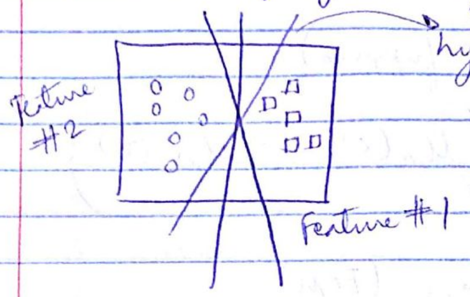
Jan 24th 2016 Classification dilema

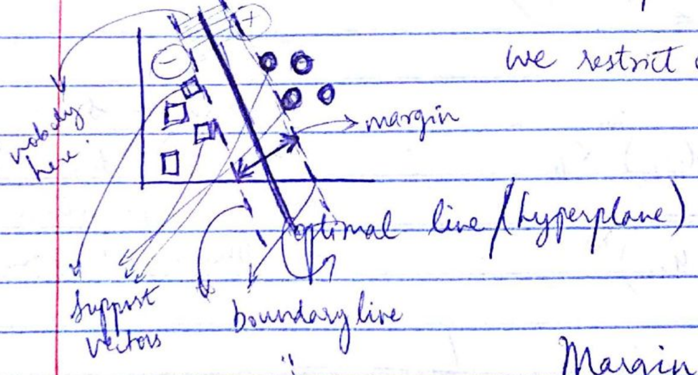
How to uniquely/reliably separate classes?



There are many solutions!

Which one of these hyperplanes (lines) optimally separate classes?

Lets assume we have an optimale line (hyperplane)

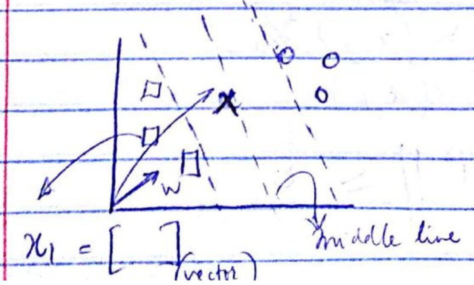


We restrict ourselves to two-class problems.

- nobody on the highway
- widest highway you can build.

optimal = widest margin between classes.

Margin maximization



To make a decision:

$w \cdot x \geq$ constant

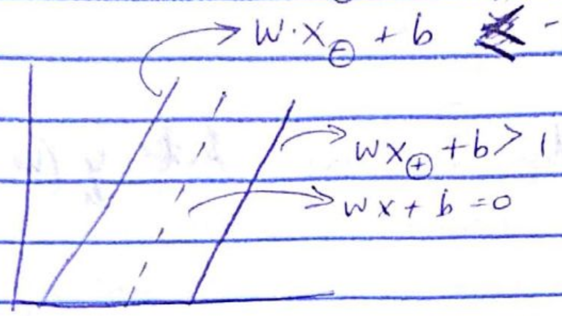
or $\boxed{w \cdot x + b \geq 0}$

this line !!

$\bar{X}$, $\vec{X}$, $\mathbf{X}$ } all vectors here today!

Let's focus on two class problems. e.g. Class 1 : $\oplus \triangleq +1$
Class 2 : $\ominus \triangleq -1$

$w \cdot x_{\oplus} + b \geq 1$ for positive samples
$w \cdot x_{\ominus} + b \leq -1$ for negative samples.

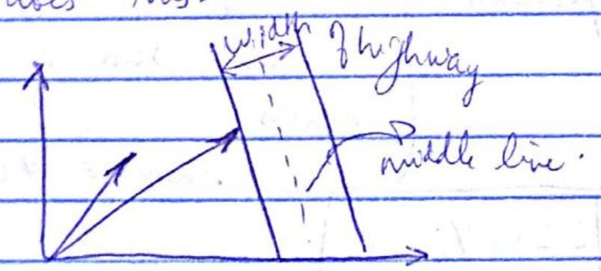


one line : too much variability.
if we have a strip with a margin : more constrained.
optimize this. SVM.

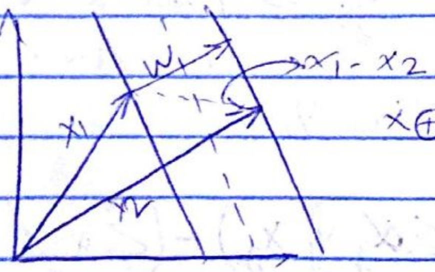
Let us define the output
y_i such that $y_i = +1$ for $x_{\oplus}$
$y_i = -1$ for $x_{\ominus}$

Multiply both sides with y_i
$y_i (w \cdot x_{\oplus} + b) \geq 1$
$y_i (w \cdot x_{\ominus} + b) \geq 1$
doesnt matter if the class is +1 or -1 class.

$y_i (w \cdot x_i + b) - 1 \geq 0$
$y_i (w \cdot x_i + b) - 1 = 0$ middle line
doesnt matter if it is class -1 or class +1.
this is a line. middle line. Find a hyperplane which does this.



We have to know the width of highway to calculate where the vector ends. this side of highway or other side
width of vector from



$x_{\oplus} - x_{\ominus}$

$$(x_{\oplus} - x_{\ominus}) \cdot \frac{w}{\|w\|}$$

$y_i = 1$
$\therefore y_i (w \cdot x_{\oplus} + b \geq 1)$ will be $\boxed{(1-b) + (1+b)}$

$\therefore \frac{2}{\|w\|}$

Scanned by CamScanner

$\frac{2}{\|w\|}$ has to be maximized (broad highway)
"maximize this" i.e., minimize $\|w\|$

Find the ~~stat~~ ~~stw~~ >

Subject to ~~the ellin~~ the condition: s.t. $y_i(w \cdot x_i + b) - 1 = 0$

Lagrange Multiplier.

minimize $f(x)$ s.t. $g(x) = 0$ → constraint

$L = f(x) - \lambda g(x)$ — we build it as a multiplicat of constraint.

→ multipliers

$$L = \frac{1}{2}\|w\|^2 - \sum \alpha_i \left[y_i (w \cdot x_i + b) - 1 \right]$$

Only on the middle lines $\alpha_i \neq 0$. Anywhere else, it can be 0

Build partial derivatives:

$$\frac{\partial L}{\partial w} = w - \sum \alpha_i y_i x_i = 0 \Rightarrow \boxed{w = \sum \alpha_i y_i x_i}$$

width of highway is dependant on nature of data.

[width of highway/margin is a linear sum of (some of) inputs (when $\alpha \neq 0$)]

$$\frac{\partial L}{\partial w} = -\sum \alpha_i y_i \Rightarrow \boxed{\sum \alpha_i y_i = 0}$$

(set $w = 0$ to calculate min/max)

Replace w in L with $\sum \alpha_i y_i x_i$:

$$L = \frac{1}{2} \underbrace{\left(\sum \alpha_i y_i x_i\right)\left(\sum \alpha_j y_j x_j\right)}_{w^2 \text{ (dot product / magnitude)}} - \left(\sum \alpha_i y_i x_i\right) \cdot \left(\sum \alpha_j x_j y_j\right) - \underbrace{\left(\sum \alpha_i y_i\right)}_{=0} b + \sum \alpha_i$$

$$\boxed{L_1 = \sum \alpha_i - \frac{1}{2} \sum\sum \alpha_i \alpha_j Y_i Y_j X_i \cdot X_j}$$

(dot product)

minimize ~~maximize~~ this

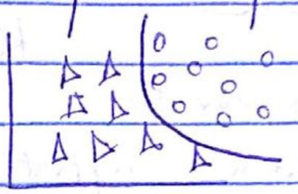
everything depends on dot product of input vectors. we just need alpha

[Give this to "Quadratic Optimization"]

works if:

- if you have 2 classes.
- if you have a linearly separable problem.

if we have:



Cannot be done by SVM.

Decision rule:

$$\sum \alpha_i Y_i X_i \cdot \underline{u} + b \geq 0 \quad \text{then} \quad \oplus$$

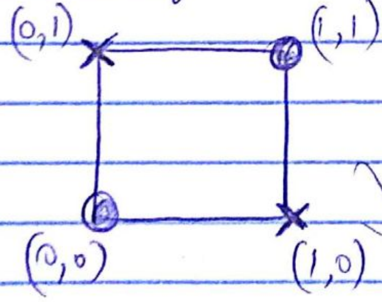
unseen (new) data

support vector

every thing that has non zero Lagrangian multiplier is a support vector.

This only works for linearly separable cases.

What if we have XOR like problems?

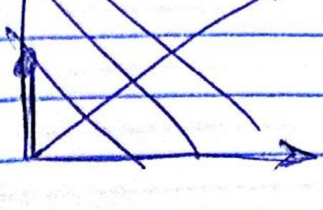


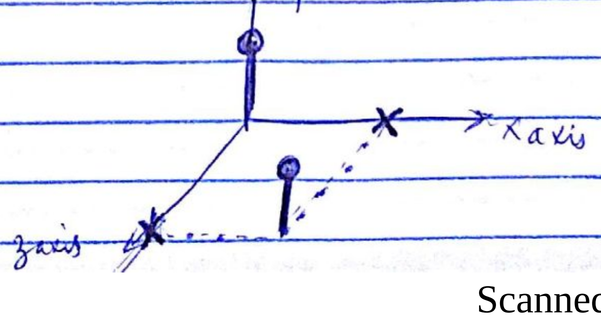
11 → 0
10 → 1
01 → 1
00 → 0

draw line to separate ⊙ from ×

No way to draw a line to separate this using a line

∴ extend the above solution to solve non linearly separable problems. Eg for this example, extend to 3D





If $\phi(x)$ is a transformation that maps x to higher dimensions to make it linearly separable.

$\phi(x) \cdot \phi(u)$ $\Rightarrow$ we have to find some kernels to do that.

support vector   new data

Popular kernels: $K(u,v) = (u \cdot v + 1)^n$

$K(u,v) = e^{-\frac{\|u-v\|}{\delta}}$

RBF

let you measure the similarity between support vectors and new data.