Perceptron Learning Algorithm: Theory and Practice

Jephian C.-H. Lin 林晉宏

Department of Applied Mathematics, National Sun Yat-sen University

February 17, 2023
2023 One Day Workshop on Combinatorics and Graph Theory

How to select the students?

student subject	A	B	C	D	E	decision
1	10	10	10	10	10	accept
2	10	10	10	10	0	accept
3	0	0	15	0	0	decline

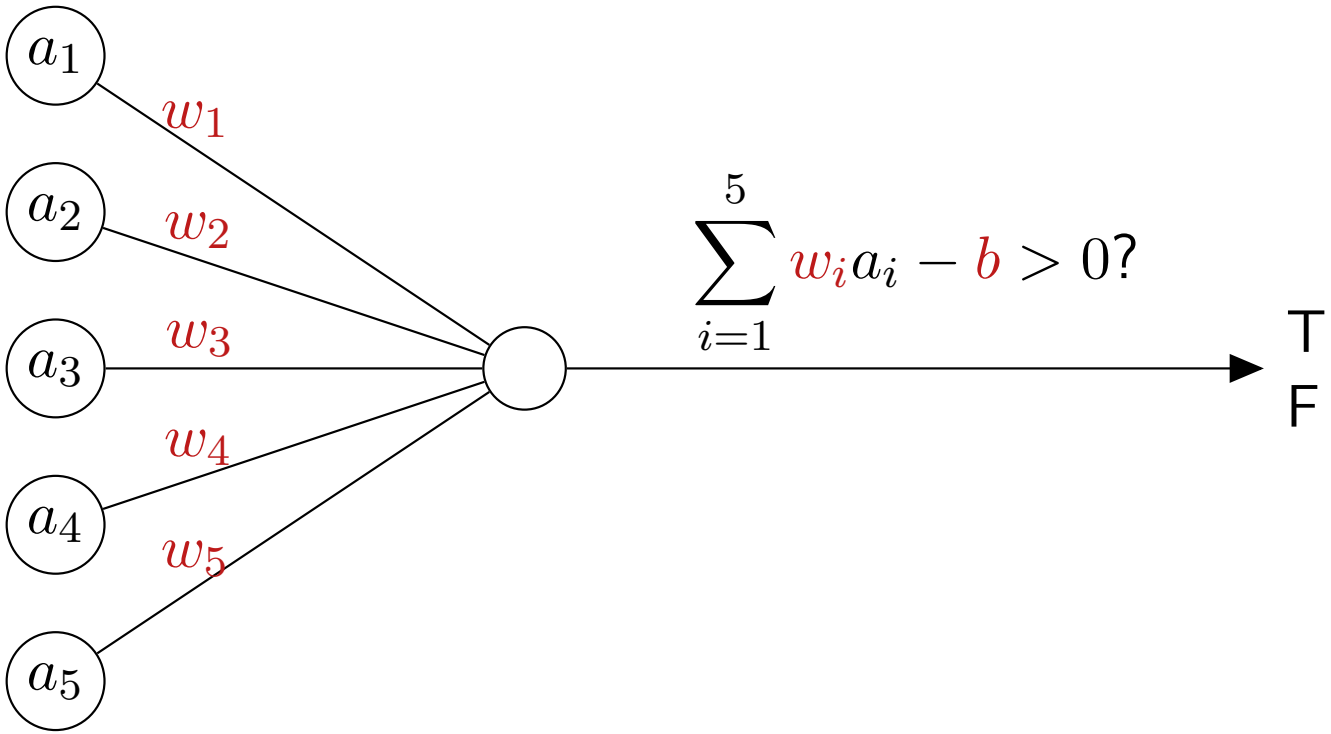
Sample criterion: $1 \cdot A + 1 \cdot B + 2 \cdot C + 1 \cdot D + 0 \cdot E > 40$?

How to select the students?

student subject	A	B	C	D	E	decision
1	10	10	10	10	10	accept
2	10	10	10	10	0	accept
3	0	0	15	0	0	decline

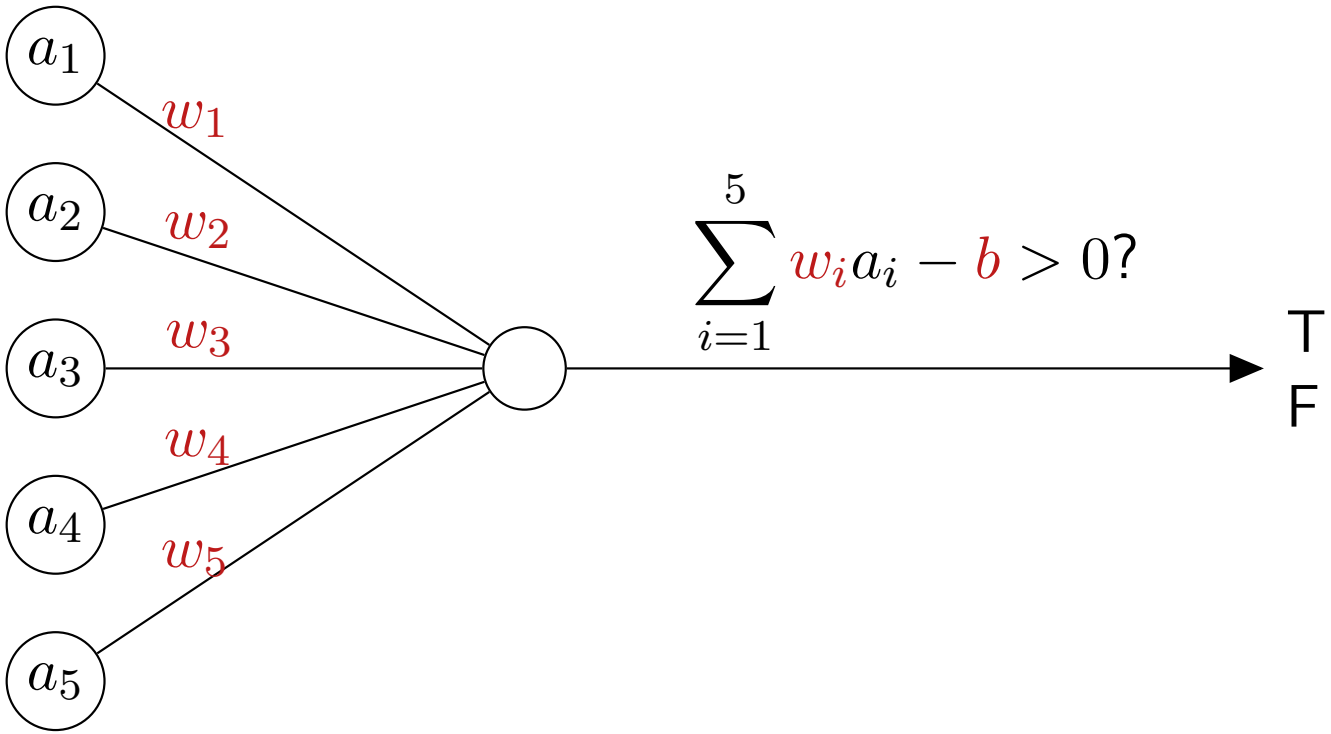
Sample criterion: $1 \cdot A + 1 \cdot B + 2 \cdot C + 1 \cdot D + 0 \cdot E > 40$?

Perceptron



A perceptron takes a weighted input and decides if it passes the threshold.

Perceptron

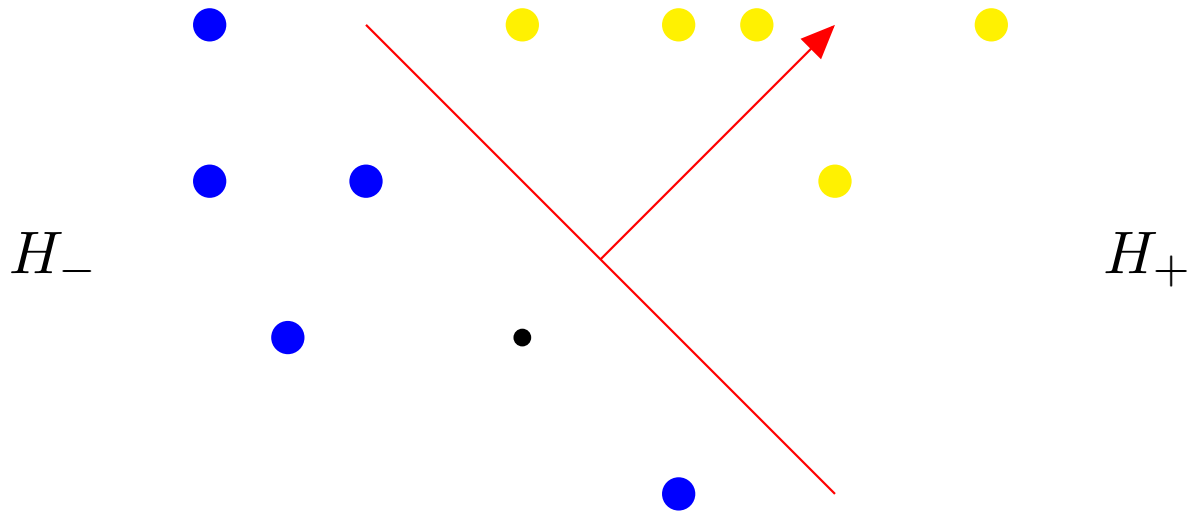


A perceptron takes a weighted input and decides if it passes the threshold.

Affine hyperplane

Given a normal vector $\mathbf{v}$ (weights) and a bias b (threshold), define:

$$
\begin{aligned}
H_+ &= \{\mathbf{x} : \langle \mathbf{x}, \mathbf{v} \rangle - b > 0\}, \\
H_0 &= \{\mathbf{x} : \langle \mathbf{x}, \mathbf{v} \rangle - b = 0\}, \text{ and} \\
H_- &= \{\mathbf{x} : \langle \mathbf{x}, \mathbf{v} \rangle - b < 0\}.
\end{aligned}
$$

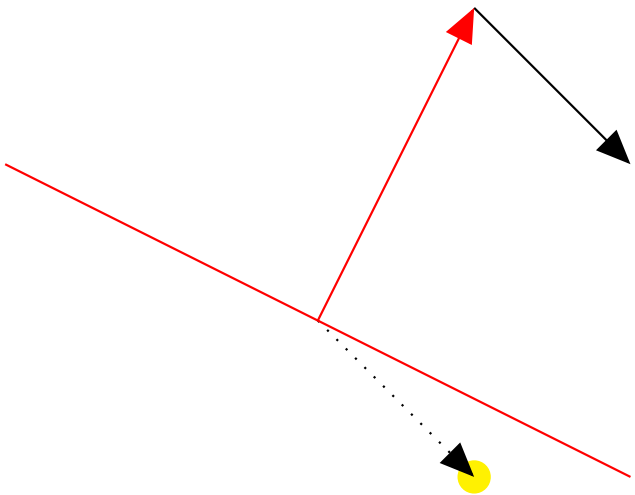


Training: data with labels $\rightarrow$ affine hyperplane

Prediction: data without labels + affine hyperplane $\rightarrow$ labels

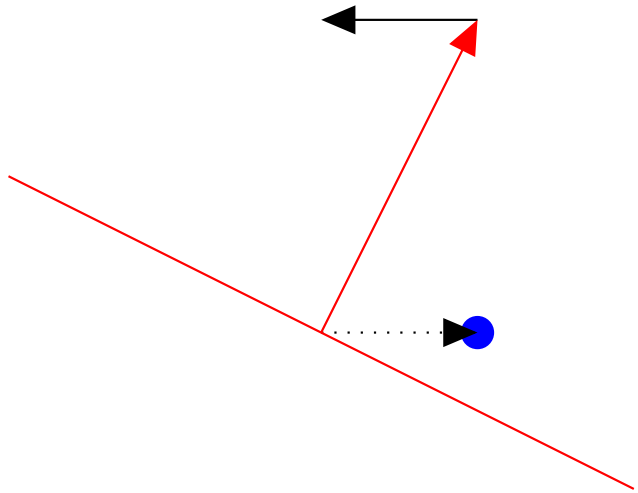
How to partition the points? (assume $b = 0$)

Wrong: $+$ points in H_-



$\mathbf{v} \leftarrow \mathbf{v} + \mathbf{x}_i$

Wrong: $-$ points in H_-

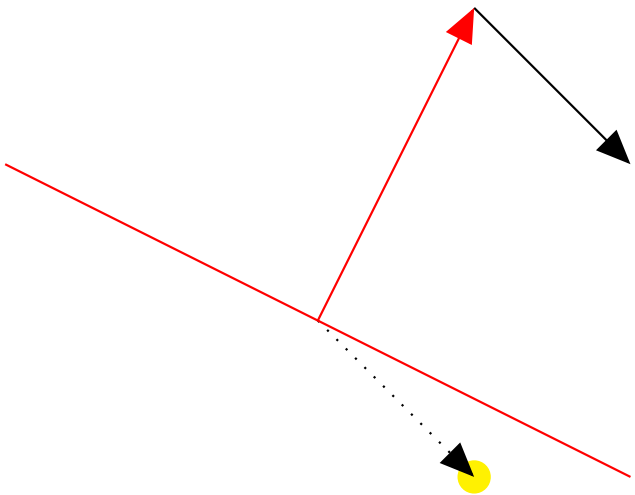


$\mathbf{v} \leftarrow \mathbf{v} - \mathbf{x}_i$

$\mathbf{v} \leftarrow \mathbf{v} + y_i \mathbf{x}_i$

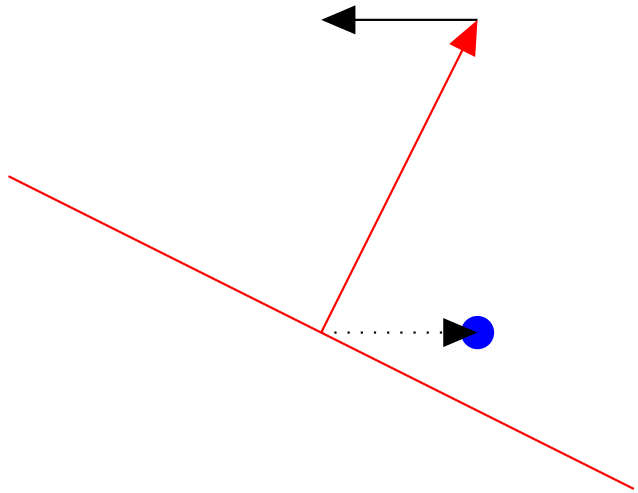
How to partition the points? (assume $b = 0$)

Wrong: $+$ points in H_-



$\mathbf{v} \leftarrow \mathbf{v} + \mathbf{x}_i$

Wrong: $-$ points in H_-



$\mathbf{v} \leftarrow \mathbf{v} - \mathbf{x}_i$

$$\mathbf{v} \leftarrow \mathbf{v} + y_i \mathbf{x}_i$$

Perceptron Learning Algorithm (without bias)

Algorithm

Input: N *points* $\mathbf{x}_i \in \mathbb{R}^d$ *and* N *labels* $y_i \in \{-1, 1\}$

Output: *a normal vector* $\mathbf{v}$ *such that* $\langle \mathbf{x}_i, \mathbf{v} \rangle \cdot y_i > 0$ *for all* i

Steps:

1. $\mathbf{v} \leftarrow \mathbf{0} \in \mathbb{R}^d$.
2. *Find a point* $\mathbf{x}_{i_0}$ *such that* $\langle \mathbf{x}_{i_0}, \mathbf{v} \rangle \cdot y_{i_0} \leq 0$. *Then* $\mathbf{v} \leftarrow \mathbf{v} + y_{i_0} \mathbf{x}_{i_0}$.
3. *Repeat Step 2 until nothing found. Return* $\mathbf{v}$.

If the points are separable, can the algorithm always find the affine hyperplane and stop?

Perceptron Learning Algorithm (without bias)

Algorithm

Input: N *points* $\mathbf{x}_i \in \mathbb{R}^d$ *and* N *labels* $y_i \in \{-1, 1\}$

Output: *a normal vector* $\mathbf{v}$ *such that* $\langle \mathbf{x}_i, \mathbf{v} \rangle \cdot y_i > 0$ *for all* i

Steps:

1. $\mathbf{v} \leftarrow \mathbf{0} \in \mathbb{R}^d$.
2. *Find a point* $\mathbf{x}_{i_0}$ *such that* $\langle \mathbf{x}_{i_0}, \mathbf{v} \rangle \cdot y_{i_0} \leq 0$. *Then* $\mathbf{v} \leftarrow \mathbf{v} + y_{i_0} \mathbf{x}_{i_0}$.
3. *Repeat Step 2 until nothing found. Return* $\mathbf{v}$.

If the points are separable, can the algorithm always find the affine hyperplane and stop?

Proof of the correctness

Suppose the points are separable by $\mathbf{u}$ with $\|\mathbf{u}\| = 1$. Assume

$$R = \max_i \|\mathbf{x}_i\| \text{ and } r = \min_i |\langle \mathbf{u}, \mathbf{x}_i \rangle|.$$

Claim 1: After k steps, $\langle \mathbf{u}, \mathbf{v} \rangle \geq kr$.
Claim 2: After k steps, $\|\mathbf{v}\| \leq \sqrt{k}R$.

As a consequence,

$$\cos\theta = \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\|\|\mathbf{v}\|} \geq \frac{kr}{\sqrt{k}R}$$

becomes unbounded if the algorithm never stops. So the algorithm must find a solution to terminate Step 2 unless the separable assumption is wrong.

Claim 1: After k steps, $\langle \mathbf{u}, \mathbf{v} \rangle \geq kr$.

True for $\mathbf{v}_0 = \mathbf{0}$.

Recall $r = \min_i | \langle \mathbf{u}, \mathbf{x}_i \rangle |$.

By induction,

$$\begin{aligned} \langle \mathbf{u}, \mathbf{v}_{k+1} \rangle &= \langle \mathbf{u}, \mathbf{v}_k + y_i \mathbf{x}_i \rangle \\ &= \langle \mathbf{u}, \mathbf{v}_k \rangle + y_i \langle \mathbf{u}, \mathbf{x}_i \rangle \\ &\geq kr + r = (k+1)r. \end{aligned}$$

Claim 2: After k steps, $\|\mathbf{v}\| \leq \sqrt{k}R$.

True for $\mathbf{v}_0 = \mathbf{0}$.

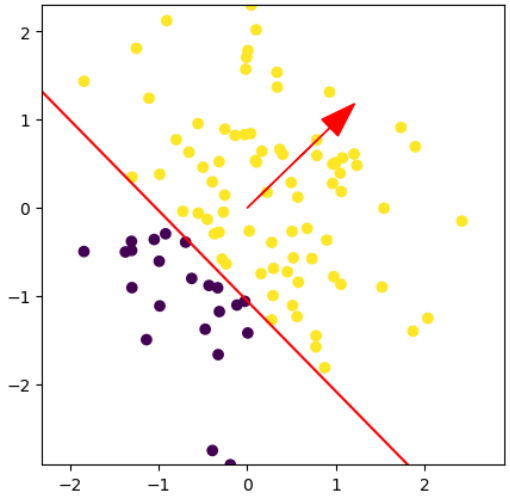
Recall $R = \max_i \|\mathbf{x}_i\|$.

By induction,

$$\begin{aligned}
\|\mathbf{v}_{k+1}\|^2 &= \langle \mathbf{v}_k + y_i\mathbf{x}_i, \mathbf{v}_k + y_i\mathbf{x}_i \rangle \\
&= \|\mathbf{v}_k\|^2 + \|\mathbf{x}_i\|^2 + 2y_i \langle \mathbf{v}_k, \mathbf{x}_i \rangle \\
&\leq (\sqrt{k}R)^2 + R^2 + 0 = (k+1)R^2.
\end{aligned}$$

Perceptron Learning Algorithm with bias

$$(x_1, x_2) \to (1, x_1, x_2)$$



- Expand $\mathbf{x}_i$ as $\hat{\mathbf{x}}_i = (1, \mathbf{x}_i)$.
- Run PLA without bias on $\hat{\mathbf{x}}_i$ and get the normal vector $(b, \mathbf{v})$.
- Since

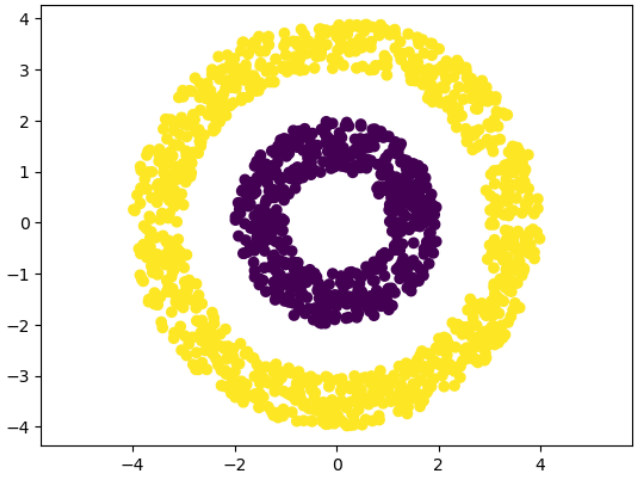
$$\langle (1, \mathbf{x}_i), (b, \mathbf{v}) \rangle = \langle \mathbf{x}_i, \mathbf{v} \rangle + b,$$

this gives the criterion:

$$\langle \mathbf{x}_i, \mathbf{v} \rangle + b > 0?$$

Adding new features

$$(x_1, x_2) \rightarrow (1, x_1, x_2, x_1^2, x_2^2)$$



Output criterion:

$$c_1 x_1^2 + c_2 x_2^2 + d_1 x_1 + d_2 x_2 + b > 0?$$

Summary

Nice things about PLA:

- Simple and easy to be implemented.
- Easy to interpret the output.
- It is a building block for a neural network.

However, it cannot tell you if a dataset is separable by an affine hyperplane.

Summary

Nice things about PLA:

- Simple and easy to be implemented.
- Easy to interpret the output.
- It is a building block for a neural network.

However, it cannot tell you if a dataset is separable by an affine hyperplane.

Some exmples

In Memory of Professor Li-Da Tong

Thank you!

In Memory of Professor Li-Da Tong

Thank you!