DMML, 19 Feb 2019

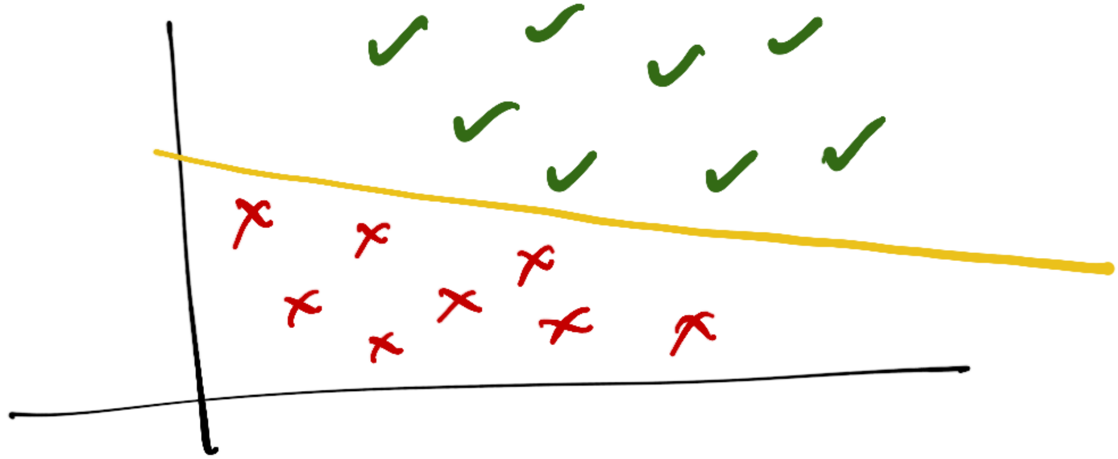
Back to supervised learning

Geometric approach

Assume (like clustering) data are points $(x_1, x_2, \ldots x_n)$ in space

Suppose category depends on position

Simplest case – linearly separable



Find separator

What if not separable? Later.

Hyperplane $w_1 x_1 + w_2 x_2 \ldots + w_n x_n + w_{n+1} = 0$

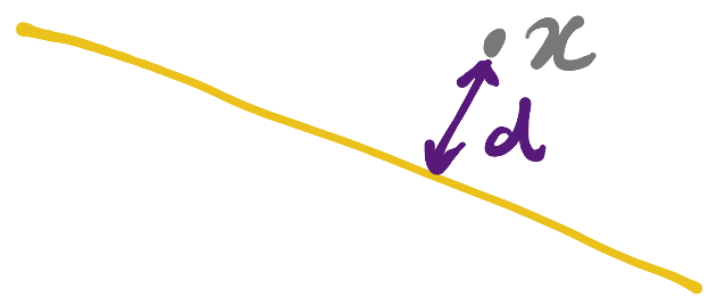
$\cdot x_{n+1}$

$x_{n+1} = 1$

$\langle w_1, w_2, \ldots, w_n \rangle \cdot \langle x_1, \ldots, x_n \rangle = 0$

$\bar{x}^T \cdot \bar{w} = 0$

Good separator — find good $\bar{w}$ say $\bar{w}^*$

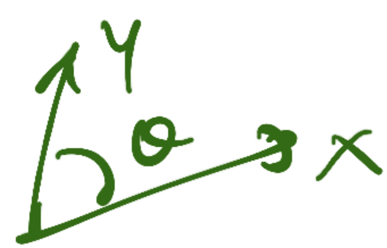


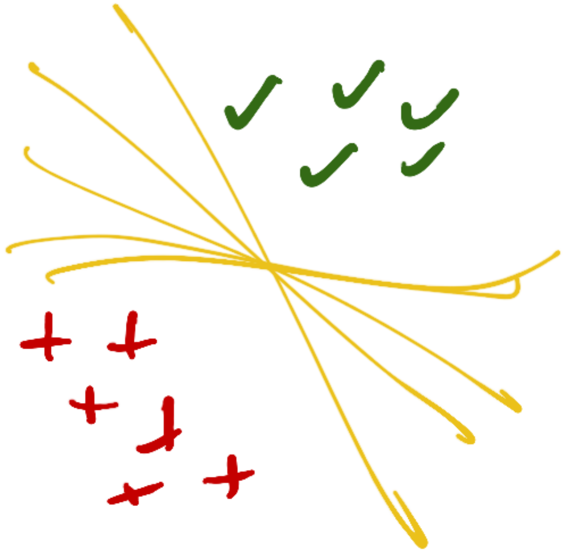
$$x^T w^* = 0$$

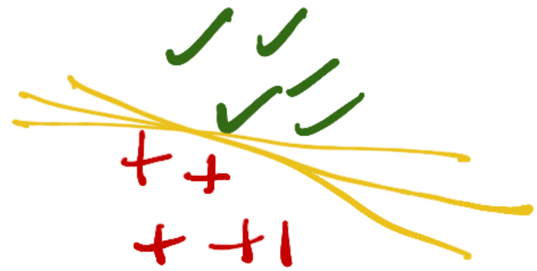
Arbitrary point x

$$d = \frac{x^T w^*}{\|w^*\|}$$

Recall: $x \cdot y = \|x\| \|y\| \cos\theta$







Bigger separation $\Rightarrow$ Easier to find a w^*

The other quantity of interest $= \|x\|$

Aim for w^* s.t. $x^T w^* \geq 1$ for all positive points

$x^T w^* \leq -1$ for all neg pts

Min distance is $\frac{1}{\|w\|} = \gamma$

Perceptron Algorithm

- Initialize w to 0
- Pick x in training set s.t $x^T w$ has wrong sign
  - If x is a positive example, $w \leftarrow w + x$
  - If x is a negative example, $w \leftarrow w - x$

Thm If there is w^* s.t. $x^T w^* \geq 1$ (≤ -1) for all positive (negative) examples, then the Perceptron algorithm makes at most $R^2 |w^*|^2$ updates

$(R/\gamma)^2$

R is $\max_{x \in X} \|x\|$

X ← training set

Proof

Ideal w^*

Current w

- Each update increases $w^T w^*$ by at least 1

+ve example

$$(w+x)^T w^* = w^T w^* + \underbrace{x^T w^*}_{\geq 1}$$

-ve example

$$(w-x)^T w^* = w^T w^* - \underbrace{x^T w^*}_{\leq -1}$$

- With each update, $|w|^2$ increase by at most R^2

$\underline{+ve}$ $$(w+x)^T(w+x) = |w|^2 + \underbrace{2x^Tw}_{<0} + |x|^2$$

$$\le |w|^2 + \underbrace{|x|^2}_{\le |R|^2}$$

$$\le |w|^2 + |R|^2$$

Suppose we make M updates

- $w^T w^* \geq M$ (M updates, add at least 1 each time)

- $|w|^2 \leq M |R|^2$ (M updates, add at most $|R|^2$ each time)

$$|w| \leq \sqrt{M}\,|R|$$

Observe $\dfrac{w^T w^*}{|w^*|} \leq |w|$ Projection of w on w^* is at most w

$$|w| \leq R\sqrt{M} \qquad w^T w^* \geq M$$

$$\frac{M}{w^*} \leq R\sqrt{M}$$

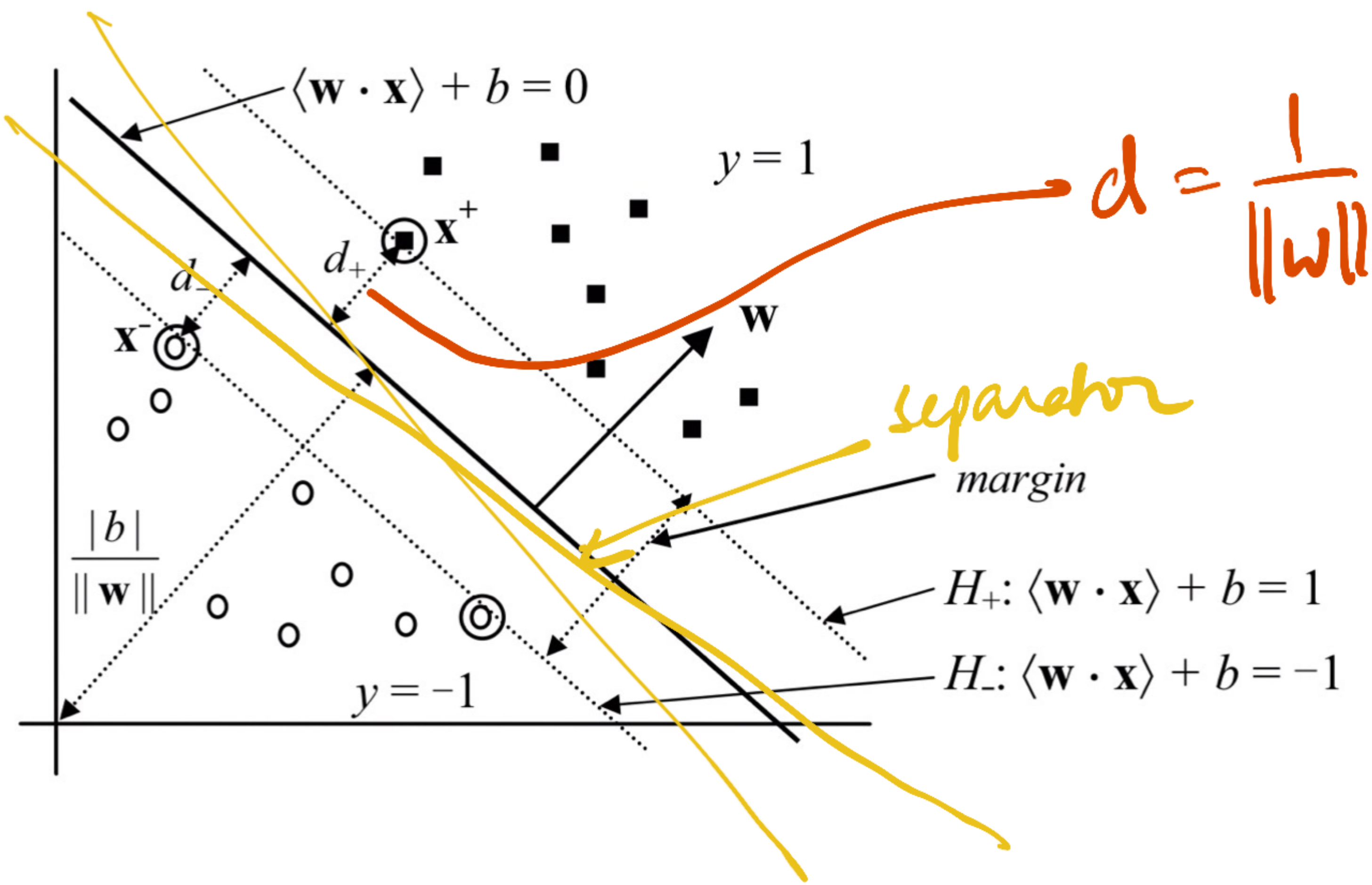
$$\sqrt{M} \leq R w^*$$

$$M \leq R^2 |w^*|^2$$

Not clear that the w we get after M updates is w^*

Can we find w^* ?

Instead of iterative update, formulate a global optimization problem.



Total width of the "margin" is $\frac{2}{\|w\|}$

Find w that maximizes margin & separates the points as $+1/-1$

For convenience.

Minimize $\frac{\|w\|}{2}$ or $\frac{\langle w \cdot w \rangle}{2}$

Optimization problem

Minimize $\frac{\langle w \cdot w \rangle}{2}$

Subject to

$\langle w \cdot x_i \rangle + b \geq 1$ for positive x_i

$\langle w \cdot x_i \rangle + b \leq -1$ for negative x_i

Associate $y_i = +1/-1$ with positive/negative x_i

$y_i(\langle w \cdot x_i \rangle + b) \geq 1$ for all x_i

Minimize $\frac{\langle w \cdot w \rangle}{2}$ ← Quadratic in w

Subject to

$$y_i \left(\langle w \cdot x_i \rangle + b \right) \geq 1 \text{ for all } i$$

Linear in w

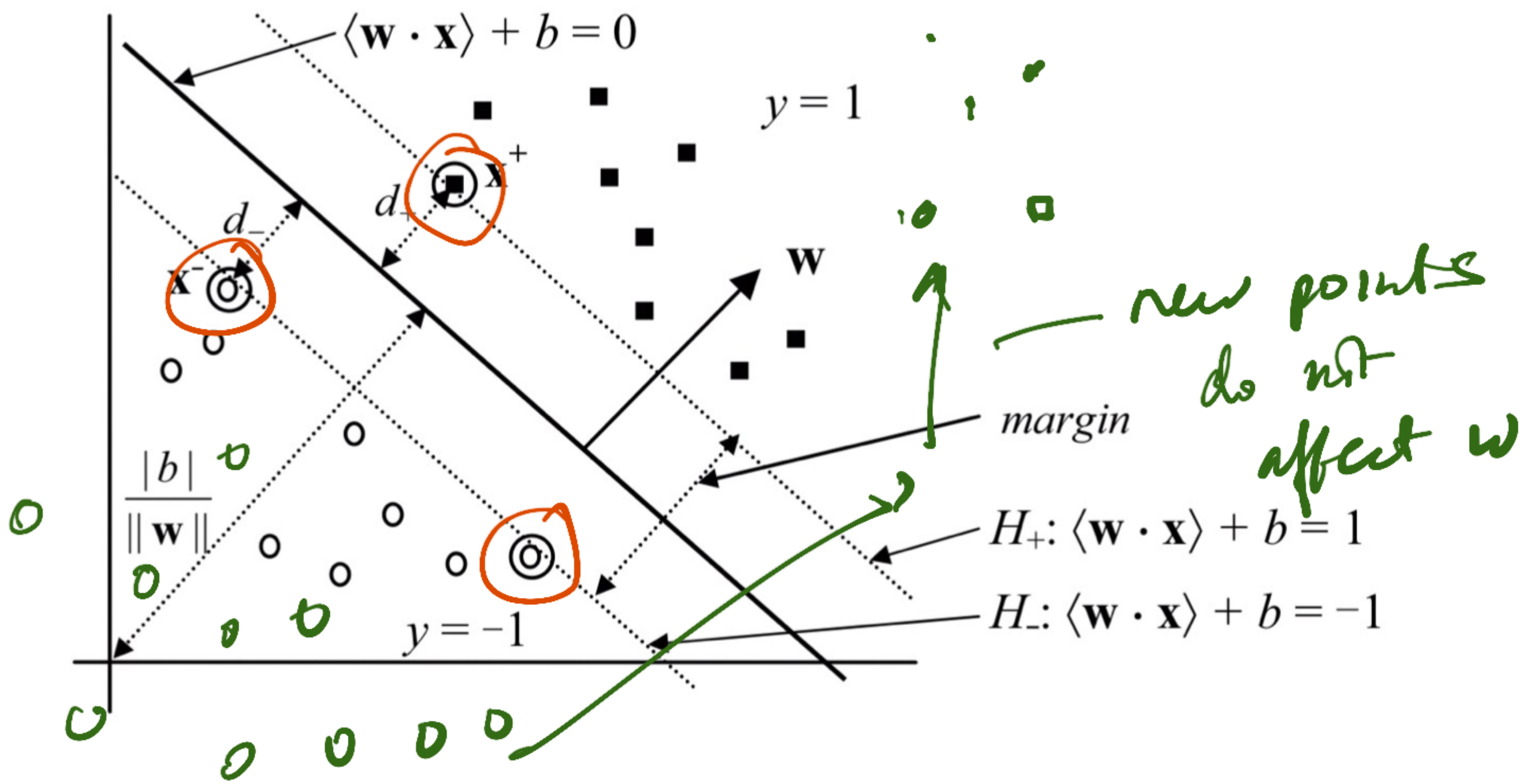
Unknown : w

Quadratic constrained optimization problem

Under certain (syntactic) conditions [KKT]
this can be solved

Solves same problem as Perceptron Algo,
but provides max_margin separator

Support Vector Machine (SVM)



Solution will be in terms of the x_i's that lie on the margin = support vectors

Optimization problem

→ Dual

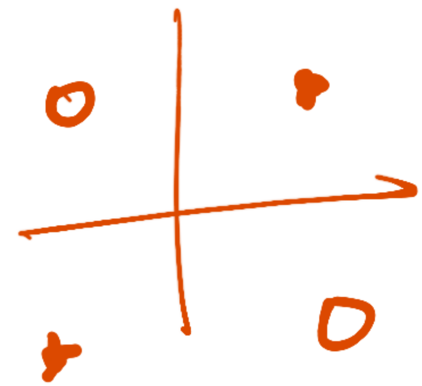
Solve dual problem – the solution can be expressed entirely in terms of dot products $x_i \cdot x_j$

If we can compute $x_i \cdot x_j$, we can find w efficiently

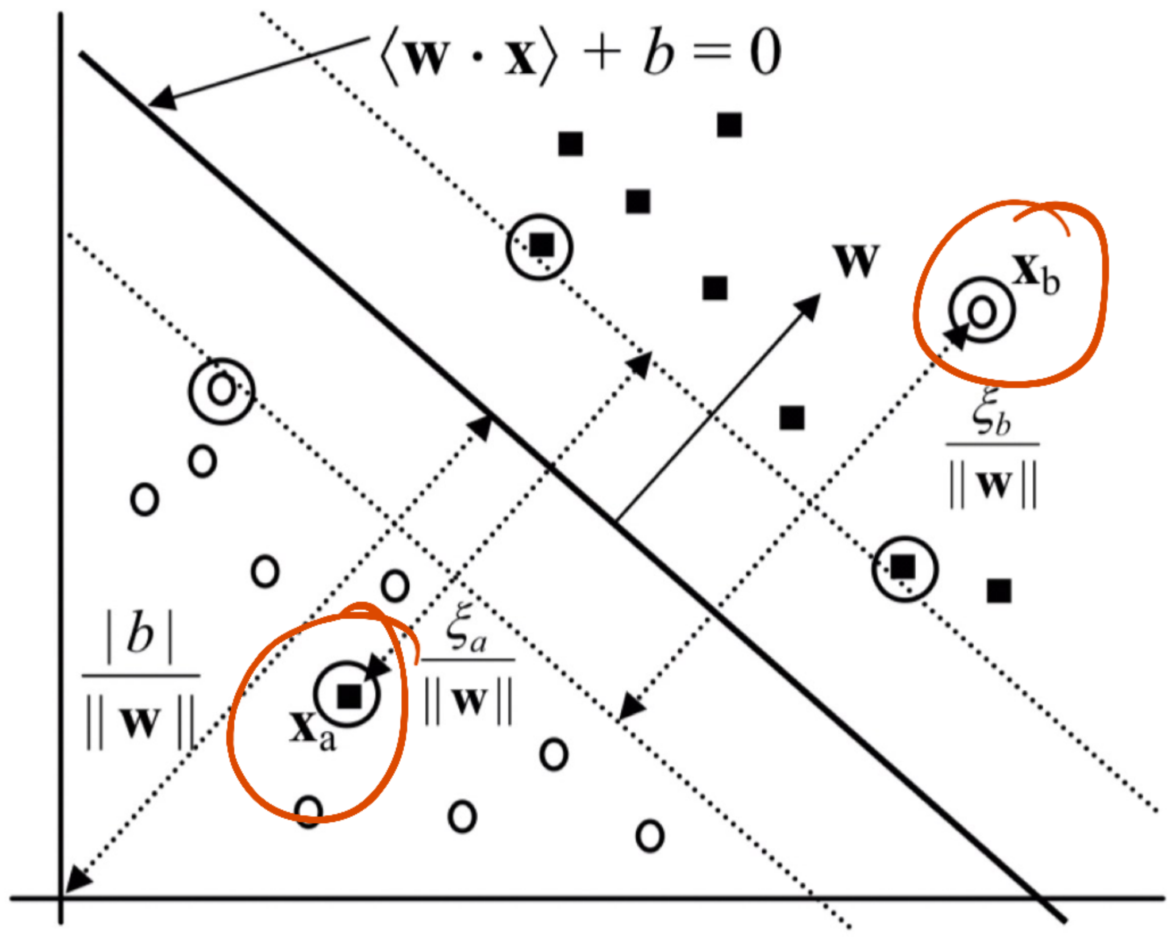
Will play a role later

Not linearly separable

Extreme case →



Simpler case – "small errors"



Fudge factor

$\langle W \cdot X \rangle + b \geq 1 - \xi_i$ for +ve x.

$\langle W \cdot x \rangle + b \leq 1 + \xi_i$ for -ve x

Find W & ξ_i's

Penalize big ξ_i – errors

Minimize $\frac{\langle w \cdot w \rangle}{2} + f(\xi_i)$

$\Rightarrow \sum_i \xi_i^2$

Subject to

$$y_i \left(\langle w \cdot x_i \rangle + b \right) \geq 1 - \xi_i \ , \forall i$$

Same type of quadratic constrained optimization

"Soft margin SVM"

Next time

Dealing with not linearly separable data in a general sense

Solution - geometrically transform data to be linearly separable

Move from (x, y) to (r, θ)