

深度強化式學習 Ch1

Deep Reinforcement Learning

Alexander Zai, Brandon Brown



Outline



1-1 深度強化式學習中的『深度』

1-2 強化式學習

1-3 動態規劃 vs. 蒙地卡羅法

1-4 強化式學習架構

1-5 強化式學習有甚麼應用

1-6 為什麼要使用『深度』強化式學習

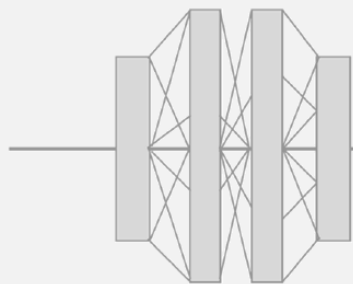
1-7 有用的說明工具-線圖(String diagram)

1-1 深度強化式學習中的『深度』



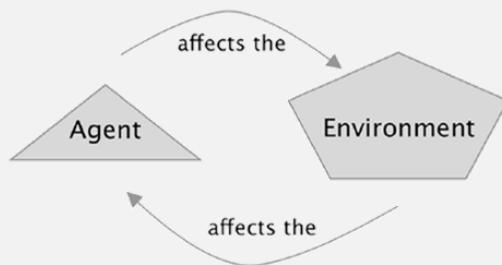
Deep Reinforcement Learning

深度學習
Deep Learning



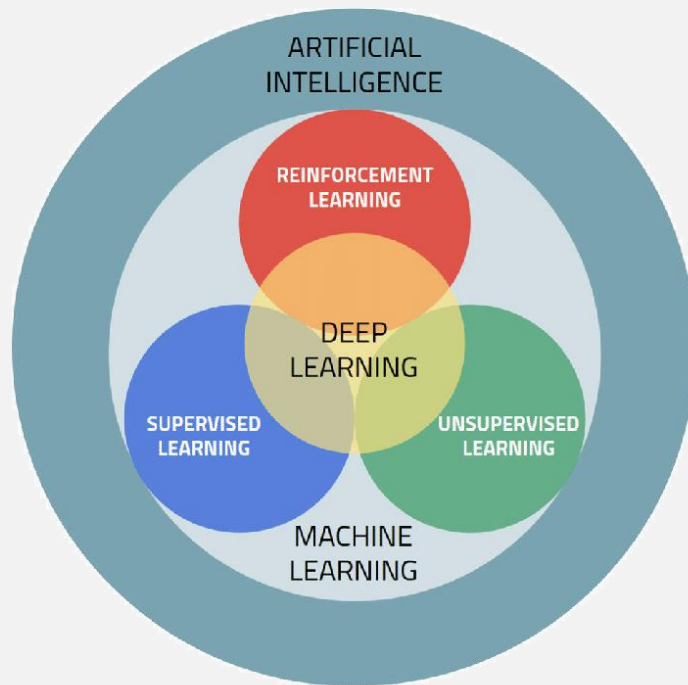
+

強化式學習
Reinforcement Learning



||

深度強化式學習
Deep Reinforcement Learning

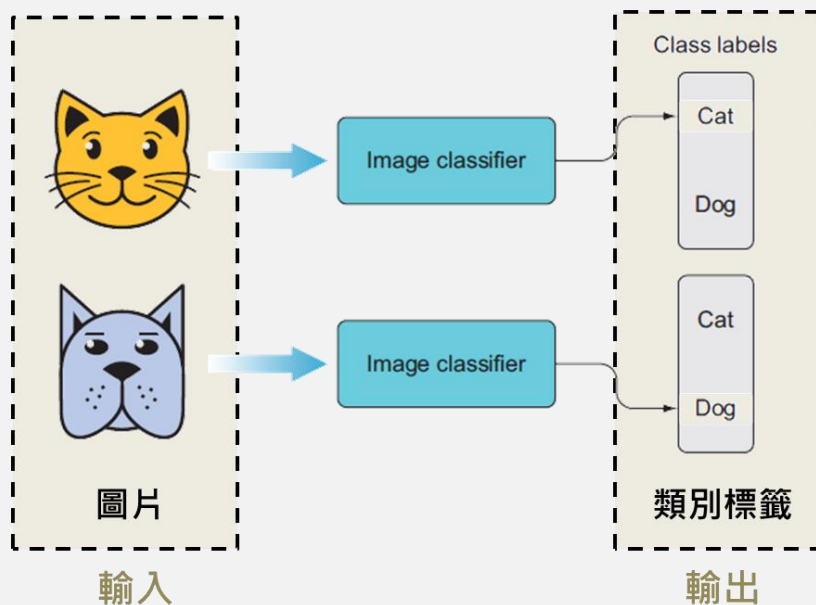


1-1 深度強化式學習中的『深度』

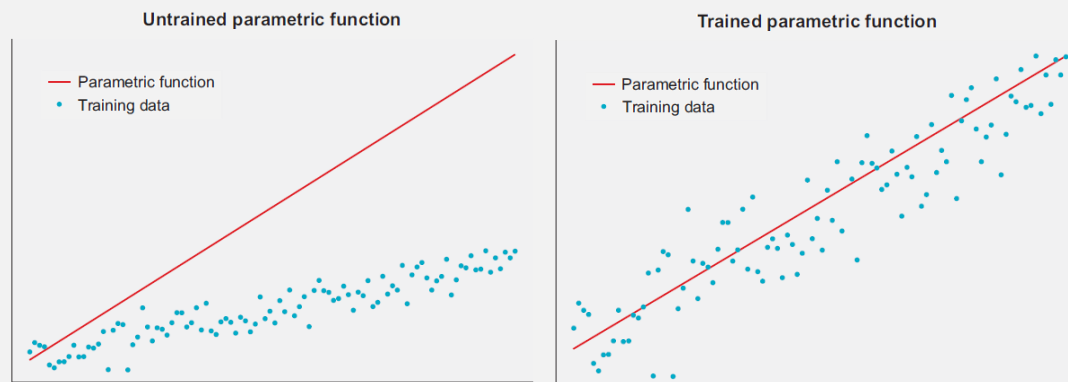


- 參數式模型 (Parametric Model)

✓ 圖片分類器 Image Classifier



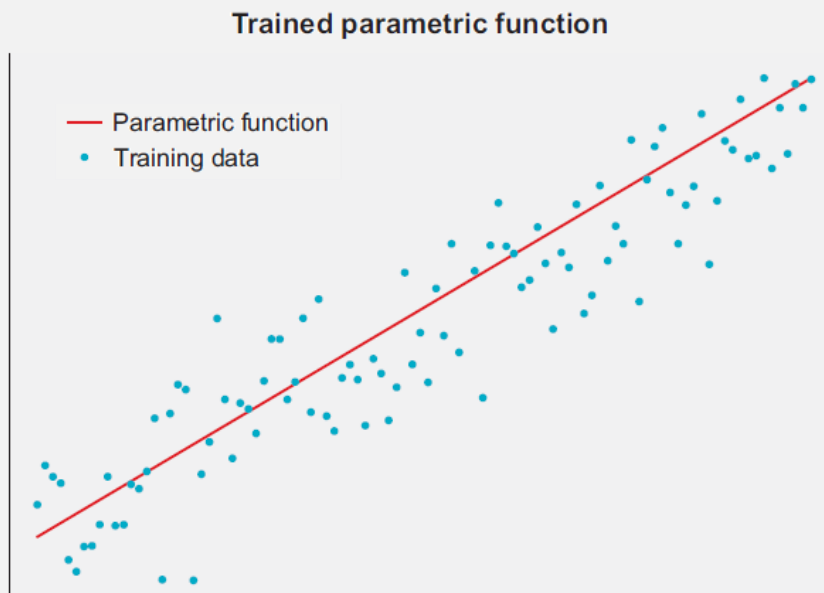
✓ 迴歸 Regression



1-1 深度強化式學習中的『深度』



- 參數式模型的參數 (Parameters)



參數

斜率slope 截距intercept

$$f(x) = \underline{m}x + \underline{b}$$

模型訓練

參數調整

模型表現越佳

1-1 深度強化式學習中的『深度』

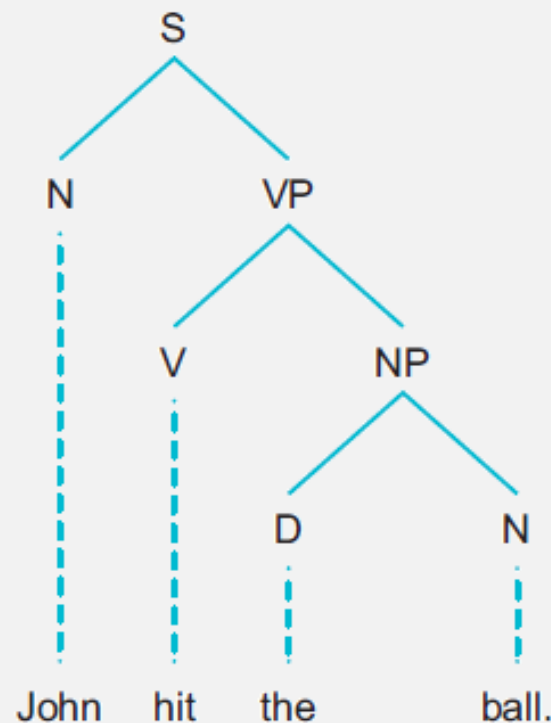


- 深度學習模型具有層的概念

- 複合性(Compositionality):

高層複雜的訊息是由簡單的訊息所組成

- 完整的句子 (S)
- 主詞名詞 (N)
- 動詞片語 (VP)
- 動詞 (V)
- 名詞短語 (NP)
- 定冠詞 (D)
- 名詞 (N)



- 句子由小單元組成

高

複雜程度

低

1-2 強化式學習



問題



cat

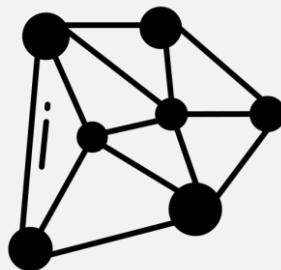


dog

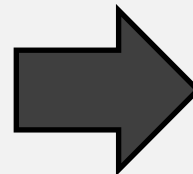
- 圖片分類



解決方法



- 深度學習演算法



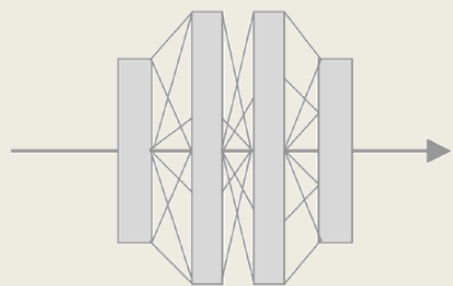
做的決策和行動

控制任務
(Control Tasks)

1-2 強化式學習

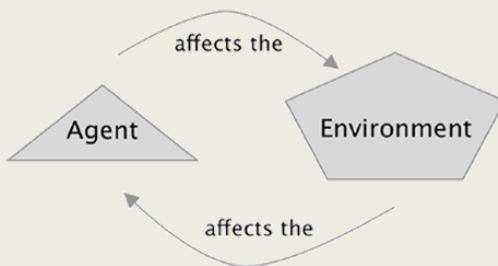
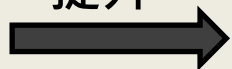


- 深度強化式學習的組成



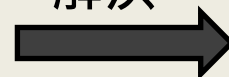
深度學習演算法
Deep Learning

提升



強化式學習演算法
Reinforcement Learning

解決



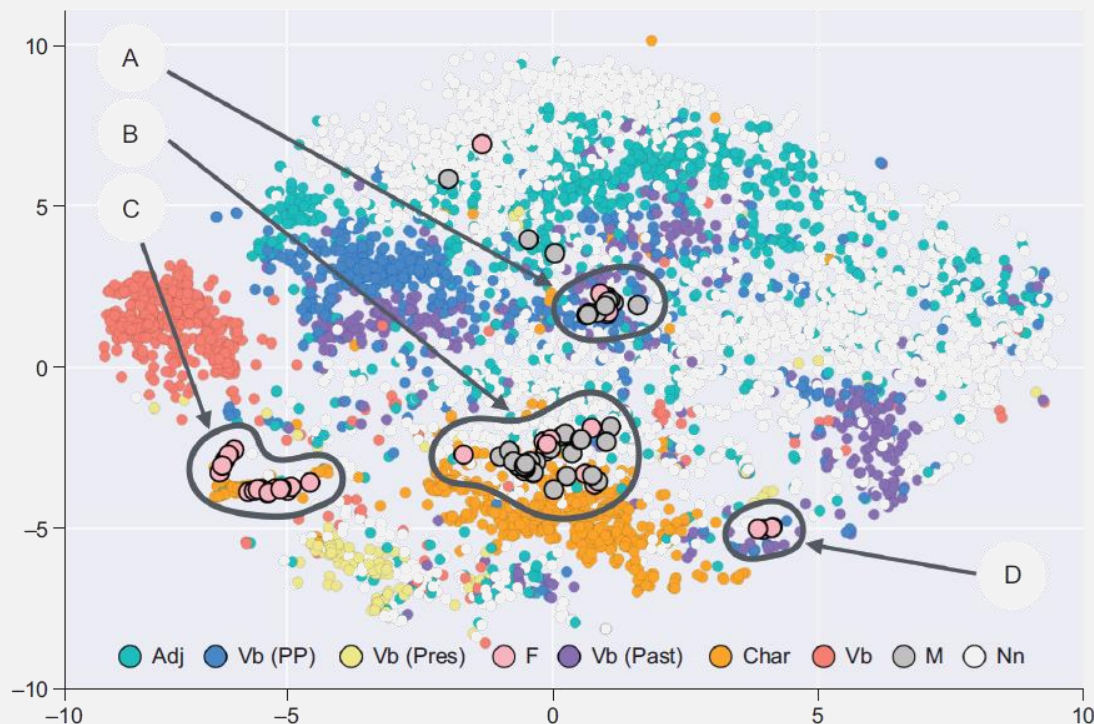
控制任務
Control Tasks

1-2 強化式學習



- 處理自然語言時的資料分布

不同詞性的單字相距較遠



相同詞性的單字在同一群

1-2 強化式學習



不需標籤資料



強化式學習只需知道

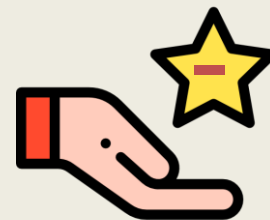


- 最終目標
- 避免做的事情

回饋機制



正回饋值



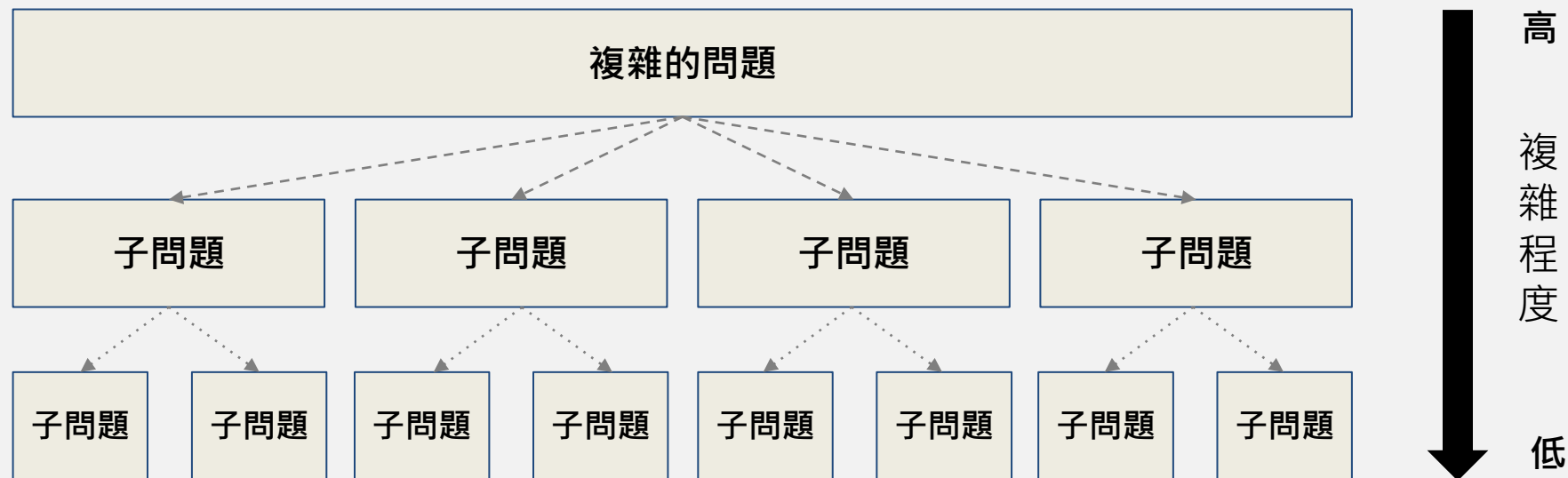
負回饋值

1-3 動態規劃 vs. 蒙地卡羅法



- 動態規劃 Dynamic Programming

→ 目標分解 Goal Decomposition

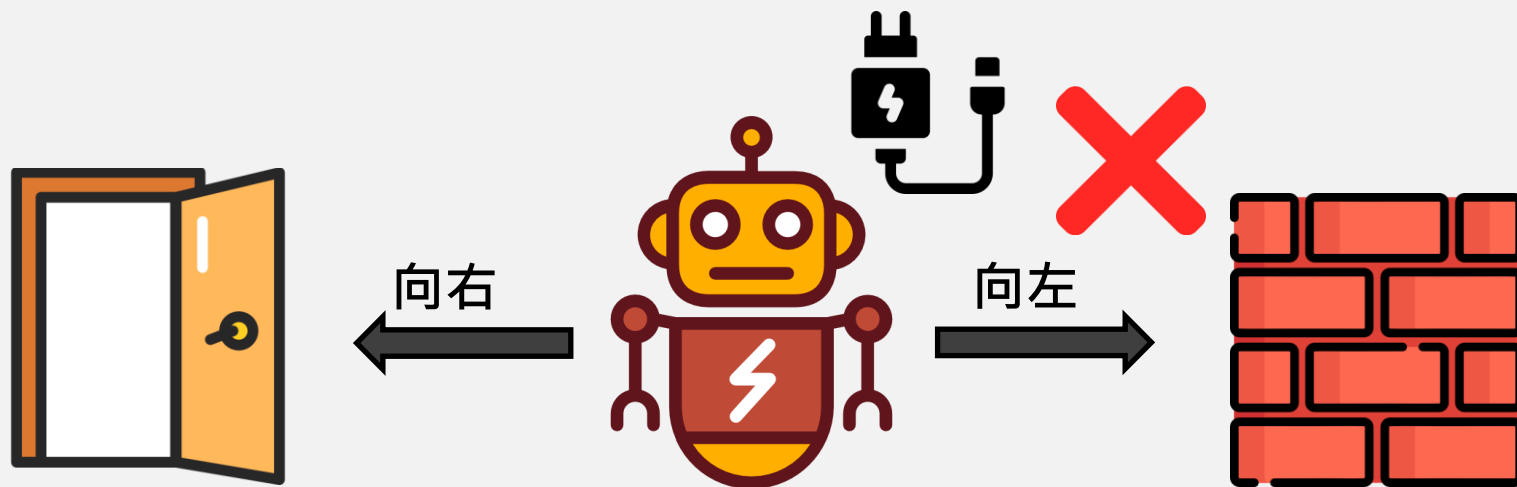


1-3 動態規劃 vs. 蒙地卡羅法



- 蒙地卡羅法 Monte Carlo Methods

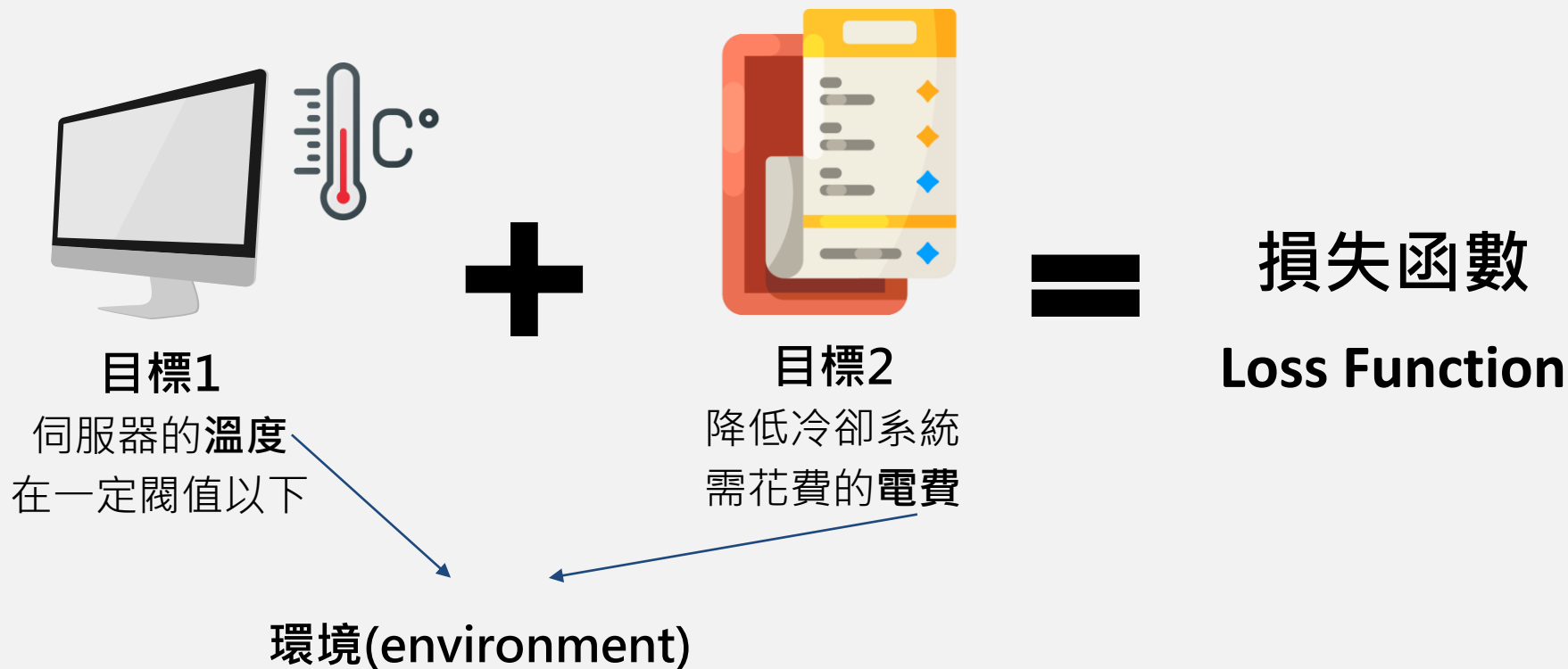
→ 試誤學習 Trial and Error



1-4 強化式學習架構



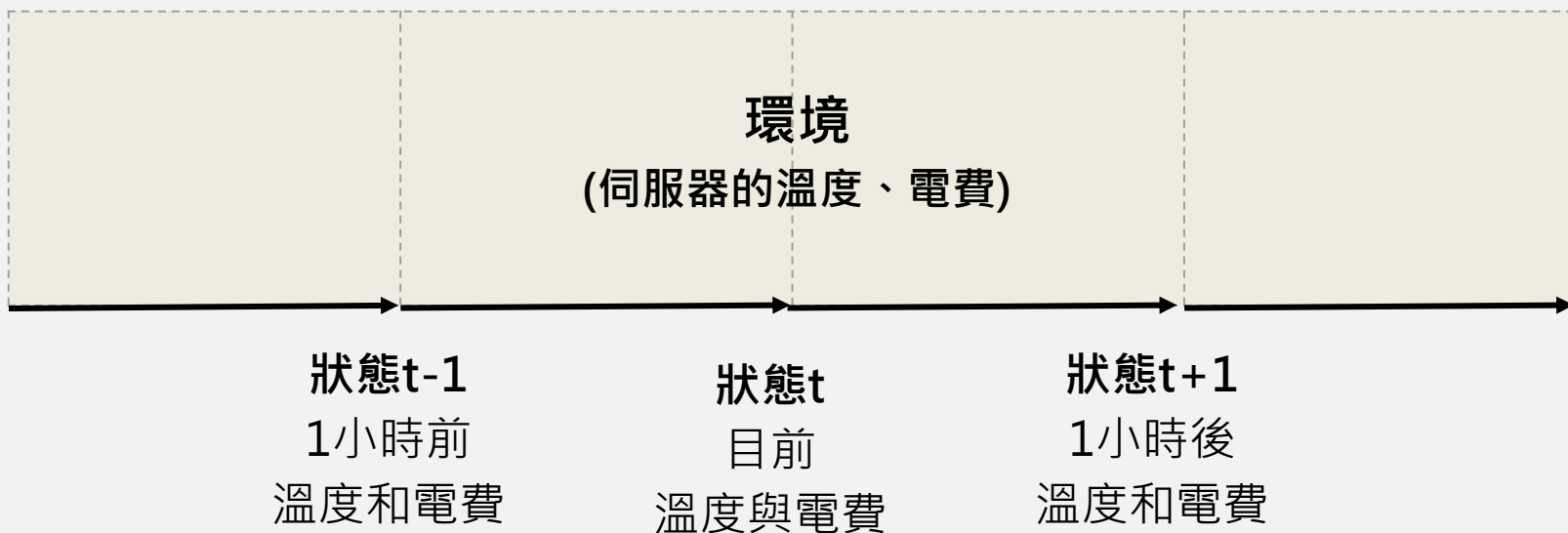
- 將目標整合為損失函數 (Loss Function)



1-4 強化式學習架構



- 環境是動態序列，依時間切割成一個個的狀態(State)



1-4 強化式學習與監督式學習



● 兩者之間的差異

強化式學習

1. 動作(Action)
2. 環境(Environment)
3. 回饋值(Reward)



監督式學習

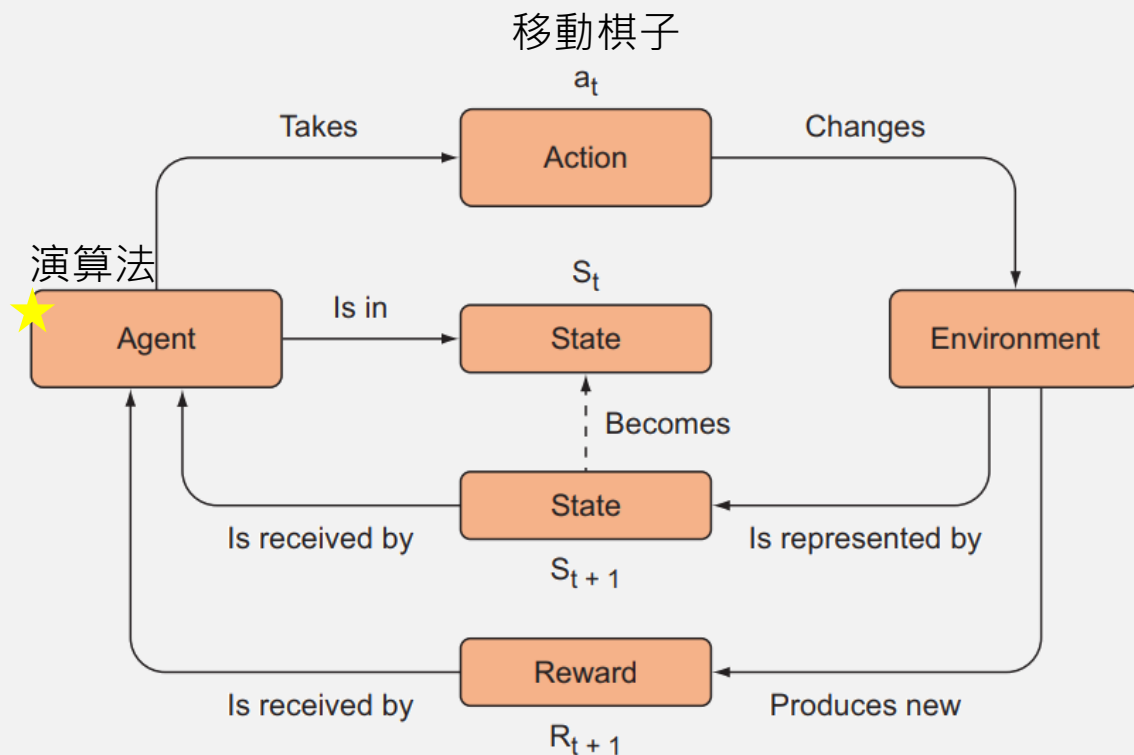
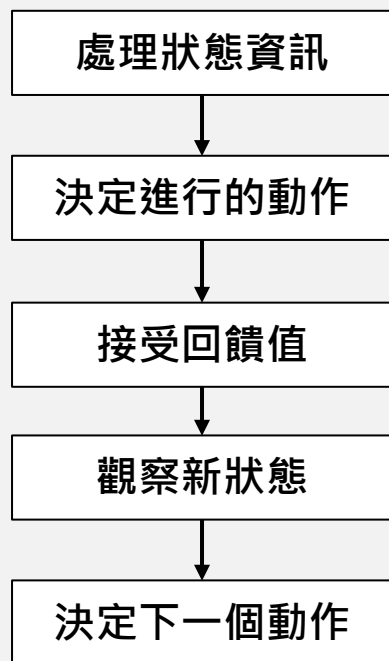
1. 資料量大
2. 需要Label

動作Action → 伺服器溫度降低,電費微幅提高 → 回饋值Reward +10
將冷氣調低

1-4 代理人 (Agent)



- 代理人即是演算法

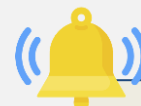
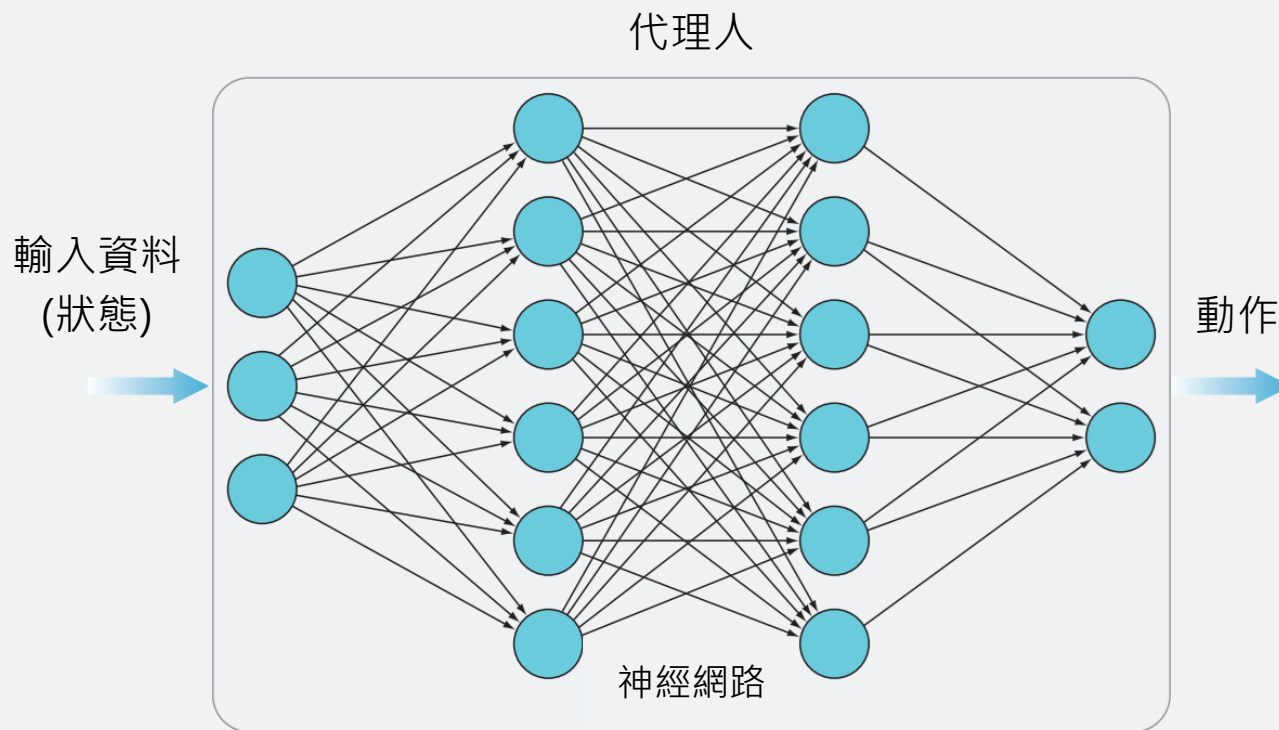


吃掉別人棋子+1
被人吃掉棋子-1
其他狀況+0

1-4 代理人 (Agent)



- 以深度學習演算法作為代理人



強化式學習

是一種特定問題的**解決方法**
而不是某種特定的演算法

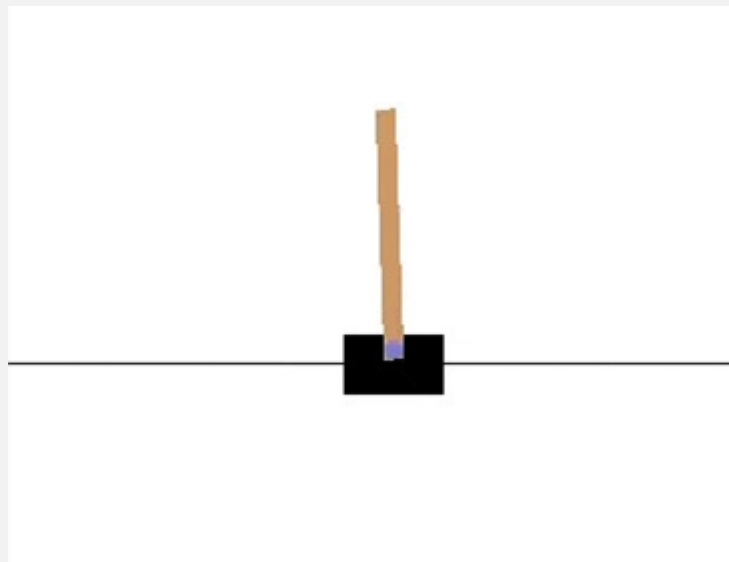
Python OpenAPI



- 不需自己重建新環境
- 利用『gym函示庫』來練習

程式碼

```
!pip install gym[box2d]  
import gym  
env = gym.make('CarRacing-v2')
```



1-5 通用人工智慧(AGI)

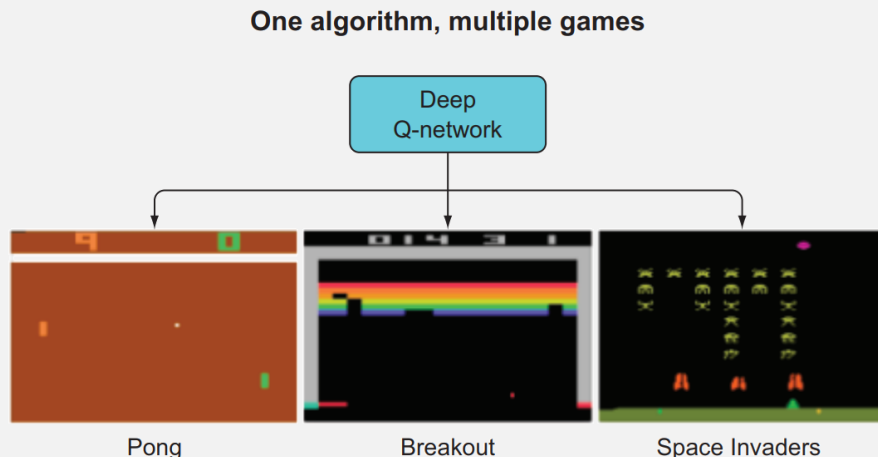


Artificial General Intelligence, AGI

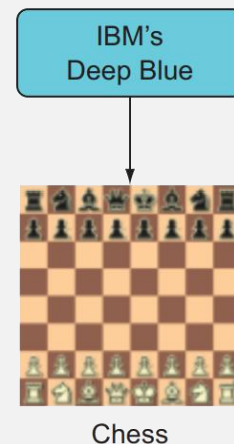
- AGI 是人工智慧研究的終極目標
 - 不受監督
 - 技能可跨領域



強化式學習最有可能實現AGI



DeepMind DQN演算法
可以應付多種遊戲目標是
得分最大化



Rule-based演算法
只在單一遊戲表現良好

1-6 為何要使用『深度』強化式學習



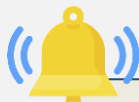
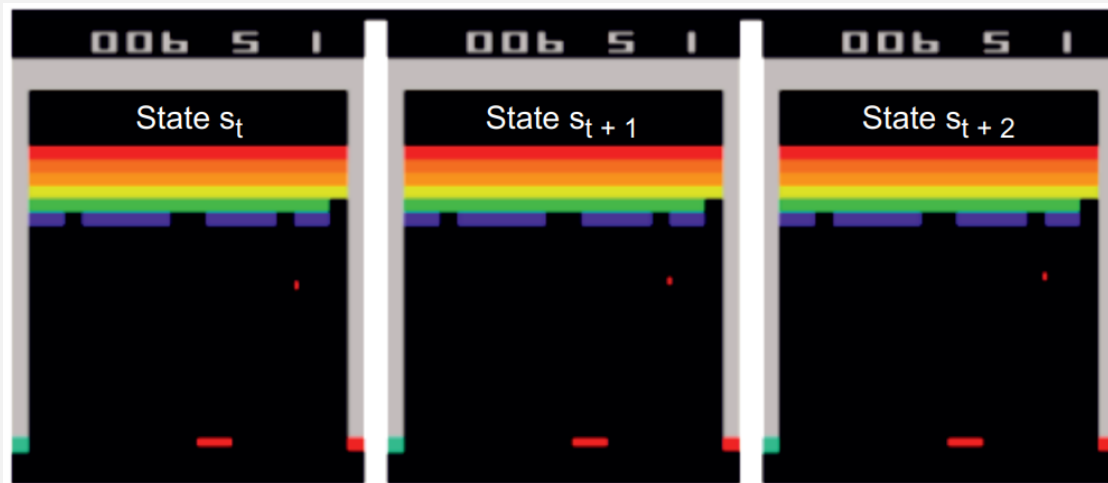
- 強化式學習只可以應付單純的環境
- 井字遊戲
 - 使用記憶表(memory table)

Game play lookup table

Key Current state	Value Action to take
	Place \times in top left
	Place \times in top right
	Place \times in bottom right

- 圈叉位置只有固定幾個

- 深度強化式學習才能應付複雜的環境
- 打方塊遊戲
 - 定義特徵(features)
ex. 球、橫桿、剩餘方塊的座標



深度強化式學習
即是使用深度神經網路作為代理人的強化式學習

- 球的位置有 256^{28224} 種

1-7 線圖 (String diagram)



- 視覺化的溝通工具
 - 表達資料在不同處理程序之間的流動

