

SVM 支援向量機演算法-原理篇

原創 小碼 碼農充電站pro 2021年01月20日 10:26

本篇來介紹**SVM 演算法**，它的英文全稱是 *Support Vector Machine*，中文翻譯為**支援向量機**。

之所以叫作支援向量機，是因為該演算法最終訓練出來的模型，由一些**支援向量**決定。所謂的支援向量，也就是能夠決定最終模型的向量。

SVM 演算法最初是用來解決**二分類問題**的，而在這個基礎上進行擴展，也能夠處理**多分類問題**以及**回歸問題**。

1, SVM 演算法的歷史

早在1963年，著名的前蘇聯統計學家**弗拉基米爾·瓦普尼克**在讀博士期間，就和他的同事**阿列克謝·切爾沃寧基斯**共同提出了支援向量機的概念。

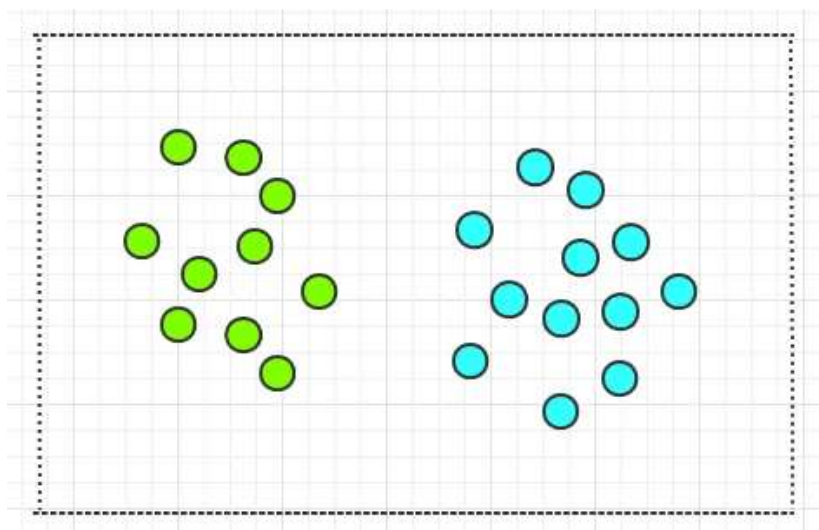
但由於當時的國際環境影響，他們用俄文發表的論文，並沒有受到國際學術界的關注。

直到20世紀90年代，**瓦普尼克**隨著移民潮來到美國，而後又發表了**SVM 理論**。此後，**SVM 演算法**才受到應有的重視。如今，**SVM 演算法**被稱為最好的監督學習演算法之一。

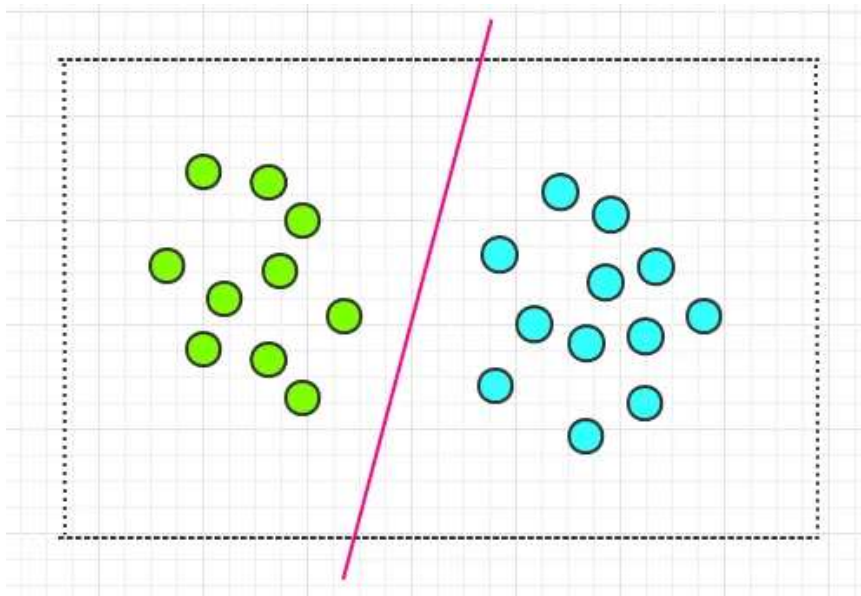
2, 線性可分的 SVM

SVM 演算法最初用於解決二分類問題，下面我們以最簡單的**二維平面**上的，**線性可分**的數據點來介紹支援向量機。

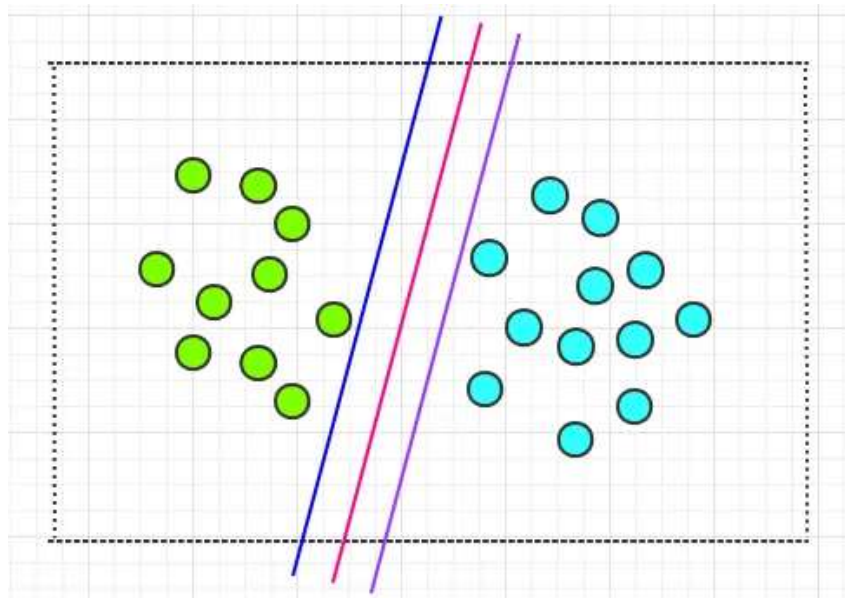
假設平面上有一些不同顏色的圓圈，這些圓圈是線性可分的，也就是可用一條直線分開。如下：



現在想在平面上畫出一條直線，將這些圓圈分開。通過觀察，你很容易就能畫出一條直線，如下：

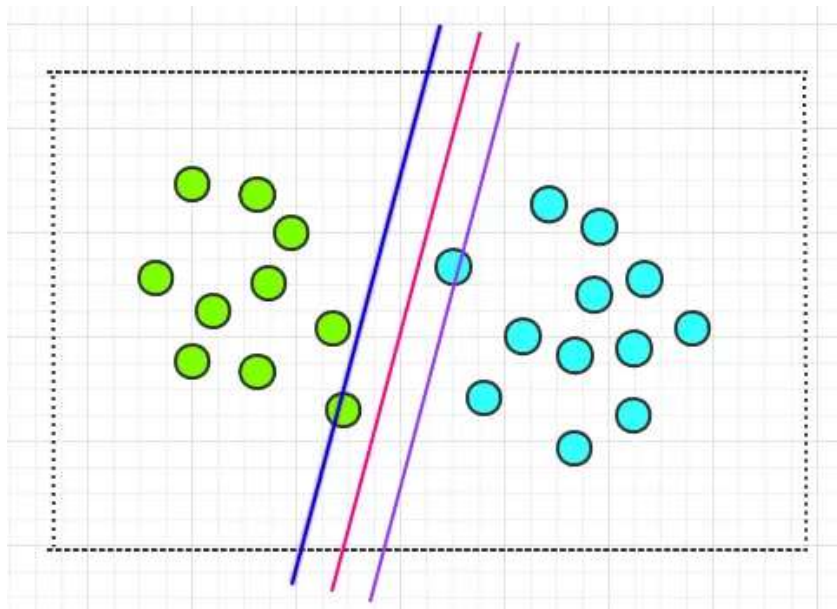


但是這樣的直線會有很多，它們都能正確的劃分兩類圓圈，就像下面這幅圖中的一樣：



那麼哪條直線才是最好的呢？通過肉眼我們無法找到那條最好的直線。但是就上圖中的三條直線而言，明顯你會覺得中間那條紅線，會比兩側的兩條線要更好。

因為，如果有一些圓圈往中間靠攏，那麼兩側的那兩條直線就不能將兩種圓圈劃分開了。而中間那條直線依然可以劃分兩種圓圈。如下：



因此，中間那條紅線會比兩側的兩條直線更好，更安全。

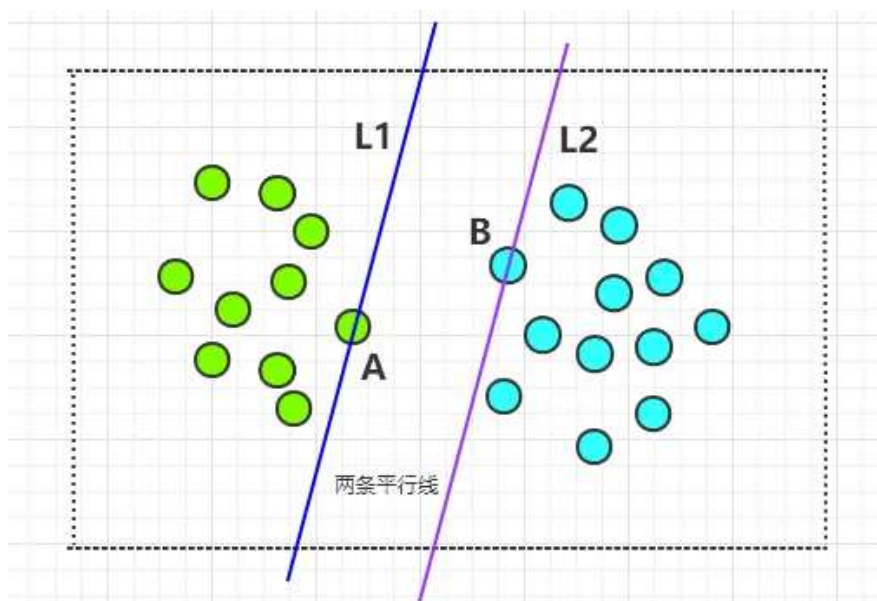
雖然通過肉眼我們能知道哪條直線**更好**，但是怎樣才能找到**最好**的那條直線呢？而 SVM 演算法就可以幫我們找到那條最好的直線。

3，找到最好的直線

下面我們來看下如何找到最好的那條直線。

首先，找**兩條平行線**，其中一條線經過一個綠色圓圈，剩下的所有綠色圓圈都在這條線的左側；另一條線經過一個淡藍色的圓圈，剩下的所有淡藍色圓圈都在這條線的右側。

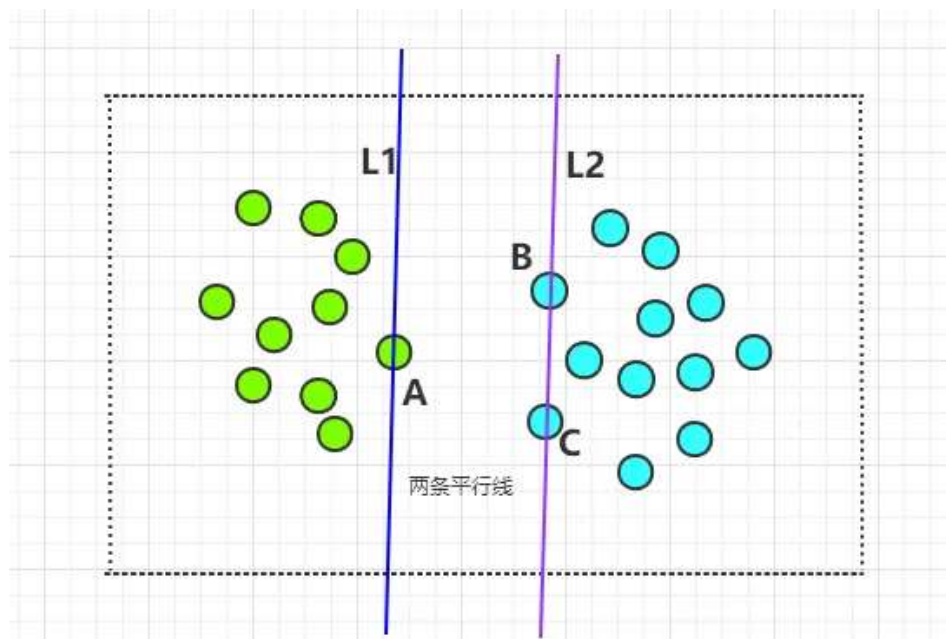
如下圖所示的兩條平行線 **L1** 和 **L2**，**L1** 經過了 **A** 圓圈，**L2** 經過了 **B** 圓圈：



接下來，讓 **L1** 以 **A** 為中心進行旋轉，**L2** 以 **B** 為中心進行旋轉，在旋轉的過程中，要保證下面兩個原則：

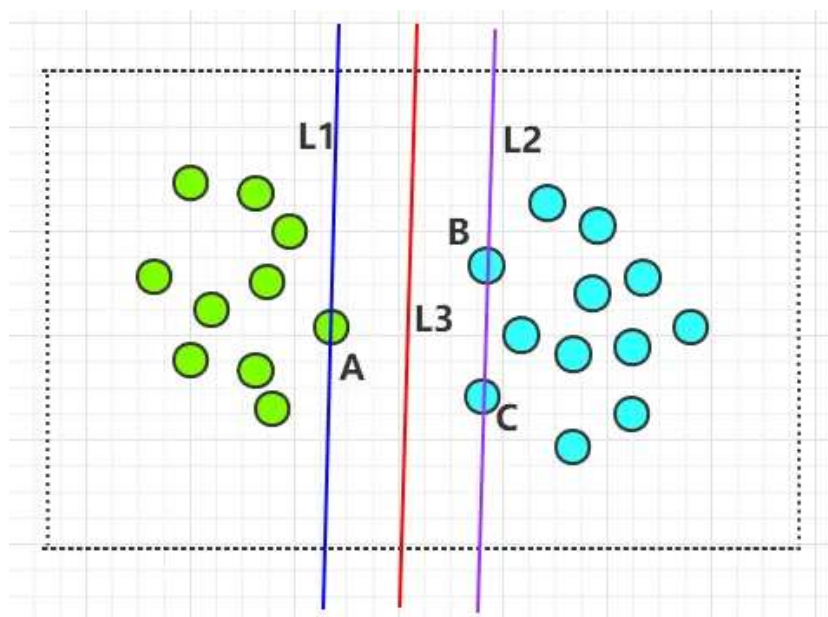
- L1 與 L2 保持平行。
- 除 A 之外的所有綠色圓圈都在 L1 的左側，除 B 之外的所有淡藍色圓圈都在 L2 的右側。

在保證上面兩個原則的前提下旋轉，直到其中的一條線經過第三個圓圈為止。如下：



此時，L2 經過了第三個圓圈 C。

到此為止，我們要找的那條最好的，能夠劃分兩種圓圈的直線，就是 L1 和 L2 中間的那條直線，如下圖中的 L3：



4 · SVM 演算法

SVM 演算法的求解過程，就是尋找那條最佳直線的過程。

在 SVM 演算法中，上面的 **A**，**B**，**C** 三個點叫作**支援向量**，直線 **L3** 是最佳直線，擴展到多維空間，這條直線叫作**超平面**。SVM 演算法的目的就是找到這個**超平面**，這個超平面是劃分數據點的最佳平面。

關於**支援向量**，更重要的是，SVM 訓練出的最終模型，只與**支援向量**有關，這也是**支援向量機**的名稱來源。就像 SVM 的發明者瓦普尼克所說的那樣：**支援向量機**這個名字強調了這類演算法的關鍵是如何根據**支援向量**構建出解，演算法的複雜度也主要取決於**支援向量的數目**。

直線 **L1** 和 **L2** 之間有**無數條平行直線**都可以將兩類圓圈分開，支援向量到這些平行直線的距離叫作**分類間隔**，而**間隔最大**的那條直線就是最佳直線，也就是 **L3**。

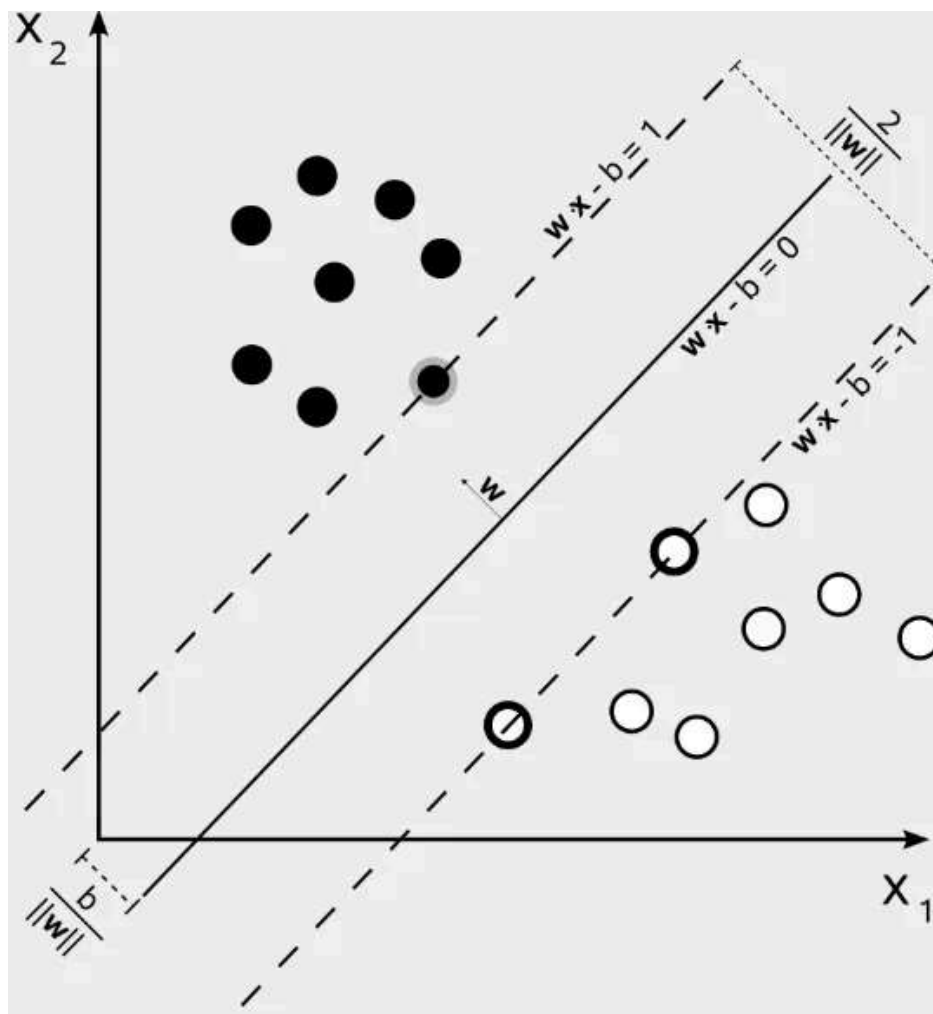
多維空間中的超平面可以用線性方程來表示：

$$\mathbf{w}^T \cdot \mathbf{x} + b = 0$$

其中的 **W** 為法向量，它決定了超平面的方向。**b** 為截距，它決定了超平面與空間原點的距離。

超平面將特徵空間分為兩部分，法向量所指向的一側的數據為正類，標記為 **+1**；另一側的數據為負類，標記為 **-1**。

二維座標系中的線性方程如下圖所示：

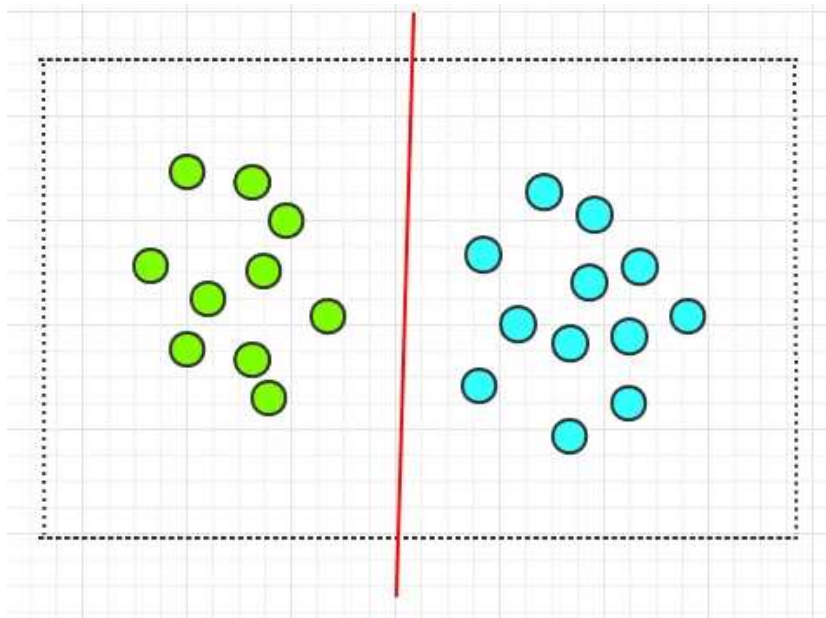


支援向量機的學習過程是一個凸二次規劃問題，可以用SMO演算法高效求解。

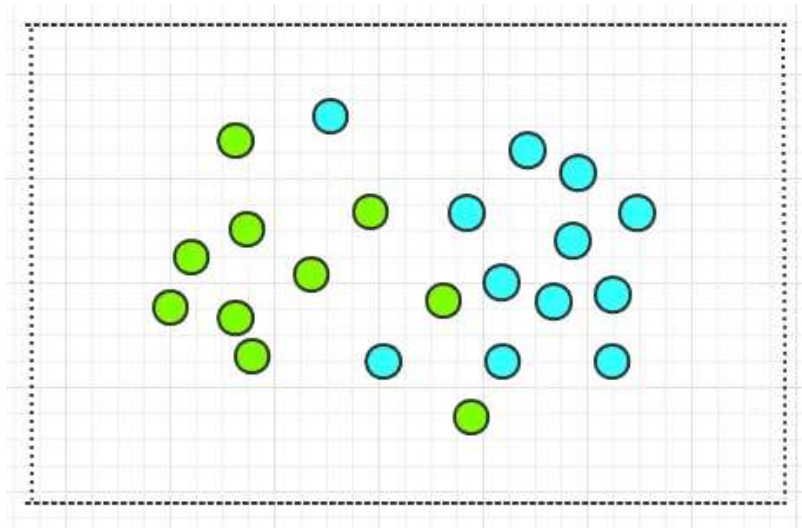
SMO 全稱為 *Sequential Minimal Optimization*，中文稱為序列最小優化，由 John Platt 於 1996 年發表。

5，線性 SVM

上文中介紹到的兩類圓圈是線性可分的，也就是可以用一條直線將所有不同的圓圈劃分開，而不允許有一個例外。如下：



但現實生活中的數據，往往是線性不可分的，如下圖中的兩類圓圈，就不可能找到一條直線將其分開：

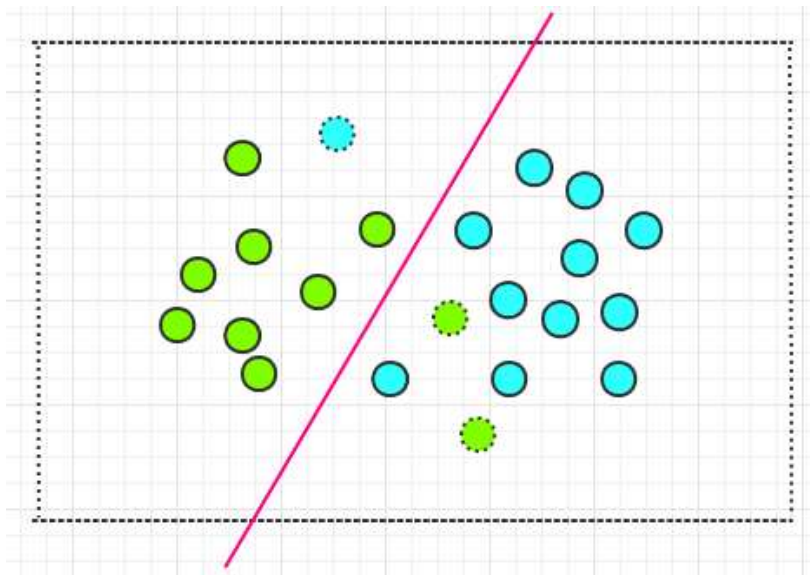


使用**線性可分的 SVM** 是無法解決線性不可分問題的。要想解決線性不可分問題，就得將支援向量機一般化，這種向量機稱為**線性 SVM**（去掉了可分二字）。

線性 SVM 比**線性可分SVM** 更具**通用性**。**線性可分SVM** 尋求的是**硬間隔**最大化，而**線性 SVM**尋求的是**軟間隔**最大化。

硬體隔不允許有一個例外的點被錯分，而軟間隔則允許個別（不會很多）的點被錯分。

在線性不可分的數據集中，導致不可分的只是**少量異常點**，只要把這些異常點去掉，餘下的大部分樣本點依然滿足線性可分的條件。如下圖所示：

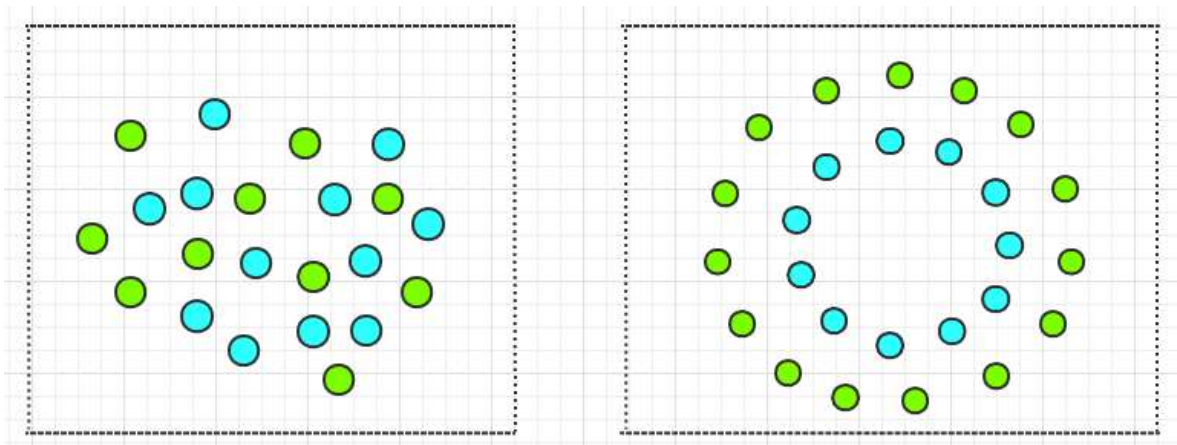


6，非線性問題

不管是線性可分 SVM 還是線性 SVM，它們都只能處理線性問題：

- 線性可分 SVM：存在一條直線將所有的點正確劃分。
- 線性 SVM：存在一條直線將絕大部分點正確劃分，而允許極個別點錯分。這些極個別點可稱為噪點或異常點。

與線性問題相對應的是非線性問題，比如下面兩種情況都是非線性問題：



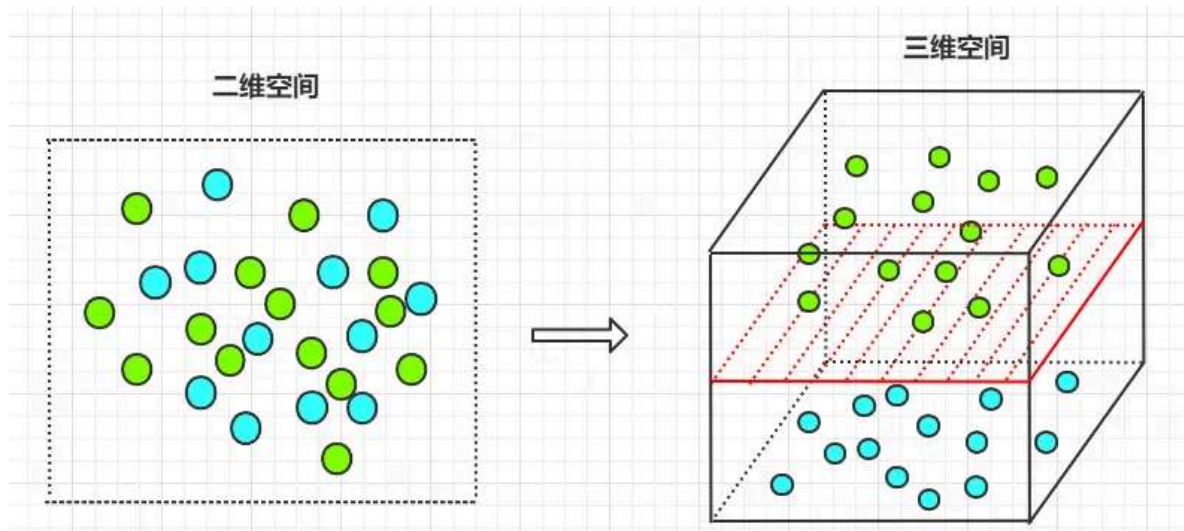
如何讓 SVM 處理非線性問題呢？

7，核函數

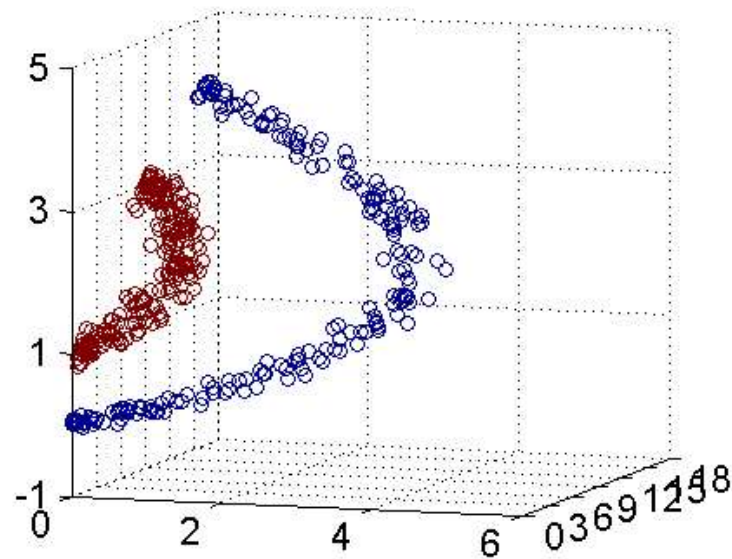
既然在二維空間中，不能找到一條直線來處理非線性問題。那麼是否可以將二維空間映射成三維空間，從而找到一個平面，來處理二維空間不能處理的問題呢？

答案是肯定的，如果樣本的屬性數有限，那麼就一定存在一個高維特徵空間使樣本線性可分。

如下圖所示，在二維空間時，不能找到一條直線將其正確的分類; 將其映射到三維空間后，可以找到一個平面，將其正確分類：



從下面這幅動圖可直觀的看到，二維空間的線性不可分映射到三維空間線性可分的轉換過程：



將原始低維空間上的非線性問題轉化為新的高維空間上的線性問題，這就是**核技巧**的基本思想。

支援向量機誕生於1963年，核技巧誕生於1995年。

在支援向量機中，**核函數**可以將樣本從低維空間映射到高維空間，從而使得**SVM**可以處理非線性問題。常用的核函數有以下幾種：

- 線性核函數
- 多項式核函數
- 高斯核函數
- 拉普拉斯核函數

- Sigmoid 核函數

不同的核函數對應不同的映射方式。在SVM 演算法中，核函數的選擇非常關鍵，好的核函數會將樣本映射到合適的特徵空間，從而訓練出更優的模型。

核函數將線性SVM擴展成了非線性 SVM，使得SVM更具普適性。

8. 多分類問題

SVM 演算法最初用於處理二分類問題，那它如何處理多分類問題呢？

對於多分類問題，可以將多個二分類器組合起來形成一個多分類器，常見的方法有一對多法和一對一法。

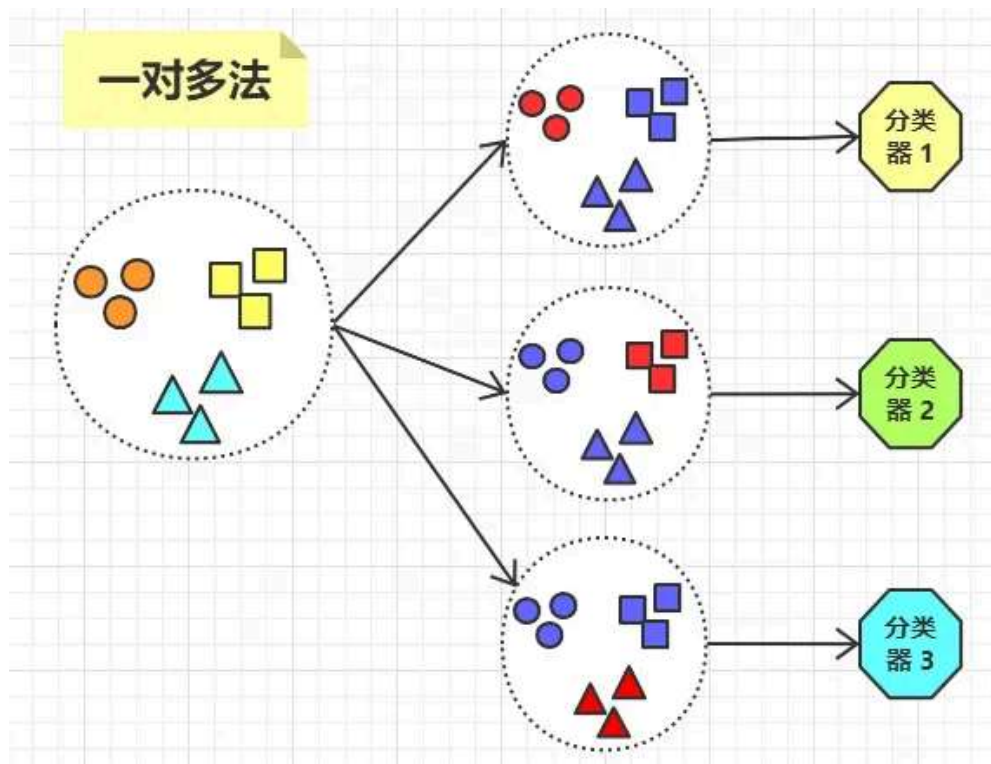
一對多法

對於有 K 個分類的數據集，把其中的一類作為正集，其它的 $K-1$ 類作為負集，這樣就把數據集分了正負兩類。

比如我們有一個數據集，該數據集有三個分類 A 、 B 、 C ，那麼將訓練出三個分類器：

- 將分類 A 作為正集， B 、 C 作為負集，訓練出一個分類器。
- 將分類 B 作為正集， A 、 C 作為負集，訓練出一個分類器。
- 將分類 C 作為正集， A 、 B 作為負集，訓練出一個分類器。

如下圖所示：



對於 K 個分類的數據集，一對多法會訓練出 K 個分類器。缺點是，每次訓練分類器時，負樣本數都大大多於正樣本數，造成樣本分佈不均。

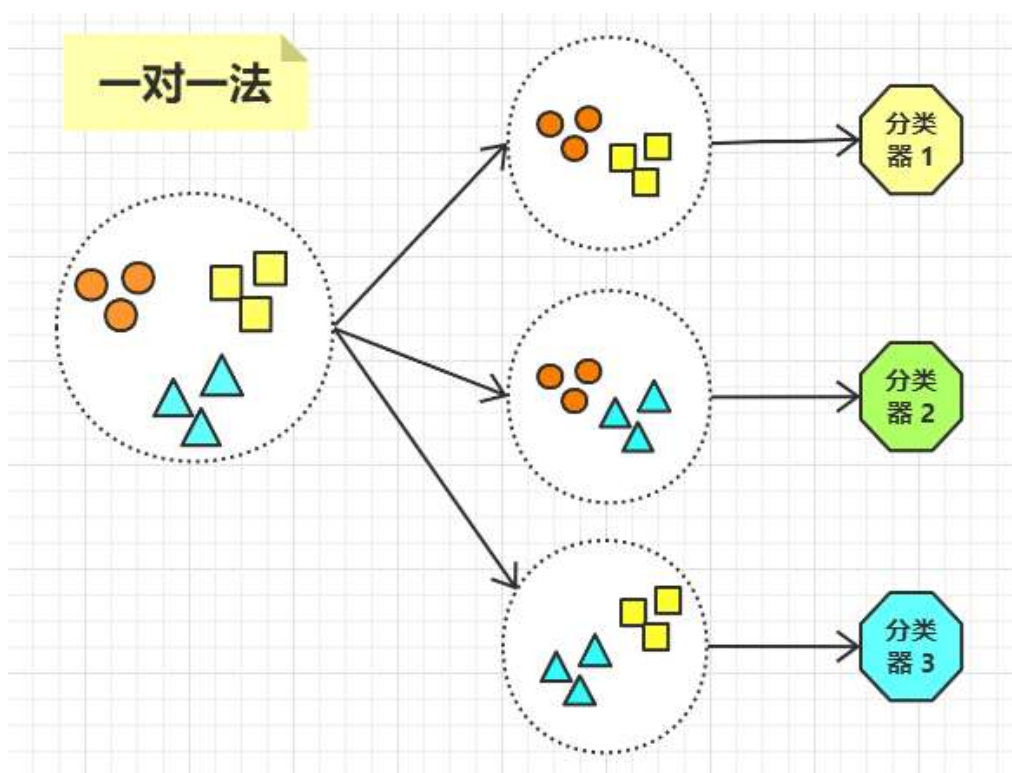
一對一法

對於有 K 個分類的數據集，任選其中的兩類樣本，然後訓練出一個分類器。

比如我們有一個數據集，該數據集有三個分類 A 、 B 、 C ，那麼將訓練出三個分類器：

- 選 A 、 B 兩類數據，訓練出一個分類器。
- 選 A 、 C 兩類數據，訓練出一個分類器。
- 選 B 、 C 兩類數據，訓練出一個分類器。

如下圖所示：



那麼這種方法將會訓練出 $C(K, 2)$ 個分類器，因此這種方法的缺點是，如果數據集的分類較多的話，訓練出的分類器將會非常多。

9. 總結

SVM 演算法通過尋找一個**超平面**來處理分類問題。這個超平面只與幾個**支援向量**有關，而不會受其它的普通向量的影響。

SVM 可分為**線性 SVM** 和**非線性 SVM**，線性 SVM 只能處理線性問題，而不能處理非線性問題。

核函數可以將低維空間的非線性問題，轉化為高維空間的線性問題，這就將線性 SVM 擴展成了非線性 SVM，從而使得 SVM 可以處理非線性問題。

SVM 演算法最初用來處理**二分類問題**，對於**多分類問題**，可以將多個二分類器組合在一起成為一個多分類器，從而處理多分類問題。方法有一對多法和一對一法。

下一節將介紹如何用SVM處理實際問題。

(本節完。)



點擊查看往期內容回顧

AdaBoost 演算法-分析波士頓房價數據集

RandomForest 隨機森林演算法與模型參數的調優

決策樹演算法-實戰篇-鳶尾花及波士頓房價預測

Apriori 演算法-如何進行關聯規則挖掘

Logistic 回歸-原理及應用



歡迎關注作者公眾號，獲取更多技術乾貨。



微信搜一搜



码农充电站pro



