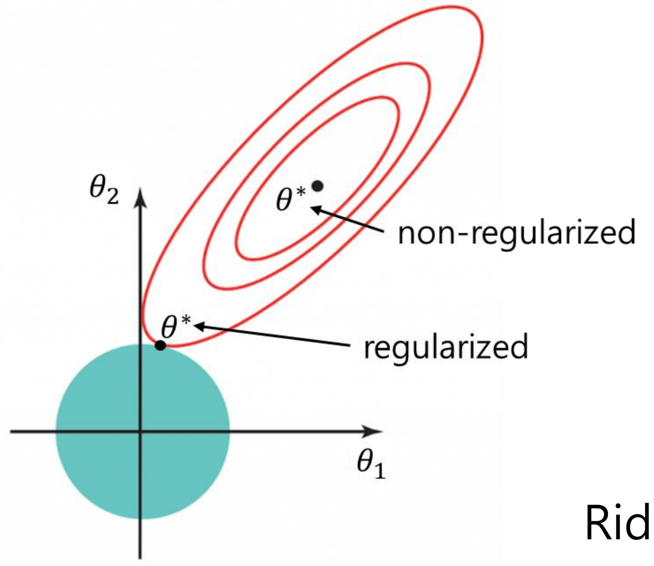
Lagrangian method

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \left(E(\theta) + \lambda \left(\|\bar{\theta}\|_p^p - c^p \right) \right)$$

$$\boxed{\theta^* = \underset{\theta}{\operatorname{argmin}} \left(E(\theta) + \lambda \|\bar{\theta}\|_p^p \right)}$$



Lasso : p=1



Ridge : p=2

복습: Ridge 정규화

1. 경사 하강법

데이터로부터 학습한 정보를 덜 신뢰한다!

$$\theta_n \leftarrow \theta_n - \alpha \left(\sum_{i=1}^{I} \left(\theta^T x^{(i)} - y^{(i)} \right) x_n^{(i)} - \lambda \theta_n \right)$$

2. Closed-form solution

$$\theta^* = \left(x^T x + \lambda \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \right)^{-1} x^T y$$

$shape : \ [N+1, N+1]$

- 행렬의 크기
  $\theta : [N+1, 1]$
  $x : [I, N+1]$
  $y : [I, 1]$

인공 신경망 (Artificial Neural Network)

- 인공 신경망(ANN)은…

  인간의 신경 조직을 수학적으로 모델링하여

  컴퓨터가 인간처럼 기억·학습·판단할 수 있도록 구현한 것

- 인공 신경망의 종류
  - Feedforward neural network

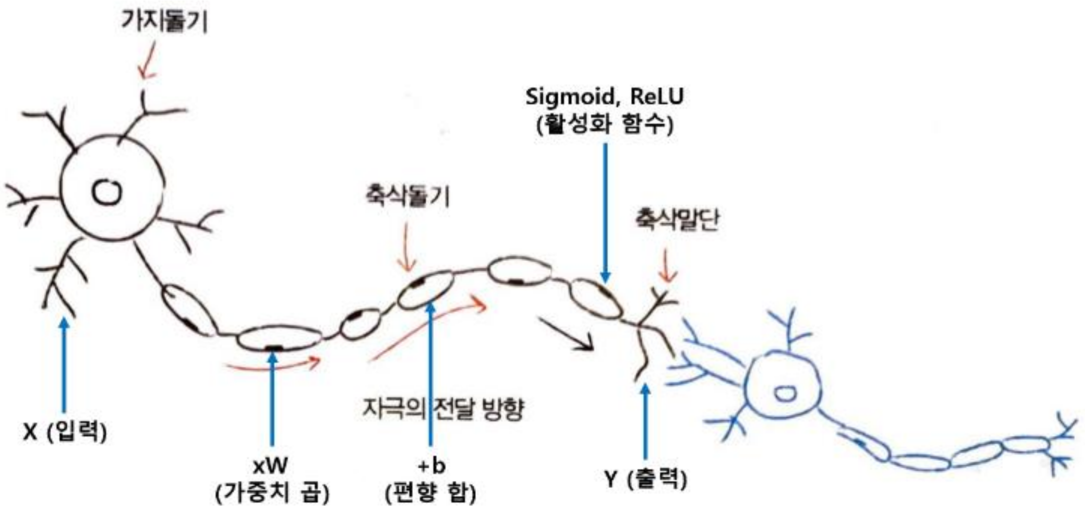
    다층 퍼셉트론(MLP)은 FFNN에 해당한다.

  - Feedback neural network

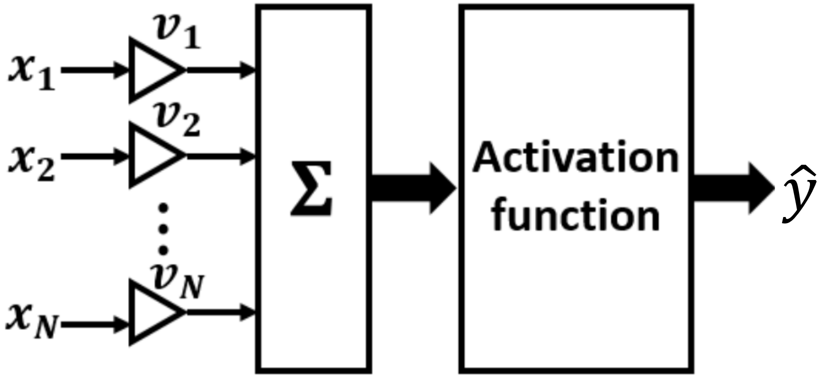
    순환 신경망(Recurrent neural network) = FBNN

퍼셉트론 (Perceptron)

- 뉴런의 기능을 수학적으로 모델링한 것



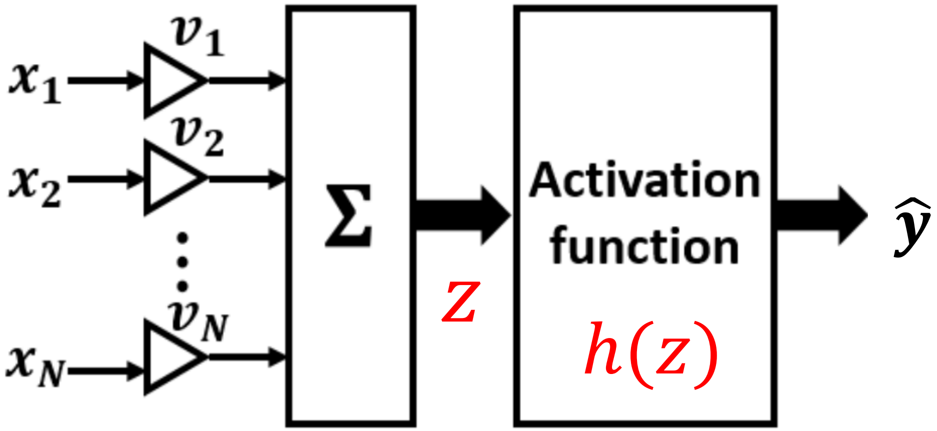
뉴런의 구조



퍼셉트론의 구조

퍼셉트론 (Perceptron)

- 퍼셉트론의 구조

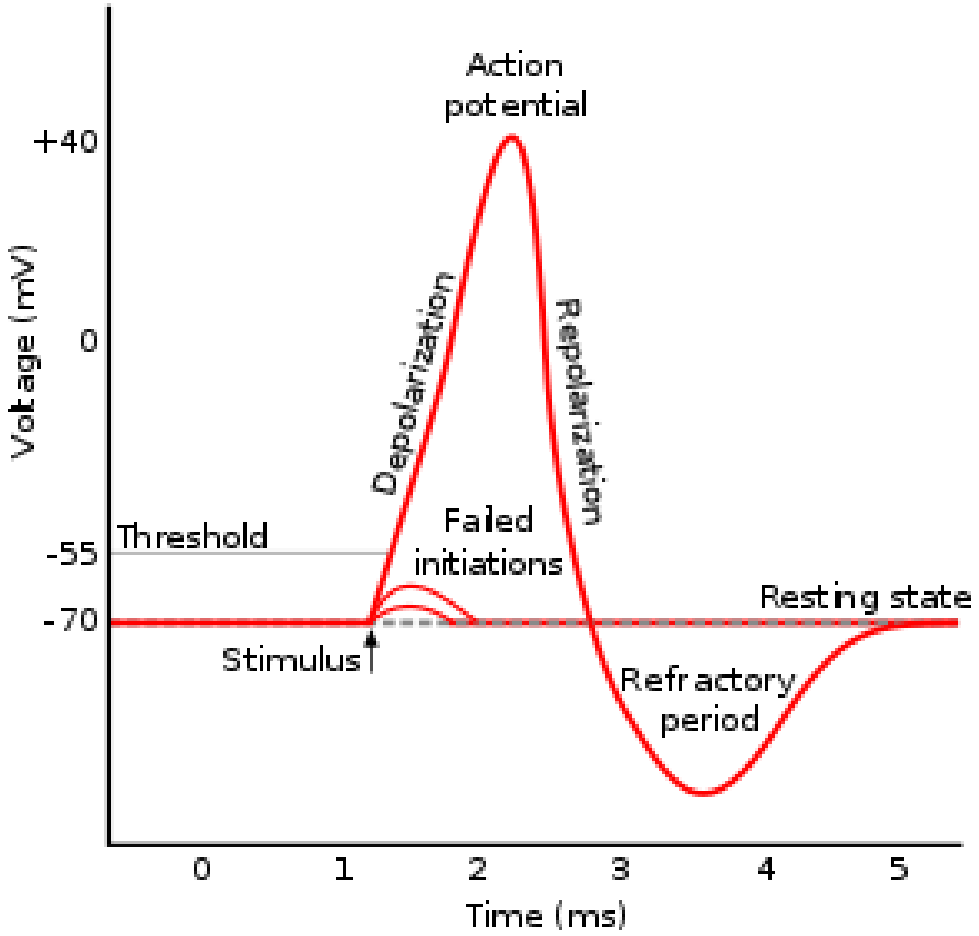


$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \qquad v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{bmatrix}$$

$$z = x \cdot v = x_1 v_1 + \cdots + x_N v_N$$

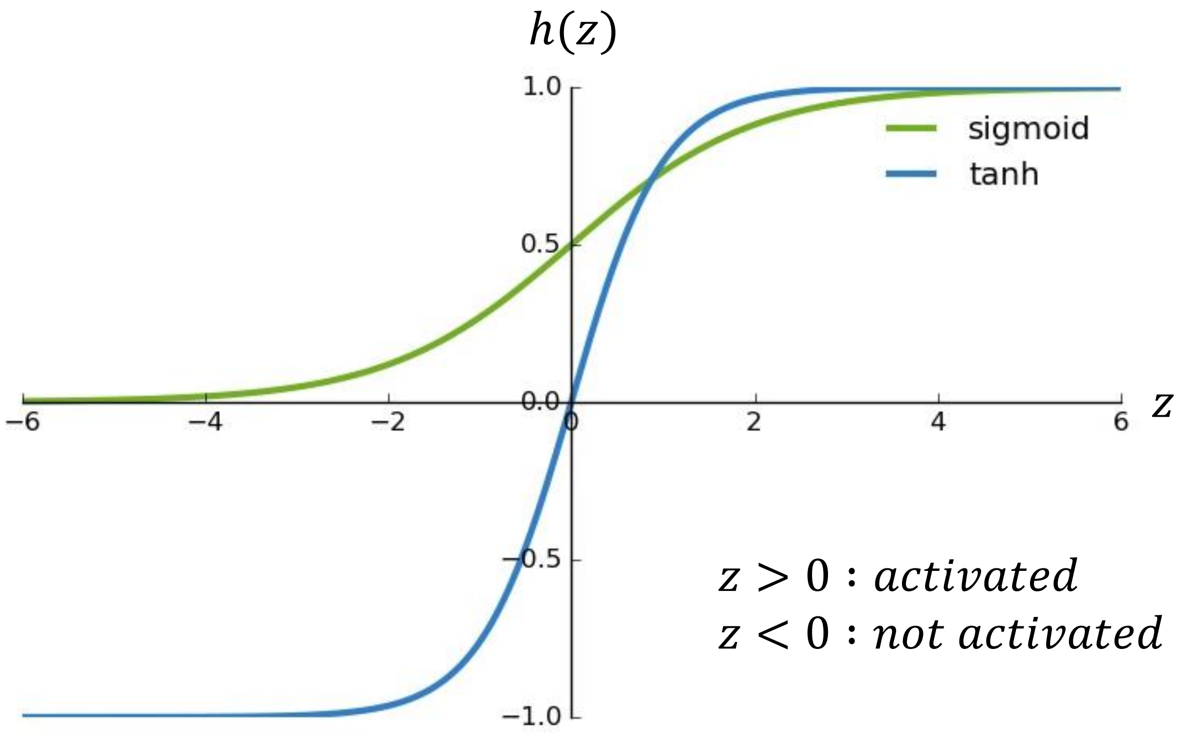
$$\hat{y} = h(z) = h_v(x)$$

활성화 함수 (Activation function)



- 뉴런의 동작
  - 뉴런의 휴지 전위는 -70mV
  - 역치 전위는 -55mV
  - 자극이 오면 막이 탈분극한다.
  - 충분히 탈분극하여 역치 전위에 도달하면 활동 전위가 일어난다.
- 자극이 기준값보다 크면 증폭!
- 자극이 기준값보다 작으면 무시!

활성화 함수 (Activation function)



1. Sigmoid (=Logistic function)

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

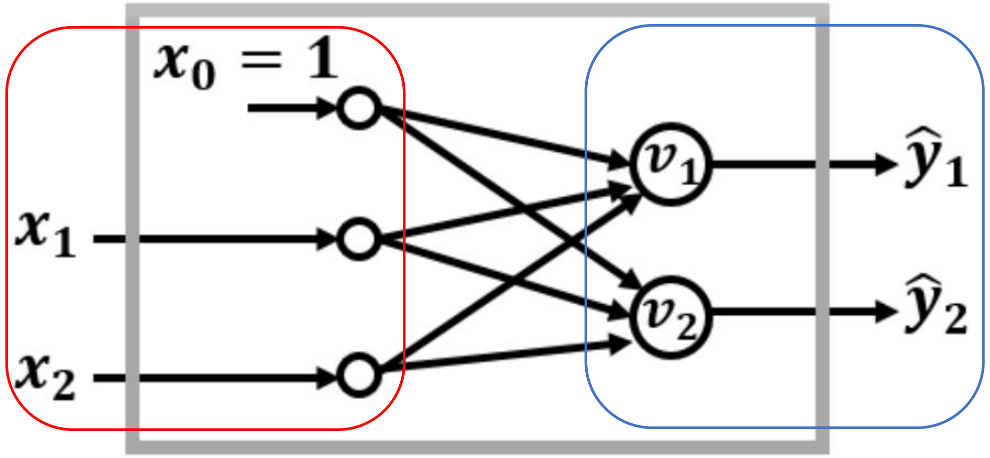
2. Hyperbolic tangent

$$tanh(z) = \frac{\sinh z}{\cosh z} = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$$

이외에도 매우 다양한 활성화 함수가 존재한다.
데이터와 모델의 특성에 맞는 활성화 함수를 선택해서 사용한다.

다층 퍼셉트론 (Multilayer Perceptron)

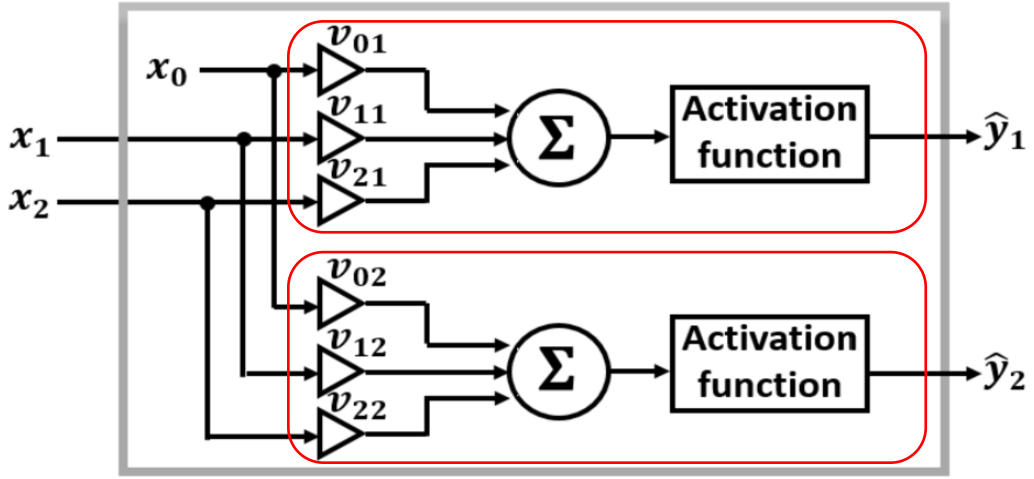
- 2개의 층을 가진 MLP 모델



Input Layer

Output Layer

모델의 Layer 구조를 단순히 나타낸 표현

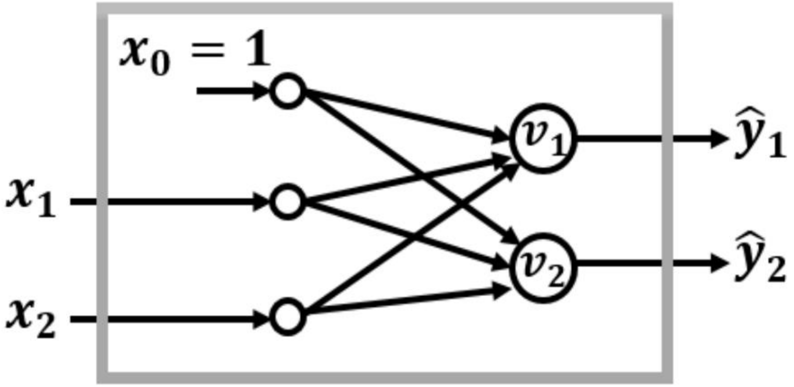


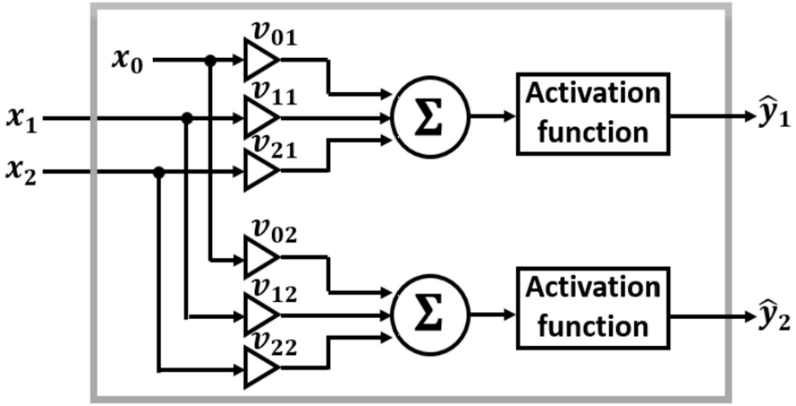
2 Perceptrons

실제 값의 연산과정을 확인할 수 있는 표현

다층 퍼셉트론 (Multilayer Perceptron)

- 2개의 층을 가진 MLP 모델





$$x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} \quad v = [v_1 \quad v_2] = \begin{bmatrix} v_{01} & v_{02} \\ v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}$$

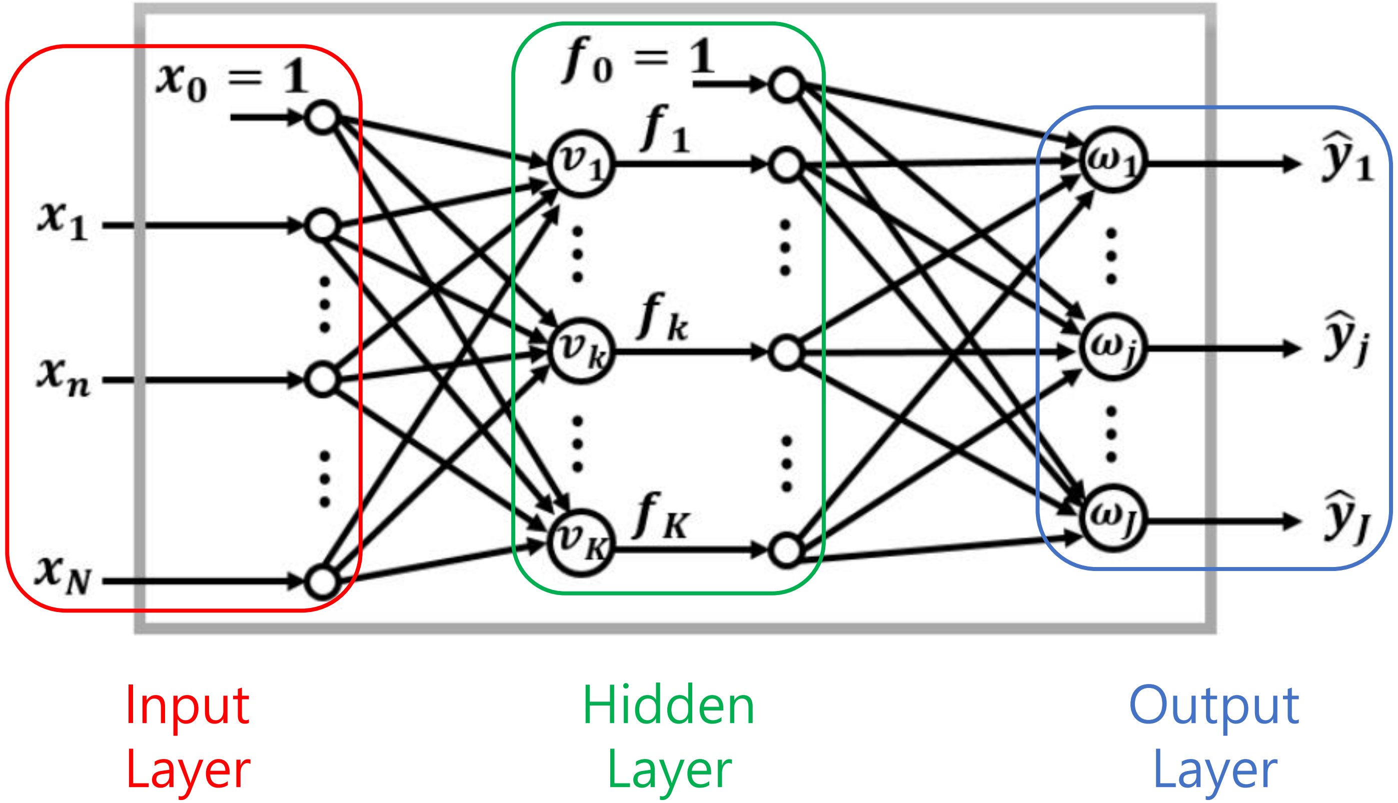
$$v_1 = \begin{bmatrix} v_{01} \\ v_{11} \\ v_{21} \end{bmatrix} \quad v_2 = \begin{bmatrix} v_{02} \\ v_{12} \\ v_{22} \end{bmatrix}$$

$$\hat{y}_1 = x \cdot v_1 = x^T v_1 = x_0 v_{01} + x_1 v_{11} + x_2 v_{21}$$

$$\hat{y}_2 = x \cdot v_2 = x^T v_2 = x_0 v_{02} + x_1 v_{12} + x_2 v_{22}$$

다층 퍼셉트론 (Multilayer Perceptron)

- 3개의 층을 가진 MLP 모델

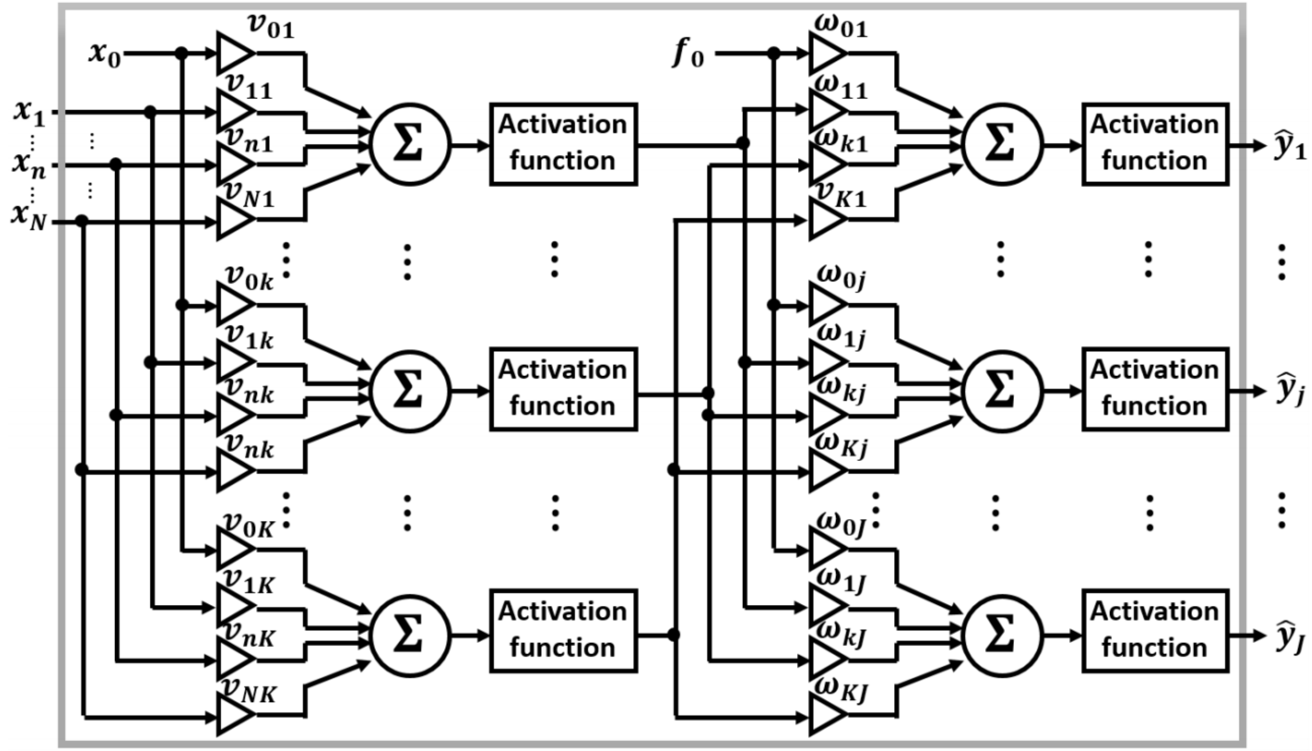


x_0, f_0 : bias (=intercept)

N : 입력 데이터의 차원
I : 입력 데이터의 개수
J : 출력 데이터의 차원
K : 은닉층의 뉴런 개수

다층 퍼셉트론 (Multilayer Perceptron)

- 3개의 층을 가진 MLP 모델



$$v = \begin{bmatrix} v_{01} & v_{02} & \cdots & v_{0K} \\ v_{11} & v_{12} & \cdots & v_{1K} \\ \vdots & \vdots & \ddots & \vdots \\ v_{N1} & v_{N2} & \cdots & v_{NK} \end{bmatrix}$$

$$w = \begin{bmatrix} w_{01} & w_{02} & \cdots & w_{0J} \\ w_{11} & w_{12} & \cdots & w_{1J} \\ \vdots & \vdots & \ddots & \vdots \\ w_{K1} & w_{K2} & \cdots & w_{KJ} \end{bmatrix}$$

다층 퍼셉트론 (Multilayer Perceptron)

- 오차 함수 : Sum of squared error

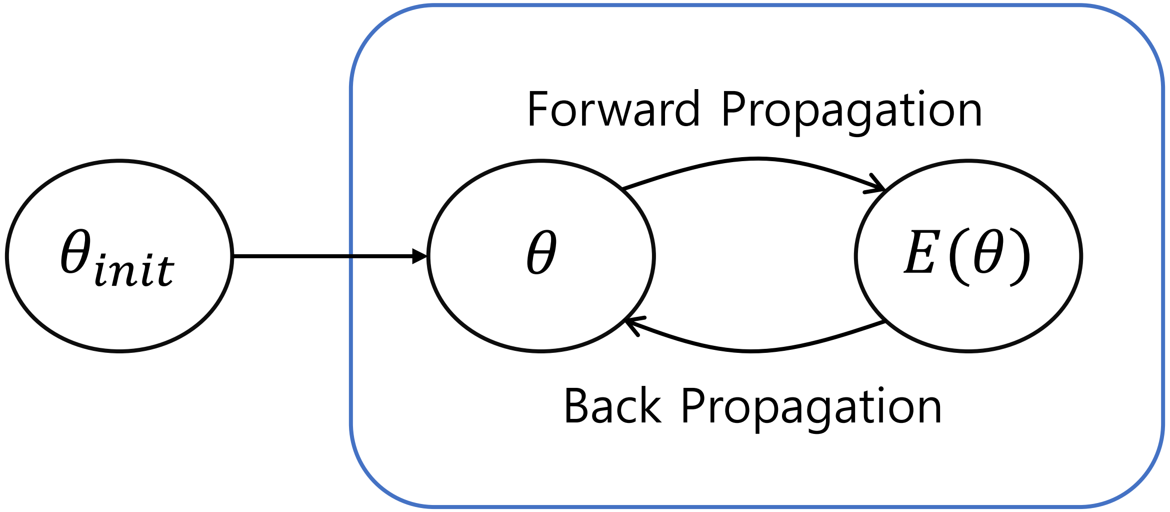
$$E(\theta) = \frac{1}{2}\sum_{i=1}^{I}\left\|\hat{y}^{(i)} - y^{(i)}\right\|_2^2 = \frac{1}{2}\sum_{i=1}^{I}\sum_{j=1}^{J}\left(\hat{y}_j^{(i)} - y_j^{(i)}\right)^2$$

- 문제 정의 : 비용 극소화 문제

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, E(\theta), \qquad \theta = (v, w)$$

다층 퍼셉트론 (Multilayer Perceptron)

- 이 문제를 풀기 위한 방법



수렴할 때까지 반복

- 순전파

  파라미터 θ값으로 부터
  오차 E를 구하는 과정

- 역전파

  오차 E를 통해 파라미터
  θ를 업데이트하는 과정

순전파 (Forward propagation)

Input data matrix

$$X = \begin{bmatrix} \left(x^{(1)}\right)^T \\ \vdots \\ \left(x^{(I)}\right)^T \end{bmatrix}, \qquad \text{where} \quad x^{(i)} \in \mathbb{R}^{N \times 1} \tag{1}$$

$I \times N$

Extended input data matrix

$$\overline{X} = [\mathcal{I} \quad X] = \begin{bmatrix} \left(\overline{x}^{(1)}\right)^T \\ \vdots \\ \left(\overline{x}^{(I)}\right)^T \end{bmatrix}, \qquad \text{where} \quad \mathcal{I} = [1, \cdots, 1]^T \in \mathbb{R}^{I \times 1} \tag{2}$$

$I \times (N + 1)$

intercept

$$\overline{\overline{X}} = \overline{X} \cdot v = \begin{bmatrix} \left(\overline{x}^{(1)}\right)^T \cdot v \\ \vdots \\ \left(\overline{x}^{(I)}\right)^T \cdot v \end{bmatrix} \tag{3}$$

$I \times K$

순전파 (Forward propagation)

Define

$$F = \left(1 + \exp\left(-\overline{\overline{X}}\right)\right)^{-1} = \begin{bmatrix} \left(f^{(1)}\right)^T \\ \vdots \\ \left(f^{(I)}\right)^T \end{bmatrix}_{I \times K} \tag{4}$$

sigmoid

where the operator $^{-1}$ represents the elementwise inversion.

$$\overline{F} = [\mathcal{I} \quad F] = \begin{bmatrix} \left(\overline{f}^{(1)}\right)^T \\ \vdots \\ \left(\overline{f}^{(I)}\right)^T \end{bmatrix}_{I \times (K+1)}, \qquad \text{where} \quad \mathcal{I} = [1, \cdots, 1]^T \in \mathbb{R}^{I \times 1} \tag{5}$$

intercept

$$\overline{\overline{F}} = \overline{F} \cdot \omega = \begin{bmatrix} \left(\overline{f}^{(1)}\right)^T \cdot \omega \\ \vdots \\ \left(\overline{f}^{(I)}\right)^T \cdot \omega \end{bmatrix}_{I \times J} \tag{6}$$

순전파 (Forward propagation)

Define

$$G = \left(1 + \exp\left(-\overline{\overline{F}}\right)\right)^{-1} = \begin{bmatrix} \left(g^{(1)}\right)^T \\ \vdots \\ \left(g^{(I)}\right)^T \end{bmatrix}_{I \times J} \tag{7}$$

sigmoid

where $^{-1}$ represents once again the elementwise inversion.

Then, one can find the labels using the following algorithm:

Algorithm 1: find labels

```
1 Initialize label;
2 for i = 1 : I do
3     label(i) ← arg max_j G(i, j) - 1;
4 return label;
```

순전파 (Forward propagation)

Let us define the **sum of squared errors (SSE)** as

$$E = \frac{1}{2}\sum_{i=1}^{I}\left(\hat{y}^{(i)} - y^{(i)}\right)^2 = \frac{1}{2}\sum_{i=1}^{I}\sum_{j=1}^{J}\left(\hat{y}_j^{(i)} - y_j^{(i)}\right)^2 = \frac{1}{2}\sum_{i=1}^{I}\sum_{j=1}^{J}\left(g_j^{(i)} - y_j^{(i)}\right)^2 \qquad (8)$$

where $g_j^{(i)} = g\left(\overline{f}\left(\overline{x}^{(i)}\right)\right) = g \circ \overline{f}\left(\overline{x}^{(i)}\right)$.

- 순전파 요약

순전파는 파라미터 θ값으로 부터 오차 E를 구하는 과정이다.

$$\boxed{X} \rightarrow \boxed{\bar{X} \rightarrow \bar{\bar{X}} \rightarrow F} \rightarrow \boxed{\bar{F} \rightarrow \bar{\bar{F}} \rightarrow G} \Rightarrow E$$

Input Layer

Hidden Layer

Output Layer

역전파 (Back propagation)

- 배치 경사 하강법(BGD)를 통해서 역전파

The update rules are

$$v_{nk} \leftarrow v_{nk} - \alpha_v \frac{\partial E}{\partial v_{nk}}$$
$$w_{kj} \leftarrow w_{kj} - \alpha_w \frac{\partial E}{\partial w_{kj}} \tag{9}$$

But, what are $\frac{\partial E}{\partial v_{nk}}$ and $\frac{\partial E}{\partial w_{kj}}$?

Let us compute first $\frac{\partial E}{\partial w_{kj}}$.

$$\frac{\partial E}{\partial w_{kj}} = \frac{\partial}{\partial w_{kj}} \left[\frac{1}{2} \sum_{i=1}^{I} \sum_{j'=1}^{J} \left(g_{j'}^{(i)} - y_{j'}^{(i)} \right)^2 \right] = \sum_{i=1}^{I} \frac{\partial}{\partial w_{kj}} \left[\frac{1}{2} \sum_{j'=1}^{J} \left(g_{j'}^{(i)} - y_{j'}^{(i)} \right)^2 \right] = \sum_{i=1}^{I} \left(g_{j}^{(i)} - y_{j}^{(i)} \right) \frac{\partial g_j^{(i)}}{\partial w_{kj}}$$

역전파 (Back propagation)

For simplicity, let us drop the notation $^{(i)}$ and compute $\frac{\partial g_j}{\partial w_{kj}}$.

$$
\begin{aligned}
\boxed{\frac{\partial g_j}{\partial w_{kj}}} &= \frac{\partial}{\partial w_{kj}}\left(\frac{1}{1+e^{\left(-(\overline{f})^T w_j\right)}}\right) \\
&= \frac{-e^{\left(-(\overline{f})^T w_j\right)}}{\left(1+e^{\left(-(\overline{f})^T w_j\right)}\right)^2}(-\overline{f}_k) \\
&= \frac{1}{1+e^{\left(-(\overline{f})^T w_j\right)}}\left(1-\frac{1}{1+e^{\left(-(\overline{f})^T w_j\right)}}\right)\overline{f}_k \\
&= \boxed{g_j(1-g_j)\overline{f}_k}
\end{aligned}
\tag{11}
$$

역전파 (Back propagation)

Hence,

$$\frac{\partial g_j}{\partial w_{kj}}$$

$$\frac{\partial E}{\partial w_{jk}} = \sum_{i=1}^{I} \left(g_j^{(i)} - y_j^{(i)} \right) g_j^{(i)} \left(1 - g_j^{(i)} \right) \overline{f}_k^{(i)} \tag{12}$$

$$\boxed{w_{kj} \leftarrow w_{kj} - \alpha_1 \sum_{i=1}^{I} \left(g_j^{(i)} - y_j^{(i)} \right) g_j^{(i)} \left(1 - g_j^{(i)} \right) \overline{f}_k^{(i)}} \tag{13}$$

where $k = \{0, 1, \cdots, K\}$ and $j = \{1, 2, \cdots, J\}$.

역전파 (Back propagation)

Now, let us compute $\frac{\partial E}{\partial v_{nk}}$.

$$\frac{\partial E}{\partial v_{nk}} = \frac{\partial}{\partial v_{nk}}\left[\frac{1}{2}\sum_{i=1}^{I}\sum_{j=1}^{J}\left(g_j^{(i)} - y_j^{(i)}\right)^2\right] = \sum_{i=1}^{I}\sum_{j=1}^{J}\frac{\partial}{\partial v_{nk}}\left[\frac{1}{2}\left(g_j^{(i)} - y_j^{(i)}\right)^2\right] \tag{14}$$

Then,

$$\frac{\partial}{\partial v_{nk}}\left[\frac{1}{2}\left(g_j^{(i)} - y_j^{(i)}\right)^2\right] = \left(g_j^{(i)} - y_j^{(i)}\right)\frac{\partial g_j^{(i)}}{\partial v_{nk}} \tag{15}$$

For simplicity, let us drop the notation $^{(i)}$ and compute $\frac{\partial g_j^{(i)}}{\partial v_{nk}}$.

$$\frac{\partial g_j}{\partial v_{nk}} = \frac{\partial g_j}{\partial \bar{f}_k}\frac{\partial \bar{f}_k}{\partial v_{nk}} \tag{16}$$

역전파 (Back propagation)

On the other hand,

$$\begin{aligned}
\boxed{\frac{\partial g_j}{\partial \bar{f}_k}} &= \frac{\partial}{\partial \bar{f}_k}\left(\frac{1}{1+e^{\left(-(\bar{f})^T w_j\right)}}\right) \\
&= \frac{-e^{\left(-(\bar{f})^T w_j\right)}}{\left(1+e\left(-(\bar{f})^T w_j\right)\right)^2}(-w_{kj}) \\
&= \frac{1}{1+e^{\left(-(\bar{f})^T w_j\right)}}\left(1-\frac{1}{1+e^{\left(-(\bar{f})^T w_j\right)}}\right) w_{kj} \\
&= \boxed{g_j(1-g_j)w_{kj}}
\end{aligned} \tag{17}$$

역전파 (Back propagation)

Now,

$$
\begin{aligned}
\boxed{\frac{\partial \overline{f}_k}{\partial v_{nk}}} &= \frac{\partial f_k}{\partial v_{nk}} \\
&= \frac{\partial}{\partial v_{nk}}\left(\frac{1}{1+e^{(-(\overline{x})^T v_k)}}\right) \\
&= \frac{-e^{\left(-(\overline{x})^T v_k\right)}}{1+e^{(-(\overline{x})^T v_k)}}(-\overline{x}_n) \\
&= \frac{1}{1+e^{(-(\overline{x})^T v_k)}}\left(1-\frac{1}{1+e^{(-(\overline{x})^T v_k)}}\right)\overline{x}_n \\
&= \boxed{f_k(1-f_k)\overline{x}_n}
\end{aligned}
\tag{18}
$$

역전파 (Back propagation)

Hence,

$$\frac{\partial E}{\partial v_{nk}} = \sum_{i=1}^{I} \sum_{j=1}^{J} \left\{ \left(g_j^{(i)} - y_j^{(i)} \right) \overbrace{g_j^{(i)} \left(1 - g_j^{(i)} \right) w_{kj}}^{\frac{\partial g_j}{\partial \overline{f}_k}} \overbrace{f_k^{(i)} \left(1 - f_k^{(i)} \right) \bar{x}_n^{(i)}}^{\frac{\partial \overline{f}_k}{\partial v_{nk}}} \right\} \tag{19}$$

Now, by considering all training examples,

$$\boxed{v_{nk} \leftarrow v_{nk} - \alpha_2 \sum_{i=1}^{I} \sum_{j=1}^{J} \left\{ \left(g_j^{(i)} - y_j^{(i)} \right) g_j^{(i)} \left(1 - g_j^{(i)} \right) w_{kj} f_k^{(i)} \left(1 - f_k^{(i)} \right) \bar{x}_n^{(i)} \right\}} \tag{20}$$

where $n = \{0, 1, \cdots, N\}$ and $k = \{1, 2, \cdots, K\}$.

역전파 (Back propagation)

- 역전파 요약

  배치 경사 하강법을 통해 파라미터 $\theta = (v, w)$를 업데이트한다

$$\boxed{w_{kj} \leftarrow w_{kj} - \alpha_1 \sum_{i=1}^{I} \left(g_j^{(i)} - y_j^{(i)}\right) g_j^{(i)} \left(1 - g_j^{(i)}\right) \overline{f}_k^{(i)}}$$

where $k = \{0, 1, \cdots, K\}$ and $j = \{1, 2, \cdots, J\}$.

$$\boxed{v_{nk} \leftarrow v_{nk} - \alpha_2 \sum_{i=1}^{I} \sum_{j=1}^{J} \left\{\left(g_j^{(i)} - y_j^{(i)}\right) g_j^{(i)} \left(1 - g_j^{(i)}\right) w_{kj} f_k^{(i)} \left(1 - f_k^{(i)}\right) \bar{x}_n^{(i)}\right\}}$$

where $n = \{0, 1, \cdots, N\}$ and $k = \{1, 2, \cdots, K\}$.

Pseudo code : MLP with 3 layers

- v, w 랜덤으로 초기화하기

$$for\ epoch = 1\ to\ max\_epoch\ \{$$

$$FP:\ X \rightarrow \bar{X} \rightarrow \bar{\bar{X}} \rightarrow F \rightarrow \bar{F} \rightarrow \bar{\bar{F}} \rightarrow G \Rightarrow E$$

$$BP:\ \forall k, j\quad w_{kj} \leftarrow w_{kj} - \alpha_w \frac{\partial E}{\partial w_{kj}} \qquad k = \{0, 1, \cdots, K\} \quad j = \{1, 2, \cdots, J\}$$

$$\forall n, k\quad v_{nk} \leftarrow v_{nk} - \alpha_v \frac{\partial E}{\partial v_{nk}} \qquad n = \{0, 1, \cdots, N\} \quad k = \{1, 2, \cdots, K\}$$

$$\}$$

Lab 3 : Multilayer Perceptron

- Tasks
  1. data_lab3.txt의 데이터를 읽어서 x, y를 반환하는 함수를 작성하여라.
     data_lab3.txt는 x_1, x_2, y의 3개의 열로 이루어진 데이터이다.
  2. MLP를 통한 분류기 모델을 구현하여라.
     단 MLP는 3개의 층을 가지며, 은닉층은 5개의 뉴런으로 이루어져있다.
  3. 은닉층과 출력층의 파라미터 값 v, w의 최적값은 무엇인가?
  4. 학습한 모델이 $(x_1, x_2) = (2, 2)$와 $(x_1, x_2) = (4, 4)$를 올바르게 분류하는지 검증하여라.

  +) iris.csv 데이터를 읽어서 $x_{train}, y_{train}, x_{test}, y_{test}$를 반환하는 함수를 작성하여라.
     단, 함수 내에서 무작위로 순서를 섞은 뒤 7:3으로 train-test split을 하여라.
     은닉층이 10개의 뉴런을 가지는 모델을 생성하고
     100epoch마다 1번씩 train/test 데이터 각각의 error와 accuracy를 출력하여라.