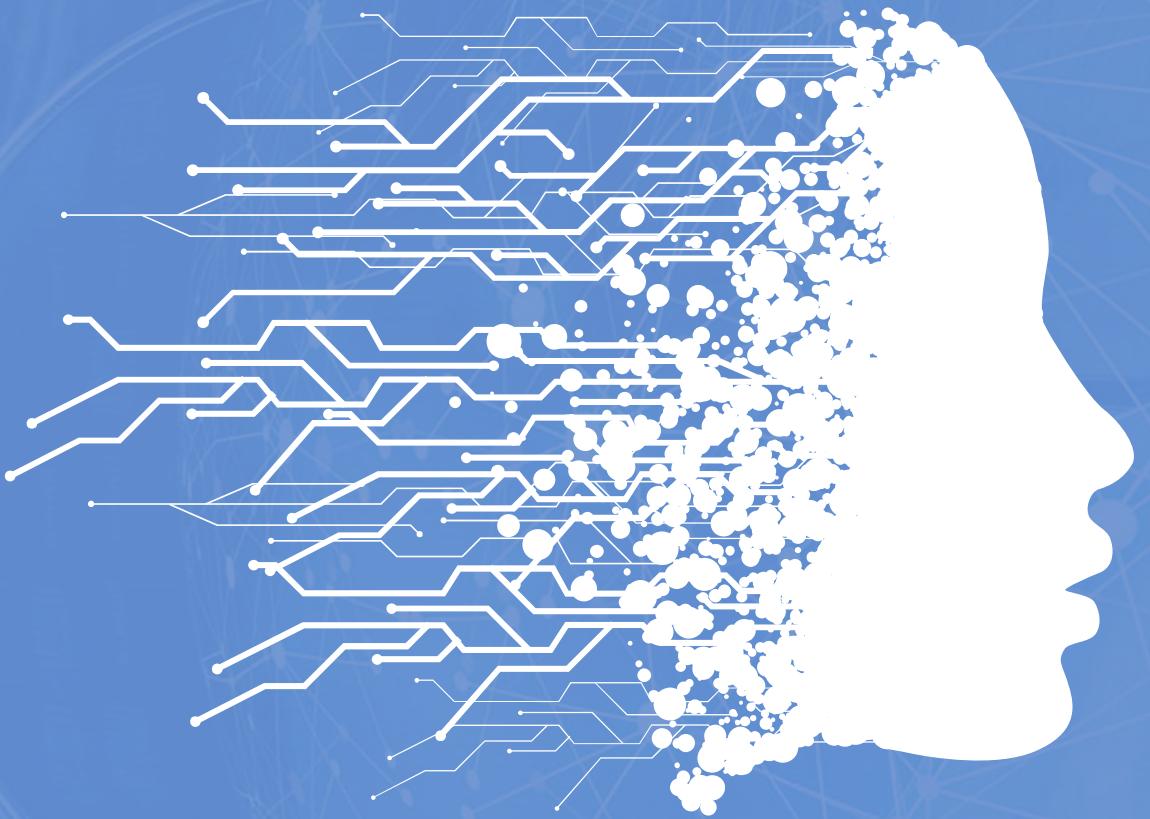


AI



紀老師程式教學網

<https://www.facebook.com/teacherchi>



機器學習

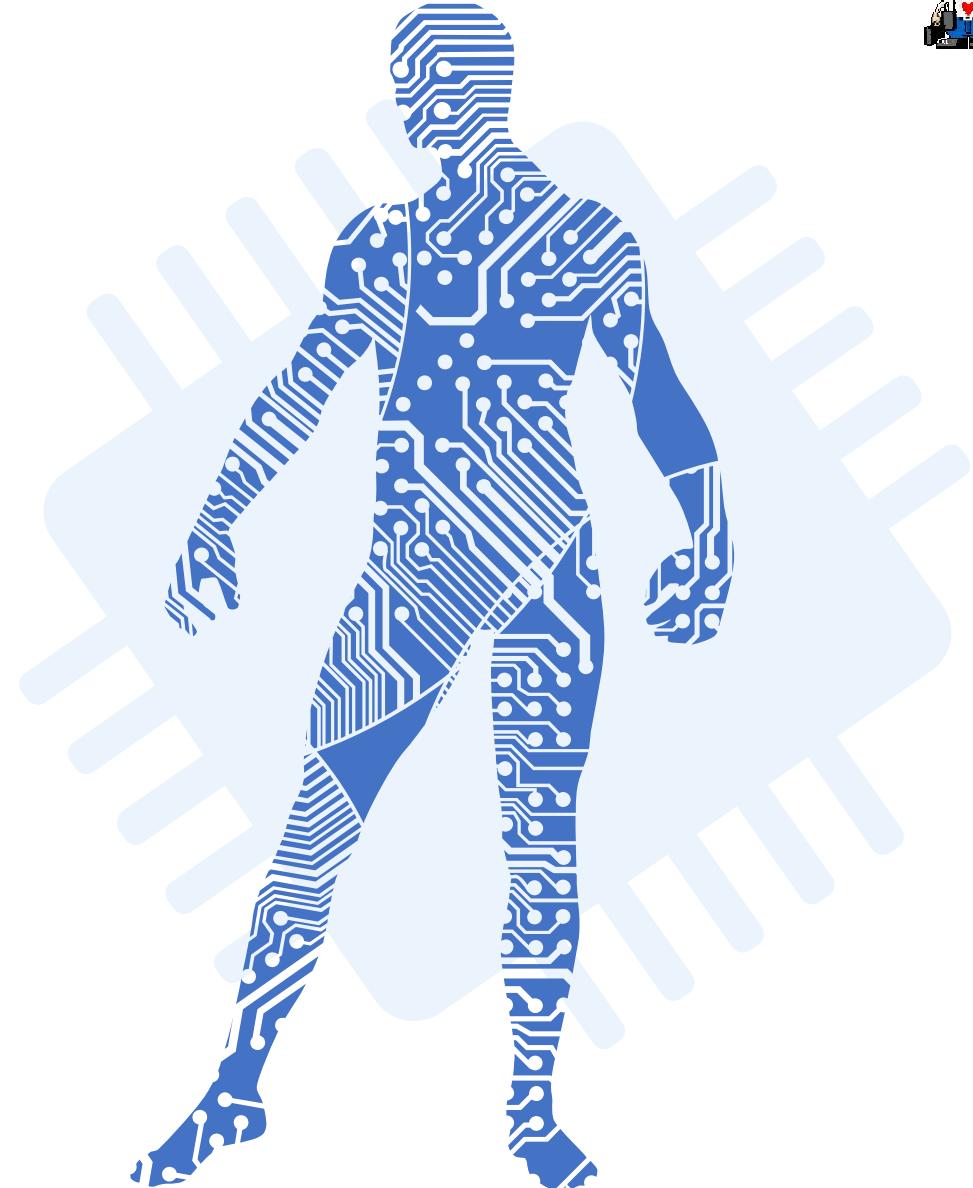
第1章 機器學習簡介

講師：紀俊男



本章大綱

- 什麼是機器學習程式
- 人工智能簡史
- 人工智能各領域的關係
- 機器如何學習
- 機器學習要會的數學基礎
- 機器學習解題手法種類
- 機器學習的應用領域

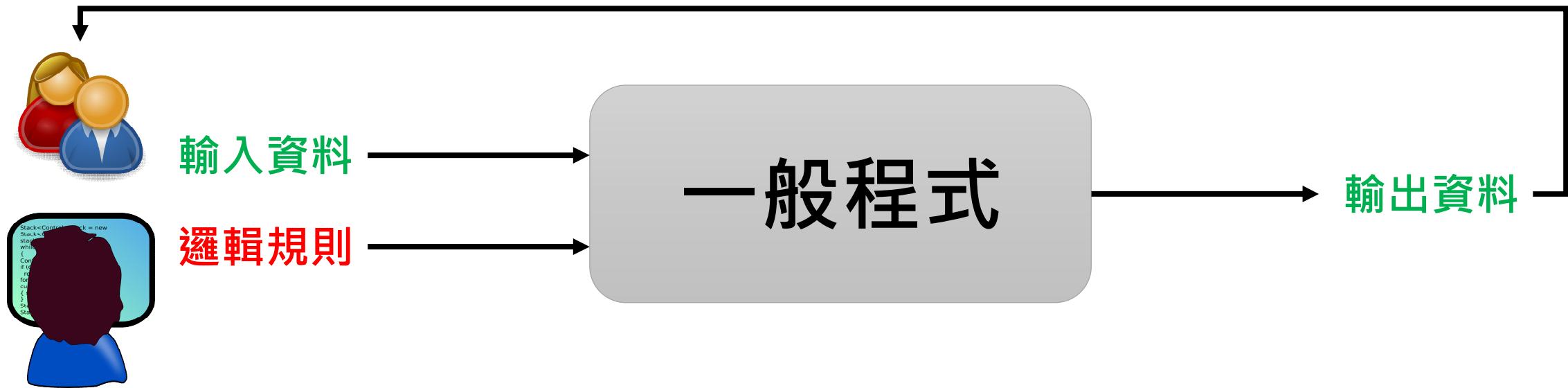




什麼是機器學習程式

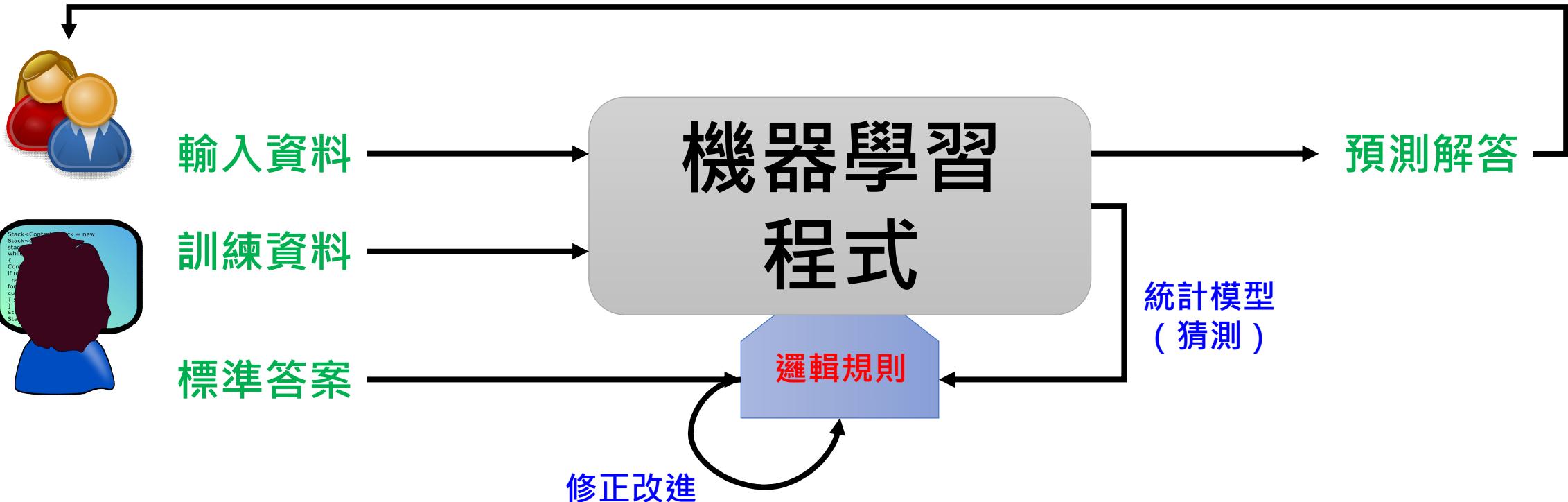


一般程式的運作原理





機器學習程式運作原理





重點整理

- 一般程式
 - 程式師：撰寫**邏輯規則**
 - 使用者：**輸入資料→輸出資料**
- 機器學習程式
 - 程式師：**訓練資料、標準答案、統計模型→邏輯規則**
 - 使用者：**輸入資料→預測解答**





人工智慧簡史





第一次人工智慧興起 (1956~1974)



• 達特茅斯會議 (1956/08/31)



John McCarthy



Marvin Minsky



Nathaniel Rochester



Claude Shannon

• 與會人員

- 約翰 · 麥卡錫 (John McCarthy)
 - 「人工智慧」一詞提出者
 - 「符號學派」之父
- 馬文 · 閔斯基 (Marvin Minsky)
 - 麻省理工學院人工智慧實驗室創始人
 - 「神經網路」之父
- 納撒尼爾 · 羅徹斯特 (Nathaniel Rochester)
 - IBM 701 設計者
- 克勞德 · 夏農 (Claude Shannon)
 - 「資訊理論」創始人

• 成就

- 正式提出「人工智慧」一詞
- 奠定「符號學派」在人工智慧的主流地位





第一次人工智慧寒冬 (1974~1980)



- 失敗原因

- 「符號推論」雖然好用，但不是萬能。

- 失敗事蹟

- 「機器跳棋」程式
 - 僅止於「州冠軍」。無法打敗人類高手。
- 「機器翻譯」成果
 - 一塌糊塗！「自動語言處理委員會」不願再投資。
- 「背景知識 (Frame Axiom)」不足
 - 「下雨 → 要帶傘」，但機器無法「感知」現在是否下雨。
- 「莫拉維克悖論 (Moravec's Paradox)」提出
 - 內容：人類無意識的直覺與感知，只需少少計算能力。但電腦用盡計算能力，卻無法產生任何直覺與感知。
 - 舉例：機會解微積分一類難題，無法辨識人臉這類易事。
- 「萊特希爾報告 (Sir James Lighthill's Report)」
 - 受「英國科學工程委員會」之邀，評估「人工智慧」到底能否能解決現實問題。
 - 紿出「沒有證據顯示，任何一個人工智慧領域，對人類進步做出重大承諾」答案。
 - 英國政府全面停止補助。人工智慧寒冬正式開始。





第二次人工智慧興起 (1980~1987)



- 興起原因
 - 「專家系統 (Expert System)」：嘗試以資料庫灌注大量背景知識
- 代表人物
 - 愛德華·費根鮑姆 (Edward A. Feigenbaum)
 - 專家系統之父
 - 認為「智慧」=「推論」+「知識」
- 實際成果
 - Dendral 專家系統
 - 可以用「質譜儀」資料，推論「分子結構」





第二次人工智慧寒冬 (1987~1993)



- 失敗原因
 - 「專家系統」的背景知識太龐大，窮究人力無法完整建立。
- 失敗事蹟
 - 日本「第五代電腦」計畫失敗
 - 日本產業經濟省撥款八億五千萬美元
 - 想打造「機械翻譯」、「理解圖像」、「如人類般推理」的機器
 - 因日本經濟泡沫，經費停止。引發各國經濟危機，人工智慧補助經費削減
 - 「仿生學派」的控訴
 - 認為「機器應該要有五感」、「背景知識應該自己學，不是人類輸入」
 - 說法受到認同。「專家系統」補助經費銳減。





第三次人工智慧崛起 (1997 ~)



- 興起原因
 - 「[神經網路](#)」證明效果顯著 + 「[硬體效能](#)」能跟上

- 代表人物
 - [麥可 · I · 喬丹 \(Michael I. Jordan \)](#)
 - 第一個把「[統計學](#)」引入「[神經網路](#)」的人
 - 「[神經網路](#)」起飛的重要貢獻人物
 - [傑佛瑞 · 辛頓 \(Geoffrey E. Hinton \)](#)
 - 提出「[自動編碼器 \(Auto Encoder \)](#)」理論
 - 能透過「[壓縮資料維度](#)」，產生「[新的概念](#)」
 - 讓電腦從一堆雜亂資料中，自動找出共同的[抽象意義](#)，變為可能
 - 「[深度學習](#)」之父，第三次人工智慧浪潮起飛的重要貢獻人物



Michael I. Jordan



Geoffrey E. Hinton

- 實際成果
 - 1997：許峰雄的「[IBM 深藍 \(Deep Blue \)](#)」，打敗人類[西洋棋棋王](#)卡斯帕洛夫 (Garry Kasparov)
 - 2006：「[自動編碼器 \(Auto Encoder \)](#)」與「[蒙地卡羅樹搜尋 \(Monte Carlo Tree Search \)](#)」理論提出
 - 2011：人工智慧程式「[華生](#)」，參加 [Jeopardy!](#) 益智問答節目，打敗所有人類，獲得冠軍
 - 2011：[Google 大腦計畫](#)，在看了 1000 萬張照片後，自動產生了「貓」這個概念。
 - 2017：[黃士傑](#)主導開發的 [AlphaGo](#) 圍棋程式，以三戰全勝的戰績，打敗人類[圍棋棋王](#)柯潔。
 - 2017：[AlphaGo Zero](#) 會自我廝殺，互相教導。40 天後，產生比 AlphaGo 還強的版本。





第三次人工智慧寒冬會來臨嗎？

- **有可能**！許多 AI 學者開始為文，表達質疑立場
 - Filip Piekniewski : 「[AI Winter is Well on its Way](#)」 (2018)
 - Gary Marcus : 「[In defense of skepticism about deep learning](#)」 (2018)
- 第三次人工智慧的隱憂
 - 相信「深度學習」，是人工智慧的**最終解藥**
 - 1956 年，「符號推論」也被視為人工智慧最終解藥
 - 1980 年，「專家系統」也被視為人工智慧最終解藥
 - 「深度學習」做出來的程式 / 模型，是**黑盒子**
 - 它可以運作，但人類不知道它如何運作的
 - 若發生問題，或想要優化，會無從著手
 - 「深度學習」對於「**開放性問題**」做得不太好
 - 對於 19x19 路、變化總數有限的圍棋，做得很好！
 - 對於「自動駕駛」，這種變化總數無上限的事情，做得不如人意！





小節整理



- 第一次人工智慧浪潮 (1956)
 - 崛起：「符號推論」
 - 没落：背景知識不足
- 第二次人工智慧浪潮 (1980)
 - 崛起：「專家系統」
 - 没落：輸入背景知識，曠日廢時
- 第三次人工智慧浪潮 (1997)
 - 崛起：「神經網路」 + 「統計學」 + 「自編碼理論」 = 「深度學習」
 - 没落：？？？（黑盒子？開放性問題？）

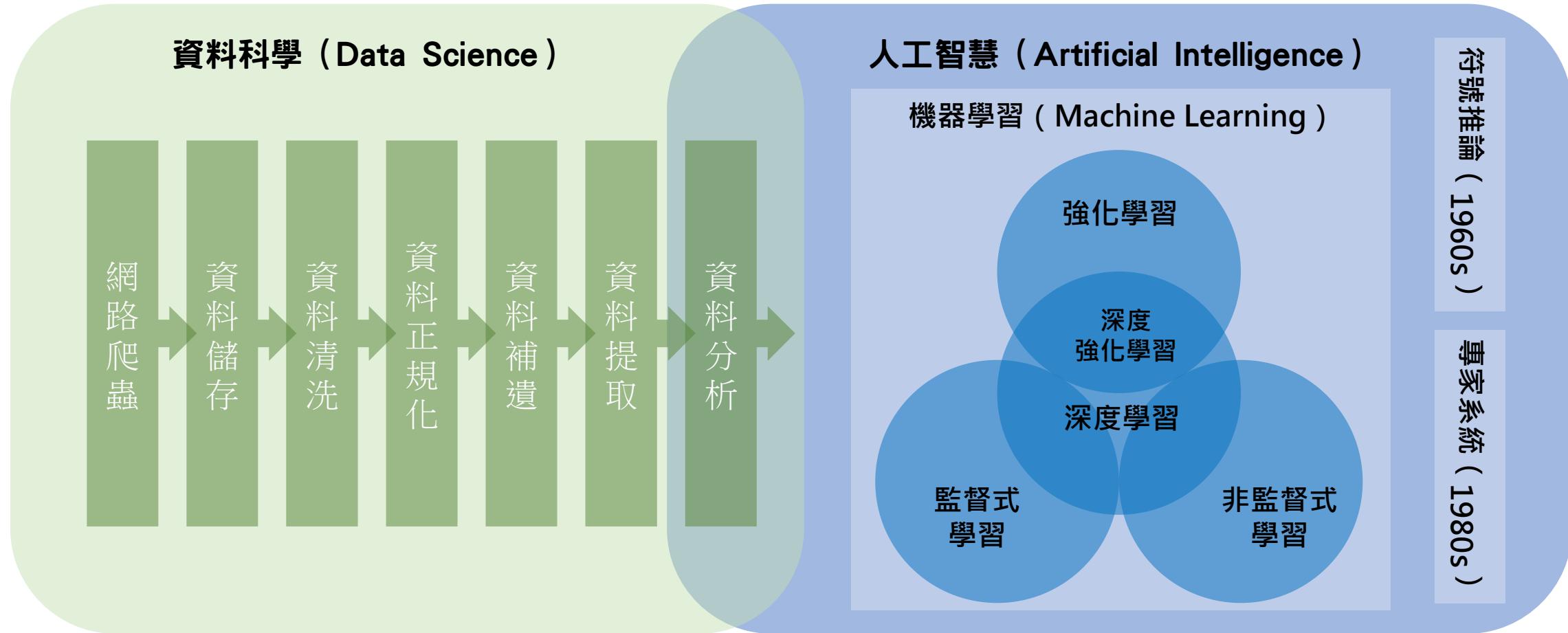




人工智慧
各領域的關係



「人工智慧」有哪些領域





機器學習的演算法有哪些



機器學習

監督式學習

非監督式 學習

迴歸

分類

集群

線性
迴歸

神經
網路

邏輯
迴歸

單純
貝氏

支援
向量
機

決策
樹

隨機
森林

神經
網路

K-平
均法

神經
網路





機器 vs. 深度 vs. 強化學習的比較



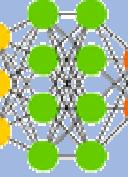
機器

A	B	C	D	E
1. City	Children	Age	Salary	ToBuy
2. Taipei	0	27	48000	Yes
3. Taichung	0	30	48000	No
4. Kaohsiung	1	30	58000	No
5. Kaohsiung	2	40	58000	Yes
6. Taipei	2	35	58000	No
7. Taichung	1	45	78000	Yes
8. Taipei	2	30	78000	No
9. Kaohsiung	1	35	68000	No
10. Taipei	2	50	68000	Yes
11. Taipei	2	37	67000	Yes

數位化資料

A	B	C	D	E
1. City	Children	Age	Salary	ToBuy
2. Taipei	0	27	72000	Yes
3. Taichung	0	30	48000	Yes
4. Kaohsiung	1	30	58000	No
5. Kaohsiung	2	40	58000	Yes
6. Taipei	2	35	58000	No
7. Taichung	1	45	78000	Yes
8. Taipei	2	30	78000	No
9. Kaohsiung	1	35	68000	No
10. Taipei	2	50	68000	Yes
11. Taipei	2	37	67000	Yes

特徵值



機器學習演算法
(迴歸、分類、集群)

深度

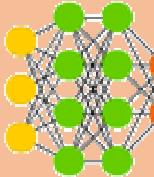


On Monday, Simon, a large red cat, by rights, acquires the Public Estate. Simon has been considering buying an apartment for his wife and son. He has agreed to sell his apartment to his wife without setting a price. His wife has agreed to keep the apartment's value at 40% above its original value. Simon's wife has agreed to keep the apartment's value at 40% above its original value. Simon's wife has agreed to keep the apartment's value at 40% above its original value.



A	B	C	D	E
1. City	Children	Age	Salary	ToBuy
2. Taipei	0	27	72000	Yes
3. Taichung	0	30	48000	Yes
4. Kaohsiung	1	30	58000	No
5. Kaohsiung	2	40	58000	Yes
6. Taipei	2	35	58000	No
7. Taichung	1	45	78000	Yes
8. Taipei	2	30	78000	No
9. Kaohsiung	1	35	68000	No
10. Taipei	2	50	68000	Yes
11. Taipei	2	37	67000	Yes

特徵值



深度學習演算法
(迴歸、分類、集群)

強化

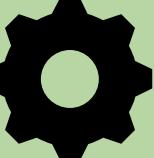


規則

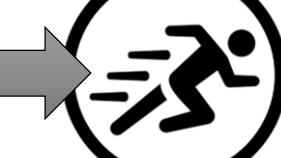


A	B	C	D	E
1. City	Children	Age	Salary	ToBuy
2. Taipei	0	27	72000	Yes
3. Taichung	0	30	48000	No
4. Kaohsiung	1	30	58000	Yes
5. Kaohsiung	2	40	58000	No
6. Taipei	2	35	58000	No
7. Taichung	1	45	78000	Yes
8. Taipei	2	30	78000	No
9. Kaohsiung	1	35	68000	No
10. Taipei	2	50	68000	Yes
11. Taipei	2	37	67000	Yes

環境
特徵值



強化學習演算法
(學習、推理、計畫)



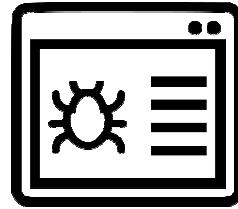
行動



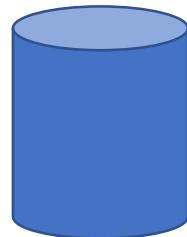
機器如何學習



Step 1：收集資料



網路爬蟲



資料庫

特徵向量1
特徵向量2
特徵向量3
特徵向量4
⋮
⋮

	A	B	C	D
1	國別	年齡	薪資	是否購買
2	France	44	72000	No
3	Spain	27	48000	Yes
4	Germany	30	54000	No
5	Spain	38	61000	No
6	Germany	40		Yes
7	France	35	58000	Yes
8	Spain		52000	No
9	France	48	79000	Yes
10	Germany	50	83000	No
11	France	37	67000	Yes

自變數 應變數
 $購買 = a \cdot (\text{國別}) + b \cdot (\text{年齡}) + c \cdot (\text{薪資})$ $a, b, c = \text{權重}$

特徵 = Features

特徵矩陣



Step 2：資料前處理 - 缺失資料補足



	A	B	C	D
1	國別	年齡	薪資	是否購買
2	France	44	72000	No
3	Spain	27	48000	Yes
4	Germany	30	54000	No
5	Spain	38	61000	No
6	Germany	40	63777.78	Yes
7	France	35	58000	Yes
8	Spain	38.78	52000	No
9	France	48	79000	Yes
10	Germany	50	83000	No
11	France	37	67000	Yes

- 刪除

- 缺失資料佔比不大時可用

- 平均

- 以該特徵值的平均為值

- 中位數

- 以該特徵值的中位數為值

- 最常出現數字

- 以該特徵值中，最常出現數字為值

- 在統計上又稱為眾數 (mode)





Step 3：資料前處理 - 類別資料數位化



- 「類別資料」無法計算 → 需要「數位化」

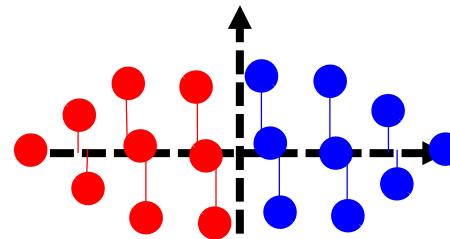
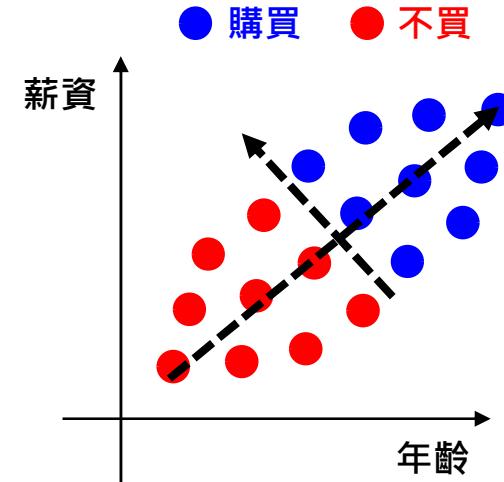
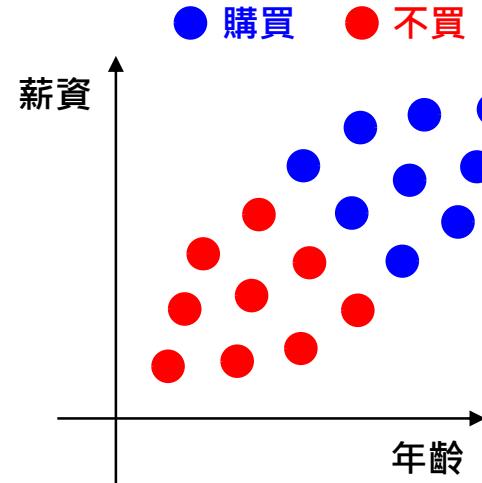
	A	B	C	D
1	國別	年齡	薪資	是否購買
2	France	44	72000	No
3	Spain	27	48000	Yes
4	Germany	30	54000	No
5	Spain	38	61000	No
6	Germany	40	63777.78	Yes
7	France	35	58000	Yes
8	Spain	38.78	52000	No
9	France	48	79000	Yes
10	Germany	50	83000	No
11	France	37	67000	Yes

	A	B	C	D
1	國別	年齡	薪資	是否購買
2	1	44	72000	0
3	2	27	48000	1
4	3	30	54000	0
5	2	38	61000	0
6	3	40	63777.78	1
7	1	35	58000	1
8	2	38.78	52000	0
9	1	48	79000	1
10	3	50	83000	0
11	1	37	67000	1





Step 4：資料前處理 - 降維 (選做)



往新 X 座標投影

僅剩一個維度，資訊量不變

- 降維原理 (以 PCA 法為例)

- 購買者 = 「年輕+有錢」 or 「年長+小康」
- 以「**最大方差法**」找最適合座標
- 所有資料點往**新 X 座標**投影
(此乃線性代數之「**座標變換**」)
- 降維成功

- 降維的好處

- 減低運算量，增加計算速度
- 以人類可理解的 2D/3D 圖描繪出結果來

- 降維的方法

- 特徵選擇法：**去除**若干影響力不強的特徵
 - 反向淘汰法、卡方檢定法...
- 特徵提取法：將影響力不強的特徵**合併**
 - 主成分分析 (Principal Components Analysis) 法





Step 5：資料前處理 - 切分訓練集、測試集

訓練集

測試集

	A	B	C	D
1	國別	年齡	薪資	是否購買
2	France	44	72000	No
3	Spain	27	48000	Yes
4	Germany	30	54000	No
5	Spain	38	61000	No
6	Germany	40	63777.78	Yes
7	France	35	58000	Yes
8	Spain	38.78	52000	No
9	France	48	79000	Yes
10	Germany	50	83000	No
11	France	37	67000	Yes

• 訓練集 (Training Set)

- 約佔 $2/3 \sim 3/4$
- 「影響因子」 + 「解答」
- 用於訓練機器學習模型之用

• 測試集 (Testing Set)

- 約佔 $1/3 \sim 1/4$
- 僅包含「影響因子」
- 用於測試機器學習模型有多好
- 測出來的答案會與真實答案對比

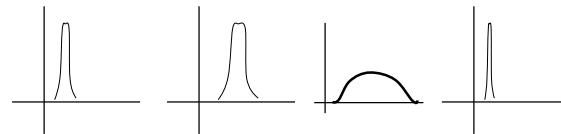


Step 6：資料前處理 - 特徵縮放 (選做)



特徵縮放前

國別	年齡	薪資	是否購買
1	44	72000	0
2	27	48000	1
3	30	54000	0
2	38	61000	0
3	40	63777.78	1
1	35	58000	1
2	38.78	52000	0
1	48	79000	1
3	50	83000	0
1	37	67000	1
平均值	1.9	38.778	63777.78
標準差	0.875595	7.253777	11564.1
			0.5
			0.527046



特徵縮放後

國別	年齡	薪資	是否購買
-1.02787	0.719901	0.711013	-0.94868
0.114208	-1.62371	-1.36438	0.948683
1.256289	-1.21013	-0.84553	-0.94868
0.114208	-0.10725	-0.24021	-0.94868
1.256289	0.168464	1.73E-07	0.948683
-1.02787	-0.52083	-0.49963	0.948683
0.114208	0.000276	-1.01848	-0.94868
-1.02787	1.271338	1.316334	0.948683
1.256289	1.547056	1.662233	-0.94868
-1.02787	-0.24511	0.27864	0.948683
平均值	0	1.97E-16	-5.3E-16
標準差	1	1	1
			0
			1

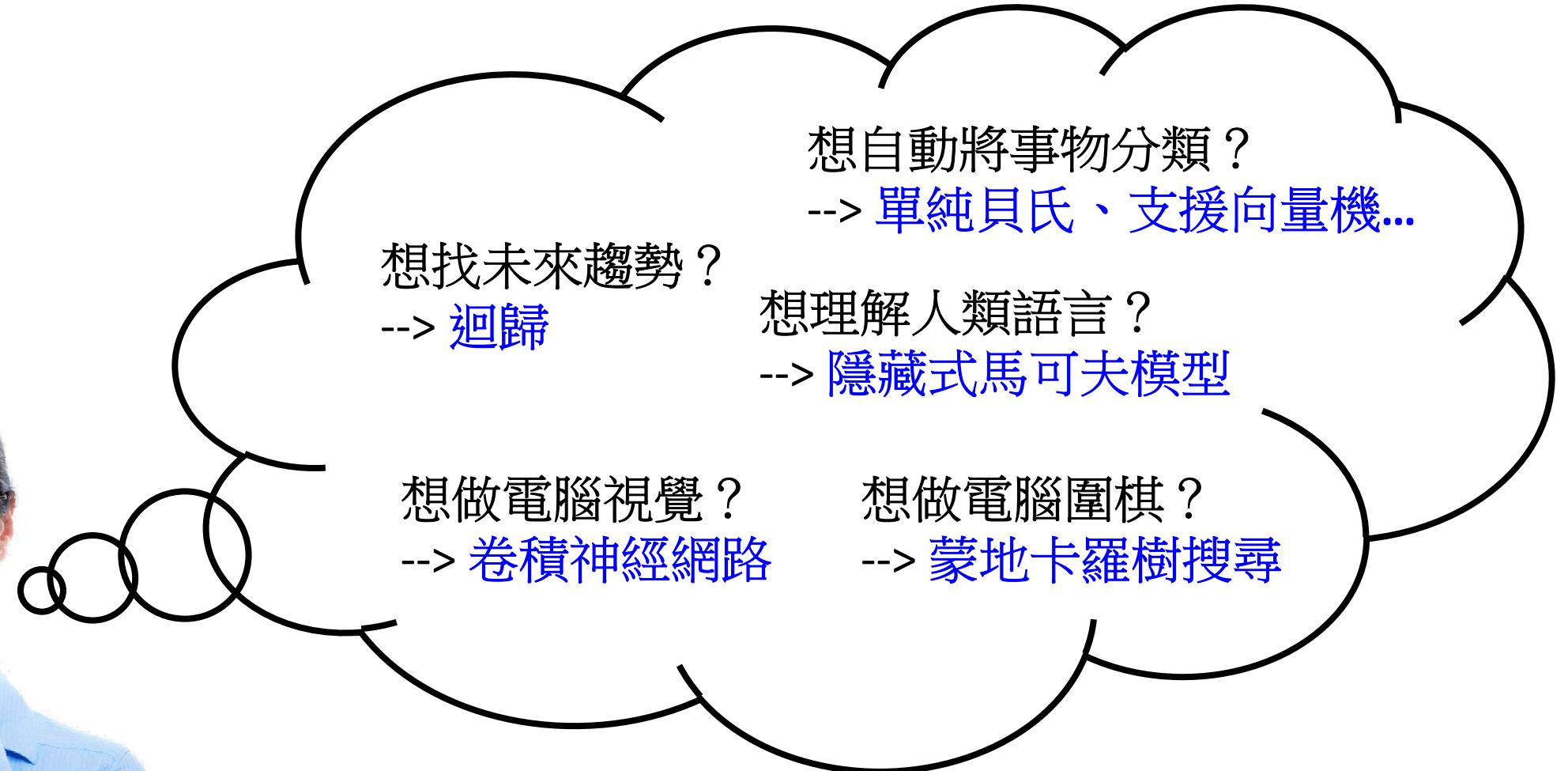


- 何謂「特徵縮放」 (Feature Scaling)
 - 讓各特徵影響力相當
 - 平均值=0、標準差=1
- 特徵縮放的好處
 - 各特徵影響力相等
 - 收斂至標準答案快
- 如何做特徵縮放

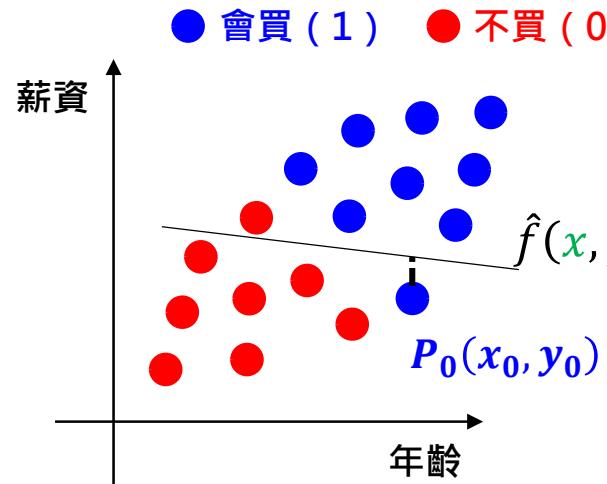
$$\hat{x} = \frac{x - Avg(x)}{Std(x)}$$



AI Step 7：選擇機器學習演算法



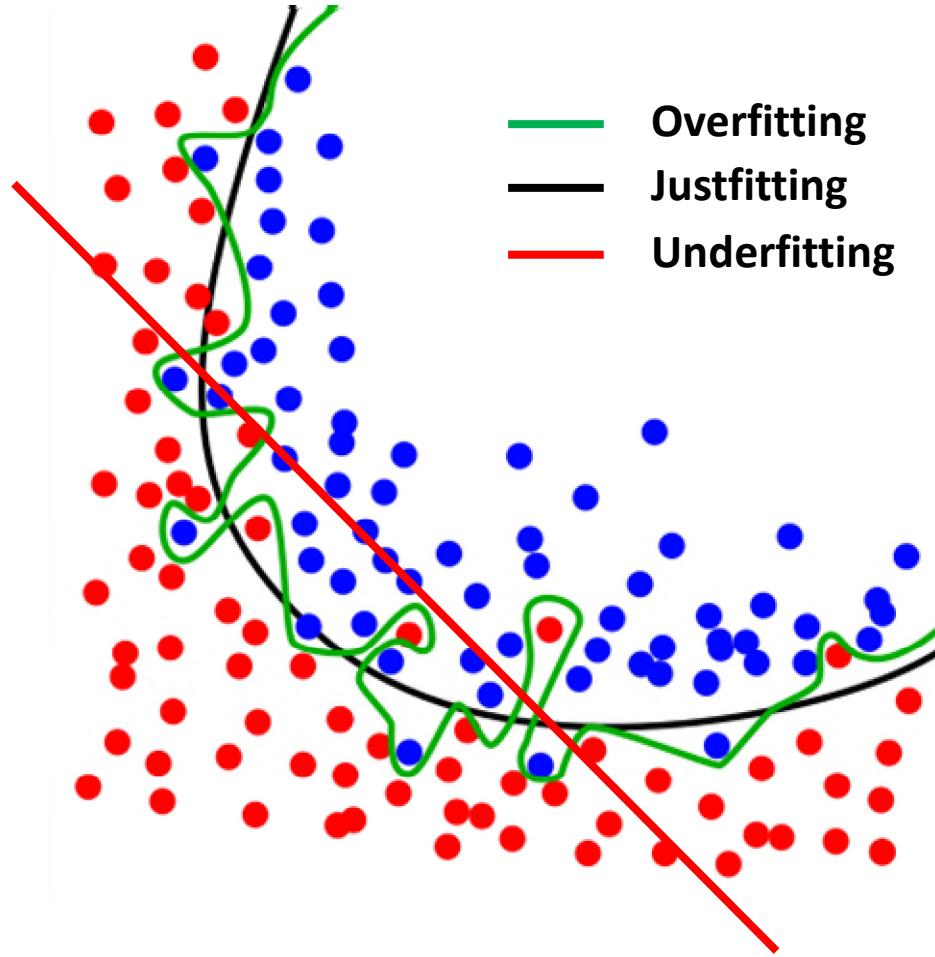
A Step 8 : 模型訓練



$$P_0 \text{ 位於線段} = \begin{cases} \text{左下：不買 } \hat{f}(x_0, y_0) = 0 \\ \text{右上：會買 } \hat{f}(x_0, y_0) = 1 \end{cases}$$

$$\text{誤差函數} = \sum_{k=0}^{n-1} |\hat{f}(x_0, y_0) - f(x_0, y_0)|$$

- 取出「訓練集」資料
 - 「殘差分析」(Residual Analysis)
 - 定義「誤差函數」
(或叫「損失函數」, Loss Function)
 - 計算各點落在哪邊 → 與真實答案比對
 - 「擬合」(Fitting)
 - 以「微分」/「偏微分」, 讓「誤差函數」有極小值
 - 又稱為「參數推估」(針對 a, b, c)
- $a \text{ 極小值} = \frac{\partial}{\partial a} \sum_{k=0}^{n-1} (\hat{f}(x_0, y_0) - f(x_0, y_0))^2$ $b \text{ 極小值} = \frac{\partial}{\partial b} \sum_{k=0}^{n-1} (\hat{f}(x_0, y_0) - f(x_0, y_0))^2$
- $c \text{ 極小值} = (\text{略})$



• 過擬合 (Overfitting)

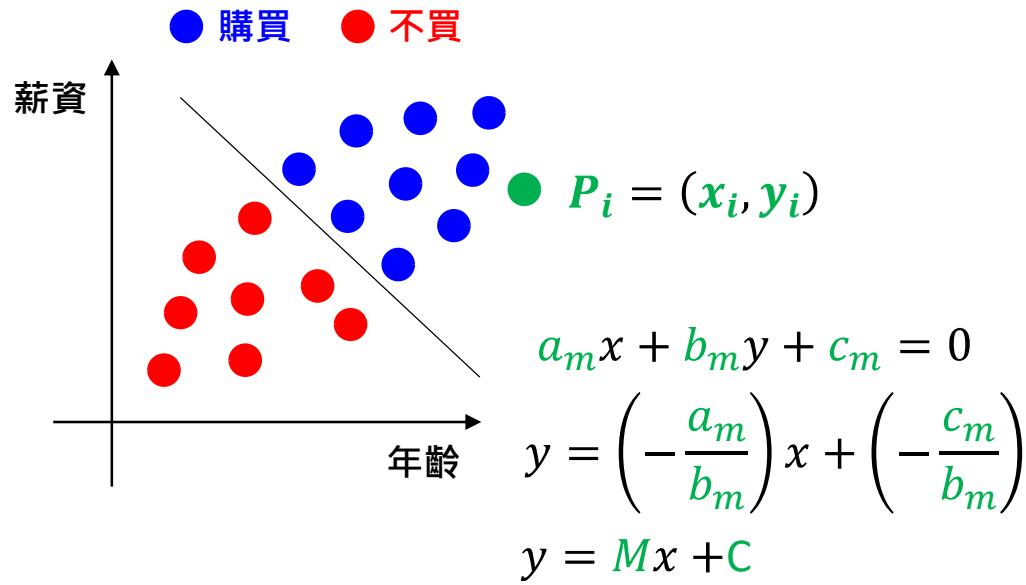
- 模型維度過高（特徵過多）
- 嘗試在訓練階段「取得高分」，但在實戰階段卻「一塌糊塗」。
- 如同考試準備時，學了太多刁鑽的題目，結果正式考試時，基本的題目卻答不出來。

• 欠擬合 (Underfitting)

- 模型維度過低（特徵過少）
- 在訓練時期偷懶，採用了較少的特徵訓練模型，只為了加快訓練速度。
- 如同考試準備時，只隨便翻了一下課本，結果正式考試時，基本的題目也答不出來。



Step 9：模型測試



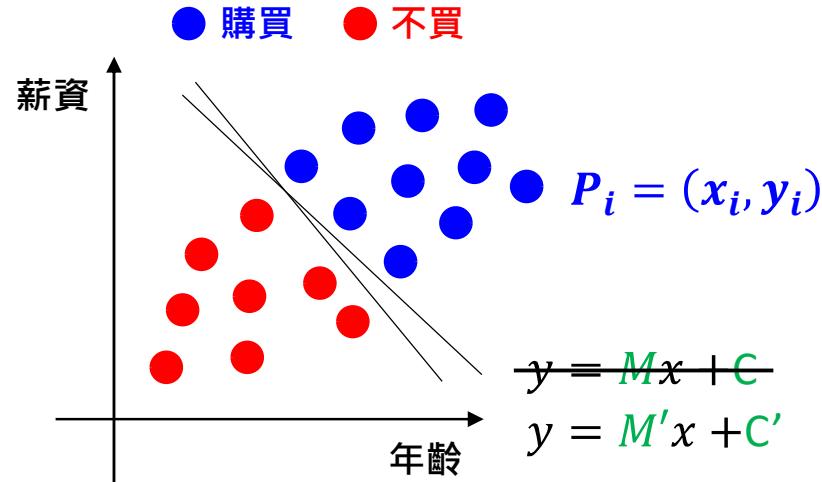
直線斜率	$y_i - Mx_i - C$ 結果	預測結果
負 ($M < 0$)	>0 --> 位於線段右側	會買
	<0 --> 位於線段左側	不買
正 ($M > 0$)	>0 --> 位於線段左側	不買
	<0 --> 位於線段右側	會買

- 將直線以「斜率型」表示
- 紿定一個新的資料點 P_i
 - 年齡 = x_i
 - 薪資 = y_i
- 根據模型 $y=Mx+C$ 預測結果
 - $M < 0$: 斜率是負的
 - $y_i - Mx_i - C > 0$: 位於線段右側
 - 預測結果 : 會買





Step 10：持續學習（Online Learning）



- 定義

- 機器學習模型 $y=Mx+C$ 製成後
- 新資料進來 → 預測 → 現實確認
- 根據「現實確認值」，重跑模型
- 得到修正後、更準的模型 $y=M'x+C'$

- 範例：Google 搜尋引擎

- 模型製成： $y=C_nX_n+C_{n-1}X_{n-1}+\dots+C_0$
- 新資料進來：使用者的關鍵字
- 預測：搜尋結果
- 現實確認：使用者點擊了哪些結果
- 修正模型： $y=C'_nX_n+C'_{n-1}X_{n-1}+\dots+C'_0$





小節整理：機器如何學習



- 資料擷取 (Data Acquisition)
 - 爬蟲、資料庫
- 資料前處理 (Data Preprocessing)
 - 缺失補足 (Missing Data Complement)
 - 類別資料數位化 (Categorical Data Digitized)
 - 降維 (Dimensionality Reduction)
 - 資料集切分 (訓練集、測試集) (Dataset Spliting)
 - 特徵縮放 (Feature Scaling)
- 模型製成 (Modeling)
 - 資料擬合 (Fitting)
 - 模型測試 (Testing)
- 持續改進 (Model Improvement)
 - 持續學習 (Online Learning)





**機器學習要會的
數學基礎**



會用於機器學習的數學基礎

- **函數**

- 自變數 (Independent Variables)、應變數 (Dependent Variables)
- 座標軸、函數圖形

- **幾何學**

- N 元一次方程式、與其圖形 (如：斜率、截距...)
- 空間點 --> 直線方程式距離、空間點位於直線的哪一側

- **線性代數**

- 矩陣運算：特徵值、特徵向量、特徵矩陣運算
- 座標變換：維度獨立性 (正交程度 Orthogonality)、降維、座標映射

- **微積分**

- 微分、偏微分：取某一維度的極值時使用

- **統計學**

- 各種統計量 & 意義：平均、中位數、標準差
- 統計分布：尤以「常態分布 (Normal Distribution)」最為重要

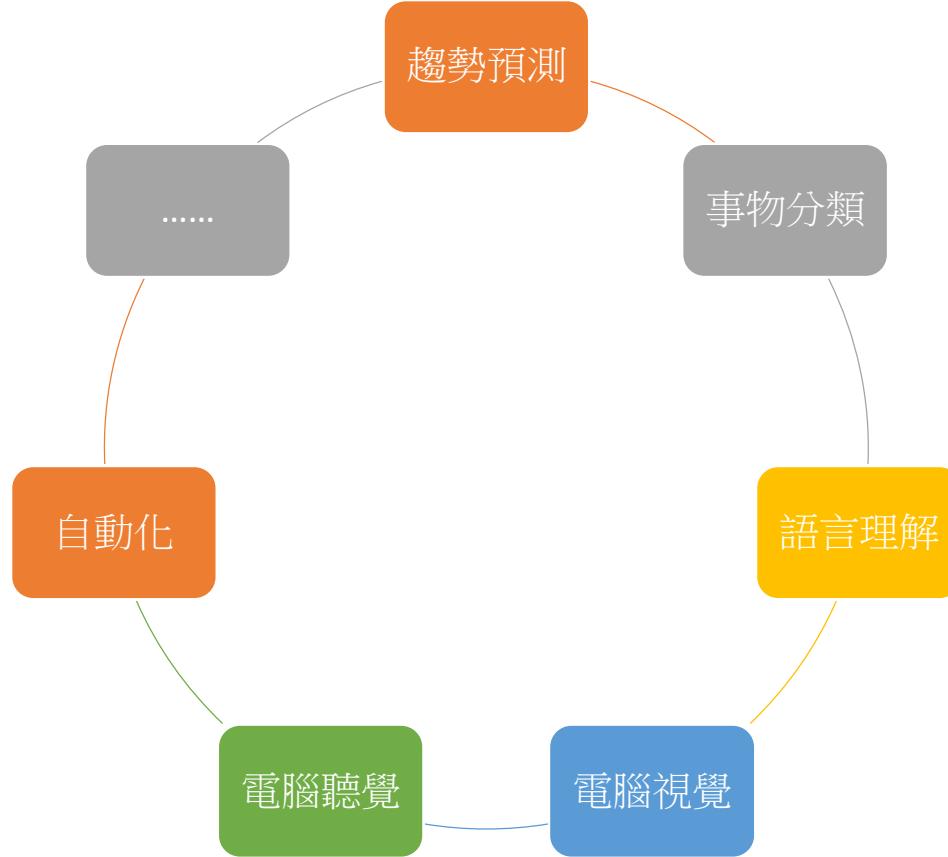




**機器學習
應用領域**



機器學習應用領域



- **趨勢預測**
 - 股票漲跌
- **事物分類**
 - 疾病診斷、購物喜好、垃圾郵件
- **語言理解**
 - 文章分類、自動翻譯、聊天程式
- **電腦視覺**
 - 臉部辨識、年齡推測、物體辨識
- **電腦聽覺**
 - 語音輸入、音樂生成
- **自動化**
 - 自動駕駛、機器人工學





本章總結

- 機器學習程式與一般程式有何不同
 - 一般程式：「邏輯規則」由程式師產生
 - 機器學習程式：「邏輯規則」由統計模型自動產生
- 人工智慧簡史
 - 已是「第三次人工智慧浪潮」，提防「第三次人工智慧寒冬」
- 人工智慧各領域關係
 - 大數據：提供機器學習的資料來源
 - 機器學習：下含各種演算法，如：神經網路、深度學習
- 機器如何學習
 - 資料獲取 --> 資料前處理 --> 模型訓練 --> 模型測試
- 機器學習三大演算法類別
 - 監督式學習：輸入 + 解答
 - 非監督式學習：輸入
 - 強化學習：自行摸索 + 獎勵
- 各種機器學習應用領域
 - 趨勢預測、事物分類、語言理解、電腦視覺、電腦聽覺、自動化

