各章概要

Chapter 1　開始接觸類神經網路

類神經網路是什麼東西？先來比較跟其他機器學習演算法的差異，再以圖片、簡單的數學式解說類神經網路的結構與能夠做到哪些事情。

Chapter 2　學習正向傳播

解說構成感知器類神經網路的單純演算法是如何進行運算，舉判別圖像尺寸為例，學習從輸入值到輸出值依序計算的「正向傳播」。

Chapter 3　學習反向傳播

說明在類神經網路上，如何求得適當的權重與偏差。使用微分更新權重與偏差，盡可能減少「誤差」，但正攻法的計算相當麻煩，因此我們會採用簡化計算的「誤差反向傳播法」。

Chapter 4　學習卷積類神經網路

學會類神經網路的基本原理後，接著學習使用卷積類神經網路處理圖像，舉出卷積類神經網路的特有機制、運算，並說明權重、偏差的更新方法。

Chapter 5　實作類神經網路

根據前面章節學到的類神經網路計算方法，使用 Python 編寫程式。以 Chapter 2、3 出現的基本類神經網路，實作圖像的尺寸判定；以 Chapter 4 出現的卷積類神經網路實作手寫文字辨識。

Appendix

收錄 Chapter 1 ～ 5 未能詳細解說的數學知識、Python 程式設計的環境設置、Python 與 NumPy 的簡易說明。

總和符號 / 微分 / 偏微分 / 合成函數 / 向量與矩陣 / 指數與對數 / Python 環境設置 / Python 基礎入門 / NumPy 基礎入門

碁峰 www.gotop.com.tw

Chapter

1

開始接觸類神經網路

對深度學習產生興趣的綾乃，

向友人美緒請求指導。

然而，綾乃對深度學習完全沒有概念，

美緒只好從頭開始說明。

在沒有困難數學式與程式的情況下，

和美緒她們一起學習

什麼是深度學習吧！

碁峰 www.gotop.com.tw

我也要！

Section 2 類神經網路的定位

首先，先畫出整體的俯瞰圖。

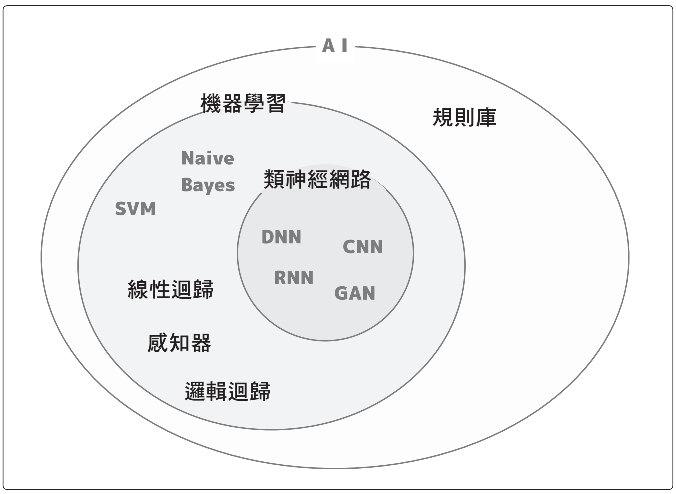

圖1-1

確認一下類神經網路在 AI 範圍中的什麼位置。

是機器學習的領域之一嘛。

妳聽過線性迴歸、感知器（perceptron）、邏輯迴歸嗎？類神經網路是這些機器學習演算法的同伴哦。

這樣啊！我還以為類神經網路是全新的東西。

碁峯 www.gotop.com.tw

類神經網路的歷史相當久遠，與其他機器學習相同，可求解迴歸、分類的問題。

因為最近很流行才誤會了，類神經網路並不是全新的技術嘛。

妳還記得迴歸和分類嗎？

回歸是處理連續值，比如，由過去的股價預測未來股價與趨勢。

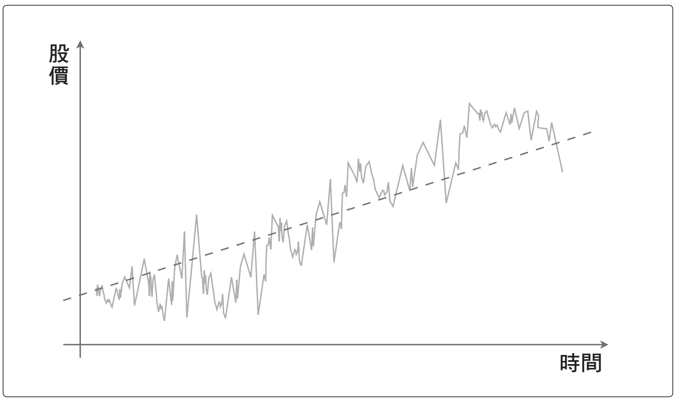

圖1-2

分類是處理非連續值，比如，將郵件分為「垃圾郵件」或「非垃圾郵件」的問題。

郵件的內容	是否為垃圾郵件
工作辛苦了。下個星期天預計去的……	×
和我交個朋友嘛。這邊有我的照片哦！ http://..	○
恭喜您抽中夏威夷旅行……	○

表1-1

www.gotop.com.tw

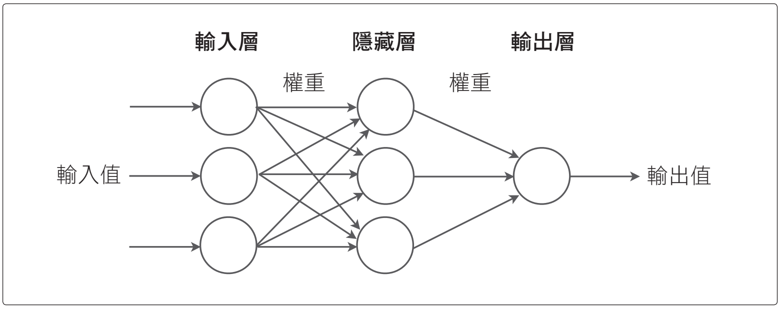

圖1-3

類神經網路形式上大多會以這樣的方式表達，圓圈表示神經元，也可稱為**單元（unit）**。

這種圖經常看到。從圓形部分畫出許多箭頭連結其他圓形。

這是輸入值加權後，向右進入單元層，最終輸出數值的動作示意圖。

隱約知道是從左向右前進，但輸入什麼樣的數值會輸出什麼樣的數值，過於抽象不好想像。

可以代入簡單的例子來幫助理解，比如……

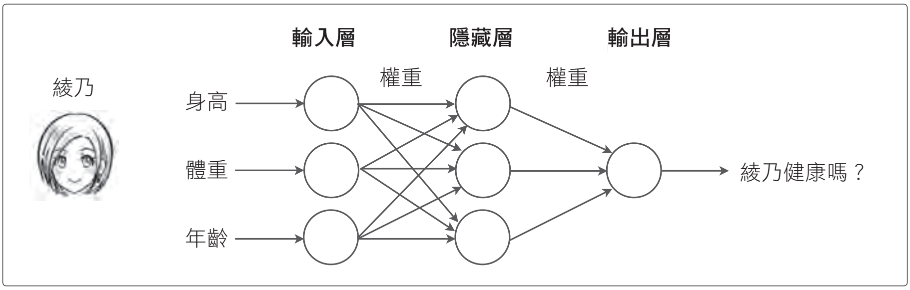

圖1-4

碁峰
www.gotop.com.tw

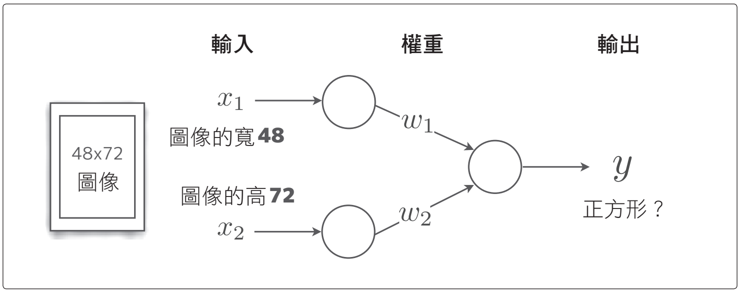

圖2-4

別忘了，與式 2-8 相同，y 是根據內積加上偏差的值輸出 0 或 1。

$$
y = \begin{cases} 0 & (\boldsymbol{w} \cdot \boldsymbol{x} + b \leq 0) \\ 1 & (\boldsymbol{w} \cdot \boldsymbol{x} + b > 0) \end{cases} \tag{2-16}
$$

跟剛才一樣計算寬和高的差值，看結果是否剛好等於 0 不就可以了嗎？

那麼，實際計算幾個例子看看吧。我想想……比如將這四個圖像輸入感知器。

$$
\boldsymbol{x}_a = \begin{bmatrix} 45 \\ 45 \end{bmatrix}, \quad \boldsymbol{x}_b = \begin{bmatrix} 96 \\ 96 \end{bmatrix}, \quad \boldsymbol{x}_c = \begin{bmatrix} 35 \\ 100 \end{bmatrix}, \quad \boldsymbol{x}_d = \begin{bmatrix} 100 \\ 35 \end{bmatrix} \tag{2-17}
$$

直接看數值可知：x_a 和 x_b 是「正方形」；x_c 和 x_d 是「非正方形」。

對。這是我刻意安排的。

碁峰 www.gotop.com.tw

當然，這樣就能分類，但這是知道正方形的特徵，才能列出來的式子吧？這條式子能夠用於正方形判定，但不能用於長邊判定。這樣的話，每次進行某種判定時，不就都得思考新的分類式子？

瞭解數據的特徵、規則，也就代表不需要機器學習。感知器的作業到底是在規則不明瞭的情況下，機器執行根據內積加上偏差的正負號決定結果。

對哦……就這個意義來說，正方形判定、長邊判定本來就是不使用感知器等機器學習的思維，也能夠求解的問題嘛。

沒錯。現在只是為了練習，舉簡單的問題來求解而已。

但是，沒辦法用感知器求解正方形的判定，只是因為權重 w 和偏差 b 的數值不好嘛？經過學習求得正確的數值後，應該就能夠正確分類吧？

很遺憾，感知器是解不開的哦！

Section 6 感知器的缺點

光由數學式沒辦法看出來，這邊畫出關係圖來幫助理解吧。

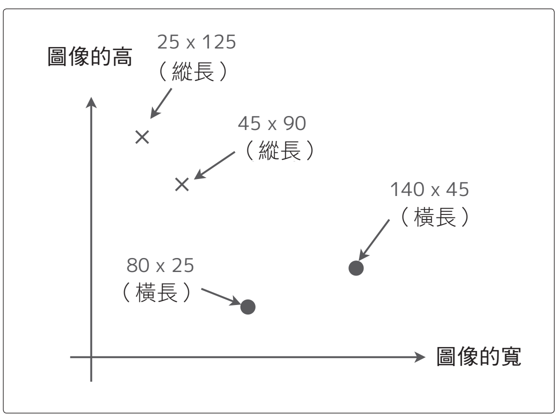

圖2-5

碁峰 www.gotop.com.tw

是的。這個問題無法只用一條直線分類哦。

果然沒辦法用一條來分類……

感知器不能解這種無法只用一條直線分類的問題。

無論怎麼改動權重 w 和偏差 b 都解不開？

解不開。

這、這樣啊……

變更權重、偏差的操作，實際上等同於變更圖中的直線。所以，一開始就無法以直線分類的問題，無論怎麼改動權重、偏差，結果還是無法分類哦。

對感知器來說，能不能用直線分類很重要的意思？

是的。可用直線分類的問題稱為**線性可分離**；無法用直線分類的問題稱為**線性不可分離**，這要好好記住哦！

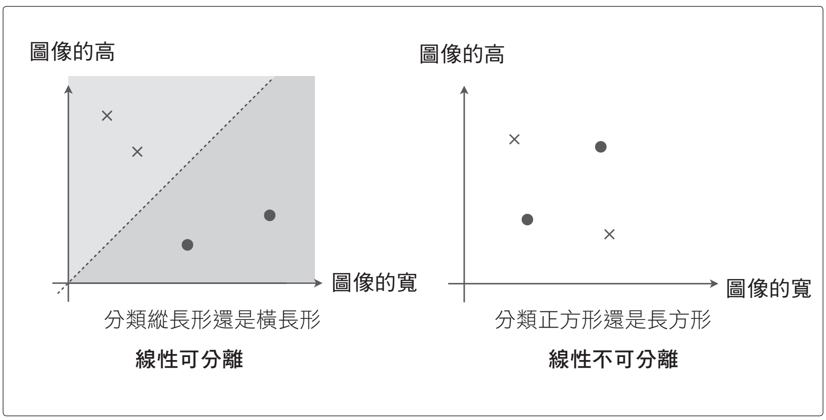

圖2-9

碁峯
www.gotop.com.tw

嗯。剛剛是分別計算 3 個感知器，得到判斷結果。

嗯。但是，我們通常不會這麼麻煩地一個個計算。還記得妳前面提到：「類神經網路感覺好像數學的函數」嗎？

$f(\boldsymbol{x}) = y$ 的函數嘛，將輸入值 $\boldsymbol{x}$ 代入類神經網路 f 得到結果 y。

妳不想從數學的角度，更詳細瞭解類神經網路 f 的內部嗎？

嗚……是想啊……但一聽到數學就會感到排斥。

呵呵。多複習幾次就行了，慢慢來理解吧。

美緒也一起的話，我就努力……看看。

後面的說明中，也會直接使用與剛才圖 2-12 相同的類神經網路。這邊將圖稍微畫得簡潔一些。

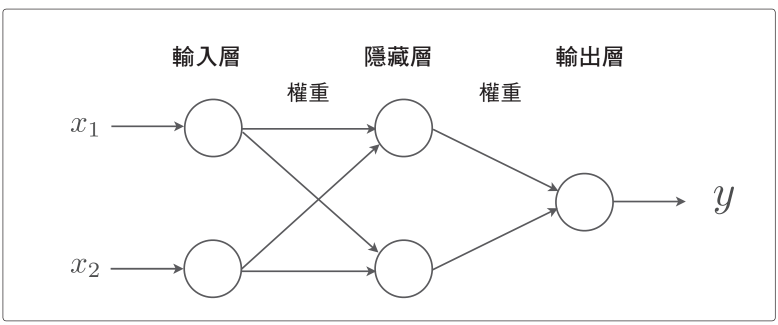

圖2-15

碁峯 www.gotop.com.tw

嗯，雖然重新畫過，但結構本身沒有改變嘛。

在類神經網路，存在單層感知器沒有的思維——「階層」。

就是輸入層、隱藏層、輸出層嘛。

是的。這邊會導入編號來識別階層，也就是令輸入層為 0，後面各層依序編號。比如，剛才的類神經網路會是：

- 假設輸入層為第 0 層；
- 假設隱藏層為第 1 層，其連接的權重為第 1 層的權重；
- 假設輸出層為第 2 層，其連接的權重為第 2 層的權重。

示意圖如下：

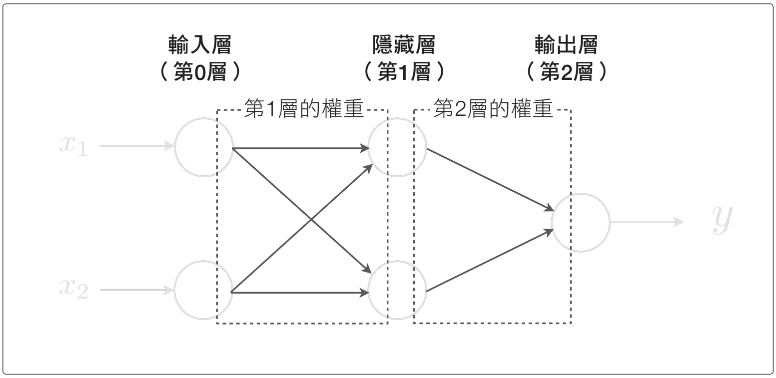

圖2-16

順便一提，類神經網路的階層結構，與權重、偏差數的關係為：

- 權重數＝連結階層間單元的線數；
- 偏差數＝該階層的單元數。

碁峰 www.gotop.com.tw

這跟美緒在式 2-23 做的計算相同嘛。最後會是權重和輸入值的內積加上偏差。

$$\boldsymbol{W}^{(1)}\boldsymbol{x}^{(0)}+\boldsymbol{b}^{(1)}=\begin{bmatrix} w_{11}^{(1)}\,x_1+w_{21}^{(1)}\,x_2+b_1^{(1)} \\ w_{12}^{(1)}\,x_1+w_{22}^{(1)}\,x_2+b_2^{(1)} \end{bmatrix} \tag{2-43}$$

沒錯。然後，套用第 1 層的激活函數：

$$\boldsymbol{a}^{(1)}(\boldsymbol{W}^{(1)}\boldsymbol{x}^{(0)}+\boldsymbol{b}^{(1)}) \tag{2-44}$$

傳到隱藏層的值，也就是式 2-34，可想成通過激活函數從隱藏層輸出哦。

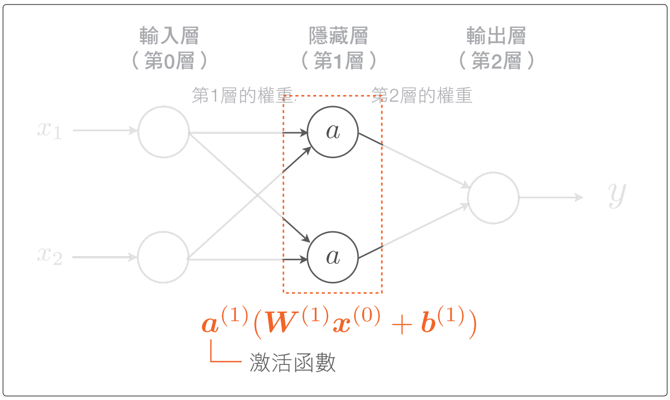

圖2-25

就是將第 1 層的激活函數 $a^{(1)}$ 套用到各元素嘛。

$$\boldsymbol{a}^{(1)}(\boldsymbol{W}^{(1)}\boldsymbol{x}^{(0)}+\boldsymbol{b}^{(1)})=\begin{bmatrix} a^{(1)}(w_{11}^{(1)}\,x_1+w_{12}^{(1)}\,x_2+b_1^{(1)}) \\ a^{(1)}(w_{21}^{(1)}\,x_1+w_{22}^{(1)}\,x_2+b_2^{(1)}) \end{bmatrix} \tag{2-45}$$

沒錯。然後，取從第 1 層到第 2 層輸入值的意思，以文字記為 $\boldsymbol{x}^{(1)}$。這也可視為第 1 層的輸出值。

$$\boldsymbol{x}^{(1)}=\boldsymbol{a}^{(1)}(\boldsymbol{W}^{(1)}\boldsymbol{x}^{(0)}+\boldsymbol{b}^{(1)}) \tag{2-46}$$

www.gotop.com.tw

補充說明

為什麼必要？

我嘗試了不使用激活函數會發生什麼事。

喔！原來如此，不錯的嘗試。

對吧。比如，假設一個像這樣單純的類神經網路：

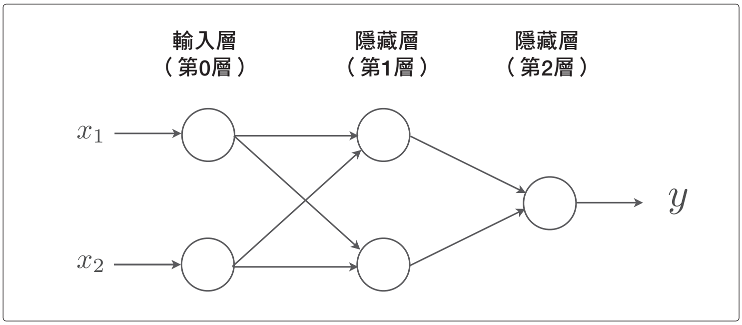

圖2-c-1

這個類神經網路是由輸入 $\boldsymbol{x}$ 和兩個權重矩陣 $\boldsymbol{W}^{(1)}$、$\boldsymbol{W}^{(2)}$ 所構成。

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \boldsymbol{W}^{(1)} = \begin{bmatrix} w_{11}^{(1)} & w_{12}^{(1)} \\ w_{21}^{(1)} & w_{22}^{(1)} \end{bmatrix}, \boldsymbol{W}^{(2)} = \begin{bmatrix} w_{11}^{(2)} & w_{12}^{(2)} \end{bmatrix} \tag{2-c-1}$$

順便一提，偏差對這個嘗試來說可有可無，姑且先忽視它。

這樣的話，類神經網路的正向傳播計算，能夠用這個式子描述：

碁峰 www.gotop.com.tw