AI.FINTECH
AI金融科技中心

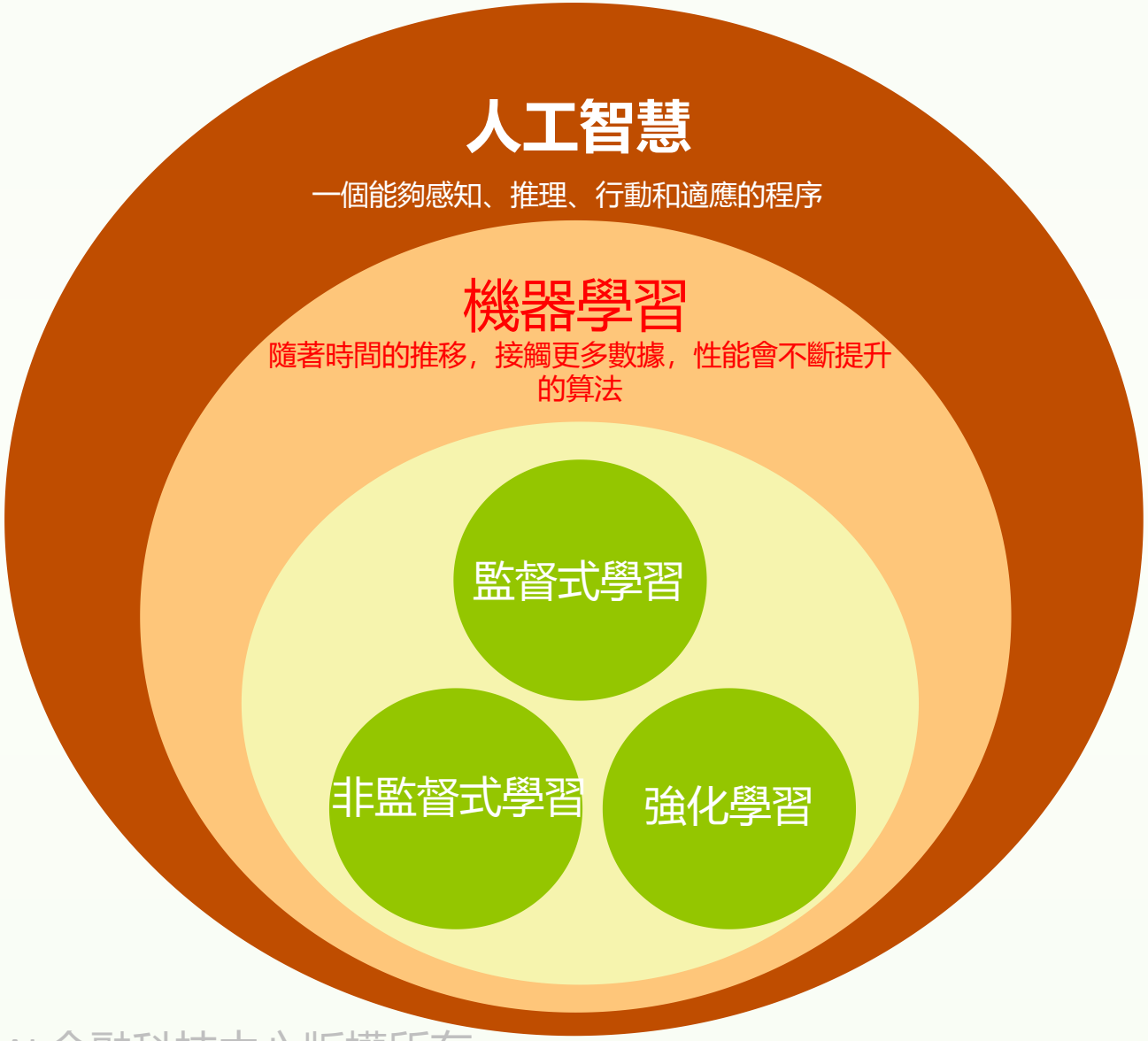
K-Nearest Neighbors Introduction

國立高雄科技大學 金融資訊系教授
AI金融科技中心主任
林萍珍教授

Outline

- **人工智慧與機器學習**
- 三種機器學習方法
- **什麼是K近鄰演算法**
- 歐式距離
- **K近鄰演算法**
- 如何選擇K值
- **K近鄰演算法的優點**
- K近鄰演算法的缺點

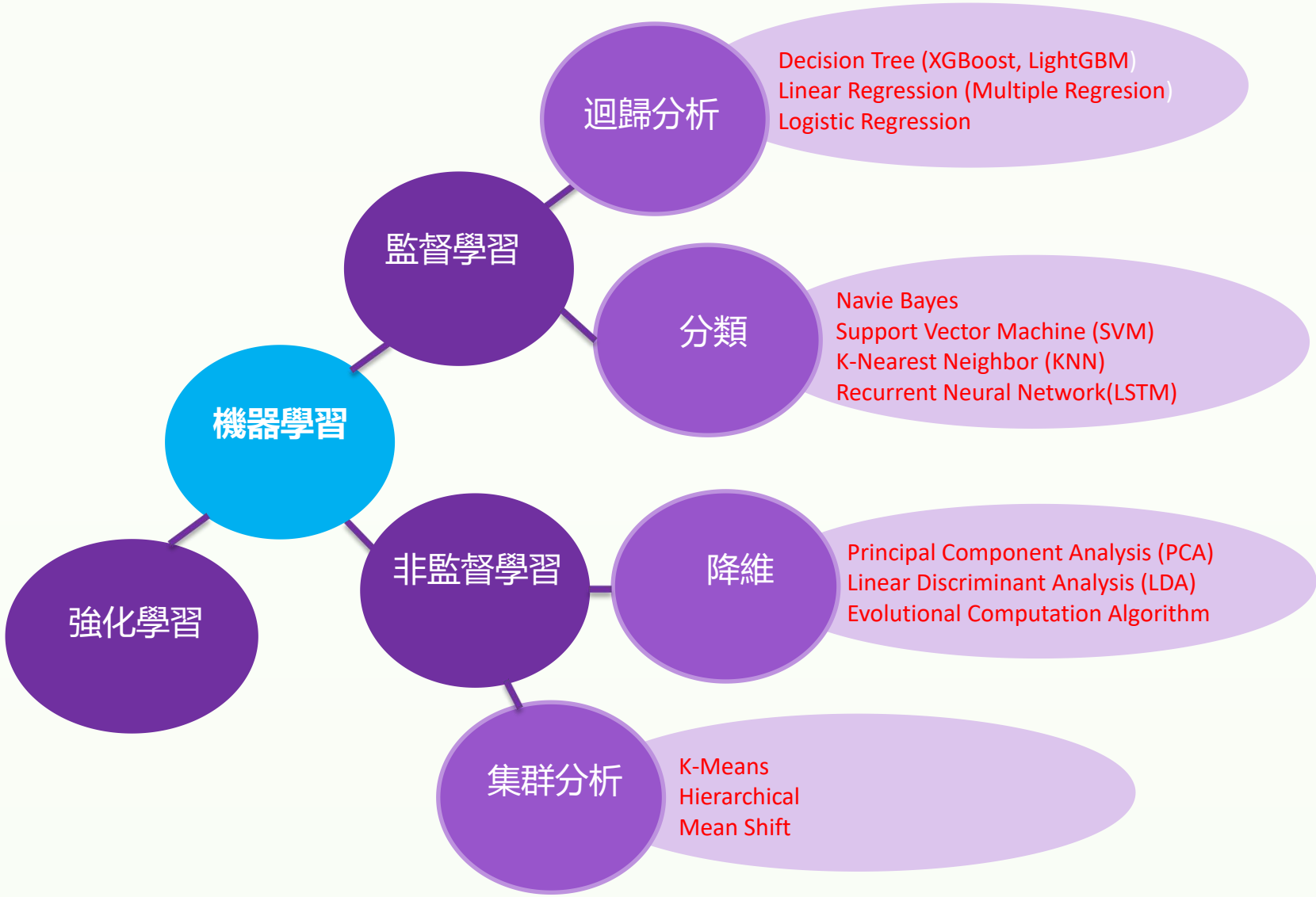
人工智慧與機器學習



三種機器學方法

- **監督式學習**
  - 在標籤好的數據集上訓練模型，每個訓練範例都有一個輸出標籤
  - 模型通過學習輸入特徵與輸出標籤之間的關係來進行預測。
  - 常見的技術包括迴歸和分類
- **非監督式學習**
  - 模型在沒有標籤的數據上進行訓練。目標是識別數據中的模式和關係
  - 常見的技術包括聚類分析和降維
- **強化學習**
  - 訓練模型通過對期望的行動進行獎勵，對不期望的行動進行懲罰來做出一系列決策。
  - 該模型通過反覆試錯來實現目標，並隨著時間的推移改進其策略。

機器學習可以分為三種類型



KNN Introduction

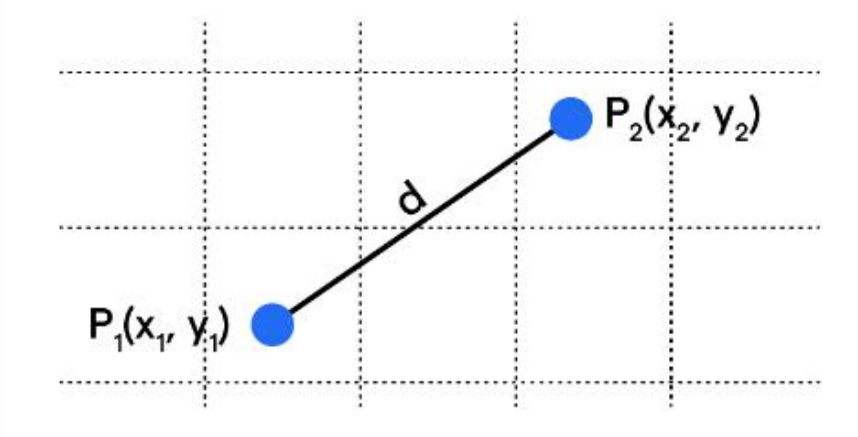
什麼是K近鄰演算法?

- **K近鄰演算法?**
  - KNN（k近鄰演算法）是一種廣泛使用的機器學習分類技術。
  - 利用鄰近性進行分類，將相似的數據點歸類在一起，因為相似的數據點往往具有相似的標籤或數值。
  - 使用歐式距離計算輸入數據點和所有訓練資料之間的距離。
  - 將K個鄰居中最常見(出現最多次)的標籤作為輸入數據點的預測標籤。
  - 性能會受到K值和距離計算的影響。

歐式距離

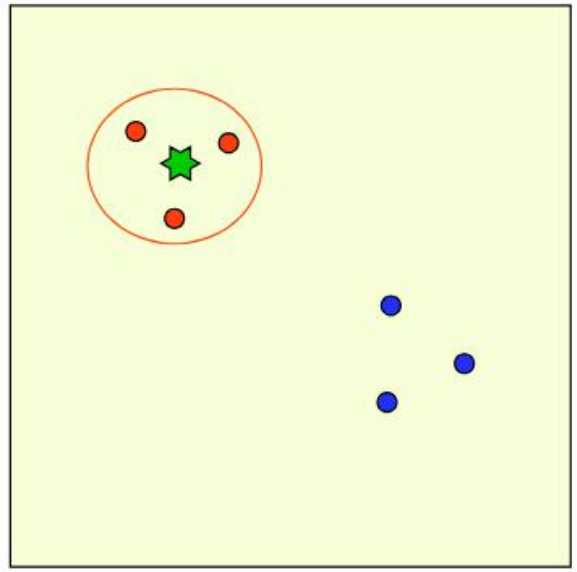
- 在平面/超平面內的兩個點之間
- 歐式距離可以視為連接這兩個點的直線的長度
- 計算兩個點(**Point1(x_1, y_1) 、 Point2(x_2, y_2))之間的直線距離公式:**

$$\text{Euclidean Distance } (d) = \sqrt{(x_2 - y_1)^2 + (y_2 - y_1)^2}$$



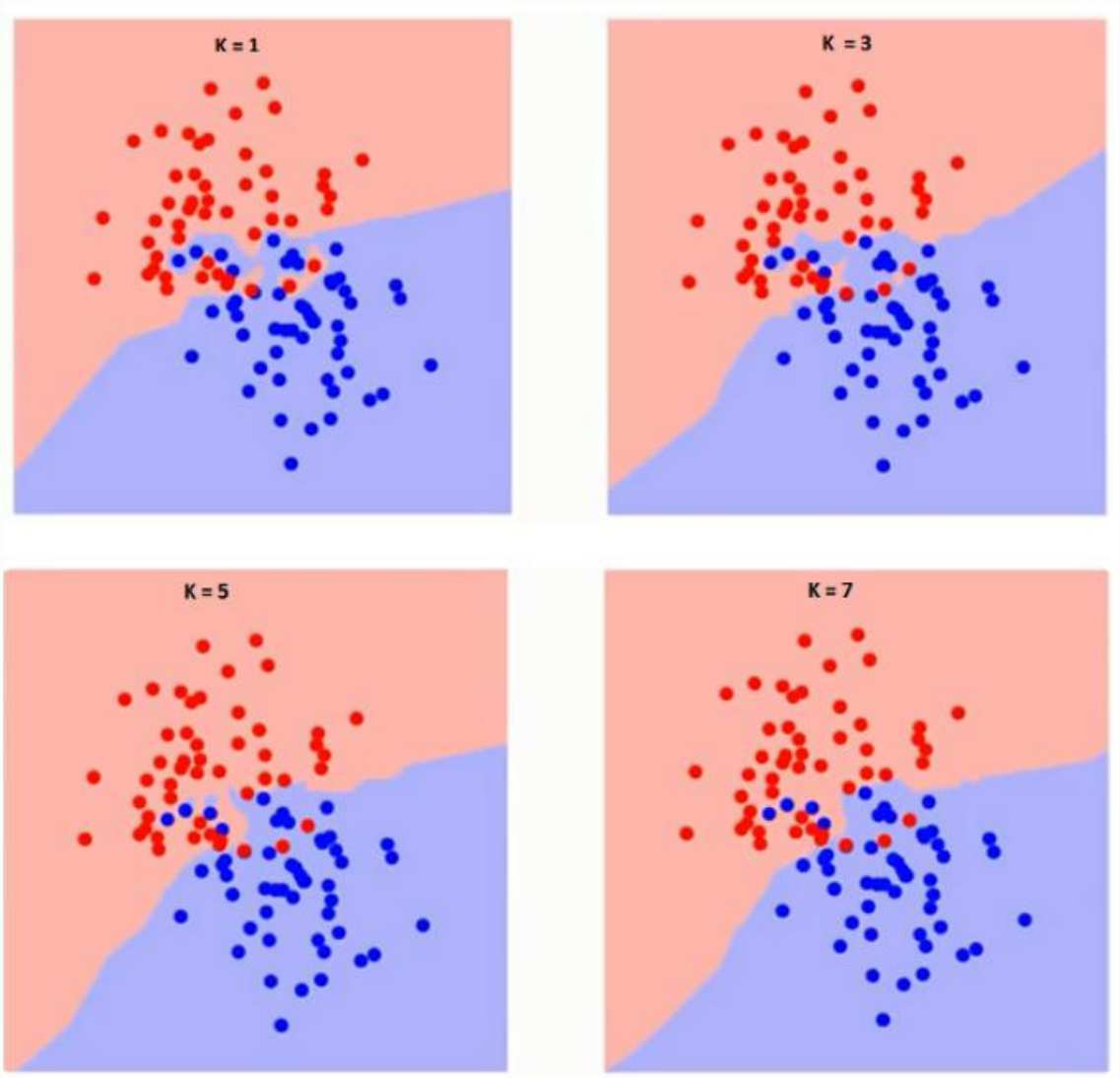
K近鄰演算法

- 保持6個訓練觀察點不變，給定的一個K值，可以為每個類別建立邊界。
- 決策邊界有效地分隔，例如，3個紅色圓圈和3個藍色圓圈。
- KNN算法中的“K”代表我們希望取出票數的最近鄰居。
- 我們打算找出綠色星星（GS）的類別 。GS可以是紅色圓圈（RS）或藍色圓圈（BS）類別。
- KNN算法中的“K”是我們希望取票的最近鄰居數。假設K = 3。
- 以GS為中心畫一個圓，使其僅包含平面上的三個數據點。
- 離GS最近的三個點都是RS。因此，我們可以有較高的可信度說GS應該屬於RS類別。
- 這個選擇變得明顯，因為最近鄰居的三票都給了RS
- 在這個算法中，參數K的選擇非常關鍵。



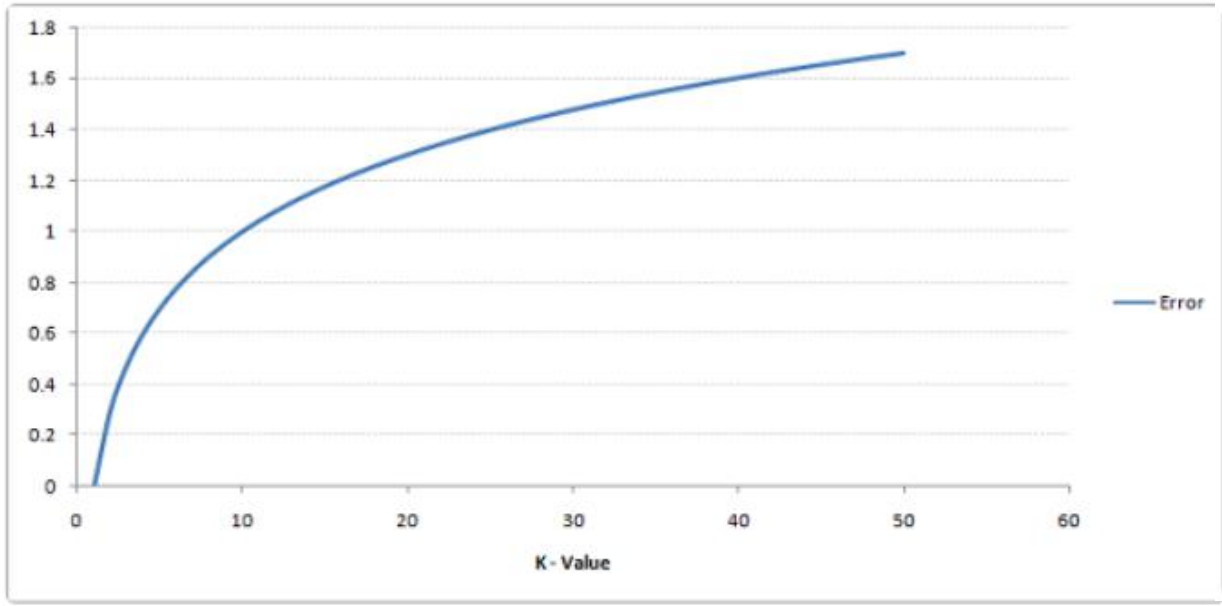
如何選擇K值?

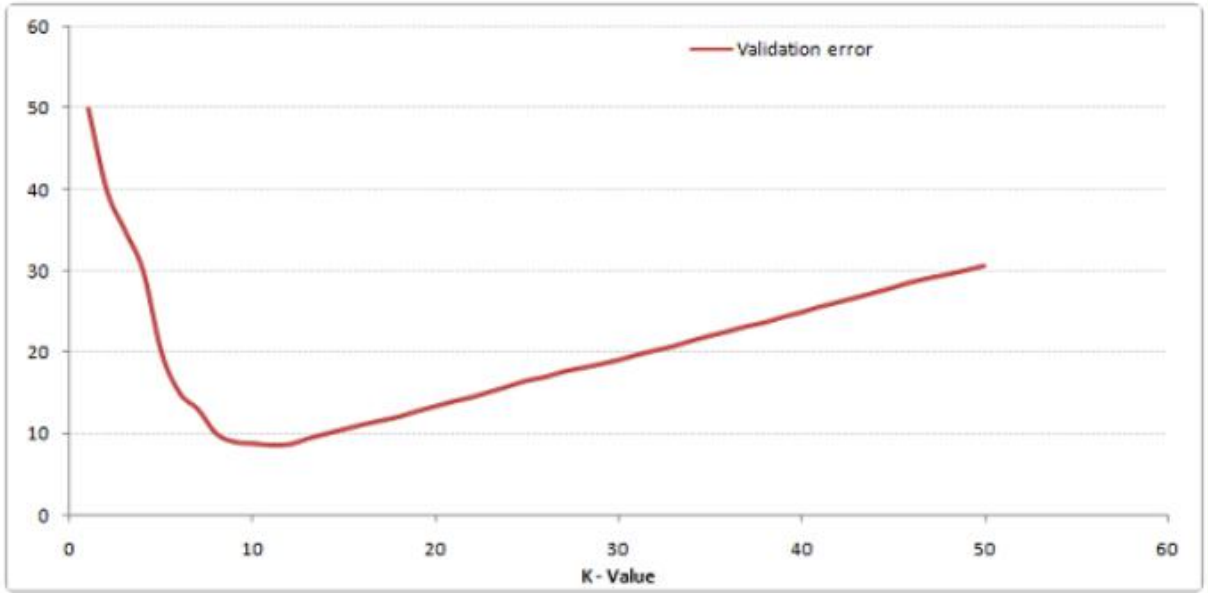
- 以下說明了對應於不同K值的兩類邊界區分。
- 隨著K值的增加，邊界變得更加平滑。
- 當K增加到無限大時，最終會根據總體多數變為全部藍色或全部紅色。



如何選擇K值?

- 訓練錯誤率和驗證錯誤率是我們需要評估不同K值的兩個參數。
- 左圖（藍線）顯示了隨著K值變化的訓練錯誤率曲線。
- 當K=1時，訓練樣本的錯誤率始終為零，因為對於任何訓練數據點，最近的點是其自身。因此，當K=1時，預測總是準確的。
- 右圖（紅線）顯示了隨著K值變化的驗證錯誤率曲線。





K近鄰演算法的優點:

- 簡單
  - 由於算法的複雜性不高，因此實現起來相對簡單。
- 容易適應
  - 每當添加一個新的範例或數據點時，算法會進行調整，並對未來的預測做出貢獻。
- 少量的超參數
  - 訓練KNN算法所需的唯一參數是K值和我們希望選擇的距離度量。

K近鄰演算法的缺點:

- 無法拓展
  - KNN算法是一種惰性算法。它在計算資源和數據存儲方面需求量大，導致算法既耗時又耗資源。
- 維度災難
  - 根據KNN算法的峰值現象，該算法會受到維度災難的影響，這意味著當維度過高時，算法難以正確分類數據點。
- 容易過擬合
  - 由於受到維度災難的影響，KNN算法容易出現過擬合的問題。
  - 因此，通常會應用特徵選擇和降維技術來應對這個問題。

Reference

- https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/
- https://botpenguin.com/glossary/euclidean-distance
- https://www.geeksforgeeks.org/k-nearest-neighbours/

Thank you.