- Makes online updates to for every misclassified point.
- Simple update rule:
- Comes with convergence guarantees given linear separability, which is a reasonable assumption in many real world scenarios include text classification!

The perceptron learns an explicit hyperplane through trial and error

9

Definition: $\text{sign}(z) = 1$ if $z \geq 0$, -1 otherwise.

Inputs: number of iterations, T; training examples $(\underline{x}_t, y_t)$ for $t \in \{1 \ldots n\}$ where $\underline{x} \in \mathbb{R}^N$ is an input, and $y_t \in \{-1, +1\}$ is a label.

Initialization: $\underline{\theta} = \underline{0}$ (i.e., all parameters are set to 0)

Algorithm:

- For $j = 1 \ldots T$
  - For $t = 1 \ldots n$
    1. $y' = \text{sign}(\underline{x}_t \cdot \underline{\theta})$
    2. If $y' \neq y_t$ Then $\underline{\theta} = \underline{\theta} + y_t \underline{x}_t$, Else leave $\underline{\theta}$ unchanged

Output: parameters $\underline{\theta}$

- **Taken from Collins,** *Convergence Proof for the Perceptron Algorithm* **(2012)**

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

W

Perceptron demo

The perceptron learns an explicit hyperplane through trial and error

- Makes online updates to $\mathbf{w}$ for every misclassified point.
- Simple update rule: $\mathbf{w} \leftarrow \mathbf{w} + y_i \mathbf{x}_i$
- Comes with convergence guarantees given linear separability, which is a reasonable assumption in many real world scenarios include text classification!

Definition: $\text{sign}(z) = 1$ if $z \geq 0$, -1 otherwise.

Inputs: number of iterations, T; training examples $(\underline{x}_t, y_t)$ for $t \in \{1 \ldots n\}$ where $\underline{x} \in \mathbb{R}^N$ is an input, and $y_t \in \{-1, +1\}$ is a label.

Initialization: $\underline{\theta} = \underline{0}$ (i.e., all parameters are set to 0)

Algorithm:

- For $j = 1 \ldots T$ Perceptron demo
  - For $t = 1 \ldots n$
    1. $y' = \text{sign}(\underline{x}_t \cdot \underline{\theta})$
    2. If $y' \neq y_t$ Then $\underline{\theta} = \underline{\theta} + y_t \underline{x}_t$, Else leave $\underline{\theta}$ unchanged

Output: parameters $\underline{\theta}$

- **Taken from Collins,** *Convergence Proof for the Perceptron Algorithm* **(2012)**

Georgetown University

Linear Support Vector Machine (SVM)