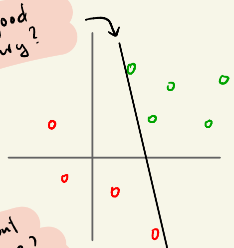
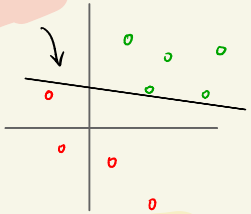
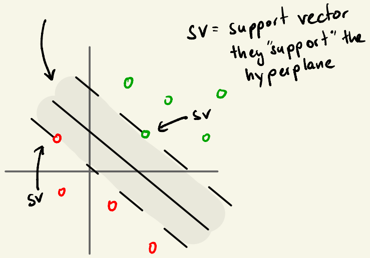
This a good Boundary?



How about this one?



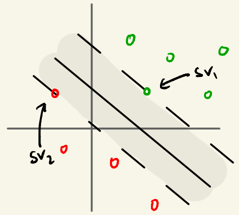
Wait: fit a street between the classes that is as wide as possible.



Question: if we move some points a tiny bit, for which does the Decision Boundary change? & which not?

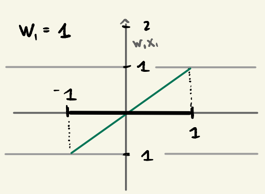
How?

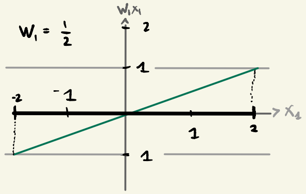
Choose w such that $w^T x - b = 1$ goes through sv_1, and $w^T x - b = -1$ goes through sv_2.

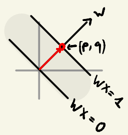


Maximum Margin: maximize the distance from sv_1, sv_2 to the Hyperplane $\Leftrightarrow$ minimize $\|w\|$ under conditions $y_i(w^T x_i - b) \geq 1$

Intuitive picture: smaller $w \Rightarrow$ wider margin

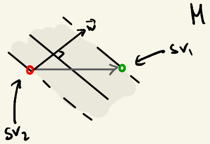




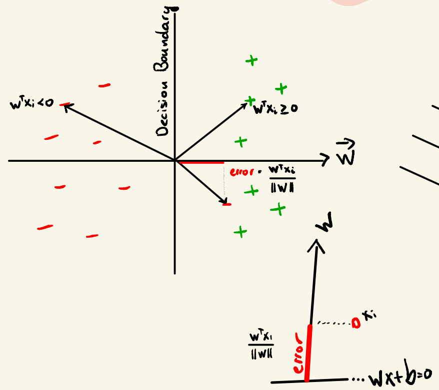


Let $sv_1 = (p,q)$
then $sv_1 = t \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$
because $wx = 1$
$\Rightarrow t \cdot w^T w = 1 \Rightarrow t = \frac{1}{\|w\|^2}$

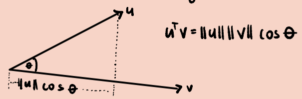
Simple case: hyperplane through origin. Below more general.

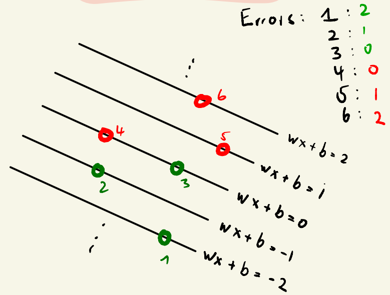


$$\text{Margin} = (sv_1 - sv_2)^T \frac{w}{\|w\|} = sv_1^T \frac{w}{\|w\|} - sv_2^T \frac{w}{\|w\|} = \frac{b+1}{\|w\|} - \frac{b-1}{\|w\|} = \frac{2}{\|w\|}$$

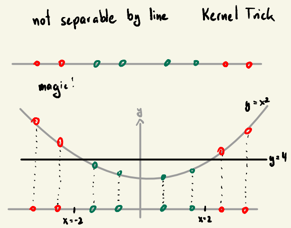


Reminder: Dot Products & Projections



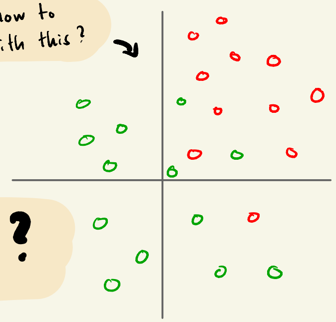


TLDR:



Problem! ← impossible if classes are not linearly separable

So..., how to deal with this?

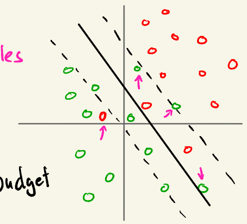


IDEA!

How?

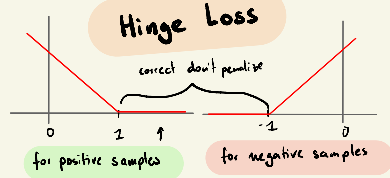
Cut classifier some slack and give it a budget for margin violations and misclassifications

Slack variables ε_i



$\sum_{i=1} \varepsilon_i \leq \text{budget}$

Hinge Loss



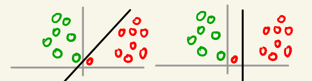
Soft Margin:

minimize: $\frac{1}{n} C \cdot \sum (\max(0, 1 - y_i(w^T x_i - b)) + \|w\|^2$

Look familiar? →

Before (i.e. Ridge)
minimize $A + \lambda B$ bigger λ more weight on B
now
minimize $CA + B$ smaller C more weight on B
$\Rightarrow C \approx \frac{1}{\lambda}$

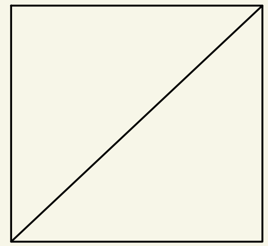
Bonus! also less sensitive to outliers!



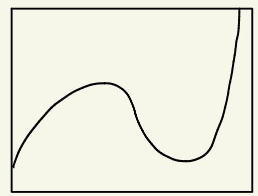
$\Rightarrow$ big C → small budget small C ⇒ bigger budget ⇒ wider margin

Shapes of Decision Boundaries from different Kernels

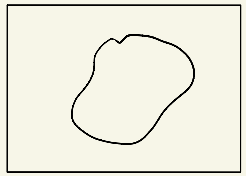
Rules of Thumb!



Straight lines ⇒ linear



Curves = polynomial



islands = RBF
radial basis f.