# 簡介

生物資訊是最近非常熱門的一個研究方向。所謂的生物資訊，是一個結合生物學、計算機科學及資訊科技所形成的新研究領域，最終的目標是發現新的生物認知，進而建立生物系統的大概念，以辨識生物學的各項準則

在DNA生物資訊之中，微陣列 (microarray) 是用來儲存基因表現的重要工具(如圖1所示)。從微陣列中找出相似群組可以幫助醫學界更準確地判斷哪些基因有可能會導致相同的疾病。本系統目的是設計一個使用者介面，可讓生物學家從微陣列進行分群 (clustering) 中尋找出的相似表現型態的基因群集來獲得需要的知識。

而本系統中，我們主要採用由Haixun Wang, Wei Wang, Jiong Yang, Philip S. Yu所提出的「Clustering by Pattern Similarity in Large Data Sets」，以及
李建億，黃乙展，吳崢榕，「新的pCluster方法：pCluster+ 與 incremental pCluster」此兩篇所提出的演算法。

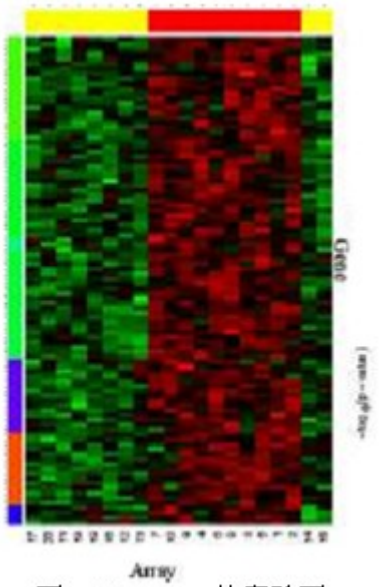


圖1 Microarray的實驗圖

# 分群技術

## 1.1 分群定義

分群是一種將相似的物件或是屬性分類到同一群的過程，分群的定義主要有以下兩點：

1. Set of like elements. Elements from different clusters are not alike.

2. The distance between points in a cluster is less than the distance between a point in the cluster and any point outside it.。

## 1.2 傳統分群

現今存在許多不同定義的分群方法，它們多數的概念都是基於物件與物件之間的距離差異，例如常用的歐幾里德距離、餘弦距離等，也就是在被歸為同一群組的資料中，它們必須有接近的數值。

如圖2所示，透過傳統的分群技術，我們會將學生1、學生2、學生3 分為一群，學生4、學生5 分為一群。

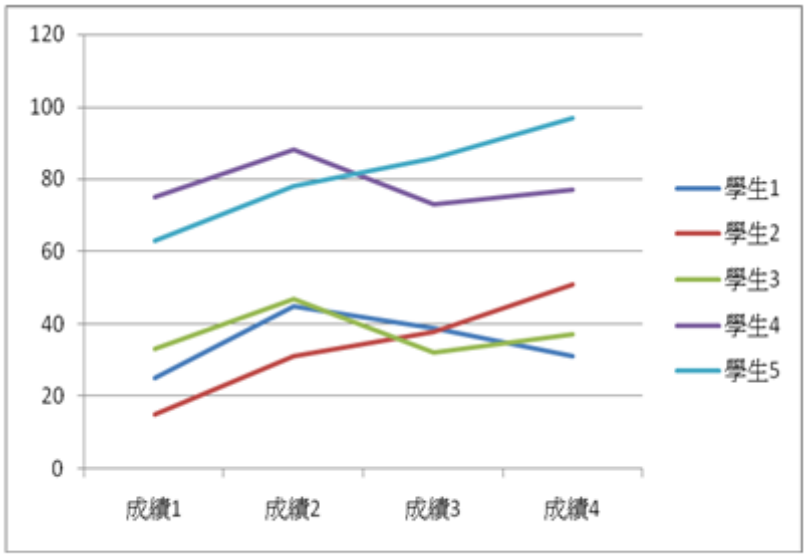



圖2 學生成績折線圖

然而如果用另一種分群方式，我們可以看見學生2跟5是慢慢進步的。因此傳

統分群方法不適用於DNA analysis，本專題所採用的分群技術是一種稱為pCluster的分群技術。它的優點是能夠找出物件在不同條件下，有一致性的變化的數據。

## 1.3 pCluster分群

在醫學上通常是將基因資料存放於Microarray中，用以表現差異性的基因，而pCluster分群技術能將Microarray中相似反應的資料分為一群，便於判斷基因中是否有相同的疾病。

### 1.3.1 Microarray

Microarray(如圖3所示)，通常在分析上會視為一個二維陣列(如圖4所示)，其中一個維度是基因，數量可達數十萬；另一個維度是某種condition，數量通常只會到達幾百。陣列中所紀錄的值，即某個基因在某個condition下，所表現的程度 (以數字來表示)。

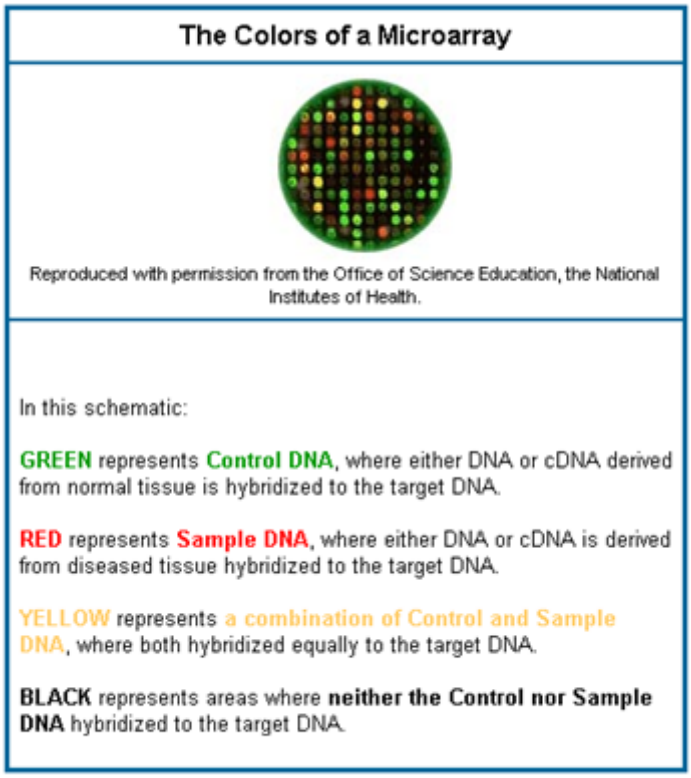



圖3 Microarray顏色示意圖

| | C1 | C2 | C3 | C4 | C5 | C6 |
|---|---|---|---|---|---|---|
| Gene 1 | 10 | 7 | 8 | 13 | 5 | 8 |
| Gene 2 | 11 | 11 | 10 | 5 | 3 | 2 |
| Gene 3 | 11 | 10 | 8 | 7 | 12 | 11 |
| Gene 4 | 5 | 6 | 2 | 4 | 7 | 2 |
| Gene 5 | 6 | 3 | 6 | 9 | 1 | 4 |
| Gene 6 | 1 | 4 | 4 | 2 | 5 | 5 |
| Gene 7 | 7 | 8 | 4 | 6 | 6 | 6 |
| Gene 8 | 9 | 6 | 5 | 12 | 4 | 7 |

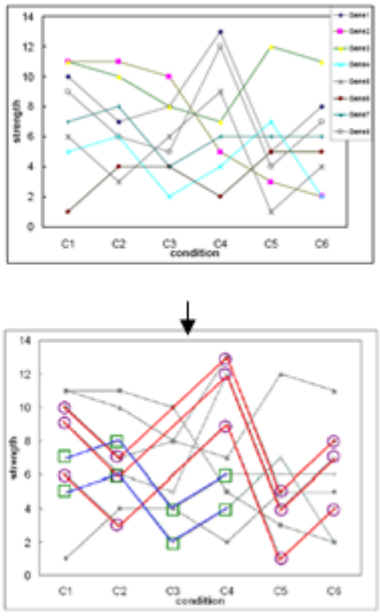



圖4 A DNA Microarray

### 1.3.2 pCluster clustering

pCluster clustering，是能從microarray這個毫亂無章的資料中，分出哪些資料是有類似的特徵或是相同的起伏區塊，這些相似的資料便能組成一組cluster，其如下圖所示，目的在於找出microarray中，哪些基因在哪些condition下，會有一致性的表現變化。從圖形上來看，這些有一致性的基因，它們所呈現出來的折線變化，會很接近是平行的狀態。

如圖3所示，一個小型的microarray，將此表格轉換成折線圖，我們很難看出其中的關連性。透過pCluster Algorithm 我們可以找出*Gene1*、*Gene5*、*Gene8*在*C1*、*C2*、*C4*、*C5*、*C6*這五個條件下表出來的特徵彼此有相似起伏的關係，則我們稱此為一組pCluster pair。

而在此圖當中可以看見有兩組不同起伏狀況的pair。

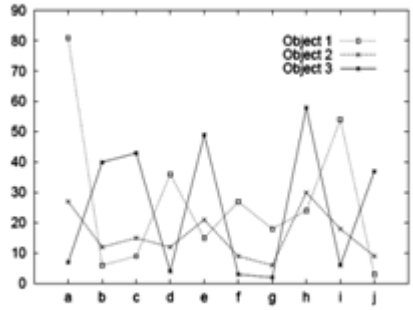



圖5 Raw data: 3 objects and 10 columns

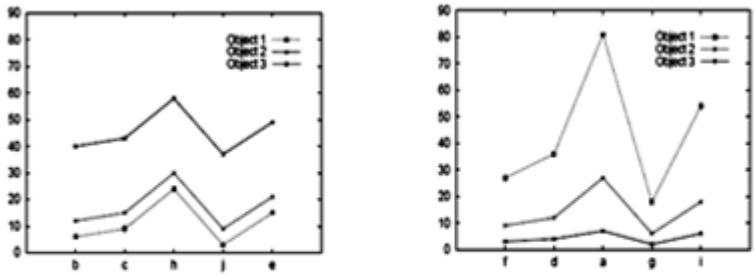


圖6 object in Figure1 form a Shifting Pattern & Scaling Pattern in subspace {b,c,h,j,e}

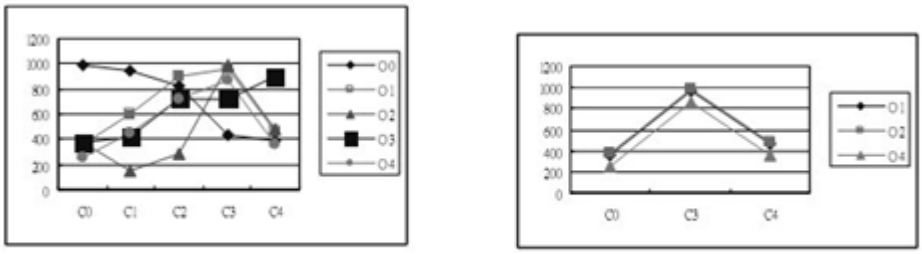


圖7 將資料集以折線圖方式呈現 並找出一組3x3 pCluster

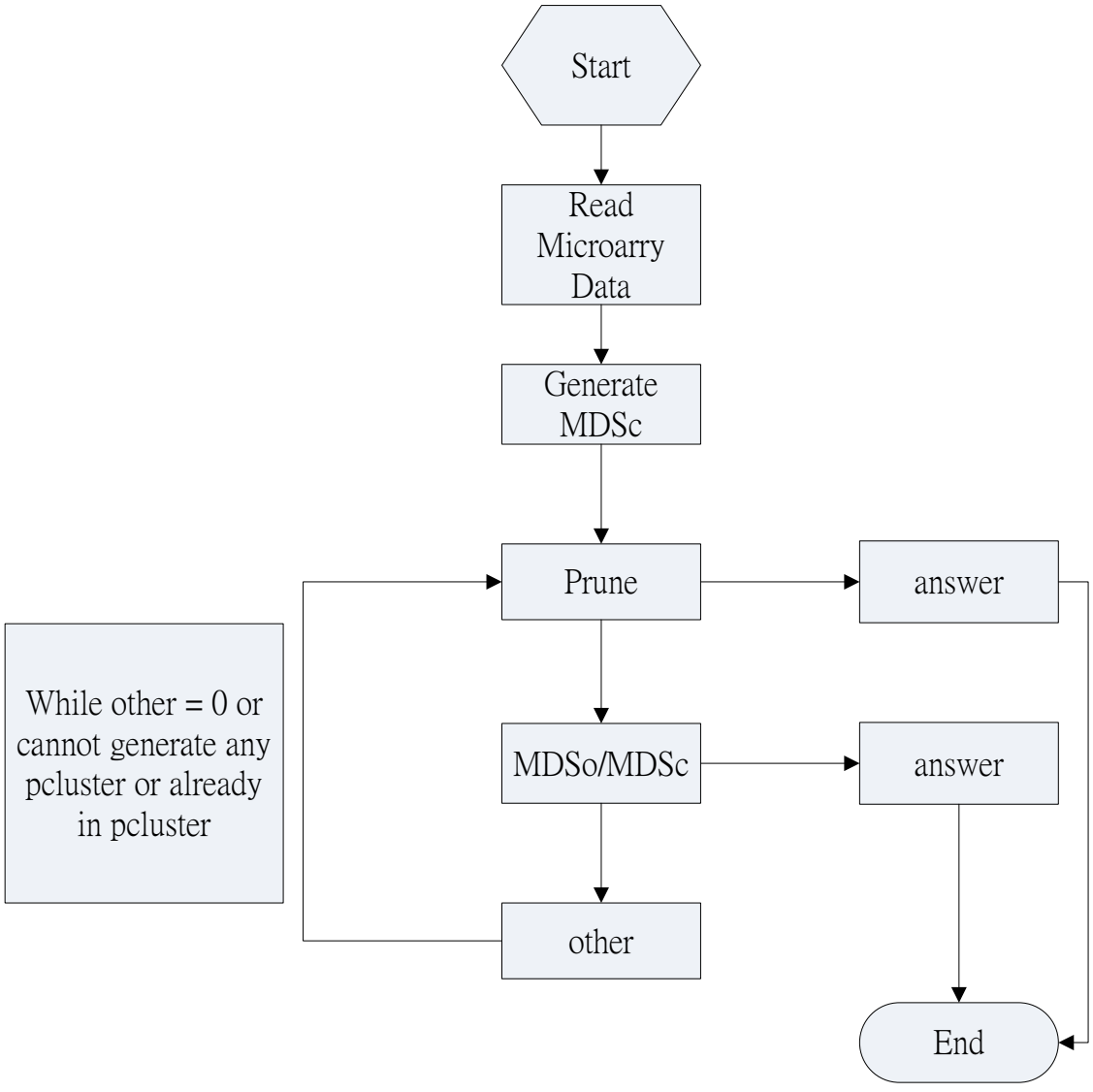
此外，pCluster Algorithm更可以找出資料集中相似的片段。

如圖5所示的資料，透過pCluster Algorithm可以找出如圖6左圖中的平行關係，甚至可以找出如如圖6右圖中的相似倍率起伏關係。

如圖7所示可以找出一組3x3的pair。

# 1. 系統架構

## 1.1 系統流程



# 1.2 系統實作

### 2.2.1 名詞說明

| | |
|---|---|
| $\delta$ | User-specified clustering threshold |
| *nr* | User-specified minimum # of rows of a pCluster |
| *nc* | User-specified minimum # of columns of a pCluster |
| *MDSc* | Maximum Dimension Sets of condition |
| *MDSo* | Maximum Dimension Sets of object |

### 2.2.2 系統演算法

在「Clustering by Pattern Similarity in Large Data Sets」此篇論文中提到對物件兩兩進行判斷並建Tree來找出所有pCluster pair。然而在一般情況下，物件的數量通常會大於屬性數量許多，因此花在對物件進行兩兩比對來產生MDSo的時間將會遠大於對屬性進行兩兩筆對產生MDSc的時間。因此我們結合「新的pCluster方法：pCluster+ 與 incremental pCluster」所提出的改良方法來實作pCluster + Algorithm。以下說明本系統詳細的演算法流程及實例介紹。

***Step1: 建立MDSc***

建立MDSc有下列規則:

1. 我們從Microarray Data中任取兩組Condition，算出兩兩的差值
2. 依屬性的差值進行sort
3. 找出所有頭尾小於$\delta$的群組($\delta$為容許的誤差值，設的越小結果則越精確)，即為一組MDSc pair
4. 重複上述步驟，直到找出所有的MDSc pairs結束第一步驟。

**Input:** $x, y$: two objects, $\mathcal{T}$: set of columns, $nc$: minimal number of columns, $\delta$: cluster threshold
**Output:** All $\delta$-pClusters with more than $nc$ columns

$s \leftarrow d_x - d_y$; /* i.e., $s_i \leftarrow d_{xi} - d_{yi}$ for each $i$ in $\mathcal{T}$ */
sort array $s$;
$start \leftarrow 0$ ; $end \leftarrow 1$ ;
$new \leftarrow$ TRUE; /* a boolean variable, if TRUE, indicates an untested column in $[start, end]$ */
**repeat**
  $v \leftarrow s_{end} - s_{start}$;
  **if** $|v| \leq \delta$ **then**
    /* expands $\delta$-pCluster to include one more columns */
    $end \leftarrow end + 1$;
    $new \leftarrow$ TRUE;
  **else**
    Return $\delta$-pCluster if $end - start \geq nc$ and $new =$ TRUE;
    $start \leftarrow start + 1$;
    $new \leftarrow$ FALSE;
**until** $end \geq |\mathcal{T}|$;
Return $\delta$-pCluster if $end - start \geq nc$ and $new =$ TRUE;

圖8 Algorithm : Find two-condition pCluster:pairCluster(x, y, T, nc)

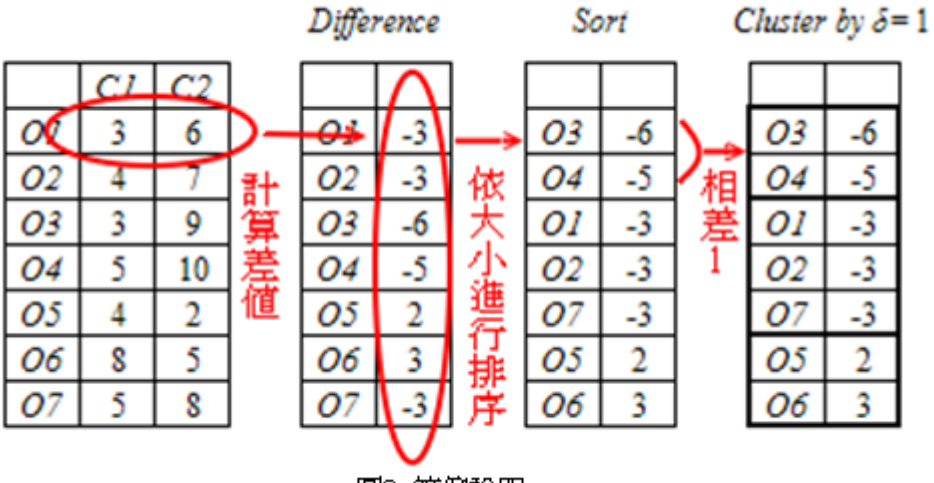



圖9 範例說明

舉例說明，上中我們有兩筆特徵數據，*Condition1*與*Condition2*，首先*Condition1-Conditon2*算出差值後，對差值進行排序。假設此刻使用者所設定的$\delta$(最小容忍誤差)為1，則我們發現*O3*到*O4*的差值為1($\leqq\delta$)，但*O3*到*O1*的差值

卻超出$\delta$，因此將*O3*, *O4*歸類為一組MDSc pair。*O1*到*O7*的差值為0($\leqq\delta$)，但*O1*到*O5*的差值卻超出$\delta$，同理。　最後我們可以得到以下三組 MDSc pair:

(*Condition 1*, *Condition 2*) ➔ (*O3*, *O4*)
(*Condition 1*, *Condition 2*) ➔ (*O1*, *O2*, *O7*)
(*Condition 1*, *Condition 2*) ➔ (*O5*, *O6*)

**實作範例 ($\delta$=1, *nr*=3, *nc*=3)**

| | $C_0$ | $C_1$ | $C_2$ | $C_3$ | $C_4$ |
|---|---|---|---|---|---|
| $O_0$ | 9 | 3 | 6 | 100 | 29 |
| $O_1$ | 10 | 5 | 8 | 10 | 56 |
| $O_2$ | 7 | 1 | 5 | 6 | 53 |
| $O_3$ | 200 | 19 | 22 | 24 | 70 |
| $O_4$ | 372 | -8 | -4 | -3 | 43 |
| $O_5$ | 11 | 5 | 9 | 11 | 57 |

6x5資料集

($C_0$ $C_1$) -> ($O_0$ $O_1$ $O_2$ $O_5$)
($C_0$ $C_2$) -> ($O_0$ $O_1$ $O_2$ $O_5$)
($C_0$ $C_3$) -> ($O_1$ $O_2$ $O_5$)
($C_0$ $C_4$) -> ($O_1$ $O_2$ $O_5$)
($C_1$ $C_2$) -> ($O_0$ $O_1$ $O_2$ $O_3$ $O_4$ $O_5$)
($C_1$ $C_3$) -> ($O_1$ $O_2$ $O_3$ $O_4$ $O_5$)
($C_1$ $C_4$) -> ($O_1$ $O_2$ $O_3$ $O_4$ $O_5$)
($C_2$ $C_3$) -> ($O_1$ $O_2$ $O_3$ $O_4$ $O_5$)
($C_2$ $C_4$) -> ($O_1$ $O_2$ $O_3$ $O_4$ $O_5$)
($C_3$ $C_4$) -> ($O_1$ $O_2$ $O_3$ $O_4$ $O_5$)

產生出來的所有MDSc

***Step2: 刪去不符合條件的MDS　(MDS Pruning)***

因為產生出MDSc之後也會有雜質存在，因此我們必須將他依條件去掉。

1. MDSc中的前兩個屬性欄位分別為*firstValue*及*secondValue* (如圖10所示)，後面欄位是代表物件的值。而每一個物件欄位都給一個*object' count*=1。另一個變數*MDS_num*也是=1。

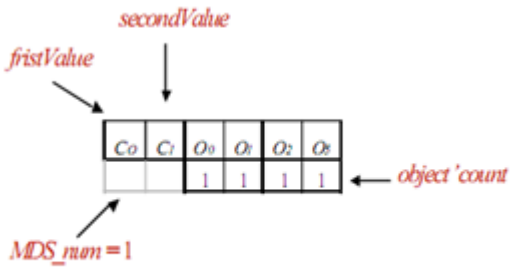



圖10 MDSc的呈現結構

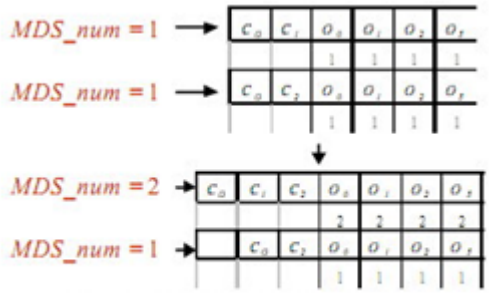



圖11 對MDSc進行分析比對

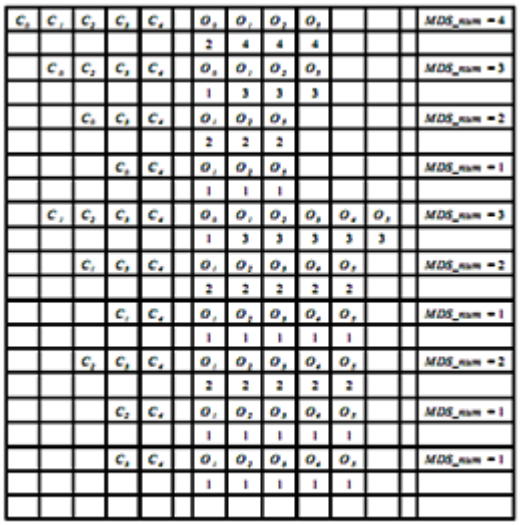


圖12 分析後的結果

2. 接下來所有MDSc進行比對，比對原則如下(如圖11所示)

2.1. 如果它們的*firstValue*相等，且*secondValue*不相等，則比對MDSc的物件欄位。

2.2. 如果比對後發現，它們的物件欄位相同的部份大於等於 *nr*，則將相同部份的*object' count*加1，同時合併它們屬性欄位，並將*MDS_num*加1。

經由上述比對原則，我們可以得到如圖12所示

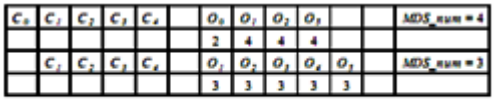



圖13 pCluster候選者

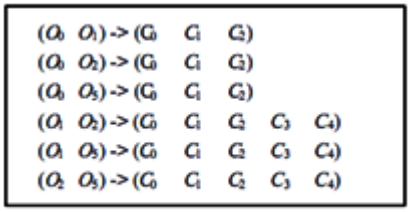



圖14 合併相同欄位之MDS

3. 接下來對於每個MDSc中的物件欄位進行分析，以決定是否要砍除。

   3.1. 如果其*MDS_num*小於*nc*-1則將其砍除

   3.2. 如果其*object'count*小於*nc*-1則將其砍除

   3.3. 如果砍除物件此動作造成物件欄位數目小於*nr*則將此MDSc砍除

重複此步驟，直到所有pairs都檢驗過符合條件後，再將所有存在的子集合併，依序產生出pClusrer的候選人(如圖13所示)。

***Step3: 判斷是否繼續產生MDSo***

1. 如果每個候選者的*object'count*都等於屬性欄位的數目-1，則可以直接輸出為一個pCluster，如圖13所示的第二組；不等於的話就必須對候選者繼續進行產生MDSo的動作(如圖14所示)。然後再將屬性欄位相同的MDSo合併，輸出成為pCluster，也就是輸出答案。
2. 重複步驟二的砍除動作，先產生候選者然後再比對產生出新的pCluster直到沒有辦法新增任何的pCluster才做結束。

**實作範例最後分群結果 － 3組pCluster pair**

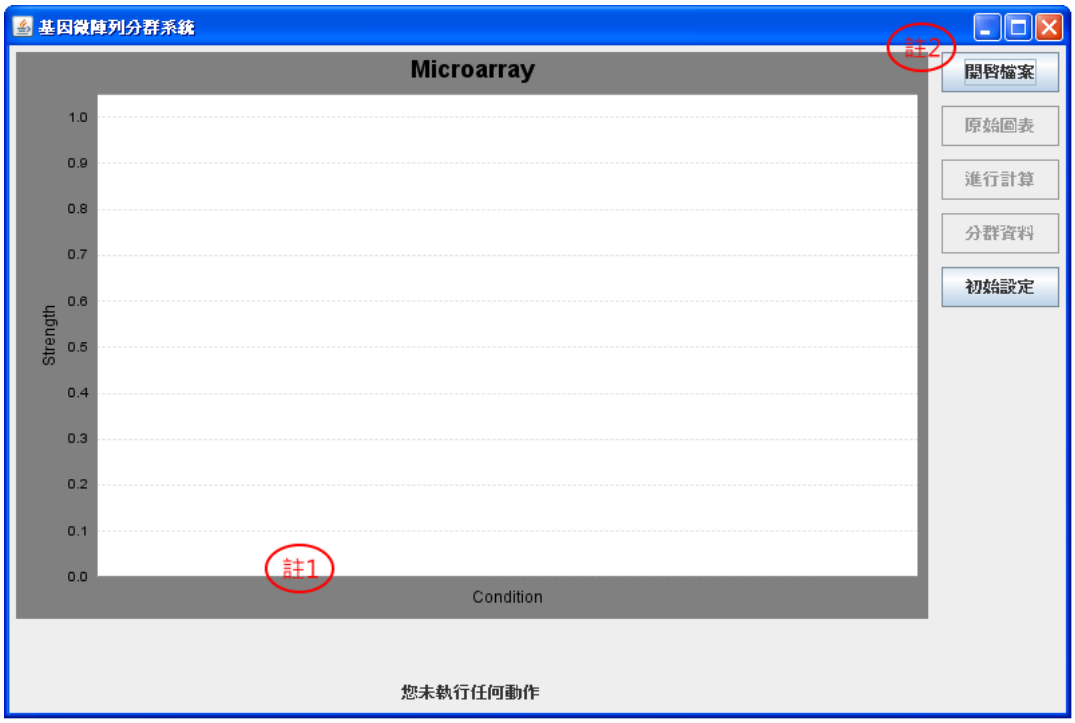
$(O_1,O_2,O_3,O_4,O_5) \longrightarrow (C_1,C_2,C_3,C_4)$

$(O_0,O_1,O_2,O_5) \longrightarrow (C_0,C_1,C_2)$

$(O_1,O_2,O_5) \longrightarrow (C_0,C_1,C_2,C_3,C_4)$

# 3. 成果展示

## 操作介面




圖註 1：顯示原始資料之折線圖處

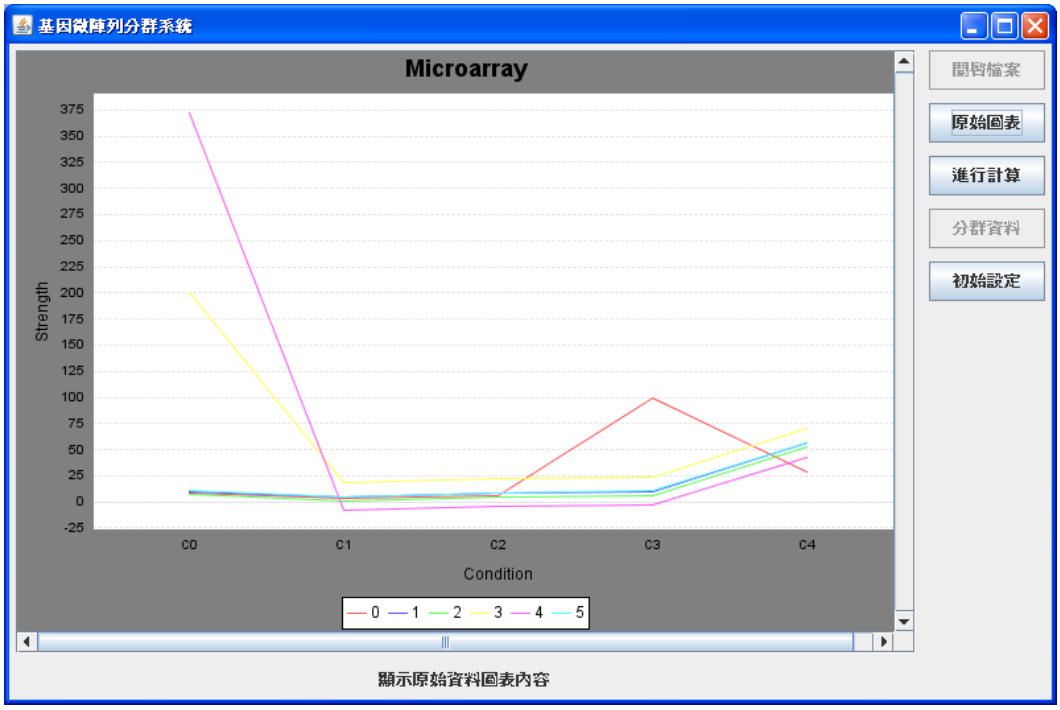
圖註 2：功能按鍵，包括

1. 開啟檔案：開啟 Microarray 檔案
2. 原始資料：將 Microarray 顯示於註 1 處
3. 進行計算：實作 pCluster Algorithm，尋找出所有 pair
4. 分群資料：用折線圖顯示所有 pair
5. 初始設定：使用者參數設定( $\delta, nc, nr$ )

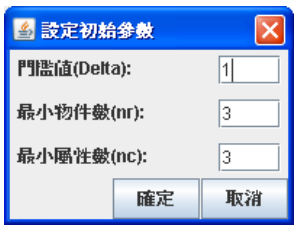
以下將會實作系統演算法中所提到的例子，產生 3 組 pair

## 實作系統演算法中之 6x5 數據

1. 顯示出原始資料之折線圖

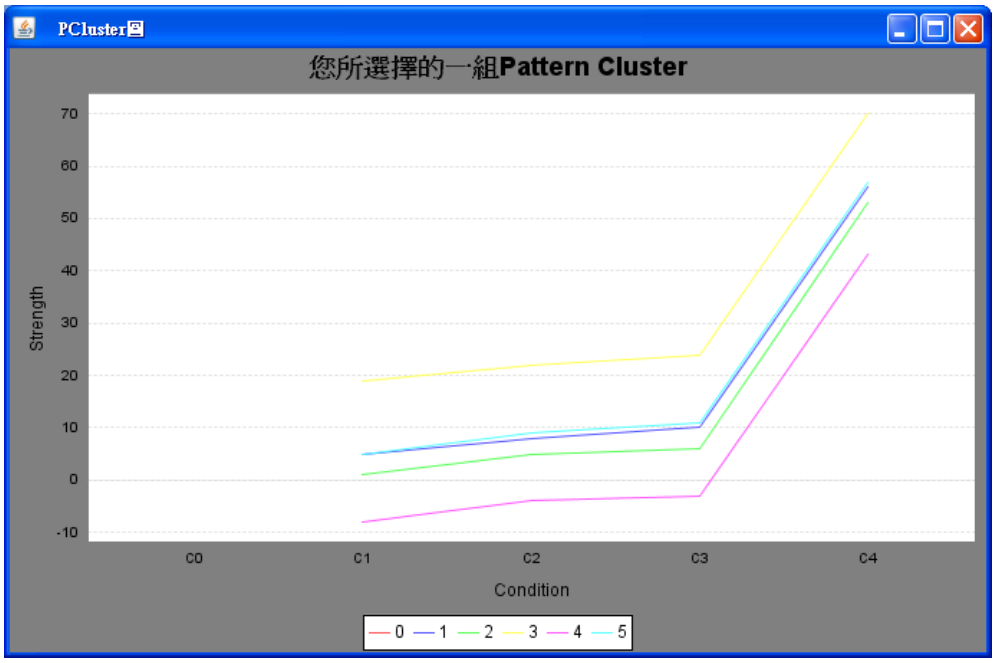



2. 初始參數設定 $\delta$ $nr$ $nc$



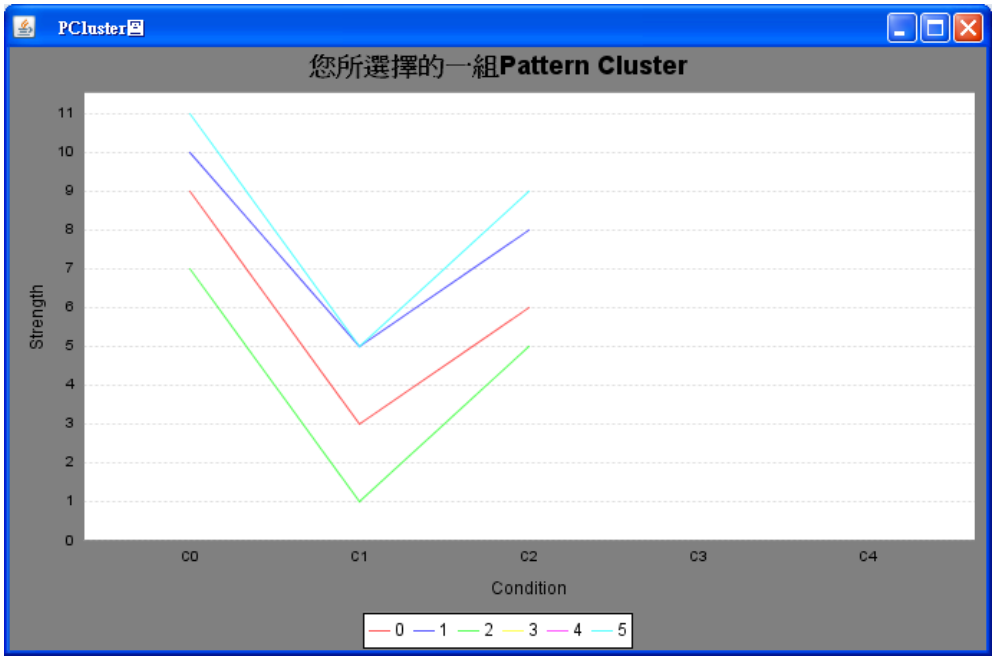

3. 經過運算後－找出下列三組 pCluster pair

4. 當選擇第一組 pair 時，產生出以下結果，我們可以看見這 5 組基因在 $C_1$ 到 $C_4$ 有相似的起伏情況



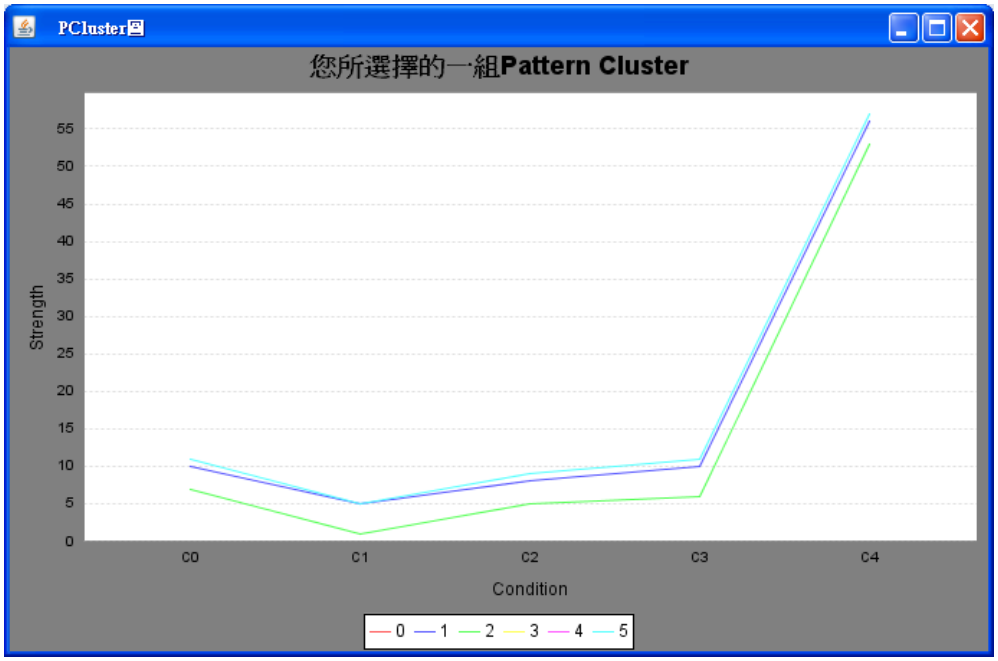

第一組 pair

$(O_1, O_2, O_3, O_4, O_5) \rightarrow (C_1, C_2, C_3, C_4)$




第二組 pair

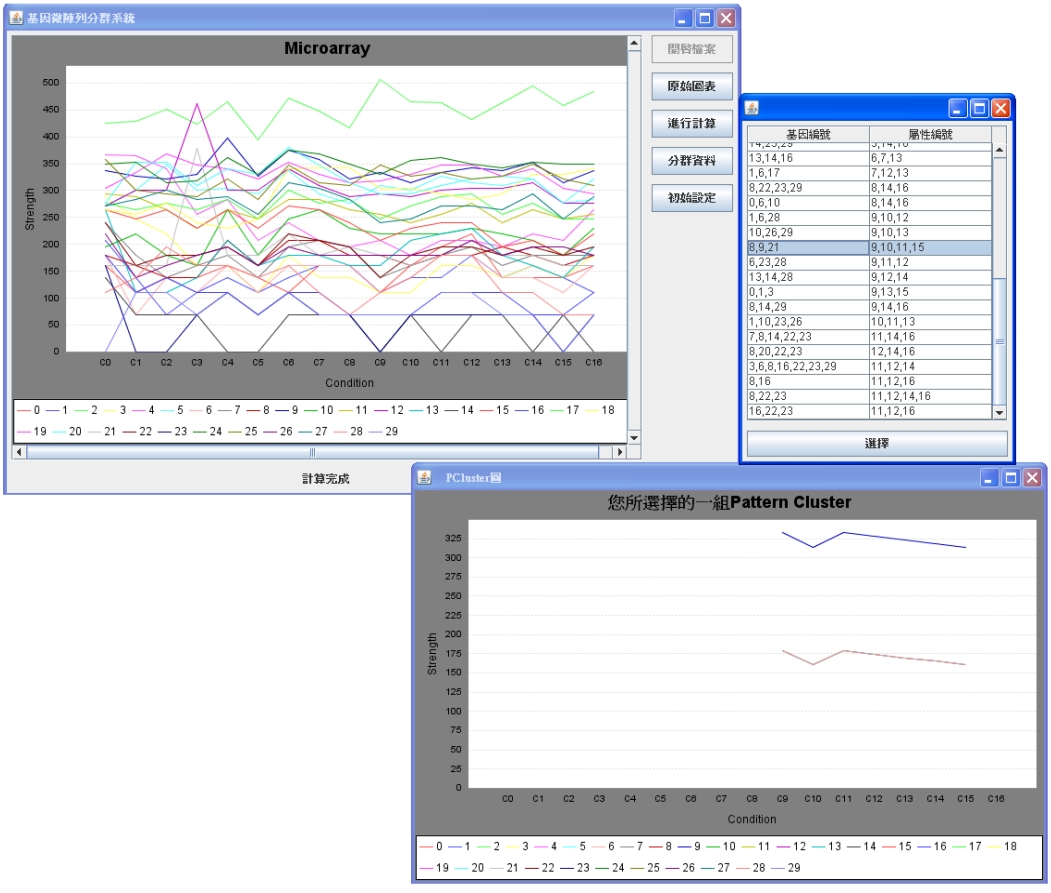
$(O_0, O_1, O_2, O_5) \rightarrow (C_0, C_1, C_2)$




第三組 pair

$$(O_1, O_2, O_5) \longrightarrow (C_0, C_1, C_2, C_3, C_4)$$

## 其他實驗數據



# 4. 未來展望

在 pCluster 中還存有許多議題值得研究，本系統所使用到的演算法，時間複雜度為 $n^2$，一旦第一階段產生的 MDSc 數量太多，仍舊需要許多時間進行計算，也佔用極大的記憶體量。

是否能尋找出更佳的演算法來計算龐大的資料量呢？以及 pCluster 是否能有其它的擴展方式，像是應用在非數值上？　這些問題都值得我們一一探討，未來的工作期許能對這些議題做進一步研究。

# 5. 結論

在整個系統當中我們使用 java 來撰寫程式，因 JAVA 具有 portability 且有許多 library 可供使用，在視窗化的部份 JAVA 也提供 AWT 與 Swing 套件，包括使用某公司所推出的 jfreechart 套件來呈現我們的圖表。系統實作成折線圖表也提供了生物學家可以更迅速從現有資料中獲得需要的知識。

在測試過程中也發現了一些問題，如輸入的基因量過大時，會產生記憶體不足的現象等等。這部分我們與主程式互相調整，以期能夠解決遇到的問題。

現在已可以成功利用 pCluster+的演算法成功實作出幫助生物學家對 microarray 進行分群比對的介面。

因系統的開發環境是使用 Java 攥寫，因此電腦內仍須安裝 JVM 才能執行本程式，未來希望能讓使用者透過網頁上執行，以其更能達到便利的特性。

# 參考資料


- Haixun Wang, Wei Wang, Jiong Yang, Philip S. Yu, 「Clustering by Pattern Similarity in Large Data Sets」Proc. of the 2002 ACM SIGMOD International Conf. on Management of Data, pp. 394-405, 2002.
- 李建億，黃乙展，吳崢榕，「新的pCluster方法：pCluster+ 與 incremental pCluster」