群集分析 (Clustering)

定義

依據資料相似度(similarity)或相異度(dissimilarity)將資料分群歸屬到群集(clusters)，使同一群內資料或個體相似程度大，各群間的相似程度小，屬於非監督式學習。群集分析的方法雖多，但主要關心下列三個問題:

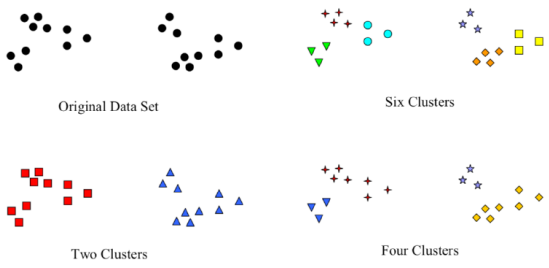
1. 如何以數量來表示事物(包括人)和事物之間的相似性(similarity)。
2. 如何根據這些相似性指標將類似的個體分成同一群集。
3. 所有事物分類完畢後，對於每一群聚性質應如何描述。

群集分析的過程:

1. 資料準備與特徵選取: 根據問題特性、資料類型及分群演算法等，選取具代表性的變數作為分群特徵屬性。
2. 相似度計算: 在選擇衡量相似度的方法時，需考慮資料的類型以及後續使用的分群演算法。
3. 分群演算法: 為群集分析中最重要的階段，利用分群演算法將資料分組，有些分群演算法可能需要自行決定群數。
4. 分群結果評估與解釋: 當分群結束後須檢視分群結果是否合理。另外，由於分群結果可能作為另一個方法的輸入資料，需要對群集結果進行定義或命名。

群集分析的限制與要求:

1. 群集是模稜兩可的事情，要定義分群取決於資料的特性與使用者的預設立場。



2. 可度量性(scalability): 許多群集的方法運用在少量資料的分群效果很好，但對於龐大的資料其結果會造成偏差(bias)。
3. 發現任意形狀群體的能力: 基於距離的群集演算法往往發現的是球形的群集，然而現實的群集可能是任意形狀的。
4. 決定輸入參數的最少領域知識: 許多群集方法都需要輸入參數，然而參數粉難決定，尤其是對於高維度資料，這使得群集的結果品質很難控制。
5. 處理雜訊資料的能力: 對空缺值、離異值、資料雜訊不敏感。

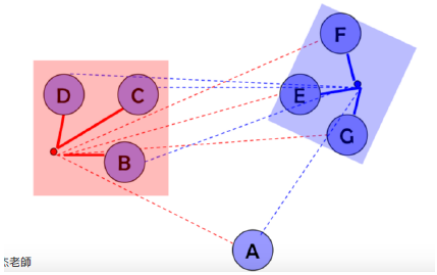
6. 可解釋性和可用性: 使用者會希望群組的結果具解釋性、可了解性與使用性。

群集方法

分割式方法 (partitioning method)、階層式的方法(hierarchical method)、基於密度的方法(density-based methods)

I. 分割式方法 (partitioning method)

事先挑選群集核心和訂定臨界值(群集數目),所有 objects 與該群集核心之距離只要沒有超過臨界值，一律併入該群聚內，否則屬於其他群集。
採不重疊的方法劃分，將原有的資料分到不同的群集中。
每個群集至少包含一個物件；每個物件對象僅屬於一個群集。
同一個群集中的對象盡可能的接近或相關,不同群集中的對象盡可能地遠離或不同。



常見距離公式:

- 歐幾里得(Euclidean)距離

$$d(i,j)=\sqrt{(|x_{i1}-x_{j1}|^2+|x_{i2}-x_{j2}|^2+...+|x_{ip}-x_{jp}|^2)}$$

- 曼哈頓(Manhattan)距離

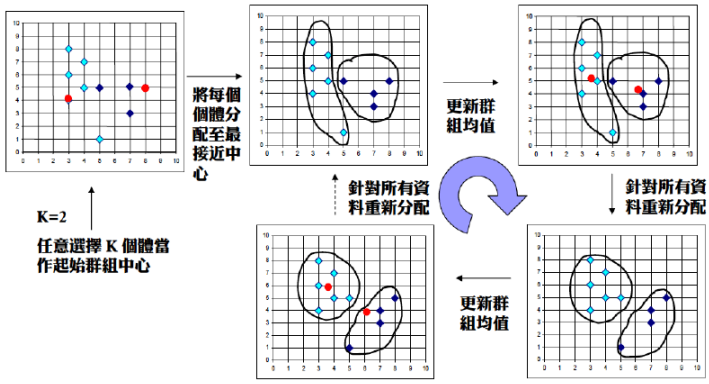
$$d(i,j)=|x_{i1}-x_{j1}|+|x_{i2}-x_{j2}|+...+|x_{ip}-x_{jp}|$$

- 明可夫斯基(Minkowski)距離

$$d(i,j)=\sqrt[q]{(|x_{i1}-x_{j1}|^q+|x_{i2}-x_{j2}|^q+...+|x_{ip}-x_{jp}|^q)}$$

1. K-Means

A. 將 n 筆待分群資料依資料特性，選出 K 個資料點(假若資料可被分成 K 個群集)，而這 K 個資料點即為 K 個群集的中心點(Centroid 質心，初始中心點可選用現有的資料點或隨機挑選可行座標點來表示)
B. 將所有資料與此 k 個中心點做距離運算，若某資料 j 與 k 個中心點當中之一距離最近，則此資料 j 可被歸類於該中心點所表示之群集 i。
C. 若所有資料皆被歸類完畢，則重新計算 k 個群集的中心點。
D. 回到第二步驟重新執行，直到滿足條件為止(目標函數 J 與計算結果與前次循環計算結果保持不變)。



K-Means 法主要目標: 求取各個輸入資料 Xi 與其相對應群集中心點 Ci 的距離平方和之最小值(誤差平方和 SSE)

$$J = \sum_{i=1}^{k} J_i = \sum_{i=1}^{k} \sum_{j=1}^{n} w_{ji} \left\| X_j - C_i \right\|^2$$

處理類別型態資料時可採 K-mode 方法，群集中心為眾數，頻率為基礎 (frequency-based)。

$$d(i,j) = \frac{p-m}{p}$$

- **m: 匹配的數目，即對象i和j取值相同的變數的數目 (也可加上權重)**
- **p: 類別變數的個數**

易受到離群值或雜訊影響，可改用 k 中心點法(k-medoids)

2. K-means++

A. 初始化一個空的集合 M 來儲存被選取的質心。

B. 從輸入的樣本中，隨機選取第一個質心，並加入 M。

C. 對於每一個不在 M 裡的樣本 xi 計算出對 M 中所有質心的最小距離平方 $d(xi, M)^2$

D. 使用加權機率分配

$$\frac{d\left(x^{(i)}, M\right)^2}{\sum_i d\left(x^{(i)}, M\right)^2}$$

來隨機選取下一個質心，並加入 M

E. 重複步驟 3, 4 直到選取了 k 個質心

F. 以古典 k-means 演算法完成後續工作

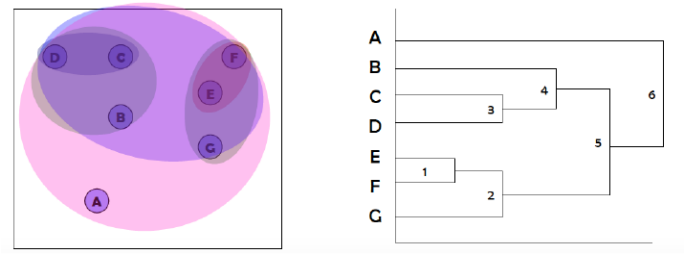
(在選取初始質心時，盡可能讓他們彼此遠離)

- k-means 缺點:
  A. 無法直接處理類別型資料，可改用 K 眾數法 (k-mode)
  B. 易受到離群值或雜訊影響，可改用 k 中心點法(k-medoids)
  C. 須事先決定群集數目
  D. 起始群集中心選擇的不同會造成不同的分群結果
  E. 無法適用於所有的資料群集型態
  F. 當群集間的特性非常相似時，可改用柔性群集(soft clustering)

3. Fuzzy C-Means
   其目標函數定義如同 k-means 法，，但應用了模糊理論的概念，使得每一輸入資料不在僅以是否歸屬於某一特定的群集，而以其歸屬程度來表現屬於個群集的程度。

II. 階層式群集

通常可用樹狀圖(dendrogram)表示，可顯示群集-子群集的關係，以及群集被合併/分割順序。分成凝聚式和分裂式。



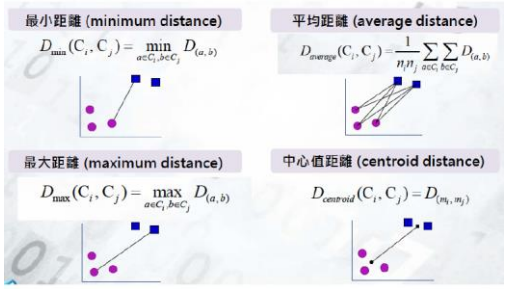
凝聚式: 一開始將每個對象視為單獨一組，然後相繼合併鄰近的對象或組別，直到所有的組別合併為一個，或者達到終止條件。這需要定義群集鄰近值(cluster proximity)的概念。

分裂式:

開始將所有的對象至於同一個群集中，在迭代的每一步，一個群集被分裂為多個更小的群集，直到最終每個對象在一個單獨的群集中，或達到一個終止條件。

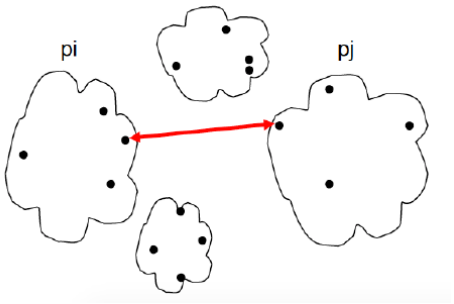
1. 凝聚式階層式群集

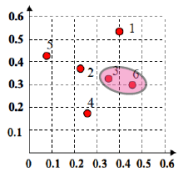
分成 Min(單一連結法)、Max(完全連結法)、群平均、中心值距離、華德法



A. 單一連結法 (single linkage method)

將不同群集中之兩個最近的資料點之間的距離，當成是群集的距離值。

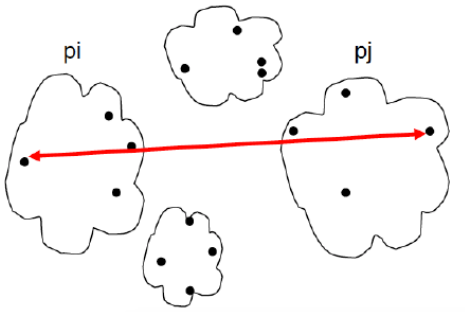


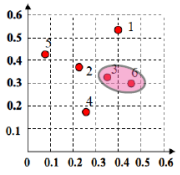


	點1	點2	點3	點4	點5	點6
點1	0.00	0.24	0.22	0.37	0.34	0.23
點2	0.24	0.00	0.15	0.20	0.14	0.25
點3	0.22	0.15	0.00	0.15	0.28	0.11
點4	0.37	0.20	0.15	0.00	0.29	0.22
點5	0.34	0.14	0.28	0.29	0.00	0.39
點6	0.23	0.25	0.11	0.22	0.39	0.00

B. 完全連結法 (complete linkage method)

是將不同群集中之兩個最遠的資料點之間的距離，當成是群集的距離值。

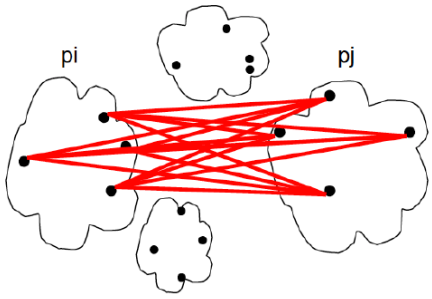




	點1	點2	點3	點4	點5	點6
點1	0.00	0.24	0.22	0.37	0.34	0.23
點2	0.24	0.00	0.15	0.20	0.14	0.25
點3	0.22	0.15	0.00	0.15	0.28	0.11
點4	0.37	0.20	0.15	0.00	0.29	0.22
點5	0.34	0.14	0.28	0.29	0.00	0.39
點6	0.23	0.25	0.11	0.22	0.39	0.00

C. 平均連結法 (average linkage method)

是將不同群集中所有成對資料點之平均成對距離，當成群集間的距離值。



D. Ward's linkage

是以兩個群集合併後所增加的 SSE，來評估兩個群集之間的鄰近性。

凡使群內距離平方和最小之兩群集即予以優先合併，愈早合併之群集表示其間的相似性愈高。目的希望合併後之群集內的組內變異量達到最小。

E. BIRCH

根據定義的半徑或直徑範圍將欲分的資料點劃分為數個相似的子群集，透過反覆合併的過程，得到分群結果。

利用群集特徵數(Cluster Feature tree, CF tree)，透過結構式分群進行層級式分群的演算法，達到較快的計算速度。

過程:

掃描資料集來建立初始的 CF-tree(可視為資料多層級壓縮)

採用其他分群技術，對 CF-tree 中的葉結點進行分群，他將移除稀疏群集(直徑變大)，或將緊密的群集組成更大的群集(直徑變小)

$$CF =< N, LS, SS >$$

N 代表該群集中有的資料點個數

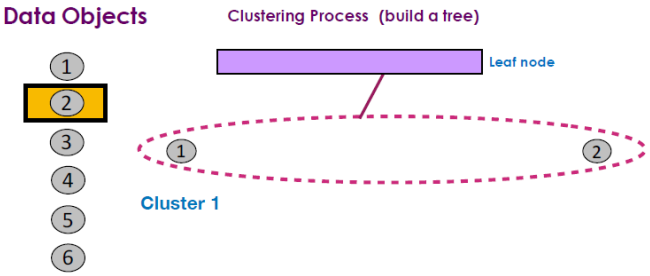
$LS = \sum_{i=1}^{N} x_i$ 為該群集中N筆資料點的線性總和

$SS = \sum_{i=1}^{N} x_i^2$ 是該群集中N筆資料點的平方加總

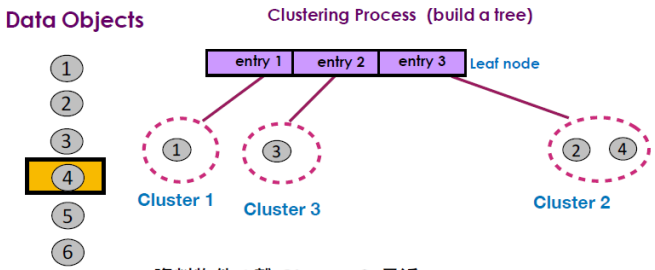
重心： $x_0 = \frac{\sum_i x_i}{n} = \frac{LS}{n}$

半徑： $R = \sqrt{\frac{\sum_i (x_i - x_0)^2}{n}} = \sqrt{\frac{nSS - 2LS^2 + nLS}{n^2}}$

直徑： $D = \sqrt{\frac{\sum_i \sum_j (x_i - x_j)^2}{n(n-1)}} = \sqrt{\frac{2nSS - 2LS^2}{n(n-1)}}$



如果加入資料物件 2 後，Cluster 1 的 直徑 > T , 則分割節點



資料物件 4 離 Cluster 2 最近

且加入資料物件 4 後，Cluster 2 的 直徑 < T，所以將資料物件 4 加入Cluster 2

- 比較
  i. 單一連結法利用區域的鄰近性來進行階層式分群，而完全連結法傾向尋找全域相似性的群集。兩者對離群值與雜訊資料皆十分敏感。
  ii. 使用平均距離、中心點距離可以克服對離群值與雜訊資料敏感的問題，其中中心點距離能進一步處理類別型資料。
  iii. 階層式分群無法良好地擴充到大型資料集上，因為每一次合併或分裂都須檢測與評估許多物件或群集。無法還原至上一階段。BIRCH 則可解決此問題。

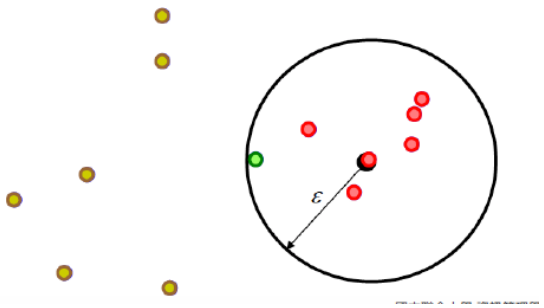
III. 密度式群集

以密度為基礎之群集法(density-based clustering) 會找出遠離低密度區域之高密度的區域

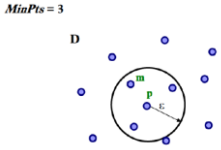
A. DBSCAN:

可以用密度的概念分群外，也可以剔除不屬於群集資料的所有雜訊點。

- Eps: 以某資料點為圓心所設的半徑長度。
- MinPts: 通常做為密度門檻值之用。以某資料點為圓心、eps 為半徑所構成之區域內所包含的資料點最小個數。若滿足此一門檻值，表示區域內的資料點密度達到最小要求。
- Core points(核心點): 以某一點為圓心，eps 為半徑所圍繞出來的範圍能包含超過 minpts 指定的資料點數目(包含該點)，則此一圓心點即為核心點。
- Border Points(邊緣點): 若有一點被某個核心點包含，但若以它為圓心卻沒辦法包含超過 minpts 指定的資料點數目，則該點即為邊緣點。
- Noise Points(雜訊點): 不屬於核心點，也不屬於邊界點，即為雜訊點。

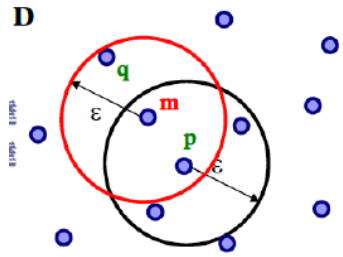


直接密度可達(Directly density reachable): 集合 D 包含一組資料點。當資料點 m 在資料點 p 的半徑 epts 內，且資料點 p 為核心點時，稱資料點 m 與資料點 p 直接密度可達。



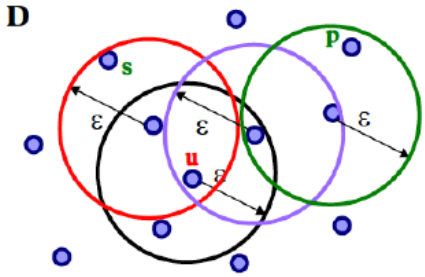
密度可達:

資料點 m 為資料點 p 的直接密度可達；資料點 q 為資料點 m 的直接密度可達；資料點 q 為資料點 p 的密度可達。

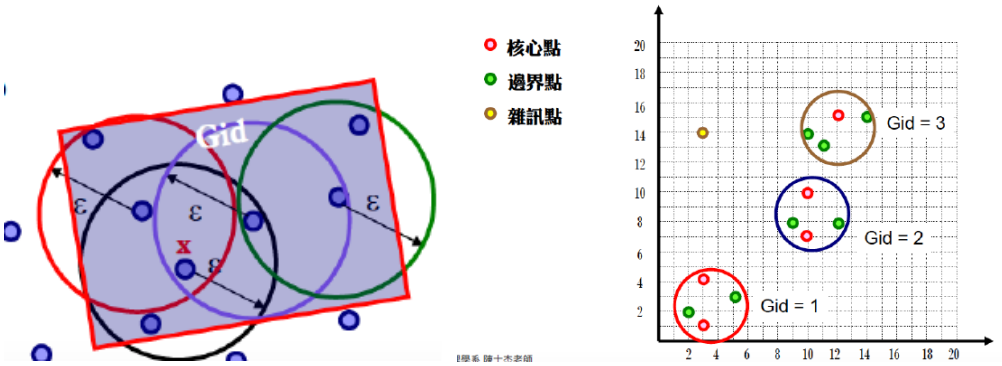


密度相連:

資料點 s 為資料點 u 的密度可達；資料點 p 為資料點 u 的密度可達；資料點 s 為資料點 p 的密度相連。



挑選一個未挑選過之核心點 x，找出核心點 x 密度可達與直接密度可達的所有其他資料點，將這些找出來的資料點與核心點 x 指派給同一個群集。

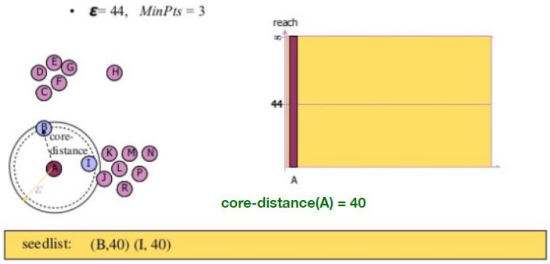


- 缺點: 對參數非常敏感。很難找到全域的密度參數。

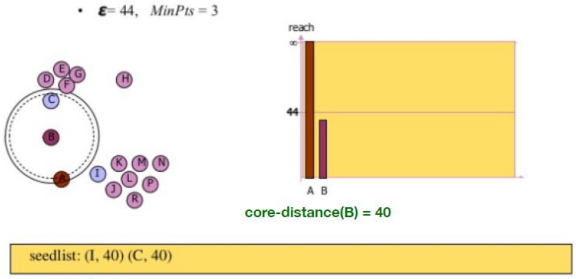
B. OPTICS

固定 Minpts，初始時隨機從資料庫中挑選一個物件作為目前物件，假設為 A，決定 A 為核心距離，並設定 A 的可達距離為未定義的。

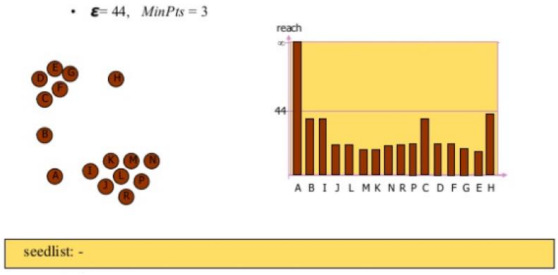
對 A 的 epts 鄰近區域中的每一個物件依據其從 A 的可達距離，插入種子順序中

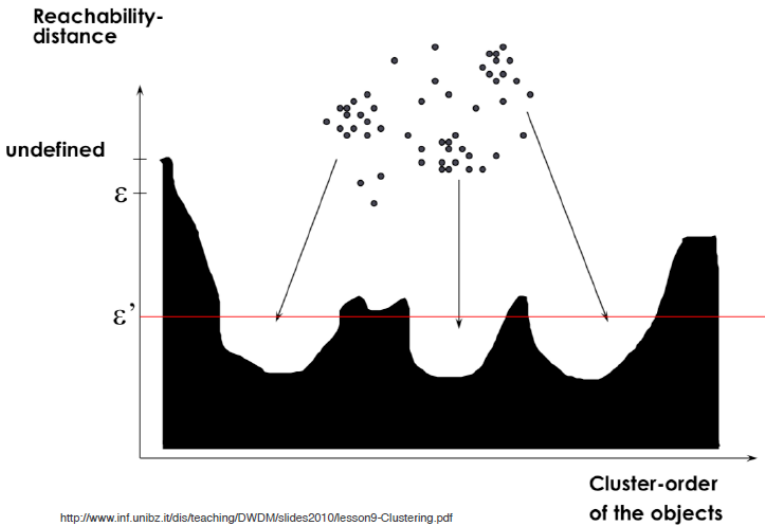


令種子順序中的下一個物件 B 作為目前物件。對 B 的 epts 鄰近區域中的每一個物件依據其從 B 的可達距離，插入種子順序中。



此迭代持續進行，直到所有資料物件都處理過，且種子順序是空的。





- 找出最佳的群集數目:

i. K = $\sqrt{\frac{n}{2}}$

ii. 評估一個群集好壞的指標: 群內誤差平方和、側影係數

iii. 使用轉折判斷法來評估最佳的 k。找出失真開始迅速增加的群集數目 k，所以曲線上的第一個轉折點就建議為正確的群集數目

- 評估一個群集好壞的指標: 側影係數

$$s(i)=\frac{b(i)-a(i)}{\max\{a(i),b(i)\}}$$

$$a(o)=\frac{\sum_{o'\in C_i,o\neq o'} dist(o,o')}{|C_i|-1};\quad b(o)=\min_{c_j:j\neq i}\left\{\frac{\sum_{o'\in C_j} dist(o,o')}{|C_j|}\right\}$$

i. ai 為樣本 i 的群內不相似度，計算樣本 i 到同群其他樣本的平均距離 ai，ai 愈小，說明樣本 i 愈應被群聚到該群

ii. bi 為樣本 i 到其他某群 Cj 的所有樣本的平均距離 bij，稱為樣本 i 與群 Cj 的不相似度。定義樣本 i 的群間不相似度。

iii. si 接近 1，則說明樣本 i 群集合理；
si 接近-1，則說明樣本 i 更應該分類到另外的群；
若 si 近似為 0，則說明樣本 i 在兩個群的邊界上。

- 群集分析的方法非常繁多，但尚未有任何方法被確定為最優異的方法，因此目前在做研究時兼採幾種不同的群集分析，再根據各種結果的意義性和可解釋性，從中挑選一個。

參考資料

1. Class Handout, Lee, Chia Jung professor, MDM64001, School of Big Data Management, Soochow University