1. Maximum Margin Classifiers.

Our tasks & assumptions

① Our raw data x will be mapped to another space by kernel function $\phi(x)$.

② We know before hand that there will be two classes.

③ We will use a hyperplane to separate them

④ Our dataset is linearly separable

We have a vector of true labels

$$\{t_i\} \in \{-1, 1\}$$

We have a set of input data

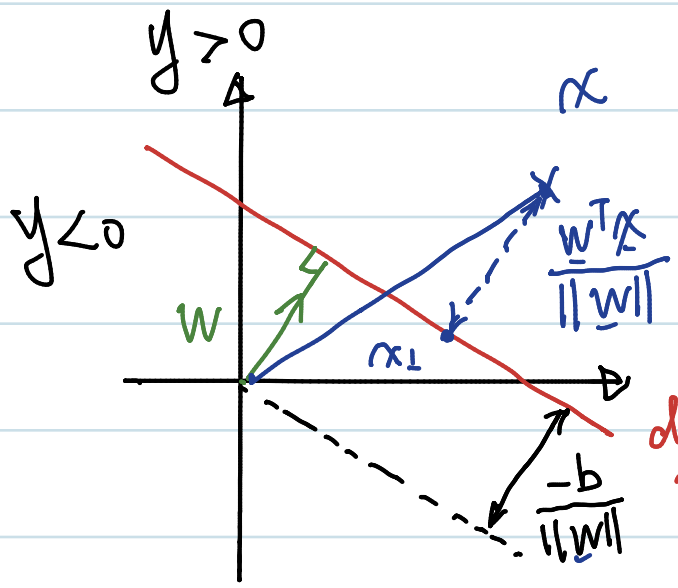
$$\{x_i\}$$

Our linear discriminant model is

$$y(x) = w^T \phi(x) + b$$

and w, b will represent the decision hyperplane that will linearly separate the two classes.

Let's remind ourselves about some geometries.



If x is on decision plane,

$$y = w^T x + b = 0$$

$$\Rightarrow \frac{w^T}{\|w\|} x = -\frac{b}{\|w\|}$$

For a point x, it's distance to the decision plane with kernel $\phi(\cdot)$ is

$$\left| \frac{y(x_n)}{\|w\|} \right|$$

now, w.l.o.g, let's assume that w's sign is carefully chosen, so $t_n y(x_n)$ is always non-negative. So the distance for point x to the decision plane is

$$d_n = \frac{t_n y(x_n)}{\|w\|} = \frac{t_n (w^T \phi(x_n) + b)}{\|w\|}$$

Now, margin is defined to be the distance of the closest point to decision plane, mathematically it is.

$$\min_n \left\{ \frac{1}{\|w\|} \cdot t_n (w^T \phi(x_n) + b) \right\}$$

Our final goal is to maximize this gap, or say a "maxmin" problem

$$\arg\max_{w,b} \left\{ \frac{1}{\|w\|} \min_n [t_n (w^T \phi(x_n) + b)] \right\}$$

However, solving this optimization problem directly is difficult. We want to convert it to one equivalent simple problem.

Because we don't care the magnitute of w and b. say mutiply them both by a non-zero veetor k, we won't change our final results. for the "min" problem.

Based on this freedom, we can choose a factor k s.t.

$$t^*(W^T\phi(x^*)+b) = 1$$

where x^* is the point that is closest to the decision plane.

So, we have

$$t_n(W^T\phi(x_n)+b_n) \geq 1 \quad \forall n \quad \text{---} \quad (*)$$

Actually, $(*)$ can be viewed as n inequality constraints. For point which makes the equality holds can be regardeed as an "active" constraint. There will be at least 2 active constraints.

The new optimization problem is

$$\min_{W,b} \quad \frac{\|W\|^2}{2}$$

$$\text{S.t} \quad t_n(W^T\phi(x_n)+b_n) \geq 1 \quad \forall n$$

So far, we simplify our original target. Now thing is to solve it.

We will use the lagrangian.

$$\mathcal{L}(w, b, \lambda)$$

$$= \frac{1}{2}\|w\|^2 - \sum_{n=1}^{N} \lambda_n \{ t_n (w^T \phi(x_n) + b) - 1 \} \qquad \lambda \geq 0$$

Let's compute dual function $g(\lambda)$

$$\nabla_w \mathcal{L}(w, b, \lambda) = w - \sum_{n=1}^{N} \lambda_n t_n \phi(x_n) = 0$$

$$\Rightarrow w^* = \sum_{n=1}^{N} \lambda_n t_n \phi(x_n)$$

$$\nabla_b \mathcal{L}(w, b, \lambda) = \sum_{n=1}^{N} \lambda_n t_n = 0$$

Replace w with w^*, we have

$$g(\lambda) = \frac{1}{2}\left(\sum_{n=1}^{N} \lambda_n t_n \phi(x_n)\right)^T \left(\sum_{m=1}^{N} \lambda_m t_m \phi(x_m)\right)$$

$$-\left(\sum_{n=1}^{N} \lambda_n t_n \phi(x_n)^T\right)\left(\sum_{m=1}^{N} \lambda_m t_m \phi(x_m)\right) + \sum_{n=1}^{N} \lambda_n$$

$$= \sum_{n=1}^{N} \lambda_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n \lambda_n t_m t_n \underbrace{\phi(x_m)^T \phi(x_n)}_{K(x_m, x_n)}$$

$$= \sum_{n=1}^{N} \lambda_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1} \lambda_n \lambda_m t_n t_m K(x_m, x_n)$$

and $\lambda_n \geq 0 \quad \forall n$

$$\sum \lambda_n t_n = 0$$

Then, we simply assume that "according to some optimization technique", the dual problem is solved. and we heme the optimal λ^*.

From the formula of w^* when we minimize $L(w, b, \lambda)$ w.r.t w & b. we have.

$$w^* = \sum_{n=1}^{N} \lambda_n t_n \phi(x_n)$$

For the optimal value for b, the step is not that difficult.

$$t_n (w^T \phi(x) + b) \geq 1 \qquad t_n \in \{-1, 1\}, \forall n$$

$$\Rightarrow \quad t_n b \geq 1 - t_n w^T \phi(x)$$

$$\Rightarrow \quad t_n b = \min_n (1 - t_n w^T \phi(x))$$

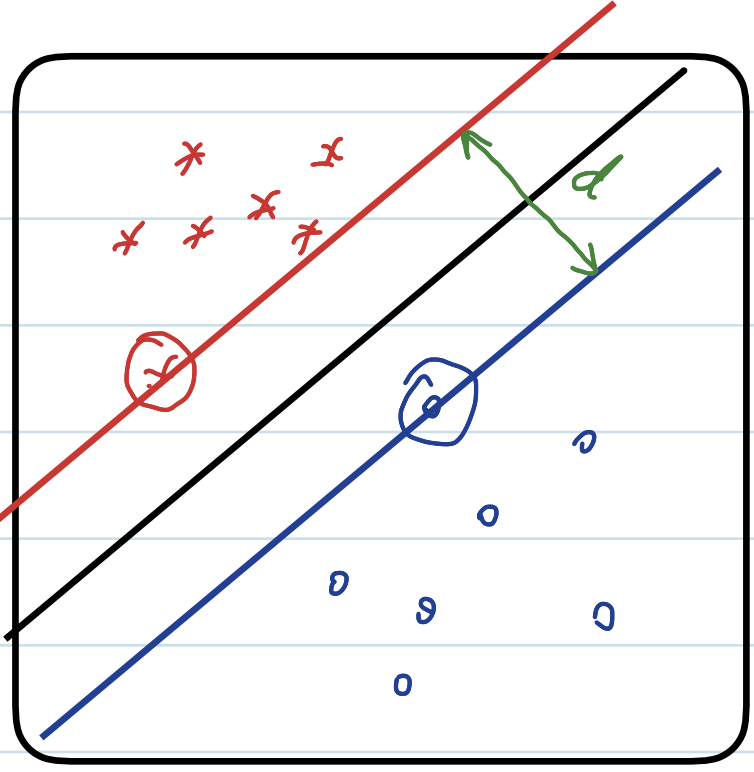
$$\Rightarrow \quad b^* = t_n \min (1 - t_n w^T \phi(x))$$

Later, we can use this formula & λ^*, we have

$$y(x) = \sum_{n=1}^{N} \lambda_n^* t_n k(x, x_n) + b^*$$

We can simply determine the class of sample x by checking the sign of $f(x)$.

Now, let's talk about the intuition for maximum margin classifier.



First, we only care the so called "Support Vector".
In left case. the circled points are regarded as support vectors, because they support the boundaries of red and blue samples.

The middle black line is regarded as the decision hyperplane. The direction of planes is chosen by maximizing the distance "d" between two boundary planes. So we know that, only the support vectors that really care. So, this algorithm won't be affected by weird outliers. E.g, points extremely far away from common class central won't affect the decision plane.