

# 非監督學習與類神經網路

2018.12.20





# 基本精神

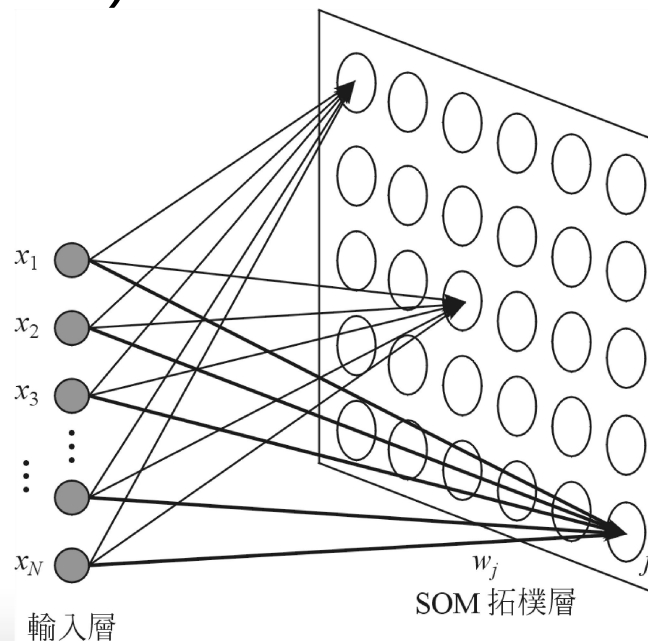
- 非監督式的類神經網路在缺乏期望輸出值的情況下，能夠自行發掘出資料中的那些特徵是重要的或是可忽略的，以便將資料作“群聚” (clustering) 的處理。
  - 這些特徵是根植於非監督式學習
  - 此種演算法多用於聚類型的類神經網路，可降低高維度系統的複雜性



# 自組特徵映射網路

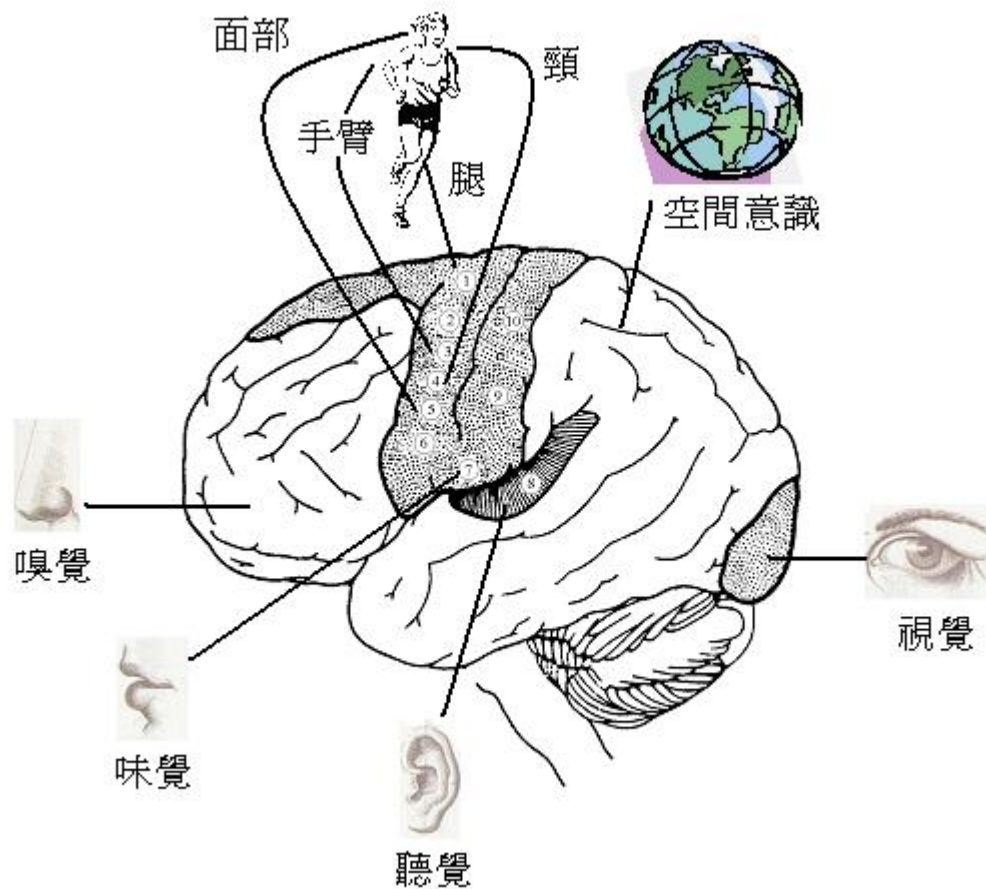
( Self-Organizing Map, SOM )

- 首先由 Kohonen 提出，屬於前饋式、非監督式神經網路
- 以特徵映射的方式，將任意維度的輸入向量，映射至較低維（度）的特徵映射圖上





# 大腦皮質中的特徵映射





# SOM網路架構

- 輸入層

- 用以表現網路的輸入變數，即訓練範例的輸入向量，或稱特徵向量，其處理單元數目依問題而定，每一個處理單元代表著輸入向量的每一個元素，亦即該輸入資料所擁有的特徵。

- 輸出層

- 用以表現網路的輸出變數，及訓練範例的聚類，其處理單元數目依問題而定。其結構本身有『網路拓樸』以及『鄰近區域』(Neighborhood)的觀念。



- 依據目前的輸入向量在神經元間彼此相互競爭，優勝的神經元可獲得調整連結權重向量的機會；
- 而最後輸出層的神經元會依據輸入向量的「特徵」以有意義的「拓樸結構」（topological structure）展現在輸出空間中，
- 由於所產生的拓樸結構圖可以反應所有輸入值間的分布關係，因此將此網路稱作為自組特徵映射網路，而該映射圖也可稱為拓樸圖（topology）。





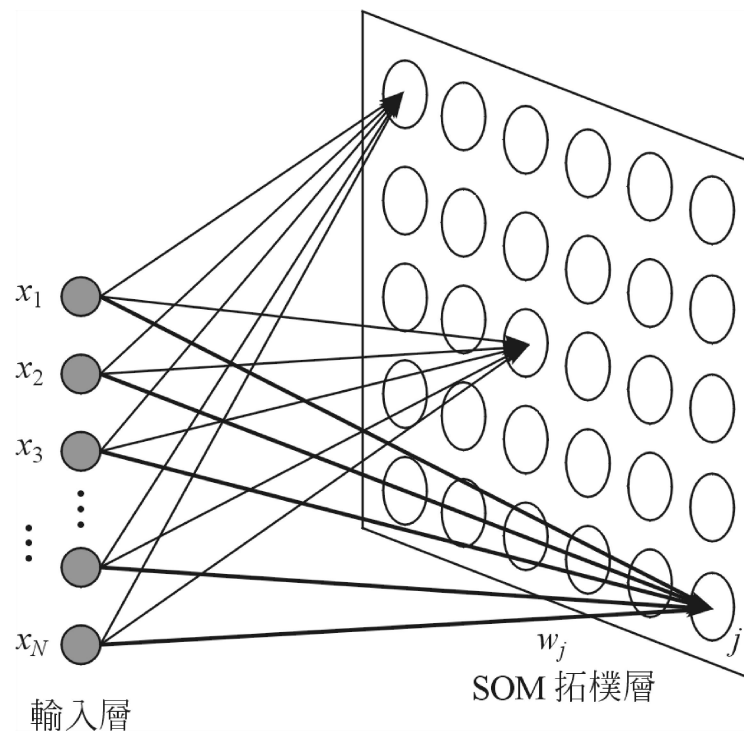
# 演算法

- 對於  $N$  個維度的輸入值，以  $X$  表示輸入向量

$$X = [x_1, x_2, \dots, x_N]^T$$

- 第  $j$  個神經元的連結權重

$$w_j = [w_{j1}, w_{j2}, \dots, w_{jN}]^T$$
$$j = 1, 2, \dots, M$$





# 演算法

- 計算距離，並選取出優勝神經元

$$q(X) = \min_{\forall j} \|X - w_j\|_2 \quad j=1,2,\dots,M$$

$q(X)$  為輸入向量 $X$ 與所有神經元連結權重的最短距離  
該神經元則稱為優勝神經元

- 每一筆輸入向量所對應出的優勝神經元都不盡相同，因此每個神經元被調整的次數及時機也不一定相同或有規則可循，完全端視輸入向量間的分布關係。





- 競爭式學習法則
  - 每筆輸入向量都必須尋找其對應的優勝神經元，即與該輸入向量最近似的神經元，進而調整該神經元的連結權重。
- 距離計算公式
  - 比較所有神經元的連結權重與輸入向量間的距離
  - 歐幾里德基距離公式（簡稱歐氏距離）、加權距離公式、Manhattan距離公式等



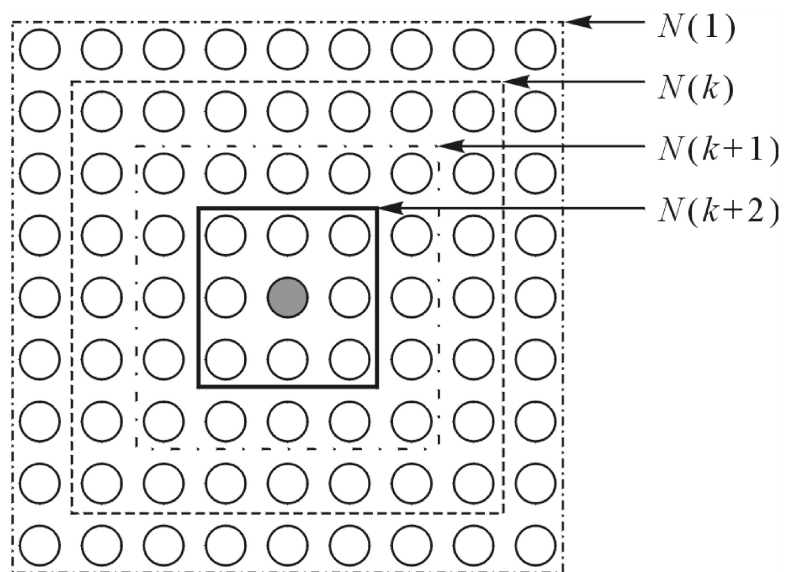
# 鄰近區域

- SOM網路學習的過程中，有一個重要的關係存在於網路的神經元間.....

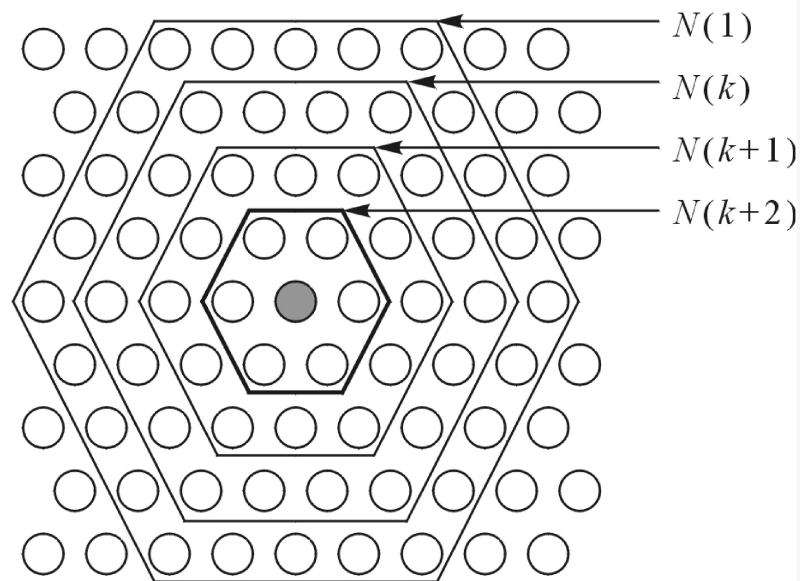
神經元間有著鄰近關係，讓優勝的神經元在進行連結權重調整時，也會將這樣的訊息傳遞給鄰近的神經元，讓鄰近的神經元也隨著進行連結權重調整，如此有助於網路神經元間的拓樸映射關係。



## 拓樸層間鄰近神經元及鄰近半徑遞減示意圖



(a) 矩形的鄰近區域



(b) 六角形的鄰近區域



# 範例

- kohonen package in R
- MBA admission data



## 從另一個角度來看 .....

- 自組織特徵映射的過程也是一個聚類的過程
- SOM可視為聚類演算法 ( clustering algorithm ) 的一種。
  - 可將一群未經標示的樣本，透過此演算法，從中尋找某些相似的特性，然後再將這些具有相似特性的樣本聚集成一類。



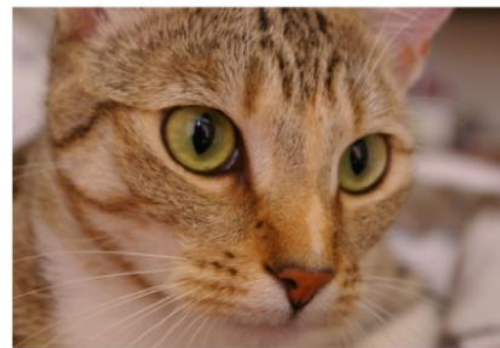
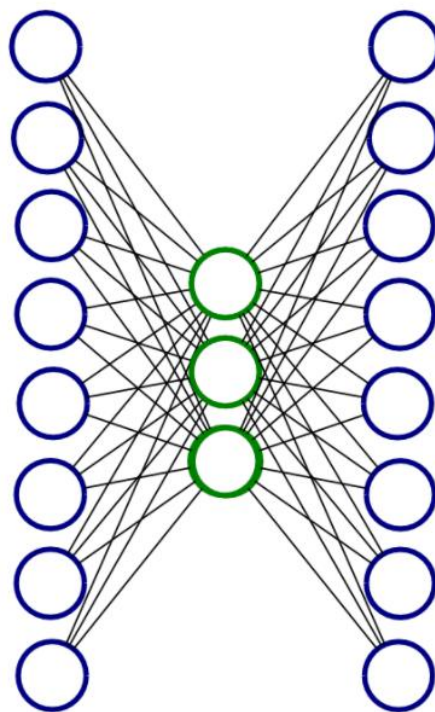
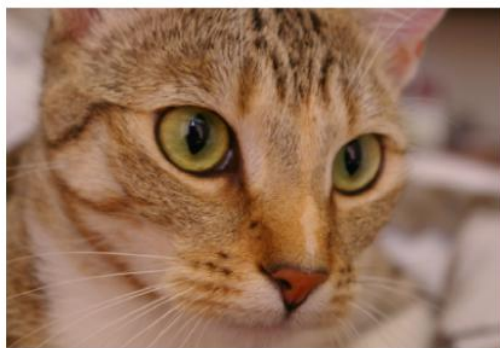
# 自動編碼器 (Autoencoder)

- Autoencoder 簡單來說就是將有很多 **Feature** 的資料進行壓縮，之後再進行解壓的過程。本質上來說，它也是一個對資料的非監督學習
- 它的主要功能即對資料進行非監督學習，並將壓縮之後得到的“特徵值”
- 之後再將壓縮過的“特徵值”進行解壓，得到的最終結果與原始資料進行比較，對此進行非監督學習。





# 編碼與解碼 (壓縮與解壓縮)

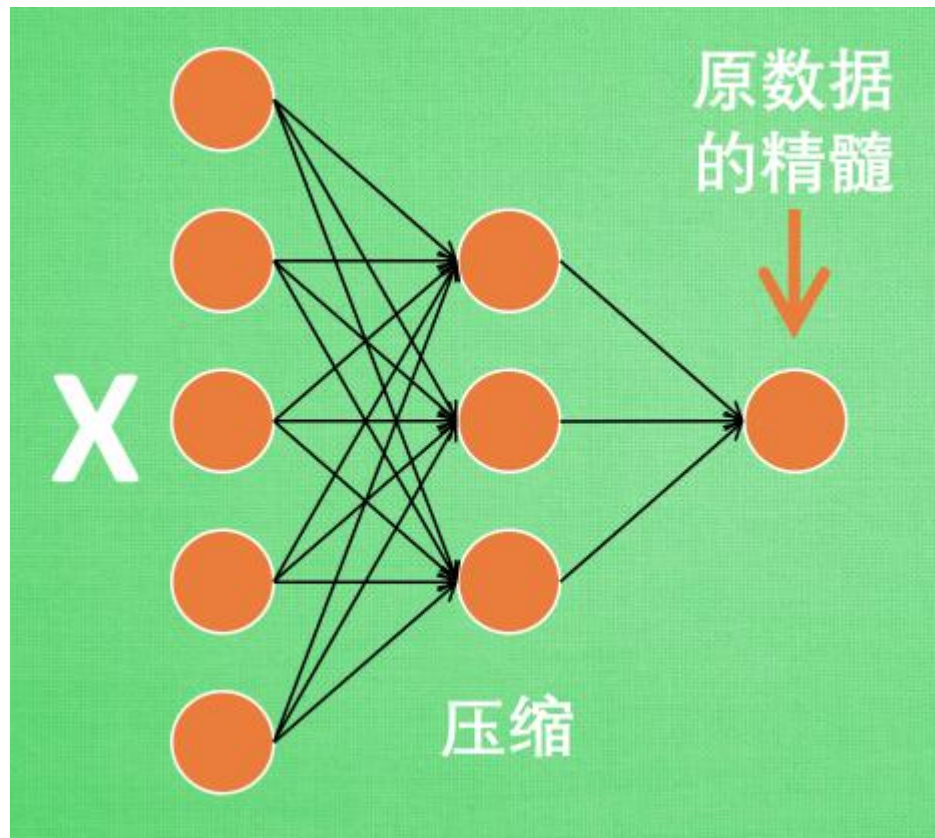




- 有時神經網路要接受大量的輸入資訊，比如輸入資訊是高畫質圖片時，輸入信息量可能達到上千萬，讓神經網路直接從上千萬個資訊源中學習是一件很吃力的工作。
- 所以，何不壓縮一下，提取出原圖片中的最具代表性的資訊，縮減輸入信息量，再把縮減過後的資訊放進神經網路學習。這樣學習起來就簡單輕鬆了，所以自編碼就能在這時發揮作用。

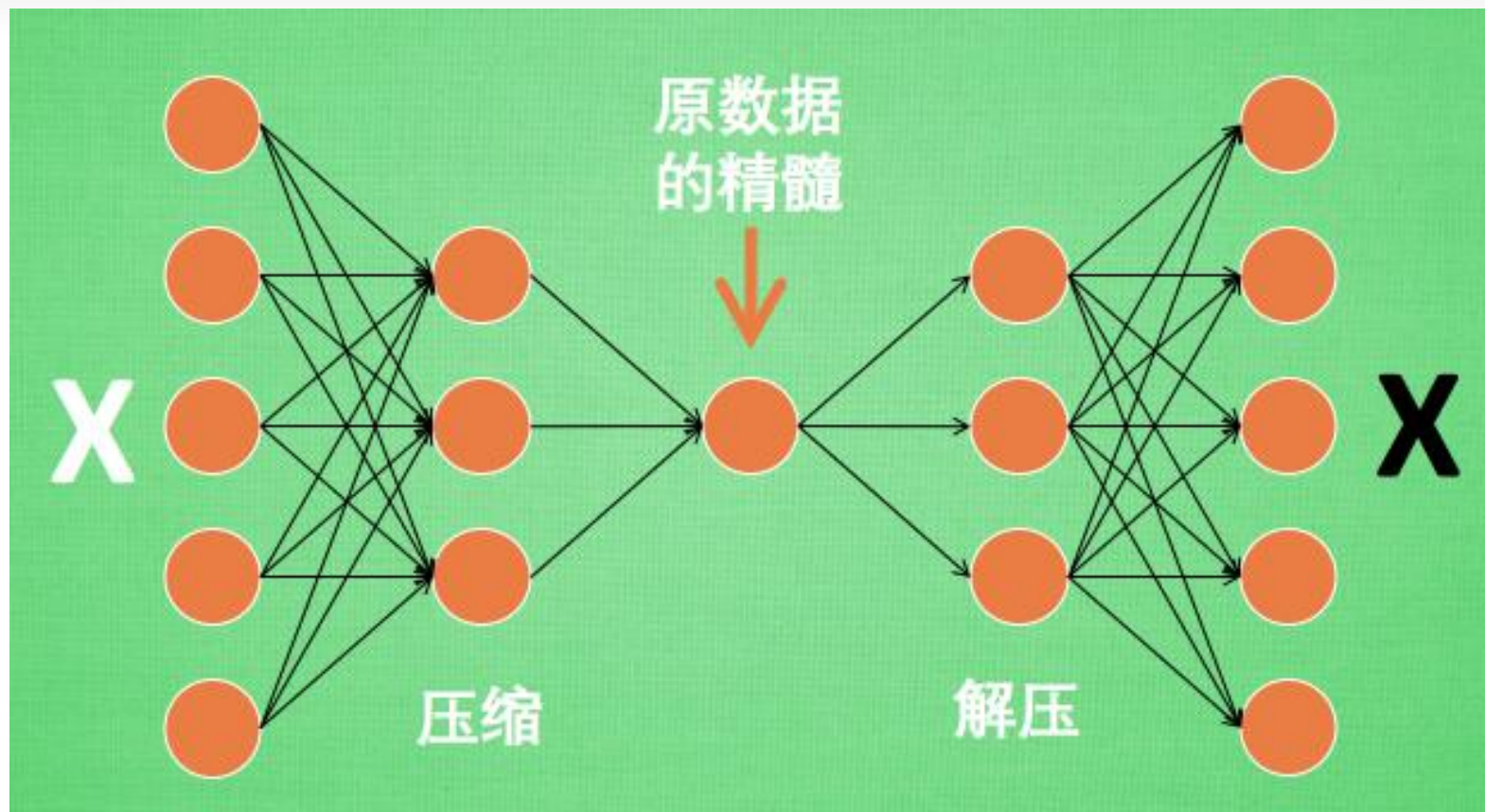


# 編碼器 (Encoder)





# 解碼器 (Decoder)







# 範例

- autoencoder package in R
- MBA admission data