Lec 3.1 Binary Linear classification

1. Basic setup:

↳ Training set

$\mathcal{D} = \{ (\underline{x}_1, y_1), (\underline{x}_2, y_2), \cdots (\underline{x}_N, y_N) \}$,

where $\underline{x}_n \in \mathbb{R}^d$, $\underline{x}_n = (x_{n1}, x_{n2}, \cdots x_{nd})$

$y_n \in \{-1, +1\}$ // binary label

↳ Task:

Given any $\underline{x} \in \mathbb{R}^d$, output $y \in \{-1, +1\}$

↳. (Hypothesis $\mathcal{H}$) Decision Rule:

Weighted vector $\underline{w} = (w_1, w_2, \dots w_d) \in \mathbb{R}^d$

Threshold $b \in \mathbb{R}$

Given data point $\underline{x} = (x_1, x_2 \cdots x_d)$

if $\sum_{i=1}^{d} w_i x_i > b$, then $y = +1$

if $\sum_{i=1}^{d} w_i x_i < b$, then $y = -1$

if $\sum_{i=1}^{d} w_i x_i = b$, output either $+1$ or -1 (unimportant)

↳. Training phase:

- Compare decision rule with training data, to choose the "best" parameter values for decision rule —— "best" hypothesis.
- Given $\mathcal{D}$, find $(\underline{w}, b)$ to minimize the training errors

$$E_{in}(\underline{w}, b) = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}(y_n \neq \hat{y}_n(\underline{w}, b))$$

avg error on training set for given $(\underline{w}, b)$

take Average

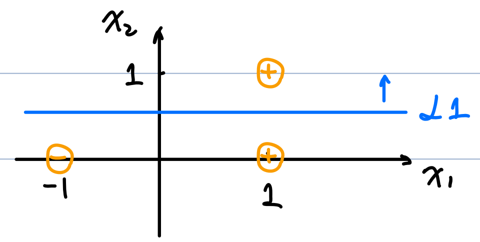
output of decision rule on example $\underline{x}_n$

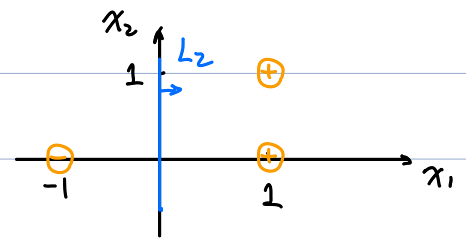
true label for $\underline{x}_n$

↳ <u>Goal</u>: find ($\underline{w}$, b) to minimize $E_{in}(\underline{w}, b)$

2. <u>Example</u>:

$d=2$ (i.e. two attributes x_1, x_2)





↳ consider $L1$: $x_2 = \frac{1}{2}$

- This correspond to : $(0)\, x_1 + (1)\, x_2 = \frac{1}{2}$

  $\therefore\ \underline{w} = (0, 1)$, $b = \frac{1}{2}$

- Find $E_{in}(\underline{w}, b)$

  1' $\underline{x}_1 = (1, 0)$, $y_1 = +1$

  $w^T x_1 = 0 < b \Rightarrow \hat{y}_1 = -1$ // Make one error

  2' $\underline{x}_2 = (1, 1)$, $y_2 = +1$

  $w^T x_2 = 1 > b \Rightarrow \hat{y}_2 = +1$

  3' $\underline{x}_3 = (-1, 0)$, $y_3 = -1$

  $w^T x_3 = 0 < b \Rightarrow \hat{y}_3 = -1$

  $\therefore\ E_{in}(\underline{w}, b) = \frac{1}{3}(1 + 0 + 0) = \frac{1}{3}$

↳ Try L_2 : $x_1 = 0$

$\underline{w} = (1, 0)$, $b = 0$

$E_{in}(\underline{w}, b) = 0$

↳. Minimize $E_{in}(\underline{w}, b)$ is an NP_hard problem in general.

i.e. No efficient algo to solve it

Lec 3.2 Perception Learning Algo (PLA)

↳ efficiently finds a perfect classifer for linearly sepertable data.

1. change of notation:

↳ Decision Rule:

$$w_1 x_1 + w_2 x_2 + \cdots + w_d x_d \underset{y=-1}{\overset{y=+1}{\gtrless}} b$$

$$\Rightarrow -b + w_1 x_1 + w_2 x_2 + \cdots + w_d x_d \underset{y=-1}{\overset{y=+1}{\gtrless}} 0$$

- Let $w_0 = -b$, $x_0 = 1$

  Then we have an <u>Augmented vectors</u>:

  $$\underline{w} = (w_0, w_1, \cdots w_d) \in \mathbb{R}^{d+1}$$

  $$\underline{x} = (x_0 = 1, x_1, \cdots x_d) \in \mathbb{R}^{d+1}$$

- Then the new decision rule becomes:

  $$\underline{w}^T \underline{x} \underset{y=-1}{\overset{y=+1}{\gtrless}} 0$$

  and $y = h_w(\underline{x}) \triangleq \text{sign}(\underline{w}^T \underline{x})$ ← "perceptron learning"

  // $\text{sign}(t) = \begin{cases} +1, & t < 0 \\ -1, & t < 0 \end{cases}$

2. PLA:

↳ suppose training set $\mathcal{D}$ is linearly separable. find $\underline{w} \in \mathbb{R}^{d+1}$ s.t.

$$E_{in}(\underline{w}) = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}(y_n \neq h_w(\underline{x}_n)) = 0$$

↳ Input: Training set $\mathcal{D}$ that is linearly seperable

Output: $\underline{w} \in \mathbb{R}^{d+1}$ that achieves $E_{in}(\underline{w}) = 0$

Initialization: arbitary e.g. $\underline{w} = \underline{0}$ // d- dimentional zero vector

b. step 1: check whether $E_{in}(\underline{w}) = 0$

If yes, stop and output $\underline{w}$.

step 2: Let $(\underline{x}_n, y_n)$ be a miss-classified example

(including points on boundary)

If $y_n = +1$, then replace $\underline{w}$ with $\underline{w} + x_n$

If $y_n = -1$, then replace $\underline{w}$ with $\underline{w} - x_n$

same as saying

$\underline{w} \leftarrow \underline{w} + y_n x_n$

Then go to step 1.