

Linear classifier

$$f(x, W) = Wx + b$$

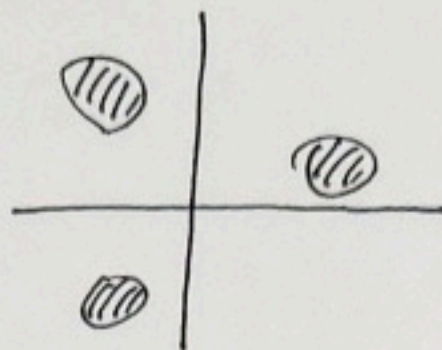
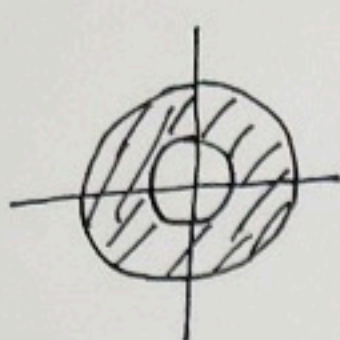
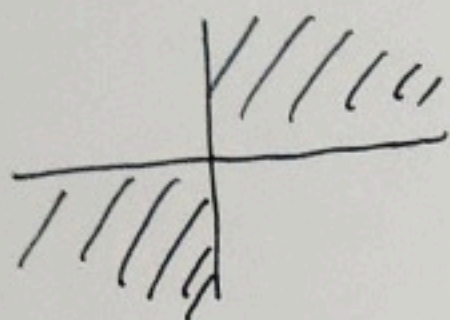
W : 가중치

x : classes

W 를 잘 만들어야 잘 예측

W 를 주어진 데이터로 학습

- 선형이기 때문에 특징에 대해 다음과 같이 그려진다면 예측이 어려움



⇒ 모든 특징을 사용하지 않고
중요한 특징만 선택하면 유리

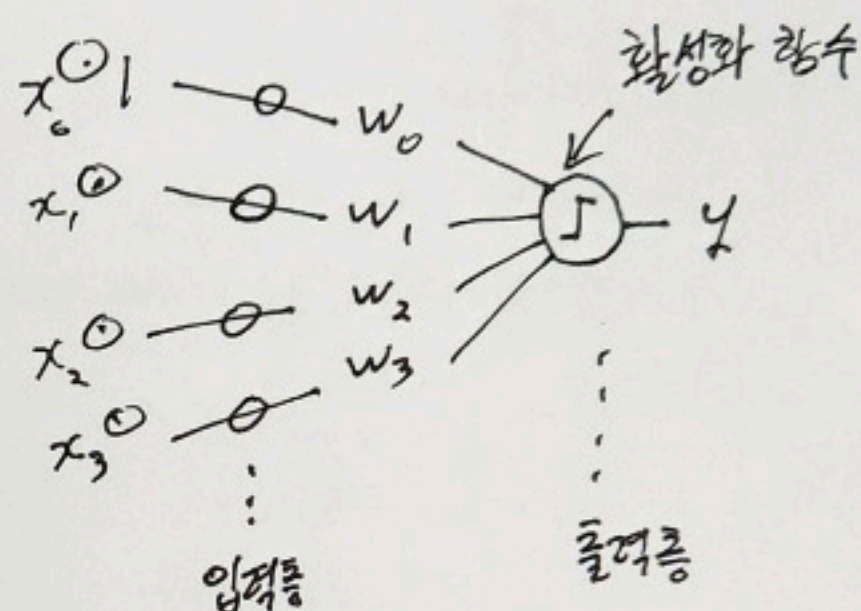
퍼셉트론

- 사람의 뉴런과 유사

- 입력층과 출력층을 가짐

- 특징값과 가중치를 곱한 후에 활성화 함수를 적용

- 최종 출력 값은 +1 or -1



Gradient : 경사하강법

$$J(w) = \sum_{x_k \in \mathcal{D}} -y_k (w^T x_k)$$

$$\frac{\partial J(w)}{\partial w_i} = \sum_{x_k \in \mathcal{D}} \frac{\partial (-y_k (w_0 x_{k0} + w_1 x_{k1} + \dots + w_d x_{kd}))}{\partial w_i}$$

$$= \sum_{x_k \in \mathcal{D}} -y_k x_{ki}, \quad i = 0, 1, \dots, d$$

배치모드 : Batch

- 배치단위로 업데이트

에포크 : epoch (세대)

- 모델이 한번 학습하는 것

Stochastic 모드

- 특정한 샘플이 발생하면 즉시 갱신

- 샘플 순서를 섞음

→ 선형 분리 불가능하면 무한 반복

목적함수 : $J(\theta)$, $J(w)$

- 퍼셉트론의 목적함수 : 오분류된 샘플의 집합

- 조건

- $J(w) \geq 0$
- w 가 최적이면 $J(w) = 0$
- 특정한 샘플이 많으면 $J(w)$ 는 큰 값

$$J(w) = \sum_{x_k \in \mathcal{D}} -y_k (w^T x_k)$$

이미지 분류

문제 • View point variation

• Illumination

• Background clutter

• Occlusion

• Deformation

• Intra class variation

• Context

Nearest Neighbor

- 데이터와 가장 비슷한 값으로 선택

- image data 일 때는 image의 픽셀값의 차이를 가지고 트레이닝 이미지를 선택

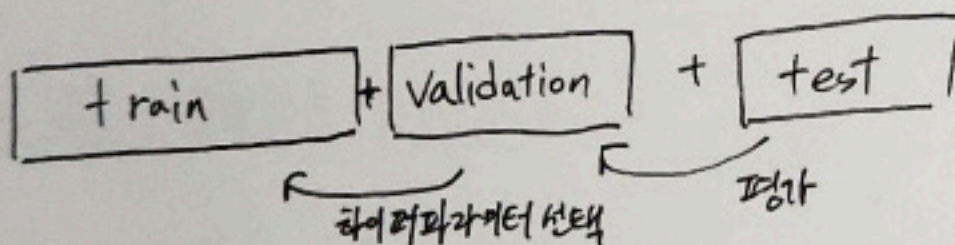
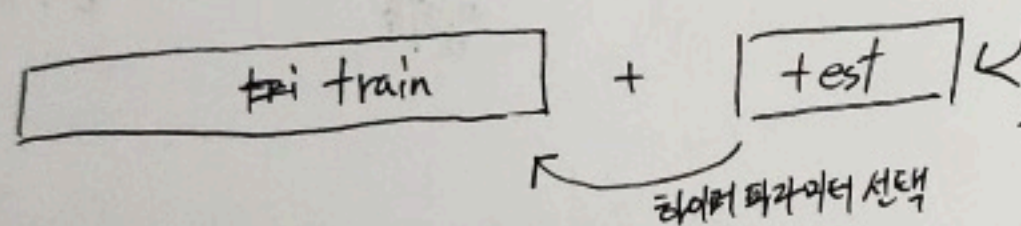
- 무식한 방법 (여러 예외 사항이 발생)

K-Nearest Neighbor

- 하이퍼 파라미터 : K , distance

→ 어떤 파라미터가 좋을지 평가 정수가 높은 파라미터 선택

Setting Hyperparameters



?: 다음 데이터를 넣었을 때 모델이 잘 분류하는지 알지 못함.

더 일반화 시키기 위해

Cross-Validation : 표준단위로 분류

5 fold

