

K-Nearest Neighbors

Probabilistic Approach

林哲緯

February 3, 2025

1 K-近鄰演算法

有時候我們對資料的來源並非一無所知，但是採用最小平方法的迴歸模型，並沒有充分利用資料本身的訊息，譬如資料的變異性。我們希望資料的來源或資料的本身可以提供更多的訊息，做為新資料所屬群組的判別依據。這樣的想法把問題帶進「機率」的範疇來解決。

假設 Y 及 $f(X)$ 分別代表輸出變數與輸出預測值，其中 $X \in \mathbb{R}^p$ 表示有 p 個輸入變數。我們期望輸出值與預測值的誤差愈小愈好，如果輸入變數 X 與輸出變數 Y 的聯合機率密度函數 $Pr(X, Y)$ 已知的話，這個問題可以寫成：

$$\min_{f(X)} E_{XY} [(Y - f(X))^2] \quad (1)$$

也就是找一個輸入與輸出變數間的關係式 $f(\cdot)$ ，使得真正的輸出值 Y 與其預測值 $f(X)$ 間的誤差的平方期望值越小越好。有別於不論機率特性的「最小平方法 (Least Squared Errors, LSE)」，這個方法稱為「最小均方誤差 (Minimum Mean Squared Error, MMSE)」。在已知樣本值 $X = \mathbf{x}$ 的條件下，它的最佳解如 (未知函數 $f(X)$ 的最佳選擇)

$$y = \hat{f}(\mathbf{x}) = E_{Y|X} (Y | X = \mathbf{x}) \quad (2)$$

其中 X, Y 代表輸入輸出變數， \mathbf{x} 與 y 表示輸入值及輸出的預測值 (或稱擬合值)。式 (2) 說明當輸入值為 \mathbf{x} 時，最佳的輸出預測值為輸出變數的「條件式均值 (Conditional Mean)」。

接下來的問題是如何計算 $y = E_{Y|X} (Y | X = \mathbf{x})$ ？期望值代表的是理論值，至於要如何落實到實際的應用呢？或說若不知道機率密度函數 $Pr(Y | X)$ ，如何得到這個期望

值呢？實務的作法，一般都是利用平均數來估計這個期望值。譬如式(3)是個不錯的估計式

$$\hat{y} = Ave(y_i \mid X = \mathbf{x}) \quad (3)$$

其中 $Ave(\cdot)$ 代表求平均值。這個估計式解讀為「將輸入資料為 \mathbf{x} 的所有資料，找出對應的所有輸出值 y_i 取平均」。雖然樣本平均數是期望值的不偏估計，不過這個做法面臨實際的困難是已知的多變量連續型資料中，剛好等於 \mathbf{x} 的機率等於 0，估計式(3)在實務上不可行。

將式(3)稍作修改後，下面這個輸出預測值的估計式舒緩了這些困擾。

$$\hat{y} = Ave(y_i \mid \mathbf{x}_i \in N_K(\mathbf{x})) = \frac{1}{K} \sum_{\mathbf{x}_i \in N_k(\mathbf{x})} y_i \quad (4)$$

式(4)解讀為：從已知的資料中找到 K 個最靠近 \mathbf{x} 的資料（這是 $N_K(\mathbf{x})$ 的意義），將這些鄰近的 K 筆資料所對應的 y 值平均起來作為「條件式均值」的估計，這個方法叫做 K Nearest-Neighbor method。目前為止所提到的輸出變數並不侷限任何型態，但若輸出變數 Y 的群組屬性屬類別資料時，如式(1)的 MMSE 問題可以寫成

$$\min_{g(X)} E_{XG} [L(G, g(X))] \quad (5)$$

由於是類別資料的關係，其輸出群組變數改寫為 G ，預測群組寫成 $g(X)$ ，兩者的誤差以「Loss function」 $L(G, g(X))$ 取代原先的平方差。當 $L(G, g(X))$ 定義為

$$L(G, g(X)) = \begin{cases} 0 & \text{if } G = g(X) \\ 1 & \text{if } G \neq g(X) \end{cases}$$

式(5)的最佳解為

$$\hat{g}(X) = g_k \text{ if } Pr(g_k \mid X = \mathbf{x}) = \max_{g \in G} Pr(g \mid X = \mathbf{x}) \quad (6)$$

其中 g_k 代表第 k 個群組 (group)， G 是所有群組的集合。這個結果說明：當輸入值為 \mathbf{x} 時，其所屬群組的 MMSE 預測為

「在所有的群組中，群組機率密度函數在 \mathbf{x} 處的值為最大者」

又稱為貝式分類器 Bayes classifier。不管哪一種輸出的型態，這裡都使用到「後驗機率」(Posterior Probability)的觀念，也就是當給定輸入變數 $X = \mathbf{x}$ ， Y (或 G) 值的可能性(機率)。

群組判別：從式(6)中似乎看不出一個明顯的「分界線」方程式，無法像迴歸模型或判別式分析那樣根據方程式畫出一條分界線，更何況機率密度函數 $Pr(G | X = \mathbf{x})$ 也是未知。不過如迴歸模型應用在類別資料上，當假設兩個群組的輸出為 0 (群組 g_1) 與 1 (群組 g_2) 時，式(4)可以當作式(6)的估計式，並配合下列的群組判別式

$$\mathbf{x} \in \begin{cases} g_1 & \text{if } \hat{y} \leq 0.5 \\ g_2 & \text{if } \hat{y} > 0.5 \end{cases} \quad (7)$$

式(4)的 $\hat{y} \leq 0.5$ 相當於式(6)的 $Pr(g_1 | X = \mathbf{x}) > Pr(g_2 | X = \mathbf{x})$ 。

2 K-近鄰演算法之訓練

由於 K Nearest-Neighbor method 並沒有定義出一個分界線的方程式，無法在所在的空間明確地畫出群組界線，可以在一定範圍的空間內，將空間等份成格子狀 (grids)，每個格子的座標代表一個資料值 \mathbf{x} ，將每一個座標點當作新資料一樣的拿出來判斷其類別，依式(4)與(7)，為每個座標點依其群組判斷劃上不同的符號或顏色。

式(4)的估計式中需要找出「最靠近 \mathbf{x} 的 K 個已知資料」，這個「靠近」的測量方式可以採用歐式距離 (Euclidean Distance)。假設 x_1, x_2, \dots, x_N 為 N 個已知資料 (含群組別)， \mathbf{x} 為空間中某個待判別群組的資料，程式中需要計算 \mathbf{x} 與所有已知資料的距離，再從中選取最靠近的 K 筆資料，最後再將這 K 筆資料的群組值 (0 或 1) 平均起來，即為式(7)中的 \hat{y} 值

訓練 1.

此訓練將透過資料檔 `1a_1.txt` 繪製分界線及計算訓練之誤判率，如圖 1 所示，圖 1 左上圖為設定 $K = 5$ 時畫出之分界線，右上為設定 $K = 10$ 時畫出之分界線，可以發現左邊能將資料切割得較好，而其誤判率為 0.05 也比右邊誤判率 0.07 低。若將 K 值繼續往上調整，可以觀察出分界線的表現結果並沒有更好，可以推斷出 K 值太小，模型容易受到個別異常點的影響，導致過度擬合 (Overfitting)。反觀，如果 K 值選得過大，模型可能會忽略細節，導致過度簡化 (Underfitting)，難以捕捉數據的真實分佈。

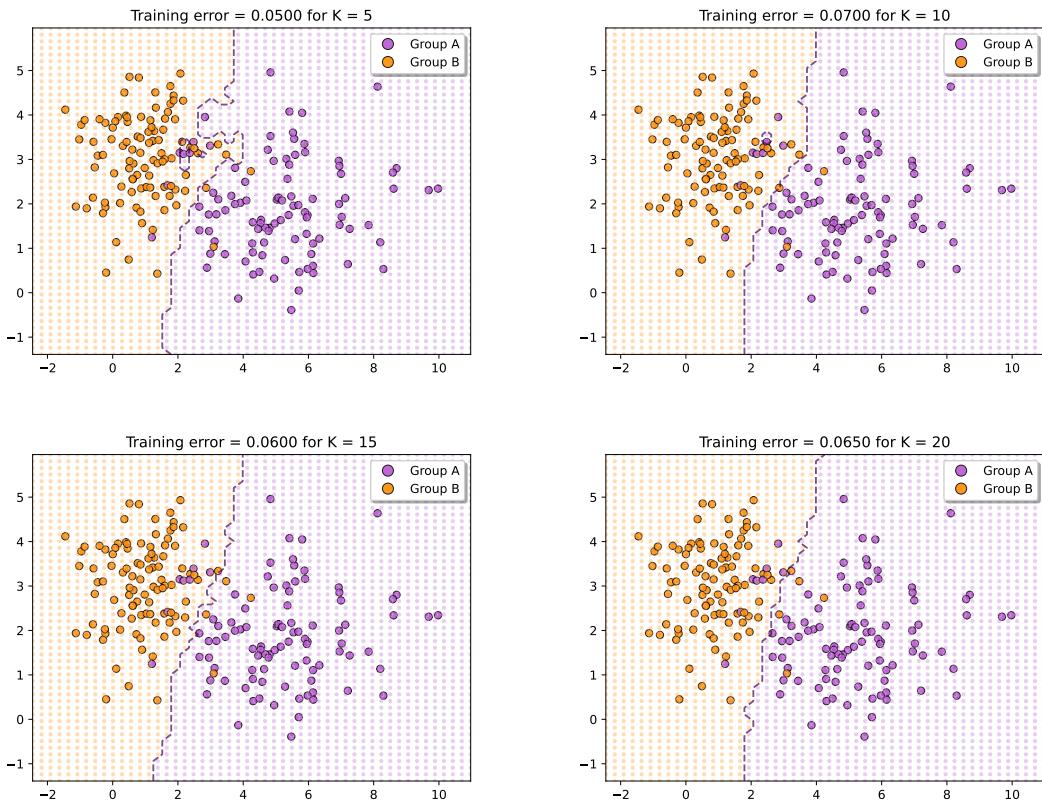
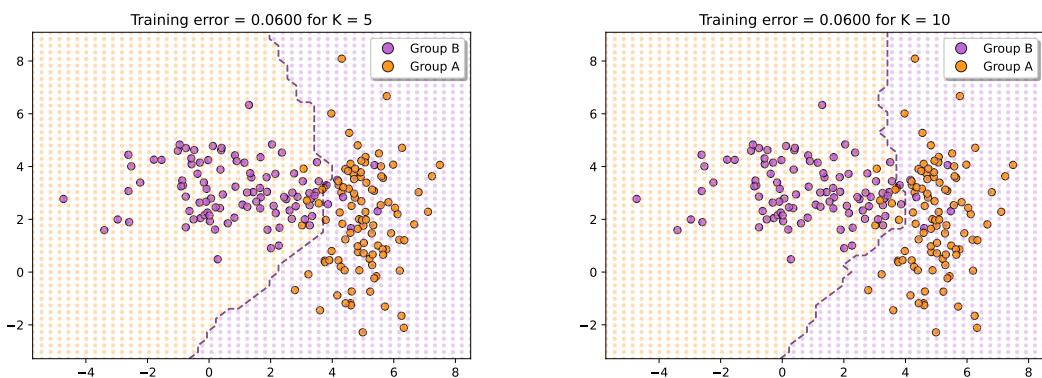


Figure 1: K-近鄰演算法的群組分界線: 資料檔 la_1.txt

訓練 2.

此訓練將透過資料檔 la_2.txt 繪製分界線及計算訓練之誤判率，如圖 2 所示，圖 2 左上圖為設定 $K = 5$ 時畫出之分界線，右上為設定 $K = 10$ 時畫出之分界線，可以發現雖然分界線有些許不同，但其誤判率皆為 0.06。但，當將 K 值往上調整到 $K = 20$ 時，可以發現訓練誤差反而些微上升，來到 0.07。



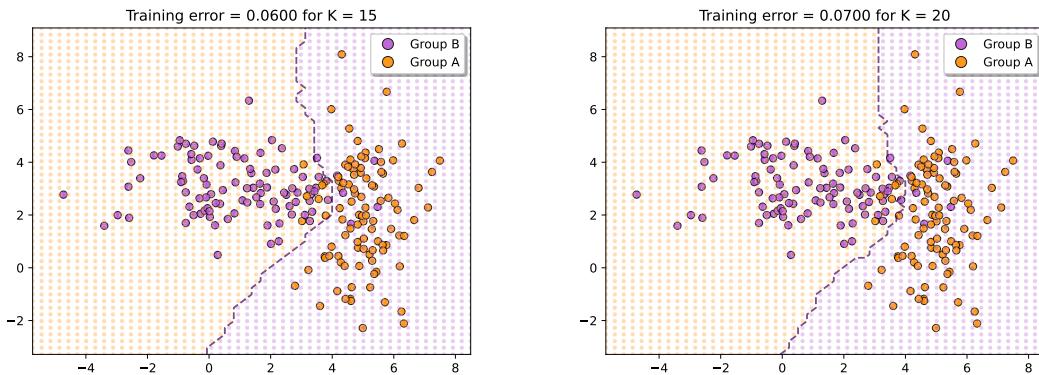


Figure 2: K-近鄰演算法的群組分界線: 資料檔 la_2.txt

訓練 3.

此訓練將透過資料檔 la_3.txt 繪製分界線及計算訓練之誤判率，如圖 3 所示，圖 3 左上圖為設定 $K = 5$ 時畫出之分界線，右上為設定 $K = 10$ 時畫出之分界線，可以發現左邊能將資料切割得較好，而其誤判率為 0.13 也比右邊誤判率 0.155 稍低。

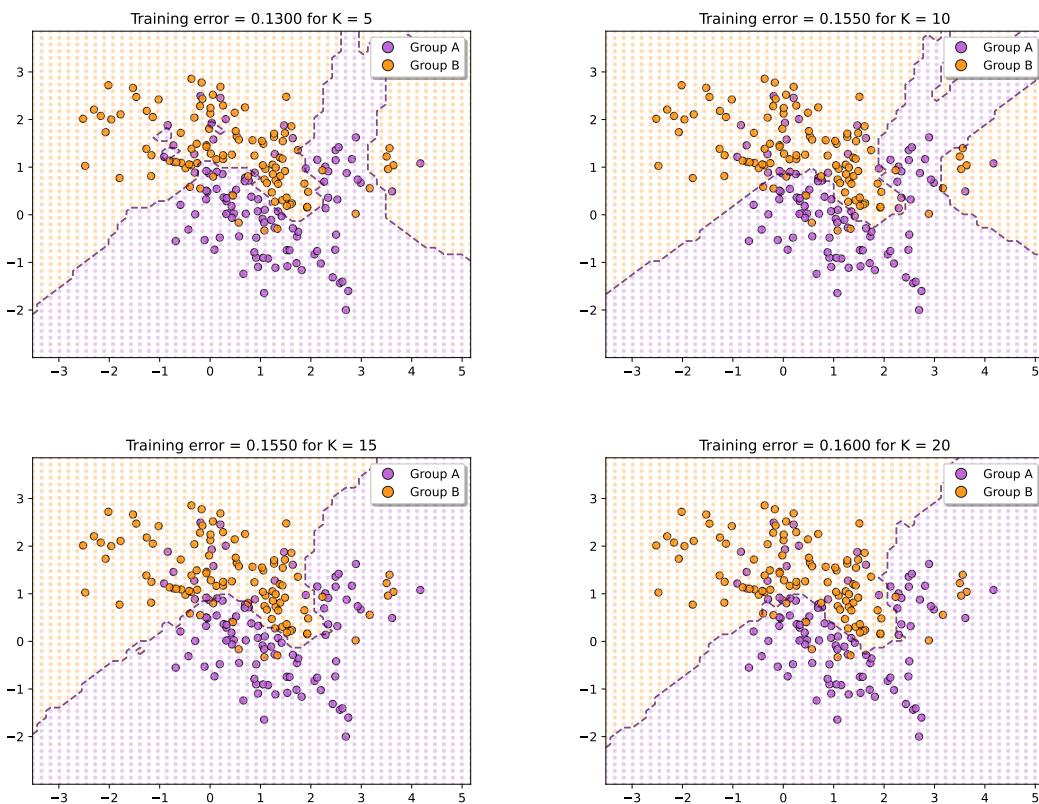


Figure 3: K-近鄰演算法的群組分界線: 資料檔 la_3.txt

3 學習器之評比

在機器學習的領域，將不同的學習方法(模型)通稱為學習器，而學習器的選擇、訓練與評比是機器學習的重要步驟。下列將透過模擬雙變量常態母體的資料，探討學習器面對不同的資料時的表現。

模擬訓練 1. (資料中心位置較遠、分散程度小且共變異矩陣相同)

DATA 1.

Group A :

$$n_1 = 200, \mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Group B :

$$n_2 = 200, \mu_2 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

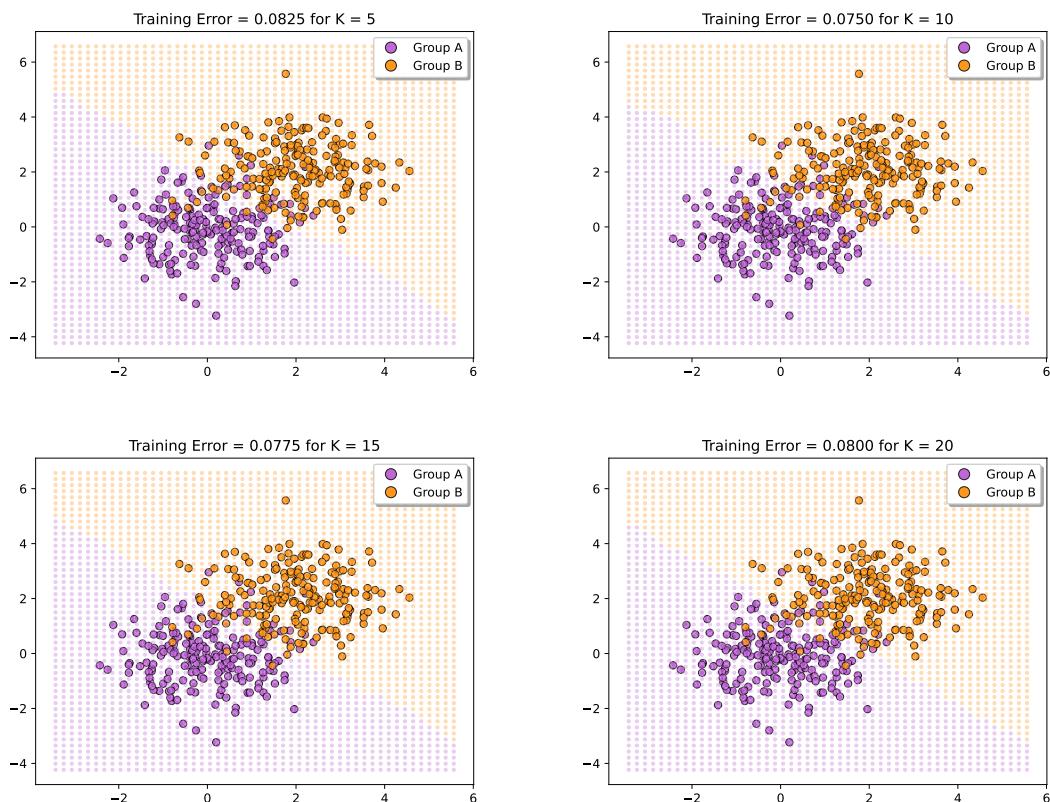


Figure 4: K-近鄰演算法的群組分界線: 模擬訓練 1

模擬訓練 2. (資料中心位置較遠、分散程度大且共變異矩陣不同)

DATA 2.

Group A :

$$n_1 = 200, \mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Group B :

$$n_2 = 200, \mu_2 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$$

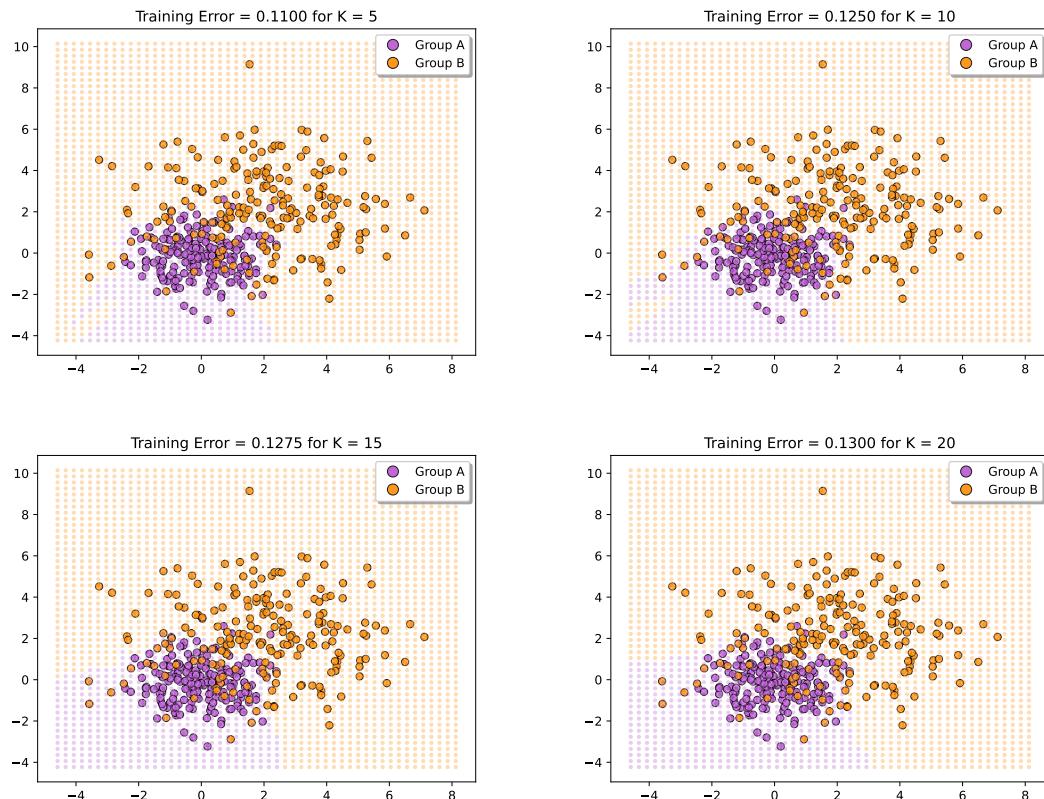


Figure 5: K-近鄰演算法的群組分界線: 模擬訓練 2

模擬訓練 3. (資料中心位置較近、分散程度小且共變異矩陣相同)

DATA 3.

Group A :

$$n_1 = 200, \mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Group B :

$$n_2 = 200, \mu_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

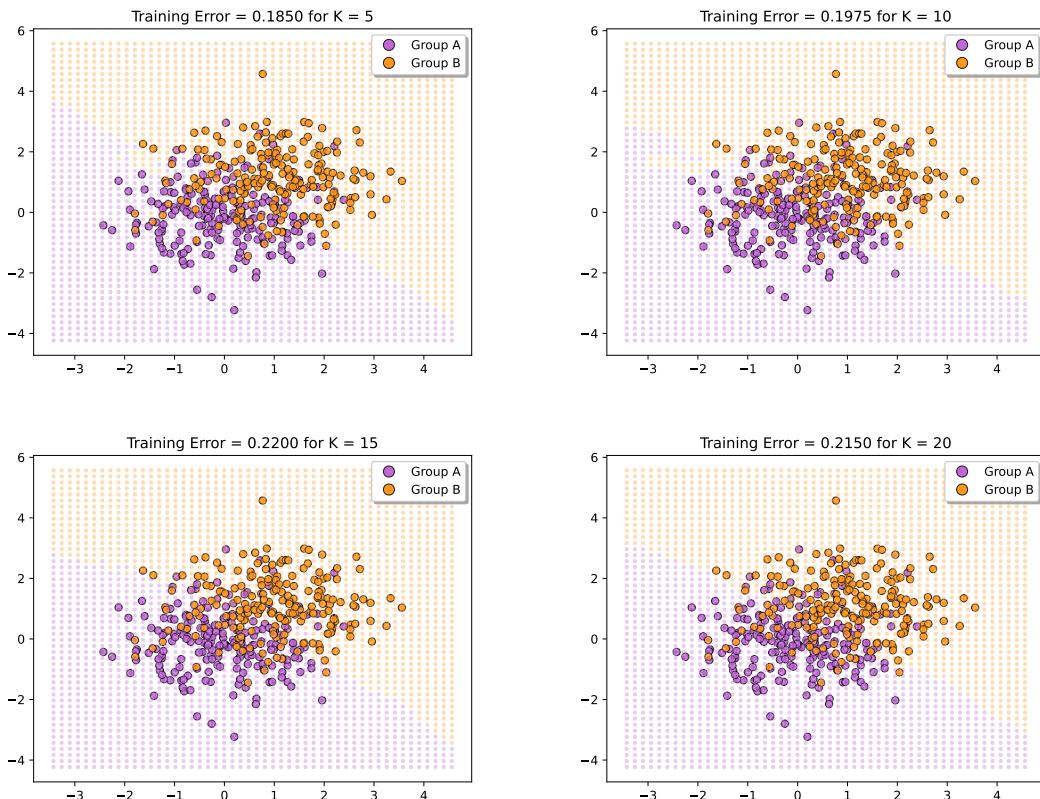


Figure 6: K-近鄰演算法的群組分界線: 模擬訓練 3

模擬訓練 4. (資料中心位置較近、分散程度大且共變異矩陣不同)

DATA 4.

Group A :

$$n_1 = 200, \mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Group B :

$$n_2 = 200, \mu_2 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$$

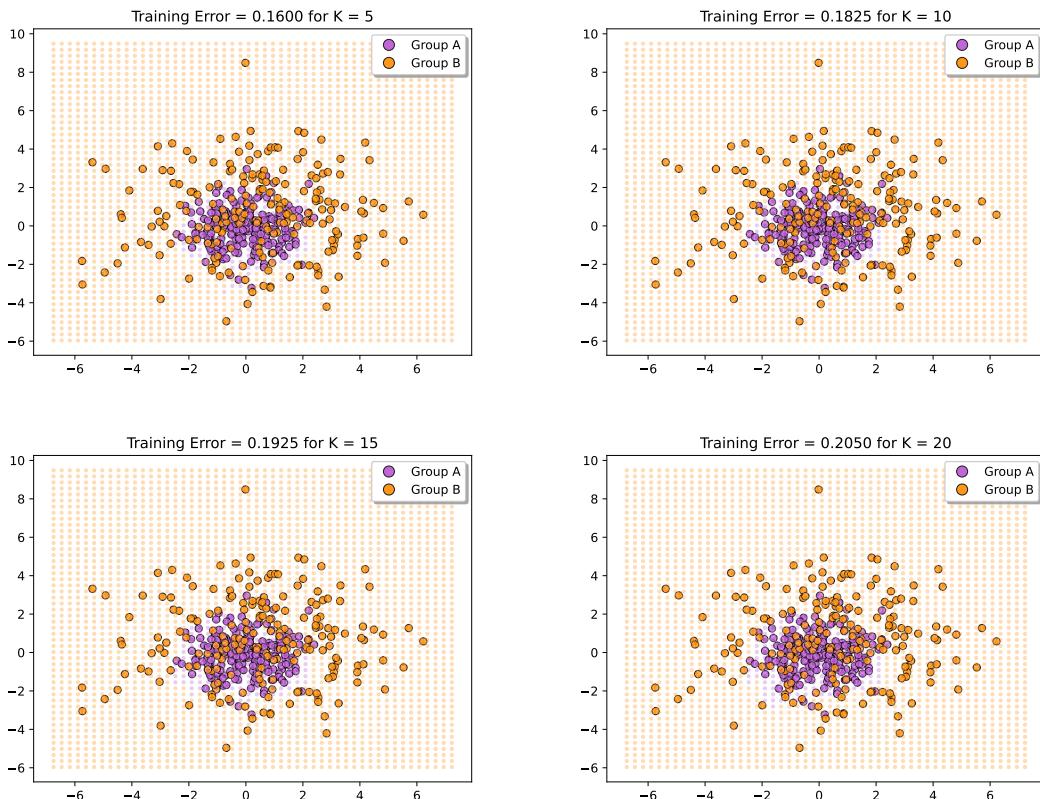


Figure 7: K-近鄰演算法的群組分界線: 模擬訓練 4

模擬訓練 5.(資料分散程度小、 X_1, X_2 具相關性且共變異矩陣相同)

DATA 5.

Group A :

$$n_1 = 200, \mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

Group B :

$$n_2 = 200, \mu_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

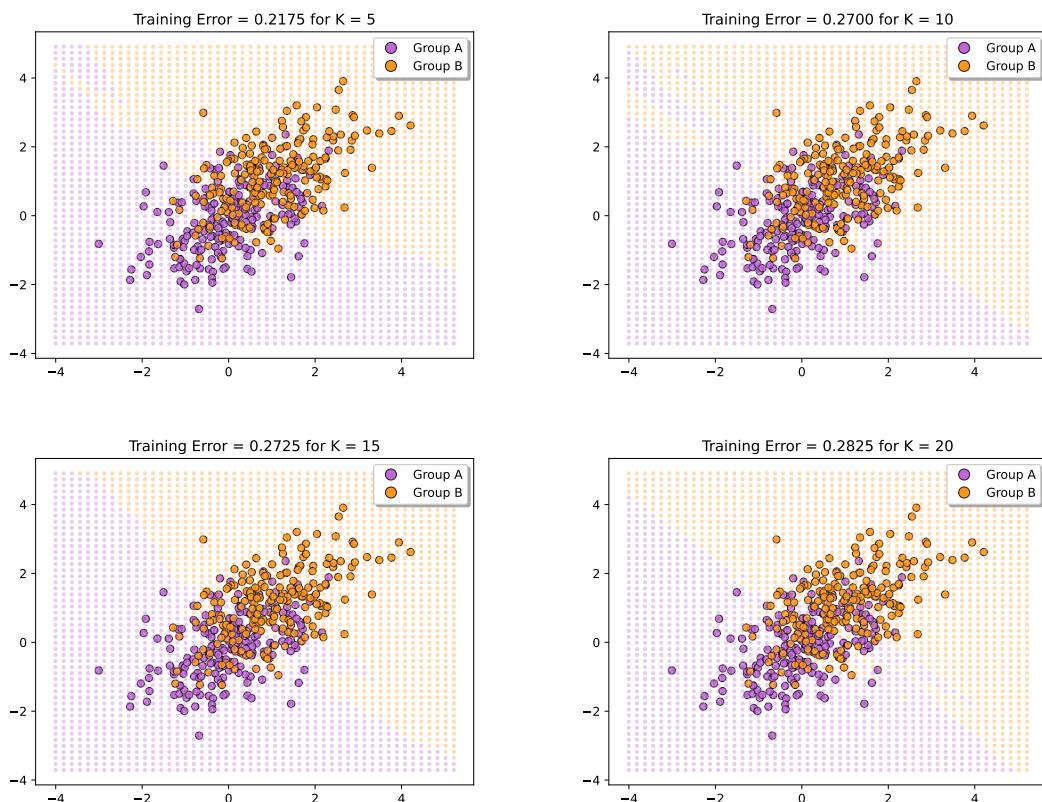


Figure 8: K-近鄰演算法的群組分界線: 模擬訓練 5

模擬訓練 6. (資料分散程度大、 X_1, X_2 具相關性且共變異矩陣不同)

DATA 6.

Group A :

$$n_1 = 200, \mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 2 & 1.5 \\ 1.5 & 2 \end{bmatrix}$$

Group B :

$$n_2 = 200, \mu_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 5 & 1.5 \\ 1.5 & 5 \end{bmatrix}$$

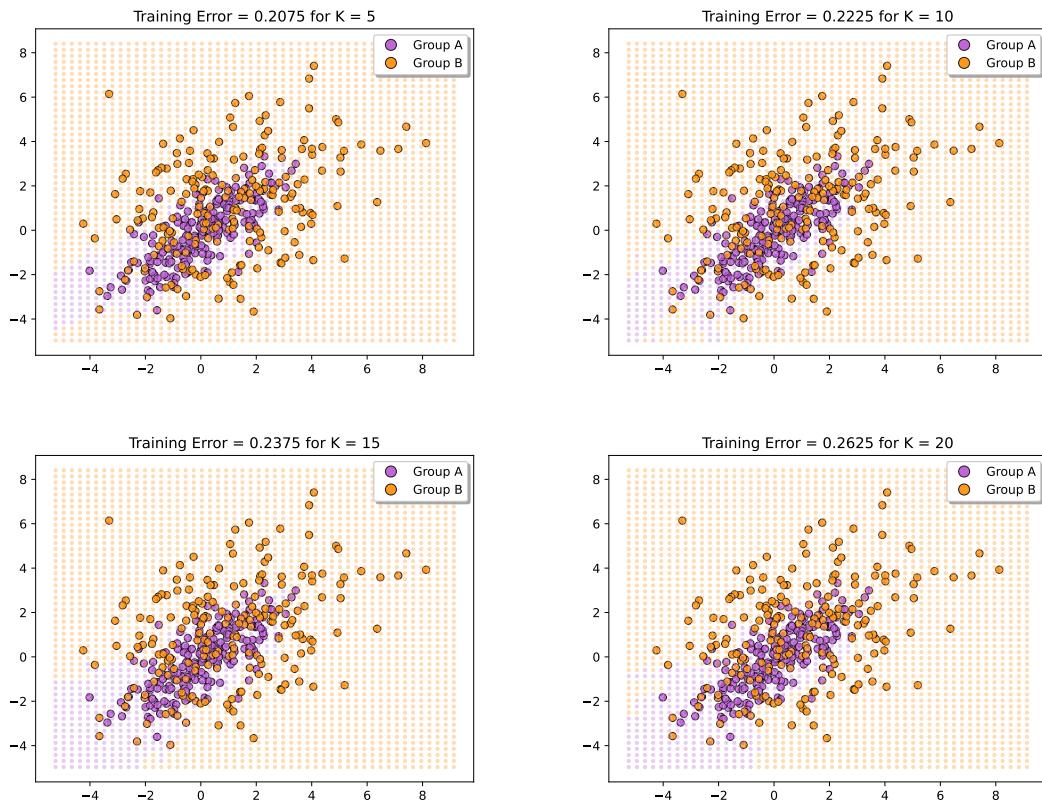


Figure 9: K-近鄰演算法的群組分界線: 模擬訓練 6

3.1 選用三群不同的資料展現三種學習器的學習情況

以下將針對三個群組的資料，利用不同的方法，對其進行檢視及群組的判別，共有兩種不同型態的資料。其中，令 Group A = 0、Group B = 1、Group C = 2。

DATA 7.

Group A :

$$n_1 = 300, \mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Group B :

$$n_2 = 200, \mu_2 = \begin{bmatrix} -4 \\ -2 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 2 & -0.7 \\ -0.7 & 2 \end{bmatrix}$$

Group C :

$$n_3 = 100, \mu_3 = \begin{bmatrix} -4 \\ 2 \end{bmatrix}, \Sigma_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

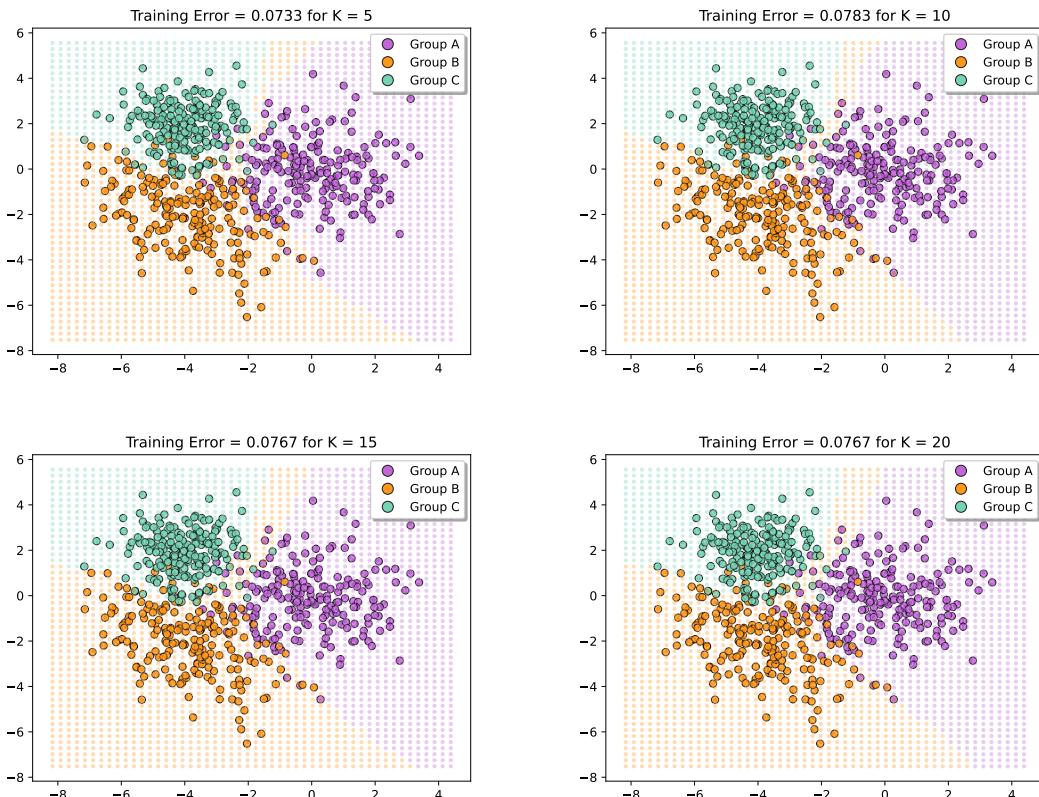


Figure 10: K-近鄰演算法的群組分界線: 模擬訓練 7

模擬訓練 2.(資料分散程度大且共變異矩陣不同)

DATA 8.

Group A :

$$n_1 = 300, \mu_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Group B :

$$n_2 = 200, \mu_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$$

Group C :

$$n_3 = 100, \mu_3 = \begin{bmatrix} 3 \\ 3 \end{bmatrix}, \Sigma_3 = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$$

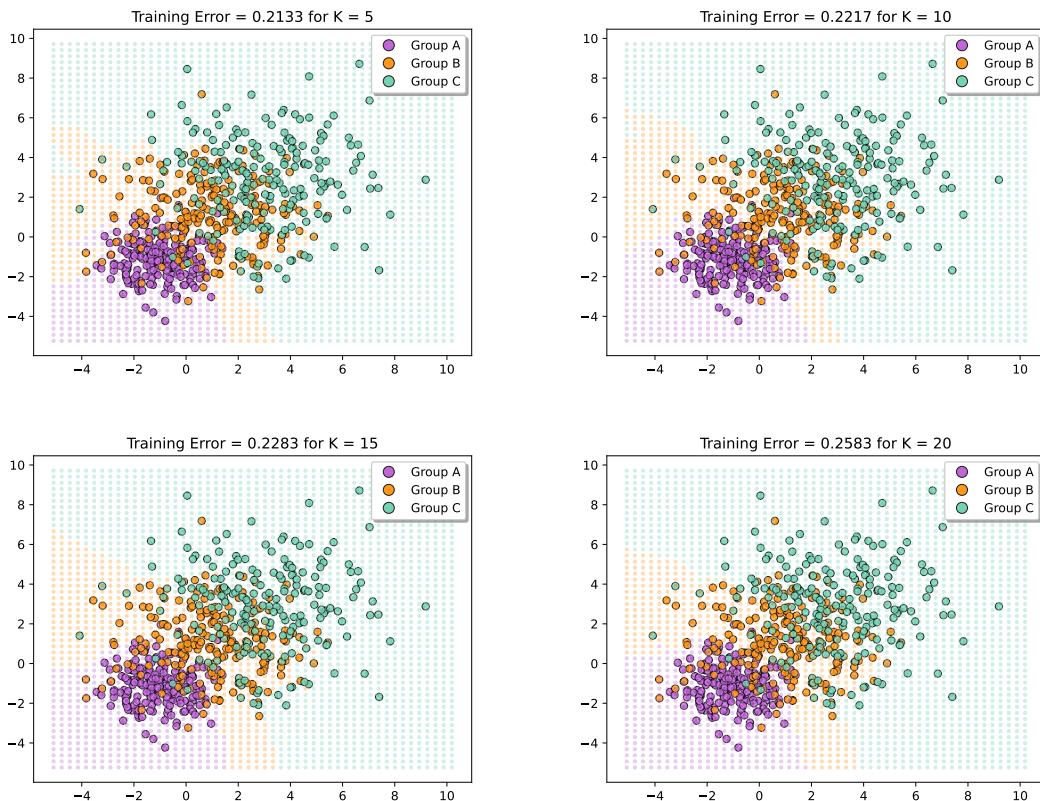


Figure 11: K-近鄰演算法的群組分界線: 模擬訓練 8