Intro to ML (Lecture - 6)

- k-nearest neighbor classifier
- Perceptron

Structure | Problem setting

① Training samples $\mathcal{X} = \{(x_1,y_1), (x_2,y_2) \dots \}$

(Binary classification) $x_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$ $\{0,1\}$ $\{-1,1\}$

② Test point $\hat{x} \in \mathbb{R}^d$

Aim:

Find $\hat{y}$

find function $f : X \to Y$

$\mathbb{R}^d \to \mathbb{R}$

Any guess | any function is as goods as any other

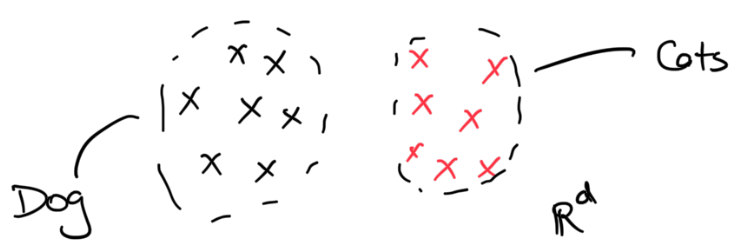
Underlying Assumptions about : Inductive Bias

Applications:

- Image classification | Tagging
- Spam classification
- Mp3 — Audio

① K-nearest neighbor

Bias | Assumption: Similar data points generally lie close to each other.



Algorithm / Psuedo code:

① For test point $\hat{x}$ compute $d(\hat{x}, x_i)$ $\forall\, i \in \{1 \ldots n$

distance b/w $\hat{x}$, x_i

$\| \ \|_2$

$\| \ \|_1$

② $i^* = \underset{i}{\operatorname{argmin}}\ d(\hat{x}, x_i)$

③ $\hat{y} = y_i$

1-nearest neighbor classifier

↓ extend

k- nearest neighbor

Running time each test point

n: training points

$O(nd)$

Problems?

① outliers

② label noise

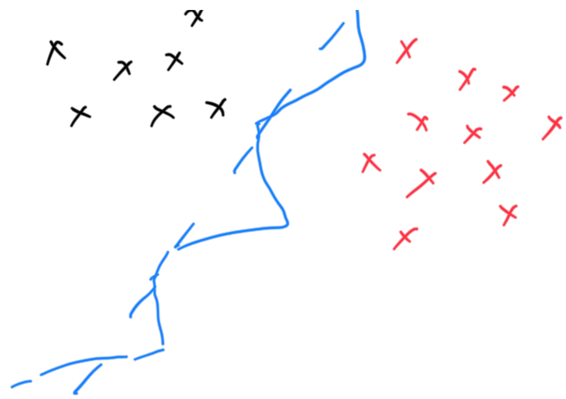
③ $O(nd)$ large

$\cancel{3}\ 224 \times 224 \times 3$

(*) It ends up generating very "complex" decision boundaries

are not the best ones { do not generalize well
- overfit }

"Occam's Razor"



for i in range(N):
    $d(\hat{x}, x_i) = \|x - x_i\|_2$
    if d ≤ current_best
        i_star = i

k=1

k= anything

$\hat{y} = y_i$

⊗ Perceptron :

① Early attempts to solve classification
② 1950's   Frank Rosenblatt
③

Aim/Goal:
Learn a linear separator b/w two classes

Setting:

$\mathcal{X} = \{(x_1, y_1), (x_2, y_2) \ldots (x_n, y_n)\}$

$x_i \in \mathbb{R}^d \qquad y_i \in \mathbb{R} \quad \{$

Binary $\quad y_i = \{-1, 1\}$

Aim

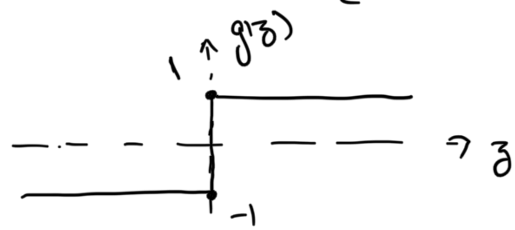
$$Y = W^T X + b$$
$$= W^T X$$

Problem:

Y is not really labels

$$g(z) = \{ \; 1 \quad \text{if } z \geq 0$$

$\quad\quad -1 \quad$ otherwise



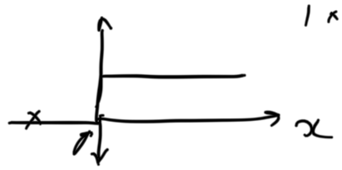
Loss: (simplest possible)

$L = 1$ if misclassify
$L = 0$ if classified correctly

$$L = \text{step}(-\underbrace{y_i}_{1} \underbrace{f(x_i)}_{\times 0.5}) \qquad f(x_i) = w^T x$$

step(-0.5)

step



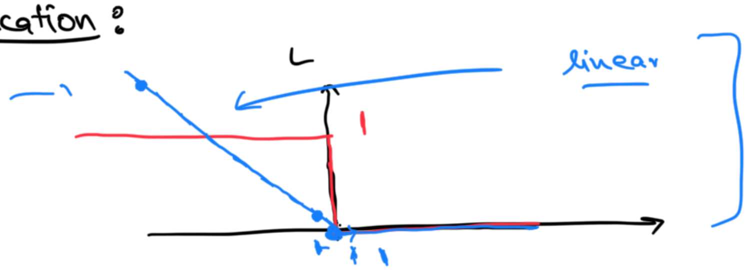
y_i	$f(x_i) = w^T x$	Loss
1	0.5	0
-1	-0.5	0
1	-0.5	1

$y_i f(x_i)$

Update: (grdient descent)

$$\frac{\partial L}{\partial w} = \begin{cases} 0 & \text{everywhere} \\ \bullet & \text{label flip} \end{cases}$$

Modification:



$$L(y, f(x,w)) = \begin{cases} -y_i f(x_i) \\ 0 & \text{correct} \end{cases} \qquad -y_i w^T x$$

$\partial L \quad \{ \quad -y_i x_i \quad$ incorrectly classified

$$\overline{\partial w} = \begin{cases} 0 & \text{otherwise} \end{cases}$$

$$w^{t+1} = w^t + \eta(y_i x_i)$$

Algorithm:

1. $w_0 = 0$
2. for epoch in range (10000):
   for (x_i, y_i) in X:
   if $f(x_i) \neq y_i$:
   update
   ~~else~~
3. all data are correctly classified: terminate

Problems:

1. previously correctly classified example can be misclassified.
2. No. of iteration

Theorem: assumption:

1. $\|x\| \leq 1$ ←
2. $b=0$
3. Data is perfecty separable

If there exists a solution then perceptron algorithm is guaranteed to find it.

[converge to 0 error in finite time]

Pf: → $w^0 = 0$ $\quad w^* =$ optimum

$\|x_i\| \leq 1$

margin — $\gamma = \arg\min_i |w^T x|$ $\quad \gamma > 0$

y_i	$w^T x$
1	0.1
1	0.5
1	10

$\eta = 1$

① $(w^*)^T w^t = (w^*)^T (w^{t-1} + y_i x_i)$

$= (w^*)^T w^{t-1} + y_i (w^*)^T x_i$

$\geq \underline{(w^*)^T w^{t-1}} + \gamma$

recursively

$$\boxed{(w^*)^T w^t \geq t\gamma}$$

② $\|w^t\|^2 = \|w^{t-1} + y_i x_i\|^2$

$= \|w^{t-1}\|^2 + \|y_i x_i\|^2 + \underline{2 y_i w^{t-1} x_i}$

$\underline{2 y_i w^{t-1} x_i} < 0$

$\leq \|w^{t-1}\|^2 + \|x_i\|^2 = 1$

$\leq t$

③ Angle b/w optimal w^* & current solution

$1 \geq \cos(w^*, w^t) = \dfrac{(w^*)^T w^t}{\|w^*\| \|w^t\|}$ from ①

from ②

$t\gamma \leq \|w^*\| \, \|w^t\|$

$\leq \|w^*\| \sqrt{t}$

$\sqrt{t} \leq \dfrac{\|w^*\|}{\gamma}$ $\qquad t \leq \dfrac{1}{\gamma^2} \|w^*\|^2$

if $\|w^*\| = 1$ $\qquad \boxed{t \leq 1/\gamma^2}$

No. of update

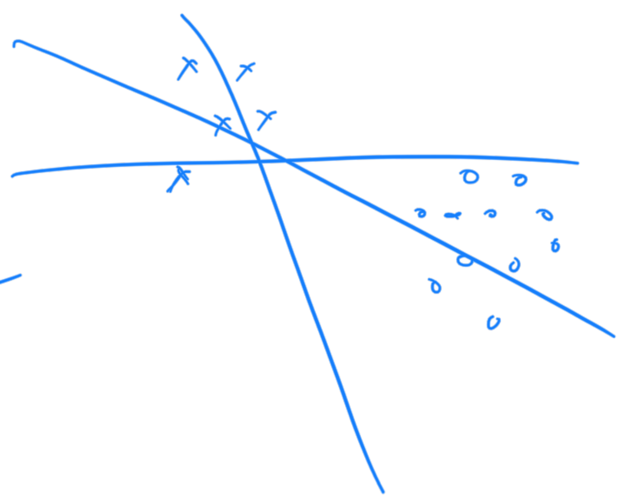
Perceptron vs logistic :

① Differences :

→ output – probabilities

① if solution exist both of them will find it.

② logistic regression → motivated statistics



w=0 x