機器學習
Decision Tree

授課老師：林彥廷

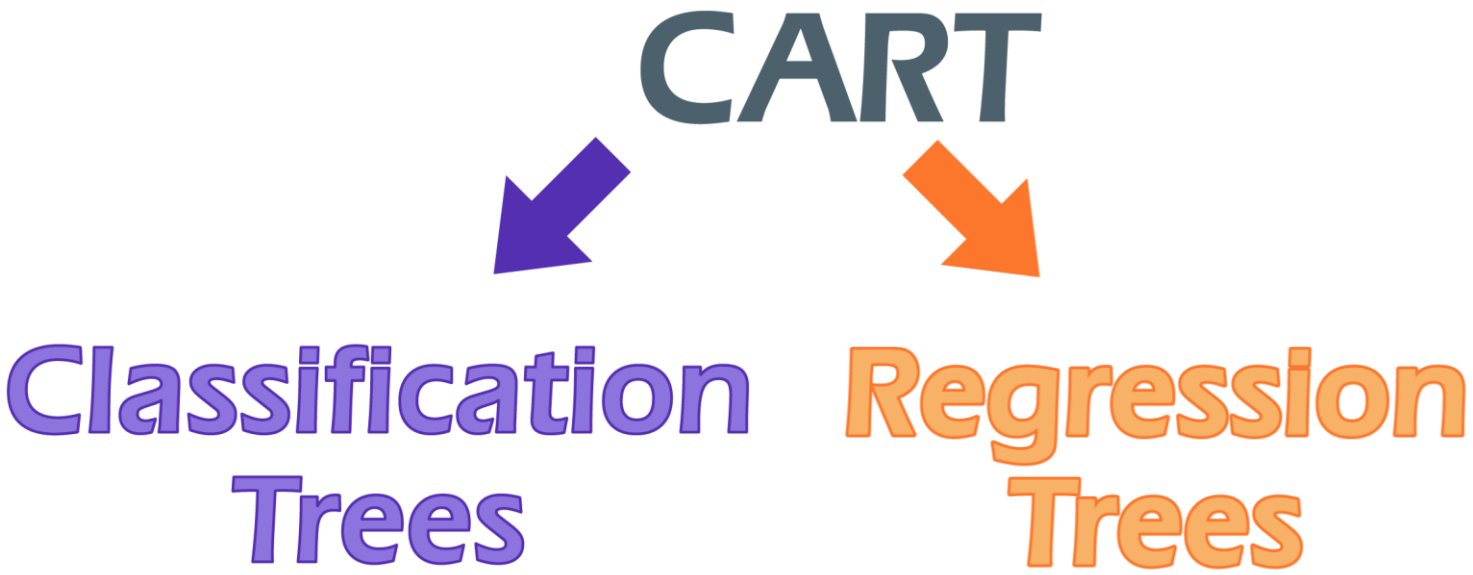
決策樹（Decision Tree）

- 決策樹(Decision Tree) 主要是使用樹狀分枝的概念來作為決策模式，是一種強大且廣受歡迎的分析方法。

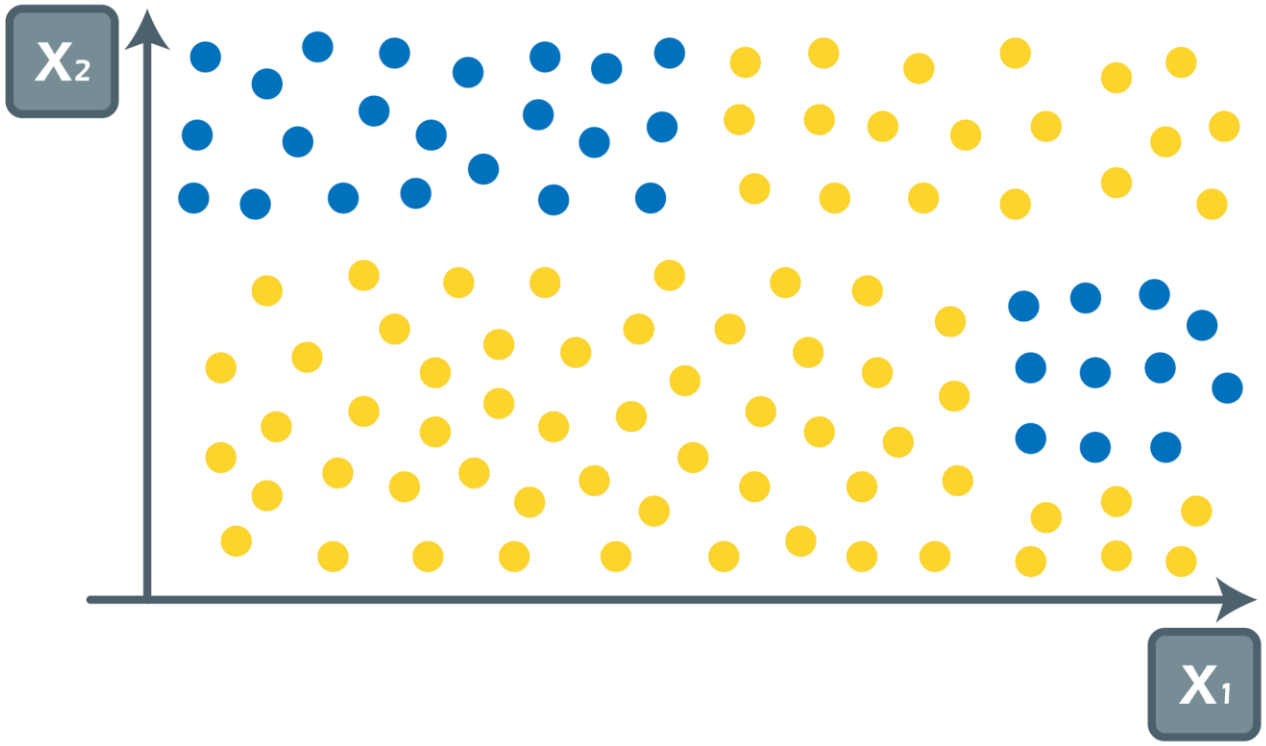
- 大多數的決策樹可以運用在分類預測上。當其用來預測的應變數類別型態(例如：生或死、男或女)時，該決策樹便稱為分類樹(Classification Tree)。有些決策樹演算法也可以像迴歸分析一樣，預測的結果呈現的是一個實數(例如：身高、體重)，這種決策樹就稱為迴歸樹(Regression Tree)。

- 常見的決策樹演算法如：ID3、C4.5、CART等

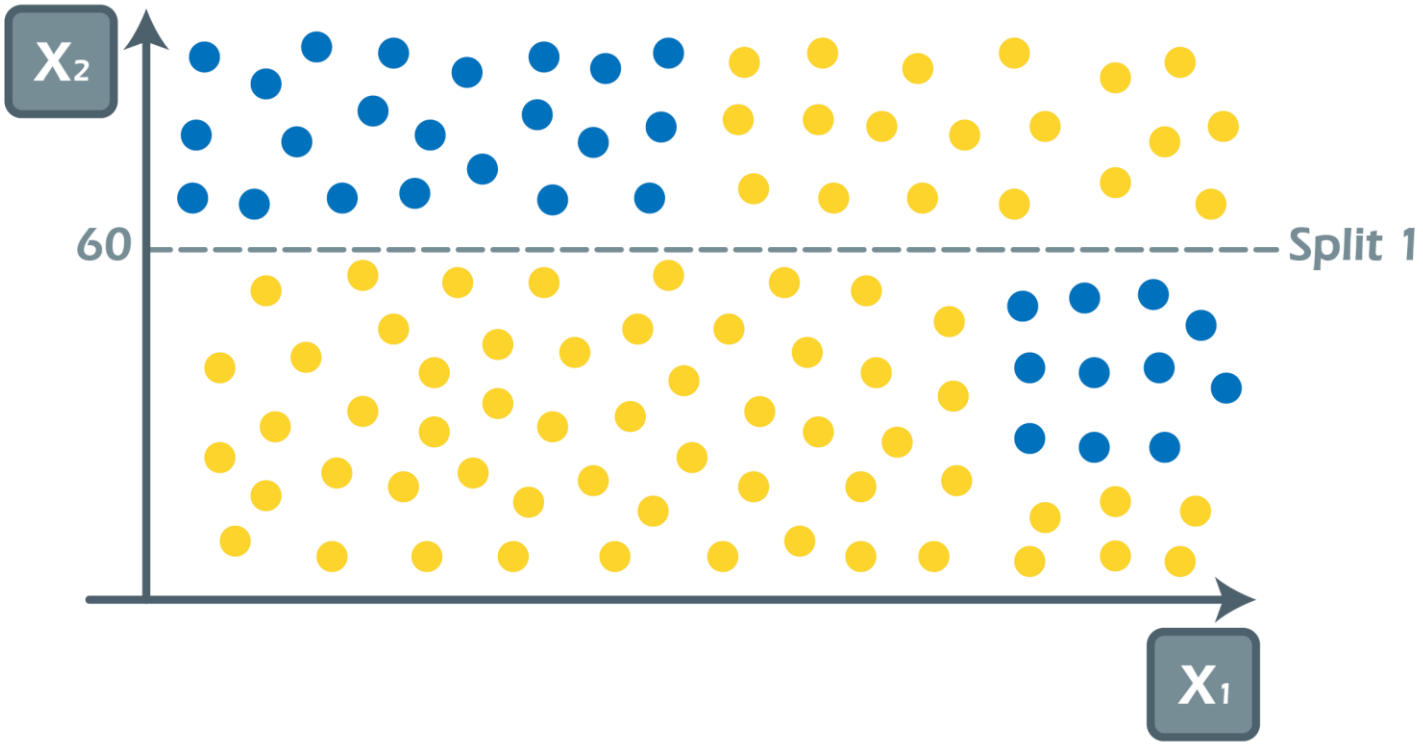
決策樹（Decision Tree）



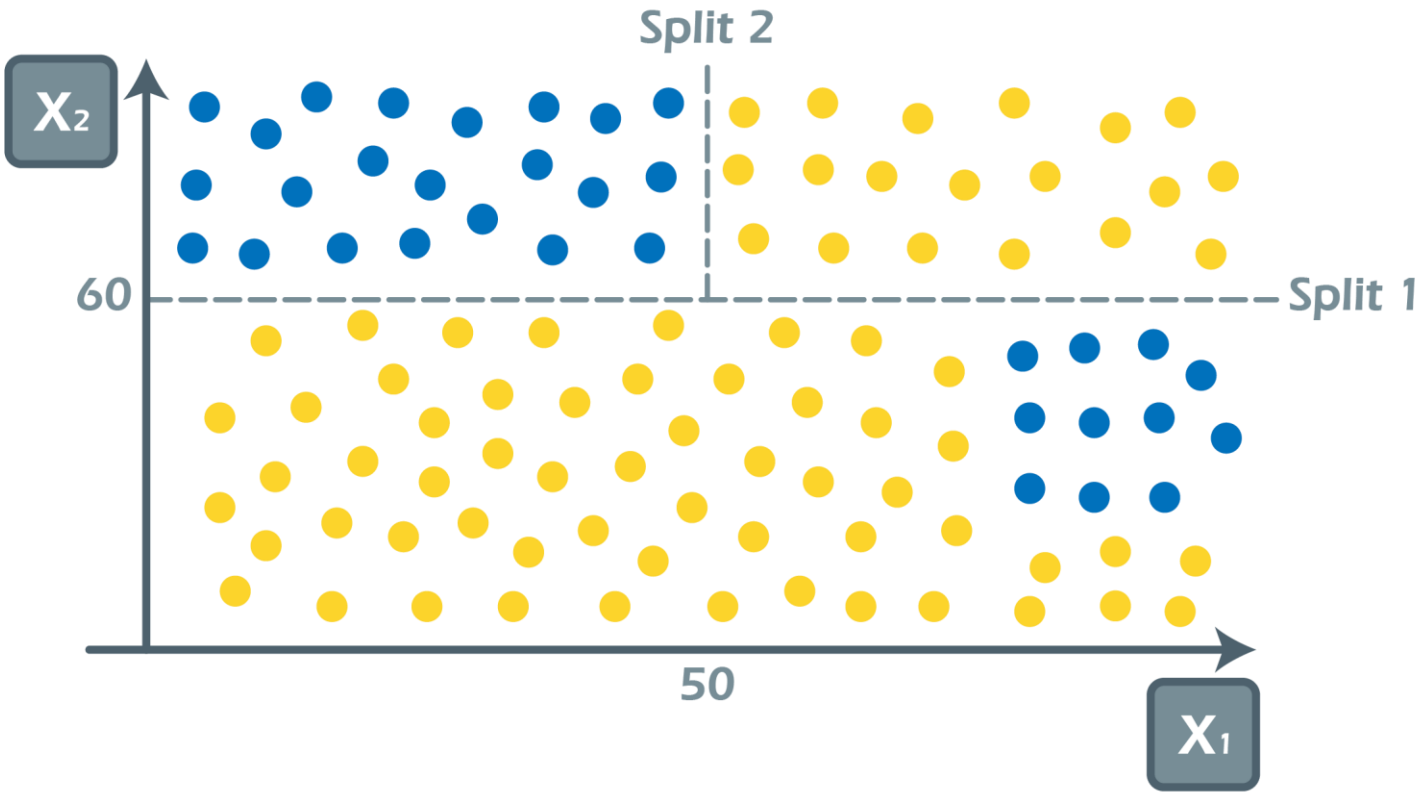
決策樹（Decision Tree）



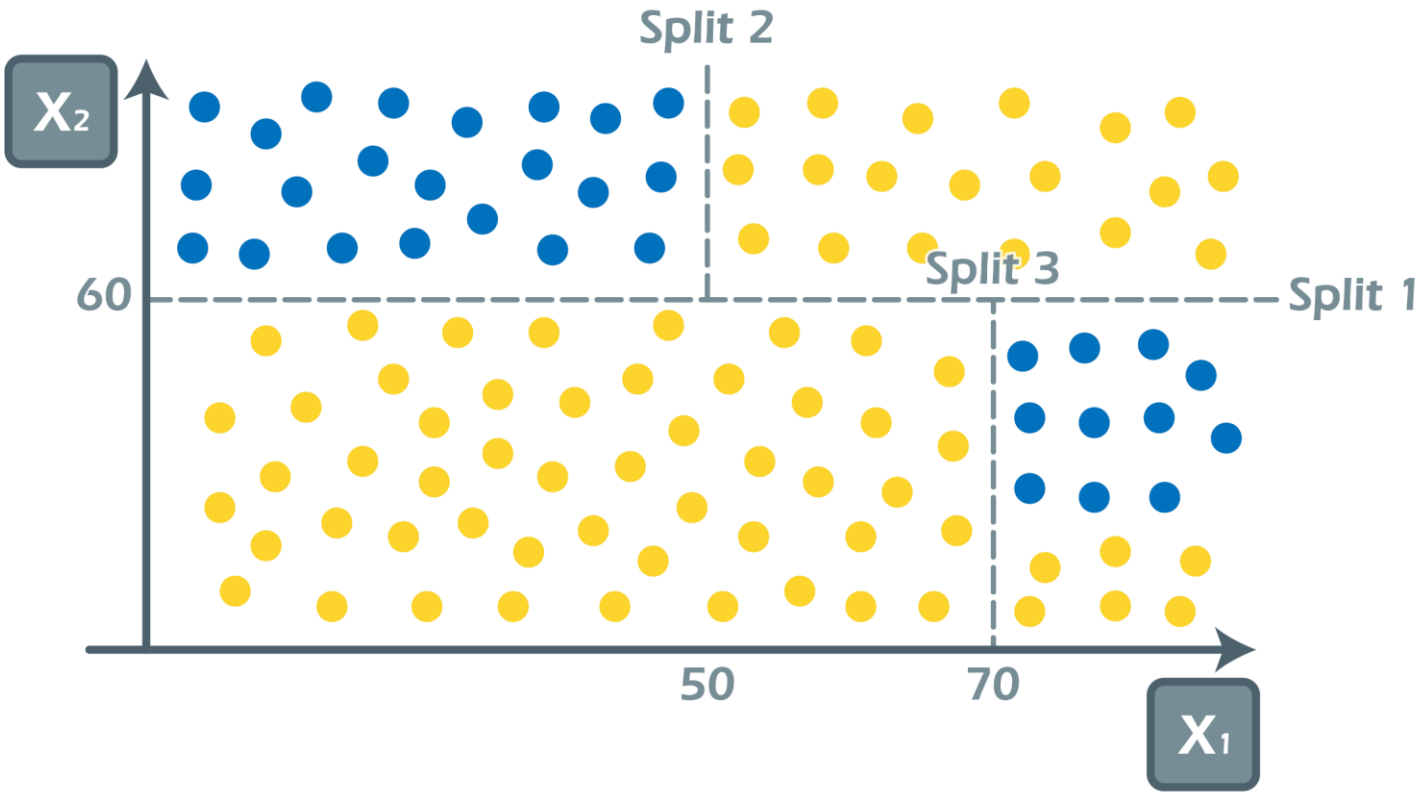
決策樹（Decision Tree）



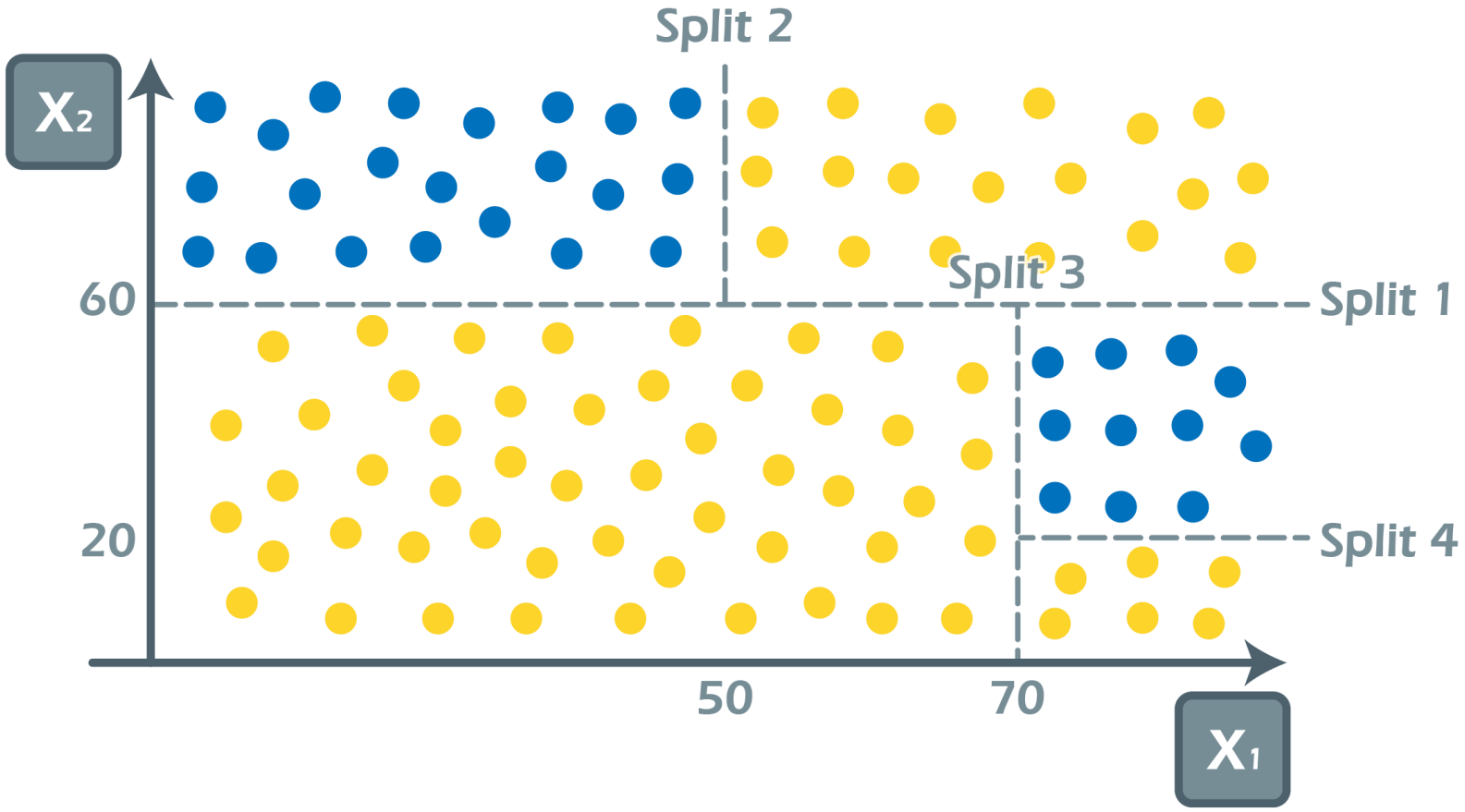
決策樹（Decision Tree）



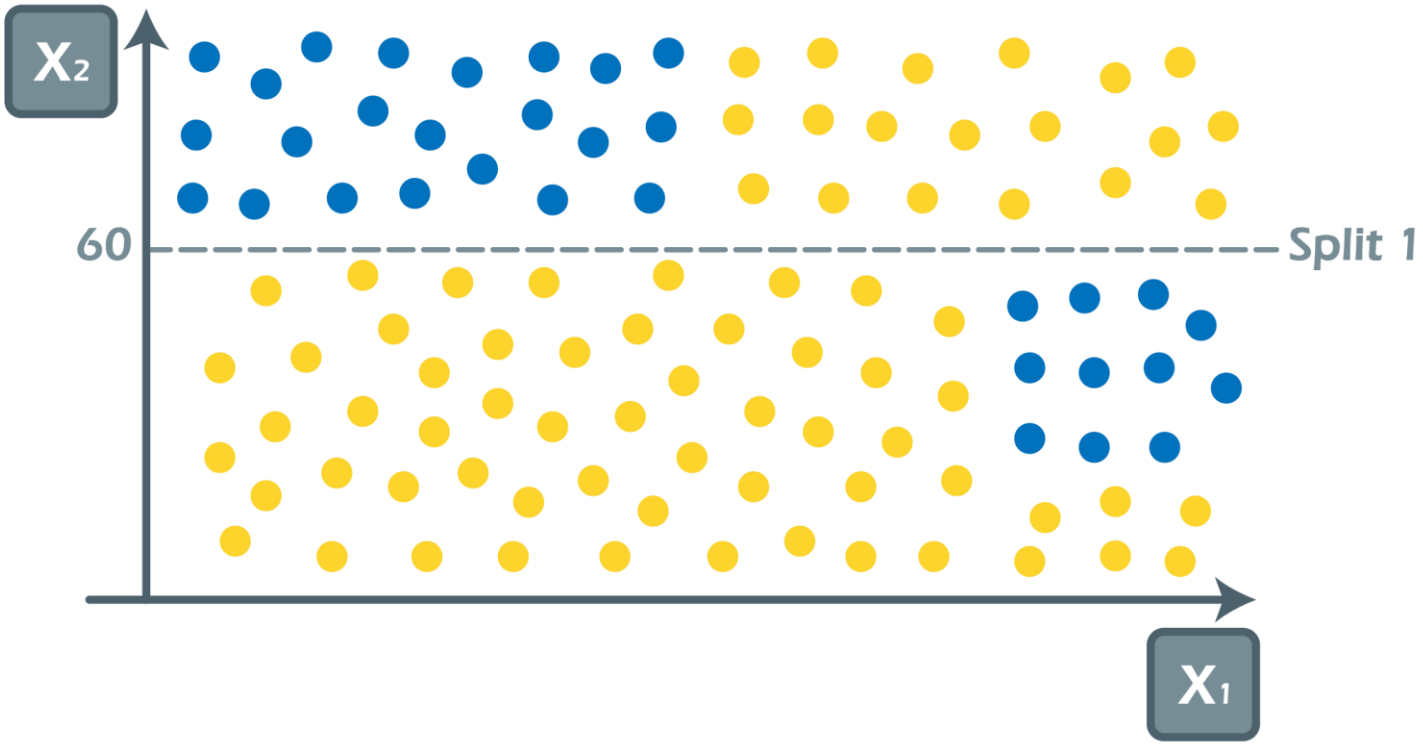
決策樹（Decision Tree）



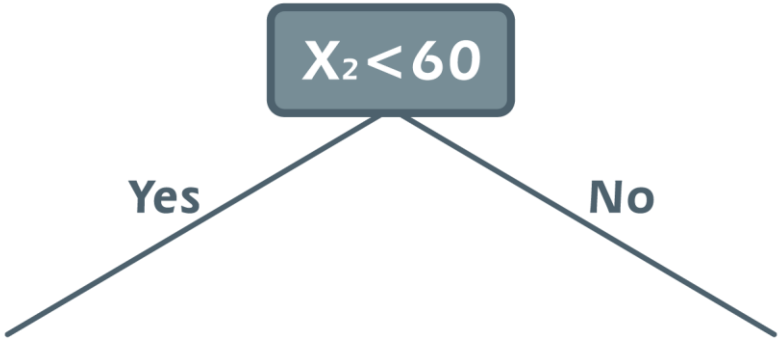
決策樹（Decision Tree）



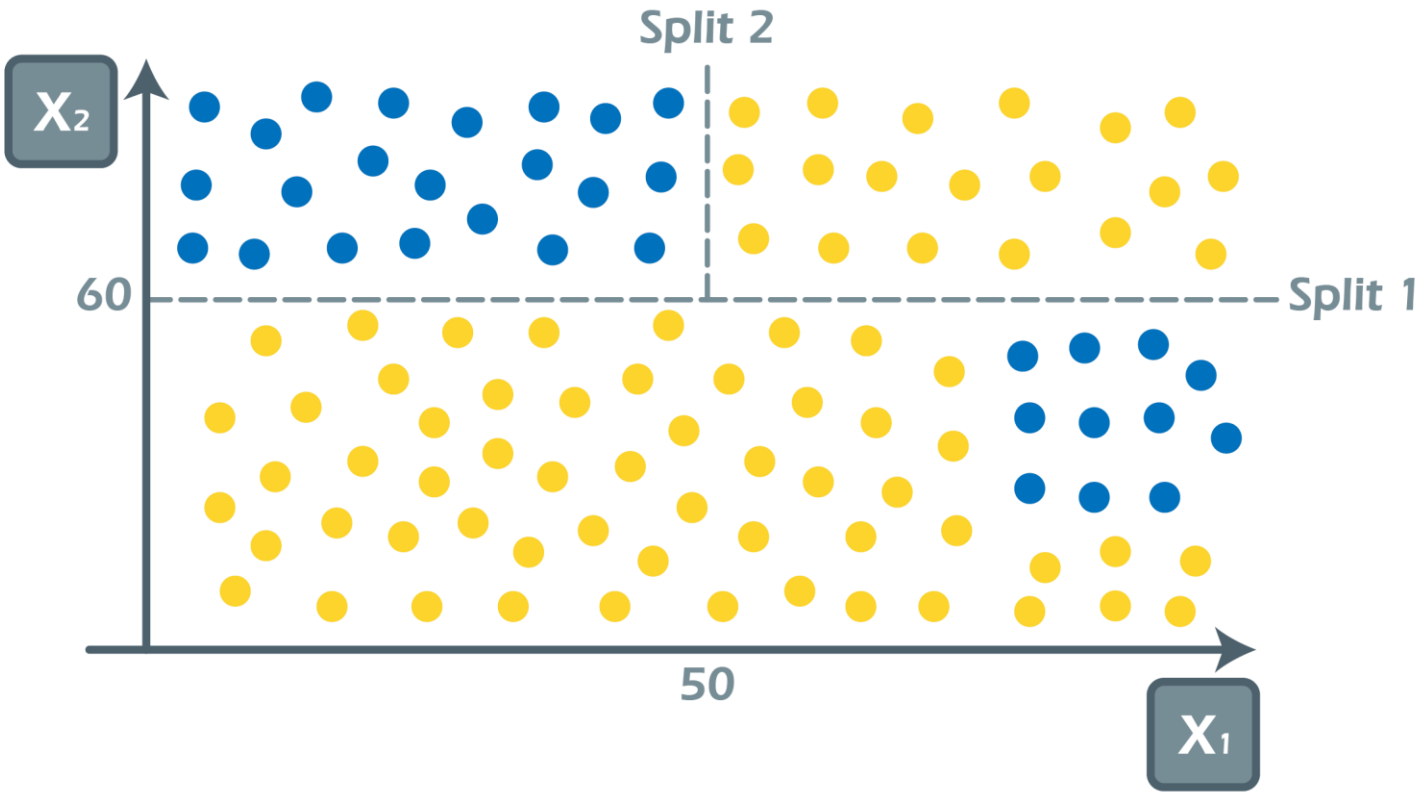
決策樹（Decision Tree）



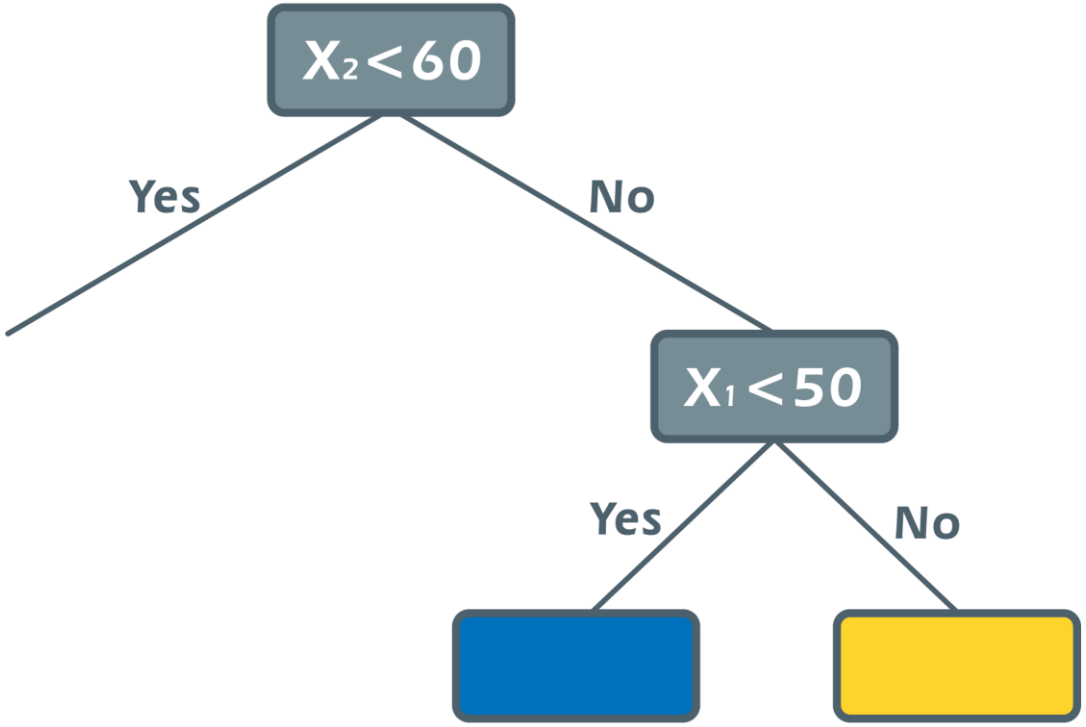
決策樹（Decision Tree）



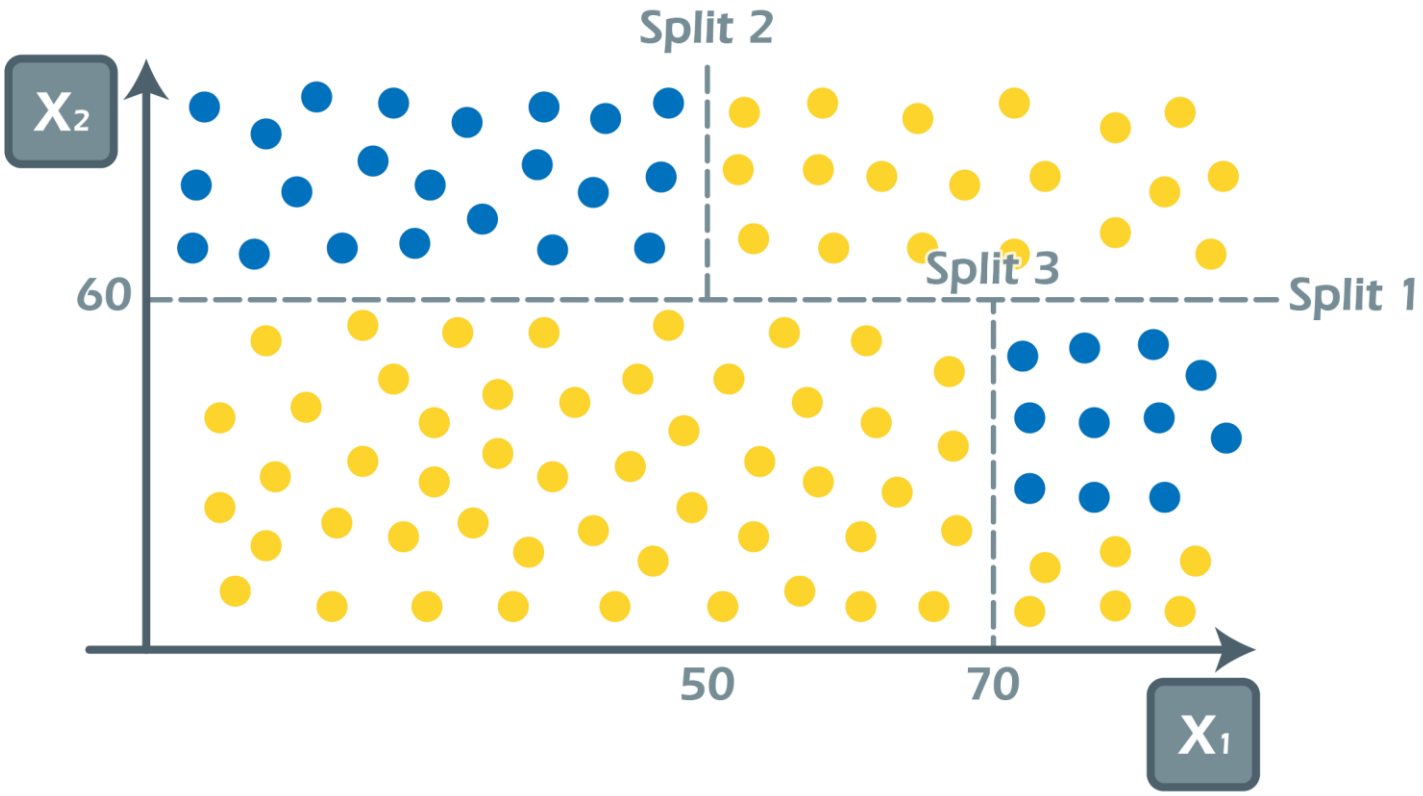
決策樹（Decision Tree）



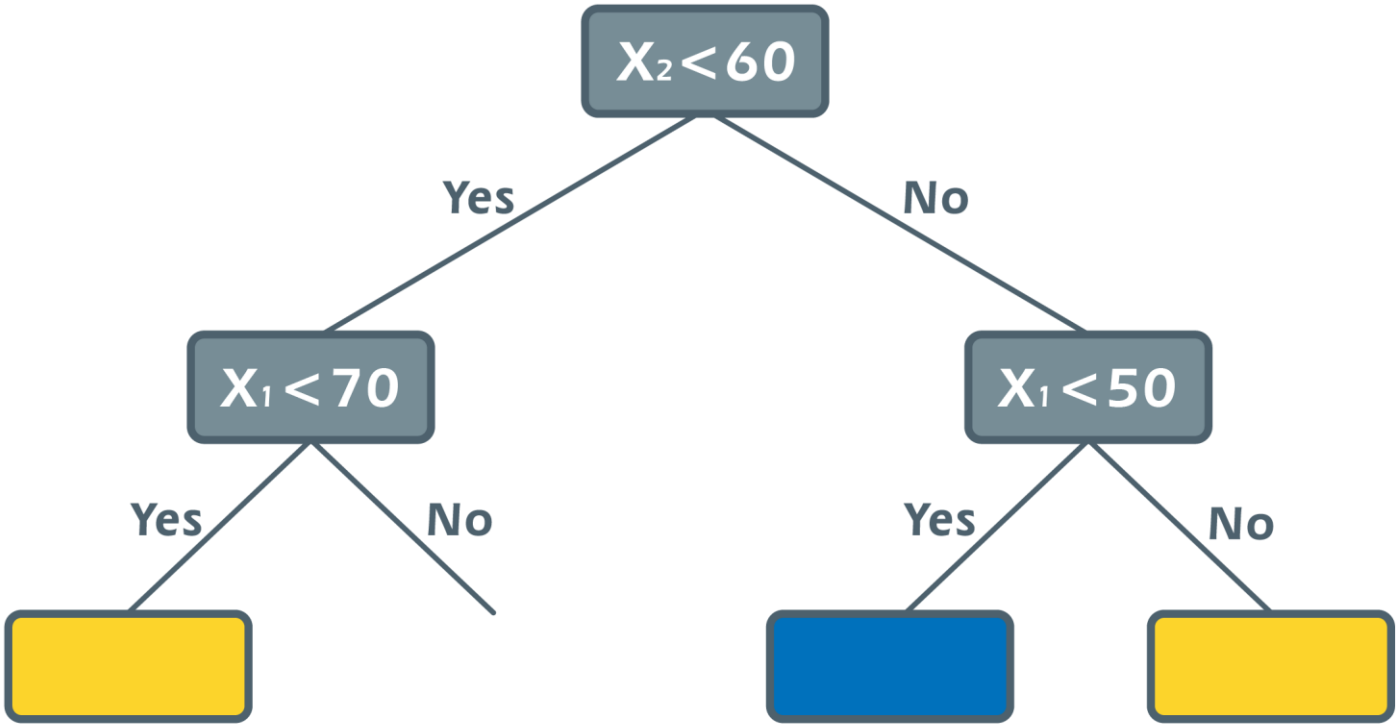
決策樹（Decision Tree）



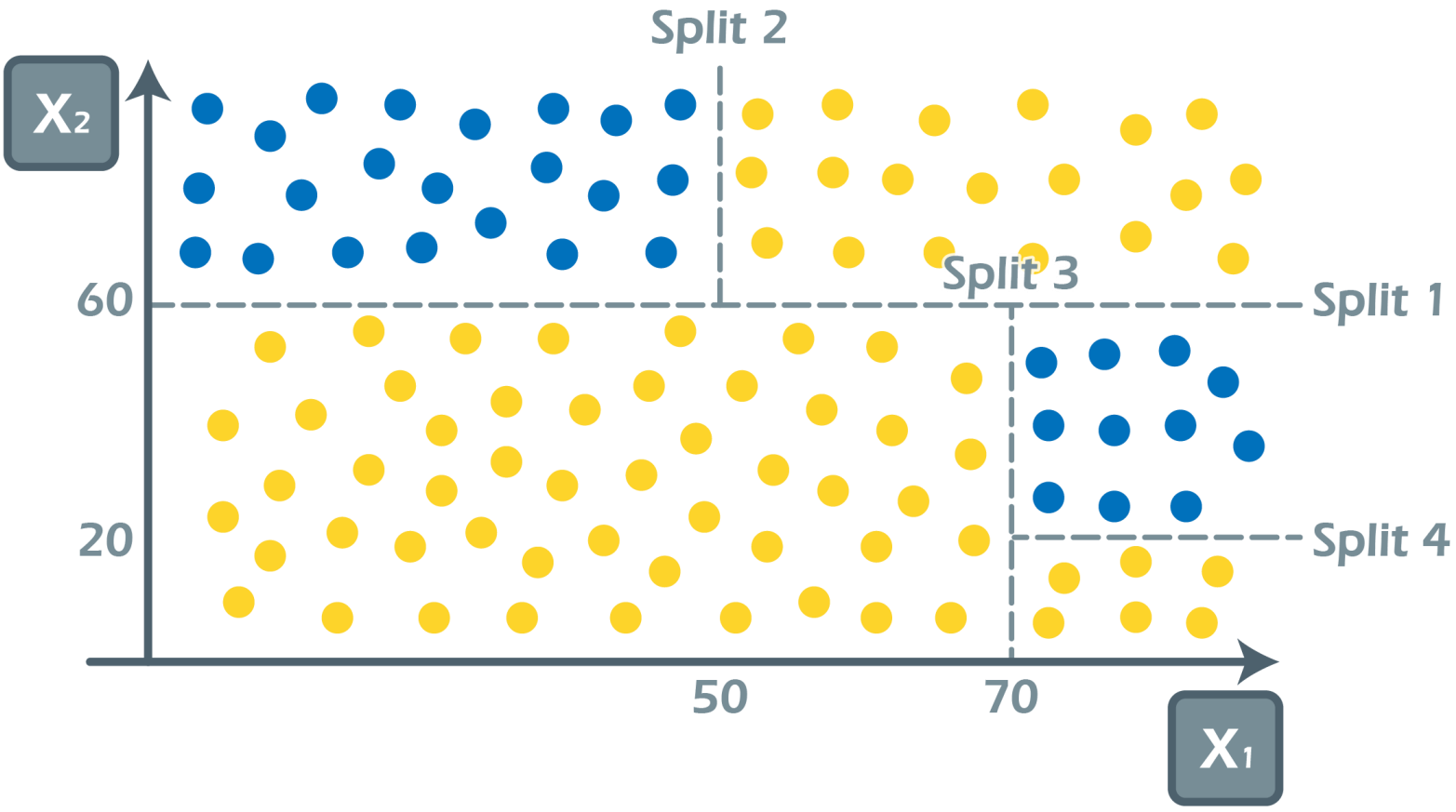
決策樹（Decision Tree）



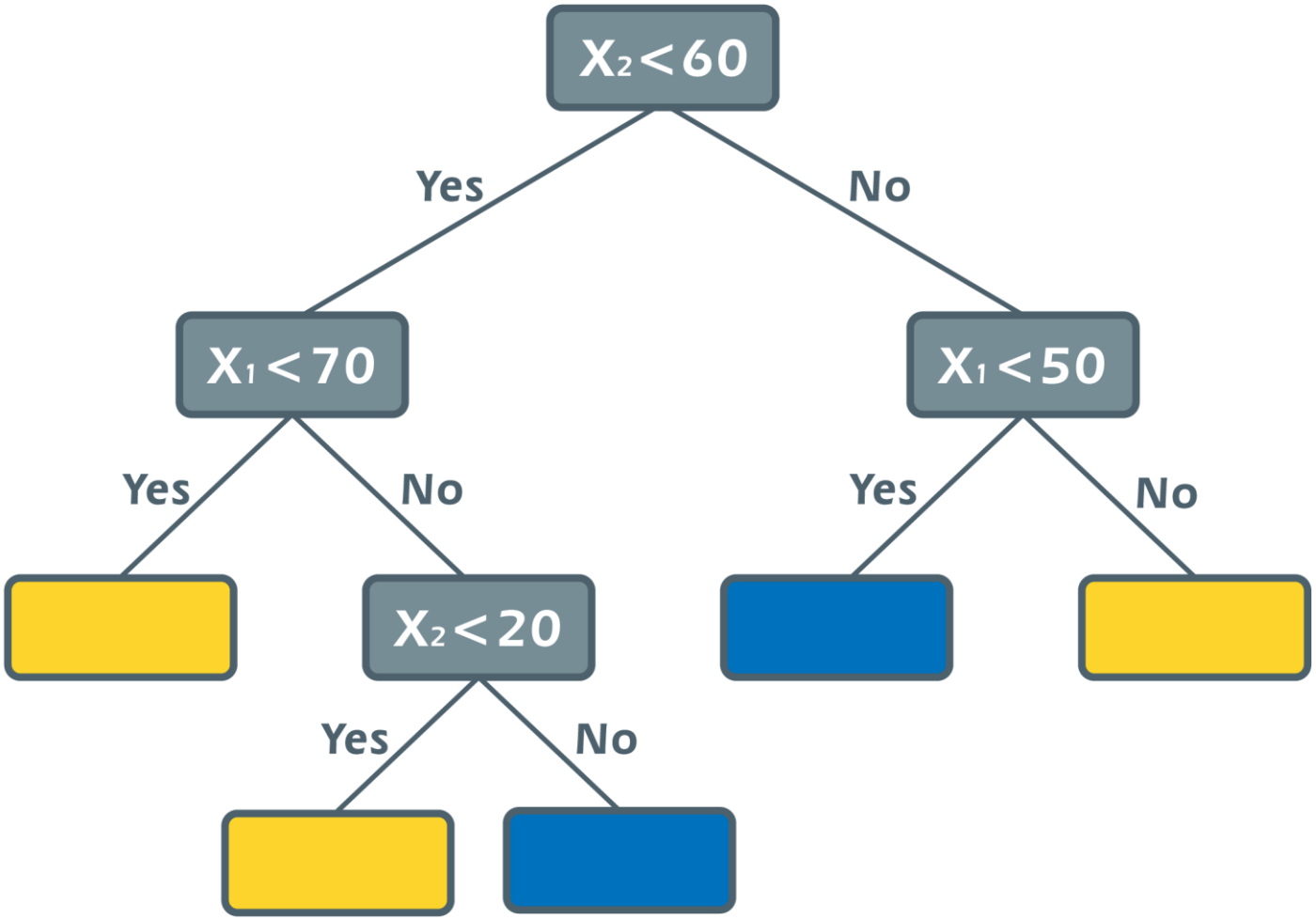
決策樹（Decision Tree）



決策樹（Decision Tree）



決策樹（Decision Tree）

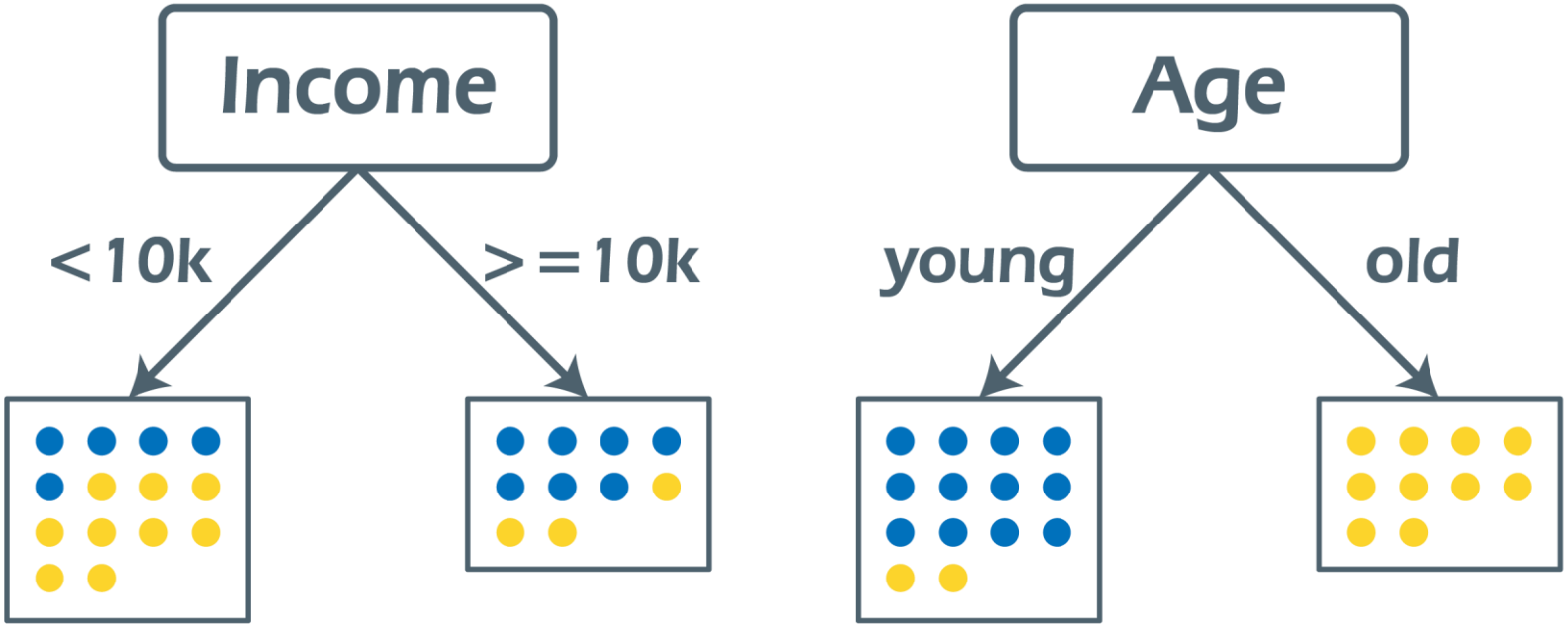


決策樹（Decision Tree）

- 如何決定分割結果是否較佳？
- 假設有一個表格共有24筆顧客資料。其類別欄位為“Customers”，可分成“好客人Good Customers”與“一般客人Fair Customers”兩類。

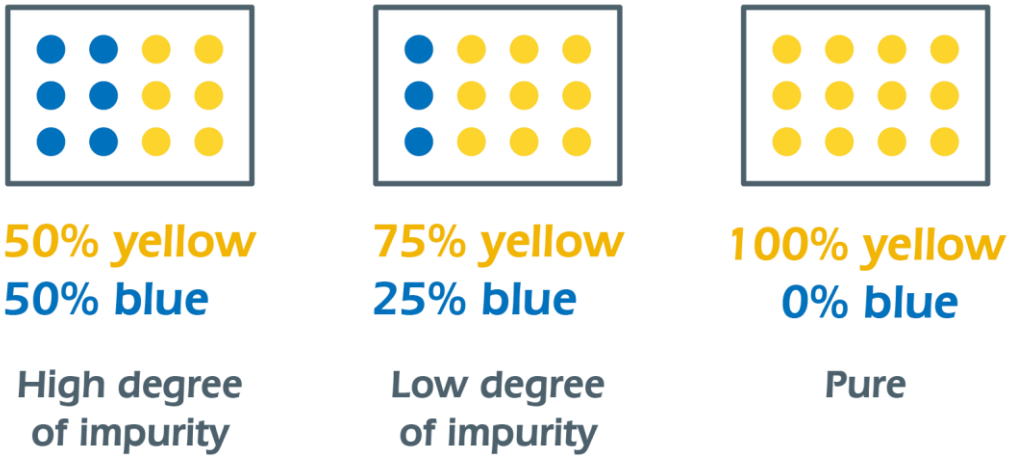
決策樹（Decision Tree）

- 分別用Income和Age兩個欄位，對這24筆顧客資料加以分割，結果如下。



決策樹（Decision Tree）

- 如何決定分割結果何者較佳: 分割結果中，若具有較高同質性(Homogeneous)類別的節點，則該分割結果愈佳。
- 因此，需要檢驗節點的不純度(Node Impurity)
- 不純度愈低愈好。



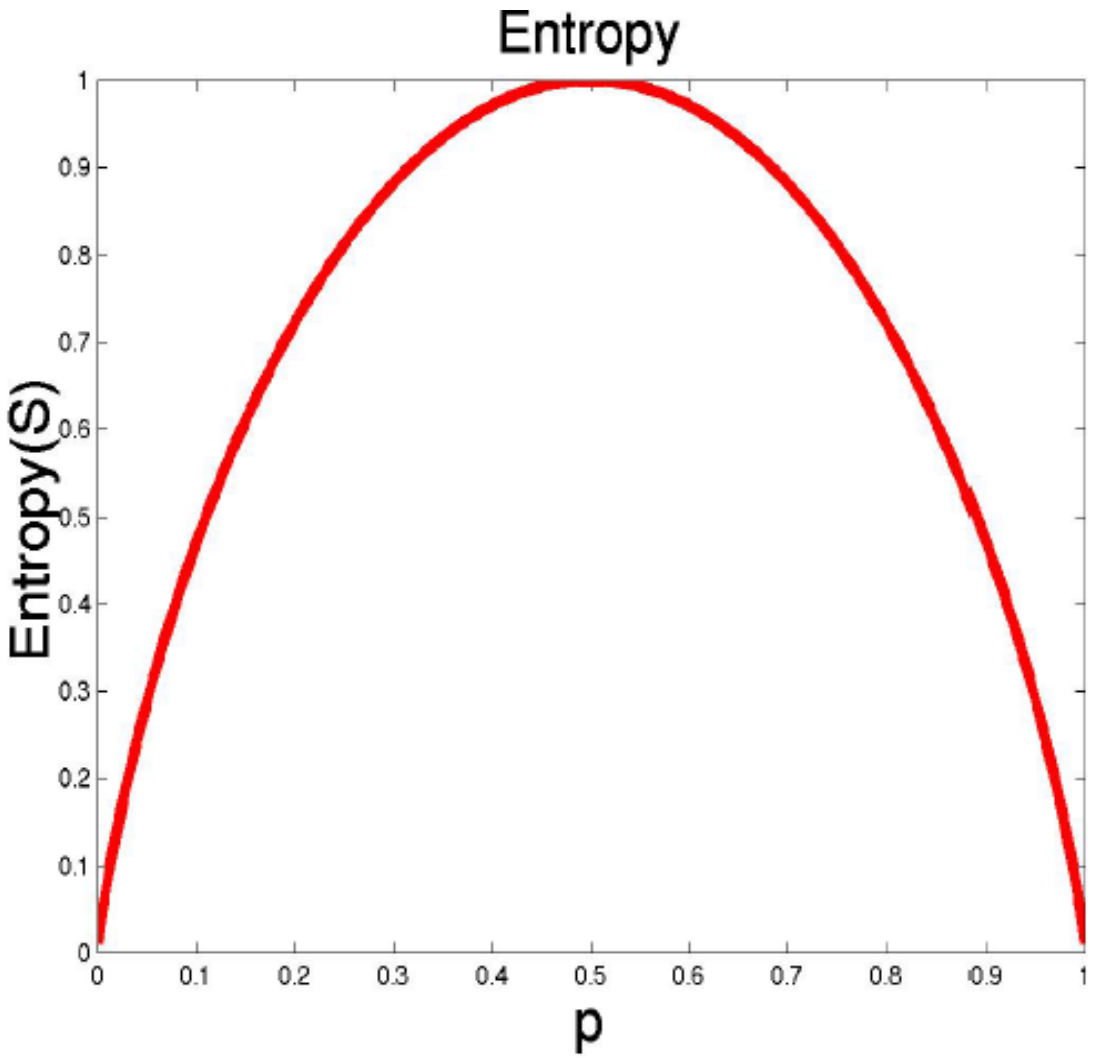
決策樹（Decision Tree）

- 用熵(Entropy) 衡量資料的一致性
- 熵(亂度)，可當作資訊量的凌亂程度(不確定性) 指標，當熵值愈大，則代表資訊的凌亂程度愈高。
- 【範例】丟銅板
- 若銅板是公平的，則丟出正面與反面的機率是一樣的(最凌亂)
- 若銅板是動過手腳的，則丟出正面與反面的機率不會是一樣的(愈不凌亂)

決策樹（Decision Tree）

- 給定一組丟銅板後之資料集合S，該組資料的熵值計算公式為
- $Entropy(S)=-p^{+}\log_2 p^{+} -p^{-}\log_2 p^{-}$
- Ex: 若丟了14次銅板，出現了9個正面與5個反面(記為[9+, 5-])，則這個 範例的熵為:
- $Entropy([9+, 5-])= -(9/14)\log_2 (9/14) – (5/14) \log_2 (5/14) = 0.94$
- 若銅板丟出正面與反面的數量是一樣，則熵為1 (最凌亂)
- 若銅板是動過手腳的，不論怎麼丟都只會出現正面(或反面)，則熵為0 (最 不凌亂)

決策樹（Decision Tree）



決策樹（Decision Tree）

- 如果資料集合S具有c 個不同的類別，那麼資料集合S 的熵值計算方式為：

$$\textbf{Entropy(S)}= \sum_{i=1}^{c} - p_i \log_2 p_i$$

- 其中p_i 為類別i 在資料集合S 出現的機率

決策樹（Decision Tree）

- ID3演算法
- ID3在建構決策樹過程中，以資訊獲利(Information Gain)為準則，並選擇最大的資訊獲利值作為分類屬性。
- 以熵(Entropy) 為基礎
- ID3演算法是利用資訊獲利來衡量屬性於分類資料的能力。
- 屬性A在資料集合S的資訊獲利Gain(S, A)被定義為:

$$Gain(S,A) = Entropy(S) - \sum_{j=1}^{v} \frac{|S_j|}{|S|} Entropy(S_j)$$

決策樹（Decision Tree）

- 假設屬性A 中有v 個不同值$\{a_1, a_2, \ldots, a_v\}$，而資料集合S會因為這些不同值而產生(分割)出$v$ 個不同的資料子集合$\{S_1, S_2, \ldots, S_v\}$
- *Entropy*(S)：資料集合S 整體的亂度
- *Entropy*(S_j)：資料子集合Sj 的亂度，其中$j = 1, 2, \ldots, v$
- $\frac{|S_j|}{|S|}$：第j 個子集合之資料個數佔總資料集合的比率(即：權重)

決策樹（Decision Tree）

- $\sum_{j=1}^{v} \frac{|S_j|}{|S|} Entropy(S_j)$ ：依據屬性A來判定資料集合S的亂度

- *Gain*(*S*, *A*) ：利用屬性A對資料集合S進行分割的獲利
  - Gain值愈大，表示屬性A內資料的凌亂程度愈小，用來分類資料會愈佳
  - Gain值愈小，表示屬性A內資料的凌亂程度愈大，用來分類資料會愈差

決策樹（Decision Tree）

- 【範例】天氣評估
  - 假設有一套天氣評估系統，它有一些評估屬性(如: 風力、濕度、...)，用以評估該天氣是否適合打網球。
  - 以風力(Wind)為例，它在所有的訓練資料中所會出現的值為: weak, strong
  - 若目前的資料集合*S* 有14筆資料，其中有9個正例與5個反例(記為[9+, 5-])
- 這14個範例資料中，關於風力的資料:
  - Wind = weak 有8筆資料(*Sweak*)，其中有6個正例與2個反例[6+, 2-]
  - Wind = strong 有6筆資料(*Sstrong*)，其中有3個正例與3個反例[3+, 3-]

決策樹（Decision Tree）

- 我們想要得知風力這個屬性的資訊獲利為多少。

Day	Outlook	Temp.	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Weak	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cold	Normal	Weak	Yes
D10	Rain	Mild	Normal	Strong	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

決策樹（Decision Tree）

- 我們想要得知風力這個屬性的資訊獲利為多少。

$$Values(Wind) = Weak, Strong$$

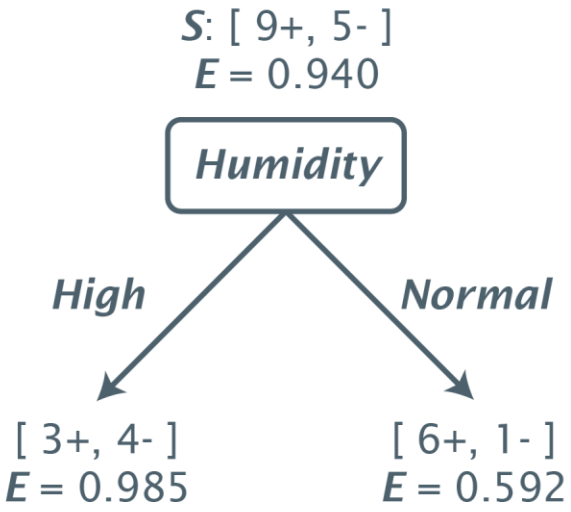
$$S = [\,9+,\ 5-\,]$$

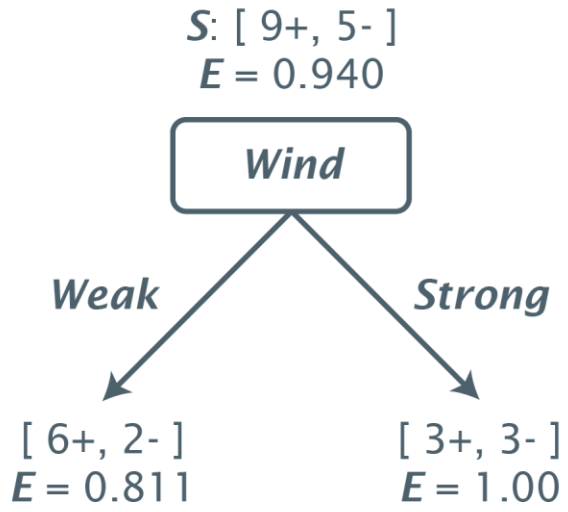
$$S_{Weak} \leftarrow [\,6+,\ 2-\,]$$

$$S_{Strong} \leftarrow [\,3+,\ 3-\,]$$

決策樹（Decision Tree）

Which attribute is the best classifier?

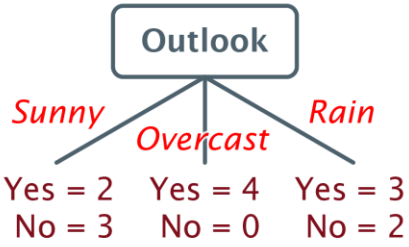




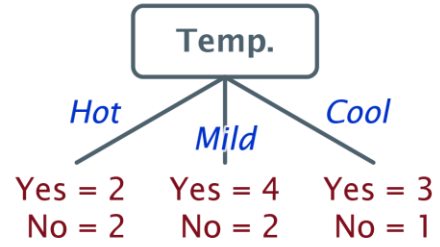
Gain (S, Humidity)
= .940 - (7/14).985 - (7/14).592
= .151

Gain (S, Wind)
= .940 - (8/14).811 - (6/14)1.0
= .048

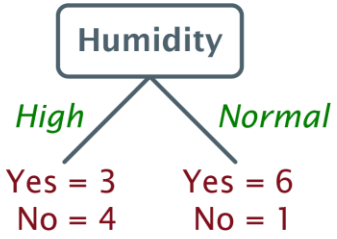
Day	Outlook	Temp.	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Weak	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cold	Normal	Weak	Yes
D10	Rain	Mild	Normal	Strong	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No



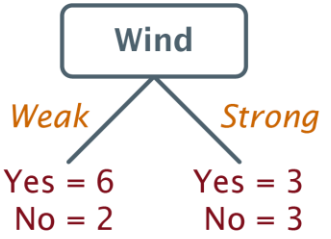
Gain(S, Outlook) = 0.246



Gain(S, Temperature) = 0.029



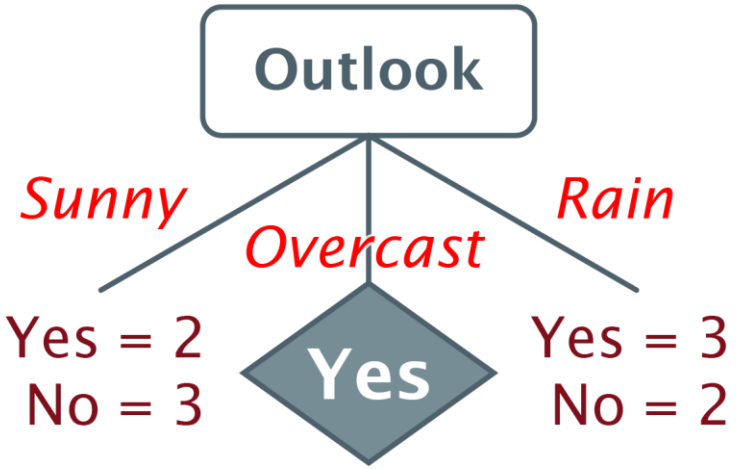
Gain(S, Humidity) = 0.151



Gain(S, Wind) = 0.048

決策樹（Decision Tree）

- 挑出具最大資訊獲利的屬性，因此以Outlook為根節點(root)
- 由於Outlook的三個評估值中，Overcast(多雲)的這個評估值得到4個正例(Yes)，沒有任何反例，因此Outlook = Overcast可得到一個葉子節點 “Yes”。

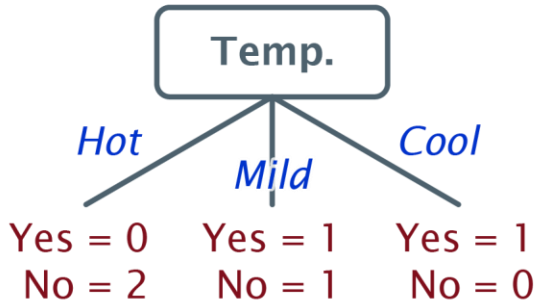


決策樹（Decision Tree）

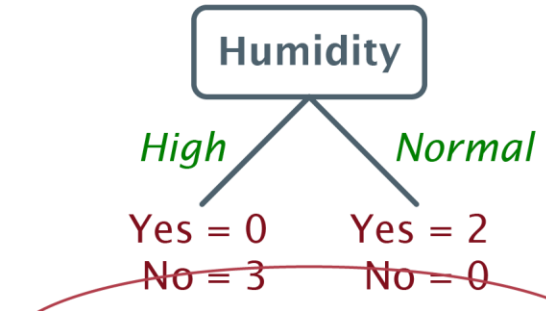
- 接著，將原始資料中，**Outlook=Sunny**的所有資料列出，並對**Outlook**以外的其它所有內部欄位計算其資訊獲利

Day	Outlook	Temp.	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cold	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes

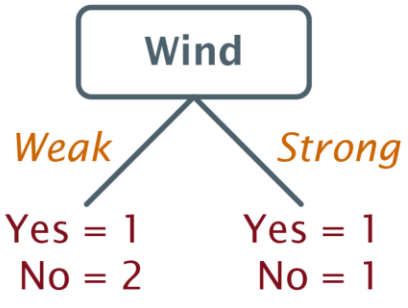
決策樹（Decision Tree）



Gain(S_{Sunny}, Temp.) = 0.571



Gain(S_{Sunny}, Humidity) = 0.971

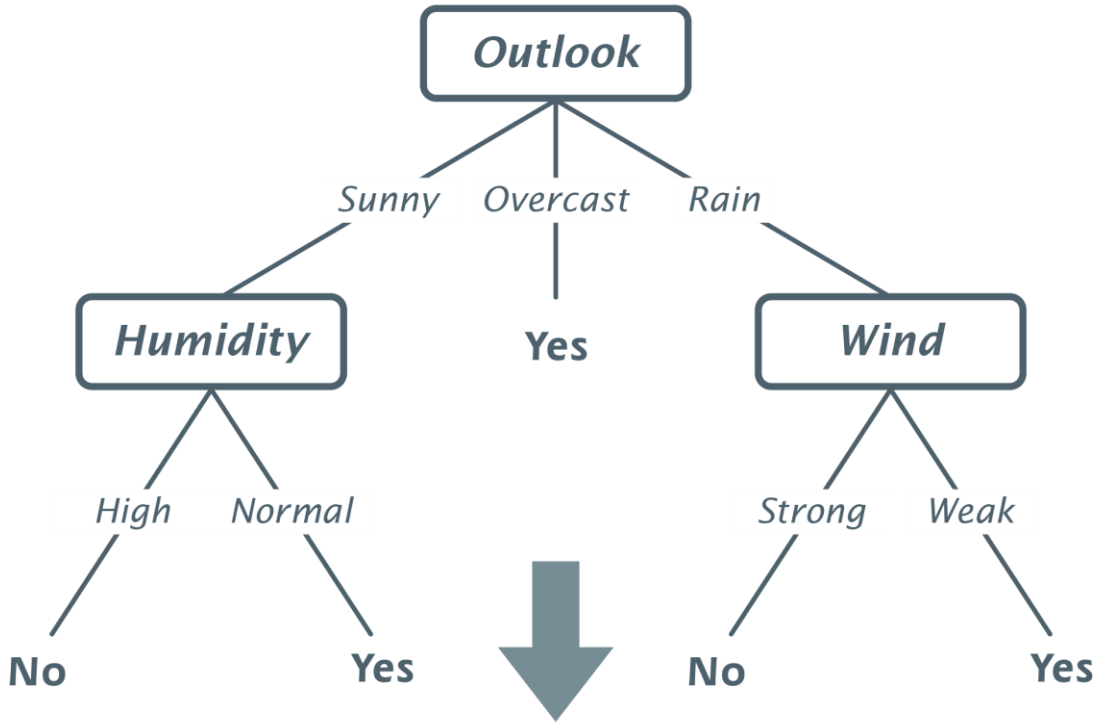


Gain(S_{Sunny}, Wind) = 0.020

接著，將原始資料中，**Outlook=Rain**的所有資料列出，並對**Outlook**以外的其他所有內部欄位計算其資訊獲利。

Day	Outlook	Temp.	Humidity	Wind	Play Tennis
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D10	Rain	Mild	Normal	Strong	Yes
D14	Rain	Mild	High	Strong	No

決策樹（Decision Tree）



分類規則：

If Outlook = Sunny and Humidity = High Then Play Tennis = No

If Outlook = Sunny and Humidity = Normal Then Play Tennis = Yes

If Outlook = Overcast Then Play Tennis = Yes

If Outlook = Rain and Wind = Strong Then Play Tennis = No

If Outlook = Rain and Wind = Weak Then Play Tennis = Yes

決策樹（Decision Tree）

- Old Method
- Reborn with upgrades
- Random Forest
- Gradient Boosting
- etc.

THE END

ytlin@mail.nptu.edu.tw