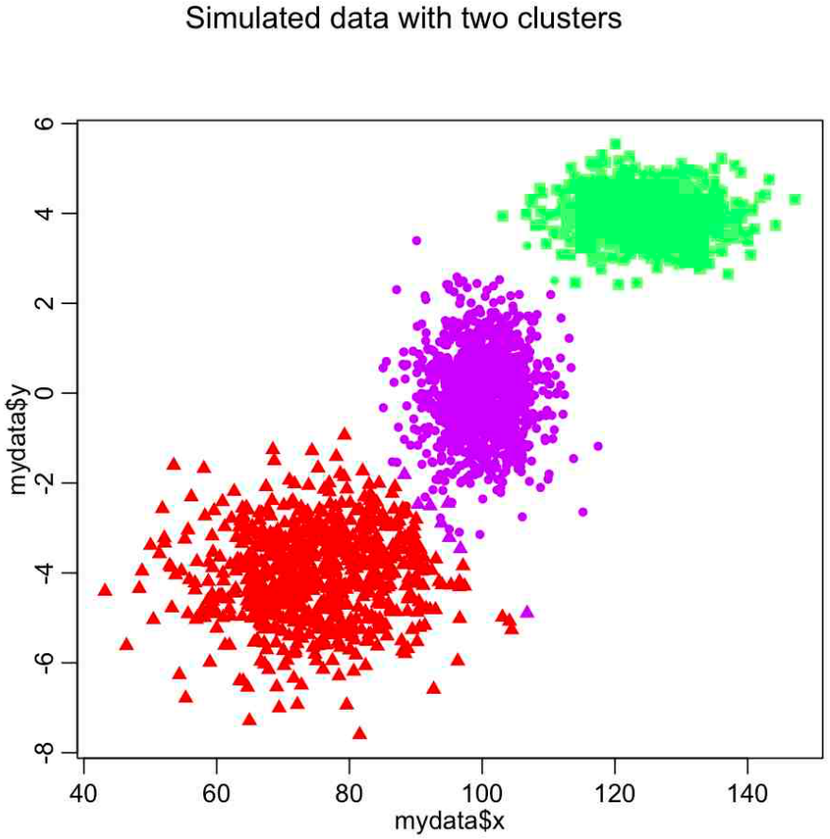
平行程式設計 Final Project:

DBSCAN

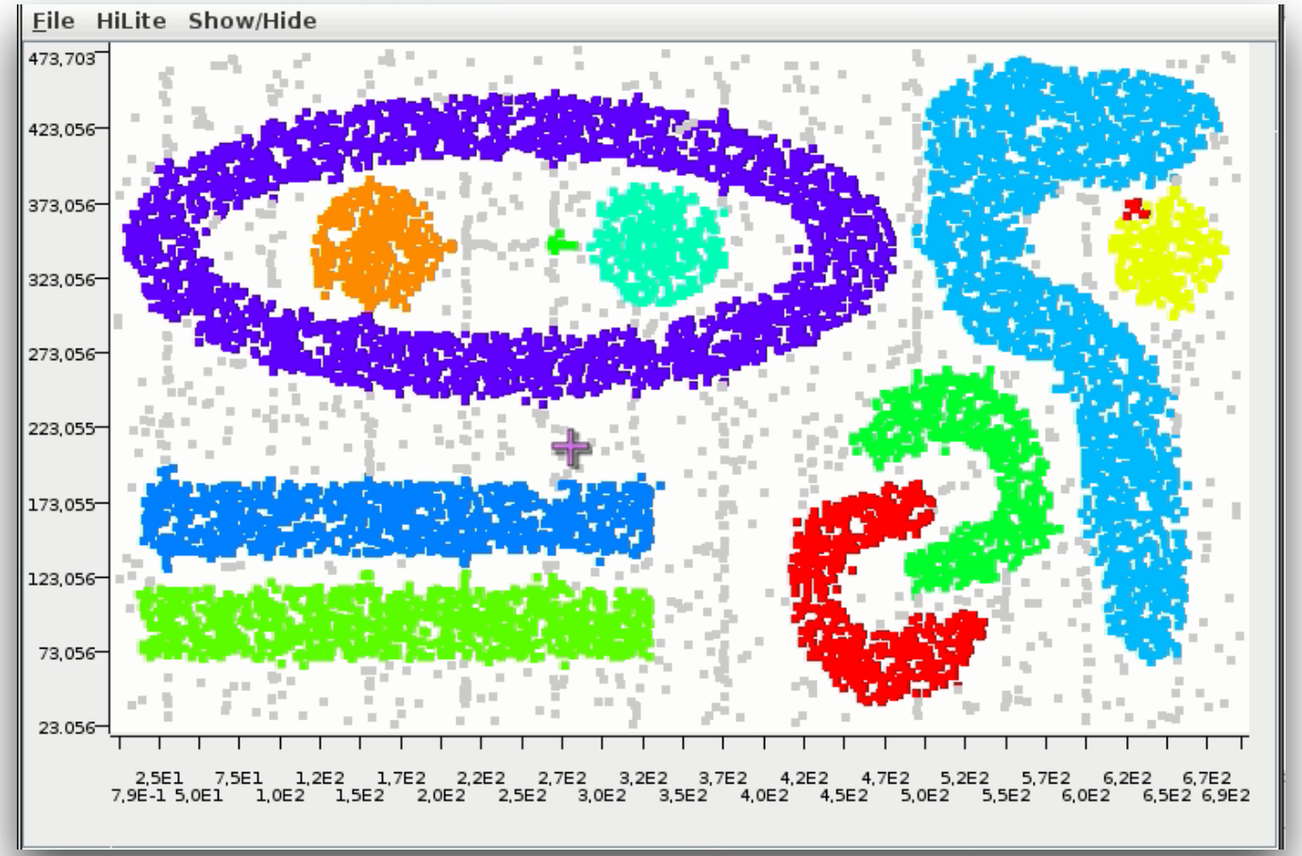
鄭博元

DBSCAN

- 一種Clustering的方式。
- Density Based的Clustering方式。
- Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd (Vol. 96, No. 34, pp. 226-231).

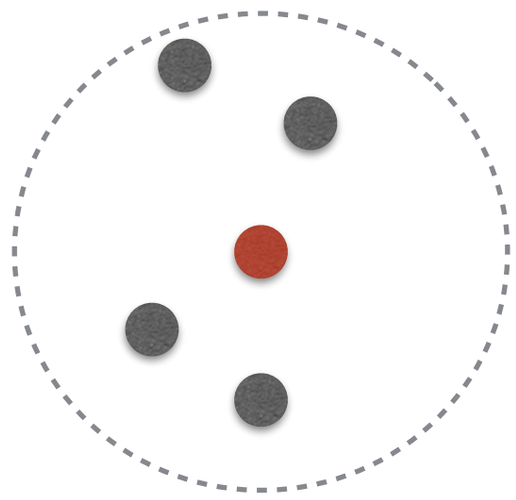


K-means

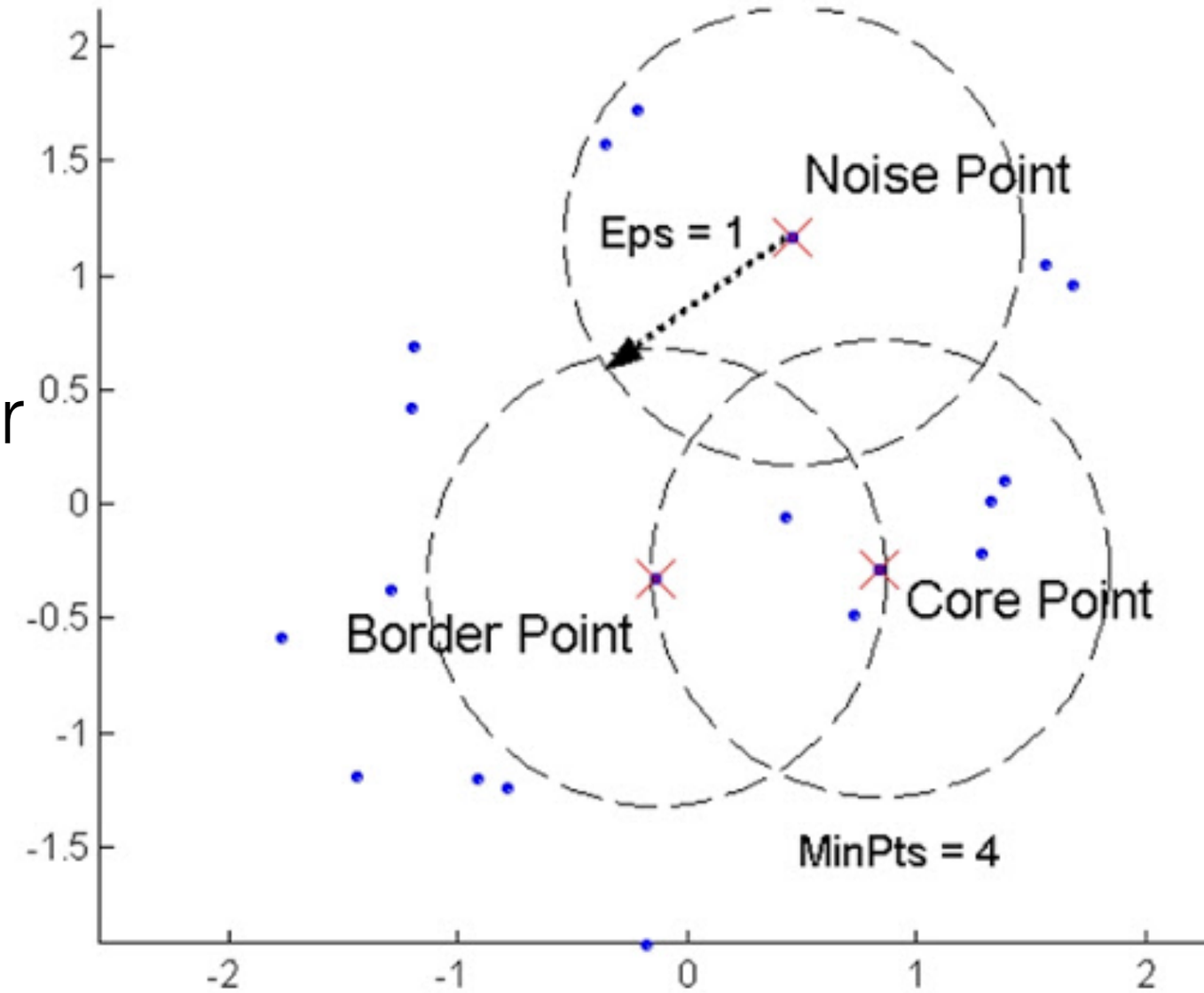


DBSCAN

- Parameters
  - radius **(Eps)**
  - **$N_{Eps}(p)$**: subset contained in the Eps neighborhood of p.
  - minimum number of objects **(MinPts)**

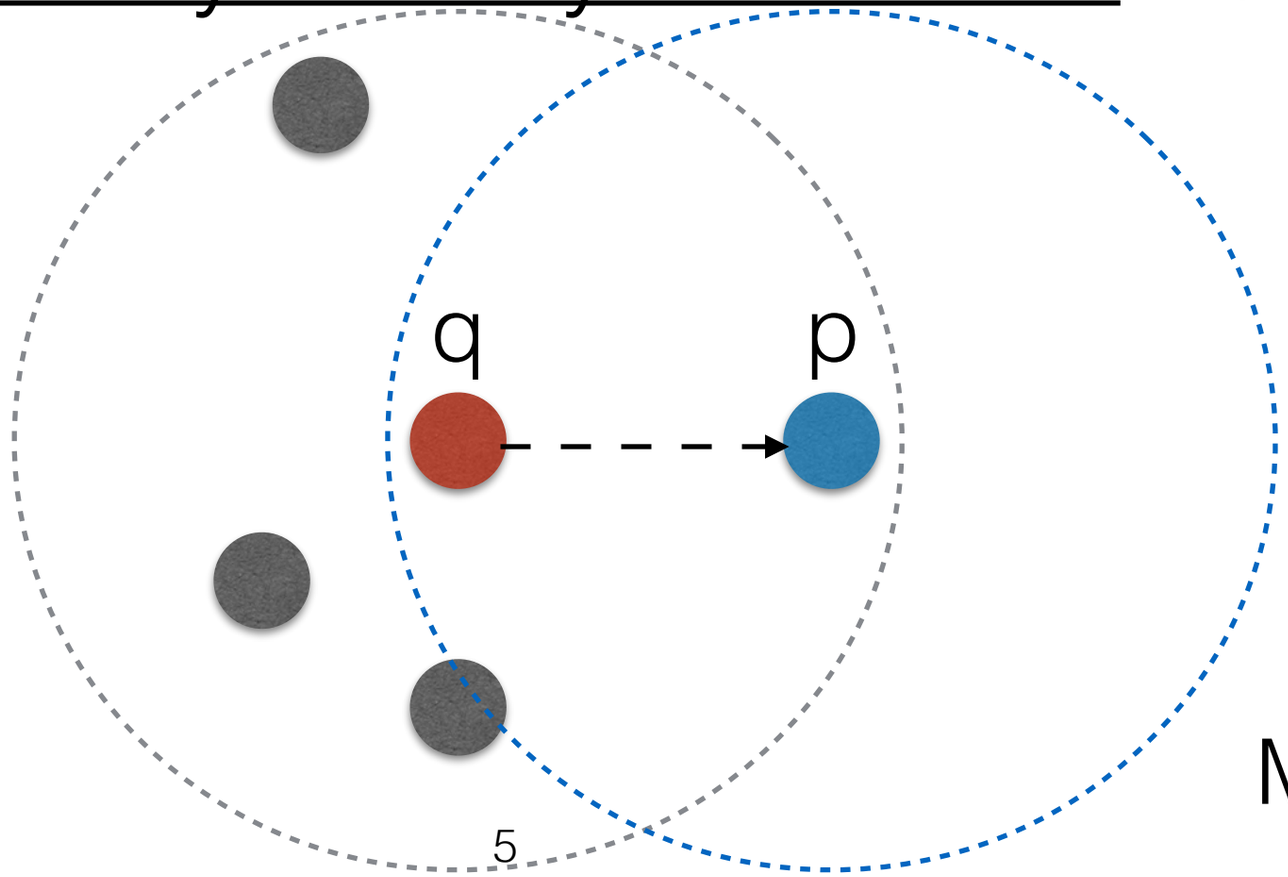


- Types of object in the Dataset
  - Core object
    - $|N_{Eps}(p)| > MinPts$
  - Border object
    - Core obj's neighbor but not a core obj
- Noise object
  - other



Definitions

- **Definition 1**: Directly density-reachable
  - If **p** ∈ N_{Eps}(**q**), and **q** is a core object
  - **p** is <u>directly density-reachable</u> from **q**



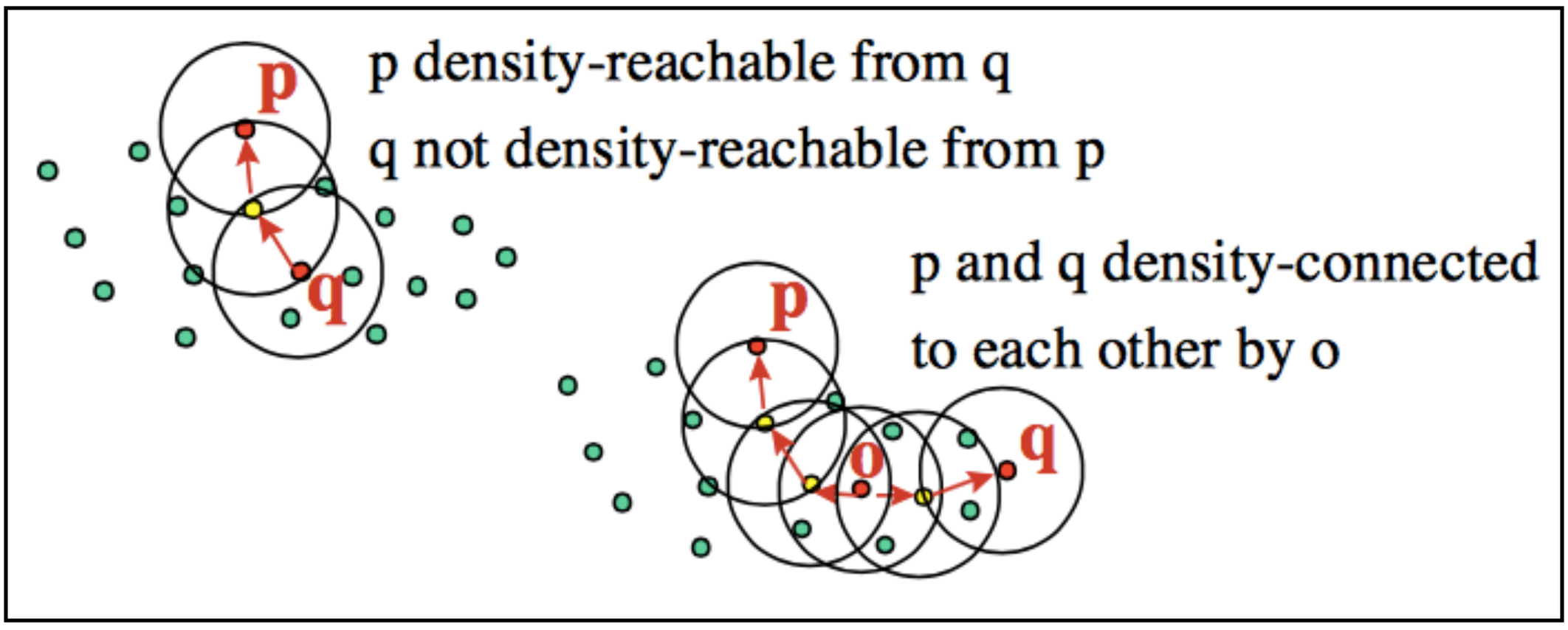
- **Definition 2**: Density-reachable

  - if there's a chain of object $\mathbf{p_1, \ldots, p_n}$, and p_{i+1} is directly density-reachable from p_i

  - $\mathbf{p_1 = q}$ and $\mathbf{p_n = p}$, we say **p** is density-reachable from **q** (denoted as $\mathbf{p >_D q}$).

- **Definition 3**: Density-connected

  - if both **p** and **q** are density-reachable from **o**, **p** is density-connected to **q**.


p density-reachable from q
q not density-reachable from p
p
q
p and q density-connected
to each other by o
p
o
q

- **Definition 4**: Cluster

  - Maximality:
    ∀**p**,**q** ∈D(a set of object):
    If **p** ∈ **cluster C** and **q** > **$_D$p**, then also **q** ∈**C**.

  - Connectivity:
    ∀p,q ∈ C: **p** is density-connected to **q**.

Algorithm

Algorithm *DBSCAN*(Data: $\mathcal{D}$, Radius: *Eps*, Density: τ)
begin
  Determine core, border and noise points of $\mathcal{D}$ at level (Eps, τ);
  Create graph in which core points are connected
    if they are within *Eps* of one another;
  Determine connected components in graph;
  Assign each border point to connected component
    with which it is best connected;
  return points in each connected component as a cluster;
end

Figure 6.15: Basic *DBSCAN* algorithm

Basic DBSCAN

平行化方法

- 主要在兩處做平行化

Algorithm *DBSCAN*(Data: $\mathcal{D}$, Radius: *Eps*, Density: τ)
begin
  Determine core, border and noise points of $\mathcal{D}$ at level (Eps, τ);
  Create graph in which core points are connected
    if they are within *Eps* of one another;
  Determine connected components in graph;
  Assign each border point to connected component
    with which it is best connected;
  return points in each connected component as a cluster;
end

Figure 6.15: Basic *DBSCAN* algorithm

平行化的DBSCAN

第一部分

- Data partition.

- 在把資料點分成core, broader, noise時，把資料切割分別做分類。

第二部分

- 原本使用BFS把同個cluster的core object都串起來。
- 平行化後變成每個thread都去做BFS。
- 原本應該同一個cluster的資料會被瓜分成多個。

處理方式

- 在global宣告一個equivalent matrix
  - 當兩個thread在trace同一個cluster，然後撞在一起時會去紀錄他們了個標的其實是同一個cluster。
- sync

	A	B	C	D
A		1		1
B	1			1
C				
D	1	1		

- 對equivalent matrix做trace，做出一個矩陣，最後統一全部資料點的cluster id時使用。

  - 最後在把全部點讀過一遍，照著表格更改原本的cluster ID。

A	B	C	D	E	F
A	A	C	C	C	F

實驗結果

- 資料量: 8000筆
- serial program時間: 6.281s

thread數	1	2	4	8	12
時間	6.422	3.459	1.989	1.179	0.911