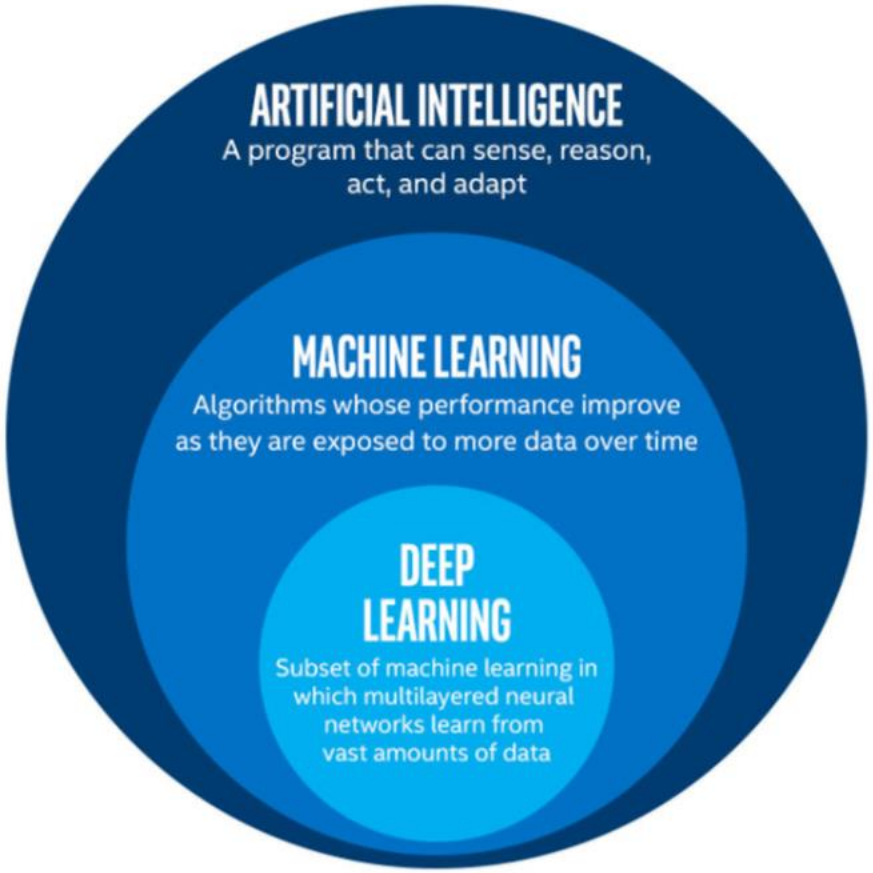
機器學習實作教育訓練

Agenda

- 機器學習基本概念與實作流程
- 預測方法簡介

AI 機器學習 深度學習

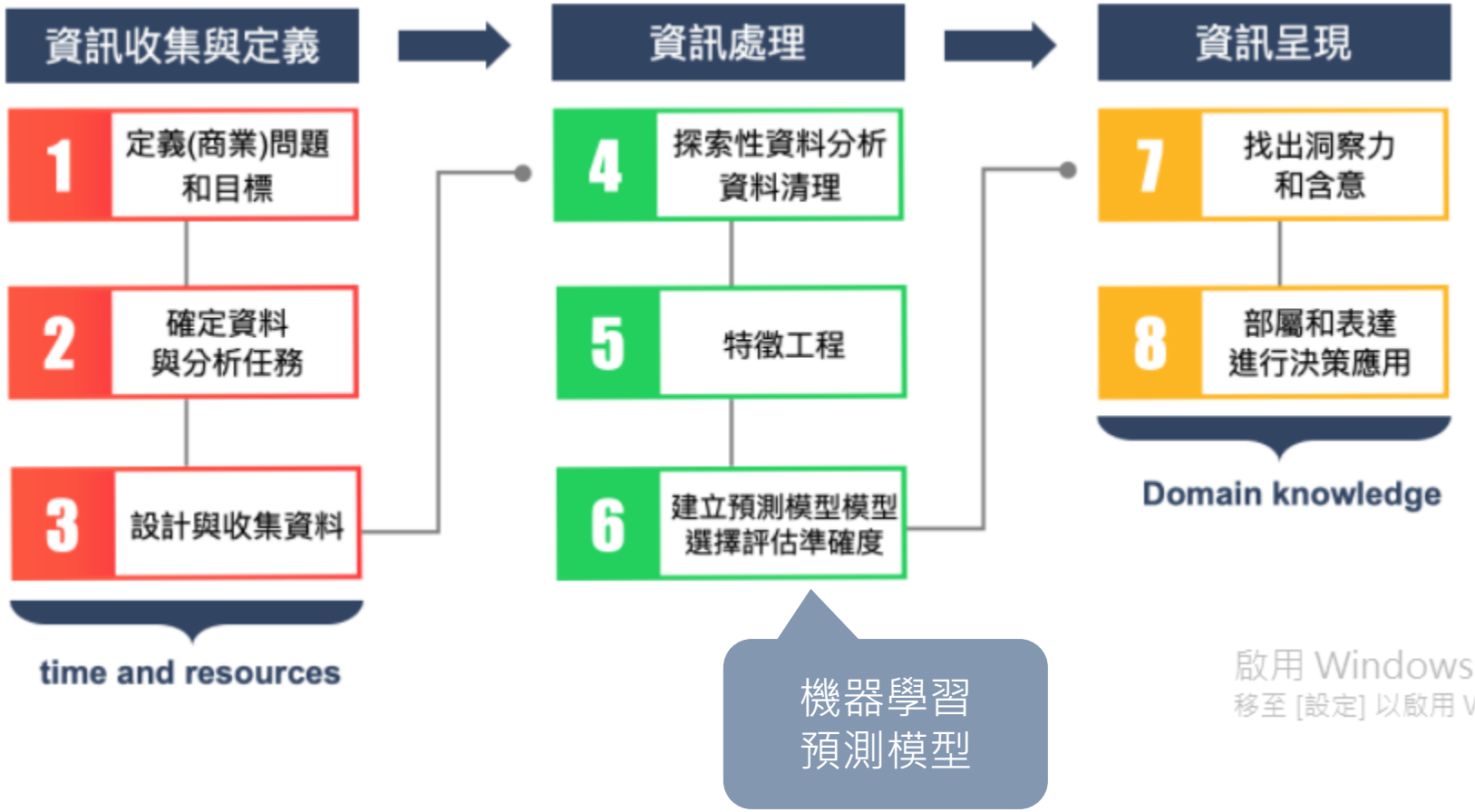


Interception from Prowesscorp website.

AI: 計算機模仿人類思考進而模擬人類的能力/行為。
ML: 從資料中學習模型。
DL: 利用多層的非線性學習資料表徵。

機器學習實作流程

“數據和特徵決定了機器學習的上限，
而模型和算法只是逼近這個上限而已”



何謂機器學習

- 「透過從過往的資料和經驗中學習並找到其運行規則，最後達到人工智慧的方法。」

- 機器學習是關於如何預測未來。它透過以下的方式去進行訓練：
  - 它需要資料(去訓練系統)
  - 從資料中學習樣本
  - 根據步驟2所獲得的經驗，替未曾見過的新資料做分類，並推測它可能是什麼

機器如何學習

- **監督式學習（Supervised Learning）**
  - 在訓練的過程中提供物件（向量）和預期輸出，可以是「有標籤」的分類資料或是一個連續的值（迴歸分析），例如輸入了大量已標示清楚標籤的腳踏車和機車給機器後，讓機器分辨尚無標簽的照片是機車還是腳踏車。類似於動物和人類的認知感知中的「概念學習」（concept learning）。
- **半監督式學習（Semi-supervised learning）**
  - 介於監督學習與非監督學習之間。這樣的學習方式會先將「有標籤」的資料和「無標籤」的資料切出一條分界線，再將「無標籤」資料依據整體分布，調整出兩大類別的新分界。不需要百分之百大量的「有標籤」資料，讓半監督學習同時能降低成本又具有非監督式學習高自動化的優點。
- **非監督式學習（Unsupervised Learning）**
  - 這樣的機器學習方式不需要人力事前的輸入標籤，僅僅提供了輸入範例，便直接以沒有標準答案的資料來訓練機器，在學習時機器會自動找出潛在類別的規則，並且反覆以經過測試後的學習結果應用到新的案例上。
- **增強學習（reinforcement learning）**
  - 源自於心理中行為主義理論的學習方法，即如何在環境給予的獎懲刺激下，一步步形成對於這些刺激的預期，來產生能夠獲得最大利益的習慣性行為，強調的是透過環境而行動，並會隨時根據輸入的資料逐步修正。這個方法具有普適性，因此在其他許多領域，如博弈論、統計學及遺傳算法等都有研究

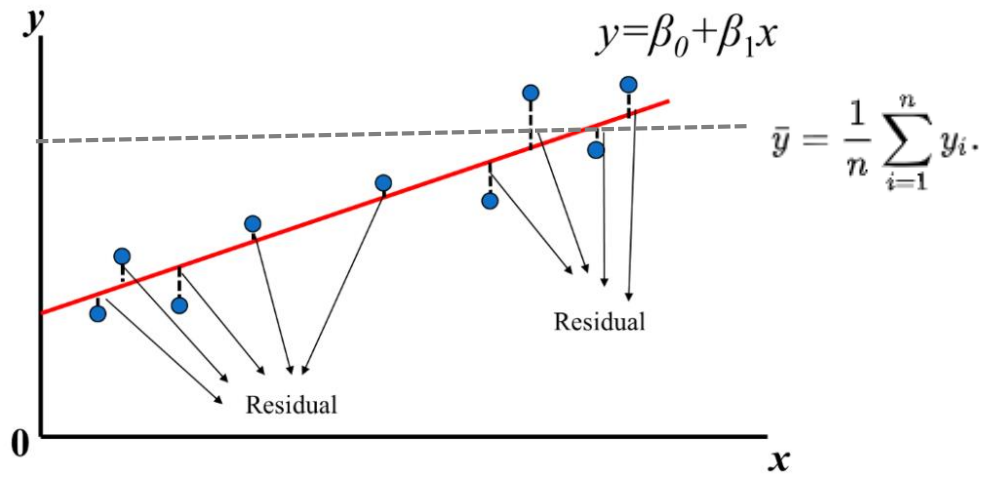
機器學習簡介

- 監督式學習
  - 迴歸分析(Regression)
  - 邏輯迴歸分類(Logistic Regression Classification)
  - 決策樹(Decision Tree)
  - 支援向量機(Support Vector Machine)
  - 最近鄰居分類(K-nearest Neighbors)
- 非監督式學習
  - 分群(Clustering)
  - 異常偵測(Anomaly Detection)
  - 主成分分析(Principal-Component Analysis)
- 整體學習(Ensemble Learning)
- 神經網路(Neural Network)
- 深度學習(Deep Learning)
- 特徵工程(Feature Engineering)

線性回歸

- 線性回歸（Linear regression）是統計上在找多個自變數(independent variable)和依變數(dependent variable)之間的關係建出來的模型。只有一個自變數和一個依變數的情形稱為簡單線性回歸(Simple linear regression)，大於一個自變數的情形稱為多元回歸(multiple regression)。

$$\varepsilon i = yi - \hat{y}i$$



1.最小平方法(Least Square)估計參數
2.判定預測好與壞:判定係數R^2

過度配適 Overfitting

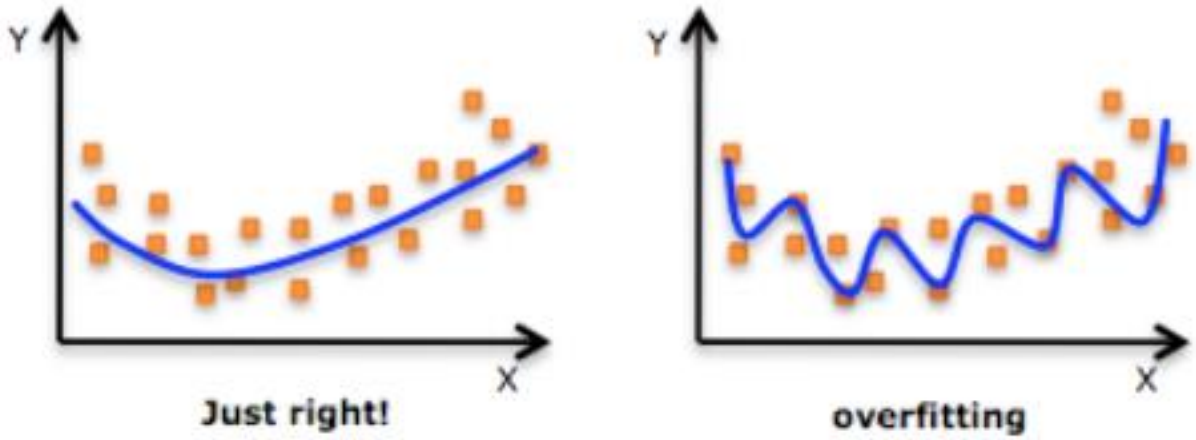
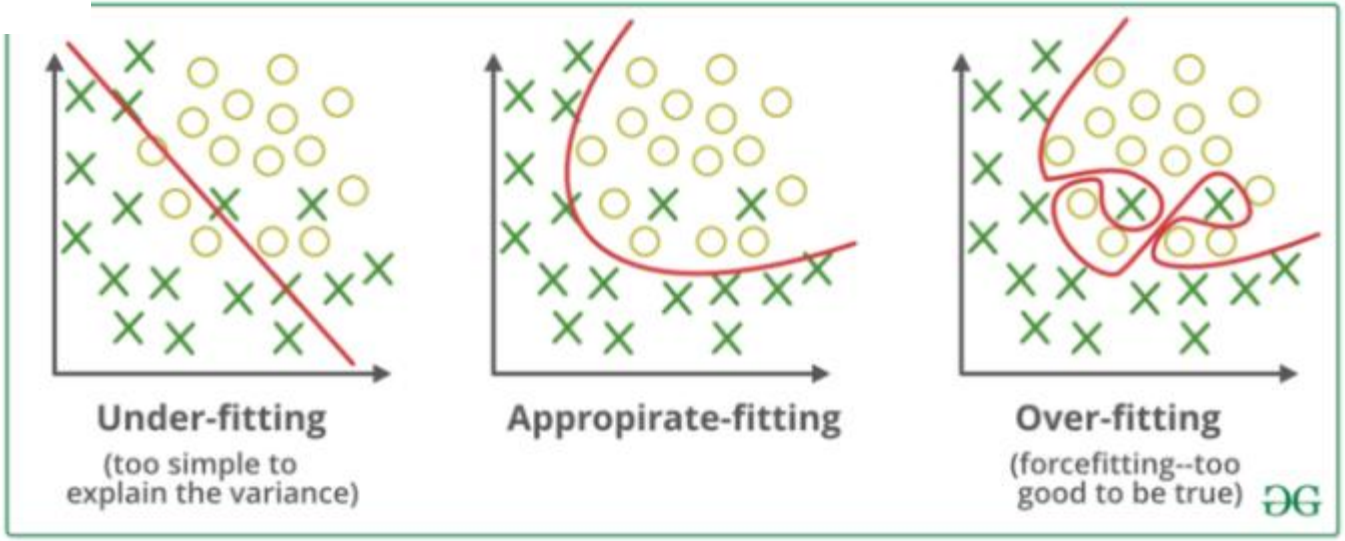


圖1 擬合過度(OverFitting)

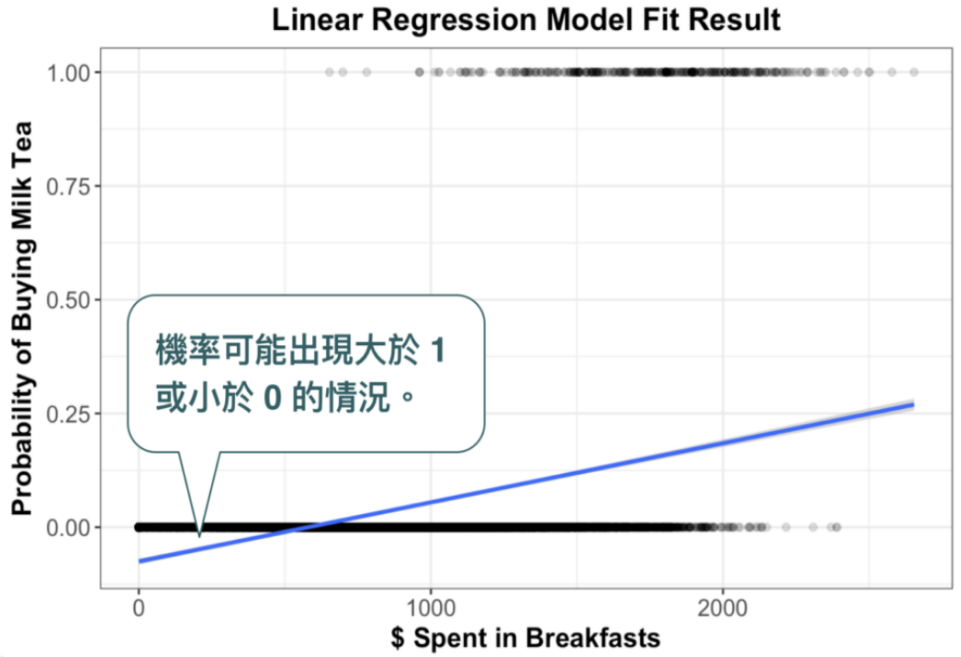
如何避免
1.增加資料量
2.漸少特徵數

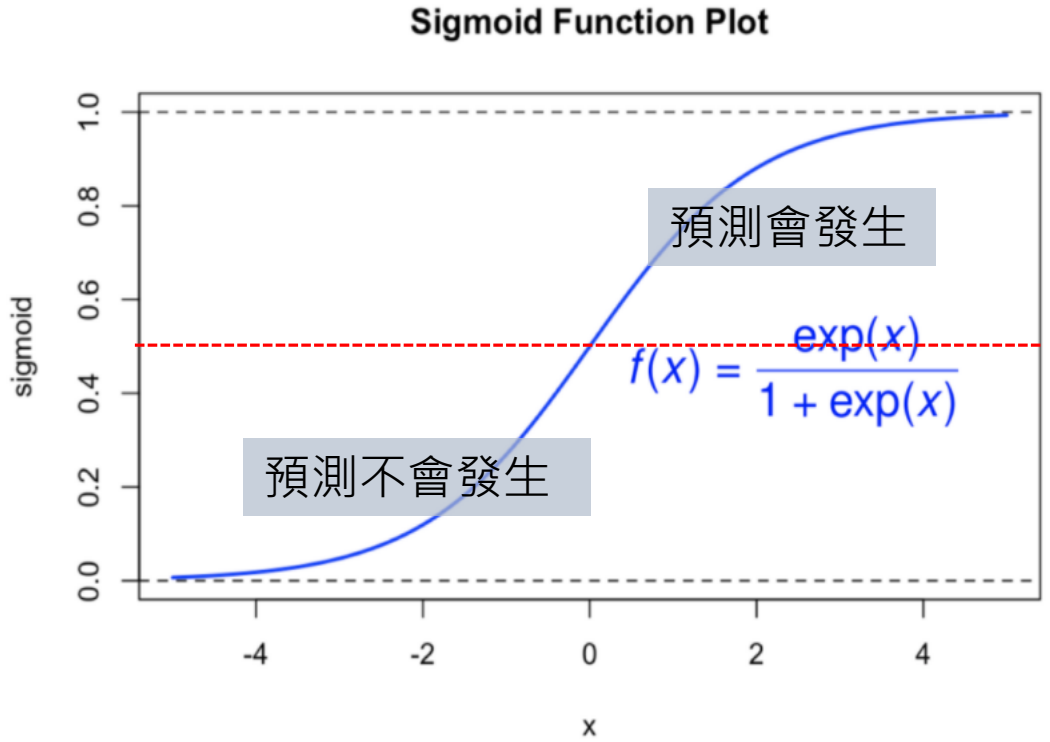


欠擬合、良好、過擬合模型(由左至右)[Source]

logistic回歸

- 邏輯迴歸通常是在學分類問題 (classification) 第一個會接觸到的模型，用來建立「二元目標變數」(Binary Output Variable) 跟解釋變數之間的關係，模型形式如下
- Sigmoid函數是落在 (0,1) 之間，符合邏輯迴歸針對機率建模的特性





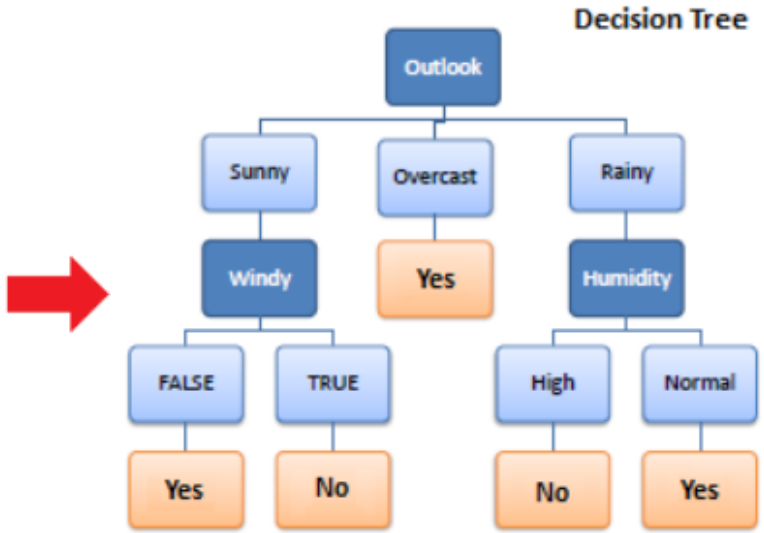
決策樹

- 決策樹是一種解決分類問題的算法，想要了解分類問題和迴歸問題，採用樹形結構，使用層層推理來實現最終的分類。
- 決策樹由下面幾種元素構成：

根節點：包含樣本的全集
內部節點：對應特徵屬性測試
葉節點：代表決策的結果

Predictors　　Target

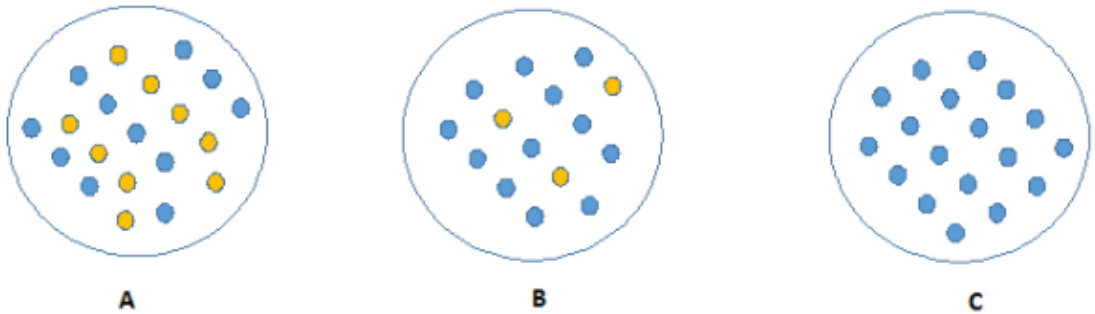
Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No



常見的決策樹演算法比較

演算法	資料屬性	分割規則
ID3	離散型	Entropy, Gain Ratio
C4.5	離散型	Gain Ratio
CHAID	離散型	Chi-Square Test
CART	離散與連續型	Gini Index

我們用下面的例子來說明「熵」：如下這三筆dataset，何者只需要最少的資訊便可清楚的說明呢？



常見的資訊量有兩種：熵(Entropy) 以及 Gini不純度(Gini Impurity)

$$I_H(t) = -\sum_{i=1}^{c} p(i|t) \log_2 p(i|t)$$

熵資訊量函式

資訊增益

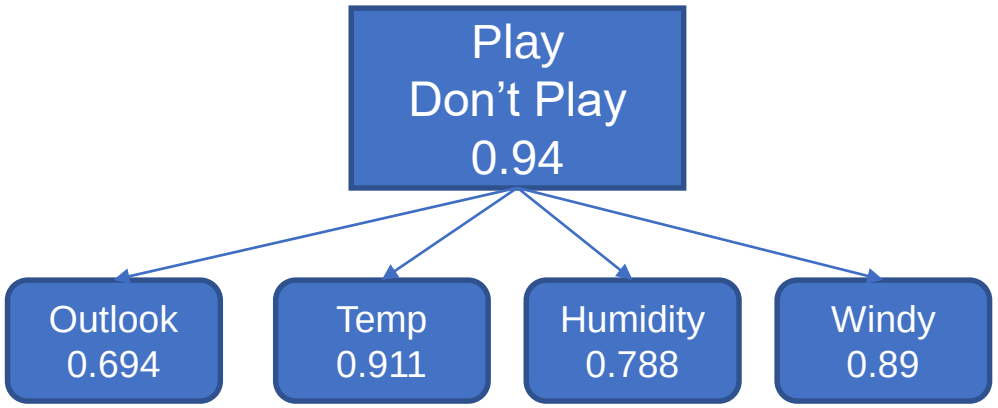
獲得的資訊量　原本的資訊量　經由分割後的資訊量

$$IG(D_p, f) = I(D_p) - \sum_{j=1}^{m} \frac{N_j}{N_p} I(D_j)$$

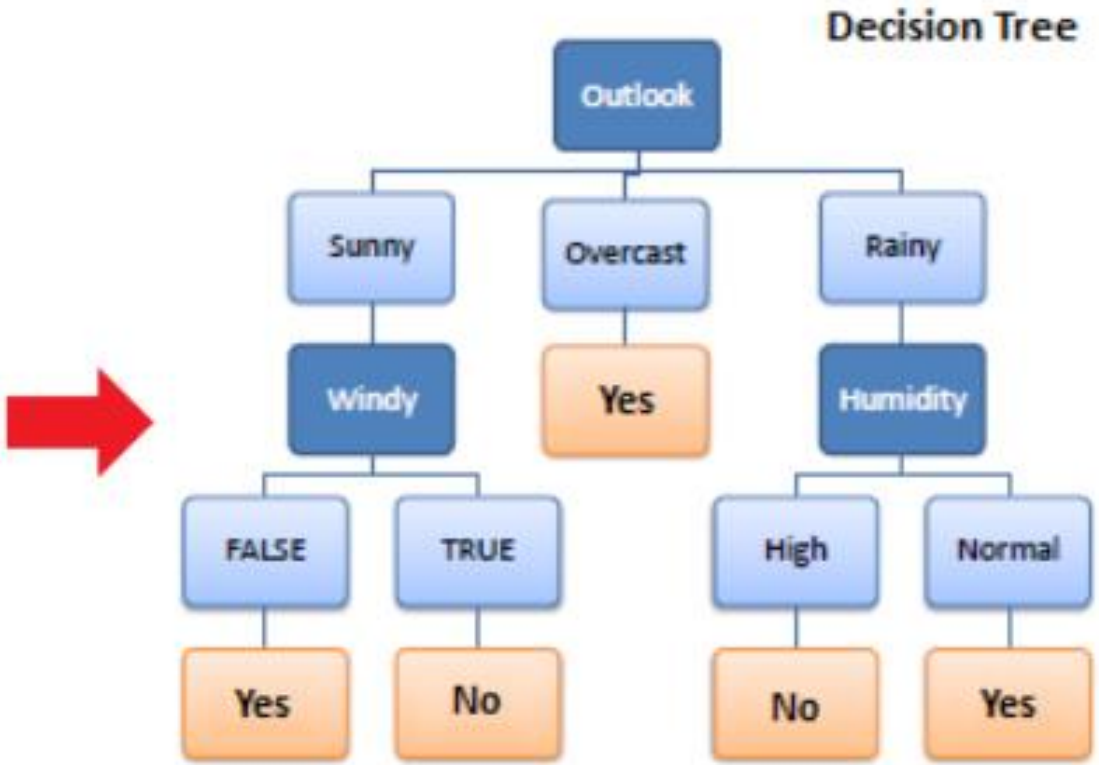
$$I_G(t) = \sum_{i=1}^{c} p(i|t)(1 - p(i|t)) = 1 - \sum_{i=1}^{c} p(i|t)^2$$

Gini Impurity資訊量函式

entropy

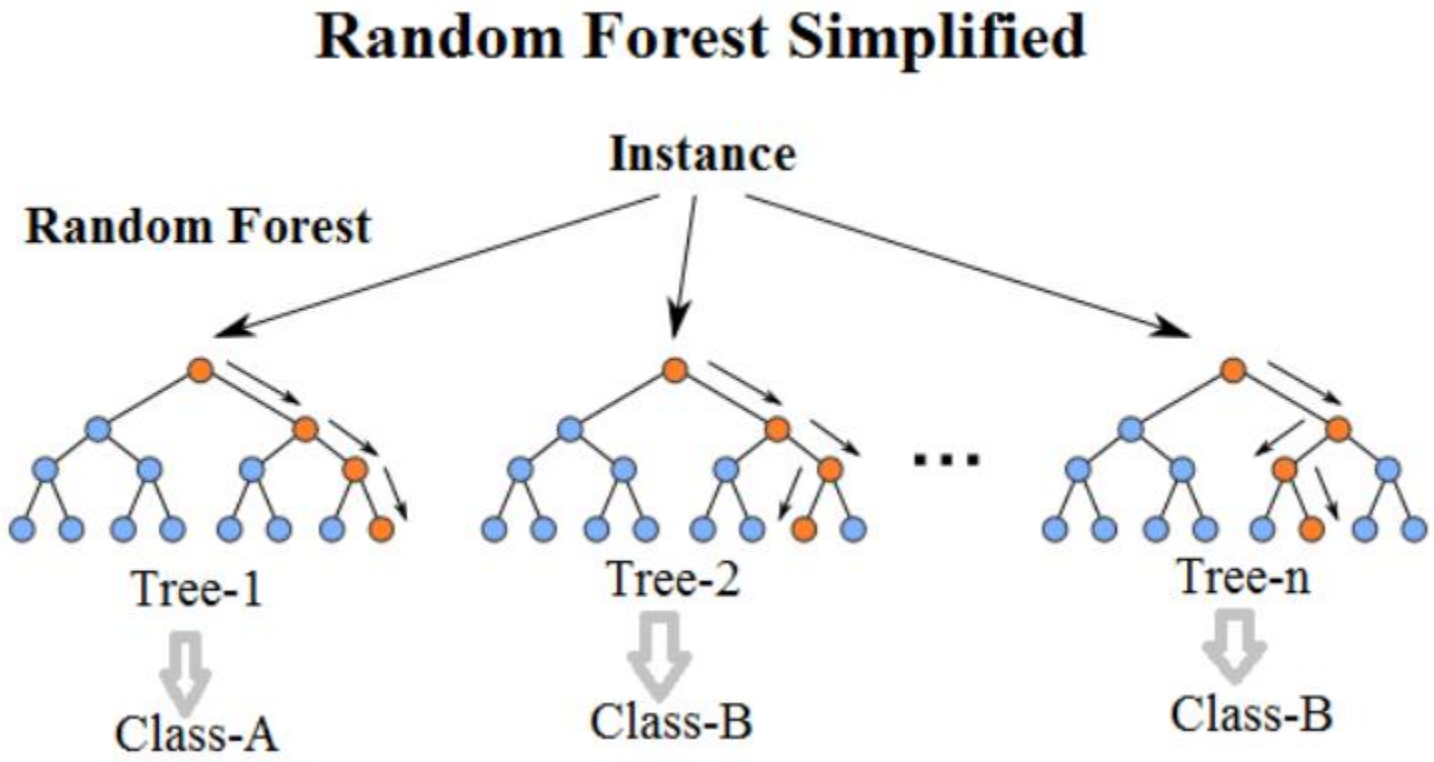


0.248 0.029 0.152 0.04



隨機森林

- Random Forest的基本原理是，結合多顆決策樹，並加入隨機分配的訓練資料，以大幅增進最終的運算結果。顧名思義就是由許多不同的決策樹所組成的一個學習器，其想法就是結合多個「弱學習器」來建構一個更強的模型：「強學習器」。這種方法又稱為Ensemble Method，也就是「三個臭皮匠勝過一個諸葛亮」的概念。



隨機森林

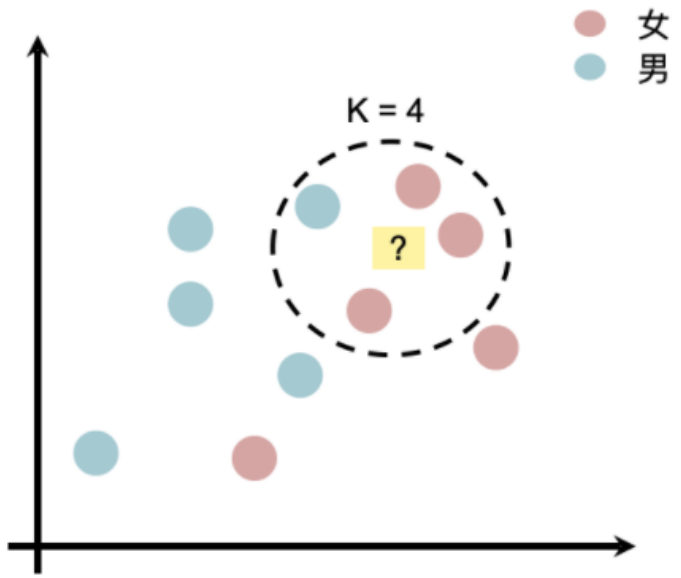
產生多棵樹做法

- 不過我們只有一個數據集，所以要形成多顆具差異性的樹以進行Ensemble Method，就是要產生不同的數據集，才能產生多顆具差異性的樹，其作法有兩種方式：
- 1.Bagging(Bootstrap Aggregation):
  - 指的是「重新取樣原有Data產生新的Data，取樣的過程是均勻且可以重複取樣的」，使用Bootstrap我們就可以從一組Data中生出多組Dataset。
  - 此種方法會從Training dataset中取出K個樣本，再從這K個樣本訓練出K個分類器（在此為tree）。每次取出的K個樣本皆會再放回母體，因此這個K個樣本之間會有部份資料重複，不過由於每顆樹的樣本還是不同，因此訓練出的分類器（樹）之間是具有差異性的，而每個分類器的權重一致最後用投票方式(Majority vote)得到最終結果。
- 2. Boosting:
  - 與Bagging類似，但更強調對錯誤部份加強學習以提升整體的效率。是透過將舊分類器的錯誤資料權重提高，加重對錯誤部分的練習，訓練出新的分類器，這樣新的分類器就會學習到錯誤分類資料(misclassified data)的特性，進而提升分類結果。就好像在學校考試時，面臨大型考試前都會加強比較弱的部分或是把之前考試時錯誤的題目再多練習幾次。

KNN

什麼是 KNN？

- K-Nearest Neighbor(KNN) 是一種無須機率分配的假設下的演算法，跟距離預測值最近的 k 個數值，來估計預測值。
- 以下圖為例， 要預測這個人是男生或女生，以這個例子來看，透過 KNN (k=4) 的預測值為女生

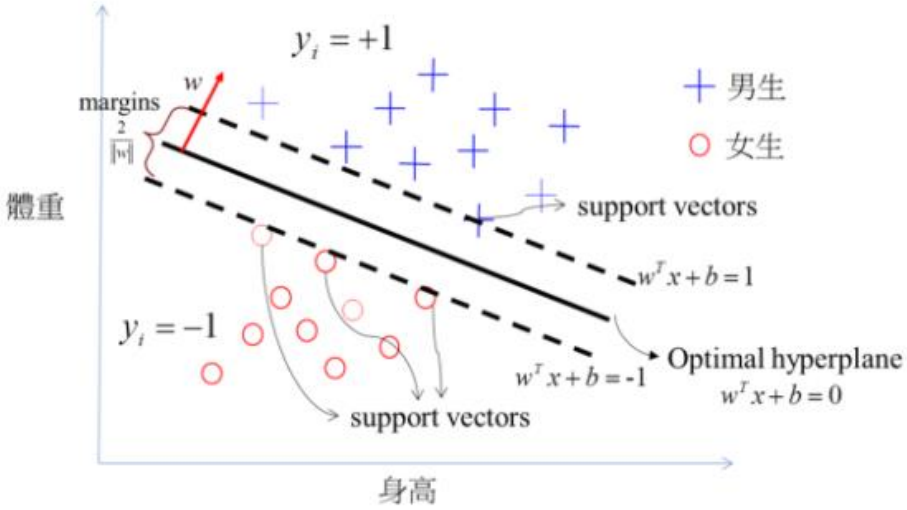


- 由於類別型資料，透過投票來決定預測值，因此 k 建議以奇數為主，避免掉平手的問題。

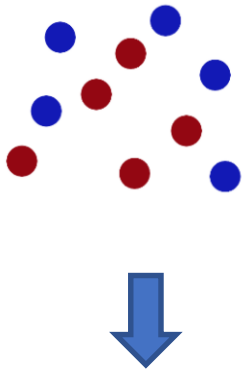
KNN 的三個步驟：
1.計算距離
2.尋找最近 k 組數值
3.類別型態資料以多數決
　數值型態用 k 組資料的統計值 (如平均)

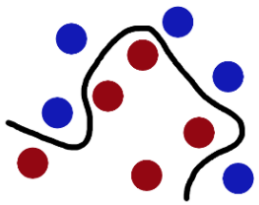
SVM

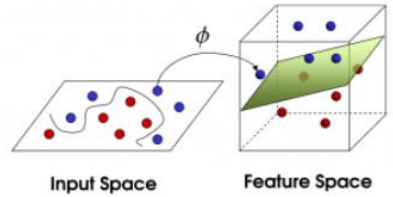
- SVM是一種監督式的學習方法，用統計風險最小化的原則來估計一個分類的超平面(hyperplane)，其基礎的概念非常簡單，就是找到一個決策邊界(decision boundary)讓兩類之間的邊界(margins)最大化，使其可以完美區隔開來



「如何只用身高體重就來判斷是男生還是女生」。
e.g. 分類男生和女生兩類，特徵資料只有「身高」和「體重」。



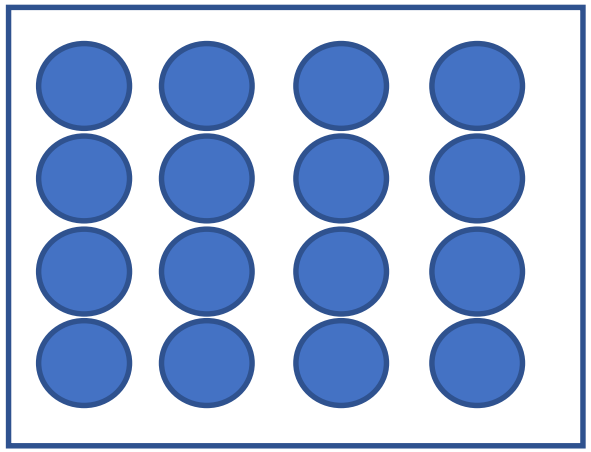




現在，從魔鬼的角度看這些球，這些球好像是被一條曲線分開了。

在之後，無聊的大人們，把這些球叫做 「data」，把棍子 叫做 「classifier」, 最大間隙trick 叫做「optimization」， 拍桌子叫做「kernelling」, 那張紙叫做「hyperplane」。

資料集切分



X ,y

training

testing

交叉驗證 (Cross-validation)

CV=3

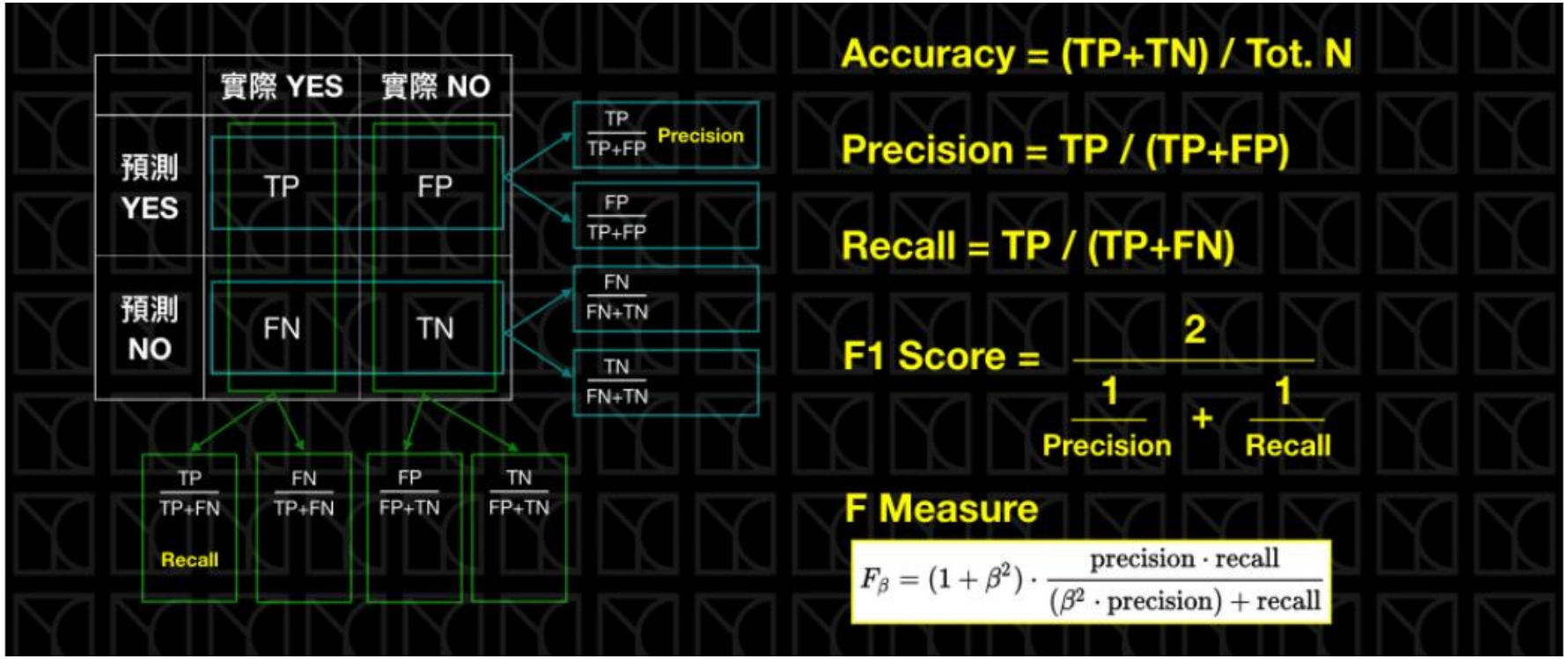
T	T	V
T	V	T
V	T	T

使用測試集確認模型效果

依據驗證集結果調整模型參數

混淆矩陣

- 在機器學習中，最常見的就是分類模型，混淆矩陣(confusion matrix)可去判定一個分類模型表現的到底好不好
- 在統計學上FP被還被稱為第一型錯誤(Type 1 Error)，FN被稱為第二型錯誤(Type 2 Error)



K-means

- 它是依靠距離的聚類算法，簡單來說就是距離越進的代表相似性越高就會視作同一群
- **K-means運作概念步驟:**
  - 1. 我們先設定好要分成多少(k)群。
  - 2. 然後在feature space(x軸身高和y軸體重組出來的2維空間，假設資料是d維，則會組出d維空間)隨機給k個群心。
  - 3. 每個資料都會所有k個群心算歐式距離(歐基李德距離Euclidean distance，其實就是直線距離公式，從小學到大的那個距離公式，這邊距離當然也可以換成別種距離公式，但基本上都還是以歐式距離為主)。
  - 4. 將每筆資料分類判給距離最近的那個群心。
  - 5. 每個群心內都會有被分類過來的資料，用這些資料更新一次新的群心。
  - 6. 一直重複3–5，直到所有群心不在有太大的變動(收斂) 。