Data Mining & Machine Learning

CS37300
Purdue University

Sep 18 , 2023

Today's topics

- Linear classifiers
  - Separable case (today)
  - Non-separable (Wednesday)

Binary Classification

- We have a **training data set** **S** = $\{(x_1,y_1),\ldots,(x_n,y_n)\}$ of pairs (x,y)
- x: feature vector $x \in \mathbb{R}^d$
- y: **binary** labels: $y \in \{0,1\}$ or $y \in \{-1,1\}$
- Examples:
    - x = Email, y = 1 for Spam, 0 for Not Spam
    - x = Image, y = 1 if image contains a hot dog, 0 if not
    - x = Audio clip, y = 1 if contains voice saying ‘Alexa’, 0 if not

Linear Classification

- A simple representation for classifiers: **linear classifier**

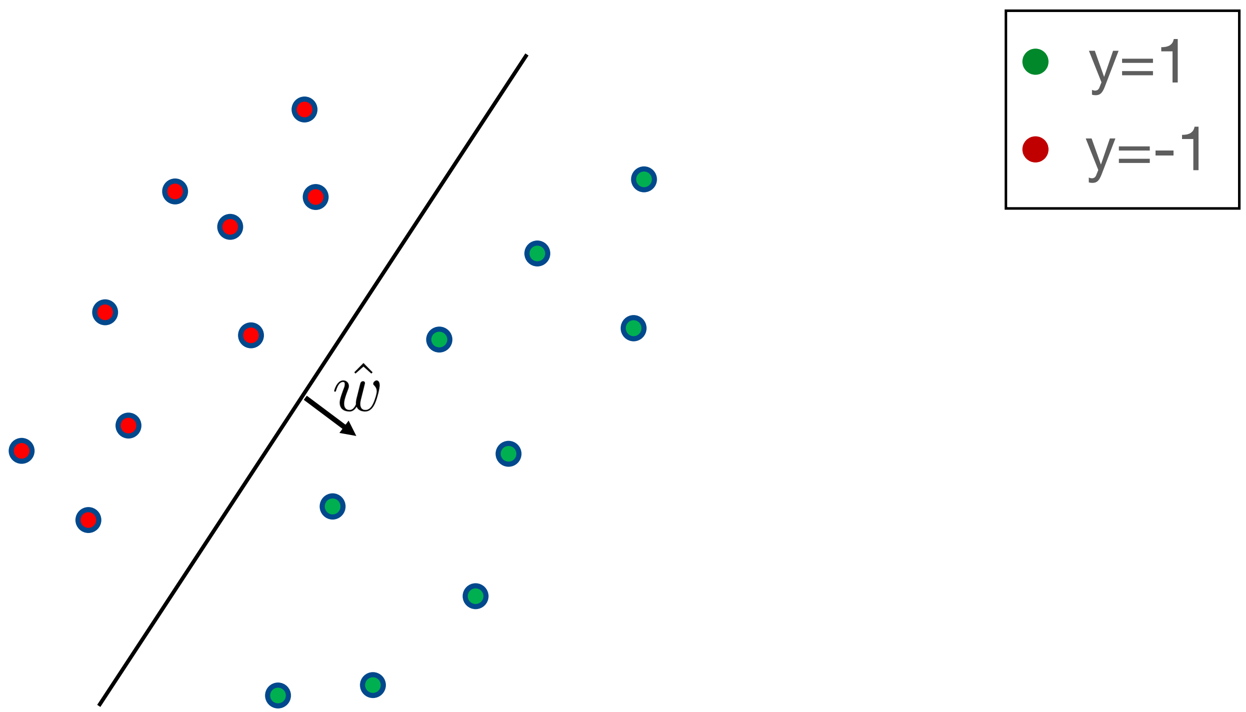
- Learn parameters $\widehat{w} \in \mathbb{R}^d$ and $\widehat{b} \in \mathbb{R}$

- After training, classify any new point as

$$\hat{h}(x) = \mathrm{sign}\Big(\hat{w}^\top x + \hat{b}\Big) \in \{-1, 1\}$$

Linear Classification: Geometric Interpretation

$$\hat{h}(x) = \text{sign}\left(\hat{w}^\top x + \hat{b}\right) \in \{-1, 1\}$$



- Note: In this case, we have zero training error

Linear Separability

- Generally, for any $\mathrm{w} \in \mathbb{R}^d$ and $\mathrm{b} \in \mathbb{R}$ define a linear classifier:

$$h_{w,b}(x) = \mathrm{sign}\left(w^\top x + b\right)$$

- We say S is **linearly separable** if there exists (w, b) such that

$$h_{w,b} \text{ is correct on all training data}$$

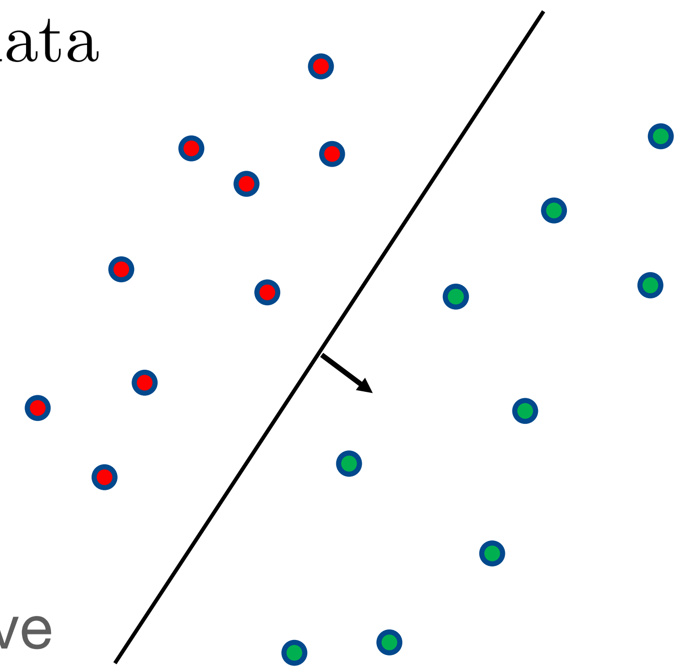
- In other words, every $(x_i, y_i) \in S$ has

$$\mathrm{sign}(w^\top x_i + b) = y_i$$

- Geometrically, the hyperplane

$$\{x : w^\top x + b = 0\}$$

perfectly separates the positive and negative data points

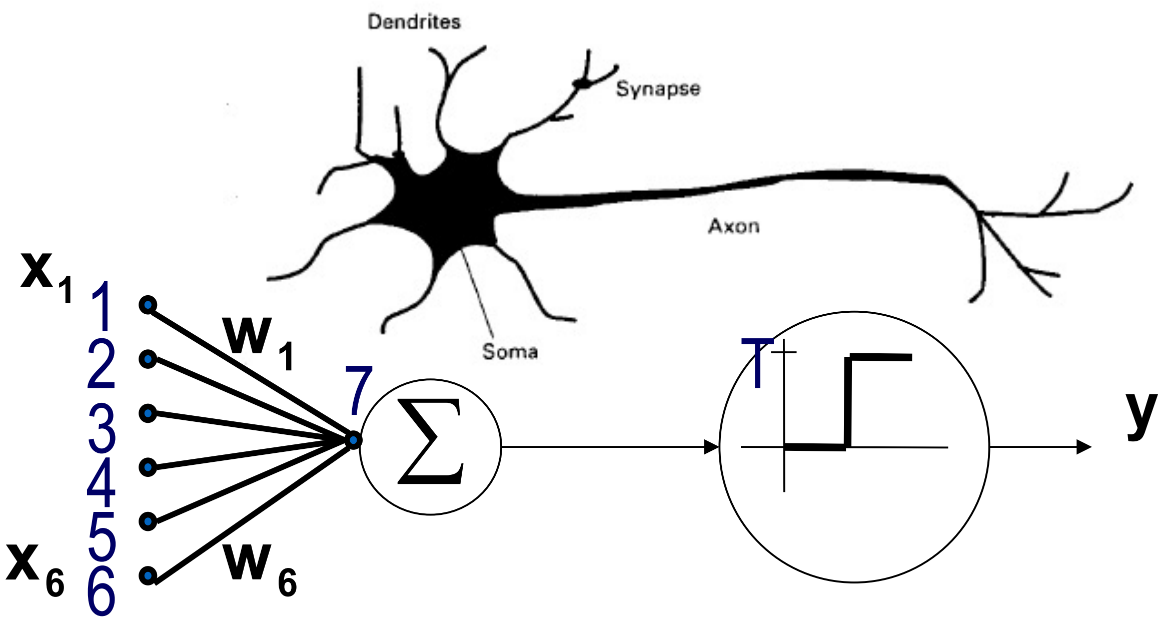


Perceptron: Biological motivation

- How can we find a good $(\hat{w}, \hat{b})$?

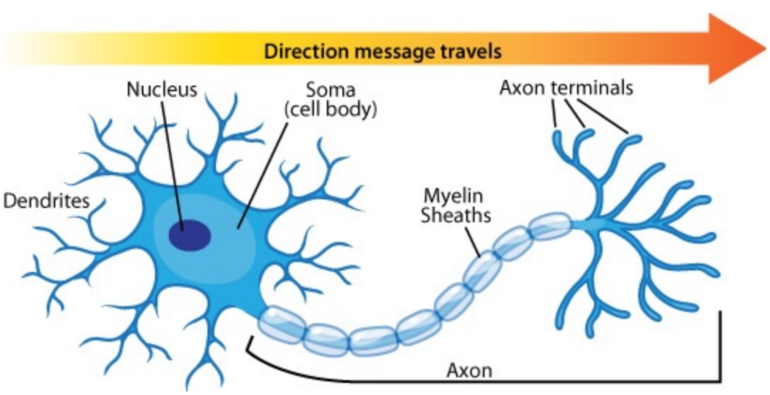
- **Rosenblatt** suggested that when a target output value is provided for a single neuron with fixed input, it can incrementally change weights and learn to produce the output using the Perceptron learning rule

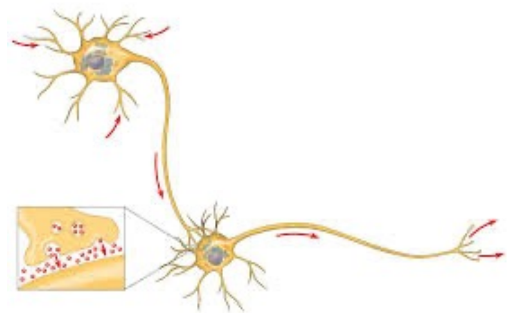
- Perceptron = Linear threshold unit

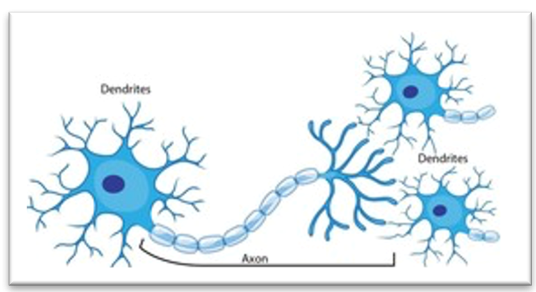


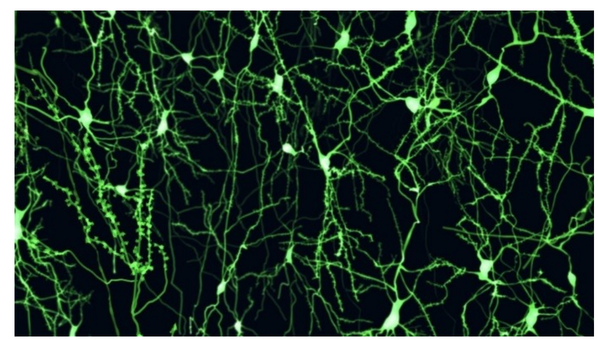
Neurons

- Neuron
  - takes inputs from many other sources (neurotransmitters into dendrites)
  - Each input signal is attenuated by some learned amount
  - If the aggregate of these inputs exceeds some threshold, the neuron "fires", sending out a signal (neurotransmitters from its axon terminals) to all the other neurons connected to it



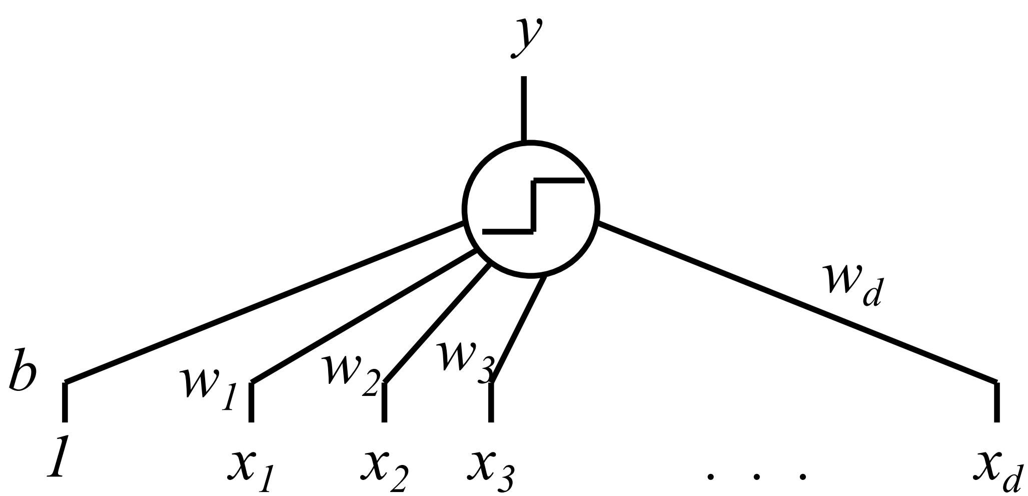






Neurons

- Abstracting this: A mathematical neuron
  - Takes numerical inputs from other sources
    (either inputs or other neurons… we'll cover that later)
  - Each input signal is weighted by a learned real value
  - If the sum of weighted inputs exceeds a threshold, neuron outputs 1 (fire), else 0 (don't fire).



$$y = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} w_i x_i + b \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

- Notice: **A neuron is a linear classifier!**

Perceptron Learning Algorithm

Perceptron Algorithm:

1. Initialize $w_0 = 0$, $b_0 = 0$, $m = 0$
2. For $t = 1, 2, \ldots, n, 1, 2, \ldots, n, \ldots$ (until no more mistakes)
3. If $\text{sign}(w_m^\top x_t + b_m) \neq y_t$ (mistake)
4. Update $w_{m+1} = w_m + y_t x_t$
5. Update $b_{m+1} = b_m + y_t$
6. $m \leftarrow m + 1$

Perceptron Learning Algorithm

Perceptron Algorithm:

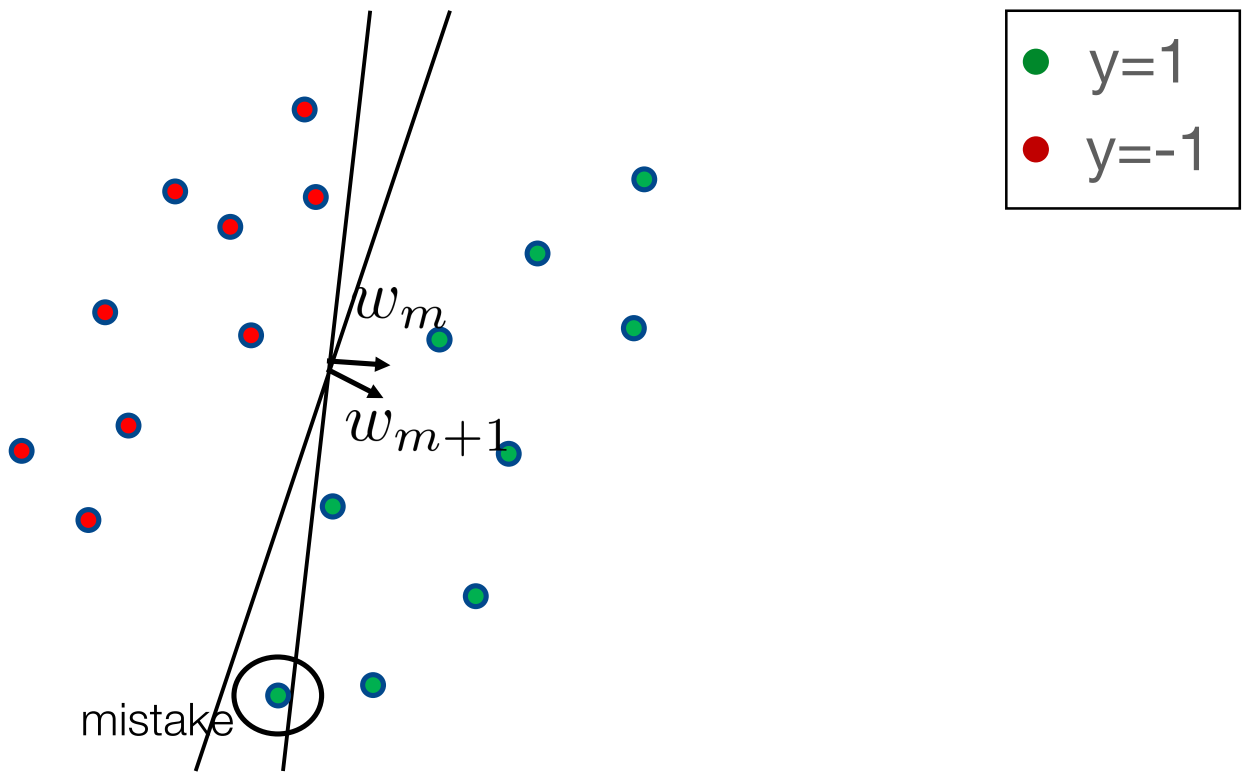
1. Initialize $w_0 = 0$, $b_0 = 0$, $m = 0$
2. For $t = 1, 2, \ldots, n, 1, 2, \ldots, n, \ldots$ (until no more mistakes)
3. If $\text{sign}(w_m^\top x_t + b_m) \neq y_t$ (mistake)
4. Update $w_{m+1} = w_m + y_t x_t$
5. Update $b_{m+1} = b_m + y_t$
6. $m \leftarrow m + 1$

- If the data are linearly separable, this algorithm will find a good (w,b)

Perceptron

Perceptron Algorithm:

1. Initialize $w_0 = 0$, $b_0 = 0$, $m = 0$
2. For $t = 1, 2, \ldots, n, 1, 2, \ldots, n, \ldots$ (until no more mistakes)
3. If $\text{sign}(w_m^\top x_t + b_m) \neq y_t$ (mistake)
4. Update $w_{m+1} = w_m + y_t x_t$
5. Update $b_{m+1} = b_m + y_t$
6. $m \leftarrow m + 1$



- In the end, it separates the data
- But is this the best linear separator?
- Intuitively seems better to be far from all the data points

Margin

- Claim: For linearly separable data, Perceptron makes a finite number of updates
- The sequence is linearly separable, so let w_*, b_* be such that

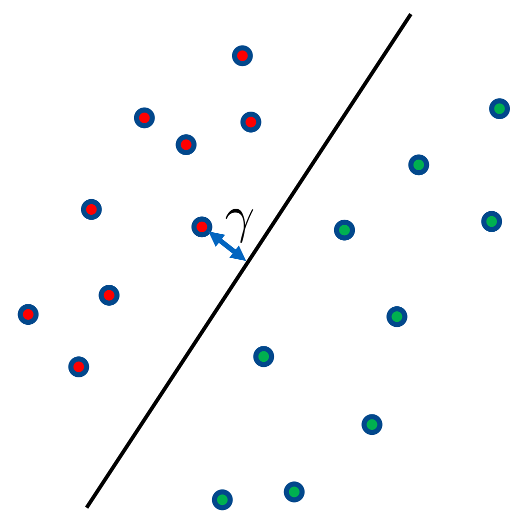
$$\forall t \leq n, \quad \mathrm{sign}(w_*^\top x_t + b_*) = y_t$$

- Define the **geometric margin** :

$$\gamma = \text{ distance of closest point to the separator}$$

- Data scale

$$r = \max_{t \leq n} \|x_t\|$$



Theorem: Perceptron makes at most $\frac{r^2+1}{\gamma^2}$ updates and the final (w_m, b_m) is correct on the data set

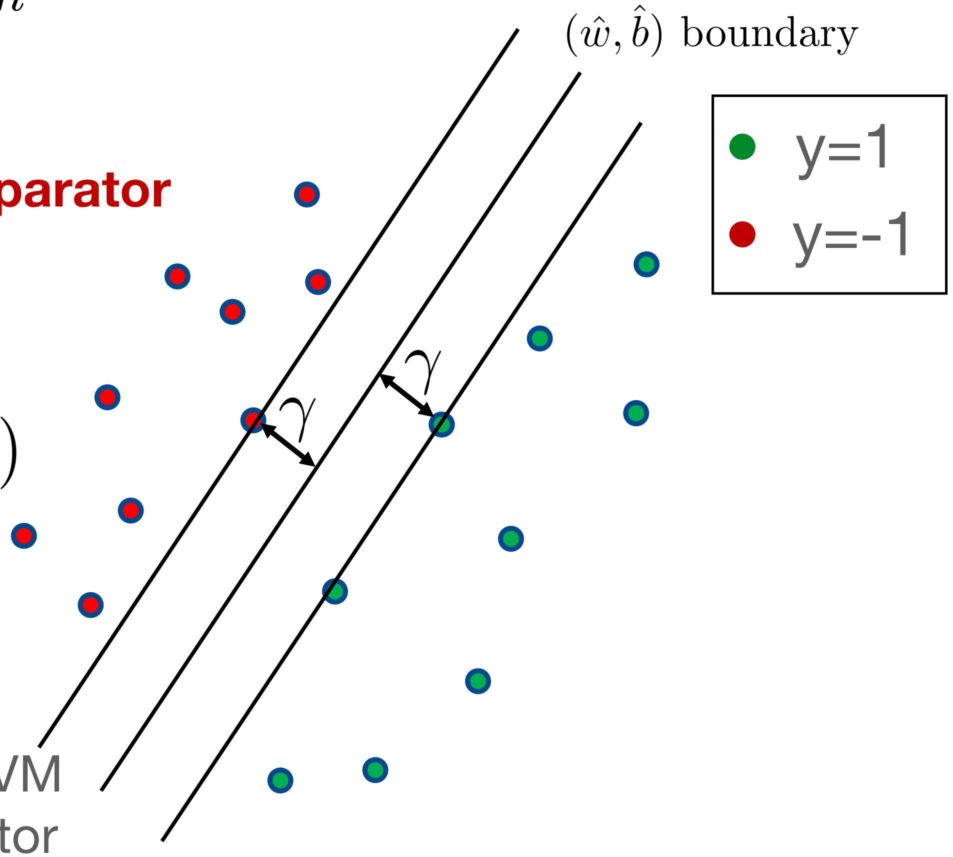
Support Vector Machine (SVM)

$$(\hat{w}, \hat{b}) = \underset{(w,b):\|w\|=1}{\arg\max} \ \min_{1 \le i \le n} y_i \left(w^\top x_i + b \right)$$

- SVM: pick the **maximum margin separator**

geometric margin:

$$\gamma = \max_{(w,b):\|w\|=1} \ \min_{1 \le i \le n} y_i \left(w^\top x_i + b \right)$$



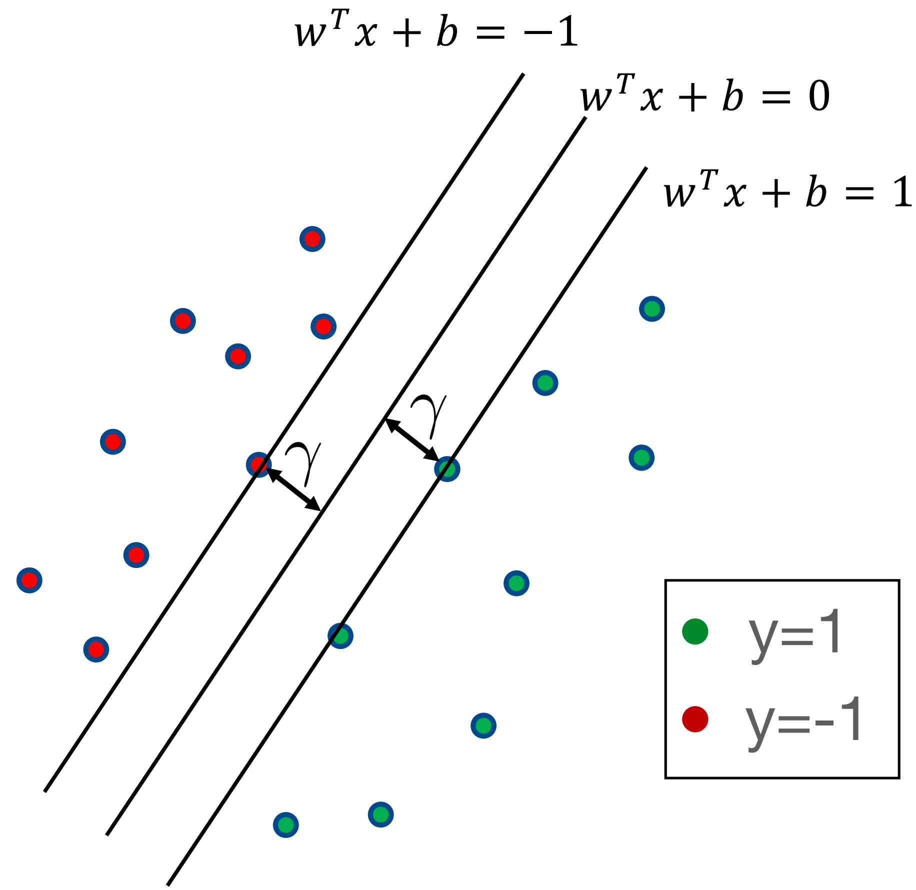
- Closest positive and negative point to SVM separator have same distance to separator

- There are no points inside a slab of width 2γ

- In higher dimensions, there can be many points that determine the solution (called “support vectors”)

Support Vector Machine (SVM)

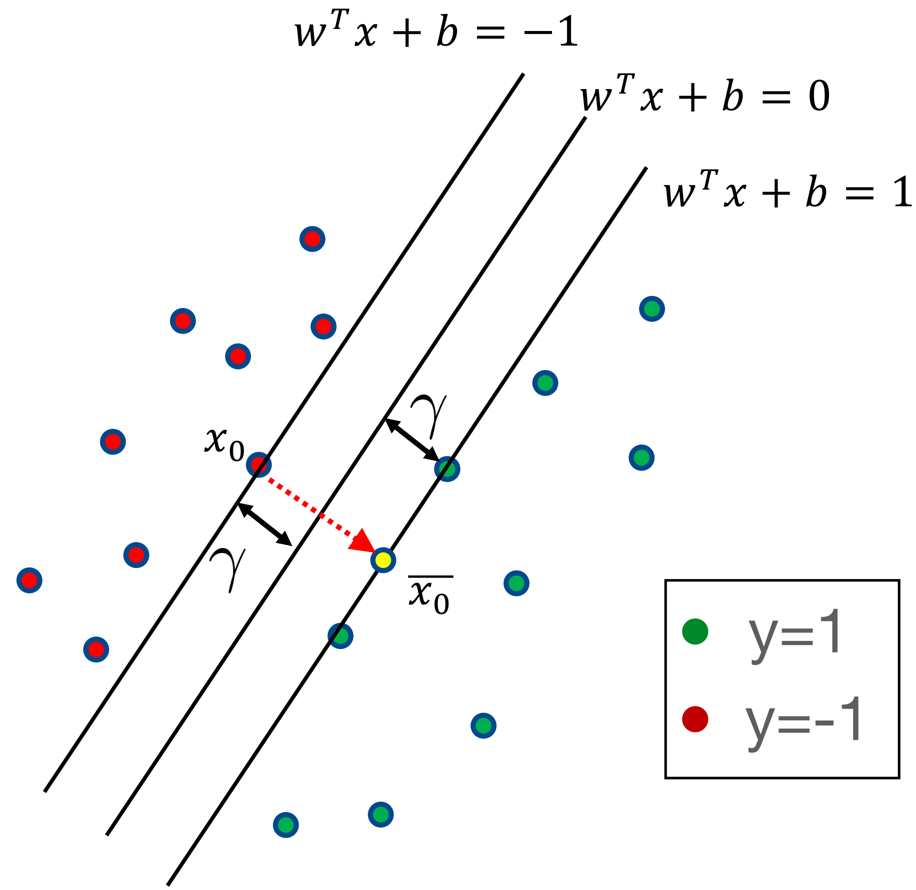
- Can write the three hyperplanes as:
  - $w^T x + b = -1$
  - $w^T x + b = 0$
  - $w^T x + b = 1$



Support Vector Machine (SVM)

- What is the distance between x_0 and $\overline{x_0}$?
- Here $\overline{x_0}$ is the projection of x_0 onto $w^T x + b = +1$
- Claim: $\overline{x_0} = x_0 + 2\gamma \frac{w}{||w||_2}$.

  [Why?]



Finding the SVM Solution

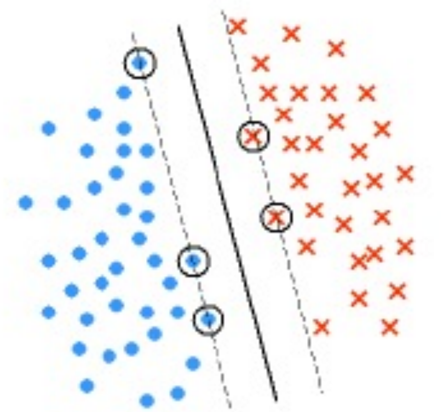
$$(\hat{w}, \hat{b}) = \underset{(w,b):\|w\|=1}{\operatorname{argmax}} \; \min_{1 \le i \le n} y_i \left(w^\top x_i + b\right)$$

- Can express SVM as a **quadratic program**:

$$\boxed{\begin{array}{ll} \text{Minimize} & \|w\|^2 \\ \text{subject to} & y_i \left(w^\top x_i + b\right) \ge 1, \; \forall i : \; 1 \le i \le n \end{array}}$$

- Why is this important? There are standard software packages to solve optimization problems expressed in this form.

- The name "*Support Vector Machine*" stems from the fact that supported by (i.e., is the linear span of) the examples that are at a distance $1 / \|w^*\|$ from the separating hyperplane. These
  are therefore called support vectors.



Support Vector Machines

- The name "*Support Vector Machine*" stems from the fact that w^* is supported by (i.e., is the linear span of) the examples that are exactly at a distance $1 / ||w^*||$ from the separating hyperplane.

- The vectors x_i that w^* is expressed as a linear combination of are therefore called support vectors.