第十一章
分群與應用

內容

- 11.1 前言
- 11.2 K-means分群法
- 11.3 植基於K-D樹的分群法
- 11.4 植基於對稱假設的分群法
- 11.5 變異數控制式的分群法
- 11.6 模糊分群法及其加速
- 11.7 結論

11.1 前言

分群 (Clustering) 是將一組資料依據某種距離的量度將其分割成若干群。

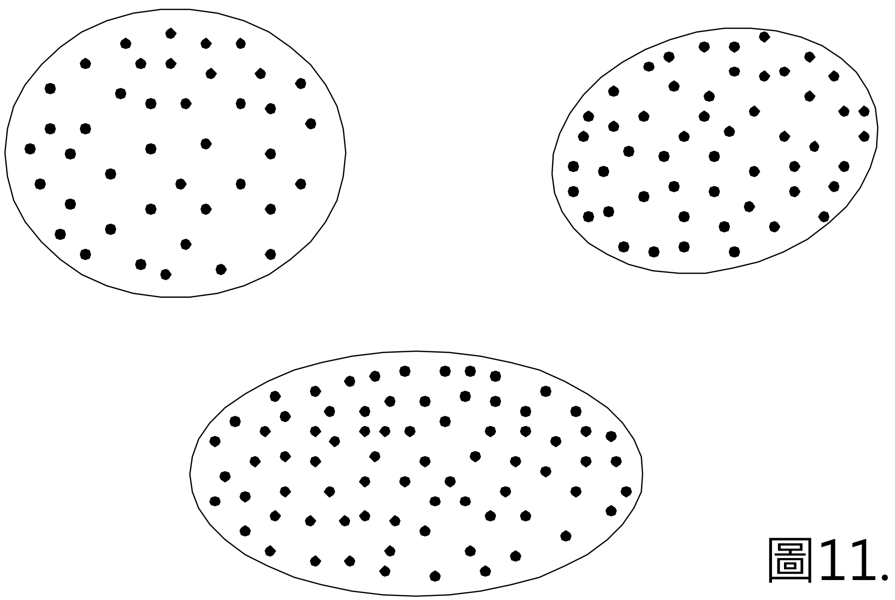


圖11.1.1 分群示意圖

11.2 K-means 分群法

K：分群數；means：分群質心。

範例1：給 n 筆資料，利用 K-means 分群法來進行分群的工作，如何決定起始的 K 個分群質心？

解答：隨機挑選 K 個資料當作起始分群質心。例如點 v_1 和點 v_8。

解答完畢

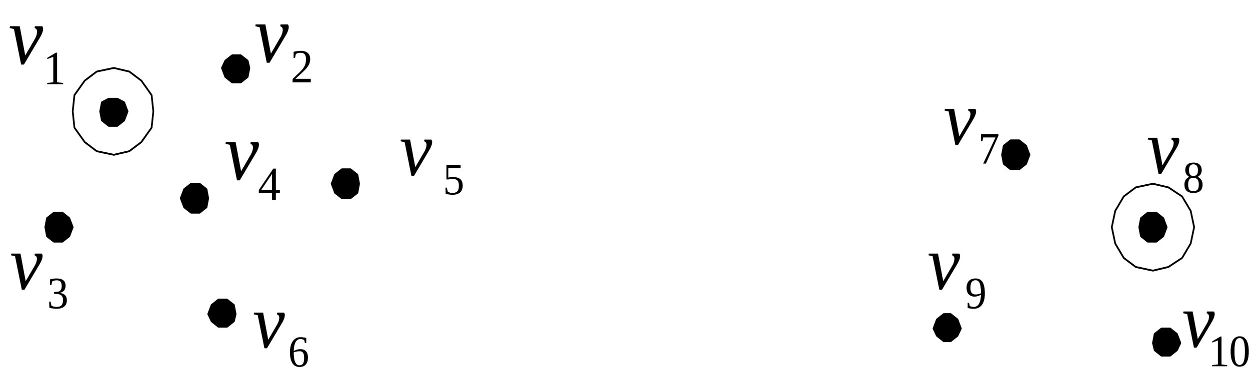


圖11.2.1 起始的兩個群心選定

範例2：K=2的情況下，如何以疊代(Iterative)的方式繼續修正兩個群心以達到最後穩態為止？

解答：

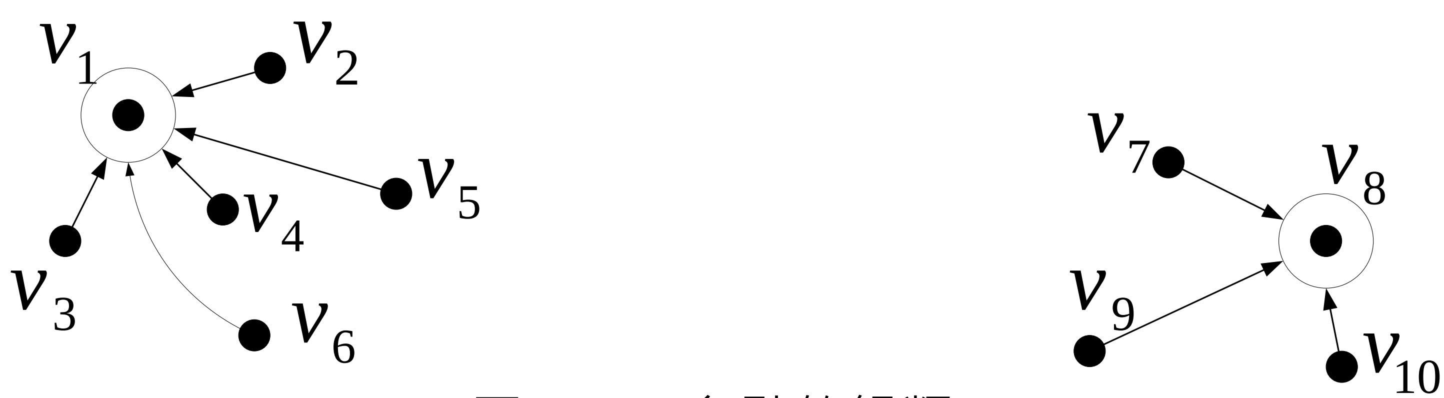


圖11.2.2 各點的歸類

計算各資料點分別與兩個群心的距離，將資料點歸類到距離最短的群心那一類。之後，再以歸類好的資料點計算出新的質心位置 $\overline{v_1}$ 和 $\overline{v_2}$。反覆為之，直到群心不改變。

解答完畢

範例3：在前面介紹的K-means分群法中，碰到較極端的例子，例如：Outlier 的例子，是否會產生不理想的分群結果？

解答： [9]

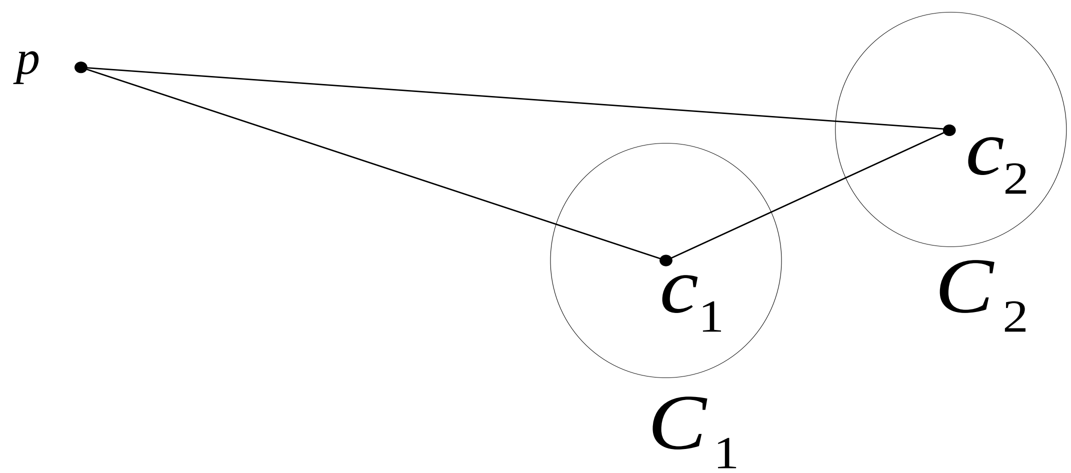


圖11.2.3 一個Outlier例子

利用$\min(d(p,c_1),d(p,c_2)) > d(c_1,c_2)$的條件，將點 p 這個Outlier另外歸為一類；否則就遵循K-means分群法。

解答完畢

11.3 植基於 K-D 樹的分群法

範例1：何謂 K-D 樹?

解答：

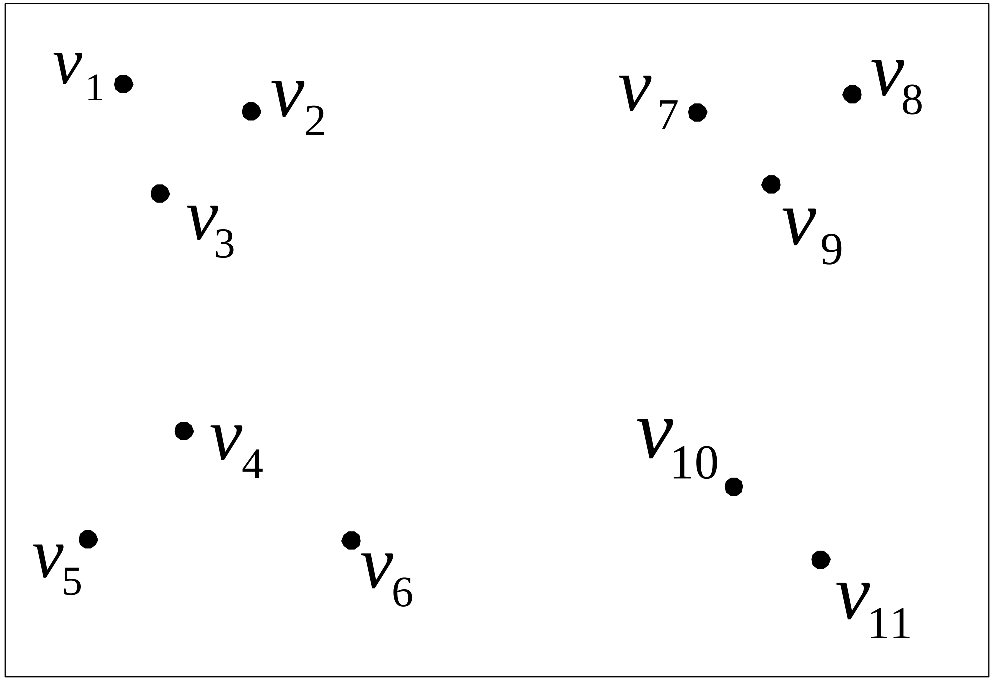


圖11.3.1 十一筆資料的分佈圖

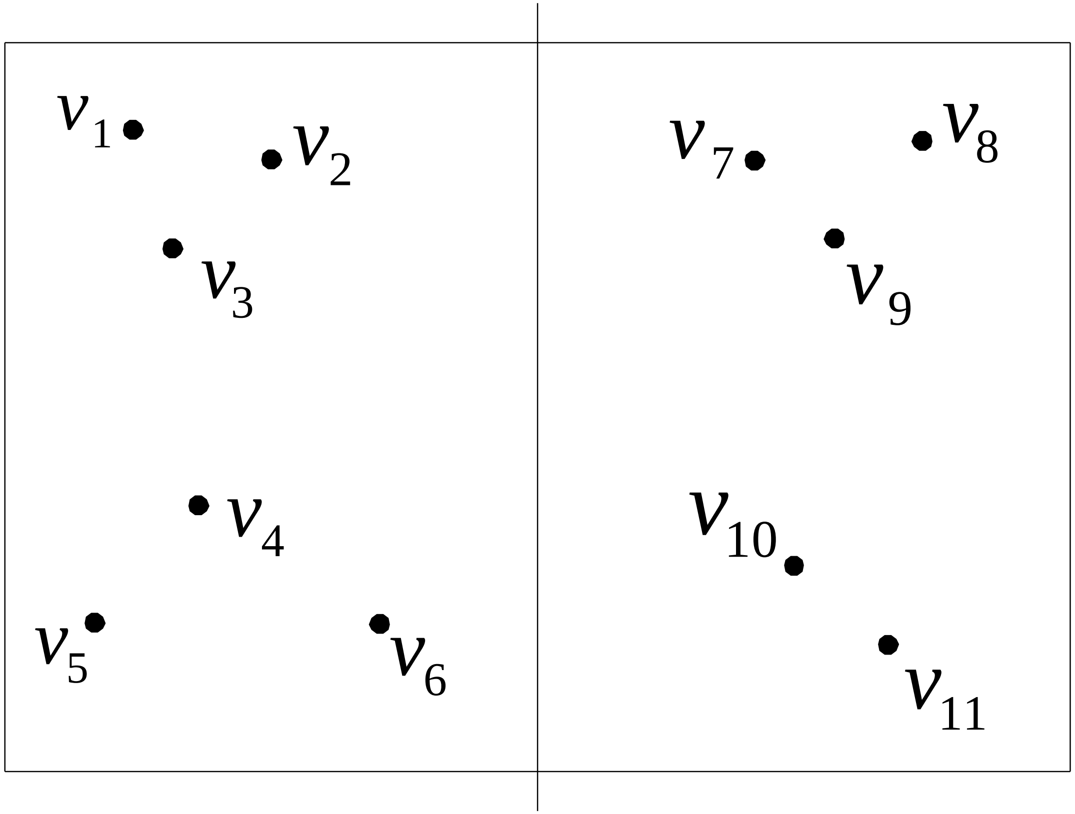


圖11.3.2　第一次分割

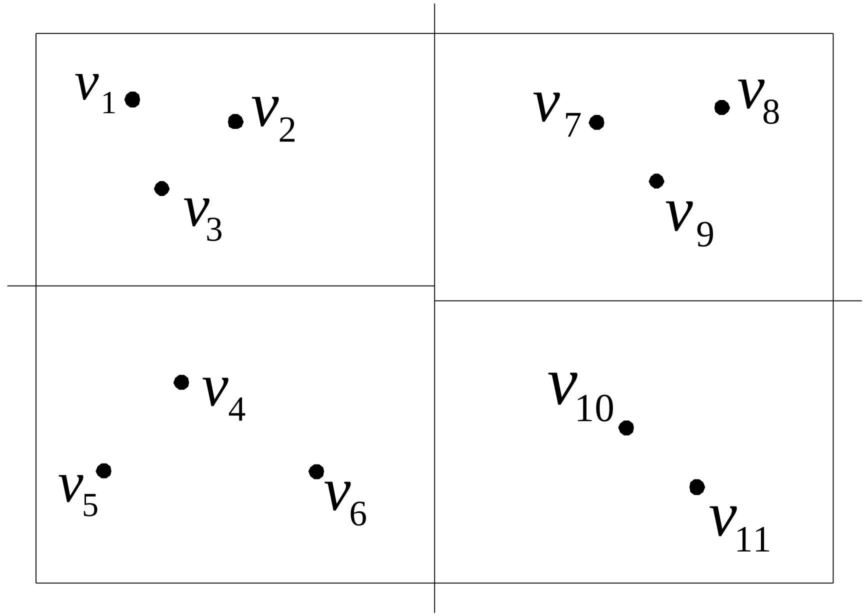
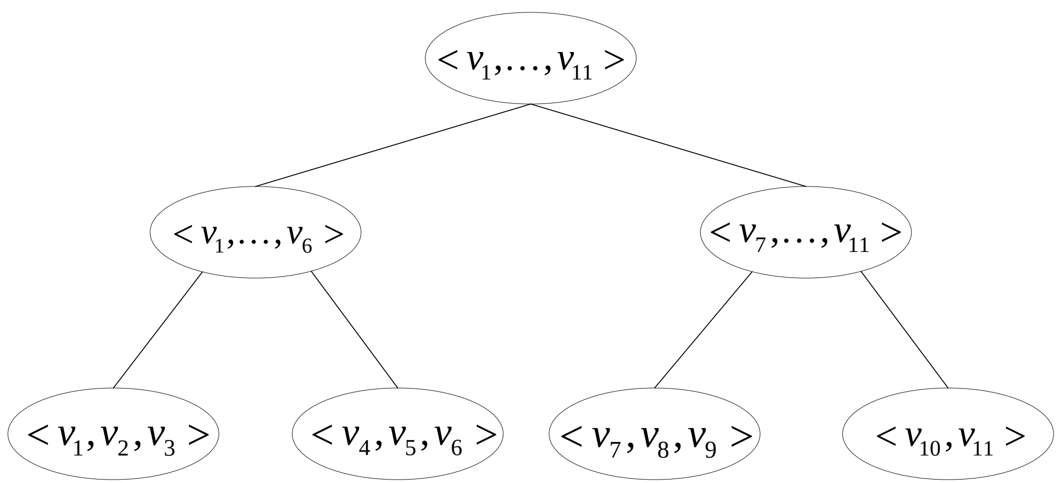


圖11.3.3　最後分割的結果



解答完畢

圖11.3.4　K-D樹

11.4 植基於對稱假設的分群法

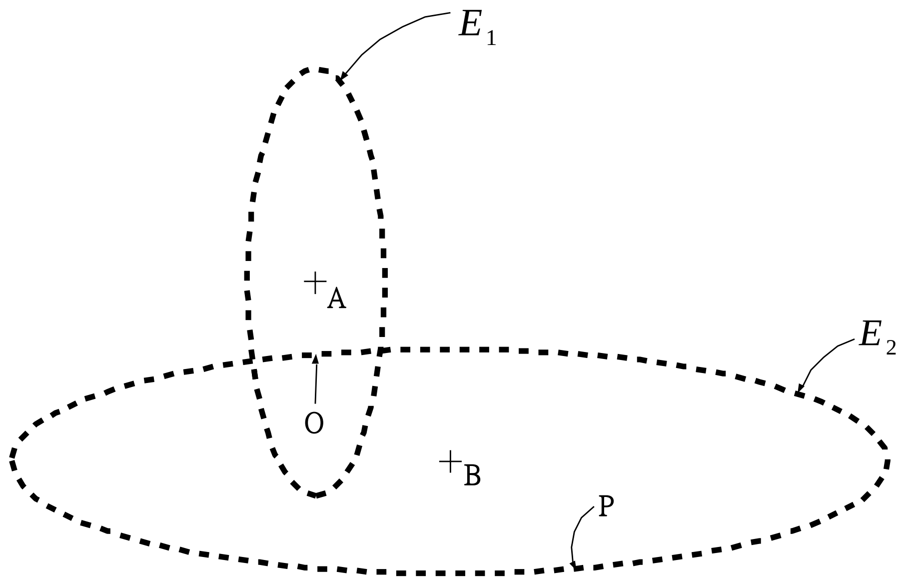


圖11.4.1 一個對稱圖的例子

根據K-means的作法，資料點O因為距離群心A較近，它會被歸屬於群心A。事實上，從資料集 E_2 的分佈來看，資料點O應該被歸屬於群心B的。造成了誤判的情形。

範例2：如何修改K-means分群方法中的距離公式避免上述誤判情形？

解答：

$$d(X_j, C) = \min_{\substack{i=1,\ldots,N \\ i \neq j}} \frac{\left\| (X_j - C) + (X_i - C) \right\|}{\left(\left\| X_j - C \right\| + \left\| X_i - C \right\| \right)} \qquad (11.4.1)$$

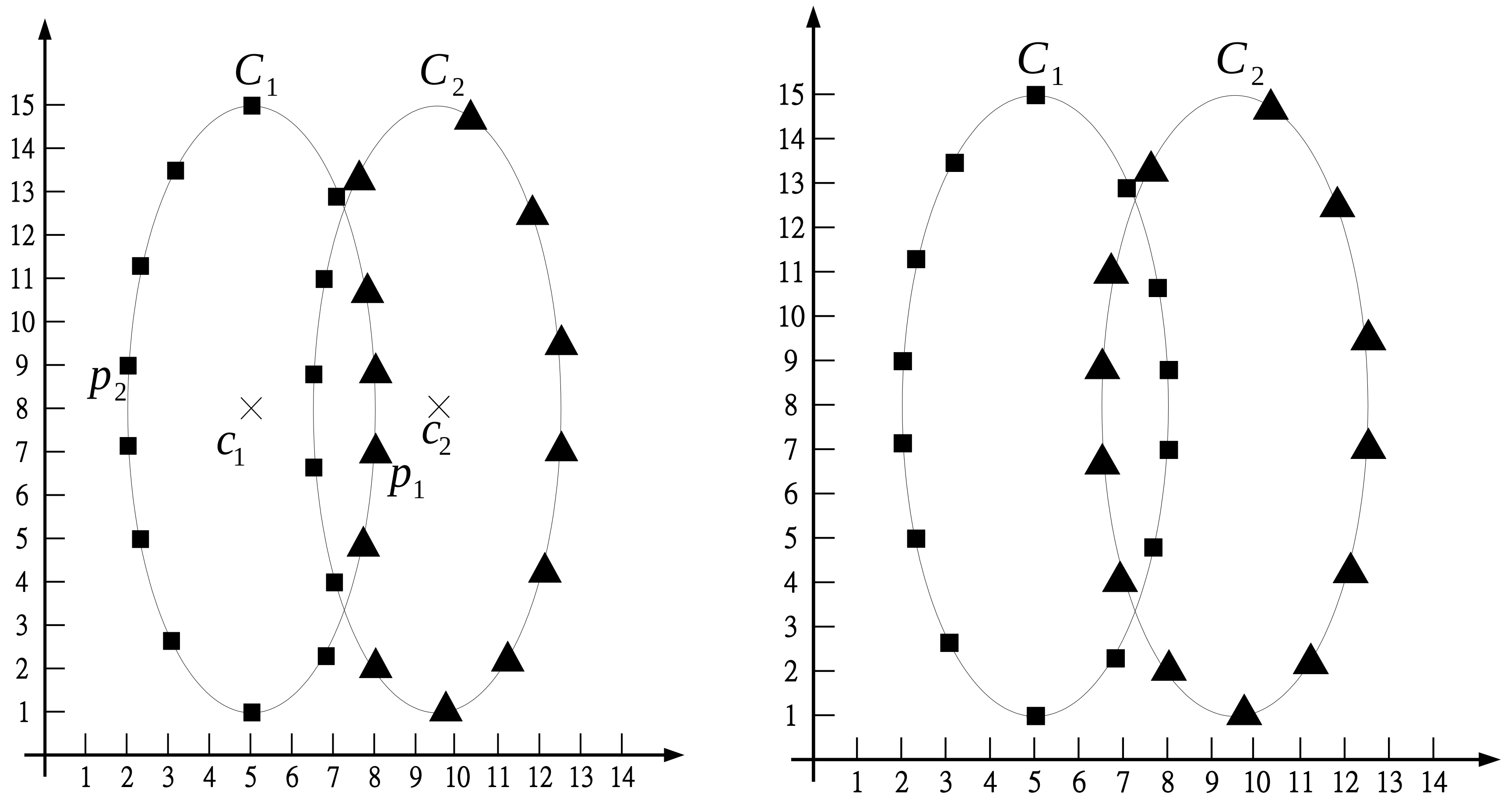
利用式(11.4.1)，圖11.4.1中的點O之對稱點為點 *P*，如此一來，點O就不會被歸屬為群心A了。

解答完畢

利用K-means演算法找到了K個群心 $\{c_k \mid 1 \le k \le K\}$。
點對稱距離量度可表示為

$$d_s(p_j, c_k) = \min \frac{\left\| (p_j - c_k) + (p_i - c_k) \right\|}{\left\| p_j - c_k \right\| + \left\| p_i - c_k \right\|}$$

圖11.4.3 (a)為一K-means方法所得的分群結果，而圖11.4.3 (b)為SC方法所得的分群結果。SC方法由於反應了對稱的考量，故得到較佳的分群結果。



(a) K-means所得的分群結果

(b) SC所得的分群結果

圖11.4.3 分群結果

範例4：SC方法有哪些可能的小弱點？

解答：SC方法的第一個小弱點為缺乏對稱的強健性。給一圖如圖11.4.4所示：

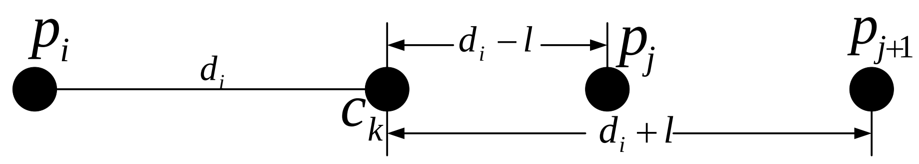


圖11.4.4 SC第一個小弱點的例子

依式(11.4.2)可推得，對資料點 p_i 而言，相對於群心 c_k ，最對稱的點為 p_{j+1}，這說明了SC方法會較偏愛較遠的點。這也多少減低了SC方法在對稱上的強健性。

SC方法的第二個小弱點為碰到資料集為SIIC(Symmetrical Intra/Inter Clusters)時，分群的效果不是很理想。如圖11.4.5中 $p_2 = \arg d_s(p_1, c_1) = p_4$ 但p_4屬於c_3：

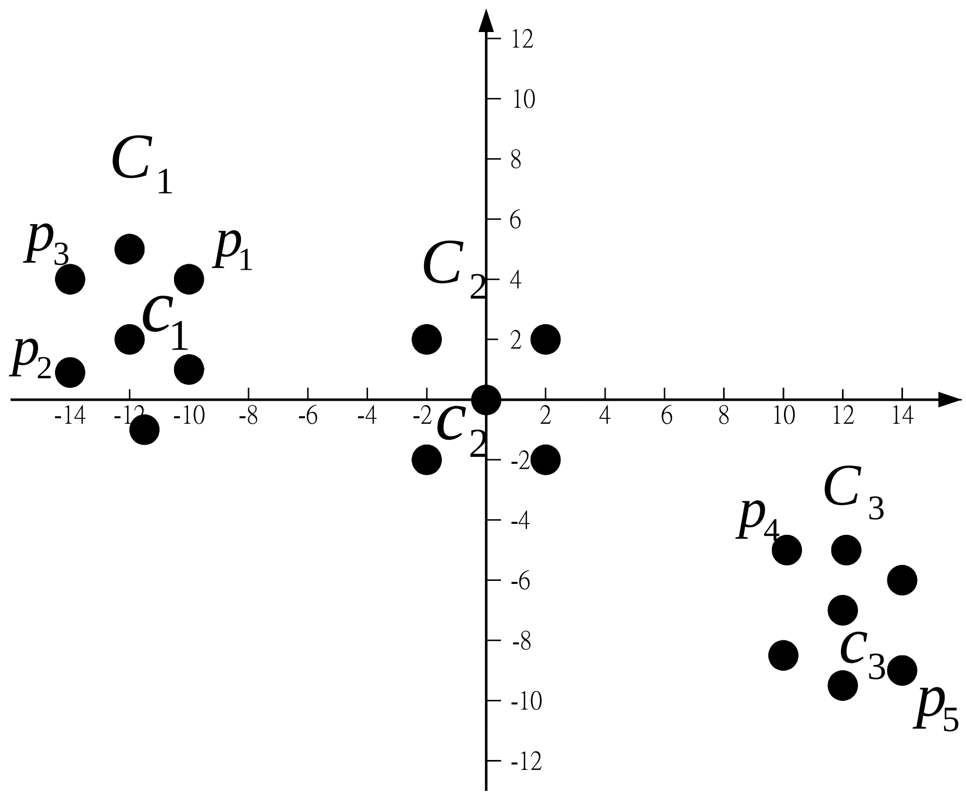


圖11.4.5 SIIC的一個例子

這造成了p_1和 p_4 分別屬於不同群，破壞了封閉性(Closure Property)。

解答完畢

為了克服SC方法中的缺乏對稱強健性，我們提出一種稱為DSL(Distance Similarity Level)的算子。為了能納入方向近似程度，我們定義一種稱為OSL(Orientation Similarity Level)的算子。

DSL算子

$$DSL(p_i, c_k, p_j) = \begin{cases} 1 - \dfrac{|d_i - d_j|}{n \times d_i}, & \text{若 } 0 \leq \dfrac{d_j}{d_i} \leq n+1 \\ 0, & \text{其他} \end{cases}$$

OSL算子

$$OSL(p_i, c_k, p_j) = \frac{v_i \cdot v_j}{2|v_i||v_j|} + 0.5$$

將這兩個算子整合成SSL(Symmetry Similarity Level)

$$SSL'(p_i, c_k, p_j) = \sqrt{\frac{DSL^2(p_i, c_k, p_j) + OSL^2(p_i, c_k, p_j)}{2}}$$

為了保有封閉性，上式改寫為

$$SSL(p_i, c_k, p_j) = \max_{p_j \in c_k} \sqrt{\frac{DSL^2(p_i, c_k, p_j) + OSL^2(p_i, c_k, p_j)}{2}}$$

SSL算子可說是對SIIC資料集分群的核心算子。

- 給定一資料集，如圖11.4.6所示。在圖 11.4.7 中分別顯示K-means方法、SC方法以及SSL方法所得分群結果與分群效果評比。

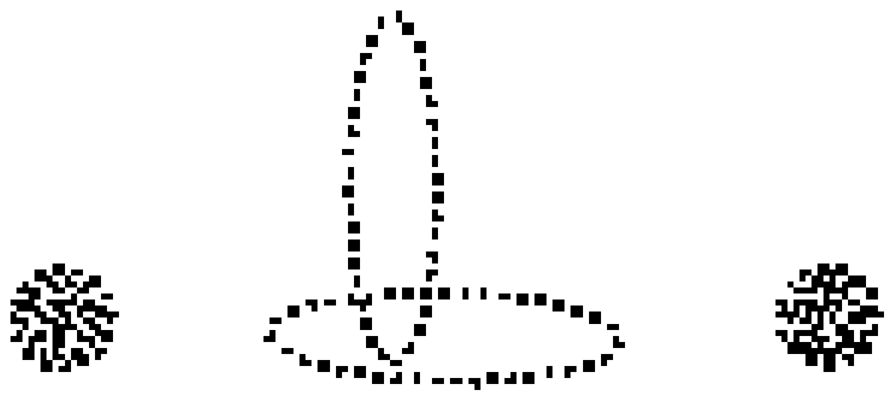
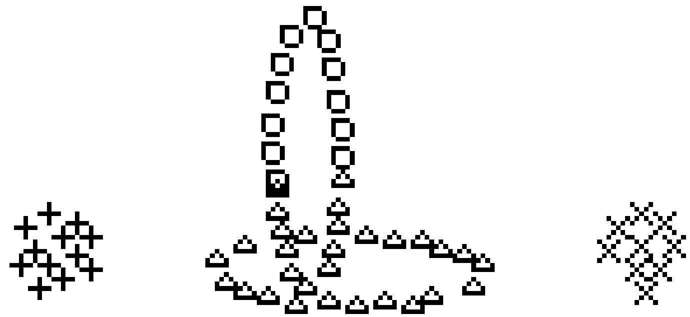
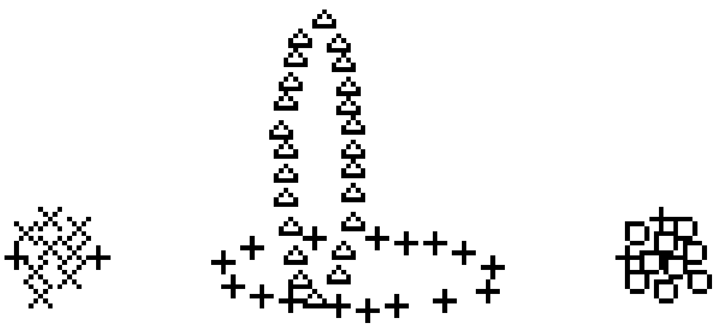


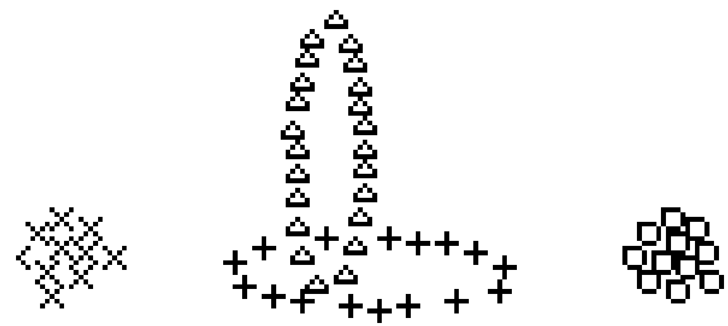
圖 11.4.6 SIIC 資料集



(a) K-means方法所得分群結果



(b) SC方法所得分群結果



(c) SSL方法所得分群結果

圖 11.4.7 三種方法的分群效果評比

11.5 變異數控制式的分群法

資料集 $X=\{x_1,x_2,\ldots,x_n\}$，將集合 X 分割成 $C_1,C_2,\ldots,C_M$ ，

$$\mathrm{Var}(C_i)\le\sigma_{\max}^2$$

這裡

$$\mathrm{Var}(C_i)=\frac{\sum_{x\in C_i}|x-\mu(C_i)|^2}{|C_i|}$$

一開始，我們任意將 X 分割成成 $C_1, C_2, \ldots, C_I$。接下來，我們計算出它們的變異數，$\mathrm{Var}(C_i)$，$1 \le i \le I$。如果 $Var(C_i) > \sigma_{\max}^2$，則 C_i 進行隔離(Isolation)的動作。

$$\mathrm{IB}(C_i) = \bigcup_{x \in C_i} \{ y \mid FD(x, y), y \in C_i \}$$

這裡 $FD(x, y)$ 表示 y 和 x 有最遠距離。接下來，我們將 $\sqrt{|IB(C_i)|}$ 個內部邊點予以分離出去。
如果 $\mathrm{Var}(C_i) < \sigma_{\max}^2$，則對 $x \in C_i$ 算出

$$\left\{ y \mid \min_{y \in X - C_i} \| y - x \|^2 \right\}$$

y：視為點 x 的外部邊緣點。

我們針對外部邊緣集

$$OB(C_i) = \bigcup_{x \in C_i} \{y \mid ND(x, y), y \in X - C_i\}$$

這裡 ND(x, y) 表示 y 和 x 有最近距離。隨機選取 $\sqrt{|OB(C_i)|}$ 個外邊緣點並將它們和 Ci 合併(Union)起來。

在[1]中，學者們提出利用干擾法(Perturbation)來改善最後的群效果。對群 Ci 來說，我們在 OB(Ci) 中挑一點 $x \in C_j$，如果下式成立，則將 x 從 Cj 中移除，而將 x 納入 Ci 中

$$G_{ab} = S(C_i) - S(C_i \cup \{x\}) + S(C_j) - S(C_j - \{x\}) > 0$$

$$S(C_i) = \sum_{x \in C_i} |x - \mu(C_i)|^2$$

11.6 模糊分群法及其加速

給資料點集 $X=\{x_1,x_2,...,x_n\}$，FCM分群法將資料點集 X 分成C群。令這C群的群心集為 $V=\{v_1,v_2,...,v_c\}$。令資料點 x_j 對群心 v_i 的隸屬函數值為 u_{ij}，隸屬矩陣 U 表示為

$$U=\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ u_{21} & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ u_{c1} & u_{c2} & \cdots & u_{cn} \end{bmatrix}$$

群心集 V和資料點集 X的誤差為：

$$E(U,V:X)=\sum_{i=1}^{c}\sum_{j=1}^{n}(u_{ij})^m\left\|x_j-v_i\right\|^2 \qquad (11.6.1)$$

$$\sum_{i=1}^{c}u_{ij}=1$$

依據Lagrange Multiplier方法，可得：

$$L(U,\lambda)=\sum_{j=1}^{n}\sum_{i=1}^{c}(u_{ij})^m\left\|x_j-v_i\right\|^2-\sum_{j=1}^{n}\lambda_j\left(\sum_{i=1}^{c}u_{ij}-1\right)$$

$L(U,\lambda)$ 對 λ_j 微分後令為零，可得到

$$\frac{\partial L(U,\lambda)}{\partial \lambda_j} = 0 \Leftrightarrow \sum_{i=1}^{c} u_{ij} - 1 = 0 \qquad (11.6.2)$$

$L(U,\lambda)$ 對 u_{ij} 微分後令為零，可得到

$$\frac{\partial L(U,\lambda)}{\partial u_{ij}} = 0 \Leftrightarrow \left[m(u_{ij})^{m-1} \left\| x_j - v_i \right\|^2 - \lambda_j \right] = 0 \qquad (11.6.3)$$

由式(11.6.3)可解得

$$u_{ij} = \left(\frac{\lambda_j}{m \left\| x_j - v_i \right\|^2} \right)^{\frac{1}{m-1}} \qquad (11.6.4)$$

由式(11.6.2)和式(11.6.4)可得到

$$\sum_{i=1}^{c} u_{ij} = \sum_{i=1}^{c} \left(\frac{\lambda_j}{m \left\| x_j - v_i \right\|^2} \right)^{\frac{1}{m-1}} \tag{11.6.5}$$

$$= 1$$

從式(11.6.5)可得

$$\left(\frac{\lambda_j}{m} \right)^{\frac{1}{m-1}} = 1 \Bigg/ \sum_{i=1}^{c} \left(\frac{1}{m \left\| x_j - v_i \right\|^2} \right)^{\frac{1}{m-1}} \tag{11.6.6}$$

將式(11.6.6)代入式(11.6.4)，得

$$u_{ij} = 1 \Big/ \sum_{k=1}^{c} \left(\frac{\left\| x_j - v_i \right\|}{\left\| x_j - v_k \right\|} \right)^{\frac{2}{m-1}} \tag{11.6.7}$$

群心 v_i 可調整為

$$v_i = \frac{\sum_{j=1}^{n} (u_{ij})^m x_j}{\sum_{j=1}^{n} (u_{ij})^m}, 1 \le i \le c \tag{11.6.8}$$

模糊 C-means 共分下列五個步驟：

步驟一：選定群數C、次方m、誤差容忍度ε 和起始隸屬矩陣 U_0 。

步驟二：根據資料點集和 U_0 算出起始的群心集。

步驟三：重新計算U_{ij} ，$1 \le i \le c$ 和 $1 \le j \le n$ 。修正各個群心值。

步驟四：計算出誤差 $E = \sum_{i=1}^{c} \left\| v_i^{前} - v_i^{後} \right\|$ ，這裏 $v_i^{前}$ 和 $v_i^{後}$ 代表群心 v_i 連續兩個疊代回合的值。

步驟五：若 $E \le \varepsilon$ 則停止；否則回到步驟三。