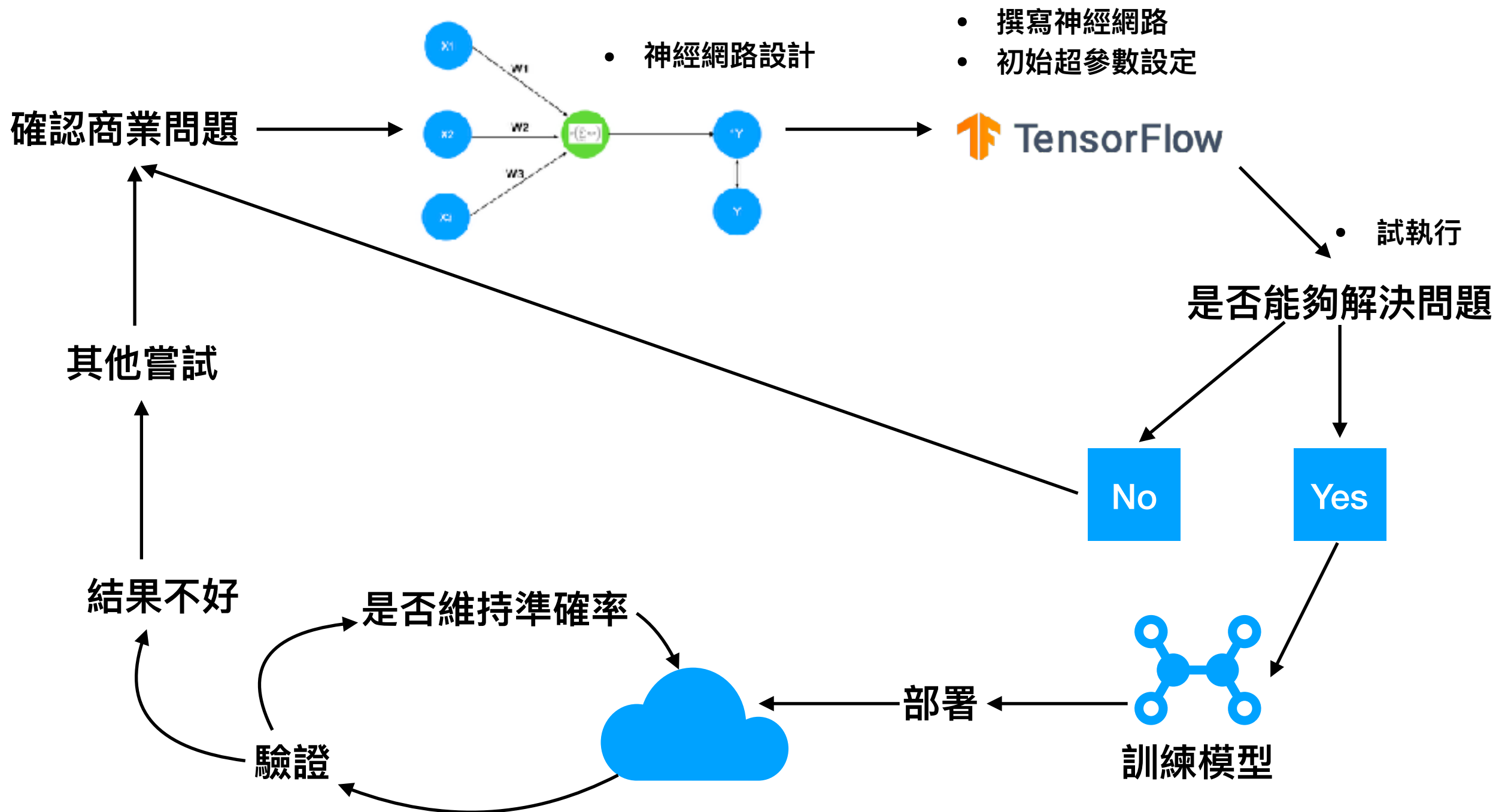


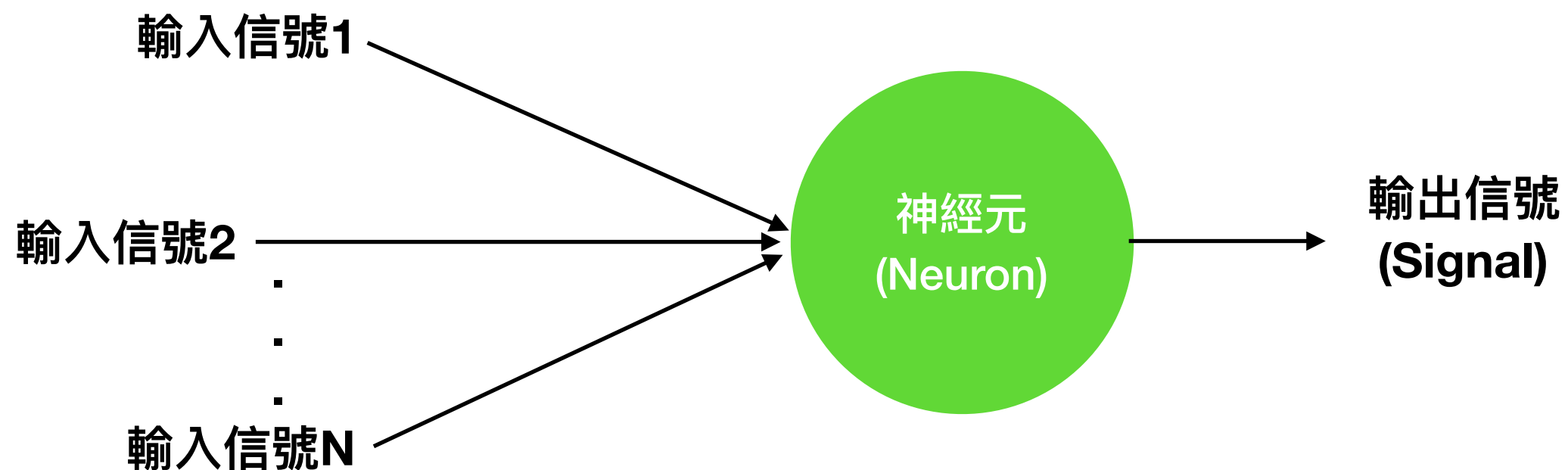
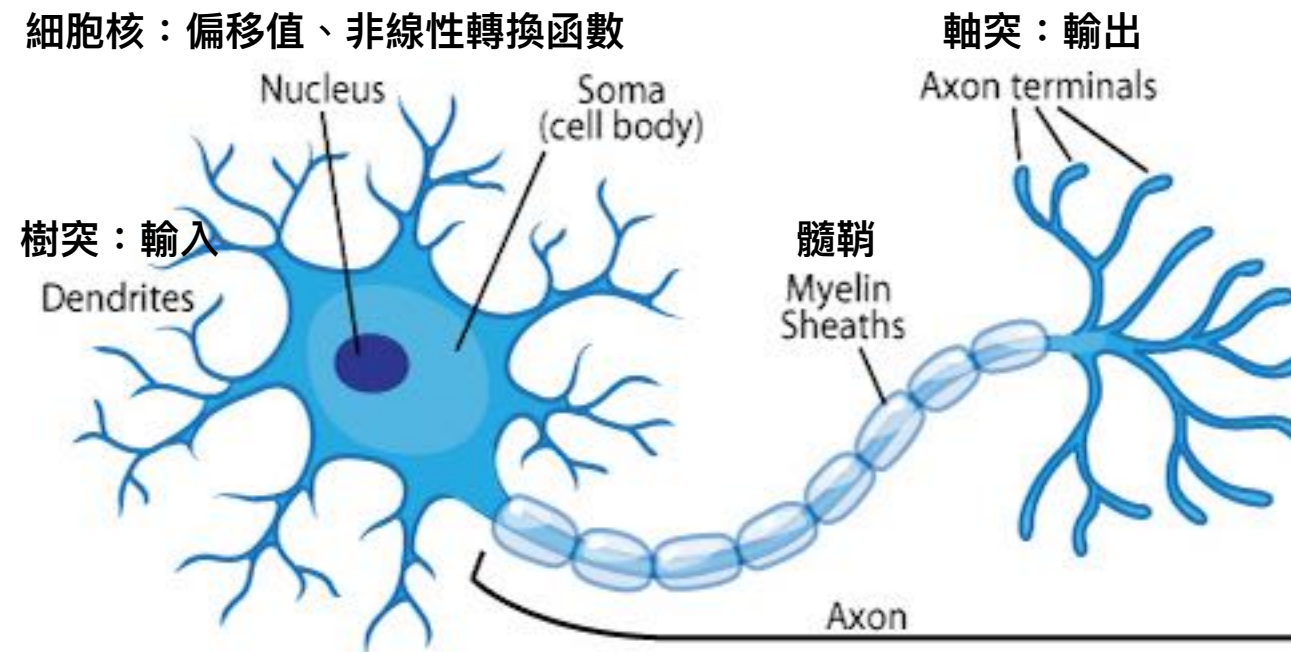
神經網路設計工作流程



人工神經網路(ANN)

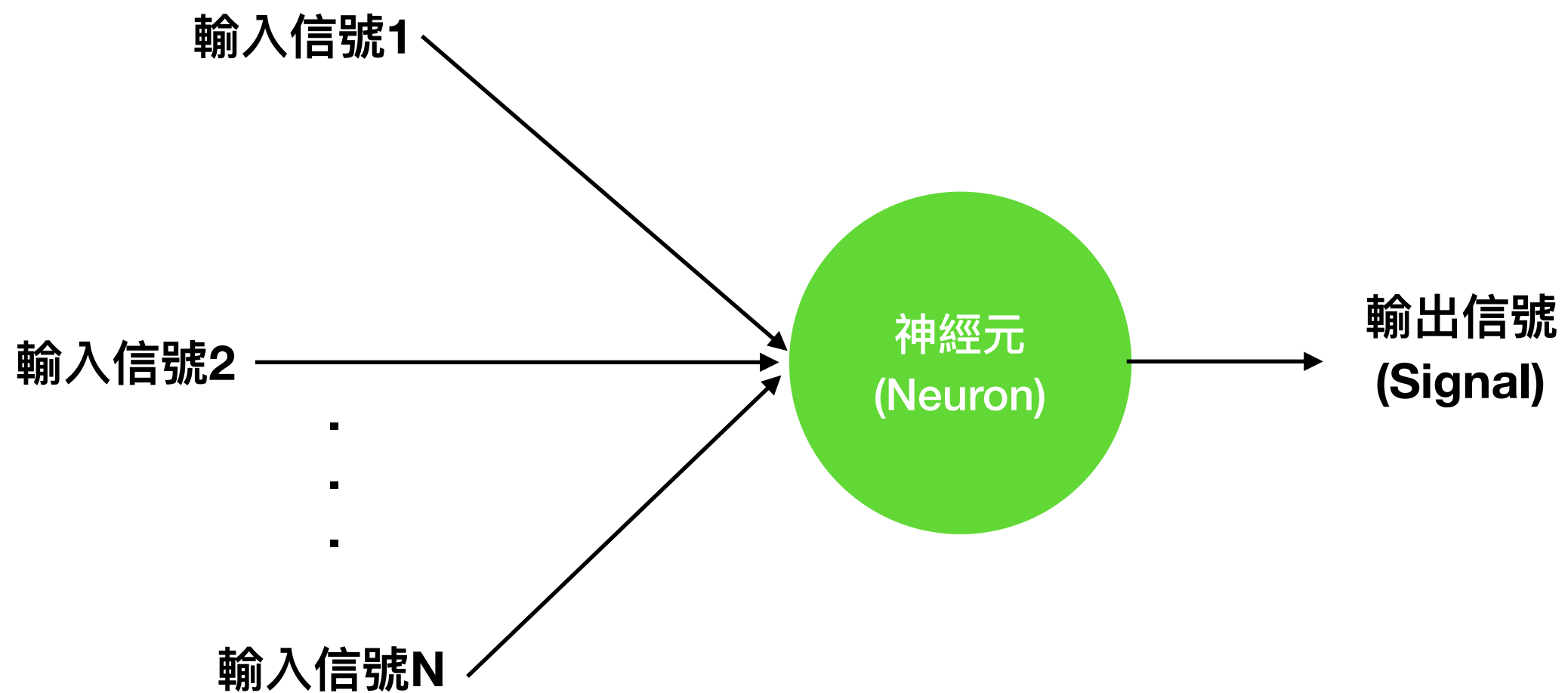
- 神經元(Neuron)
- 激活函數(Activation Function)
- 神經網路(Neural Network)如何運作？
- 梯度下降(Gradient descent)
- 反向傳播(Backpropagation)

神經元(Neuron)



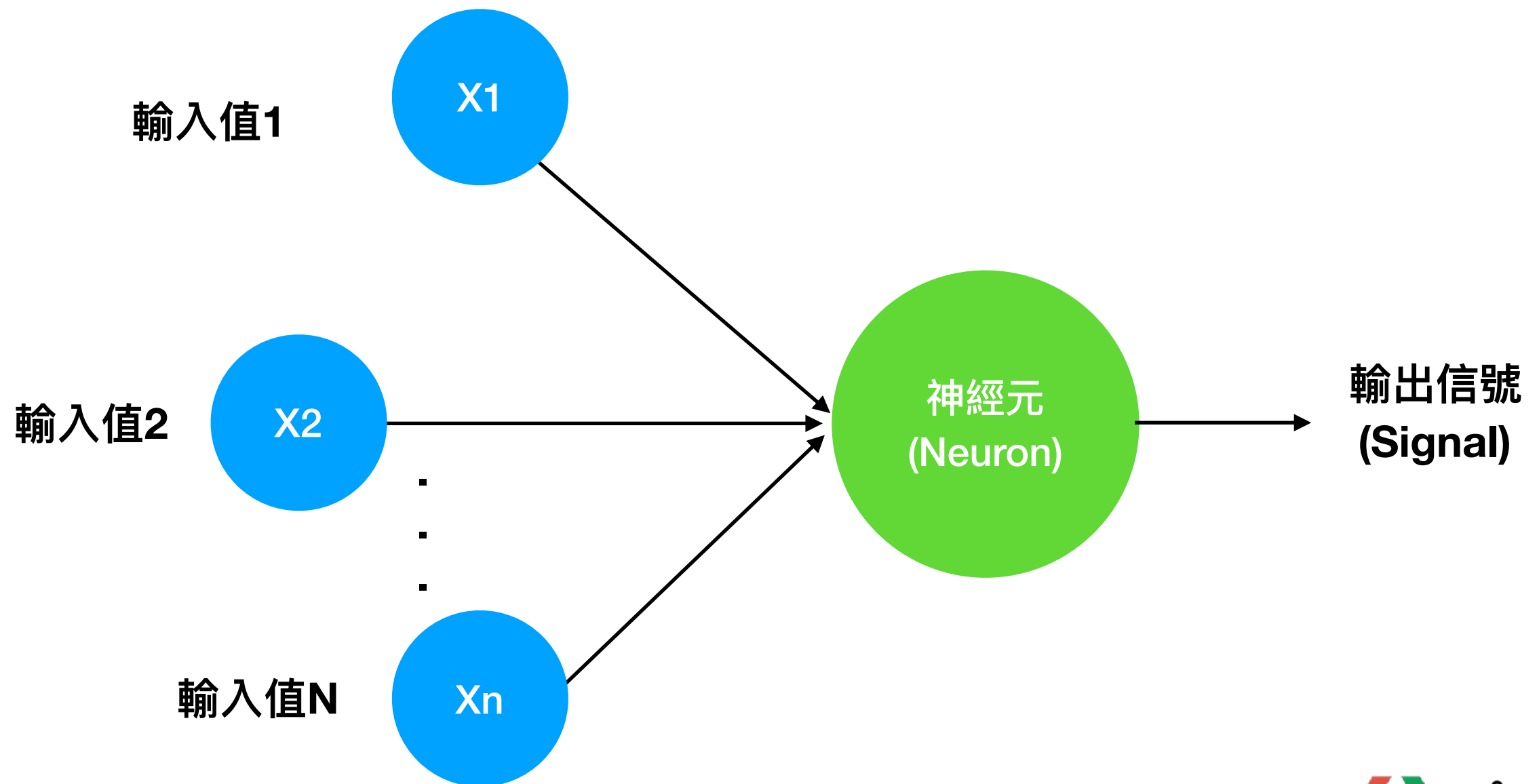
神經元(Neuron)

- 神經元的功能為執行一個轉換函數



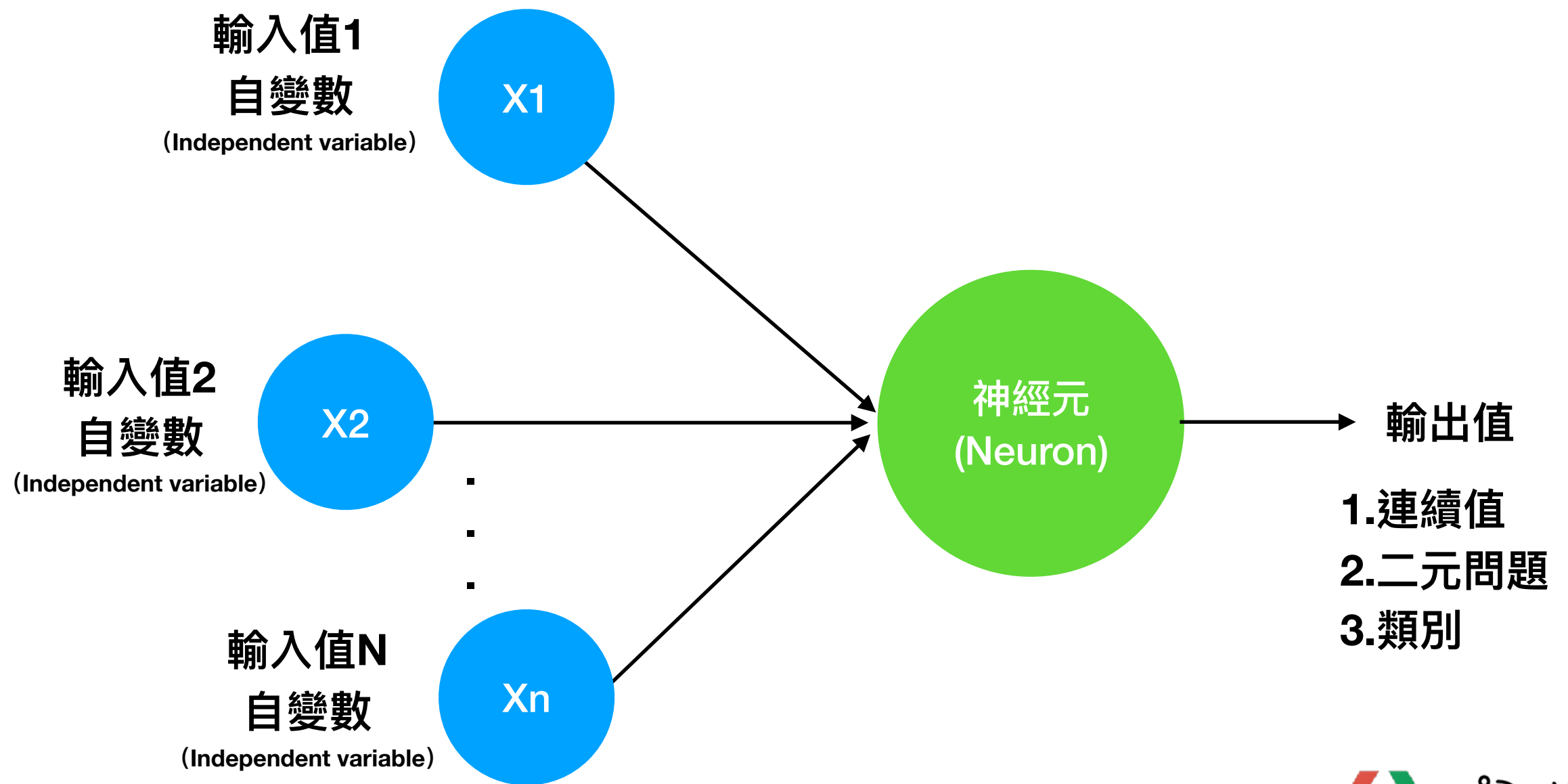
神經元(Neuron)

- 神經元的功能為執行一個轉換函數



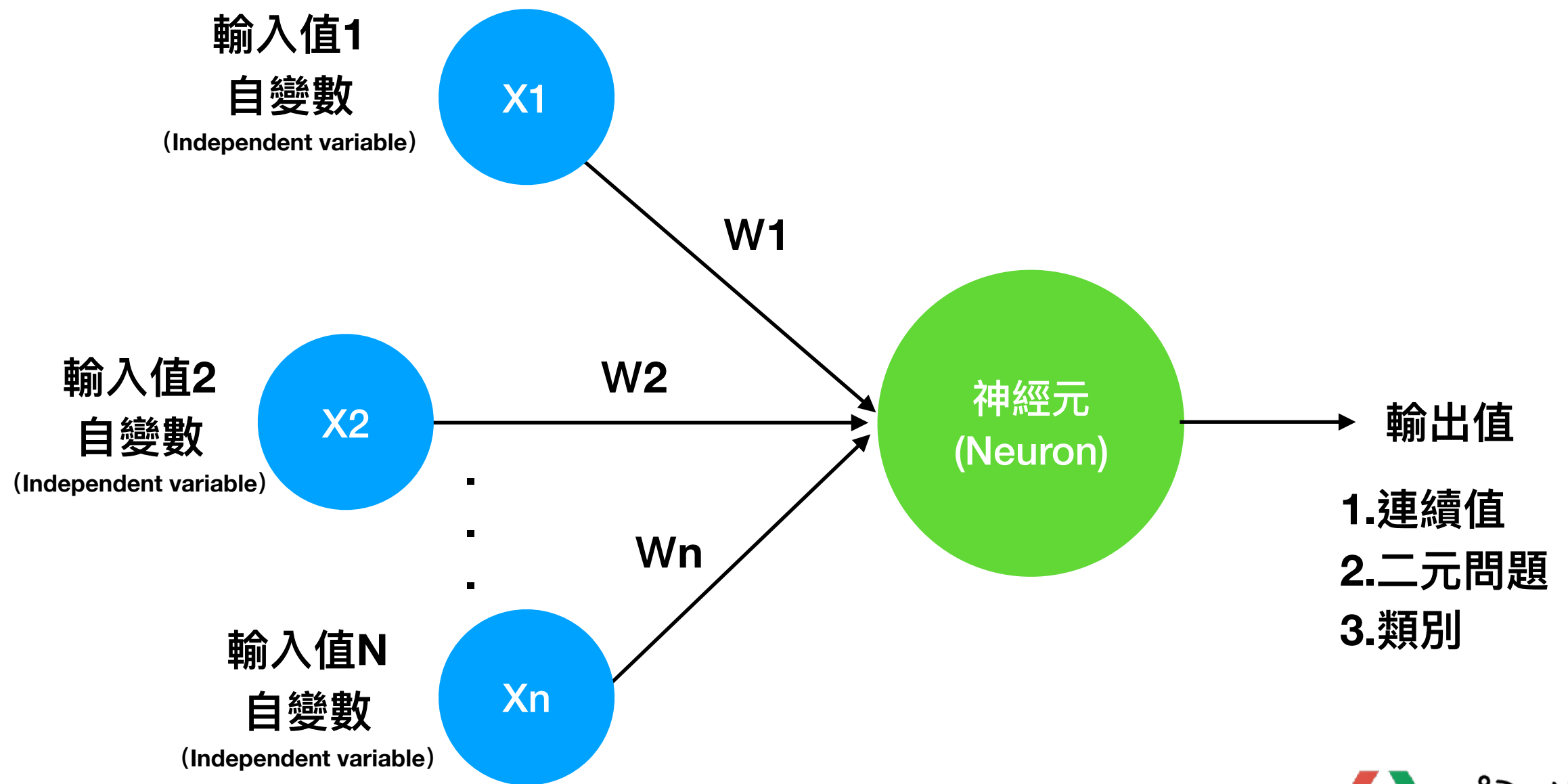
神經元(Neuron)

- 神經元的功能為執行一個轉換函數



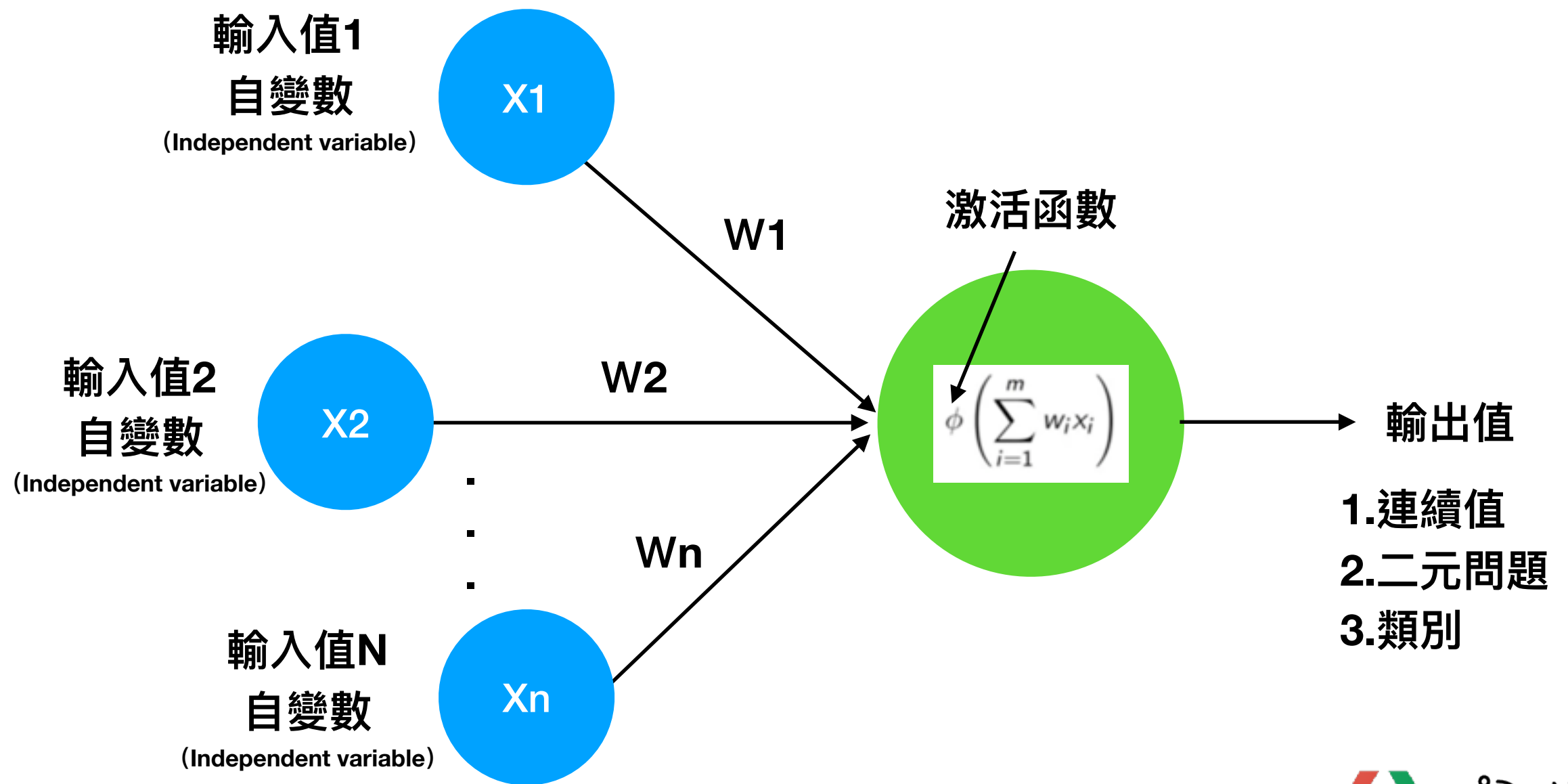
神經元(Neuron)

- 神經元的功能為執行一個轉換函數



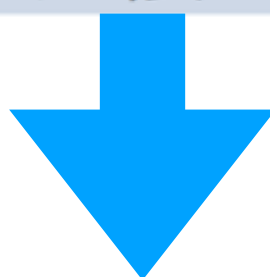
神經元(Neuron)

- 神經元的功能為執行一個轉換函數



神經元(Neuron)

屋齡	樓層	建築型態
大於20	小於9樓	公寓
大於20	大於9樓	商辦
小於20	大於9樓	商辦



屋齡	轉換	樓層	轉換	建築型態	轉換
小於20	0	大於9樓	1	商辦	0
大於20	1	小於9樓	0	公寓	1
大於20	1	大於9樓	1	商辦	0

神經元(Neuron)

屋齡	轉換	樓層	轉換	建築型態	轉換
小於20	0	大於9樓	1	商辦	0
大於20	1	小於9樓	0	公寓	1
大於20	1	大於9樓	1	商辦	0

Input	Weight(0,0)	Output
(0, 1)	(0, 0)	0
(1, 0)	(1, 0)	1
(1, 1)	(0, 0)	0

$$0*0 + 1*0 = 0$$

$$0*1 + 0*0 = 0 \text{ 不等於 } 1$$

$$\text{調整權重} \Rightarrow 1*1 + 0*0 = 1$$

$$1*1 + 0*1 = 1 \text{ 不等於 } 0$$

$$\text{調整權重} \Rightarrow 1*0 + 1*0 = 0$$

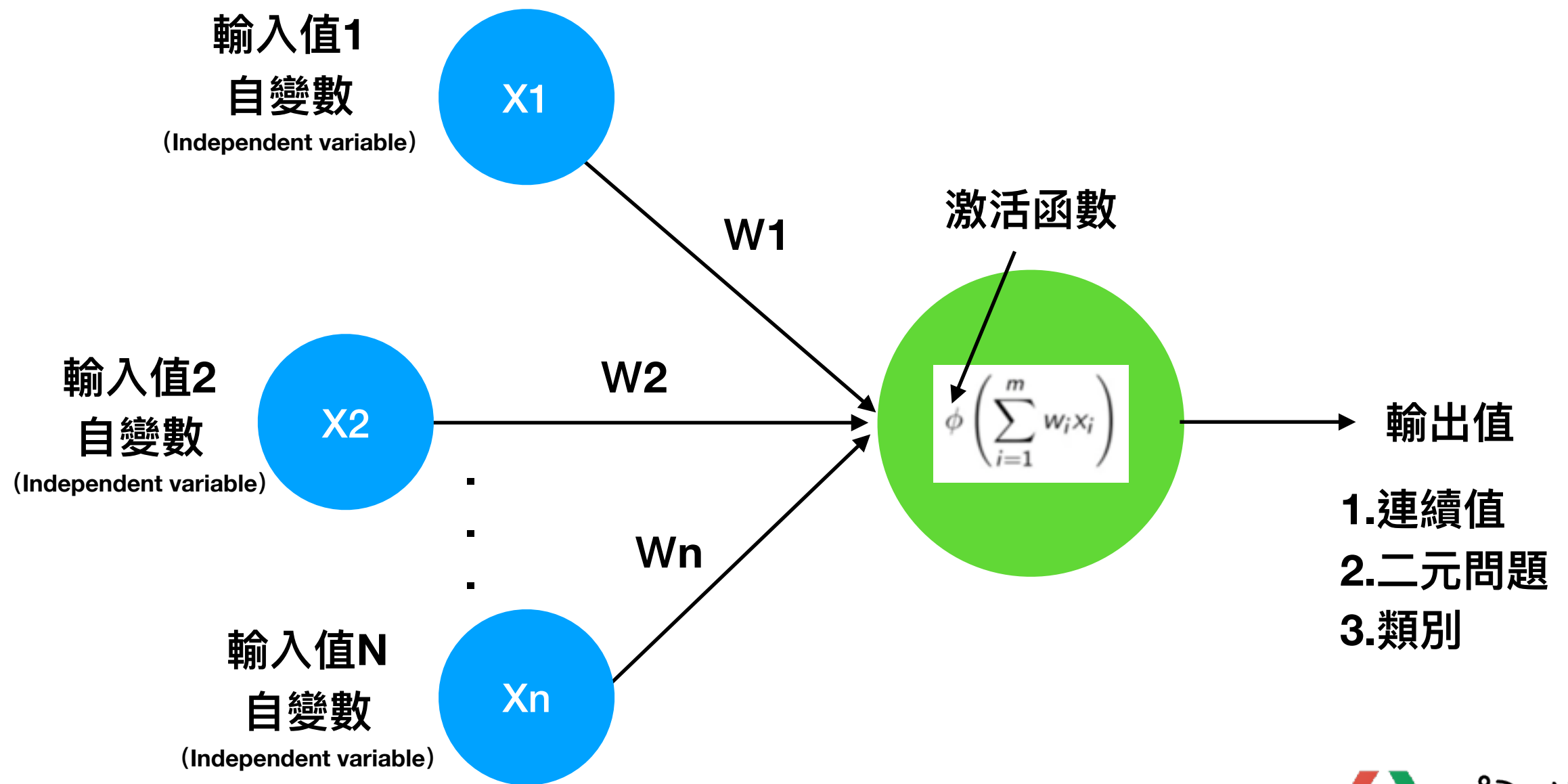
備註：不考慮激活函數，以線性的概念來看

人工神經網路(ANN)

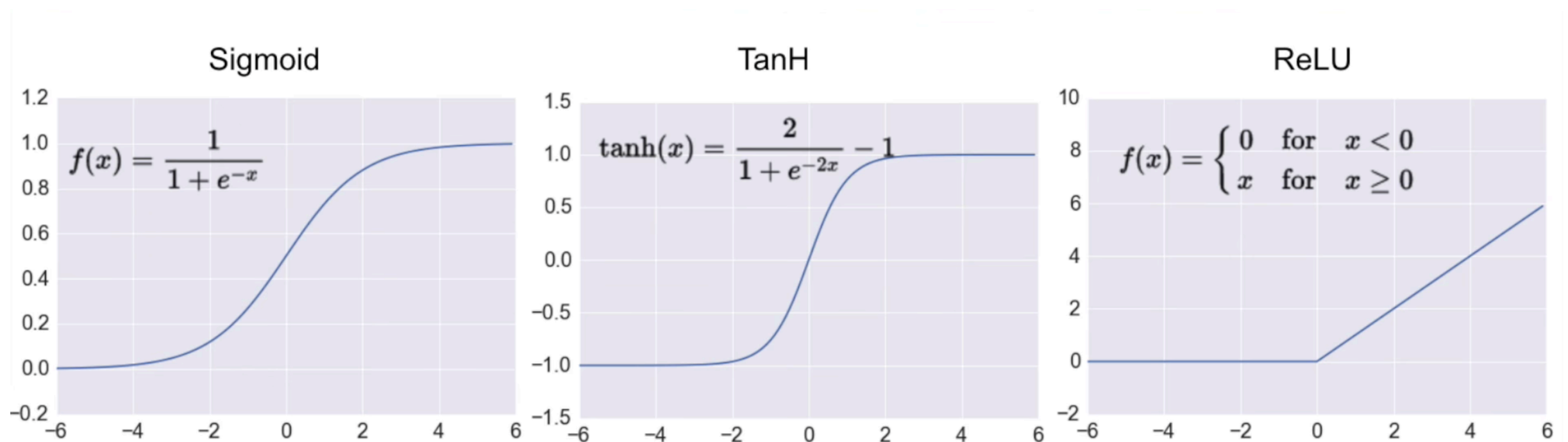
- 神經元(Neuron)
- 激活函數(Activation Function)
- 神經網路(Neural Network)如何運作？
- 梯度下降(Gradient descent)
- 反向傳播(Backpropagation)

神經元(Neuron)

- 神經元的功能為執行一個轉換函數



激活函數(Activation Function)



Sigmoid函數

取值範圍(0, 1)

適合用於特徵相差小效果好

雙曲正切函數 (TANH)

取值範圍(-1, 1)

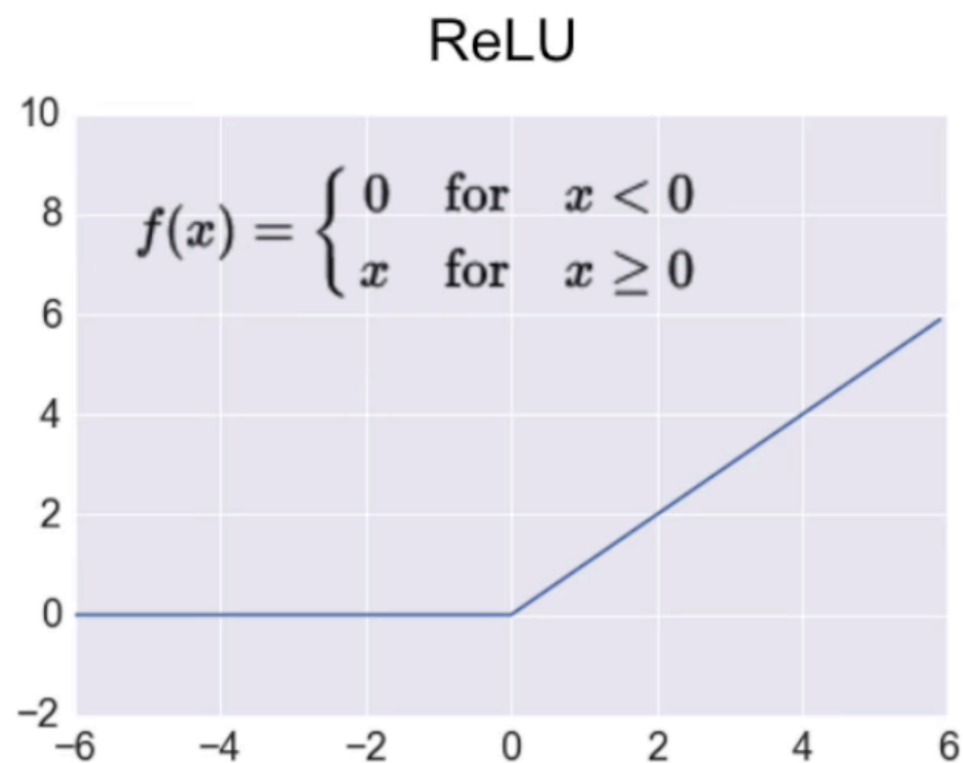
適合用於特徵相差大效果好

線性整流函數 (ReLU)

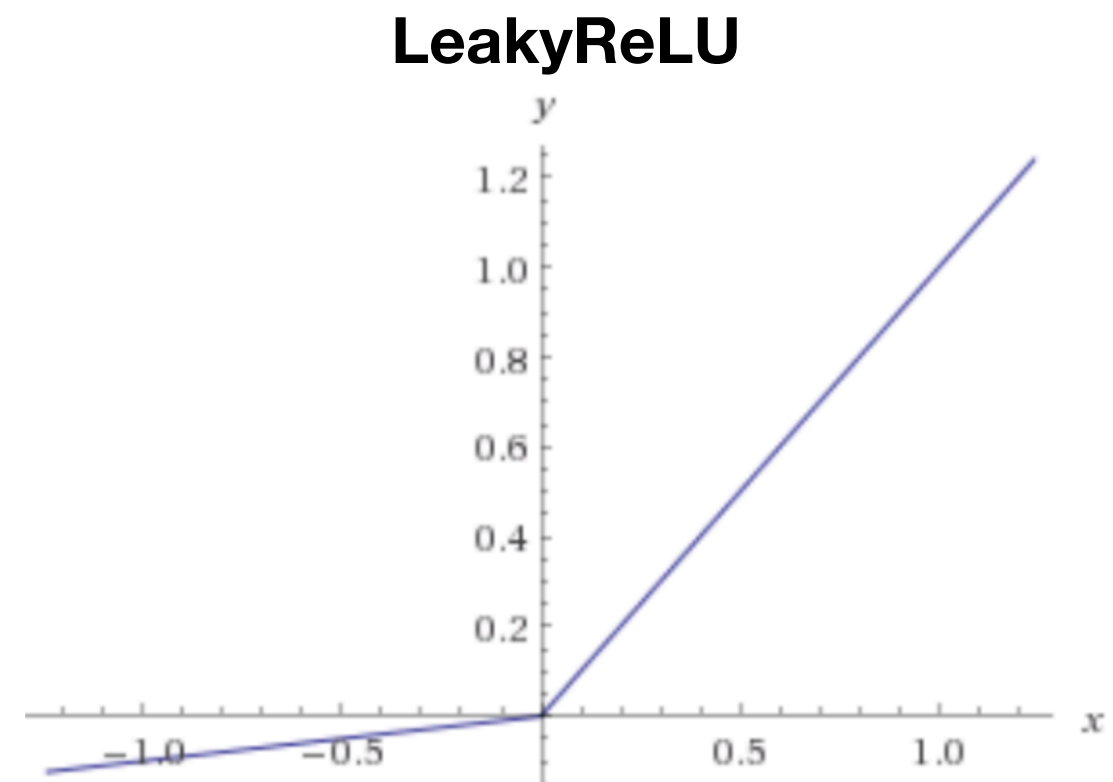
小於0為0，輸出等於輸入
運算效率高

激活函數(Activation Function)

是線性整流函數(Rectified linear unit)的另一種版本，能夠有非零的輸出值，而ReLU是直接給0。



$$f(x) = \max(0, x)$$



$$f(x) = \begin{cases} x & \text{if } x > 0 \\ \lambda x & \text{if } x \leq 0 \end{cases}$$

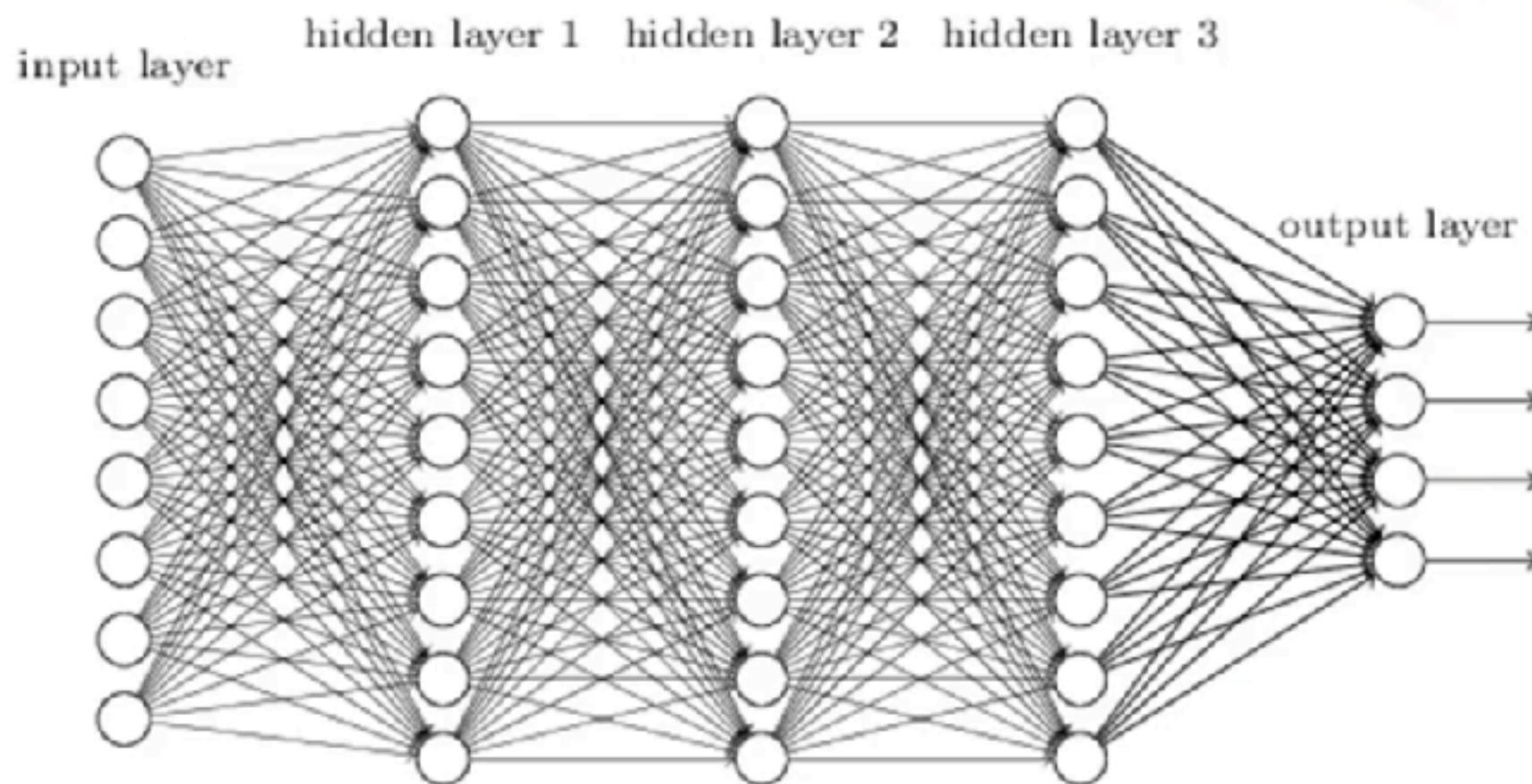
人工神經網路(ANN)

- 神經元(Neuron)
- 激活函數(Activation Function)
- 神經網路(Neural Network)如何運作？
- 梯度下降(Gradient descent)
- 反向傳播(Backpropagation)

神經網路(Neural Network)

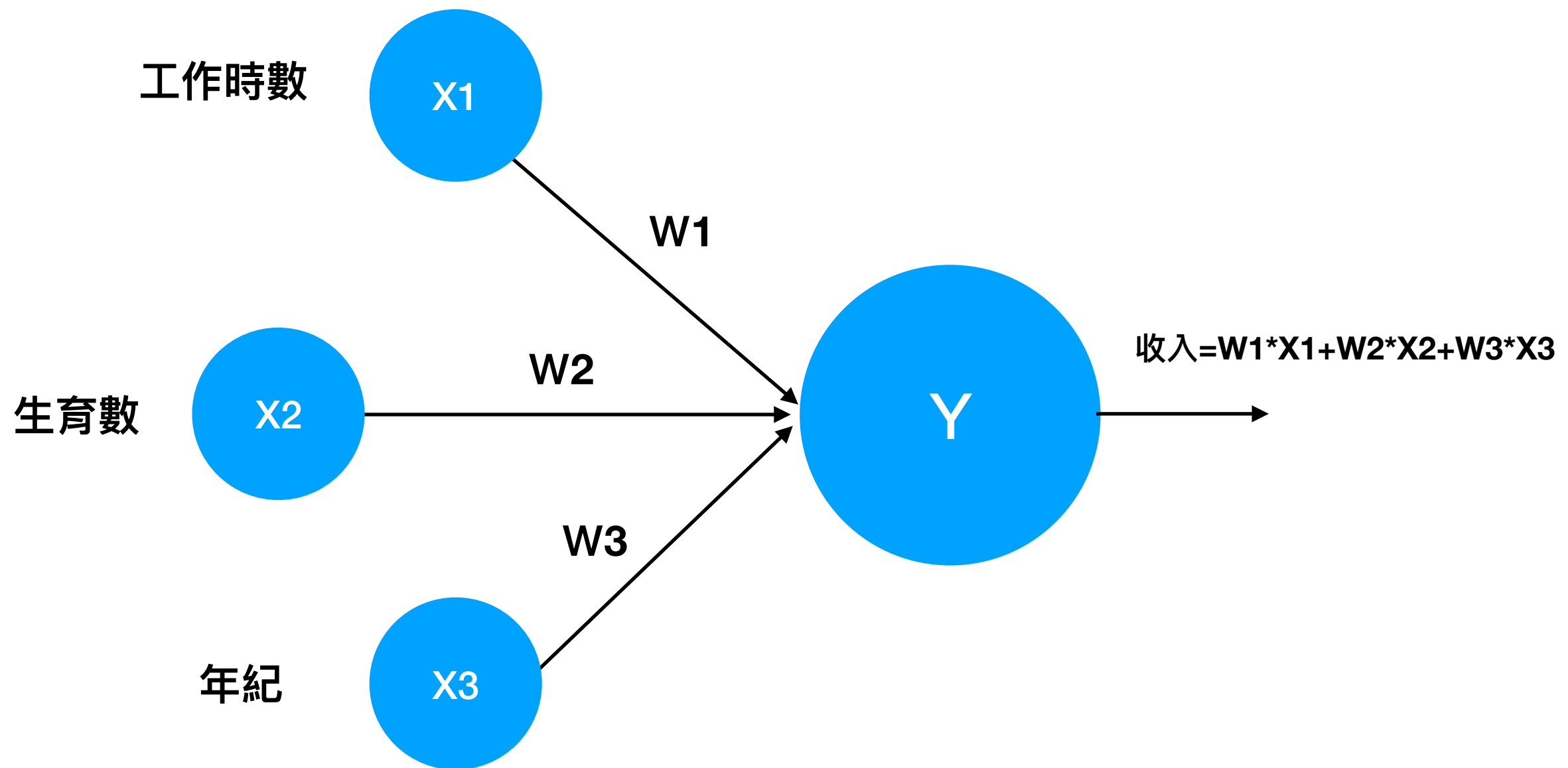
如何運作？

- 神經網路包含N個神經元
- 神經網路具有 $N \times N$ 條互連網路



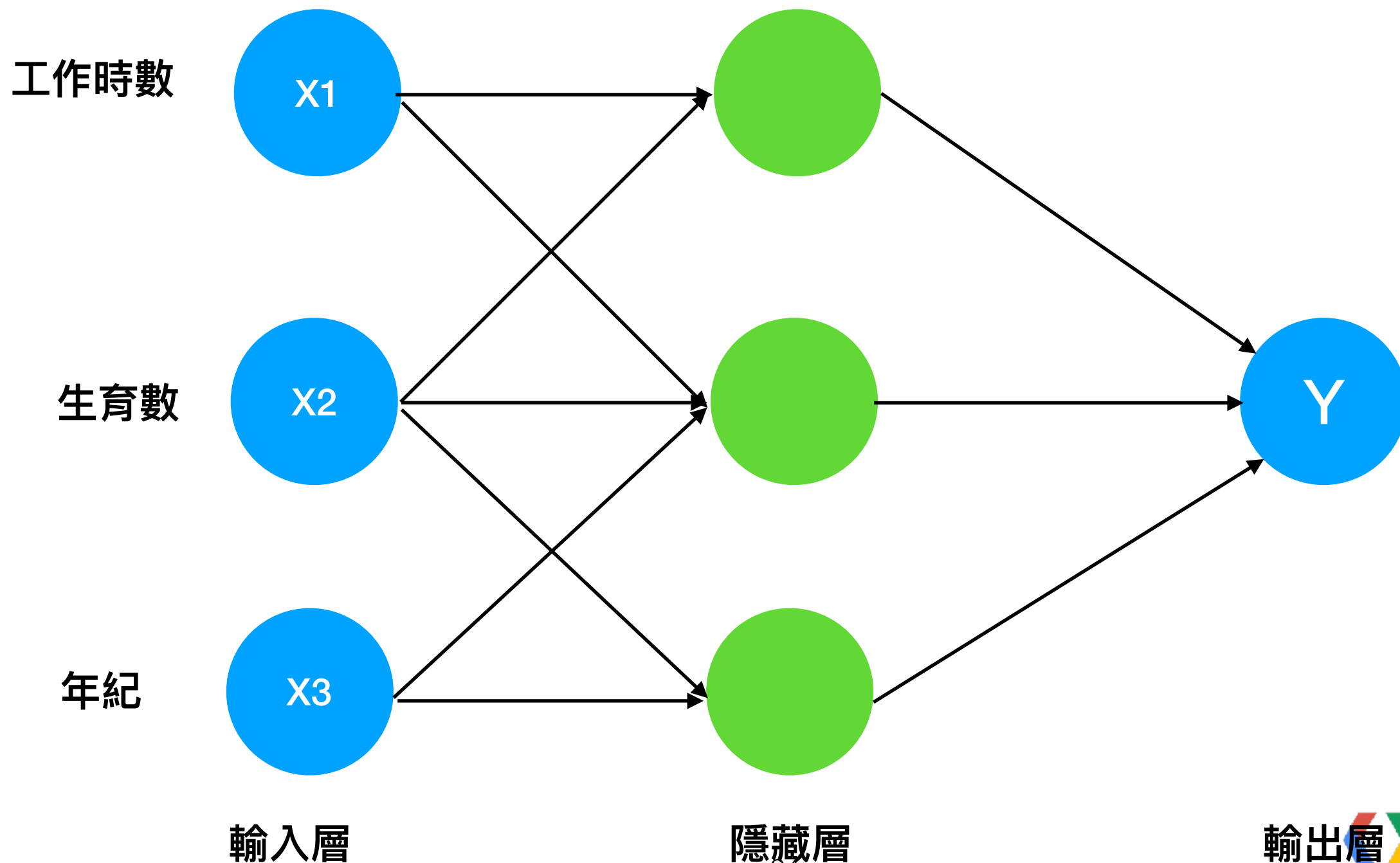
神經網路(Neural Network)

如何運作？



神經網路(Neural Network)

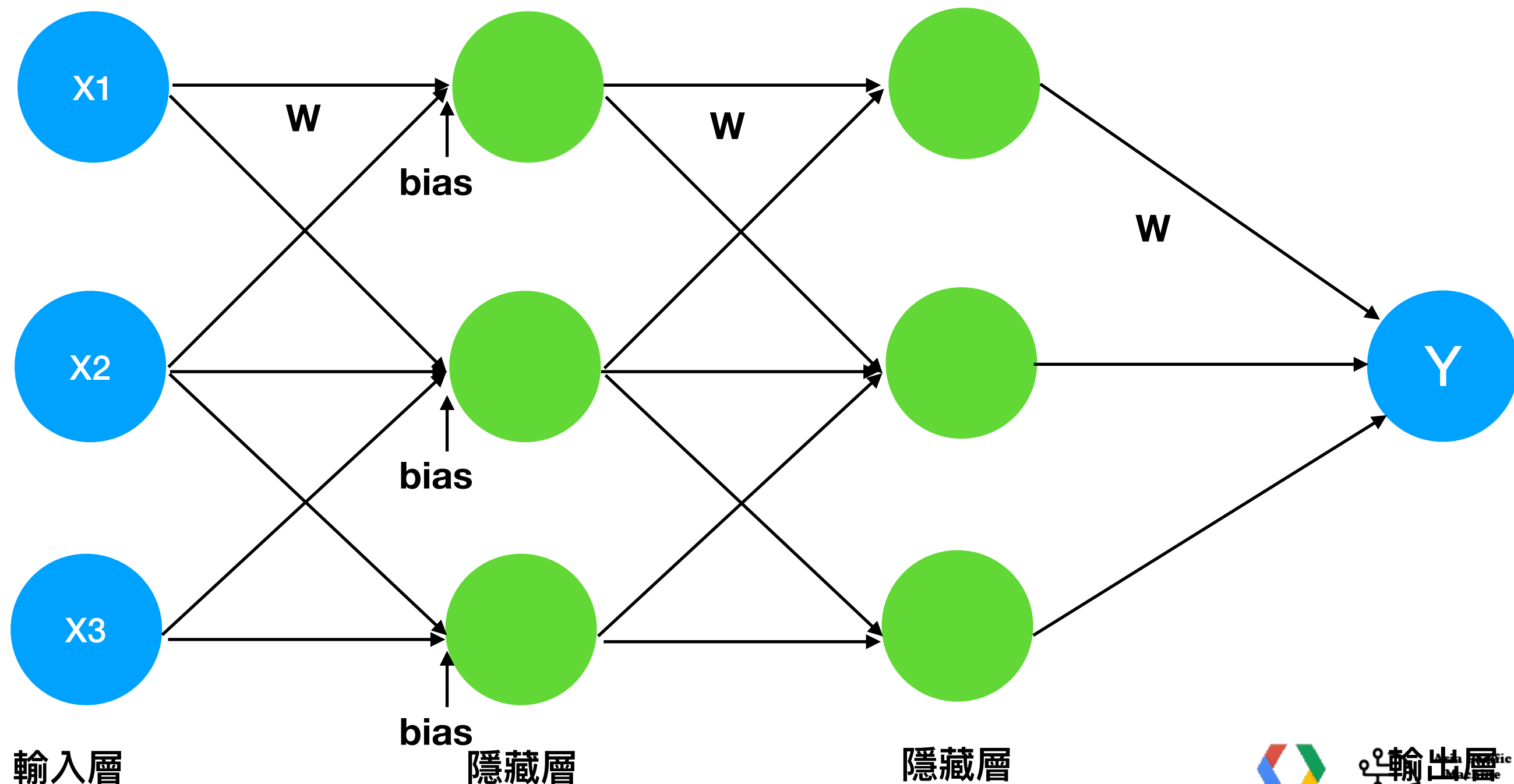
如何運作？



神經網路(Neural Network)

如何運作？

透過Fully Connect Feedforward來做連接



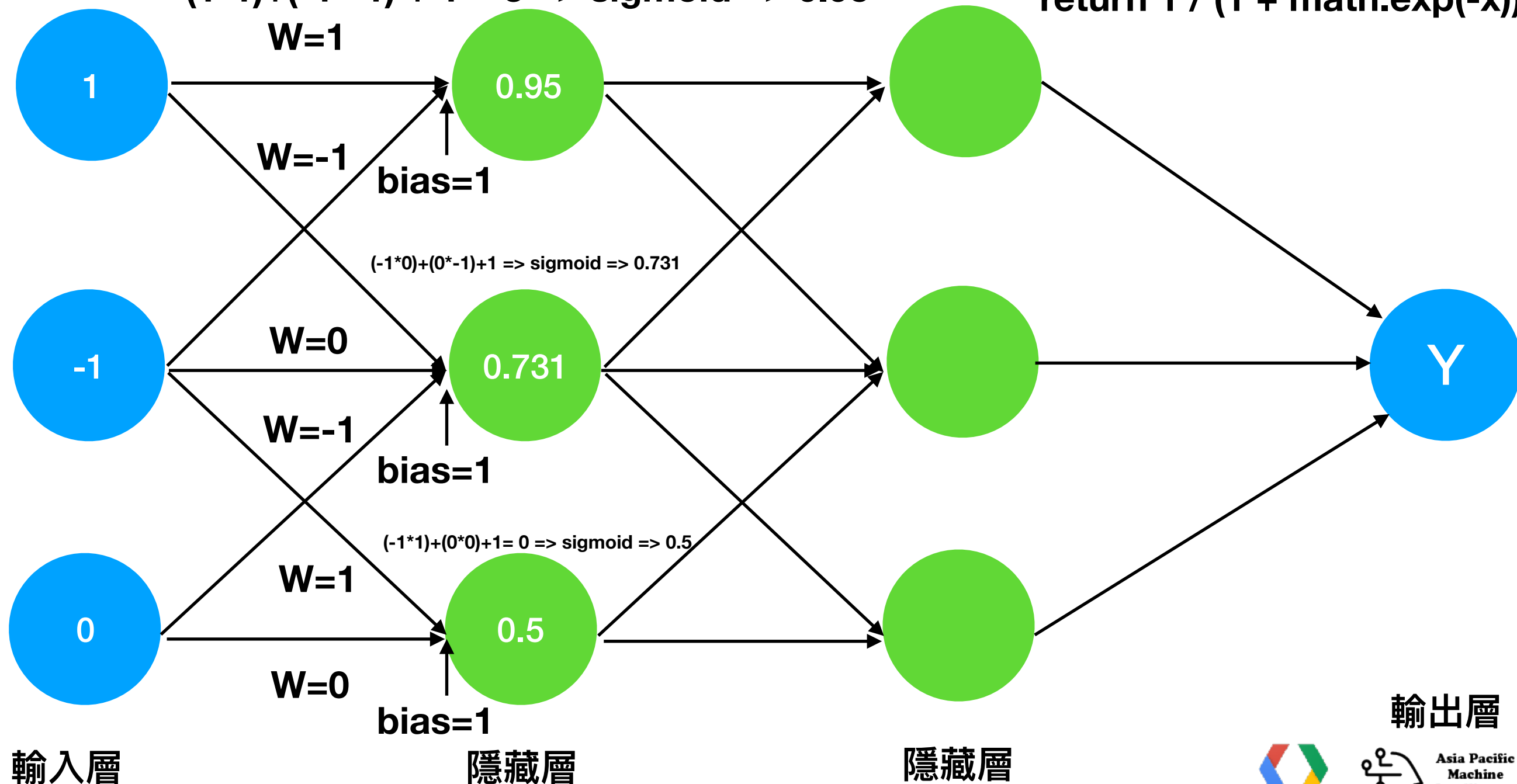
神經網路(Neural Network)

如何運作？

透過Fully Connect Feedforward來做連接

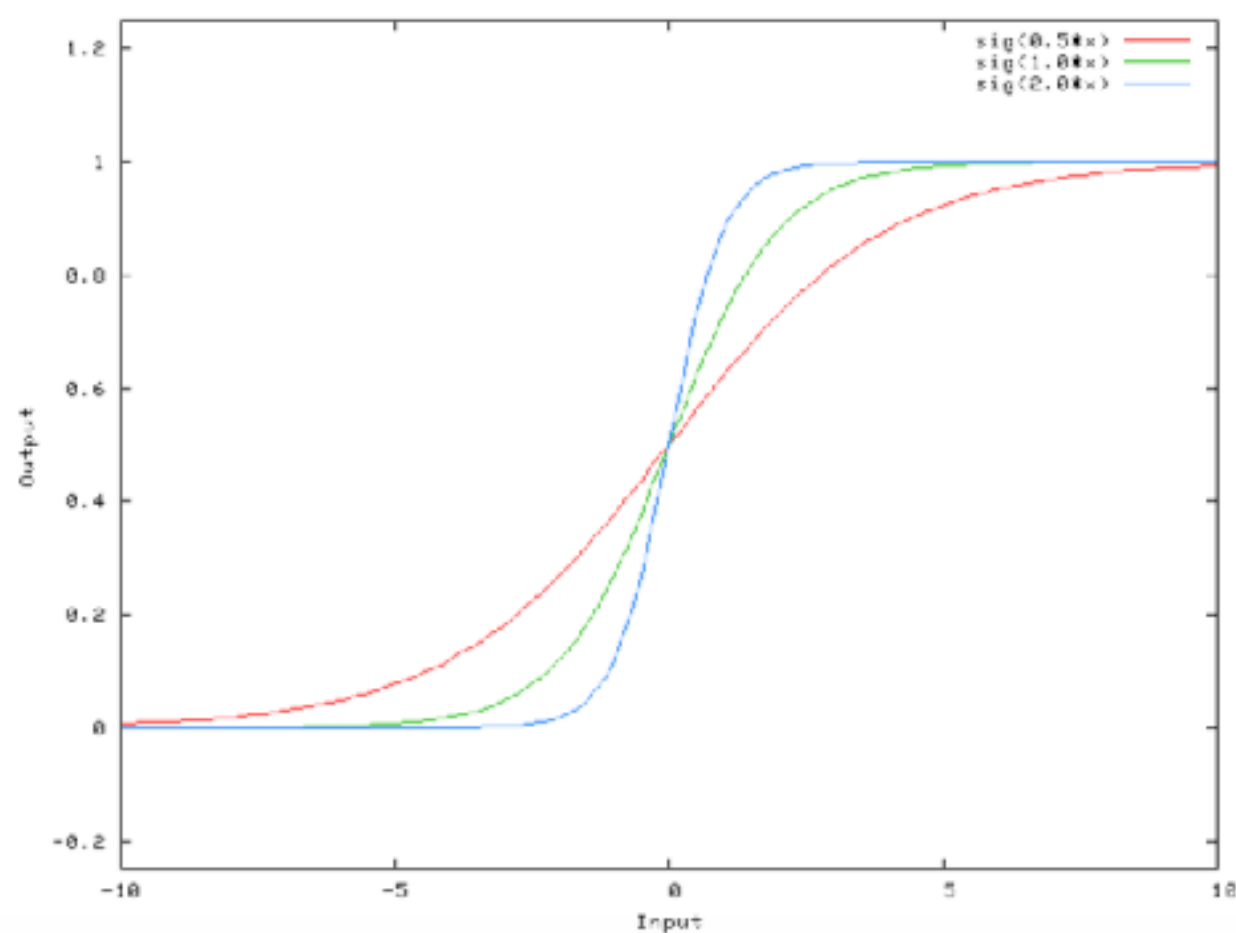
$$(1*1)+(-1*-1) + 1 = 3 \Rightarrow \text{sigmoid} \Rightarrow 0.95$$

```
import math
def sigmoid(x):
    return 1 / (1 + math.exp(-x))
```

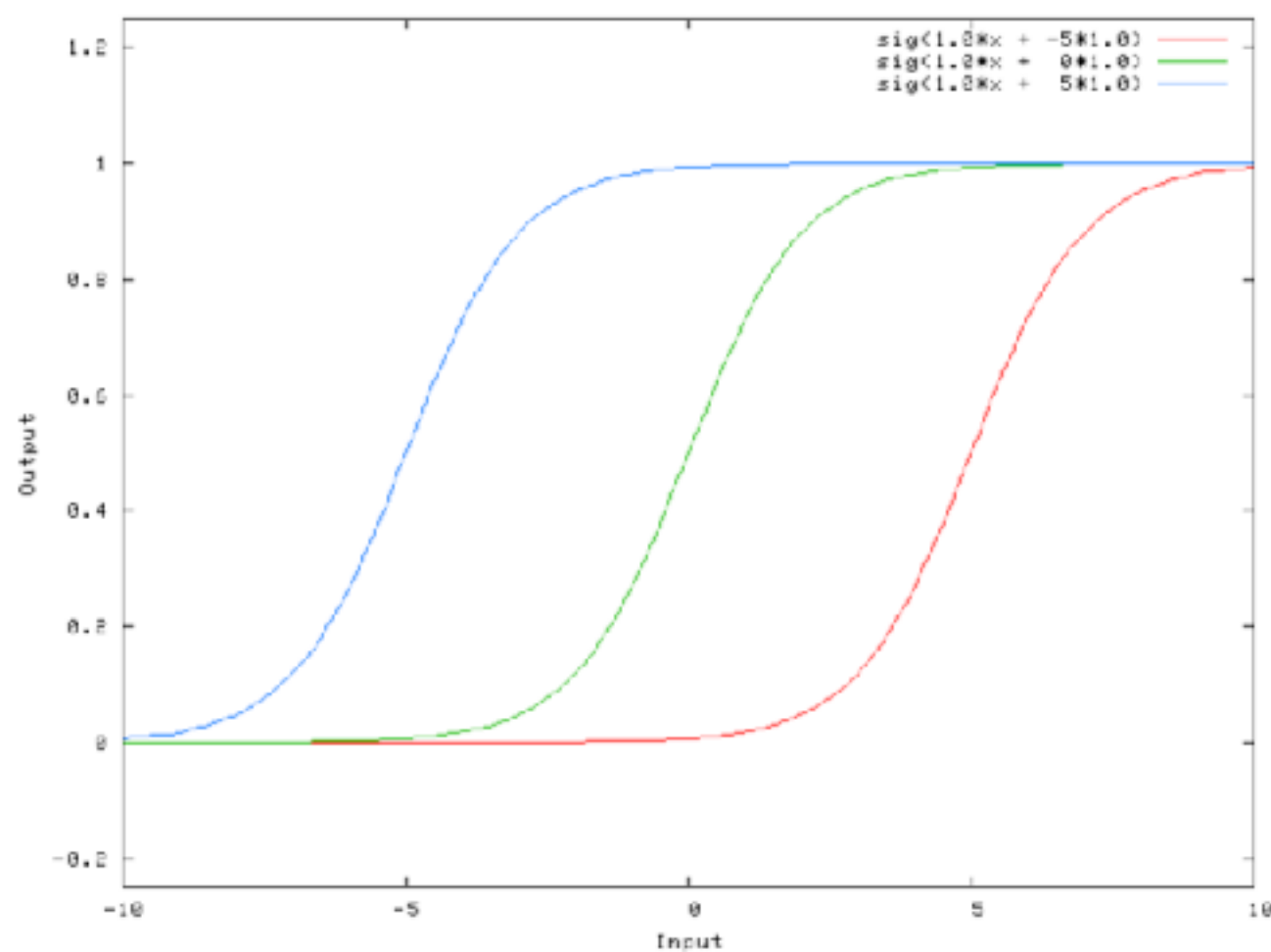


偏移量的功能

- 增加網路分類的表現能力



沒有偏移量

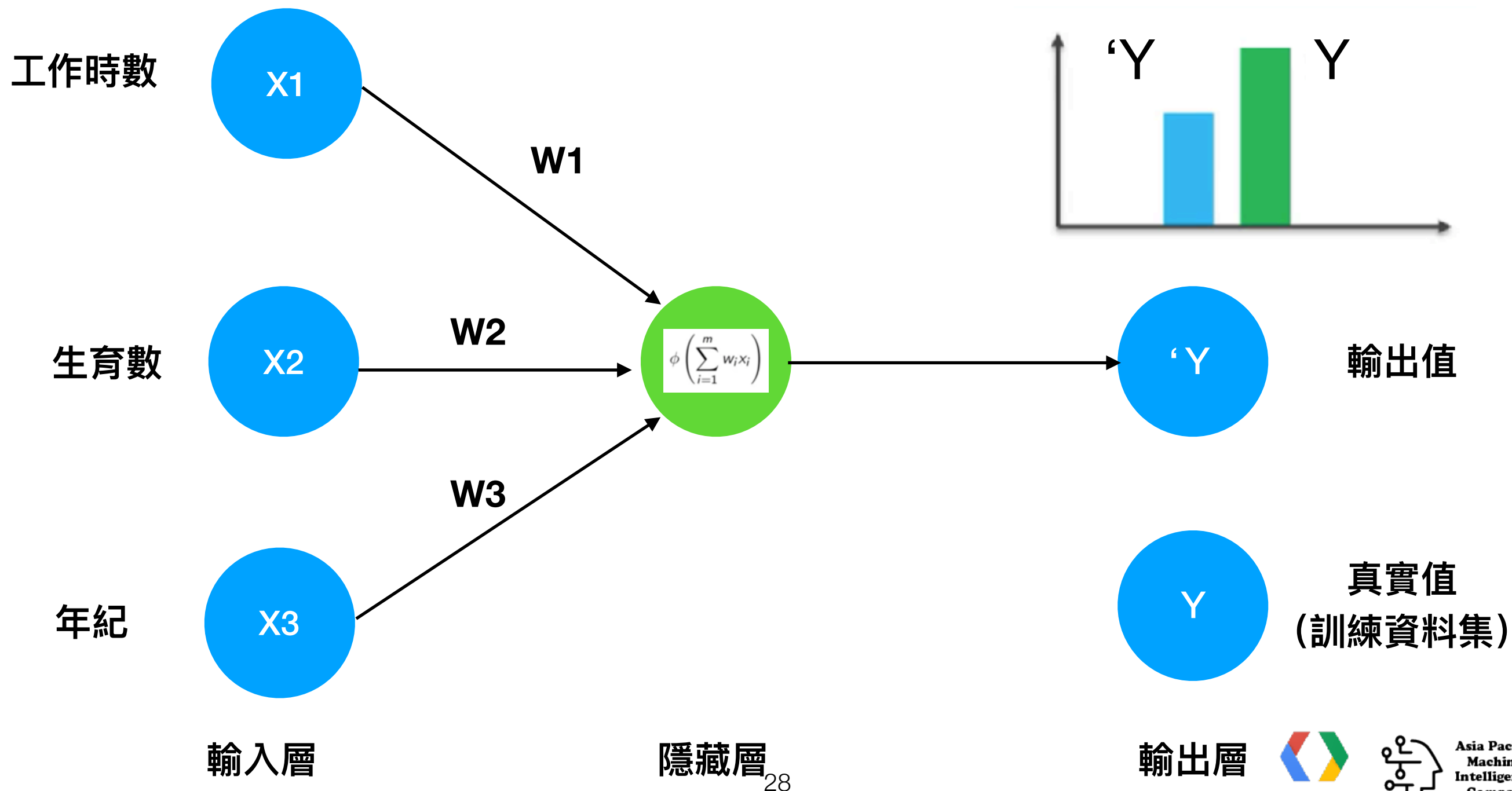


有偏移量

神經網路(Neural Network)

如何運作？

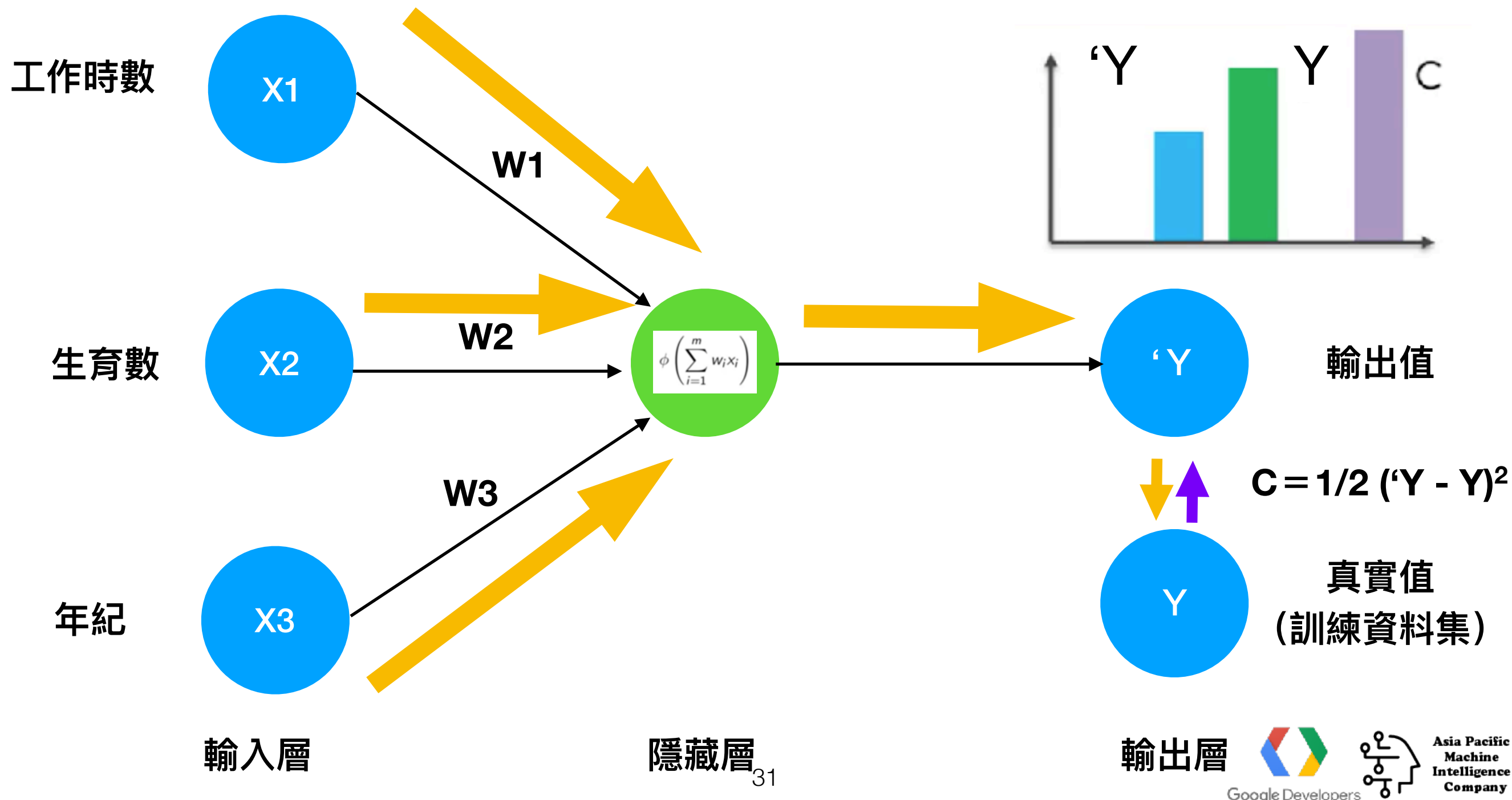
而當把每一次「輸出值'Y」與「真實值Y」相比，會得到一個差值，我們把它叫做 Cost，把這些 Cost 加起來，也就是整個訓練集的誤差，叫做成本函數(Cost Function)



神經網路(Neural Network)

如何運作？

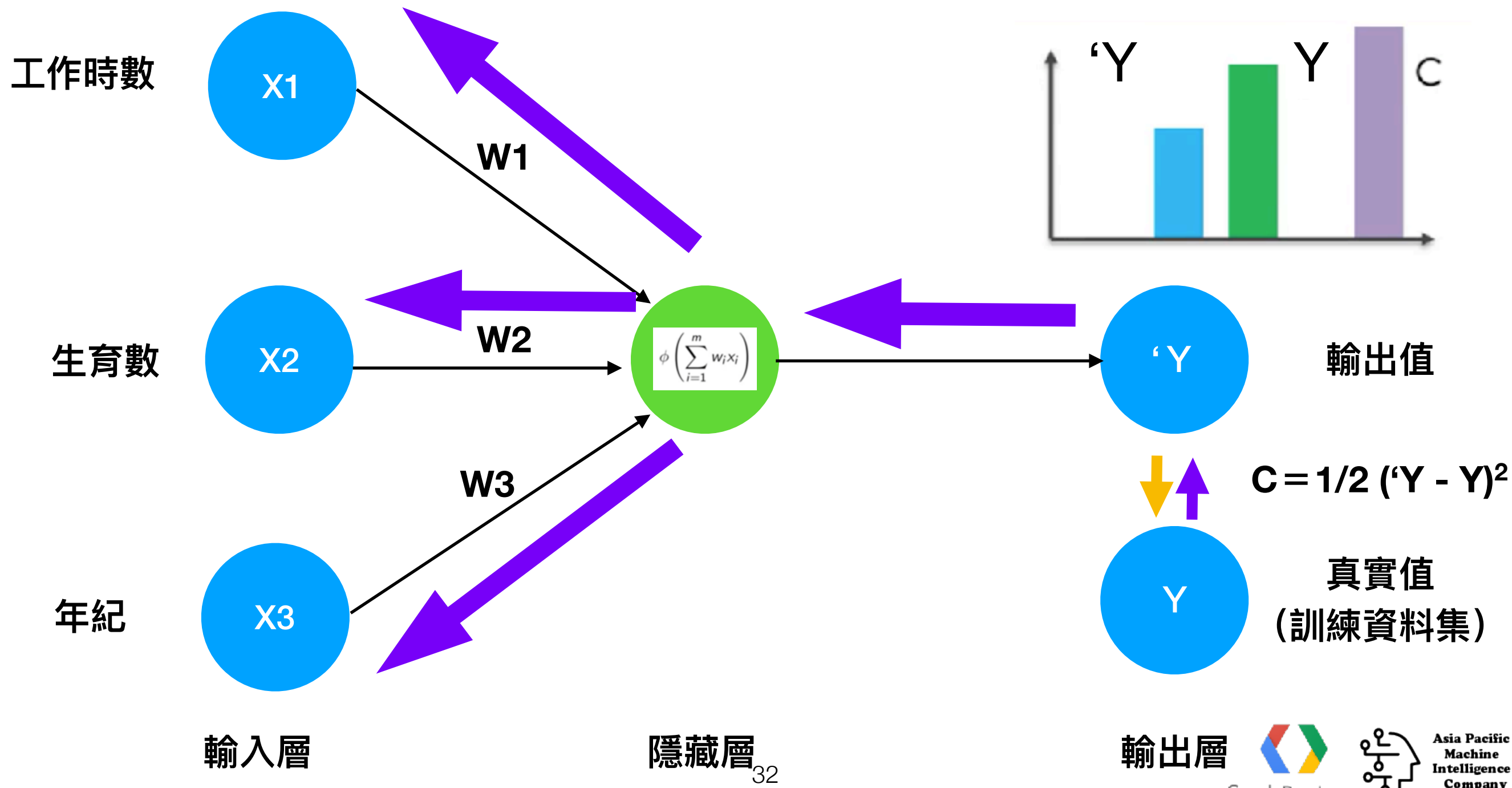
而當把每一次「輸出值'Y」與「真實值Y」相比，會得到一個差值，我們把它叫做 Cost，把這些 Cost 加起來，也就是整個訓練集的誤差，叫做成本函數(Cost Function)



神經網路(Neural Network)

如何運作？

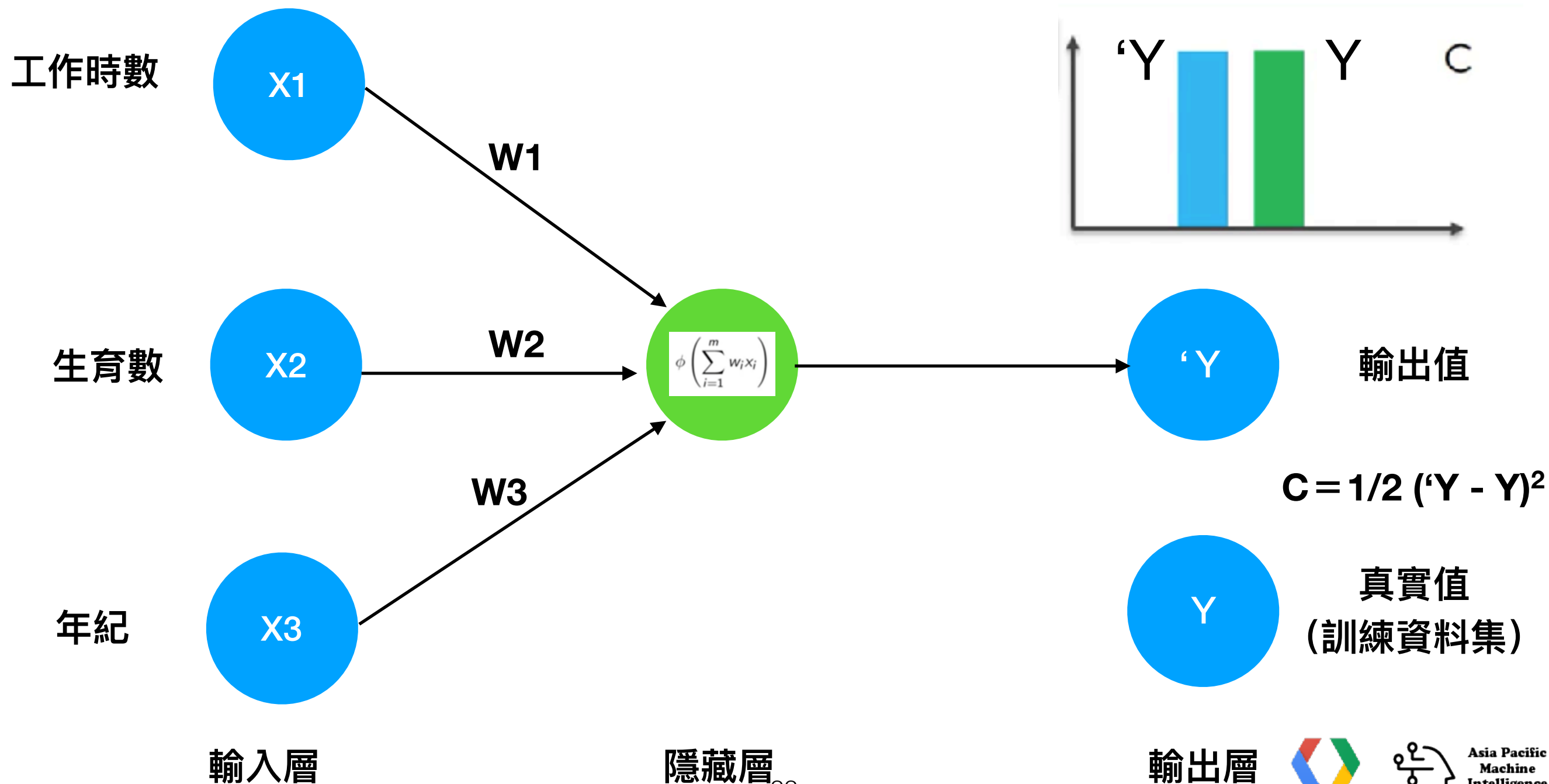
而當把每一次「輸出值'Y」與「真實值Y」相比，會得到一個差值，我們把它叫做 Cost，把這些 Cost 加起來，也就是整個訓練集的誤差，叫做成本函數(Cost Function)



神經網路(Neural Network)

如何運作？

而當把每一次「輸出值'Y」與「真實值Y」相比，會得到一個差值，我們把它叫做 Cost，把這些 Cost 加起來，也就是整個訓練集的誤差，叫做成本函數(Cost Function)

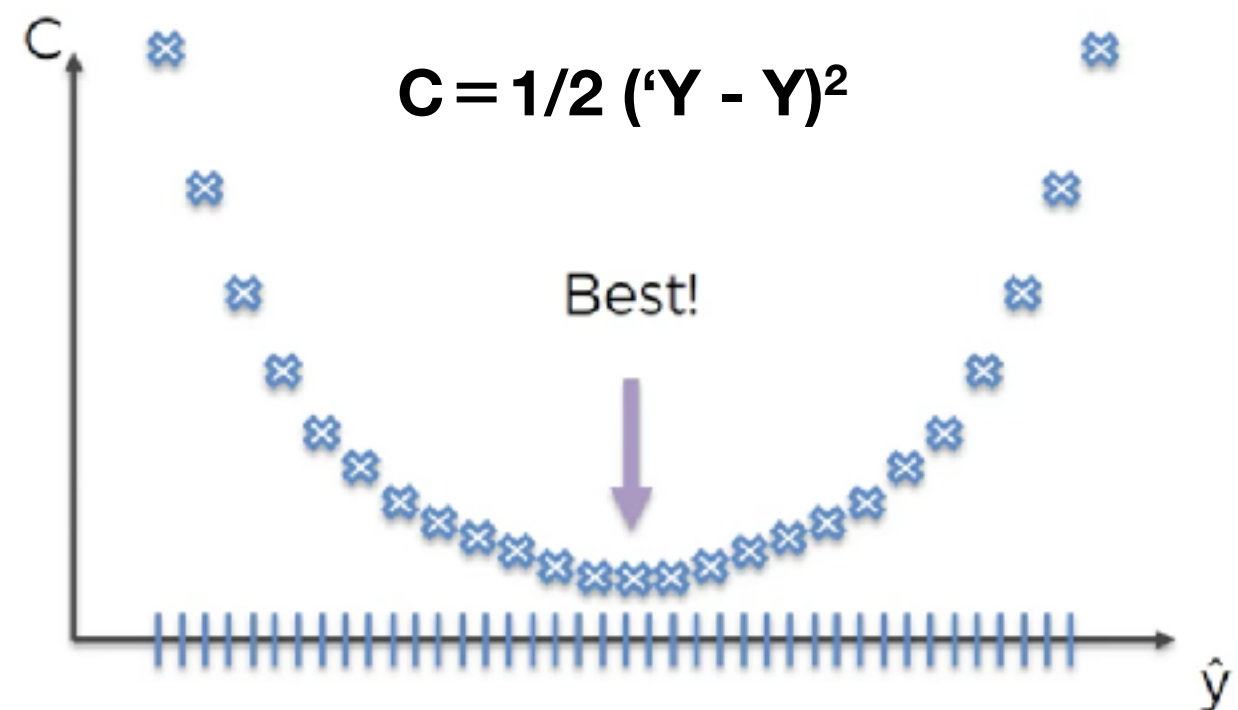


人工神經網路(ANN)

- 神經元(Neuron)
- 激活函數(Activation Function)
- 神經網路(Neural Network)如何運作？
- 神經網路(Neural Network)如何學習？
- 梯度下降(Gradient descent)
- 反向傳播(Backpropagation)

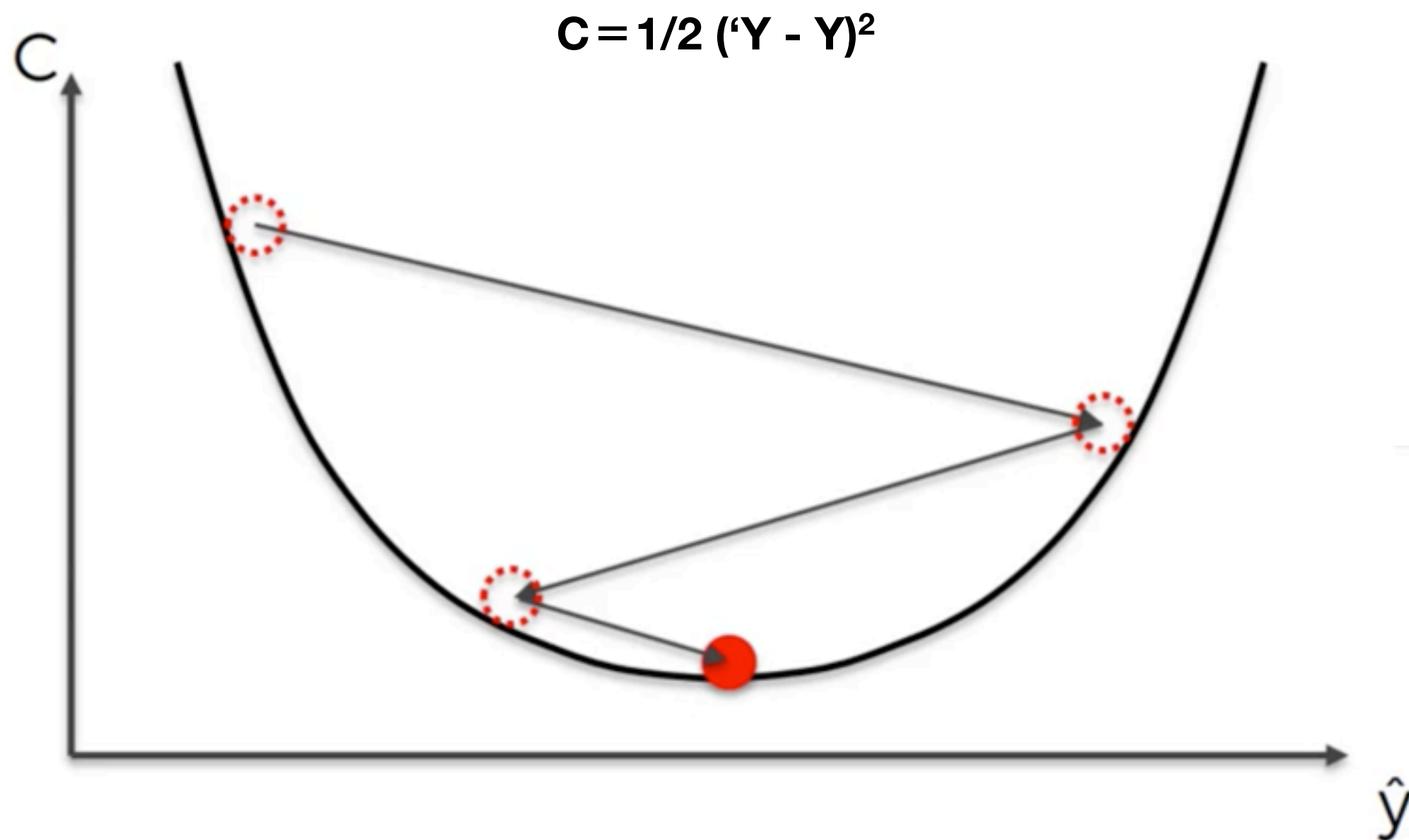
梯度下降(Gradient descent)

- 要使用梯度下降法找到一個函數的局部極小值，必須向函數上當前點對應梯度的反方向（往最小值）的規定步長距離點進行疊代搜索。
- 希望能夠透過梯度下降的方法，來求得最小損失函數 (Loss function)，也就是單一樣本的誤差。



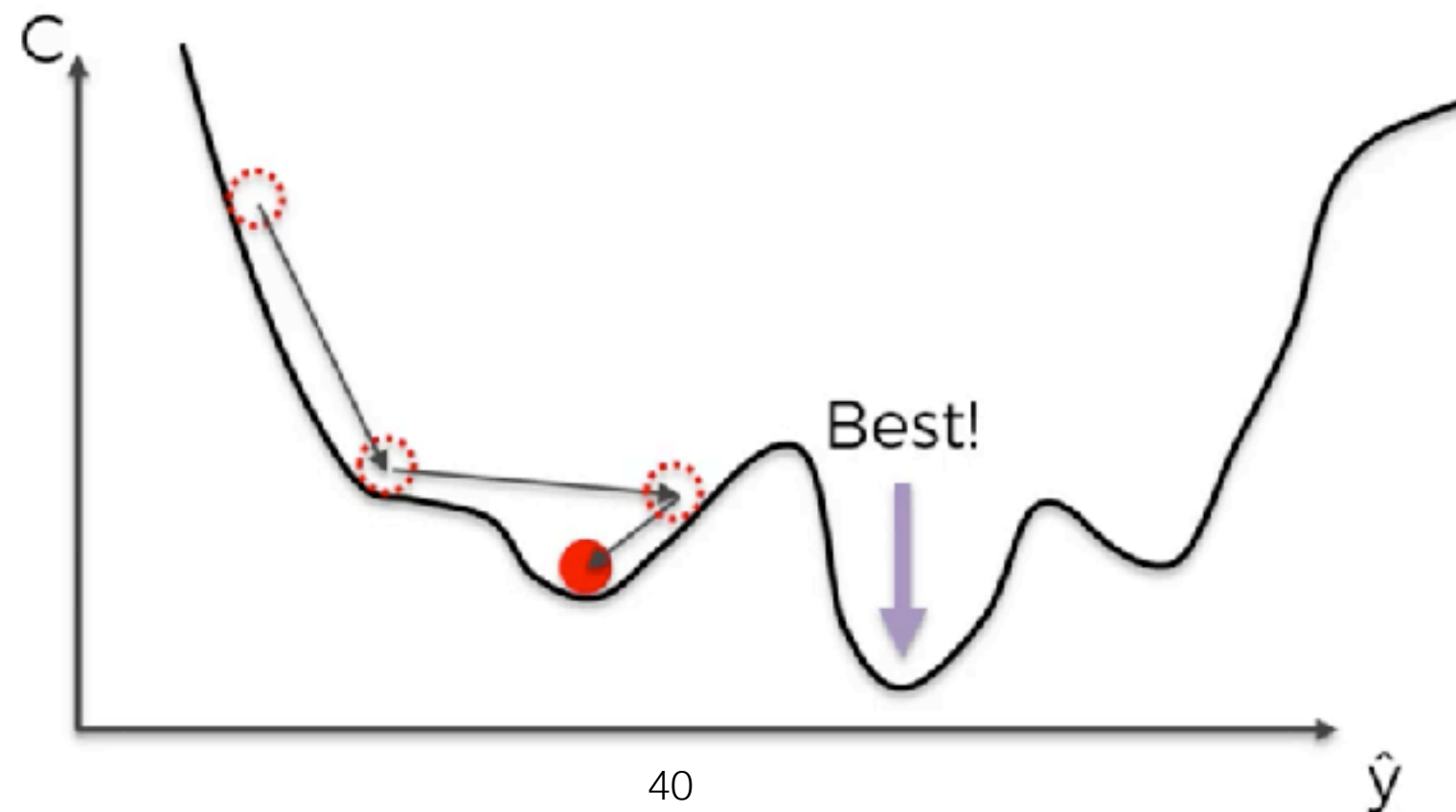
- 假設我們有1250個結果，每個都要用眼睛看，要耗費相當多的時間

梯度下降(Gradient descent)



梯度下降(Gradient descent)

- 每次迭代都用整批資料 叫做批量梯度下降
- 為了減少複雜度，當資料很大的時候，採用隨機梯度下降 (Stochastic gradient descent)



梯度下降缺點

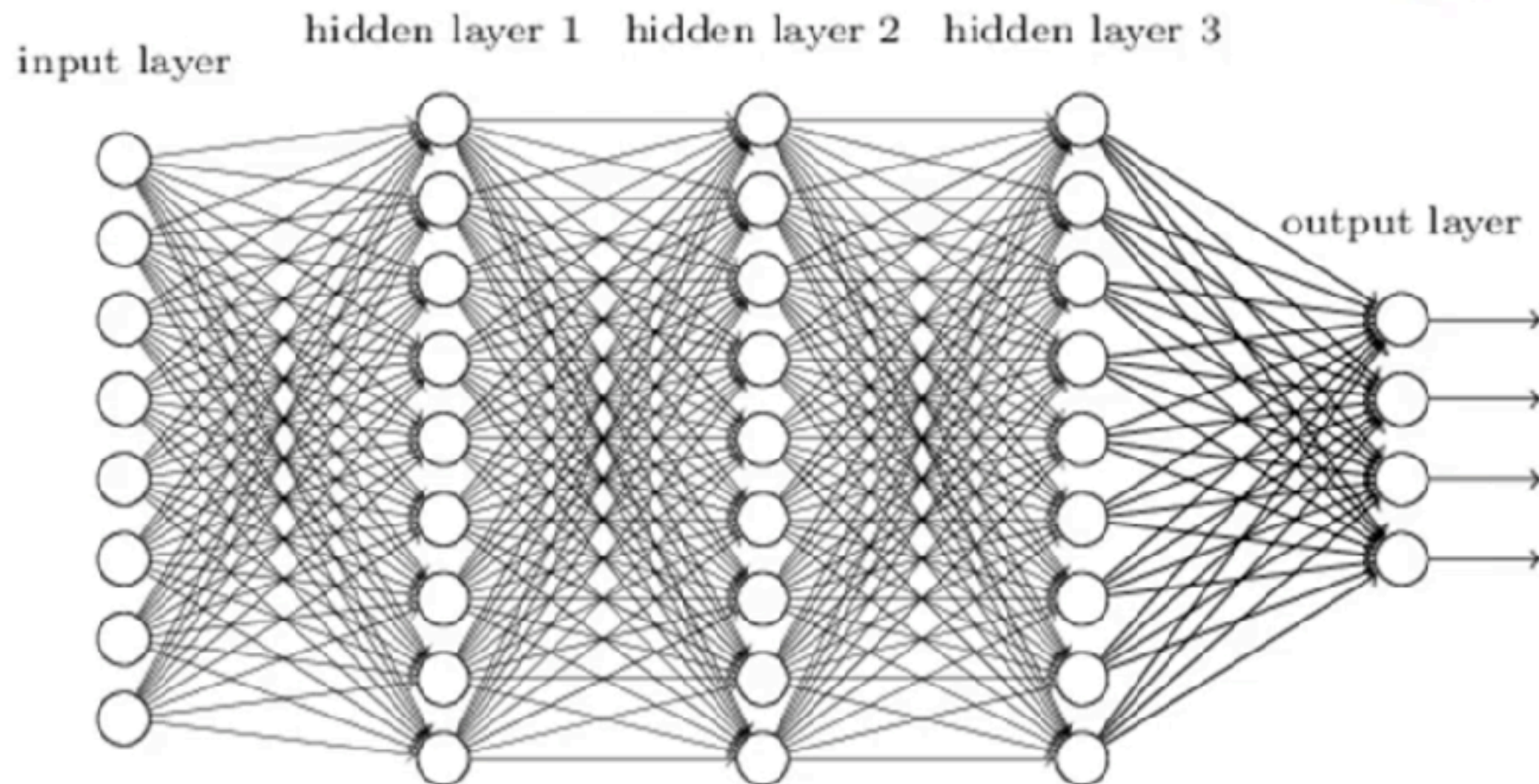
- 批次梯度下降：檢查全部數據算一次損失函數，然後更新梯度，每更新一次參數都要把數據集裡的所有樣本都檢查一遍，計算量大，計算速度慢，不支持線上學習。
- 隨機梯度下降：每看一個數據就算一下損失函數，然後更新梯度，方法速度比較快，但是收斂不太好，可能在最優點附近來回卻找不到最佳解。
- 為了克服兩種方法的缺點，現在一般採用「小批量梯度下降」，將數據分為好幾個批次，按照批次更新參數。
- 一個批次中的一組數據共同決定了本次梯度的方向，下降起來就比較聚焦，減少了隨機性。另外因為批次的樣本數與整個數據集相比小了很多，計算效率就可以增加。（但次批次大小的設定又是另一個問題了.....）

人工神經網路(ANN)

- 神經元(Neuron)
- 激活函數(Activation Function)
- 神經網路(Neural Network)如何運作？
- 神經網路(Neural Network)如何學習？
- 梯度下降(Gradient descent)
- 反向傳播(Backpropagation)

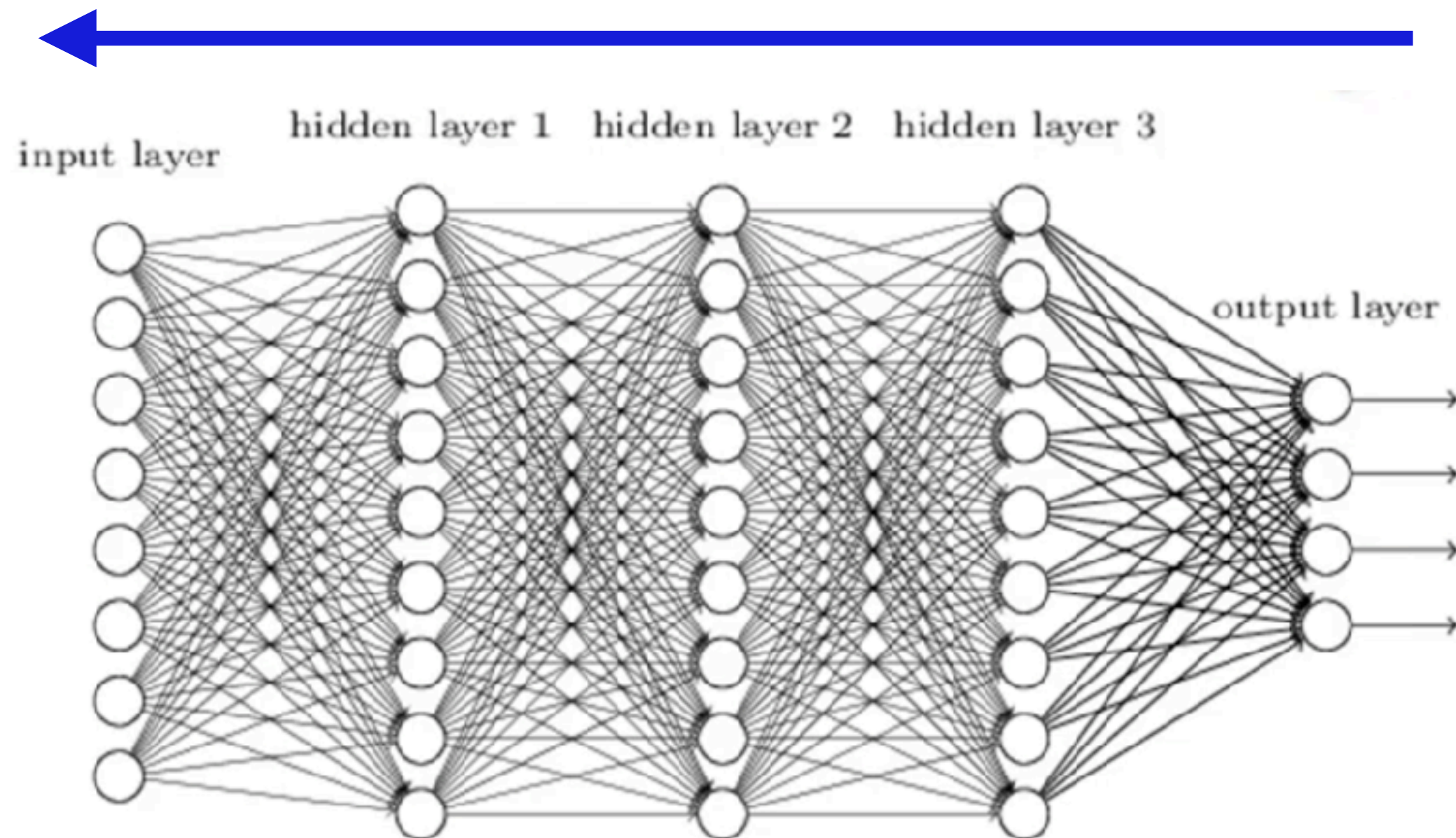
反向傳播(Backpropagation)

前向傳播(Forward Propagation)



反向傳播(Backpropagation)

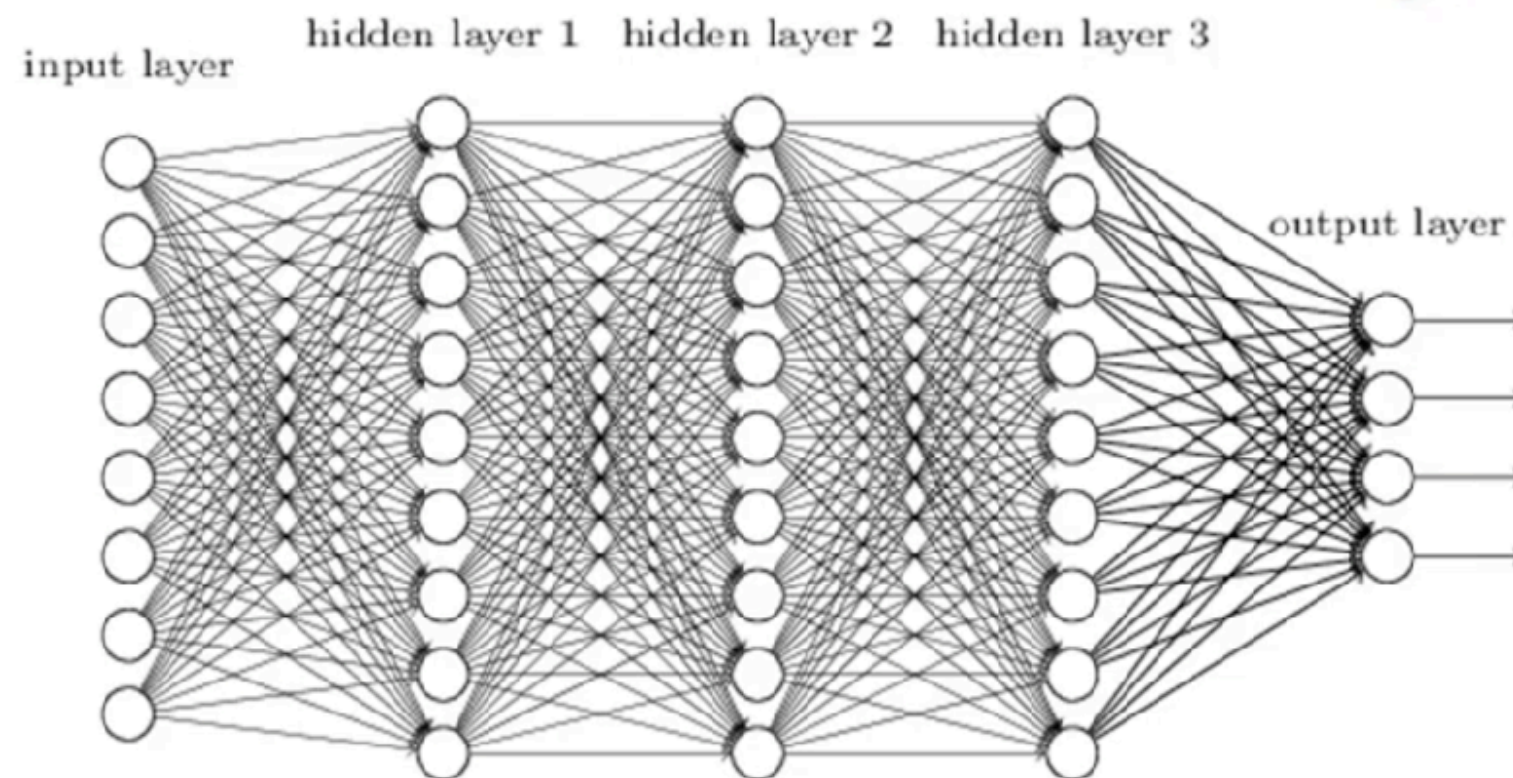
反向傳播(Back Propagation)



反向傳播(Backpropagation)

前向傳播(Forward Propagation)

反向傳播(Back Propagation)



前向傳播 x 反向傳播 = 獲得某一個權重的結果
如此才能透過 C 進行權重的更新

感謝聆聽

jerry@ap-mic.com

