

SVM 訓練過程

SVM 目標在於找到一條線(2 維)或一個平面(3 維)來將不同類型的資料做區隔，以線來說，雖然有很多條都能達到這個目的，但 SVM 想找那條**最中間、最穩定**的線。而最中間就是看這條線到「最近的點」的距離，這個距離叫 **margin**。

- 1.margin 大 → 分類更穩定
- 2.margin 小 → 分類容易被錯一點點就誤判

然後最靠近的點叫**支援向量**，用來決定所要求的線。而最佳 w 其實就是這些支援向量加起來，算出來的方向向量。也就是說找最佳 w 的過程，其實就是一個挑出支援向量 → 算出中間線的過程。

我們要「最大化間隔 (margin)」，等價於「最小化權重的長度」：

$$\min_{w,b} \frac{1}{2} \|w\|^2$$

條件是：

$$y_i(w \cdot x_i + b) \geq 1, \quad \forall i$$

這就是 SVM 的核心最佳化問題。

圖 1、間隔(margin)計算式

接著，首先我們的目標如下圖：

找一條線（或超平面）：

$$w \cdot x + b = 0$$

可以 完美區分類別，而且兩邊的距離 (margin) 最大。

圖 2、SVM 目標

其中訓練過程就如下方所示：

1. 設定目標函數：

我們要讓 margin 最大化 \rightarrow 相當於要讓 $\|w\|$ 最小化。

$$\min_{w,b} \frac{1}{2} \|w\|^2$$

條件是：

$$y_i(w \cdot x_i + b) \geq 1$$

2. 引入拉格朗日乘子 (Lagrange Multipliers)：

把限制條件放進去，建立拉格朗日函數：

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i [y_i(w \cdot x_i + b) - 1]$$

3. 對 w, b 求偏導，讓它等於 0：

$$\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_i \alpha_i y_i x_i$$

代表 w 是由各樣本的線性組合組成（但只有支援向量的 $\alpha_i > 0$ ）。

4. 把它帶回去 \rightarrow 解對偶問題 (Dual Problem)：

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (x_i \cdot x_j)$$

用數學最佳化演算法（像 SMO）解出最佳 α_i 。

5. 算出最終結果：

$$w^* = \sum_i \alpha_i y_i x_i$$

$$b^* = y_k - w^* \cdot x_k \quad (k \text{ 為支援向量之一})$$

圖 3、SVM 訓練過程

MLP 訓練過程

MLP 的一開始會有一個輸入 X ，而我們希望它的預測輸出值能夠盡可能接近真實輸出，但一開始的權重 W 、 b 都是隨機的，需用損失函數計算預測和實際相差多少，再藉由反向傳播算出權重對損失的影響，最後再用梯度下降法來更新權重使預測輸出逐漸接近實際輸出，有點類似於疊代調整的概念。

其中 MLP 的目標就是要找出一組權重，讓神經網路的輸出值盡可能接近實際結果，而下圖就是其訓練過程。

⚙️ 訓練過程 (找最佳 W)

1. 初始化權重：

一開始所有權重 W 都是隨機的。

2. 正向傳遞 (Forward Propagation) :

資料從輸入層 → 隱藏層 → 輸出層流動，

每層都做：

$$z = W \cdot x + b$$

再經過非線性函數 (ReLU、Sigmoid、Tanh) 得到輸出。

3. 計算誤差 (Loss) :

比較預測結果 y_{pred} 與真實值 y ：

$$\text{Loss} = \frac{1}{2}(y - y_{pred})^2$$

4. 反向傳播 (Backpropagation) :

根據誤差計算每個權重對誤差的影響 (梯度)。

例如：

$$\frac{\partial \text{Loss}}{\partial W}$$

表示如果改變這個權重，Loss 會變多少。

5. 梯度下降 (Gradient Descent) :

按照「誤差下降的方向」去調整權重：

$$W_{\text{new}} = W_{\text{old}} - \eta \frac{\partial \text{Loss}}{\partial W}$$

其中 η 是學習率 (控制更新的步伐)。

6. 重複步驟 2-5 :

一直更新權重，直到誤差夠小或達到迭代上限。

圖 4、MLP 訓練過程