Linear Classification: Maximizing the Margin

Yufei Tao

Department of Computer Science and Engineering
Chinese University of Hong Kong

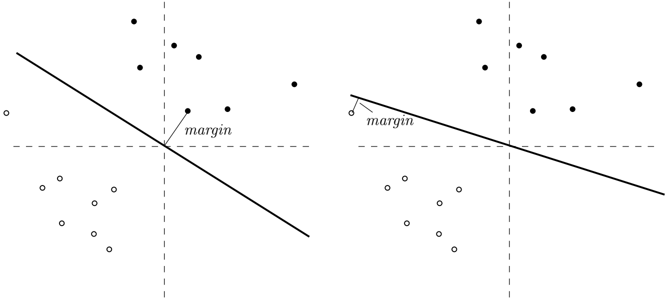
Let P be a **linearly separable** set of points in $\mathbb{R}^d$. That is, each point in P carries a label that is either 1 or -1; and we can find a d-dimensional vector $\vec{c}$ such that given a point $p \in P$:

- if p has label 1, then $\vec{c} \cdot \vec{p} > 0$.
- if p has label -1, then $\vec{c} \cdot \vec{p} < 0$.

The plane $\vec{c} \cdot \vec{x} = 0$ is referred to as a **separation plane**.

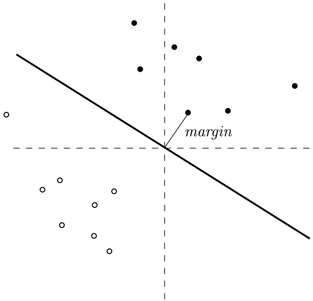
> There can be many such separation planes. As shown in the previous lecture, we should strive to find the plane that has the **largest margin**. In this lecture, we will discuss how to achieve the purpose algorithmically.

Review: Margins



We prefer the left plane.

Let P be a linearly separable set of points in $\mathbb{R}^d$. The goal of the **large margin separation problem** is to find a separation plane with the maximum margin.



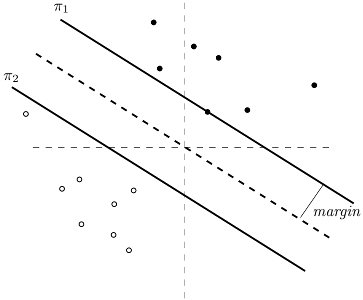
Any algorithm solving this problem is called a **support vector machine**.

Next, we will discuss two methods to approach the problem. The first one finds the optimal solution, but is computationally expensive. The second method is asymptotically as fast as Perceptron, but gives an approximate solution that is guaranteed to be not too far from optimality.

Finding the Optimal Plane

We will model the problem as a quadratic programing problem.

Consider an arbitrary separation plane $\vec{c} \cdot \vec{x} = 0$. Imagine two copies of the plane; one of them moves up, and the other moves down, both at the same speed. They stop as soon as one of the two planes hits a point in P.



Now, focus on the two copies of the plane in their final positions. If one copy has equation $\vec{c} \cdot x = \lambda$, then the other copy must have equation $\vec{c} \cdot x = -\lambda$, where λ is a strictly positive value.

For each point $p \in P$, we must have:

- if p has label 1, then $\vec{c} \cdot \vec{p} \geq \lambda$;
- if p has label -1, then $\vec{c} \cdot \vec{p} \leq -\lambda$.

By dividing λ on both sides of each inequality, we have:

- if p has label 1, then $\vec{w} \cdot \vec{p} \geq 1$;
- if p has label -1, then $\vec{w} \cdot \vec{p} \leq -1$

where

$$\vec{w} = \frac{\vec{c}}{\lambda}.$$

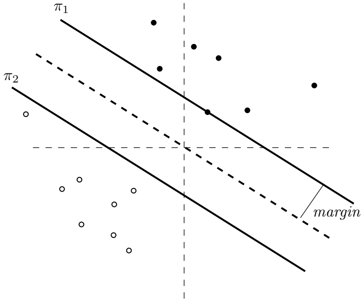
We will refer to the following plane as π_1:

$$\vec{w} \cdot \vec{x} = 1$$

the following plane as π_2:

$$\vec{w} \cdot \vec{x} = -1$$

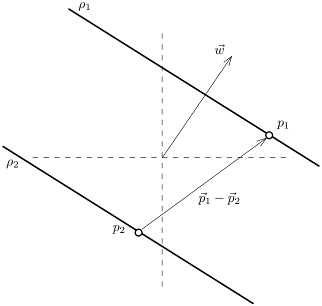
The margin of the original separation plane is exactly half of the distance between π_1 and π_2:



Lemma: The distance between π_1 and π_2 is $\frac{2}{|w|}$.

Note that it implies that the margin of the separation plane $\vec{w} \cdot \vec{x} = 0$ is $\frac{1}{|w|}$.

Proof: Take an arbitrary point p_1 on π_1, and an arbitrary point p_2 on π_2. Hence, $\vec{w} \cdot \vec{p}_1 = 1$ and $\vec{w} \cdot \vec{p}_2 = -1$. It follows that $\vec{w} \cdot (\vec{p}_1 - \vec{p}_2) = 2$.



The distance between the two planes is precisely $\frac{\vec{w}}{|\vec{w}|} \cdot (\vec{p}_1 - \vec{p}_2) = \frac{2}{|\vec{w}|}$. □

In summary of the above, to solve the large margin separation problem, we want to find $\vec{w} = (w_1, ..., w_d)$ to minimize $\sum_{i=1}^{d} w_i^2$, subject to the following constraints:

- For each label-1 point $p \in P$:

$$w_1 p[1] + ... + w_d p[d] \geq 1$$

- For each label-(-1) point $p \in P$:

$$w_1 p[1] + ... + w_d p[d] \leq -1$$

This is an instance of quadratic programming.

We will not discuss how to solve the instance, except to mention:

- In theory, it can be solved using convex-optimization techniques, which falls out of the scope of the course. The efficiency of those technique, however, is rather difficult to analyze.
- Numerous tools online can be readily deployed to solve the instance reasonably fast on many inputs.

Next, we will learn an alternative method that is as fast as Perceptron asymptotically, and returns an approximate solution whose margin is guaranteed to be close to the optimal achievable margin.

Finding an Approximate Plane

Define γ_{opt} as the maximum margin of all separation planes.
A separation plane is ***c*-approximate** if its margin is at least $c \cdot \gamma_{opt}$.

We will give an algorithm to find a $(1/4)$-approximate separation plane.

Remark: The constant $1/4$ here is chosen to simplify our presentation. Our algorithm can be extended to achieve better approximation ratios

Recall that a separation plane is given by $\vec{c} \cdot \vec{x} = 0$. The goal is to find a good $\vec{c}$.

Our weapon is once again Perceptron.
But we will correct $\vec{c}$ **not only** when a point falls on the wrong side of the plane, **but also** when the point is too close to the plane.

For now, let us assume we are given an arbitrary value $\gamma_{guess} \leq \gamma_{opt}$.
A point p causes a **violation** in any of the following situations:

- Its distance to the plane $\vec{c} \cdot \vec{x} = 0$ is less than $\gamma/2$, regardless of its label.
- p has label 1 but $\vec{c} \cdot \vec{p} < 0$.
- p has label -1 but $\vec{c} \cdot \vec{p} > 0$.

Margin Perceptron

The algorithm starts with $\vec{c} = \vec{0}$, and runs in iterations.

In each iteration, it tries to find a **violation point** $p \in P$. If found, the algorithm adjusts $\vec{c}$ as follows:

- If p has label 1, then $\vec{c} \leftarrow \vec{c} + \vec{p}$.
- If p has label -1, then $\vec{c} \leftarrow \vec{c} - \vec{p}$.

The algorithm finishes where no violation points are found.

Define $R = \max_{p \in P}\{|\vec{p}|\}$, i.e., the maximum distance from the origin to the points in P.

Theorem: If $\gamma_{guess} \le \gamma_{opt}$, margin Perceptron terminates in at most

$$12R^2/\gamma_{opt}^2$$

iterations, and returns a separation plane with margin at least $\gamma_{guess}/2$.

The proof can be found in the appendix.

Margin Perceptron requires a parameter $\gamma_{guess} \leq \gamma_{opt}$. The theorem of the previous slide tells us that a larger γ_{guess} promises a better quality guarantee.

Ideally, an ideal value for γ_{guess} is γ_{opt}, but unfortunately, we do not know the value of γ_{opt}.

Next, we present a strategy to estimate γ_{opt} up to a factor of $1/2$.

An Incremental Algorithm

1. $R \leftarrow$ the maximum distance from the origin to the points in P
2. $\gamma_{guess} \leftarrow R$
3. Run margin Perceptron with parameter γ_{guess}.
    - **[Self-Termination]**
      If the algorithm terminates with a plane π, return π as the final answer.
    - **[Forced-Termination]**
      If the algorithm has not terminated after $\frac{12R^2}{\gamma_{guess}^2}$ iterations:
        - Stop the algorithm manually.
        - Set $\gamma_{guess} \leftarrow \gamma_{guess}/2$.
        - Repeat Line 3.

Theorem: Our incremental algorithm returns a separation plane with margin at least $\gamma_{opt}/4$. Furthermore, it performs $O(R^2/\gamma_{opt}^2)$ iterations in total (including all the repeats at Line 3).

Proof: Suppose that we repeat Line 3 in total h times. For each $i \in [1, h]$, denote by γ_i the value of γ_{guess} at the i-th time we execute Line 3.

By the fact that the $(i-1)$-th repeat required a forced termination, we know that $\gamma_{h-1} > \gamma_{opt}$. Hence, $\gamma_h = \gamma_{h-1}/2 > \gamma_{opt}/2$. It thus follows that the plane we return must have a margin at least $\gamma_h/2 > \gamma_{opt}/4$.

The total number of iterations performed is

$$\begin{aligned} O\left(\sum_{i=1}^{h} \frac{R^2}{\gamma_i^2}\right) &= O\left(\frac{R^2}{\gamma_h^2} + \frac{R^2}{4\gamma_h^2} + \frac{R^2}{4^2\gamma_h^2} + \ldots\right) \\ &= O(R^2/\gamma_h^2) = O(R^2/\gamma_{opt}^2). \end{aligned}$$

□

Appendix: Proof of the Theorem on Slide 18.

(Will not be tested)

Let π^* be the the optimal plane with margin γ_{opt}.

Define $\vec{u}$ as the unit normal vector of π^* pointing to the positive side of π^*; in other words, we have:

- $|\vec{u}| = 1$.
- For every label-1 point $p \in P$, $\vec{p} \cdot \vec{u} > 0$.
- For every label-(-1) point $p \in P$, $\vec{p} \cdot \vec{u} < 0$.
- $\gamma_{opt} = \min_{p \in P}\{|\vec{p} \cdot \vec{u}|\}$.

Recall that the perceptron algorithm adjusts $\vec{c}$ in each iteration. Let k be the total number of adjustments. Denote by $\vec{c}_i$ ($i \geq 1$) the value of $\vec{c}$ after the i-th adjustment; and define $\vec{c}_0 = (0, ..., 0)$.

Claim 1: $|\vec{c}_k| \geq \vec{c}_k \cdot \vec{u} \geq k\gamma_{opt}$.

Proof: We will first prove: for any $i \geq 0$, it holds that.

$$\vec{c}_{i+1} \cdot \vec{u} \geq \vec{c}_i \cdot \vec{u} + \gamma_{opt}. \tag{1}$$

Due to symmetry, we will prove the above only for the case where $\vec{c}_{i+1}$ is adjusted from $\vec{c}_i$ due to a violation point $\vec{p}$ of label 1. In this case, $\vec{c}_{i+1} = \vec{c}_i + \vec{p}$; and hence, $\vec{c}_{i+1} \cdot \vec{u} = \vec{c}_i \cdot \vec{u} + \vec{p} \cdot \vec{u}$. From the definition of γ_{opt}, we know that $\vec{p} \cdot \vec{u} \geq \gamma_{opt}$, which gives (1).

It then follows from (1) that

$$\begin{aligned}
|\vec{c}_k| &\geq \vec{c}_k \cdot \vec{u} \\
&\geq \vec{c}_{k-1} \cdot \vec{u} + \gamma_{opt} \\
&\geq \vec{c}_{k-2} \cdot \vec{u} + 2\gamma_{opt} \\
&\dots \\
&\geq \vec{c}_0 + k\gamma_{opt} = k\gamma_{opt}.
\end{aligned}$$

□

Claim 2: $|\vec{c}_{i+1}| \leq |\vec{c}_i| + R$.

Proof: We will prove only the case where $\vec{c}_{i+1}$ is adjusted from $\vec{c}_i$ using a violation point $\vec{p}$ of label 1. In this case:

$$|\vec{c}_{i+1}| = |\vec{c}_i + \vec{p}| \leq |\vec{c}_i| + |\vec{p}| \leq |\vec{c}_i| + R.$$

□

Claim 3: $|\vec{c}_{i+1}| \le |\vec{c}_i| + \frac{R^2}{2|\vec{c}_i|} + \frac{\gamma_{opt}}{2}$.

Proof: We will prove only the case where $\vec{c}_{i+1}$ is adjusted from $\vec{c}_i$ using a violation point $\vec{p}$ of label 1. In other words, $\vec{c}_{i+1} = \vec{c}_i + \vec{p}$. Hence:

$$\begin{aligned} |\vec{c}_{i+1}|^2 &= \vec{c}_{i+1} \cdot \vec{c}_{i+1} = (\vec{c}_i + \vec{p})^2 = \vec{c}_i \cdot \vec{c}_i + 2\vec{c}_i \cdot \vec{p} + \vec{p} \cdot \vec{p} \\ &= |\vec{c}_i|^2 + 2\vec{c}_i \cdot \vec{p} + |\vec{p}|^2. \end{aligned}$$

Since p is a violation point, it must hold that $\frac{\vec{c}_i}{|\vec{c}_i|} \cdot \vec{p} < \gamma_{guess}/2 \le \gamma_{opt}/2$. Furthermore, obviously, $|\vec{p}|^2 \le R^2$. We thus have:

$$|\vec{c}_{i+1}|^2 \le |\vec{c}_i|^2 + \gamma_{opt}|\vec{c}_i| + R^2 \le \left(|\vec{c}_i| + \frac{R^2}{2|\vec{c}_i|} + \frac{\gamma_{opt}}{2}\right)^2.$$

The claim then follows. □

Claim 4: When $|\vec{c}_i| \geq \frac{2R^2}{\gamma_{opt}}$, $|\vec{c}_{i+1}| \leq |\vec{c}_i| + (3/4)\gamma_{opt}$.

Proof: Directly follows from Claim 3. □

Claim 5: $|\vec{c}_k| \le \frac{2R^2}{\gamma_{opt}} + \frac{3k\gamma_{opt}}{4} + R$.

Proof: Let j be the largest i satisfying $|\vec{c}_i| < \frac{2R^2}{\gamma_{opt}}$. If $j = k$, then $|\vec{c}_k| < \frac{2R^2}{\gamma_{opt}}$, and we are done. Next, we focus on the case $j < k$; note that this means $|\vec{c}_{j+1}|, |\vec{c}_{j+2}|, ..., |\vec{c}_k|$ are all at least $2R^2/\gamma_{opt}$.

$$\begin{aligned}
|\vec{c}_k| &\le |\vec{c}_{k-1}| + (3/4)\gamma_{opt} && \text{(Claim 4)} \\
&\le |\vec{c}_{k-2}| + 2 \cdot (3/4)\gamma_{opt} && \text{(Claim 4)} \\
&... \\
&\le |\vec{c}_{j+1}| + (k-j-1)(3/4)\gamma_{opt} && \text{(Claim 4)} \\
&\le |\vec{c}_{j+1}| + (3k/4)\gamma_{opt} \\
&\le |\vec{c}_j| + R + (3k/4)\gamma_{opt} && \text{(Claim 2)} \\
&\le \frac{2R^2}{\gamma_{opt}} + R + (3k/4)\gamma_{opt}.
\end{aligned}$$

□

Combining Claims 1 and 5 gives:

$$\begin{aligned} k\gamma_{opt} &\le \frac{2R^2}{\gamma_{opt}} + \frac{3k\gamma_{opt}}{4} + R \quad \Rightarrow \\ k &\le \frac{8R^2}{\gamma_{opt}^2} + \frac{4R}{\gamma_{opt}} \\ (\text{by } R \ge \gamma_{opt}) &\le \frac{8R^2}{\gamma_{opt}^2} + \frac{4R^2}{\gamma_{opt}^2} \\ &\le \frac{12R^2}{\gamma_{opt}^2}. \end{aligned}$$

This completes the proof of the theorem.