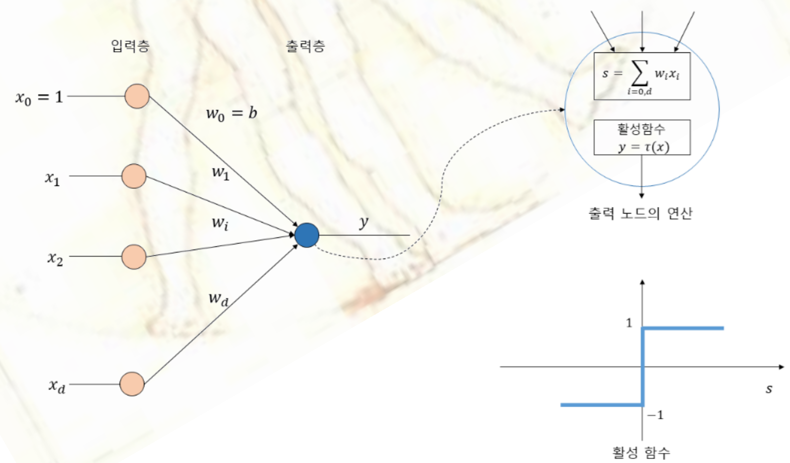
# 신경망

Pattern Recognition & Machine Learning Laboratory
Hyeon-Woo Bae

July. 07, 2021

# 퍼셉트론(1/3)

- 퍼셉트론의 구조
  - 입력층과 출력층으로 구성
    - 입력층 : 입력을 받아 출력층으로 전달하는 역할
      - $d+1$개의 입력층 ($d$ : 특징 벡터의 차원, $1$ : 바이어스 노드 ($x_0 = 1$))
    - 출력층 : 출력을 내보내는 역할
      - $\omega_1$과 $\omega_2$로 분류하는 이진 분류기
      - 실제 구현에서는 $1$과 $-1$, $1$과 $0$으로 표현
    - 입력 노드와 출력 노드는 에지로 연결됨
      - $w$ : 가중치, 연결 강도
  - 출력층에서의 합 계산 및 활성함수(계단함수) 계산
    - $y = \tau(s) = \tau\left(\sum_{i=1}^{d} w_i x_i + b\right) = \tau(\mathbf{w}^T\mathbf{x} + b)$
    - 이때 $\tau(s) = \begin{cases} +1, & s \geq 0 \\ -1, & s < 0 \end{cases}$
      - 퍼셉트론은 활성함수로 계단함수 사용
  - 위 퍼셉트론은 선형 분류기에 해당됨
    - $d(\mathrm{x}) = \mathbf{w}^T\mathbf{x} + b > 0$ 이면 $\mathbf{x} \in \omega_1$
    - $d(\mathrm{x}) = \mathbf{w}^T\mathbf{x} + b < 0$ 이면 $\mathbf{x} \in \omega_2$
    - $d(.)$는 선형 분류기에서의 결정 직선




퍼셉트론의 구조

# 퍼셉트론(2/3)

- 퍼셉트론 학습 알고리즘
  - 분류기 구조 정의 및 분류 과정을 수학식으로 표현
    - $y = \tau(s) = \tau\left(\sum_{i=1}^{d} w_i x_i + b\right) = \tau(\mathbf{w}^T\mathbf{x} + b)$
    - $\tau(s) = \begin{cases} +1, & s \geq 0 \\ -1, & s < 0 \end{cases}$
  - 분류기의 품질을 측정할 수 있는 비용 함수 $J(\Theta)$를 정의
    - 퍼셉트론의 오류율을 비용 함수로 취함
    - $J(\Theta) = \sum_{\mathbf{x}_k \in Y} (-t_k)(\mathbf{w}^T\mathbf{x}_k + b)$
      - $Y$는 틀리게 분류되는 샘플의 집합
      - $\mathbf{x}_k \in \omega_1$ 일때, $t_k = 1$, $\mathbf{w}^T\mathbf{x}_k + b < 0$ (오분류) : $J(\Theta) > 0$
      - $\mathbf{x}_k \in \omega_2$ 일때, $t_k = -1$, $\mathbf{w}^T\mathbf{x}_k + b > 0$ (오분류) : $J(\Theta) > 0$
  - $J(\Theta)$를 최대 또는 최소로 하는 $\Theta$를 찾기 위한 알고리즘을 설계
    - 최적화 알고리즘으로 내리막 경사법을 채택
    - $J(\Theta) = 0$이 될 때까지 작은 값을 갖는 학습률 $\rho(h)$을 곱하여 조금씩 이동시키며 해를 구함
      - $\Theta(h+1) = \Theta(h) - \rho(h)\frac{\partial J(\Theta)}{\partial(\Theta)}$
      - $\frac{\partial J(\Theta)}{\partial \mathbf{w}} = \sum_{\mathbf{x}_k \in Y} (t_k \mathbf{x}_k)$
      - $\frac{\partial J(\Theta)}{\partial b} = \sum_{\mathbf{x}_k \in Y} (-t_k)$

# 퍼셉트론(3/3)

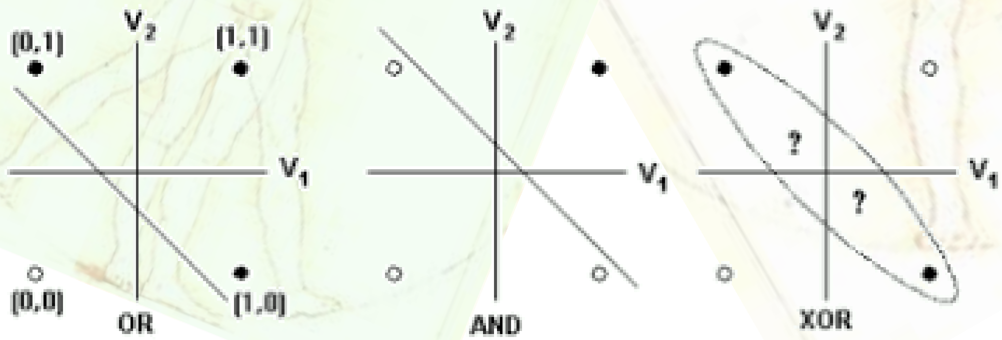
- 퍼셉트론 학습 알고리즘
  - 퍼셉트론 학습 규칙(델타 규칙)
    - $w(h+1) = w(h) + \rho(h)\sum_{x_k \in Y} t_k x_k$
    - $b(h+1) = b(h) + \rho(h)\sum_{x_k \in Y} t_k$

➢ 퍼셉트론 학습 알고리즘은 반드시 수렴한다는 바람직한 특성을 가짐

- 훈련 집합이 선형 분리 가능한 경우에는 초기값으로 무엇을 가지고 출발하던지 반드시 훈련 집합 전체를 옳게 분류하는 퍼셉트론으로 수렴한다는 것이 증명되어 있음

➢ 퍼셉트론의 한계

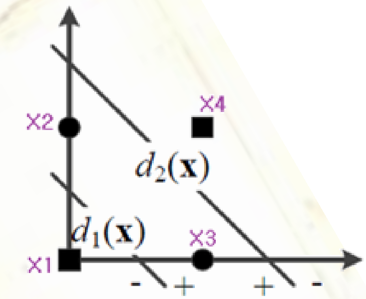
- 선형 분리 불가능한 경우 틀리게 분류하는 경우가 반드시 생김
- **OR, AND** 게이트와 달리 **XOR** 게이트의 동작에서는 샘플이 오분류될 수 밖에 없음




OR AND XOR 게이트

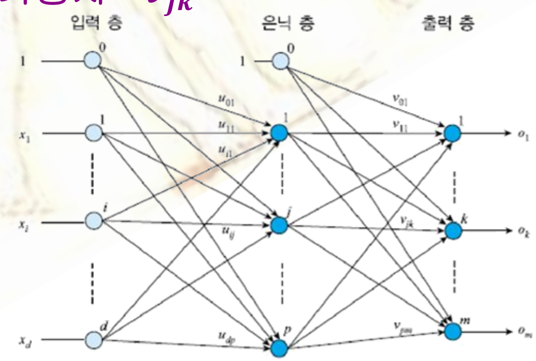
# 다층 퍼셉트론(1/4)

- **XOR 게이트**
  - **XOR 게이트**
    - $d_1(x)$와 $d_2(x)$를 각각 퍼셉트론으로 간주
      - $\mathbf{w}_1^{\mathrm{T}}\mathbf{x} + b_1 > 0$ 이고 $\mathbf{w}_2^{\mathrm{T}}\mathbf{x} + b_2 > 0$ 이면, $\mathbf{x} \in \omega_1$
      - $\mathbf{w}_1^{\mathrm{T}}\mathbf{x} + b_1 < 0$ 이거나 $\mathbf{w}_2^{\mathrm{T}}\mathbf{x} + b_2 < 0$ 이면, $\mathbf{x} \in \omega_2$
    - 새로운 공간에서 $\omega_1$과 $\omega_2$를 최종 분류하는 퍼셉트론 $d_3(x)$



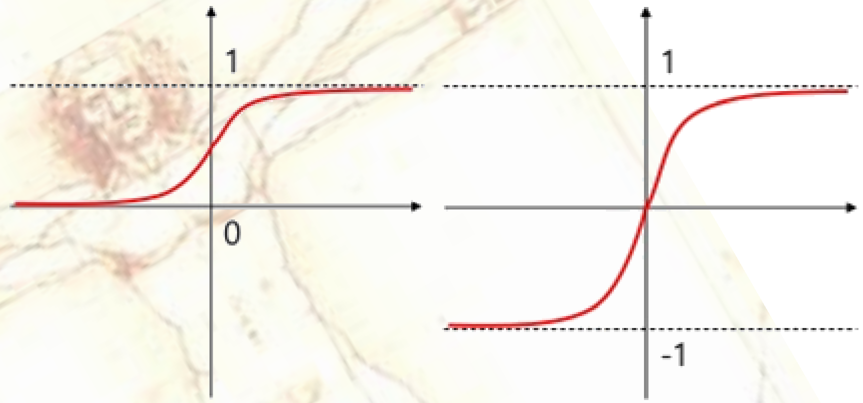

XOR 분류 문제

- **다층 퍼셉트론의 구조**
  - **입력 층, 은닉 층, 출력 층**
    - 입력 층 : 받은 값을 은닉 층으로 전달
    - 은닉 층, 출력 층 : 가중치 합과 활성 함수 계산
    - $i$번째 입력 노드에서 $j$번째 은닉 노드로 가는 에지의 가중치 : $u_{ij}$
    - $j$번째 은닉 노드에서 $k$번째 출력 노드로 가는 에지의 가중치 : $v_{jk}$
      - 은닉 층의 $j$번째 노드, $1 \le j \le p$
      - $z\_sum_j = \sum_{i=1}^{d} x_i u_{ij} + u_{0j}$
      - $z_j = \tau(z\_sum_j)$
      - 출력 층의 $k$번째 노드, $1 \le k \le m$
      - $o\_sum_k = \sum_{j=1}^{p} z_j v_{jk} + v_{0k}$
      - $o_k = \tau(o\_sum_k)$




다층 퍼셉트론의 구조

- 다층 퍼셉트론의 활성 함수
  - 시그모이드 함수
    - 이진 시그모이드 함수
      - 0~1 사이의 값을 가짐
      - $\tau_1(x) = \frac{1}{1+e^{-\alpha x}}$
      - $\tau_1'(x) = \alpha\tau_1(x)(1-\tau_1(x))$
    - 양극 시그모이드 함수
      - -1~1사이의 값을 가짐
      - $\tau_2(x) = \frac{2}{1+e^{-\alpha x}} - 1$
      - $\tau_2'(x) = \frac{\alpha}{2}(1+\tau_2(x))(1-\tau_2(x))$
    - $\alpha$가 클수록 경사가 급함
      - $\alpha = \infty$ : 계단함수와 같은 결과
  - 시그모이드 함수의 장점
    - 전 구간에서 미분가능함
    - 계산이 간단함
    - 미분한 값이 자신을 포함하기 때문에 미분값 역시 계산이 간단함



이진 시그모이드　　양극 시그모이드

# 다층 퍼셉트론(3/4)

- 다층 퍼셉트론 학습 이론
  - 비용 함수 $J(\Theta)$를 정의
    - 훈련 샘플 $\mathbf{x} = (x_1, x_2, \cdots, x_d)^{\mathrm{T}}$ 입력 시 원하는 출력 $\mathbf{t} = (t_1, t_2, \cdots, t_m)^{\mathrm{T}}$
    - 실제 출력 $\mathbf{o} = (o_1, o_2, \cdots, o_m)^{\mathrm{T}}$
    - 오류 $E = \frac{1}{2}\sum_{k=1}^{m}(t_k - o_k)^2$를 줄이는 쪽으로 매개 변수 $u$와 $v$를 수정해 나가야 함
    - $\mathbf{v}(h+1) = \mathbf{v}(h) + \Delta\mathbf{v} = \mathbf{v}(h) - \rho\frac{\partial E}{\partial \mathbf{v}}$
    - $\mathbf{u}(h+1) = \mathbf{u}(h) + \Delta\mathbf{u} = \mathbf{u}(h) - \rho\frac{\partial E}{\partial \mathbf{u}}$
  - $j$번째 은닉 노드와 $k$번째 출력 노드 간의 가중치 갱신값 $\Delta v_{jk}$ 유도
    - $\frac{\partial E}{\partial v_{jk}} = -(t_k - o_k)\tau'(o\_sum_k)z_j$
    - $\delta_k = (t_k - o_k)\tau'(o\_sum_k)z_j, 1 \le k \le m$
    - $\Delta v_{jk} = -\rho\frac{\partial E}{\partial v_{jk}} = \rho\delta_k z_j, 0 \le j \le p, 1 \le k \le m$
  - $i$번째 입력 노드와 $j$번째 은닉 노드 간의 가중치 갱신값 $\Delta u_{ij}$ 유도
    - $\frac{\partial E}{\partial u_{ij}} = -\sum_{k=1}^{m}\delta_k v_{jk}\tau'(z\_sum_j)x_i$
    - $\eta_j = \tau'(z\_sum_j)\sum_{k=1}^{m}\delta_k v_{jk}, 1 \le j \le p$
    - $\Delta u_{ij} = -\rho\frac{\partial E}{\partial u_{ij}} = \rho\eta_j x_i, 0 \le i \le d, 1 \le j \le p$

# 다층 퍼셉트론(4/4)

- 다층 퍼셉트론 학습 이론
  - 알고리즘이 오류 $E$를 줄여나가는 원리
    - 주어진 샘플에 대해 전방 계산으로 오류를 추정
    - 출력 층에서 시작하여 반대 방향으로 전진하며 오류 전파(가중치 갱신)
    - 이를 오류 역전파 알고리즘이라 함
  - 중요한 고려 사항
    - 은닉 노드의 수
      - 은닉 노드의 수가 많으면 변수에 비해 주어진 정보가 부족함
      - 은닉 노드의 수가 적으면 신경망의 용량이 작아 필요한 정보를 충분히 담을 수 없게 됨
    - 초기값 설정
      - 초기값이 수렴 속도에 영향을 미치므로 보편적으로 작은 난수로 설정함
    - 종료 시점 설정
      - 평균 제곱 오차가 어떤 임계값보다 작으면 멈춤
    - 지역 최적 점 탈출
      - 내리막 경사법을 이용하기 때문에 초기값에 따라 지역 최적 점에 빠질 위험이 있음
      - 다중 시작 방법 : 여러 초기값으로 훈련하여 그중 가장 좋은 성능을 보이는 것을 선택
    - 적절히 선택해야할 매개 변수가 많아 실험을 통해 설정할 필요가 있음