

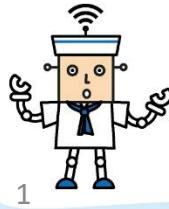


## 第7章

資料科學 Level UP !

# 認識機器學習演算法

一批批的電腦科學家在過去 60 年間不斷投入大量心力研發人工智慧 (Artifical Intelligence, AI)，如今，機器學習已經被廣泛應用在如人工智慧、大數據分析等不同的領域。正如 30 年前，資料處理是當時值得學習的基本能力，而資料分析 (如利用機器學習技術) 則將會是現在及未來不可或缺的技能。



## 7-1

# 機器學習的概念

- 機器學習就是使用電腦將大量而又紛雜的原始資料，透過資料科學的分析技巧找出其中所隱藏的資訊。
- 讓電腦從這些資料中找出其變動模式，進行學習判斷或是預測。

# 7-1-1 人工智慧的演進

- **人工智慧** (Artificial Intelligence, AI)
  - 一個期待的目標，而不是具體的方法。
  - 例如：期望電腦或機器能夠完成具有人類智慧才能達成的事情。
- **機器學習** (Machine Learning, ML):
  - 實踐人工智慧的演算法。
  - 簡單來說，就是讓機器能自動學習，從人工給予的資料中找到規則，進而擁有預測、分類等能力。
- **深度學習** (Deep Learning, DL):
  - 機器學習的一個分支，主要是用來訓練能力更強的模型，使機器的學習效果能更好，提高準確度。



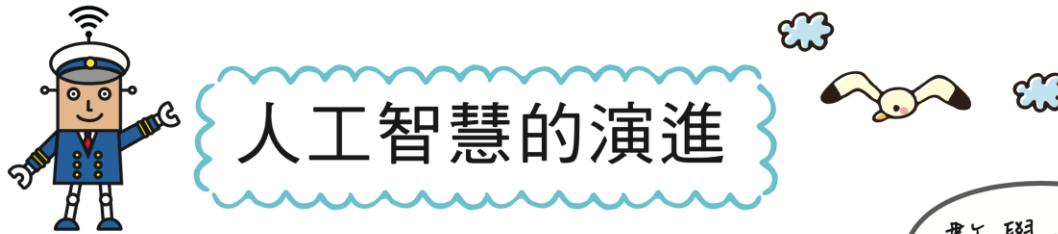
1950年代

人工智能



「夢想」  
電腦是可從經驗  
中學習的機器

人類能輕易辨識貓，  
但是不知如何訓練電  
腦學習辨識貓。



未來的AI可以  
模擬人類智慧

數學比較  
簡單！

我不知道！  
好難！

這是什麼  
動物呢？



資料科學 × 機器學習

實戰探索

Practical Exploration



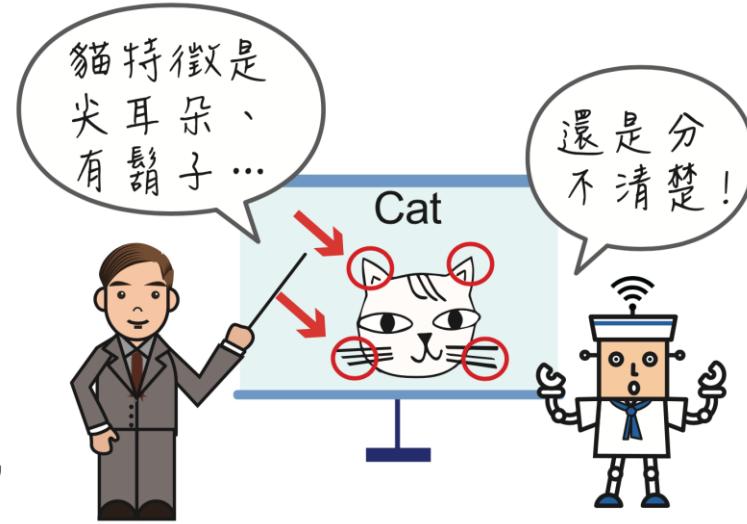


1980年代

## 機器學習

電腦可從**歷史資料**中，學習一套技能！

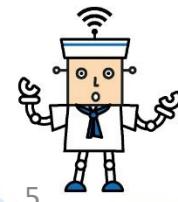
人類用貓的**特徵值**來訓練電腦學習辨識貓，但是辨識率不佳。



# 資料科學 × 機器學習

實戰探索

Practical Exploration

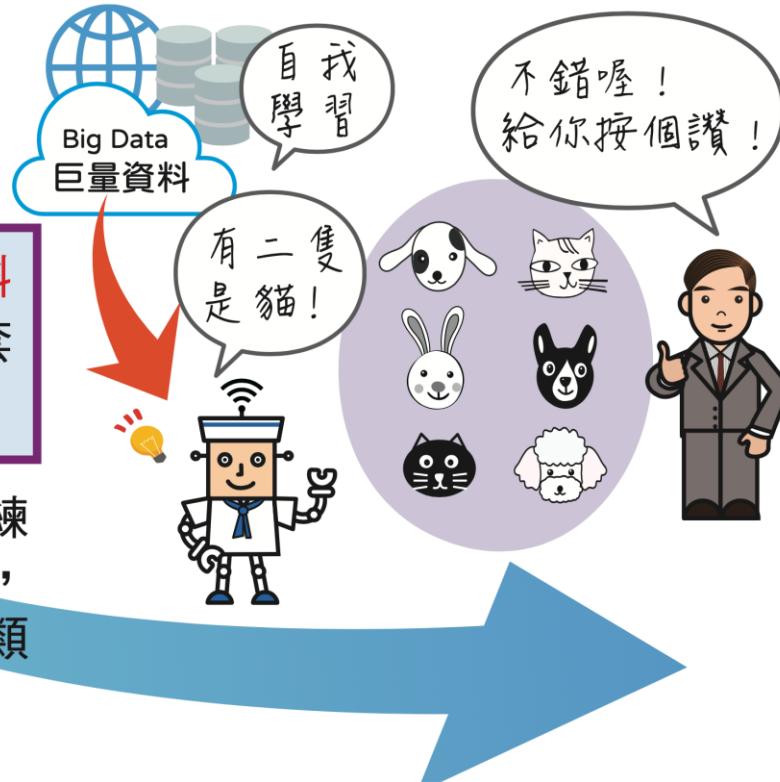




## 2010年代 深度學習

電腦可從**巨量資料**中，**自己學習**一套好的技能！

引用**大量資料**來訓練電腦提升學習效果，使得辨識率達到人類的水準。



## 7-1-2 什麼是機器學習

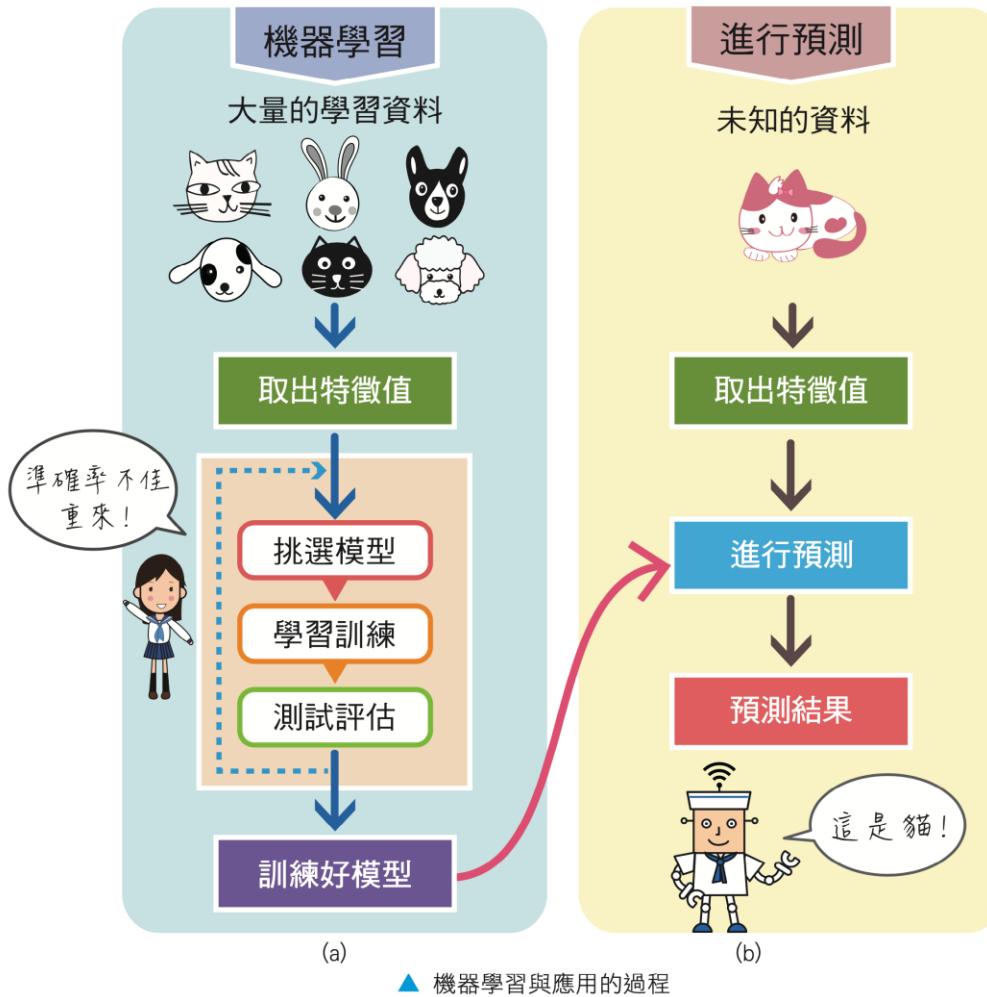
- 引用數學上的機率學、統計學等知識，企圖透過人類給予的資料進行「**訓練**」
- 讓電腦自行學會一套技能或規則，訓練完成後會產生「**模型 (Model)**」
- 這個過程就稱為「**機器學習**」



## 7-1-2 什麼是機器學習

- 模型想成是  $y = f(x)$
- 只要代入  $x$ ，就能得到  $y$
- 例如：輸入一張貓 ( $x$ ) 的照片到模型  $f(x)$  中，模型會輸出（識別）這是「貓」的答案 ( $y$ )





# 資料科學 x 機器學習

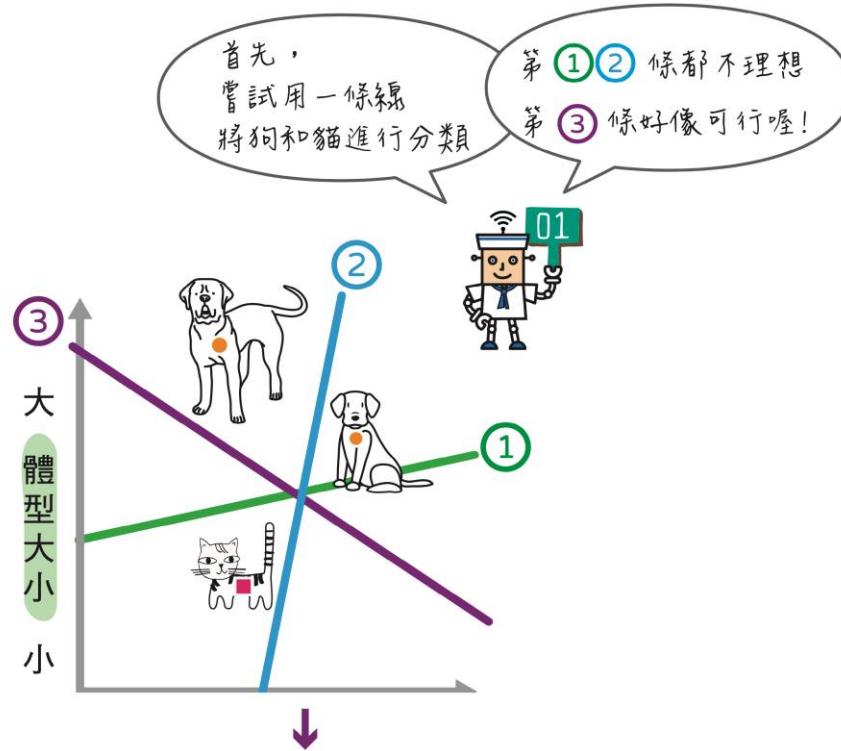
實戰探索

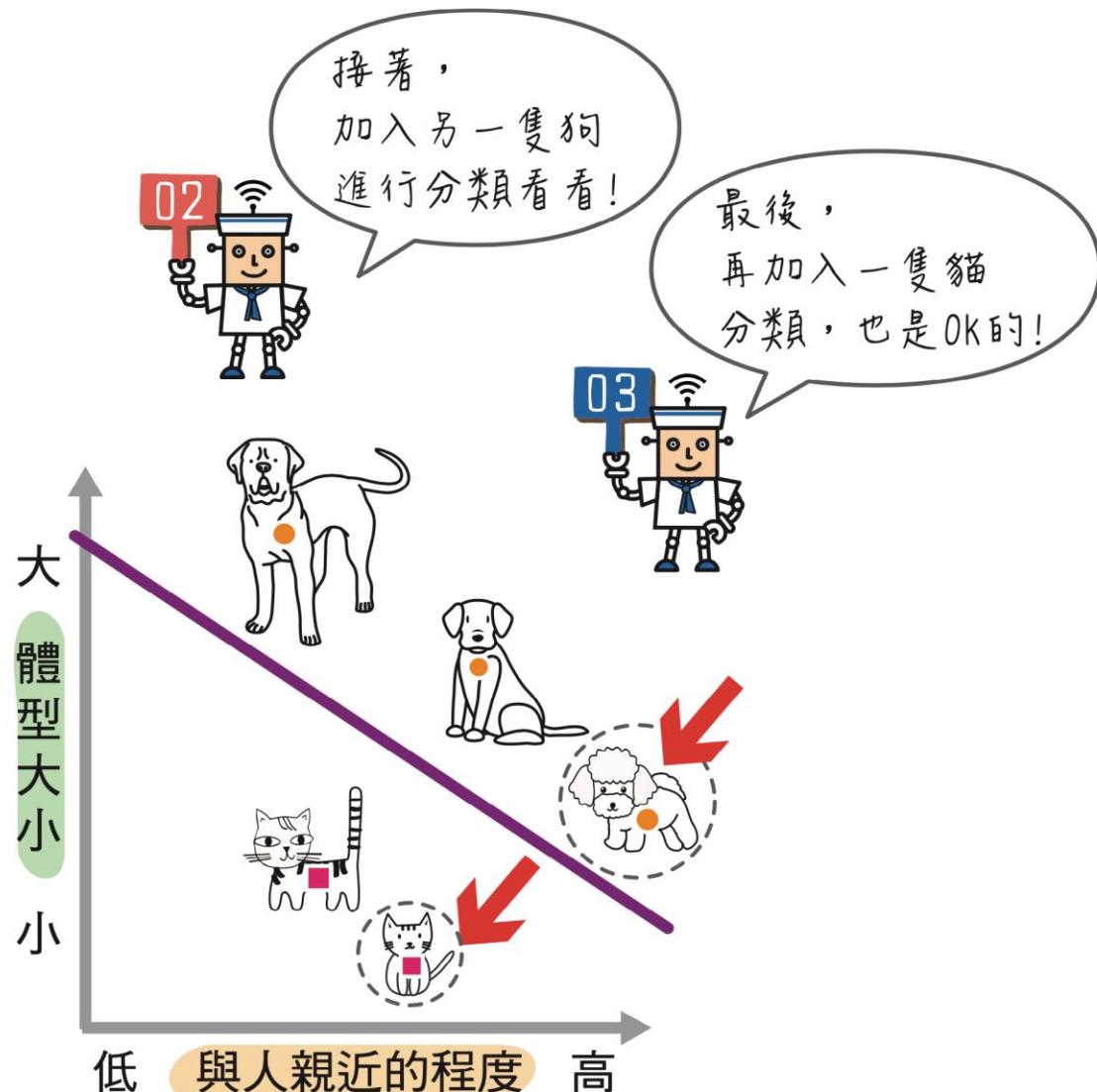
Practical Exploration



## 7-1-2 什麼是機器學習

- 假設我們的模型長得就是「 $y = m \times x + b$ 」
- 打算用一條直線表示對貓狗做分類





▲ 機器學習的概念：以一條直線區別貓與狗

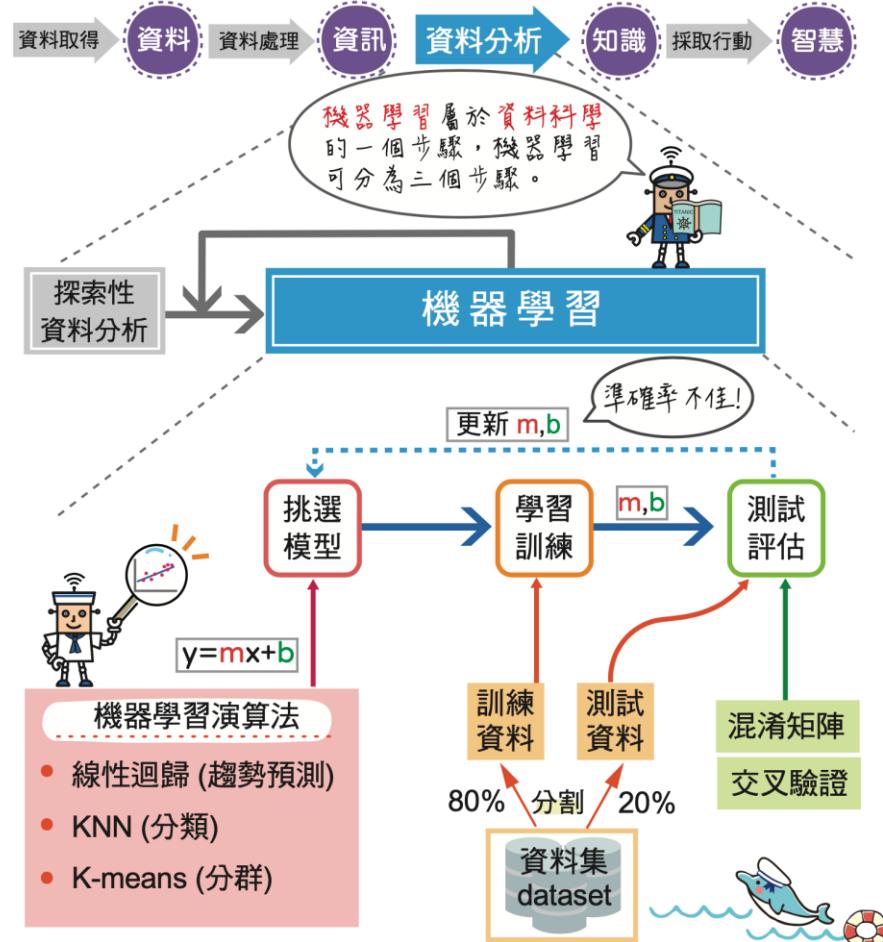
## 7-1-2 什麼是機器學習

- 一開始，不知道模型的  $m$  和  $b$  的值應該選多少才好
  - 需要經過多次的嘗試，透過修改斜率  $m$  及截距  $b$  來改變直線
  - 再經過「測試」(Test) 與「評估」(Evaluation)。
- 重複的學習 (訓練) 就能找到更好的  $m$  及  $b$  值
- 一旦決定  $m$  和  $b$  的值之後，模型就完成
  - 有新資料  $x$  時，就可以輸入到模型中，輸出預測結果  $y$ 。

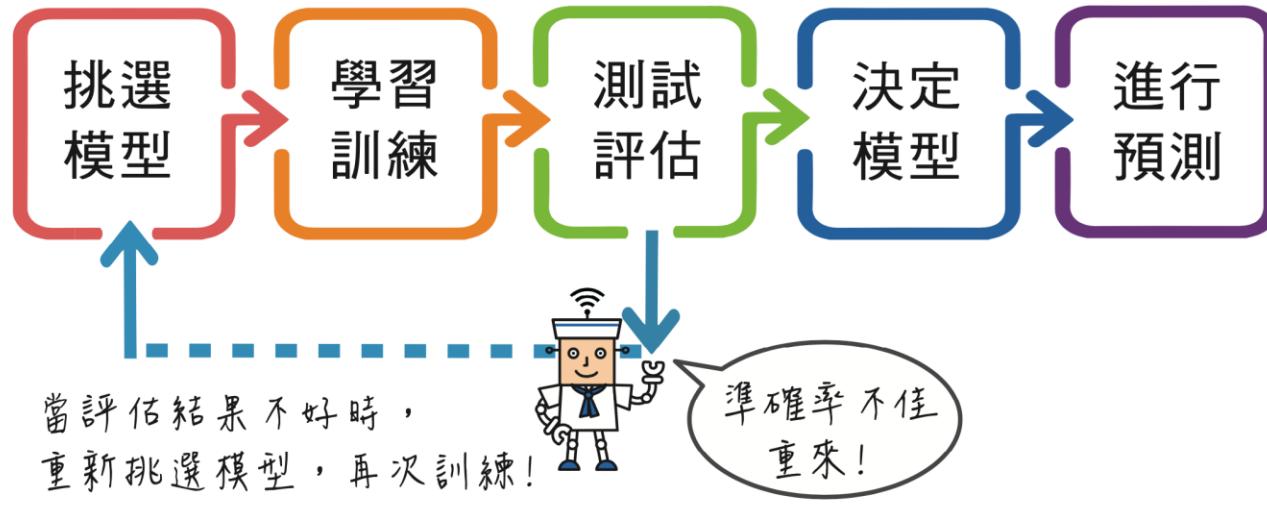


# 7-1-3 機器學習的實作步驟

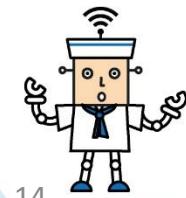
- 學習的過程包含  
**挑選模型**  
**學習訓練**  
**測試評估**  
等步驟



## 機器學習步驟



▲ 機器學習的實作步驟





## 挑選模型

首先問個感興趣的問題，再根據問題類型來挑選模型。例如：以下三個問題應該採用不同的模型。

- 問題 A：飲料店老闆能不能以氣溫預測當天冰品的銷售量？
- 問題 B：如果時光倒流，我也搭上當年的鐵達尼號，我會不會是僥倖存活下來的人之一呢？
- 問題 C：在公園內撿到許多枯葉，能不能分析出它們屬於「幾種」植物呢？





## 學習訓練

由取得的「**訓練用資料**」中取出**特徵值** (Features) 及**標籤** (Label)，交給模型做為**訓練**之用，這裡的標籤是指問題的**解答**。如下表就是上述 3 個問題的特徵值及標籤。

### ▼ 問題的特徵值及標籤

	特徵值	標籤
問題 A	氣溫	實際銷售量
問題 B	性別 年齡 艙等	實際是生或死
問題 C	葉長 葉寬	無



## 7-1-3 機器學習的實作步驟

- 「訓練」就好像平時研讀**考古題**
- 機器學習用的資料集通常會分為兩堆，
  - 一堆稱為「**訓練用資料** (Training Data)」（如：用考古題來做為學習教材）
  - 另一堆稱為「**測試用資料** (Test Data)」（如：做為模擬考試卷用來檢測學習成果）。
- 通常前者資料量會佔**80%**，而後者則佔**20%**。





## 測試評估

由取得的「**測試用資料**」中取出**特徵值及標籤**，交給模型做為**測試評估**之用，如果準確性不佳，則再重複學習訓練的步驟。就好像當模擬考試考得不理想時，就回頭重新再學習一次。

如何評估測試結果的優劣呢？主要是靠**準確度** (Accuracy)，也就是比較測試結果與真正標籤 (解答) 的差距。常見的評估工具<sup>註1</sup>有**混淆矩陣** (Confusion Matrix) 和**交叉驗證** (Cross Validation)。





## 決定模型

沒有學習過的模型就像是學齡前的幼童，缺少足夠的經驗和能力。通過測試評估之後，**確定模型的準確性**在可被接受的範圍時，模型才能算是建立完成，並且可以拿來使用。



## 進行預測

此階段開始正式上場實戰，輸入新的資料給模型，模型會輸出預測的答案。需要注意的是，重複輸入相同的資料，通常輸出的預測答案會相同，但也是會有例外，畢竟**機器學習的準確率並不是百分百**<sup>註2</sup>。



## 7-1-4 監督式與非監督式學習

- 提供資料與解答的學習方式稱為**監督式學習**(Supervised Learning)
- 提供資料、不提供解答的學習方式則稱為**非監督式學習**(Unsupervised Learning)





## 機器學習

### 監督式學習

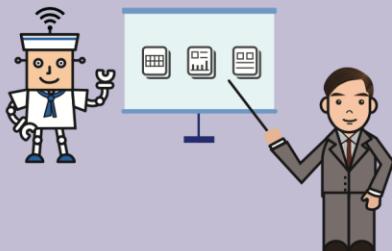
提供資料與解答的學習方式

有解答的資料集

特徵值

標籤(解答)

姓名	性別	艙等	生還否
Rose	女	頭等艙	生還
Jack	男	三等艙	死亡
Mrs. Brown	女	頭等艙	生還



### 非監督式學習

只提供資料、不提供解答的學習方式

無解答的資料集

只有「特徵值」

長度(公分)	寬度(公分)	重量(克)
5.1	3.5	198
6.3	2.5	323
5.6	2.9	250



▲ 監督式學習與非監督式學習



## 7-2

# 常見的機器學習演算法

### ▼ 機器學習領域常見的類型

類型	線性迴歸分析	分類	分群
目的	趨勢預測分析	類別或等級的識別	類別或等級的區別
功能	連續值，預測數值為多少	非連續，負責識別出哪一種	非連續，負責區別出有幾群
學習方式	提供解答（標籤）	提供解答（標籤）	沒有解答（標籤）
應用	<ul style="list-style-type: none"><li>下次考試成績會得幾分</li><li>最高氣溫預測冰品銷售量</li><li>最高氣溫與尖峰用電量</li><li>下一季的銷售額有多少</li><li>投入廣告費與銷售額</li></ul>	<ul style="list-style-type: none"><li>哪一個品種：依花的長寬分類</li><li>鐵達尼號船難者是否生還：依性別及艙等分類</li><li>判別是不是垃圾電子郵件</li><li>貓狗的識別</li><li>人臉、車牌等識別</li></ul>	<ul style="list-style-type: none"><li>班上同學分為跑得快跟跑得慢：依百米賽跑的秒數及身體的體脂肪率</li><li>哪些植物屬於相同的品種：依花的長寬分群</li><li>哪些動物屬於相同的品種：依體重及身長分群</li><li>哪些觀眾喜歡同一種類型的音樂或電影</li></ul>
常見的演算法	線性迴歸 複迴歸分析	KNN 決策樹 隨機森林 支援向量機	K-means

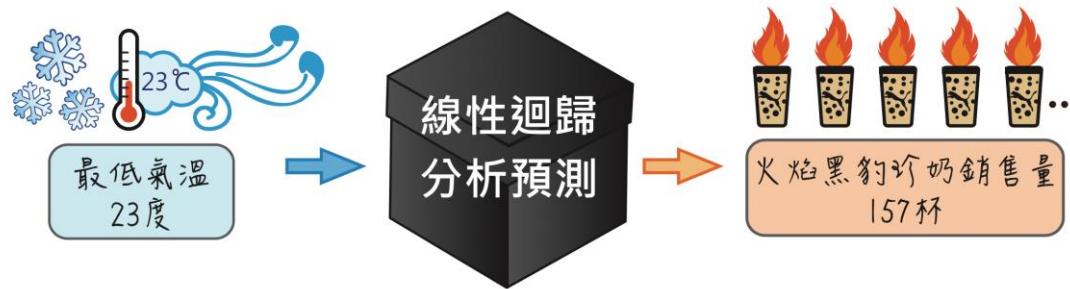
# 7-2-1 線性迴歸

- 趨勢預測
  - 由明天的氣溫可以推測熱珍奶的銷售量
  - 由產品廣告時數推測可能的銷售量等
- 趨勢預測可以使用統計學工具中的「**迴歸分析**」
- **線性迴歸** (Linear Regression)
  - 在座標上畫出一條**直線**，而這條直線可以代表資料點的變化趨勢



## 7-2-1 線性迴歸

- 線性迴歸模型想像成黑盒子，即使不知道它的內部如何運作，但是只要輸入氣溫，它就能輸出預測的銷售量

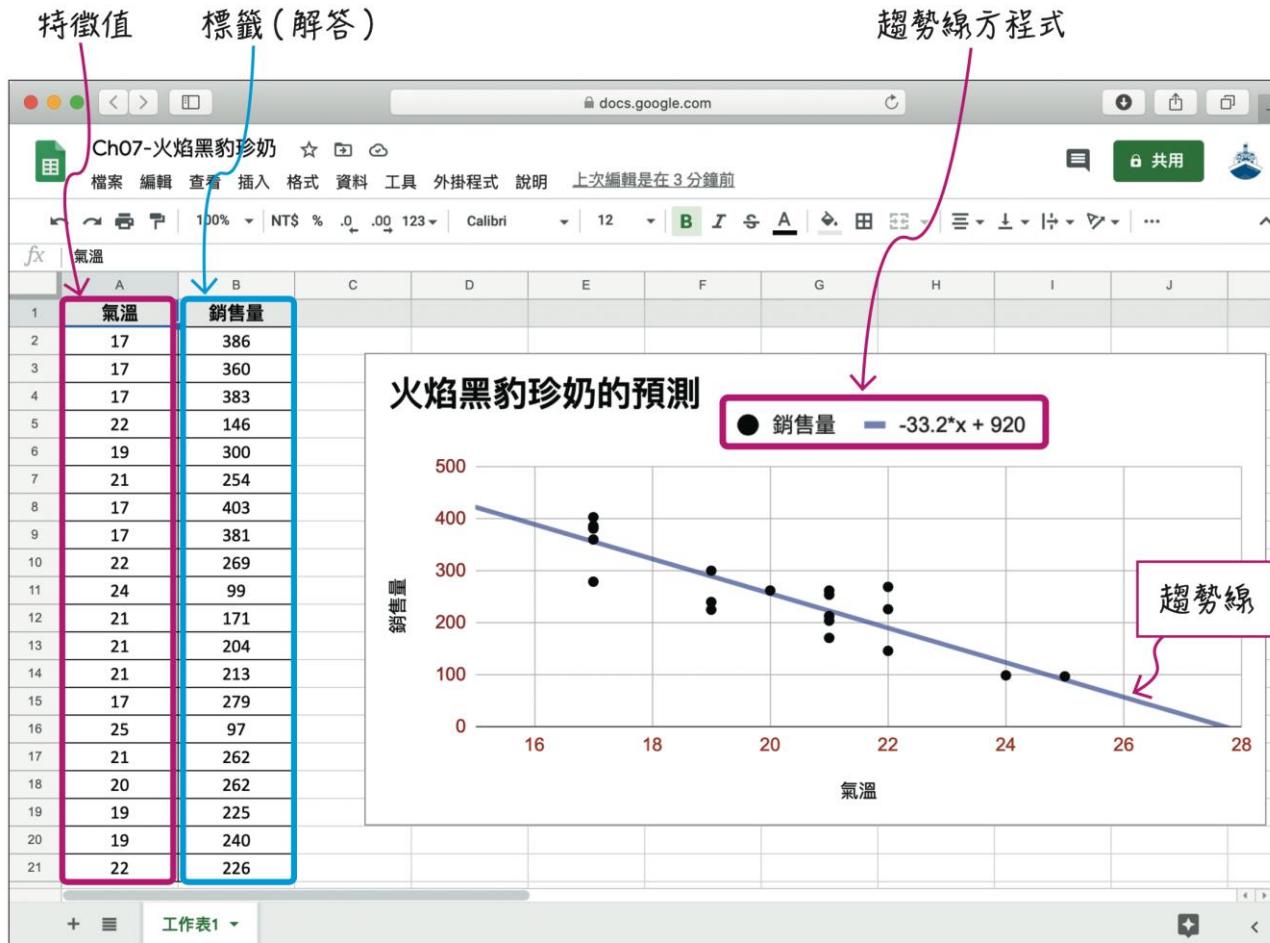


$$y = -33.2x + 920$$

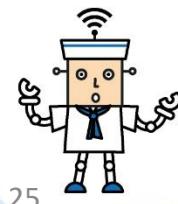
當氣溫為 $23^{\circ}\text{C}$ （即  $x = 23$ ）時，  
火焰黑豹拿鐵奶銷售量  $y = -33.2 \times 23 + 920 = 156.4$ （約157杯）

▲ 線性迴歸模型的應用





▲ 試算表軟體的線性迴歸功能



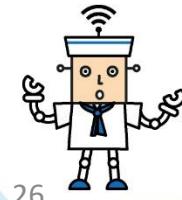
# 7-2-2 K- 最近鄰居法 (KNN) 做分類

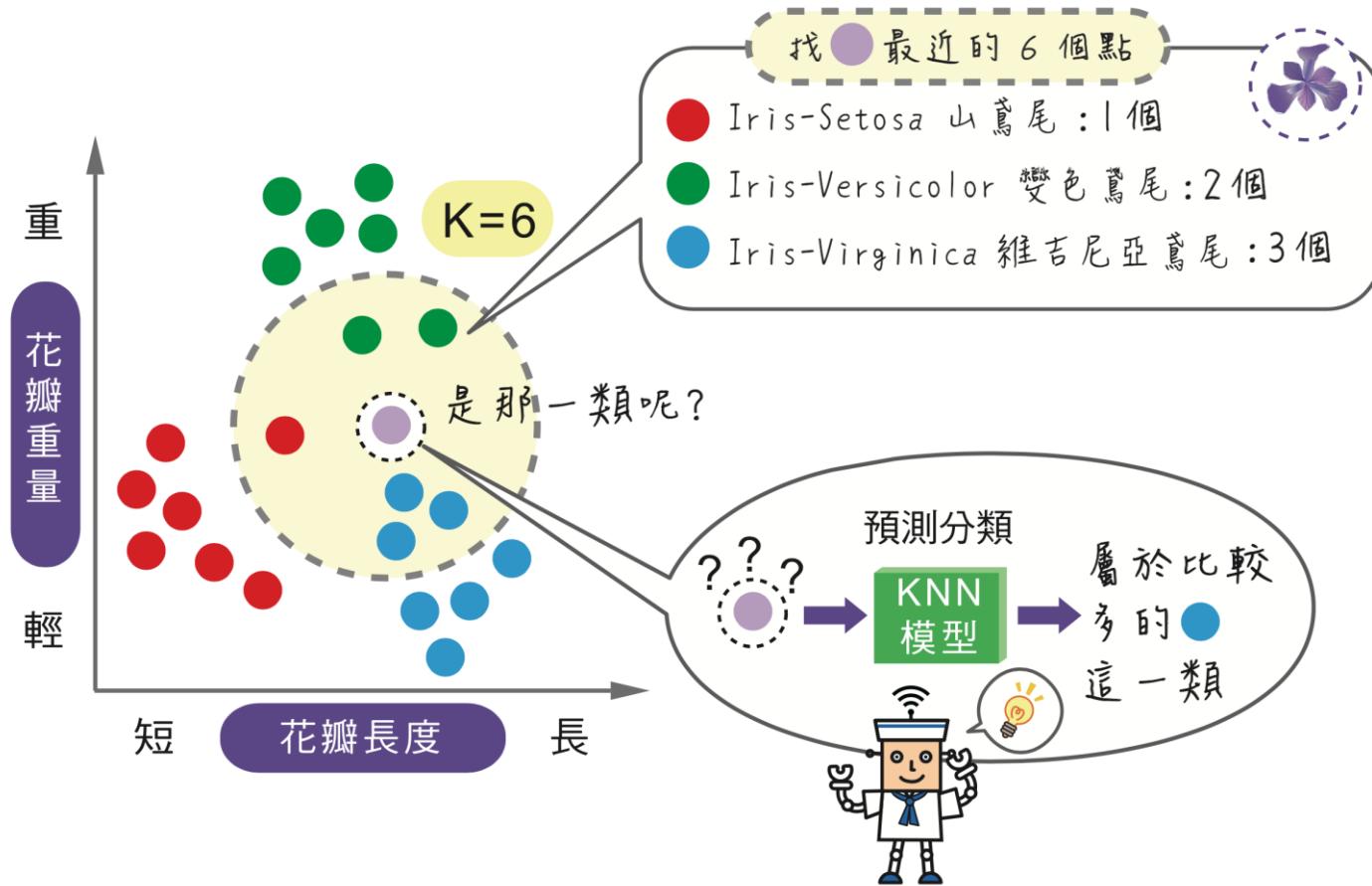
- 分類 (Classification)

- 例如：想分辨一張照片是貓或者是狗，最後的答案只能是這二種之中的其中一種
- 貓和狗就稱為標籤 (Label)，是我們在資料集中事先定義好的類型

- K- 最近鄰居法 (K Nearest Neighbor, KNN) 是分類常用的演算法

- 找「最近的 K 個鄰居」
- KNN 分類演算法運作的目標在於找出最鄰近的 K 個點，並透過「多數決」的方式決定該點屬於哪一類。





▲ 利用 KNN 模型分類及識別鳶尾花的類別

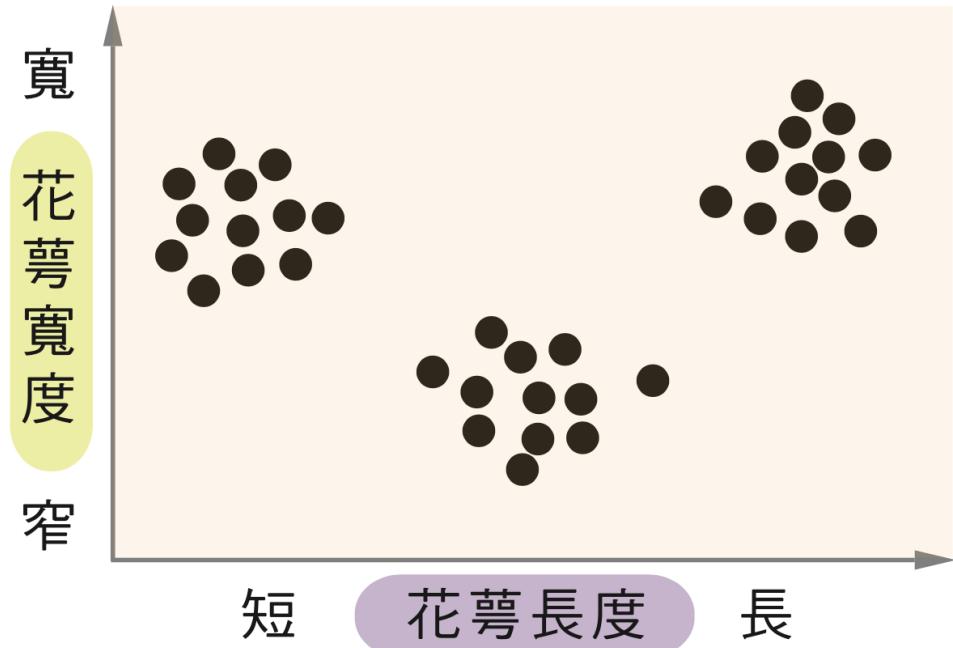


# 7-2-3 K- 平均法 (K-means) 做分群

- 分群 (Clustering)

- 把資料分成許多的群
- 與分類不同的是這些群都是我們事先沒有定義的，也就是沒有標籤的資料
- 例如：以「鳶尾花的花萼長度與寬度」為例，把所有資料以散佈圖做視覺化後，可以很容易的看出有 3 群，但是不知道每群所代表的是哪一種類別的鳶尾花





▲ 依花萼長度與寬度分為 3 群



# 資料科學 × 機器學習

實戰探索

Practical Exploration



# 7-2-3 K- 平均法 (K-means) 做分群

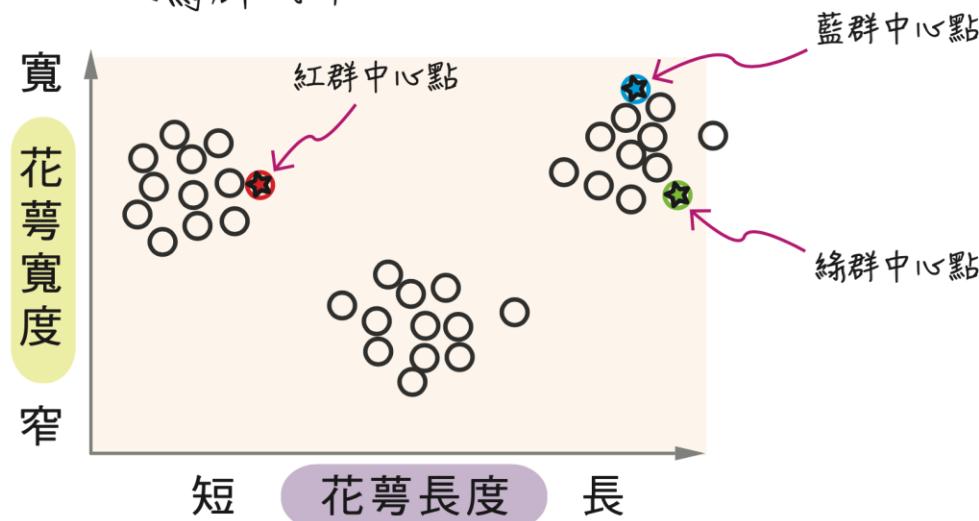
- K- 平均法 (K-means) 是知名的分群演算法，顧名思義為「依平均值 分 K 群」，也就是「找出與哪一個群的中心距離最近，再歸於該群」。假設  $K=3$  時，首先隨機從所有資料中挑出 3 個資料點做為群的中心，接著，根據初步分群的結果重新計算每群的中心位置，再重新做一次分群。以此類推



# 7-2-3 K- 平均法 (K-means) 做分群



1 隨機從所有資料中挑出3個資料點  
做為群的中心

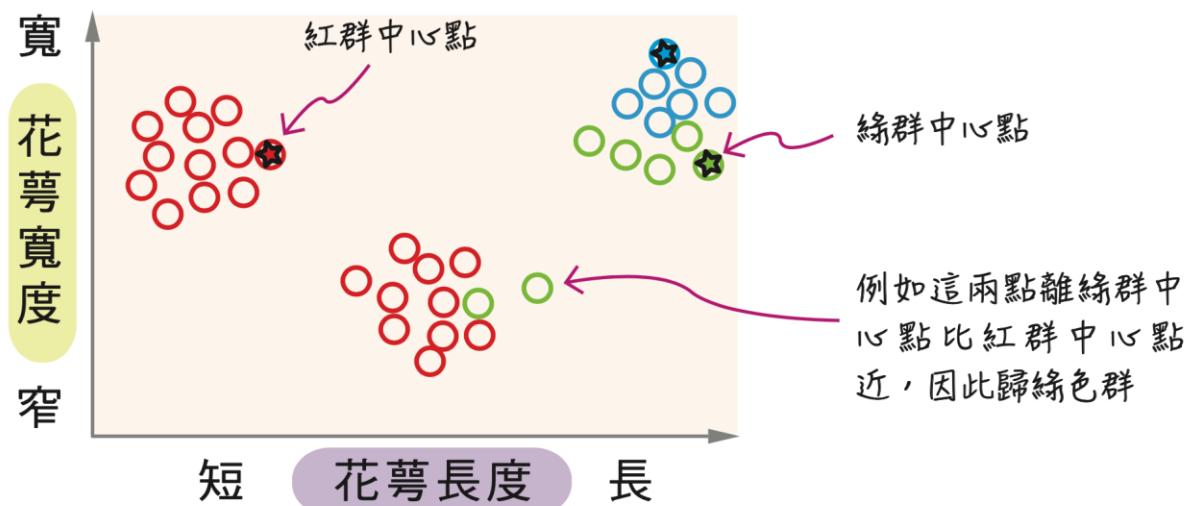


# 7-2-3 K- 平均法 (K-means) 做分群



2

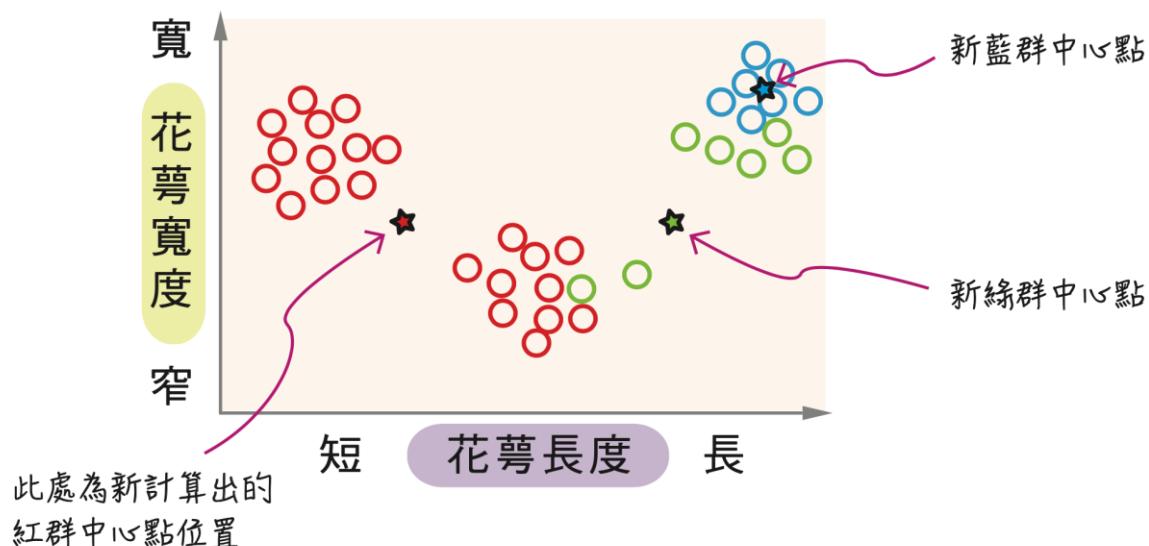
將所有的資料點分配給距離各中心點最近的群



# 7-2-3 K-平均法 (K-means) 做分群



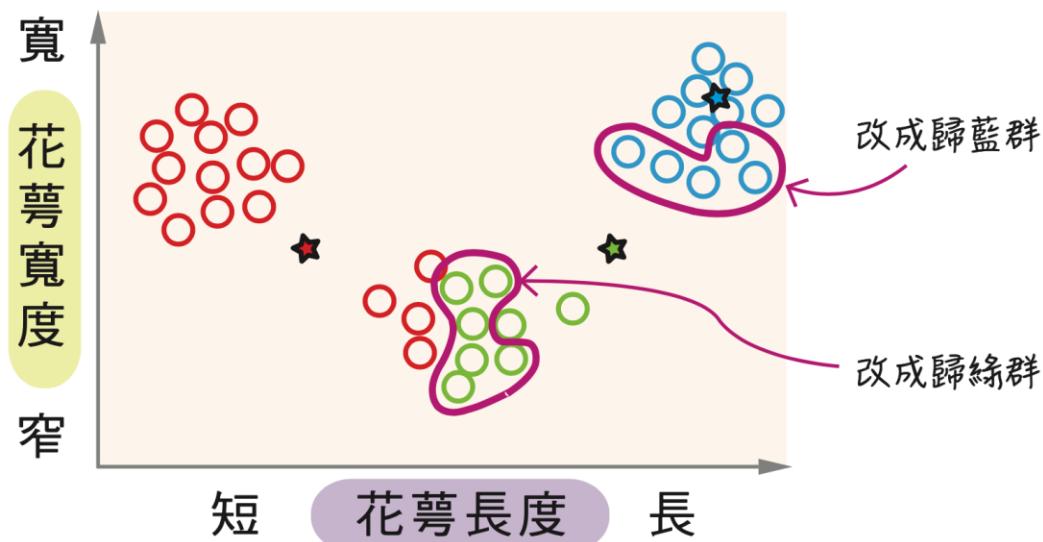
- 3 依據初步分群的結果重新計算  
每群的中心 (註：離各點距離總  
和最近者為中心點)



# 7-2-3 K-平均法 (K-means) 做分群



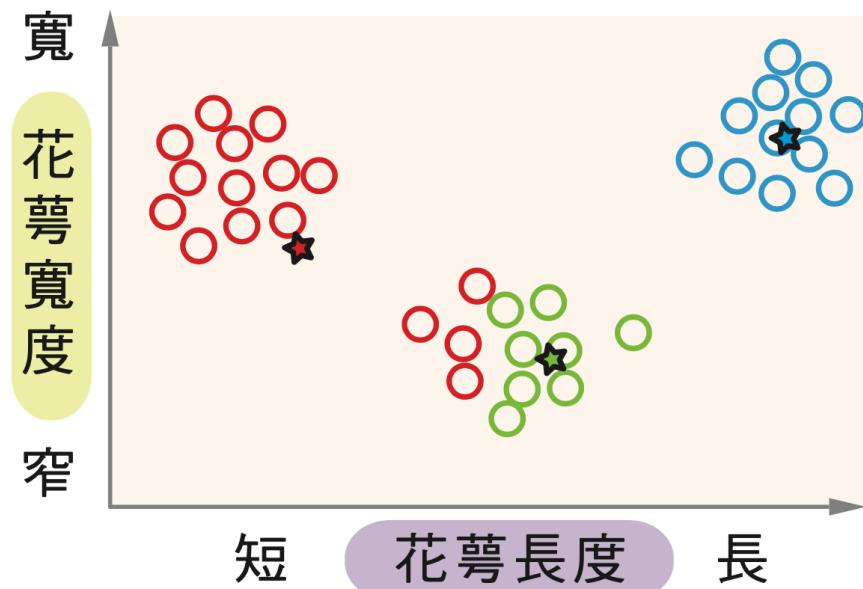
有了新中心點後，將所有的資料點重新分配給距離各中心點最近的群



# 7-2-3 K-平均法 (K-means) 做分群

5

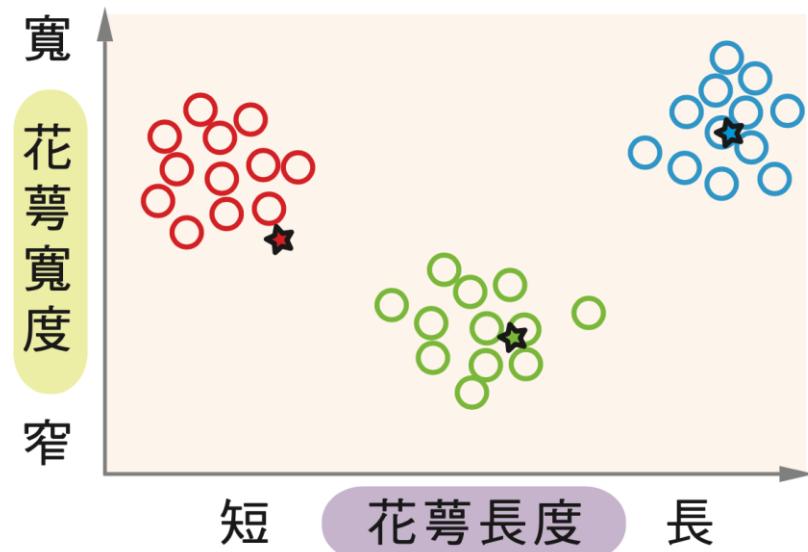
依據分群的結果再次重新計算  
每群的中心



# 7-2-3 K- 平均法 (K-means) 做分群

6

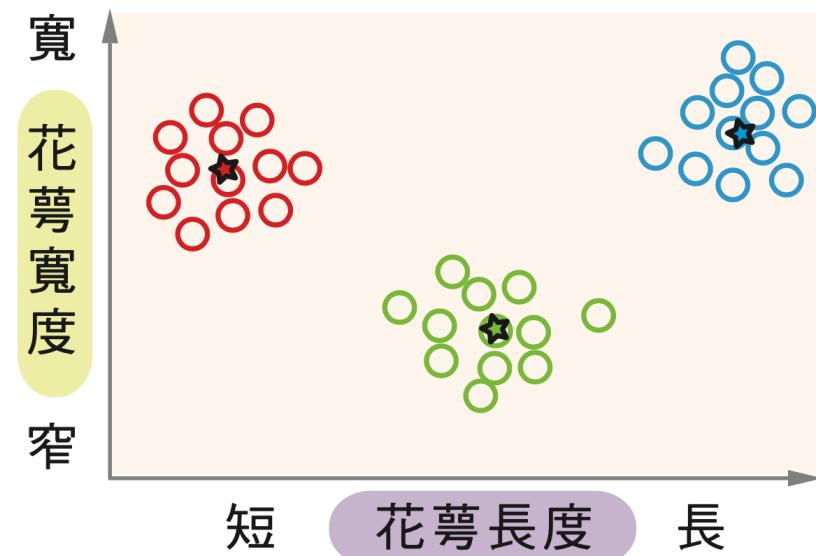
將所有的資料點再次分群(註：一樣，若離某新中心點更近，就改歸入該群)



# 7-2-3 K-平均法 (K-means) 做分群



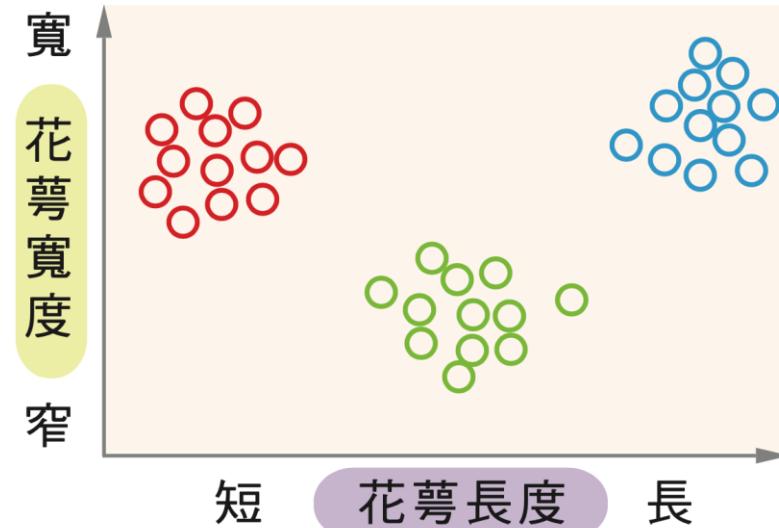
依據分群的結果再次  
重新計算每群的中心  
並再次分群，一直循  
環下去



# 7-2-3 K- 平均法 (K-means) 做分群

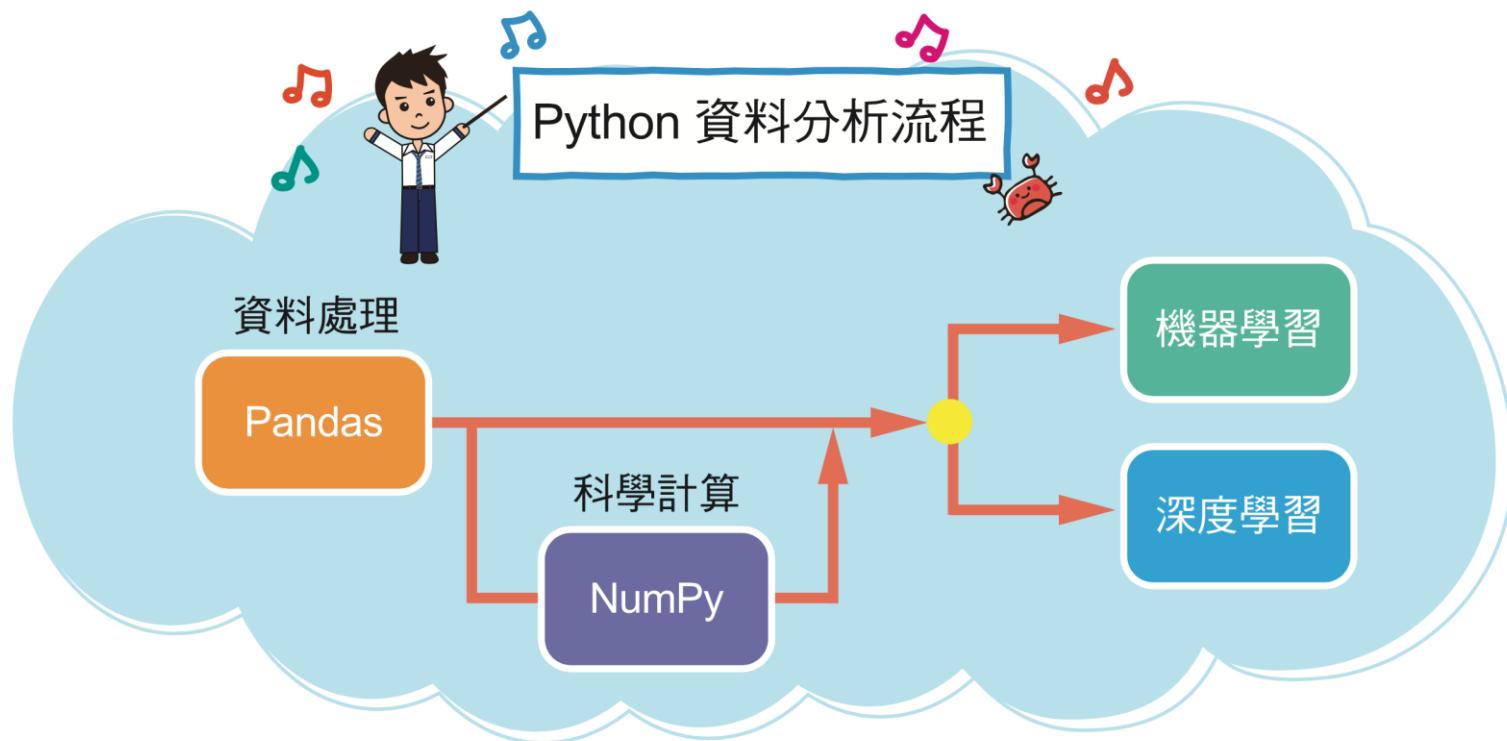
8

當所有群的中心不再有  
太大的變動，即每個資  
料點所屬的群已經穩  
定，就完成分群



## 7-3

# 實作機器學習的好工具



▲ 以 Python 進行機器學習的實作



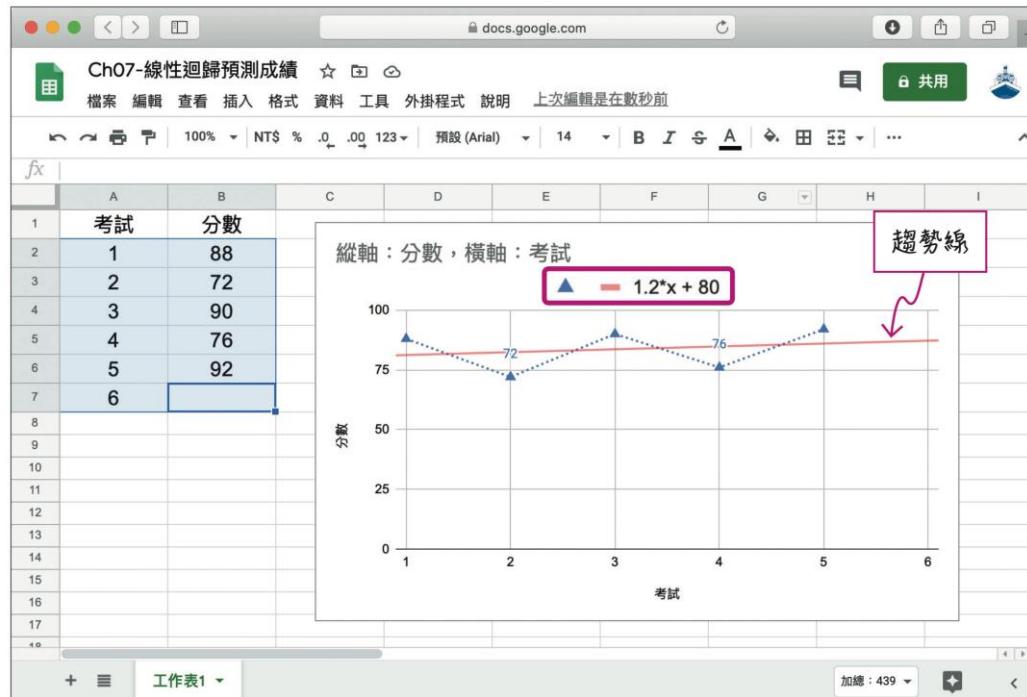
## 7-3-2 機器學習的神器 — sklearn 套件

---

- **sklearn** (全名 scikit-learn) 是一個常被運用於資料科學和機器學習的 Python 套件，包含
  - 線性迴歸
  - KNN
  - K-means
- 內建一些耳熟能詳的資料集，例如：
  - Titanic (鐵達尼號)
  - Iris (鳶尾花)
  - Digits (手寫數字辨識) 等。

# 使用試算表

- 前五次考試成績資料為  $(1, 88)$ 、 $(2, 72)$ 、 $(3, 90)$ 、 $(4, 76)$ 、 $(5, 92)$ ，使用試算表軟體可以求出如下圖的  $y=1.2x+80$  這條趨勢線



▲ 在試算表軟體中，以五次考試成績產生趨勢線來預測下一次的成績

# 使用 Python 來實作 (六行程式碼)

```
X 為特徵值          y 為標籤 (解答)
```

```
1 from sklearn.linear_model import LinearRegression      #挑選線性迴歸模型
2 lm = LinearRegression()                                #建立新模型 lm
3 X = [[1], [2], [3], [4], [5]]                          #指定特徵值為第1,2,3,4,5次考試
4 y = [ 88, 72, 90, 76, 92]                            #指定標籤為各次的分數
5 lm.fit(X, y)                                         #學習訓練
6 print('第6次考試分數:', lm.predict([[6]]))            #進行預測
```

進行訓練與測試，  
最後決定模型

根據模型進行預測

```
⇒ 第6次考試分數: [87.2]
```

▲ 以 Python 實作機器學習做成績預測

