Lecture 2 – 1/23/17

Classification

$$D = \{ (\vec{x}^{(1)}, y^{(1)}), (\vec{x}^{(2)}, y^{(2)}), \dots, (\vec{x}^{(N)}, y^{(N)}) \}$$

$\forall i \quad \vec{x}^{(i)} \in \mathbb{R}^M$ ← features (input)

$\forall i \quad y^{(i)} \in \{1, 2, \dots, L\}$ ← labels (output)

Def: Binary Classification

Above where $y^{(i)} \in \{\underset{\text{circles}}{0}, \underset{\text{crosses}}{1}\}$

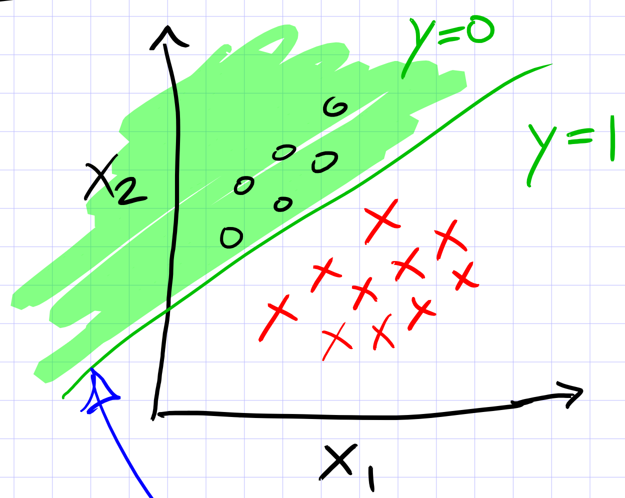
$y^{(i)} \in \{+1, -1\}$

$M = \#$ features

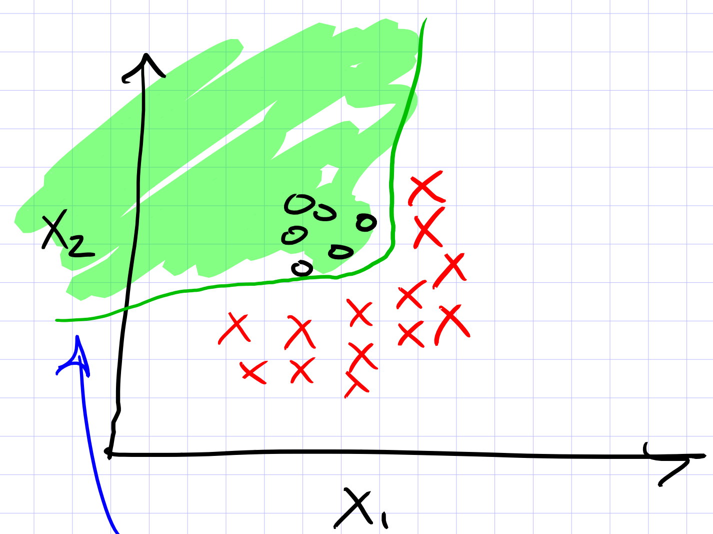
$N = \#$ examples $= |D|$

$$\vec{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(N)} \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} — (x^{(1)})^T — \\ — (x^{(2)})^T — \\ \vdots \\ — (x^{(N)})^T — \end{bmatrix} \qquad \vec{x}^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \\ \vdots \\ x_M^{(i)} \end{bmatrix}$$

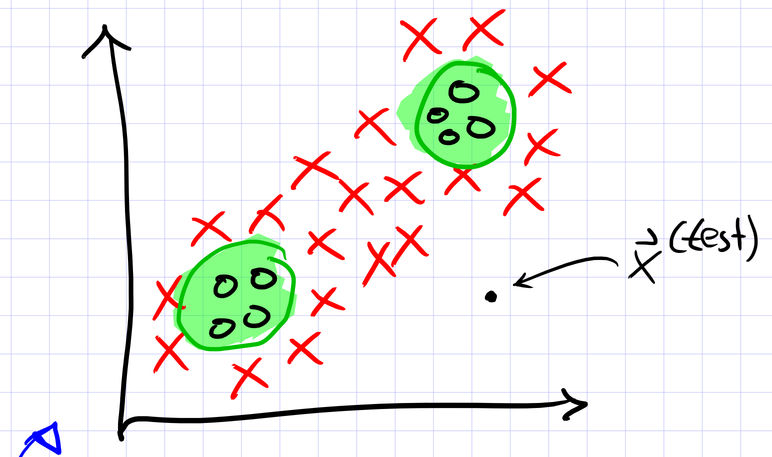
Ex: 2D Bin. Class



Linear Decision Boundary



Nonlinear Dec. Bound.



Nonlinear Decision Rule

<u>Def</u>: Decision Rule for Bin. Class.

Function $h : \mathbb{R}^M \rightarrow \{0,1\}$
(input) (output)

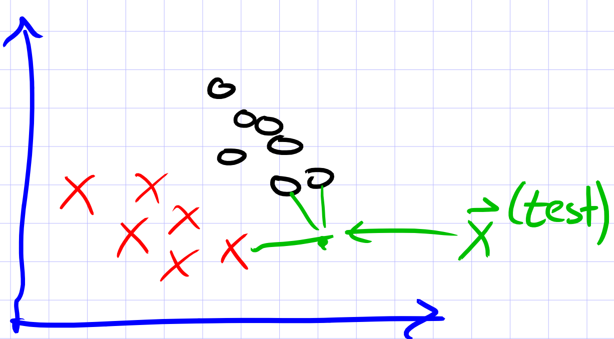
Given $\vec{x}^{(test)}$ we predict $\hat{y} = h(\vec{x}^{(test)})$

$\boxed{\text{k-Nearest Neighbors}}$ ← for Classification

<u>Def</u>: (English) "Assign most common label among k neighbors in D that are nearest to $\vec{x}^{(test)}$"

<u>Def</u>: (Picture)

k=3



Def: Algorithm

Inputs

k = # neighbors
d = distance function
D = data

① Train: – Given training ex.s D
– Store D
– Return.

② Predict: – Given test ex $\vec{x}^{(test)}$
– Find k nearest neighbors

k = # neighbors

$$I = \{n_1, n_2, \ldots, n_k\}$$

training examples

$$\forall_j,\ n_j \in I,\ \ n_j \in \{1, 2, \ldots, N\}$$

$$\text{s.t. } d(\vec{x}^{(test)}, \vec{x}^{(n_j)}) \le d(\vec{x}^{(test)}, x^{(i)})$$

$$n_j \in I \qquad i \notin I$$

– Return majority label

$$\hat{y} = h(\vec{x}^{(test)}) = \underset{\ell \in \{0,1\}}{\text{argmax}} \sum_{n_j \in I} \mathbb{1}(\ell, y^{(n_j)})$$

indicator fn.

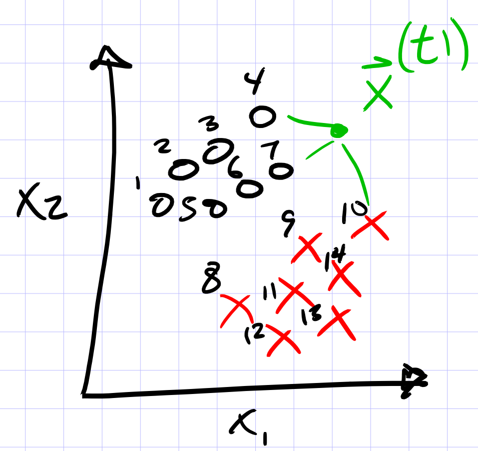
$$\mathbb{1}(u,v) = \begin{cases} 1 & \text{if } u = v \\ 0 & \text{otherwise} \end{cases}$$

Remarks: Distance fn. d is ~~usually~~ Euclidean

$$d(\vec{u}, \vec{v}) = \sqrt{\sum_{m=1}^{M} (u_m - v_m)^2}$$

Other fns are fine

$$d(\vec{u}, \vec{v}) = \sum_{m=1}^{M} |u_m - v_m|$$



$$y^{(t1)} = h(\vec{x}^{(t1)}) = 0$$

$$I = \{4, 7, 10\}$$